Capítulo 2

Anotación de vídeo

Normalmente, el estudio del movimiento en secuencias de imágenes implica una división entre el proceso de detección del movimiento y el proceso de extracción de información. Principalmente, esta división es debida a la necesidad de información contextual para realizar una descripción de alto nivel de la escena. En este capítulo se propone la utilización de la información del contexto de la aplicación para resolver todas las fases del análisis visual de secuencias de imágenes. Para demostrarlo, se presenta la resolución de una aplicación de anotación de vídeo cuyo objetivo básico es el seguimiento de jugadores de fútbol durante un partido. Para solucionar la aplicación se divide el problema en tres partes: localización, calibración y seguimiento visual. Se mostrará como estos procesos se solucionan correctamente con la ayuda de la información de la escena.

2.1 Introducción.

Una secuencia es una serie de imágenes ordenadas en el tiempo. El primer término significativo que encontramos en la definición de secuencia es que es una **serie** de imágenes, esto implica el que cómo mínimo debemos tener dos imágenes. Así, el estudio de secuencias permite acceder a un tipo de información que no está disponible en las imágenes estáticas: el **movimiento**. La información que proporciona el movimiento juega un papel importante en nuestro sistema visual, por ejemplo, un foco de atención típico es la detección de movimiento.

Una de las primeras clasificaciones que se hicieron del estudio del movimiento desde el punto de vista de la visión por computador es debida a Marr, que proponía lo siguiente[55]:

> *"...el problema consta de dos partes fundamentales: ¿cómo se obtienen las medidas de los cambios producidos por el movimiento? y ¿cómo se usa esta información? Ninguna de estas preguntas es fácil de contestar y quizá debido a que la primera es tan difícil, la segunda es en cierta medida un estudio de la información mínima necesaria que la primera parte debe ofrecer para que los cálculos posteriores ofrezcan algún tipo de resultado útil."*

Es decir, Marr postulaba que para poder extraer una información útil del movimiento, éste debe ser medido anteriormente de alguna forma. Durante los años posteriores los esfuerzos se centraron en la búsqueda de medidas de movimiento, siendo la más extendida el flujo óptico, que es el movimiento aparente[1] de los píxels de la imagen[4].

Posteriormente, se extiende la clasificación de Marr y se divide el estudio de secuencias en tres fases: Periférica, Atentiva y Cognitiva[47]. En la fase Periférica se detecta el movimiento, decidiendo las partes de la escena que pasan a la fase Atentiva. En ésta es donde se extrae la información del movimiento que utiliza la fase Cognitiva, junto al conocimiento que se posee, para realizar el análisis de la escena. Si se compara esta clasificación con la realizada por Marr se observa una división de la búsqueda de la información de movimiento en dos fases: la Periférica y la Atentiva. El paso dado es en el sentido de analizar la información del movimiento localmente, es decir, después de un proceso de bajo nivel de detección de movimiento, se analizan en profundidad las partes de la imagen donde se produce el movimiento.

Pero, ¿qué ocurre con la segunda pregunta formulada por Marr? ¿Para qué podemos utilizar la información que proporciona el movimiento? Una posibilidad es realizar aplicaciones de **anotación de vídeo**. La anotación de vídeo se define como la generación de descripciones de secuencias que pueden ser utilizadas para indexar, recuperar y resumir el contenido de la secuencia. En la mayoría de aplicaciones, para poder realizar estas descripciones es necesario conocer previamente el contexto de la escena. Así, es posible describir las relaciones de los objetos que se mueven, entre ellos y con los objetos pertenecientes a la escena.

Además, la información que nos proporciona el contexto se puede utilizar para facilitar el desarrollo de la aplicación. La idea básica es utilizar esta información contextual para crear algoritmos dependientes de la aplicación. Estos algoritmos son más sencillos y funcionan mejor que los genéricos en la aplicación para la que han sido desarrollados.

Para comprobar esta relación mutua entre el problema de la anotación de vídeo y el contexto donde se desarrolla la aplicación, en este capítulo se presenta una aplicación de anotación de vídeo en secuencias de partidos de fútbol. Básicamente, el objetivo es conocer la posición de los jugadores en todo instante de tiempo.

[1] Decimos movimiento aparente porque es el movimiento que podemos captar con un sistema de adquisición, y que en algunos casos no se corresponderá con el movimiento real.

Desde el punto de vista aplicado, un sistema capaz de seguir las posiciones de los jugadores durante el partido sería de gran interés para una reconstrucción virtual[2] del encuentro, y para los analistas de este deporte (entrenadores o periodistas), que podrían estudiar posteriormente todo lo ocurrido en el encuentro.

Finalmente, desde el punto de vista social, está claro que el fútbol es el deporte más popular del mundo, como ha quedado demostrado en los campeonatos mundiales, donde dispone del apoyo total de los medios de comunicación, las empresas y los millones de aficionados de todo el mundo.

2.2 Utilización de información contextual.

Los algoritmos de visión dependientes del contexto son herramientas importantes de análisis de imágenes en dominios de aplicación como demuestra el trabajo de Strat y Fischler[83]. El dominio de la aplicación es una forma de abordar el problema del seguimiento de múltiples objetos como demuestran los trabajos de Intille[39, 40]. En estos trabajos el conocimiento del entorno facilita la tarea del seguimiento visual.

En la mayoría de trabajos previos se asume que los objetos de interés pueden definirse a partir de un conjunto reducido de modelos. También se asume que todos los objetos de interés poseen características que pueden medirse. En dominios complejos, por ejemplo, entornos naturales pertenecientes al mundo real, estas dos asunciones no se cumplen, provocando que el algoritmo desarrollado no funcione, o que su rendimiento descienda de forma dramática.

Si se conoce la descripción completa del escenario de la aplicación, la misión de los actores y el contexto, es posible asumir un conjunto de restricciones que facilitarán el desarrollo del objetivo de la aplicación. Por ejemplo, una condición típica es la forma del objeto. Sin embargo, existen aplicaciones donde la forma del objeto puede variar de forma brusca. En estos casos deben utilizarse otras características invariantes en el escenario de la aplicación. Otra característica importante es el color. Se puede realizar un modelo de color de la escena para localizar y seguir los objetos de interés[90].

La característica de color puede ser suficiente cuando sólo hay un objeto a seguir. En el caso de seguimiento de múltiples objetos no es suficiente localizar los objetos debido a las oclusiones entre objetos. En estos casos es necesario la utilización de un filtro de estimación y técnicas estadísticas de asociación de datos[96].

Conociendo el dominio de la aplicación, la información obtenida por el seguimiento visual se puede completar con un proceso razonamiento de alto nivel para mejorar los resultados. Por ejemplo, si estamos siguiendo una persona y desaparece de la escena de forma repentina, debería ser porque se ha producido un error de localización, no

[2]Una descripción virtual sería gráfica, y dispondría de libertad de cámara para seguir el partido desde cualquier posición.

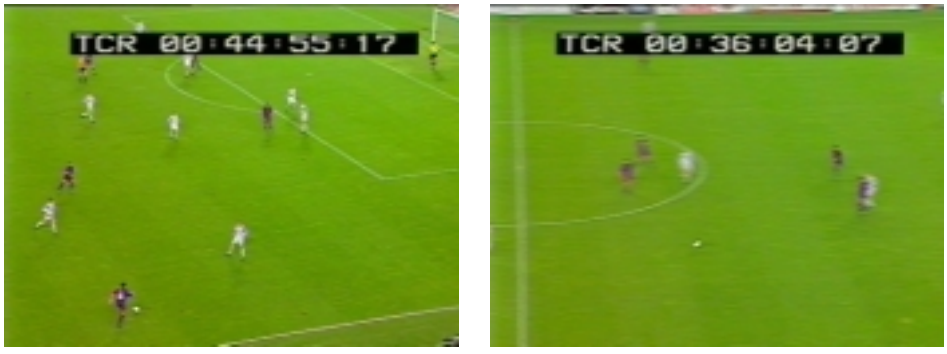
porque la persona haya desparecido realmente.

En nuestro caso hemos escogido como dominio de aplicación un partido de fútbol. Desde el punto de vista de investigación, un partido de fútbol es un escenario que permite el estudio de algoritmos de seguimiento de múltiples objetos complejos, como los jugadores.

2.3 Repetición virtual de jugadas de fútbol.

A partir de secuencias de vídeo de jugadas de un partido de fútbol, la aplicación que se desea resolver es generar una reconstrucción de la jugada. Esta reconstrucción virtual permite la generación de repeticiones de diferentes jugadas de un partido desde puntos de vista que serían imposibles de conseguir con cámaras convencionales. Para cumplir este objetivo se han de tener localizados los jugadores en todo instante de tiempo. Por tanto, nuestro objetivo es realizar un algoritmo de seguimiento visual que mantenga las trayectorias de los jugadores.

Una secuencia de la repetición de una jugada tiene una duración media de 5 a 10 segundos. La frecuencia de adquisición es de 25 imágenes por segundo, lo que implica de 125 a 250 imágenes en color, ejemplos de imágenes con las que hemos trabajado se muestran en la Fig. 2.1. Estas imágenes son digitalizadas a partir de la grabación en vídeo de cámaras utilizadas para la retransmisión del partido por la televisión. Estas cámaras poseen 3 grados de libertad: pan (movimiento horizontal), tilt (movimiento vertical) y zoom. También es importante resaltar que son cámaras entrelazadas, es decir, que primero se adquieren las líneas impares y después las pares. El efecto del entrelazado se nota especialmente cuando la cámara se mueve, ver Fig. 2.1 (a). Por este motivo se elimina un campo de cada imagen, quedando el tamaño final de las imágenes de 720 $\times$ 252 píxels. Los objetos a seguir serán los jugadores de fútbol, es decir, que el algoritmo ha de tener en cuenta que son objetos no rígidos y que su forma es dependiente de la vista.



(a) Imagen normal. (b) Efecto del entrelazado en una imagen.

Figura 2.1: Imágenes típicas de un partido de fútbol.

El problema de utilizar una cámara dinámica, de la cual no es posible conocer sus parámetros, es que no podemos utilizar el movimiento de forma directa para saber la posición de los objetos. En este caso, es posible usar diferentes técnicas para realizar el seguimiento visual. El método más utilizado en estos casos es el de *Template Matching*[89], pero no funciona correctamente cuando la apariencia del objeto cambia durante el tiempo de seguimiento. Un ejemplo es el caso del seguimiento de personas, debido a los posibles cambios de postura de la persona. Una posible solución a este problema son las plantillas adaptativas[78]. La forma más sencilla de ajuste de las plantillas es definir una nueva plantilla después de cada correspondencia. Si la escena es homogénea el método funciona correctamente, pero si es muy variable puede fallar y perder la plantilla. Si en lugar de la apariencia de los objetos, se escoge la forma como característica, es posible utilizar modelos deformables, *snakes*[12]. El problema de estos modelos es que requieren una buena inicialización, lo cual es difícil de conseguir en muchas aplicaciones.

Basados en el conocimiento del contexto de la aplicación encontramos el proyecto *Kids-room*[40]. En esta aplicación se define el concepto de *mundo cerrado* como una región espacio-temporal en la cual todos los objetos presentes son conocidos. Utilizando este concepto, Intille y Bobick en [39] resuelven una aplicación de anotación de vídeo en el entorno de un partido de fútbol americano. Sin embargo, sólo utilizan la información contextual en la definición de las plantillas de los jugadores.

Nuestro método es una amplicación del método basado en mundos cerrados para cambiar el planteamiento del problema del seguimiento visual. Utilizaremos toda la información posible del dominio de la aplicación para transformar el problema del seguimiento visual en un problema de correspondencia de movimiento. Es decir, en primer lugar, se localizarán los jugadores en la imagen, a continuación se trasladará la posición en la imagen de los jugadores al campo de juego real para conocer su posición 3D. Finalmente, se realizará una correspondencia entre las posiciones de los jugadores en tiempo $t-1$ y las posiciones de los jugadores encontrados en tiempo t. Como veremos más adelante, el proceso de localización puede contener errores, esto provocará problemas en la etapa de correspondencias. Para poder corregir estos errores se utilizará un esquema sencillo de seguimiento visual, basado en una etapa de predicción utilizando un conocido filtro de estimación y una fase de corrección a partir de las correspondencias encontradas.

2.3.1 Localización.

La primera parte de la aplicación donde se utiliza el conocimiento del contexto es en la localización de los jugadores. En el caso de un partido de fútbol se puede asumir que el terreno de juego tiene un color uniforme. Por tanto, es posible realizar un modelo estadístico del color del campo. Utilizaremos este modelo para localizar los píxels pertenecientes a los jugadores simplemente calculando la distancia del valor de color de cada píxel, $\boldsymbol{I}$, al modelo del campo.

Modelizaremos de forma probabilística el color del fondo con una distribución Normal de tres dimensiones. Es decir, la probabilidad de que el valor de un píxel pertenezca al terreno de juego es:

$$p(\boldsymbol{I}) = \frac{1}{(2\pi)^{\frac{3}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(\boldsymbol{I}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{I}-\boldsymbol{\mu})\right\} \ , \tag{2.1}$$

donde la media, $\boldsymbol{\mu}$, es un vector de 3 dimensiones, $\boldsymbol{\Sigma}$ es la matriz de covarianza 3×3, y $|\boldsymbol{\Sigma}|$ es el determinante de $\boldsymbol{\Sigma}$. Para encontrar los parámetros de la distribución, media y matriz de covarianza, se escogerán como muestras los valores de los píxels de regiones de la imagen donde no aparecen jugadores. A partir de los valores de las muestras, $\boldsymbol{I}_n$, donde $n = \{1, \ldots, N\}$, utilizaremos el método de *maximum likelihood*[6] para estimar la media y la matriz de covarianza:

$$\hat{\boldsymbol{\mu}} = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{I}_n \ , \tag{2.2}$$

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{N}\sum_{n=1}^{N}(\boldsymbol{I}_n-\hat{\boldsymbol{\mu}})(\boldsymbol{I}_n-\hat{\boldsymbol{\mu}})^T \ , \tag{2.3}$$

donde N es el número de muestras. La cantidad:

$$\Delta^2 = (\boldsymbol{I}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{I}-\boldsymbol{\mu}) \ , \tag{2.4}$$

que aparece en el exponente de la Ec. (2.1), es la distancia de *Mahalanobis* de $\boldsymbol{I}$ a $\boldsymbol{\mu}$. Utilizaremos esta distancia para clasificar los píxels entre fondo y jugadores. En la Fig. 2.2 se muestra el resultado después del proceso de clasificación.

Figura 2.2: Resultado de aplicar la distancia de *Mahalanobis* para localizar los píxels que no pertenecen al fondo.

A continuación, se binariza la imagen aplicando un umbral sobre la distancia de *Mahalanobis* y se aplican operadores de morfología matemática[79] para eliminar las líneas del campo. Finalmente, el proceso de localización realiza una clasificación de los valores de los píxels etiquetados como jugadores para identificar el equipo al que pertenece cada jugador, ver Fig. 2.3.

Figura 2.3: Localización final.

2.3.2 Calibración.

Después de la fase de localización, se conoce la posición de los jugadores en la imagen. Si se tuviera una vista ortográfica del campo, por ejemplo, con una cámara cenital, podríamos saber la posición de los jugadores en el terreno de juego utilizando sólo un parámetro de escala. Sin embargo, como se ha comentado anteriormente, la cámara de televisión posee tres grados de libertad: pan, tilt y zoom. Es necesario un proceso de calibración para conocer la posición de los jugadores sobre el terreno de juego. Además, los tres grados de libertad pueden cambiar en cada imagen, por tanto, es necesario repetir el proceso de calibración en cada paso.

La dificultad es que no es posible disponer de los valores de los parámetros de la cámara. Para solucionar este problema, utilizamos de nuevo la información del contexto. Se puede asumir que los jugadores están sobre el terreno de juego, que es un plano en la escena 3D. A partir de esta asunción podemos obtener la transformación proyectiva con una homografía[23]. Una homografía es una transformación proyectiva de un plano 2D, el terreno de juego, a otro plano 2D, la imagen. En primer lugar definirimos la notación que utilizaremos, basada en la teoría de calibración de cámaras[97]. Veremos la relación entre un proceso normal de calibración y como la restricción definida por el contexto lo convierte en una homografía.

Un punto 2D de la imagen lo notaremos como $\boldsymbol{p}_{im} = (x_{im}, y_{im})^T$. Un punto 3D del mundo lo notaremos como $\boldsymbol{p}_{wo} = (x_{wo}, y_{wo}, z_{wo})^T$. Utilizaremos la representación en coordenadas homogéneas para poder definir la transformación de un punto del mundo 3D a un punto de la imagen 2D, por tanto, añadiremos un 1 como último elemento de las coordenadas de ambos puntos, $\tilde{\boldsymbol{p}}_{im} = (x_{im}, y_{im}, 1)^T$ y $\tilde{\boldsymbol{p}}_{wo} = (x_{wo}, y_{wo}, z_{wo}, 1)^T$. Utilizando el modelo de cámara puntual, la relación entre un punto 3D, $\boldsymbol{p}_{wo}$, y su proyección en la imagen, $\boldsymbol{p}_{im}$ viene dada por:

$$s\tilde{\boldsymbol{p}}_{im} = \mathbf{C}[\mathbf{R} \quad \mathbf{t}]\tilde{\boldsymbol{p}}_{wo} \ , \tag{2.5}$$

donde s es un factor de escala arbitrario, la matriz de parámetros *extrínsecos*, $[\mathbf{R} \quad \mathbf{t}]$, es una matriz 3×4 de cambio de origen de coordenadas, es decir, la rotación y la traslación que relaciona el sistema de coordenadas del mundo con el sistema de coordenadas de la cámara, y $\mathbf{C}$, es la matriz de parámetros *intrínsecos* de la cámara, que

define la proyección del punto 3D en coordenadas cámara a un punto 2D en coordenadas del plano imagen.

Como hemos comentado anteriormente, por la información contextual sabemos que los jugadores están sobre el plano descrito por el terreno de juego, por tanto, podemos asumir que $z_{wo} = 0$. Si notamos la i-ésima columna de la matriz $\mathbf{R}$ por $\boldsymbol{r}_i$, a partir de la Ec. (2.5) tenemos:

$$\begin{aligned} s\begin{pmatrix} x_{im} \\ y_{im} \\ 1 \end{pmatrix} &= \mathbf{C}[\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{r}_3 \quad \mathbf{t}]\begin{pmatrix} x_{wo} \\ y_{wo} \\ 0 \\ 1 \end{pmatrix} \\ &= \mathbf{C}[\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{t}]\begin{pmatrix} x_{wo} \\ y_{wo} \\ 1 \end{pmatrix} . \end{aligned} \tag{2.6}$$

Por lo tanto, consideraremos que un punto del mundo tendrá sólo dos coordenadas, $\boldsymbol{p}_{wo} = (x_{wo}, y_{wo})^T$, y su representación en coordenadas homogéneas quedará como $\tilde{\boldsymbol{p}}_{wo} = (x_{wo}, y_{wo}, 1)^T$. De esta forma, llegamos a la expresión final:

$$s\tilde{\boldsymbol{p}}_{im} = \mathbf{H}\tilde{\boldsymbol{p}}_{wo} \ , \tag{2.7}$$

donde:

$$\mathbf{H} = \mathbf{C}[\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{t}] \ , \tag{2.8}$$

es la matriz de **homografía**, definida excepto por un factor de escala, con lo cual es posible establecer $h_{22} = 1$ y la matriz buscada tiene la forma:

$$\mathbf{H} = \begin{pmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & 1 \end{pmatrix} . \tag{2.9}$$

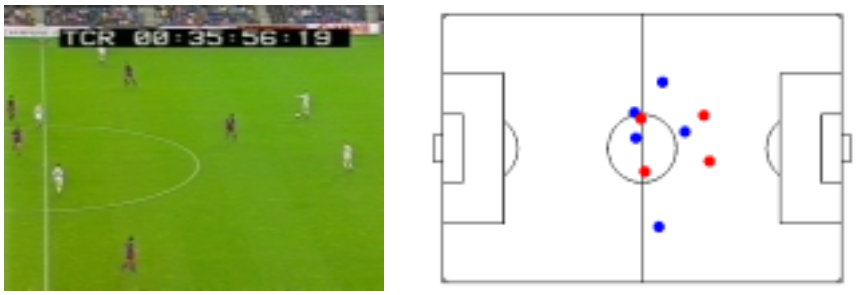
Desarrollando la Ec. (2.7) se obtiene:

$$x_{im} = \frac{h_{00}x_{wo} + h_{01}y_{wo} + h_{02}}{h_{20}x_{wo} + h_{21}y_{wo} + 1} \ , \tag{2.10}$$

$$y_{im} = \frac{h_{10}x_{wo} + h_{11}y_{wo} + h_{12}}{h_{20}x_{wo} + h_{21}y_{wo} + 1} \ . \tag{2.11}$$

La matriz de transformación, $\mathbf{H}$, tiene 8 grados de libertad. Cada correspondencia conocida entre un punto de la imagen y un punto de la escena, proporciona dos ecuaciones, Ec. (2.10) y Ec. (2.11). Por tanto, para calcular $\mathbf{H}$ se necesitan como mínimo 4 correspondencias. Las características conocidas que utilizamos para realizar la correspondencia pertenecen al contexto de la aplicación, por ejemplo, a las líneas del

campo. Cuando las características son visibles y tres de ellas no son colineales es posible encontrar la transformación de forma directa. Si tenemos más de 4 puntos se busca la mejor solución con un método de mínimos cuadrados. En la Fig. 2.4 se muestran los resultados de la calibración.



(a) Imagen original. (b) Posición de los jugadores.

Figura 2.4: Resultados del proceso de calibración.

El problema aparece en imágenes que no tienen un número suficiente de características visibles. En este caso no es posible calcular la transformación proyectiva de forma cerrada. Sin embargo, en un partido de fútbol, en pocas ocasiones ocurre este problema. Para solucionarlo, se puede realizar una interpolación temporal de las matrices de transformación.

Una vez hallada la matriz de transformación, $\mathbf{H}$, se calcula su inversa para encontrar la posición de los jugadores en el campo a partir de su posición en la imagen:

$$\tilde{\boldsymbol{p}}_{wo} = \mathbf{H}^{-1} s \tilde{\boldsymbol{p}}_{im} \quad . \tag{2.12}$$

El punto de la región del jugador en la imagen, $\boldsymbol{p}_{im}$, utilizado para transformar sus coordenadas a la escena 3D, es la posición del centro de gravedad en el eje X y su extremo inferior en el eje Y, ya que el proceso de calibración asume que los jugadores están en el plano 3D del terreno de juego, ver Fig. 2.5.

Figura 2.5: Punto de la imagen correspondiente a la localización del jugador que se proyecta a la escena 3D.

2.3.3 Seguimiento visual.

El objetivo de la etapa de seguimiento visual es mantener las trayectorias de las posiciones de los jugadores aunque ocurran errores en alguna de las fases anteriores. Es posible asumir que el movimiento de los jugadores sigue un modelo físico, por tanto, saber cuando hay un error porque la trayectoria de un jugador se desvía excesivamente de su modelo de movimiento.

Como se ha comentado en la sección anterior, la posición de un jugador sobre el terreno de juego se representará por medio de un punto. Para la etapa de seguimiento visual, caracterizaremos la serie temporal de posiciones del jugador con la variable $\boldsymbol{x}_t$, que se denomina vector de **estado**, en nuestro caso contendrá la posición del jugador en el campo en tiempo t.

El movimiento de un jugador se modela a partir de la siguiente expresión:

$$\boldsymbol{x}_t = \mathbf{f}_{t,t-1}(\boldsymbol{x}_{t-1}) + \boldsymbol{w}_t \ , \tag{2.13}$$

donde $\mathbf{f}_{t,t-1}(\cdot)$ es una función vectorial que describe la transición del vector de estado del jugador del instante de tiempo $t-1$ al instante t. Asumiremos como función de transición de estado para un jugador un modelo de velocidad constante:

$$\mathbf{f}_{t,t-1}(\boldsymbol{x}_{t-1}) = \mathbf{A}\boldsymbol{x}_{t-1} \ , \tag{2.14}$$

donde:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{x}_t = \begin{pmatrix} x \\ y \\ v_x \\ v_y \end{pmatrix} \ , \tag{2.15}$$

y T es el tiempo de adquisición de cada imagen, en nuestra aplicación $T = 40$ms. Es razonable la asunción de este modelo porque tenemos una frecuencia alta de adquisición, así la componente de aceleración queda incluída en la variable aleatoria del sistema, $\boldsymbol{w}_t$. Las propiedades estadísticas de esta variable no es posible conocerlas de forma exacta. En la práctica, se asume una distribución Normal de media cero y matriz de covarianza conocida, es decir:

$$E[\boldsymbol{w}_t] = 0 \quad \text{, y} \quad E[\boldsymbol{w}_t \boldsymbol{w}_l] = \mathbf{Q}_t \tag{2.16}$$

donde $\mathbf{Q}_t$ suele ser diagonal y se determina a partir del conocimiento del error que es posible cometer con el modelo de movimiento.

La variable de estado es una estimación del verdadero valor de la posición de un jugador ya que este valor no lo conocemos. En su lugar disponemos de una **medida**,

el punto localizado en la imagen y transformado a la escena 3D por medio de la inversa de la matriz de calibración:

$$\tilde{\boldsymbol{p}}_{wo} = \mathbf{H}^{-1} s \tilde{\boldsymbol{p}}_{im} \quad . \tag{2.17}$$

Estos puntos los denotaremos como las medidas observables de nuestro sistema, $\boldsymbol{z}_t \equiv \boldsymbol{p}_{wo}$. Para completar la descripción del sistema de seguimiento, relacionaremos el estado de cada jugador con las medidas obtenidas en la imagen por medio de la ecuación de medida:

$$\boldsymbol{z}_t = \mathbf{h}_t(\boldsymbol{x}_t) + \boldsymbol{n}_t \quad , \tag{2.18}$$

donde $\mathbf{h}_t(\cdot)$ es la función que relaciona las observaciones con el estado, que en nuestro caso es igual a la identidad, $\mathbf{h}_t(\boldsymbol{x}_t) = \boldsymbol{x}_t$. $\boldsymbol{n}_t$ es el ruido de medida, y también se asume aleatorio con distribución de probabilidad Normal de media cero y matriz de covarianza diagonal, $\mathbf{R}_t$.

Dada una secuencia de medidas, $\mathcal{Z}_t = (\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_t)$, de la posición de un jugador, el Filtro de Kalman[49] permite calcular de forma óptima una estimación $\hat{\boldsymbol{x}}_t$ del verdadero estado del jugador de forma recursiva a partir de las medidas obtenidas en la imagen. Es posible utilizar el Filtro de Kalman porque en nuestra aplicación las funciones $\mathbf{f}_{t,t-1}(\cdot)$ y $\mathbf{h}_t(\cdot)$ son lineales y porque las distribuciones de las componentes aleatorias son Normales.

Para calcular la estimación de la verdadera posición de un jugador el Filtro de Kalman utiliza un esquema basado en dos pasos: predicción y corrección. En primer lugar se utiliza el modelo de movimiento para predecir la nueva posición que ocupará el jugador. A continuación, se utiliza la medida obtenida por los módulos de localización y calibración para corregir la predicción, de esta manera si el jugador se para o cambia su dirección, se corrige la predicción de la posición. Otra ventaja de la utilización de un filtro de estimación es que si no se obtiene una medida debida a un error en alguno de los procesos anteriores podemos utilizar la predicción para dar la estimación más probable del nuevo estado del jugador.

Sin embargo, debido al modelo de movimiento que hemos escogido en nuestra aplicación, utilizaremos una simplicación del Filtro de Kalman. Es decir, en lugar de aplicar las ecuaciones de predicción y de corrección del Filtro de Kalman, al asumir un modelo de velocidad constante, se puede demostrar[22] que se reducen al conocido Filtro $\alpha - \beta$:

$$\hat{\boldsymbol{x}}_t = -(\alpha + \beta - 2) \cdot \hat{\boldsymbol{x}}_{t-1} - (1 - \alpha) \cdot \hat{\boldsymbol{x}}_{t-2} + \alpha \cdot \boldsymbol{z}_t + (-\alpha + \beta) \cdot \boldsymbol{z}_{t-1} \quad , \tag{2.19}$$

donde: $\alpha \in (0, 1)$ y $\beta \in (0, \frac{\alpha^2}{1-\alpha})$.

Con el Filtro $\alpha - \beta$, podemos seguir la trayectoria de un jugador. Para realizar el seguimiento de todos los jugadores debemos analizar los siguientes casos particulares:

- Aparición: un jugador que no estaba siendo seguido aparece totalmente en la imagen.
- Desaparición: un jugador sale parcial o totalmente de la imagen.
- Oclusión: un jugador no se localiza en la imagen porque está ocluído por otro jugador.
- Ausencia: un jugador no se localiza en la imagen debido a algún error en el proceso de localización.

Para tratar estos casos particulares utilizaremos el proceso de predicción y corrección. En tiempo t se realiza la predicción de la posición del jugador en la escena utilizando el modelo dinámico, Ec. (2.14). Por otro lado, obtenemos las medidas aplicando la matriz de calibración a los puntos obtenidos por el modulo de localización. A continuación realizaremos el proceso de **asociación de datos**. Este proceso consiste en encontrar la correspondencia entre las posiciones medidas a partir de la imagen con las posiciones predecidas por el modelo dinámico. Si sólo se encuentra una correspondencia entre la medida y la predicción de un jugador, se corrige su estado con la medida asociada, Ec. (2.19).

Después de este primer proceso podemos encontrarnos con medidas o predicciones sin correspondencia. Las medidas sin correspondencias se asocian con el caso de la aparición. Para asegurar que no es un error del proceso de localización se comprueba la condición impuesta por la aplicación de que un jugador sólo puede aparecer por los extremos de la imagen. Si cumple esta condición se considera como nuevo objeto y se inicializa un nuevo Filtro $\alpha - \beta$.

Si no encontramos ninguna correspondencia para una predicción es debido a uno de los casos de desaparición, oclusión o ausencia. Para determinar si es una desaparición se proyecta la predicción de la nueva posición a la imagen para comprobar si sale fuera de ella. El problema de la oclusión provoca que sólo exista una medida para múltiples jugadores. Por tanto, este caso se detectará en la imagen, antes de realizar el paso de calibración. Una vez comprobado en la imagen, se utiliza la misma medida para actualizar la posición de los jugadores implicados en la oclusión. Si no hay desaparición u oclusión, entonces nos encontramos ante el caso de ausencia y se actualiza el estado con el valor de la predicción.

En la Fig. 2.6 se muestra el rendimiento del algoritmo de seguimiento visual con una secuencia completa de una jugada. Se puede comprobar en las imágenes de la secuencia que los jugadores son seguidos correctamente.

Imagen 0, $t = 0s$.

Imagen 10, $t = 0.4s$.

Imagen 20, $t = 0.8s$.

Imagen 30, $t = 1.2s$.

Imagen 40, $t = 1.6s$.

Imagen 50, $t = 2s$.

Imagen 70, $t = 2.8s$.

Imagen 90, $t = 3.6s$.

Imagen 100, $t = 4s$.

Imagen 125, $t = 5s$.

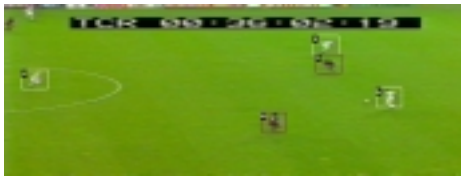


Imagen 150, $t = 6s$.

Imagen 175, $t = 7s$.

Figura 2.6: Secuencia resultado del proceso de seguimiento visual de jugadores de fútbol.

2.3.4 Repetición virtual.

Finalmente, se utiliza la información del seguimiento visual para reconstruir la jugada en la escena 3D. En primer lugar se construye una escena virtual con las dimensiones reales del terreno de juego. Como se conoce la posición 3D en la escena de cada jugador en todo instante de tiempo es posible situarlos en el escenario virtual. La ventaja de tener situados a los jugadores en la escena virtual es que es posible generar la repetición de la jugada y observarla desde cualquier ángulo de vista. En la Fig. 2.7 se muestran diferentes imágenes virtuales de la repetición de la jugada.

Figura 2.7: Imágenes de la repetición virtual.

2.3.5 Simulación.

A partir de los datos extraídos por nuestro sistema, posiciones de los jugadores en todo instante de tiempo, es posible realizar más aplicaciones, por ejemplo, las estadísticas completas del rendimiento de los jugadores. Estas estadísticas se pueden utilizar como datos de ejemplo para la realización de un programa de simulación de partidos de fútbol.

Se ha realizado un estudio previo para modelar un jugador de fútbol como un agente inteligente [56] y un equipo como un sistema multiagente[18]. La aplicación final utilizaría los datos del sistema de visión para establecer los parámetros de los jugadores de forma automática.

2.4 Discusión.

Con la resolución de esta aplicación de anotación de vídeo se ha mostrado como la información contextual puede ofrecer una vía de resolución de problemas de análisis de secuencias de imágenes. Se ha descrito como los procesos visuales necesarios para resolver la aplicación: localización, calibración y seguimiento visual, se realizan de forma computacionalmente eficiente con la ayuda de la información de la escena.

La desventaja de la utilización de la información contextual es que el trabajo desarrollado no es posible reutilizarlo en otras aplicaciones diferentes. Este hecho es debido a que la resolución en mundos cerrados acaba siendo un estudio de los casos particulares que tiene la aplicación.

La conclusión final es que la solución ideal sería la definición de un método genérico de seguimiento visual que sea que sea fácil de adaptar al contexto de la aplicación. Para conseguir definir este método, en el siguiente capítulo se repasa la teoría clásica del seguimiento visual desde un punto de vista probabilístico. El objetivo es establecer un marco teórico para presentar a continuación un nuevo método de seguimiento visual que posea las características deseadas.