

# Análisis de Video en Biomecánica

Andréi Guchin <sup>\*</sup>, Gonzalo Pereira,  
Guillermo Ottado, Mauricio Ramos, and Juan Cardelino.

Instituto de Ingeniería Eléctrica, Facultad de Ingeniería,  
Universidad de la República, Uruguay.  
`proyecto.iie.biomecanica@google.com`  
`http://www.fing.edu.uy`

---

<sup>\*</sup> Esto es por si se quiere agregar algo referido a alguno.

**Resumen** En este artículo se propone un sistema óptico de captura de movimiento basado en marcadores para facilitar 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 muestra como 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

Especialistas de distintos ámbitos académicos o profesionales se encuentran habitualmente en la necesidad de realizar estudios del movimiento del cuerpo humano. Esta tarea implica registrar la posición de miembros o articulaciones en el espacio y su correspondiente evolución en el tiempo.

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.

En este contexto, 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 de dichos puntos resultaría tedioso si se hiciera manualmente por lo que resulta necesario contar con una herramienta que realice esta tarea automáticamente.

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. Por ejemplo, en el ámbito deportivo la velocidad del movimiento es una variable importante a tener en cuenta para desarrollar una solución, de esta variable depende la elección tanto del sistema de adquisición como de los algoritmos más eficaces para el registro del movimiento. De igual manera definir la portabilidad del sistema depende de si la actividad a relevar es en condiciones de laboratorio controladas o al aire libre, debido a la protección y transporte de equipos o las variaciones en las condiciones de iluminación, ruido, etc..

Al día de hoy, las soluciones de software disponibles que podrían asistir al especialista en su tarea, son mayormente comerciales. Las pocas alternativas de código abierto, carecen de las características necesarias para el especialista o están enfocadas hacia otras áreas de aplicación. Contar con este tipo de herramientas es fundamental para las necesidades de los equipos de profesionales, cuya alternativa son productos comerciales de alto costo.

QUIZAS ACÁ REDONDEAR CON LO QUE SE PRETENDE EN EL PAPER, O SEA LOS OBJETIVOS PROPUESTOS Y LUEGO LOS REQUERIMIENTOS QUE SE TIENEN.

## **2. Base de datos**

Aquí tengo que meter por lo menos la descripción de las secuencias que utilizamos y de donde se sacaron

### **3. Implementación**

Acá por lo menos el diagrama de bloques general

6      lista abreviada de autores

## 4.    **Calibración**

## 5. Detección de marcadores

Una vez realizada la captura de video del paciente, el primer paso en el procesamiento de las secuencias de video en un sistema de captura de movimiento es reconocer los marcadores en el cuerpo del sujeto, para luego tener la posición de cada uno de ellos en el espacio y a lo largo del tiempo.

Como se pudo ver anteriormente, para este sistema las condiciones de captura de las secuencias son muy favorables lo que permite utilizar métodos simples de segmentación basados en el estudio de los píxeles de la imagen.

El bloque de detección de marcadores, se puede dividir en dos partes bien definidas: la **segmentación** y el **filtrado de objetos** hasta obtener los marcadores. En la Figura 1 se muestra el resultado del procesamiento de cada parte.

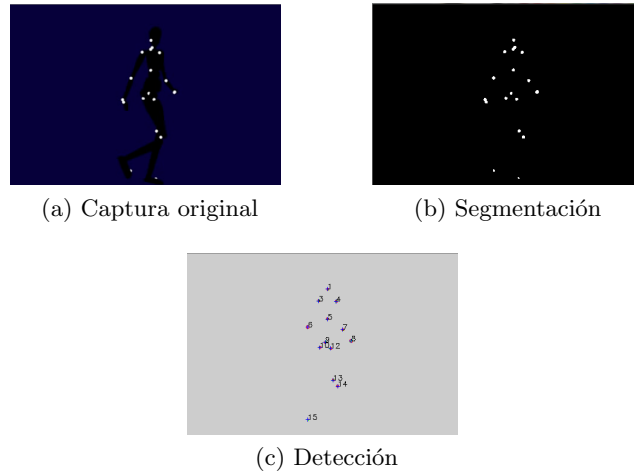


Figura 1: Ejemplo de funcionamiento del bloque.

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 calcula el umbral de Otsu. Si al comenzar la segmentación es ingresado un umbral fijo, este paso se saltea.
3. Con el umbral calculado (o ingresado), se filtra el cuadro.
4. A partir de la imagen filtrada, se identifican los marcadores.
5. Se escribe la posición de los marcadores detectados para este cuadro en un archivo con formato XML.
6. Se toma el siguiente cuadro y se repite el proceso a partir del paso 2.

El bloque de detección de este sistema fue implementado en el 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. Además, estas librerías son de gran popularidad dentro de las librerías de código abierto con similares características, lo cual implica que poseen una comunidad activa de usuarios muy grande.

Además de los videos a procesar, el algoritmo tiene como entradas opcionales un conjunto de argumentos que establecen distintos parámetros para el procesamiento. Estos argumentos se presentan a continuación:

- **t ¡valor!**, establece un umbral fijo para la umbralización. Debe ser un valor entre 0 y 255.
- **A ¡valor!**, área máxima para el filtro por área. Debe ser un valor positivo.
- **a ¡valor!**, área mínima para el filtro por área. Debe ser un valor positivo.
- **s**, guarda los videos que resultan de la salida de los bloques de umbralización y detección de blobs.

### 5.1. Descripción de las etapas de detección

**Segmentación** En lo que respecta al bloque de segmentación, existen muchos métodos para implementar el mismo. Se comenzaron probando los más simples y se fue aumentando el nivel de complejidad hasta encontrar un método que se ajuste a los requerimientos del sistema. Se eligió utilizar la umbralización, generando umbrales con el método de Otsu[?] de tres clases.

Con el método de Otsu [?] se pretende, a partir del histograma de la imagen, separar los píxeles de dicha imagen en tres niveles encontrando dos umbrales que los separen. 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.

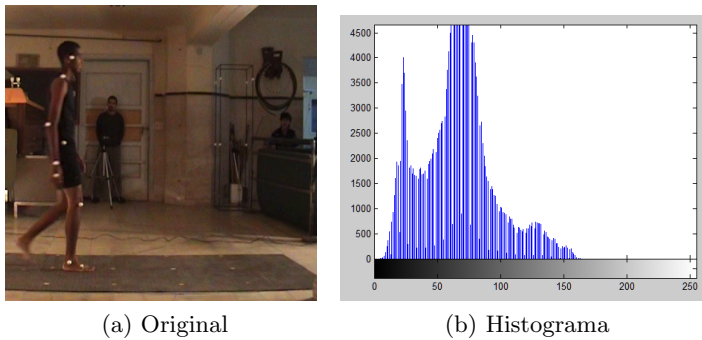


Figura 2: Captura original de un paciente real y su histograma de intensidad de píxeles.



En la Figura 2b se observa el histograma de intensidad de píxeles de la imagen 2a, que corresponde a una captura de un paciente real. Dicho histograma muestra tres picos bien definidos, por lo que utilizar el método de Otsu de tres clases parece una buena opción para determinar los umbrales de segmentación. Manteniendo esta relación de intensidades entre fondo, ropa del paciente y marcadores, este método funcionaría adecuadamente.

La justificación de la realización de este algoritmo para el bloque de segmentación puede entenderse desde dos puntos de vista:

- Por un lado, se elaboró un algoritmo de una complejidad relativamente baja como se estableció en los requerimientos, pero que cubre todas las características necesarias para esta clase de sistema. En la sección ?? se verá que los resultados obtenidos son buenos y el error cometido está dentro de los márgenes que aseguran el buen funcionamiento del sistema.
- Por otro lado, a pesar de que el estado del arte de segmentación presenta una gran cantidad de algoritmos, el estado del arte “industrial” está bastante más atrasado del estado del arte referente a investigaciones, debido al tiempo que implica realizar una nueva implementación o la dificultad para conseguir una que funcione significativamente mejor a las existentes. Por esto, implementar una primera versión del sistema con el algoritmo de segmentación elegido no está muy lejos de lo utilizado actualmente en la “industria”. Es importante aclarar que este bloque (como el resto de los bloques) está implementado de forma tal que en un futuro se pueda optimizar tanto como se quiera, sin afectar el resto de los bloques del sistema. Esto da la posibilidad de modificar el sistema no solo para mejorar la segmentación sino también para robustecerla, permitiendo por ejemplo, realizar capturas fuera de las condiciones del laboratorio, como usualmente es necesario en el ámbito deportivo.

**Filtrado** La etapa de filtrado de objetos 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 requerirá 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.

Los momentos geométricos son propiedades numéricas que se pueden obtener en una determinada imagen, los cuales proporcionan una alternativa interesante para la representación de la forma de un objeto. Tienen en cuenta todos los píxeles de la imagen, no sólo los bordes.

## 5.2. Resultados

Para el bloque de umbralización se observó que el resultado obtenido, como era de esperarse, depende fuertemente de las condiciones de captura y obviamente del umbral calculado. En el caso sintético, como el mostrado en la Figura

1, donde se imponen las condiciones de captura de manera deseable, los resultados fueron muy buenos. Esto es razonable ya que como se vio anteriormente, el método de Otsu calcula el umbral óptimo, por lo que mientras en la imagen se presenten los marcadores como los objetos más brillantes y se aprecie un alto contraste entre los mismos y el resto de los elementos (fondo, paciente, etc.), la segmentación será muy efectiva. Para el caso real, se debe tener especial atención en las condiciones de captura ya que de no cumplir con las establecidas los resultados de la segmentación no son del todo satisfactorios (ver Figura 3). Por otro lado, si las capturas se hacen dentro de las condiciones establecidas, los resultados obtenidos son mucho mejores (ver Figura 4).



(a) Captura original de una secuencia real.

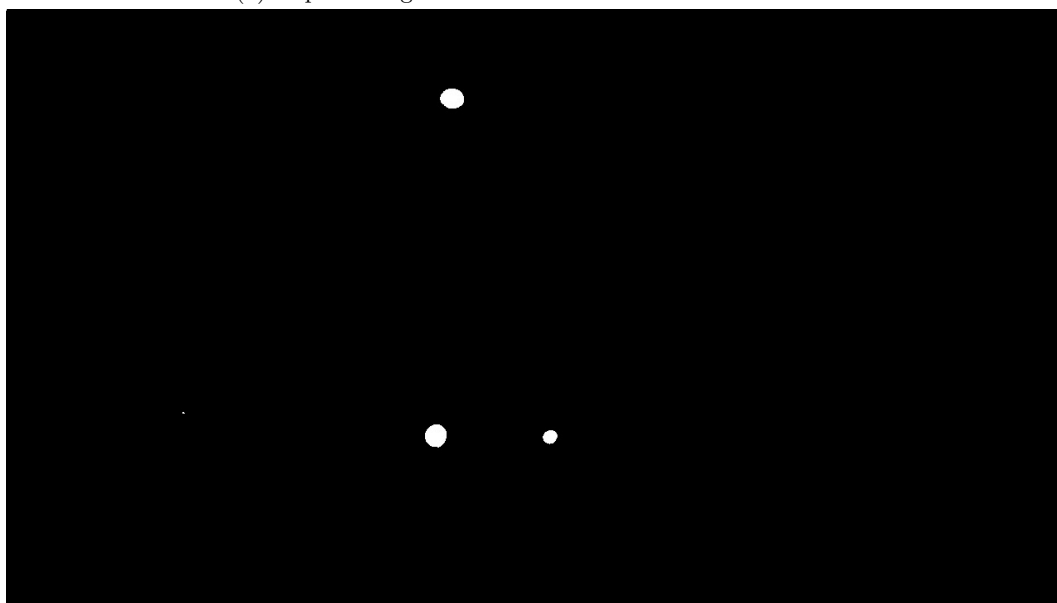


(b) Imagen filtrada con el umbral de Otsu.

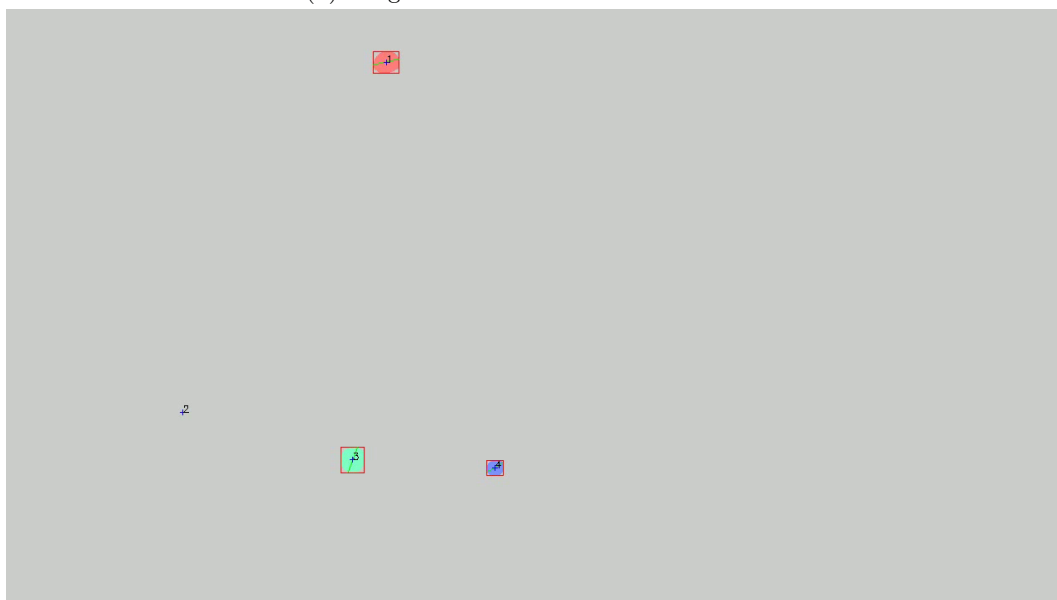
Figura 3: Entrada y salida del bloque umbralización para un caso real fuera de las hipótesis de captura.



(a) Captura original de una secuencia real.



(b) Imagen filtrada con el umbral de Otsu.



(c) Marcadores detectados

## 6. 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 las coordenadas en tres dimensiones de 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.

### 6.1. Algoritmo

El algoritmo implementado recibe como entrada los puntos 2D de los marcadores detectados y devuelve como salida los puntos 3D reconstruidos. El primer paso consiste en establecer una asociación entre ciertos puntos 2D de distintas cámaras. Luego, se pasa a un conjunto de bloques que se ejecutan de manera iterativa hasta que no queden marcadores para reconstruir. En dicho bloque se busca la mejor asociación entre puntos bajo determinado criterio, luego se reconstruye un punto 3D y se realiza un proceso de validación de dicha reconstrucción. En la iteración siguiente se actualizan las asociaciones que habían sido establecidas previamente. Cuando no hay más marcadores para reconstruir se detiene el proceso iterativo y se devuelven aquellos marcadores que fueron reconstruidos en cada iteración. En la Figura 5 se presenta un diagrama del algoritmo.

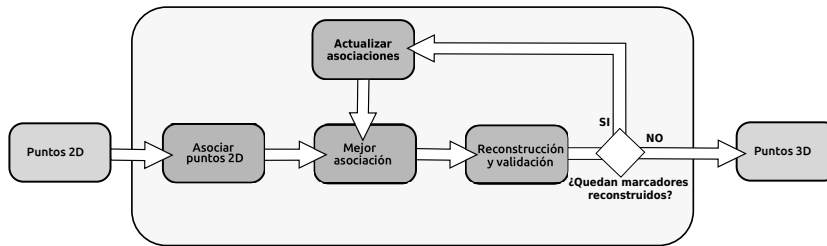


Figura 5: Diagrama de bloques del algoritmo de reconstrucción.

**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. Basándose en lo explicado anteriormente para un par de cámaras el proceso se puede ejemplificar en la Figura 6 y los pasos a seguir son los siguientes:

Figura 6: Asociación de puntos 2D en dos cámaras.

- 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 aquellos que al ser evaluados por la ecuación obtienen valores próximos a cero. 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.

## 6.2. Mejor asociación

A partir de la lista con asociaciones entre puntos de dos vistas generada anteriormente, 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.

Recordando que todas las asociaciones de puntos entre pares de cámaras se encuentran ordenadas por distancia, 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 ha podido asociar a ningún punto 3D reconstruido (ver Sección 3). 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.

### 6.3. Reconstrucción 3D y validación

Luego de encontrar la mejor asociación de puntos se procede a reconstruir y validar la misma. 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.

Figura 7: Reconstrucción entre cámaras 1, 2 y validación con cámara 3.

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 de centro  $O_1$  y  $x_2$  es un punto en la cámara 2 de centro  $O_2$  se encuentran asociados y reconstruyen al punto  $X_{12}$ . El algoritmo implementado asume que un punto en una 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. Como se muestra en la Figura 7, el punto  $x_3$  de la cámara 3 valida la reconstrucción, no así el punto  $\tilde{x}_3$ .

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, por lo que  $X_{12}$  se encuentra en el punto medio del segmento perpendicular a ambos rayos.

### 6.4. Actualizar asociaciones

Del bloque anterior se tiene, un punto  $X$  reconstruido y sus correspondientes proyecciones en cada una de las cámaras. Dichas proyecciones no deben ser consideradas nuevamente, por tanto estos puntos 2D que reconstruyen  $X$ , no se consideran como puntos 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.

## 7. Seguimiento

## 8. Performance



## 9. Conclusiones

The conclusion goes here.

## 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)