

# Análisis de Video en Biomecánica

Andréi Guchin, Gonzalo Pereira,  
Guillermo Ottado, Mauricio Ramos, and Juan Cardelino.

Instituto de Ingeniería Eléctrica, Facultad de Ingeniería,  
Universidad de la República, Uruguay.  
[proyecto.iie.biomecanica@gmail.com](mailto:proyecto.iie.biomecanica@gmail.com)  
<http://www.fing.edu.uy>

**Resumen** En este artículo se propone un sistema óptico de captura de movimiento basado en marcadores que facilita la tarea en el análisis biomecánico del movimiento de las personas.

Dicho sistema se compone de bloques independientes, dando una solución general que posibilita modificar o sustituir sus componentes fácilmente. Se describen lineamientos para generar un laboratorio virtual a partir del cual obtener una base de datos con secuencias de videos sintéticas y se exploran las características que debe tener un laboratorio de captura óptico basado en marcadores para facilitar el procesamiento.

Se reúne un conjunto de métricas para medir la performance de cada bloque y del sistema en su totalidad.

Las pruebas realizadas sobre el software implementado reflejaron que el mismo tiene una precisión del orden del centímetro sobre secuencias obtenidas en ambientes controlados. Estos resultados son buenos teniendo en cuenta que los algoritmos utilizados en cada bloque son de complejidad baja y se pueden optimizar.

**Keywords:** Biomecánica, calibración, detección de marcadores, reconstrucción, seguimiento, laboratorio virtual.

## 1. Introducción

El análisis de video es una herramienta fundamental para la recolección y estudio de datos. El seguimiento de puntos de referencia se utiliza para el cálculo de posición y otras variables asociadas como son la velocidad, la aceleración y por ende desplazamientos. Trabajar con video permite además estudiar secuencialmente situaciones estáticas en el tiempo. El seguimiento manual de dichos puntos resultaría tedioso por lo que es necesario contar con una herramienta que realice esta tarea automáticamente. Algunos ejemplos correspondientes a distintas áreas que ilustran estas necesidades son a *nivel asistencial* en el área de *fisioterapia*, *investigación académica en biomecánica*, *medidas de performance* en el deporte de alto nivel y *Animación 3D*, entre otras. Los ejemplos mencionados anteriormente definen distintos casos de uso con características disímiles, de manera que la búsqueda de una solución única que abarque las necesidades

particulares de todos ellos resulta compleja. Contar con este tipo de herramientas es fundamental para las necesidades de los equipos de profesionales, cuya alternativa son productos comerciales de alto costo.

En virtud de estas necesidades, este proyecto buscó realizar una aplicación básica y funcional de código abierto de análisis de video, que proporcione solución a las necesidades que se describieron ya sea utilizando como base algún proyecto de software libre existente, o en su defecto, desarrollando un prototipo de software básico completo que abarque el problema en forma general, para luego estudiar extender la aplicación hacia otros casos de uso.

El sistema creado pretende bajo ciertas condiciones controladas, obtener las coordenadas espaciales de un número de puntos de interés sobre un paciente. Una de las formas de obtener esto, y la estudiada en este trabajo, es colocar al sujeto con traje negro con marcadores en un ambiente con iluminación adecuada, filmar con varias cámaras a lo largo del tiempo, adquirir esta información en la computadora y mediante un posterior procesamiento obtener la posición 3D de cada uno de los puntos de interés (los marcadores) a lo largo de toda la secuencia.

## 2. Estado del arte

## 3. Explicación del algoritmo

Un sistema de captura de movimiento con las características necesarias para cumplir el objetivo de este proyecto debe estar formado por cuatro bloques generales: *calibración*, *detección de marcadores*, *reconstrucción* y *seguimiento*. En la Figura 1 se muestra un esquema del sistema a implementar.

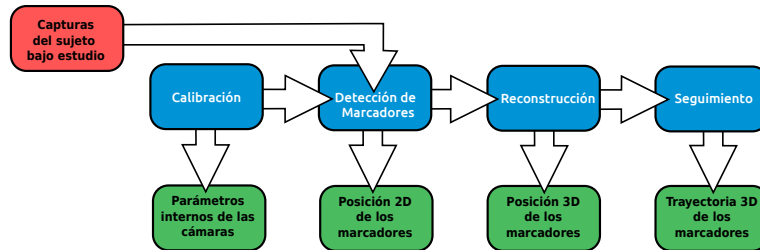


Figura 1: Diagrama de bloques del sistema completo.

Es importante destacar el hecho de poder separar el sistema en bloques independientes, esto permite en etapas futuras modificar u optimizar cada bloque individualmente sin afectar al resto.

En los capítulos que siguen, se explicará el funcionamiento de cada bloque de forma detallada, así como su implementación y el análisis de resultados de cada uno de ellos.

## 4. Implementación

### 4.1. Calibración

Con el objetivo de establecer una metodología de calibración que fuera válida para la configuración de cámaras con las que se diseñó el entorno virtual descrito en la sección ??, se probaron distintas implementaciones existentes. Para esto se evaluaron dos toolbox elaborados en *Matlab*. La metodología de calibración fue simulada en *Blender* mediante *scripts Python* y las imágenes obtenidas como resultado se procesaron con dichos toolbox. La descripción de la metodología y las simulaciones se detalla en [?].

Uno de los toolbox utilizados es el *Automatic Multi-Camera Calibration Toolbox (amcctoolbox)* [?], el cual utiliza como objeto de calibración un damero. Este método, aunque sus resultados tienen buena precisión [?], puede no ser suficientemente flexible para un sistema de muchas cámaras ya que, entre otras cosas, es necesaria la intervención manual en algunos casos.

El otro toolbox utilizado es el *Multi-Camera Self-Calibration Toolbox* [?]. Este método consiste en capturar el movimiento de una fuente puntual de luz que recorra el volumen de trabajo. Para cada cuadro se tiene un punto 3D en el espacio en una posición distinta y en cada una de las cámaras su correspondiente proyección. El error de re-proyección promedio obtenido es menor a 0.13 píxeles para todas las cámaras. Este método plantea una forma simple de calibrar un conjunto de muchas cámaras adecuado para el sistema de 17 cámaras del laboratorio virtual desarrollado en *Blender*.

### 4.2. Detección de marcadores

Una vez realizada la captura de video del paciente, el primer paso es reconocer los marcadores en el cuerpo del sujeto.

El bloque de detección de marcadores, se puede dividir en dos partes: la **segmentación** y el **filtrado de objetos**. En la Figura 2 se muestra el resultado del procesamiento de cada etapa.

El algoritmo realiza la detección siguiendo el siguiente proceso:

1. Se recibe como entrada un video y este es separado en cada uno de sus cuadros.
2. Se toma un cuadro y se lo segmenta utilizando umbralización de Otsu.
3. A partir de la imagen segmentada, se identifican los marcadores.
4. Se escribe la posición de los marcadores detectados para este cuadro en un archivo con formato XML.
5. Se toma el siguiente cuadro y se repite el proceso a partir del paso 2.

Este bloque fue implementado en lenguaje C++, debido a que es uno de los lenguajes de programación que cuenta con mayor cantidad de recursos para procesamiento de imágenes. En particular, se utilizaron las librerías *OpenCV* [?] y *CVBlob* [?] ya que funcionan para las plataformas principales de PC y dispositivos móviles, y están diseñadas para tener una gran eficiencia computacional en las implementaciones.

**Descripción de las etapas de detección** *Segmentación* En lo que respecta al bloque de segmentación, se eligió utilizar la umbralización generando umbrales con el método de Otsu[?] de tres clases.

Trabajar con tres clases permite ser un poco más flexible con los contrastes entre los marcadores y el resto de la imagen por lo que no sería estrictamente necesario, por ejemplo, que el traje del paciente y el fondo sean del mismo color.

*Filtrado* La etapa de filtrado no es más que una clasificación de los objetos segmentados. Dado que los objetos a detectar tienen formas relativamente sencillas (círculos blancos sobre fondo oscuro) y las condiciones de laboratorio son controladas al realizar la captura, esta etapa no requirió implementar algoritmos muy complejos. En particular, se implementó un detector de objetos circulares en base a momentos geométricos[?] y un filtro según el área de los mismos.

**Resultados** Para la etapa de segmentación se observó que el resultado obtenido depende fuertemente de las condiciones de captura y del umbral calculado. Se debe tener especial atención en las condiciones de captura ya que de no cumplir con las establecidas los resultados no son del todo satisfactorios (ver Figura 2). Por otro lado, si las capturas se hacen dentro de las condiciones establecidas, los resultados obtenidos son aceptables.



Figura 2: Izquierda: imagen original de una secuencia fuera de las hipótesis de captura. Centro: captura original de una secuencia real bajo las hipótesis de captura. Derecha: marcadores detectados.

### 4.3. Reconstrucción

A la salida del bloque de detección de marcadores se tiene, para cada cámara y para cada cuadro de una secuencia adquirida, un conjunto de coordenadas en dos dimensiones  $(x, y)$  que ubican la posición en la imagen de aquellos marcadores que fueron detectados. El proceso de reconstrucción consiste en obtener la posición de los marcadores en el espacio, a partir de la posición de los marcadores en al menos dos retinas.

El proceso de reconstrucción que se presenta fue inspirado en el trabajo de Herda [1] y consiste en tres pasos fundamentales:

1. Encontrar la correspondencia entre puntos en retinas diferentes.
2. Seleccionar la mejor correspondencia.
3. Reconstruir y verificar en el resto de las retinas.

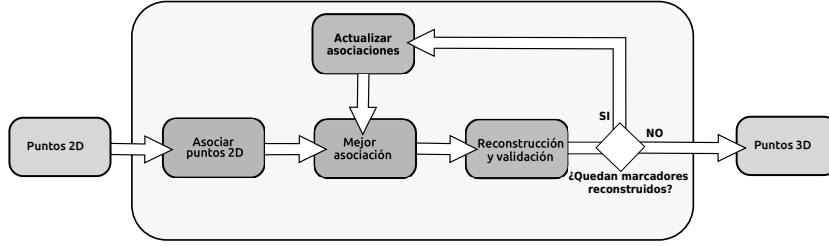


Figura 3: Diagrama de bloques del algoritmo de reconstrucción.

**Algoritmo** El algoritmo implementado recibe como entrada los puntos 2D de los marcadores detectados y devuelve como salida los puntos 3D reconstruidos. En la Figura 3 se presenta un diagrama del algoritmo.

*Asociar puntos 2D* Este bloque recibe como entrada las coordenadas de los puntos detectados en cada una de las cámaras, parámetros de las mismas tales como sus matrices de proyección y devuelve para cada punto una lista ordenada por relevancia, de las asociaciones existentes con puntos en otras cámaras. Los pasos a seguir son los siguientes:

- Seleccionar dos cámaras y considerar un punto en una de ellas, por ejemplo el punto  $x_{11}$  de la cámara 1.
- Proyectar la recta epipolar  $l_{11}$  correspondiente al punto  $x_{11}$  sobre la cámara 2.
- Tomar las distancias de los puntos detectados en la cámara 2 a la recta  $l_{11}$ .
- Generar una lista de posibles asociaciones ordenada por distancia.

Se asume que los puntos de la cámara 2 que tengan mayor posibilidad de corresponder con el punto  $x_{11}$ , son los más próximos a  $l_{11}$ . Repitiendo el procedimiento de manera inversa, esto es, de la cámara 2 a la cámara 1, se obtiene igualmente para cada punto de la cámara 2 los puntos de la cámara 1 ordenados según su proximidad a la recta epipolar correspondiente. A continuación se toman otros pares de cámaras y se vuelve a repetir el proceso. Es importante resaltar que para la elección de los pares de cámaras se han considerado dos casos. El primero de ellos evalúa cada cámara respecto a todas las restantes y el segundo considera la disposición de las cámaras en el espacio y empareja las cámaras adyacentes de manera consecutiva.

*Mejor asociación* A partir de la lista con asociaciones entre pares de cámaras es necesario elegir aquella que posea mayor probabilidad de conformar la pareja de imágenes correspondiente a la proyección de un marcador 3D sobre dichas vistas. Sobre cada par de cámaras se toma aquella asociación que posea la menor distancia y contenga puntos válidos, descartando las restantes. Un punto se considera válido si en iteraciones anteriores no se pudo asociar a ningún punto 3D reconstruido. De esta forma cada punto de una cámara es asociado, si existen puntos válidos, con un punto en otra de las cámaras. Para evaluar de todas las

relaciones entre cámaras, cuál de los pares de puntos asociados disponibles es el par que posee mayor posibilidad de corresponder a las proyecciones de un punto 3D, se proyectan los rayos de proyección de todos los pares de puntos disponibles y se toma el par que genere rayos de proyección con la menor distancia entre sí.

*Reconstrucción 3D y validación* Si se encuentra al menos un punto en otra cámara que valide la reconstrucción se asume que ésta es correcta, se retira a la pareja que genera la reconstrucción así como también a los puntos que lograron validarla y se itera nuevamente repitiendo el proceso con la siguiente mejor pareja asociada entre dos cámaras.

La información que contiene un punto en una retina se mapea en el espacio 3D sobre el rayo de proyección que contiene a dicho punto y al centro de la cámara correspondiente. Supongamos que se tienen los rayos de proyección en el espacio 3D de todos los puntos contenidos en las retinas y que  $x_1$  es un punto en la cámara 1 y  $x_2$  es un punto en la cámara 2 que se encuentran asociados y reconstruyen al punto  $X_{12}$ . El algoritmo implementado asume que un punto en alguna cámara, valida a  $X_{12}$  si junto a  $x_1$  reconstruye un punto 3D que se encuentra dentro de la esfera  $B(X_{12}, \delta)$  de centro  $X_{12}$  y radio  $\delta$ , donde  $\delta$  es un cierto valor umbral.

Idealmente  $X_{12}$  se genera al interceptar los rayos de proyección de los puntos  $x_1$  y  $x_2$ , pero debido a incertidumbres en la detección de marcadores o la calibración, comúnmente los rayos se van a cruzar. La reconstrucción, se estima como el punto del espacio de menor distancia a ambos rayos.

*Actualizar asociaciones* Los puntos 2D sobre las cámaras asociados a la reconstrucción del punto 3D  $X$  no deben ser considerados nuevamente, por tanto estos puntos 2D se consideran como puntos no válidos en las siguientes iteraciones.

Finalmente el proceso iterativo se detiene cuando no hay más marcadores para reconstruir, lo cual implica que se cumple alguna de las siguientes condiciones:

- el número de marcadores reconstruidos es igual al número de marcadores que tiene colocada la persona, o igual al número máximo de marcadores reconstruidos que se haya indicado.
- No existen puntos 2D válidos tal que pueda establecerse una asociación entre puntos de distintas vistas.

#### 4.4. Seguimiento

El seguimiento de trayectorias se realiza sobre una ventana deslizante de tres a cuatro cuadros donde se enlazan los puntos reconstruidos de manera de mantener un movimiento lo mas suave posible.

Esta metodología fue la utilizada por Herda [1] en su trabajo basándose en los estudios de Malik, Drako, Papantoniou [?] .

**Algoritmo** Sea la trayectoria de un marcador enlazada hasta el instante  $[f]$  sobre la cual desea buscarse su próximo punto en  $[f+1]$ , el movimiento entre  $[f-1]$

y  $[f]$  es prolongado para establecer un centro de búsqueda y encontrar el punto reconstruido que mejor continúa la trayectoria como se muestra en la Figura 4 .

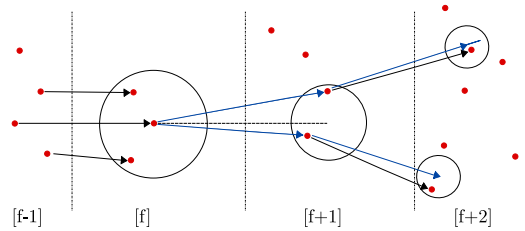


Figura 4: Seguimiento en cuatro cuadros, siendo  $[f]$  el cuadro actual que queremos seguir en  $[f+1]$ . ( Fuente Human movement science 20(3), 313–341 [1] ) .

Se presentan tres posibles casos al buscar puntos reconstruidos:

- Si solo se encuentra un punto reconstruido se agrega a la trayectoria para el cuadro  $[f+1]$ , buscando el mas cercano a la estimación calculada como aquella que mejor se aproxima a una trayectoria de tres puntos con aceleración mínima
- En el caso de encontrar mas de un punto cada posible candidato es evaluado para realizar una segunda estimación hacia  $[f+2]$  de forma que la aceleración entre  $[f-1]$ ,  $[f]$  y el candidato en  $[f+1]$  sea la misma que entre  $[f]$ , el candidato en  $[f+1]$  y la estimación en  $[f+2]$ . Luego de todos los posible caminos en cuatro cuadros, se elige el de menor variación de aceleración.
- Si no se encuentra ningún punto, se procede a aumentar de forma limitada el radio de búsqueda en  $[f+1]$  de forma excepcional. Esto se hace para continuar trayectorias que entran en estado de reposo y el último movimiento conocido es nulo o muy pequeño.

Si una trayectoria queda trunca durante el enlazado, se intenta recuperar prolongando el movimiento en próximas cuadros para encontrar puntos reconstruidos cercanos a las estimaciones y extrapolar los puntos intermedios. Por otro lado, se implementan umbrales para definir límites sobre la aceleración de los enlaces obtenidos y detectar discontinuidades durante el seguimiento.

Con estas medidas implementadas es posible detectar las trayectorias individuales sobre los puntos reconstruidos, detectar de forma simple posibles discontinuidades, y estimar reemplazos en casos de pérdidas. La captura mostrada en la Figura 5 corresponde a la marcha y se resaltan las trayectorias individuales de puntos de la pierna así como un esqueleto simple generado simplemente para visualizar la evolución entre marcadores

El conjunto de puntos reconstruidos puede ser sometido a otros algoritmos de seguimiento como por Kalman [?] requiriendo la inicialización de modelos, o algoritmos basados en restricciones mas fuertes a las presentadas en este trabajo

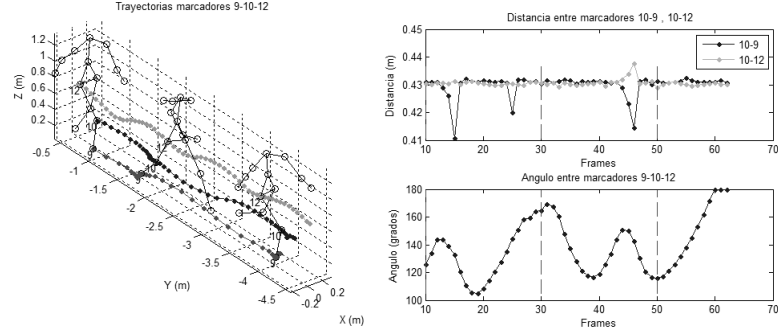


Figura 5: Posibles restricciones en ángulo y distancia, para el caso de la pierna en marcha. Izquierda: trayectorias de marcadores de pierna. Derecha: distancia y ángulo entre marcadores de la pierna.

como podrían ser las distancias relativamente constantes entre marcadores de los miembros y ángulos continuos entre articulaciones, pero requieren un estudio considerable de las características de cada sujeto y movimiento a capturar. La Figura 5 muestra algunas posibles restricciones en la marcha sobre los huesos de la pierna.

## 5. Resultados

Fueron utilizadas las métricas establecidas en HumanEva [?] para comparar los conjuntos de datos obtenidos a la salida de cada bloque individual con aquel conjunto de referencia obtenido en el ground truth de base de datos, buscando primero la correspondencia entre puntos y luego la distancia euclidiana (en 2D para cámaras y 3D para espacio reconstruido) entre los puntos de ambos conjuntos.

El error de detección de los marcadores en cada una de las cámaras no supera un par de píxeles para el caso de cámaras con resolución en imagen de  $1600 \times 600$ . Es posible reducir la resolución de las cámaras hasta  $800 \times 300$  manteniendo los mismos resultados, pero en resoluciones menores comienza a degradarse la tasa de detección de marcadores en cámaras individuales lo cual perjudica la siguiente etapa de reconstrucción. Fueron realizadas pruebas inyectando ruido en el bloque de detección y su impacto en etapas posteriores encontrando que se puede trabajar con un poco mas de ruido (no mas de tres píxeles de error) sin comprometer significativamente el error final.

Si las cámaras se encuentran en las condiciones de error anteriores, una cobertura total con 17 cámaras rodeando el movimiento a capturar permite reconstruir todas las trayectorias con un error por debajo del centímetro con errores máximos cercanos a tres centímetros. Los resultados se mantienen dentro de estos límites si se reduce a ocho cámaras colocando pares de cámaras en cada esquina



del espacio de captura. Reducir aún más el conjunto de cámaras sin degradar la performance es posible con ciertas condiciones: un conjunto de seis cámaras colocadas en dos costados de a tres, y cada costado tiene dos cámaras próximas con una tercera mas alejada. De esta manera se tiene las condiciones mínimas por las cuales se estableció la reconstrucción, tener dos cámaras para reconstruir y una tercera para confirmar.

Este caso limite de reconstrucción muestra comportamiento global similares a los anteriores, pero presenta degradaciones puntuales para ciertas trayectorias de marcadores que superan el error promedio y deberían ser reparadas con medidas avanzadas.

## 6. Conclusiones

Se obtuvo en forma íntegra un sistema óptico de captura de movimiento basado en marcadores, que a partir de las capturas de video de una persona en un ambiente de laboratorio con las condiciones adecuadas, obtiene la posición 3D de los marcadores presentes en el cuerpo de dicha persona, logrando representar su movimiento con una precisión del orden del centímetro.

Si bien inicialmente el objetivo era que el sistema funcionara por lo menos para el caso de uso de la marcha, se han probado otros movimientos con resultados aceptables. La aplicación desarrollada permite a partir de múltiples capturas de vídeo de un sujeto en movimiento, detectar los marcadores en cada toma de video. Junto a la información de las cámaras, luego reconstruye la posición de los marcadores en el espacio y finalmente logra identificar cada marcador a lo largo de la secuencia temporal. Cabe destacar que el sistema implementado no es solo óptico, sino que es lo bastante general para funcionar con cualquier sistema de adquisición que genere imágenes, por ejemplo con imágenes infrarrojas provenientes de sistemas de captura modernos.

## Referencias

1. Herda, L., Fua, P., Plänkers, R., Boulic, R., Thalmann, D.: Using skeleton-based tracking to increase the reliability of optical motion capture. *Human movement science* 20(3), 313–341 (2001)