VOLUMEN 16
XXVII Verano De la Ciencia
ISSN 2395-9797
www. jóvenesenlaciencia.ugto.mx

Movimientos antropomorfos

José Angel Alejandro Soto, Diego Emir Garcia Moreno, Sebastián Sánchez Lara, Angel Gilberto Ayala Pérez, Claudia Esteves
Universidad de Guanajuato DCNE - Departamento de Matemáticas

Resumen

El antropomorfismo ha estado presenten en la humanidad desde la antigüedad con figuras y pinturas de animales morfológicamente similares a los humanos (pensando en el esqueleto) y ha sido de mucho interés su estudio en los últimos años, principalmente por la creciente industria del entretenimiento aunque su estudio ha beneficiado muchas áreas del conocimiento como la ingeniería, la medicina, el deporte, entre otros. Poder estudiar su comportamiento por medio de simulaciones permite ahorrar tiempo y dinero además de obtener resultados más realistas por medio de la captura de movimiento, que es grabar los movimientos de objetos o personas para procesarlos digitalmente, en lo siguiente vamos a aprender más a detalle qué es, los tipos de captura de movimiento, los problemas de cuantas cámaras poner y en donde, la reconstrucción de un modelo 3D a partir de un conjunto de imágenes 2D así como reconocer marcadores, los errores al reconstruir, sus aplicaciones y el dispositivo de captura de movimiento con el que cuenta la Universidad de Guanajuato en el Departamento de Matemáticas de la División de Ciencias Naturales y Exactas.

Palabras clave: movimientos antropomorfos, captura de movimiento, mocap

Mocap

La captura de movimiento (mocap abreviado del inglés motion capture) es el proceso de grabar el movimiento de un objeto o personas para generar un modelo 3D que pueda ser ejecutado por la computadora, estos son clasificados en outside-in, inside-out e inside-in dependiendo de las fuentes de captura y de donde son puestos los sensores o marcadores.

- Los sistemas outside-in usan sensores externos para capturar los datos de fuentes colocadas en partes del cuerpo, un ejemplo de esto son los dispositivos de rastreo basados en cámaras en el que las cámaras son los sensores y los marcadores reflectantes son las fuentes
- Los sistemas inside-out son sensores que se ponen en el cuerpo que recolectan fuentes externas, un ejemplo de esto son sistemas electromagnéticos cuyos sensores se mueven en un campo electromagnético generado externamente
- Los sistemas inside-in tienen tanto los sensores como las fuentes en el cuerpo.

Los sistemas ópticos de captura de movimiento son un método muy preciso para capturar ciertos movimientos, es un sistema basado en una computadora que controla la entrada de las cámaras, las cuales son sensibles a la luz para crear las representaciones digitales. Este es el sistema con el que cuenta la universidad del cual se hablará más adelante, primero hay que tratar con el problema del posicionamiento de las cámaras.

Posicionamiento de cámaras

Un problema que surge comúnmente en la captura de movimiento es el posicionamiento adecuado de las cámaras. Para saber las coordenadas de un punto de interés, es necesario que este sea visto por al menos dos cámaras (o más, dependiendo del sistema). Así, es de interés poder calcular las posiciones que maximicen la visibilidad de puntos.

Cómo parte del proyecto, consultamos el paper Optimal Camera Placement for Motion Capture Systems. Dicha publicación propone una nueva manera de calcular posiciones óptimas para las cámaras en un entorno para la captura de movimiento.

Métrica de error basada en oclusión

La métrica de error basada en oclusión presenta una forma de medir la visibilidad de un punto objetivo específico por al menos dos cámaras, para un conjunto de cámaras, cuando se presenta oclusión dinámica.

Para tomar en cuenta la gran mayoría de oclusores posibles, se considera un plano vertical que gira alrededor del punto objetivo. Así, entre todos todos los ángulos entre 0° y 360° que puede girar este plano respecto al punto, habrá ciertos intervalos de ángulos en los cuáles el punto no es visible por al menos dos cámaras. La métrica calcula la suma de las longitudes de estos intervalos. Informalmente, esta corresponde a la "suma de ángulos" en los cuáles no es visible el punto objetivo, y nos interesa minimizar esta cantidad.

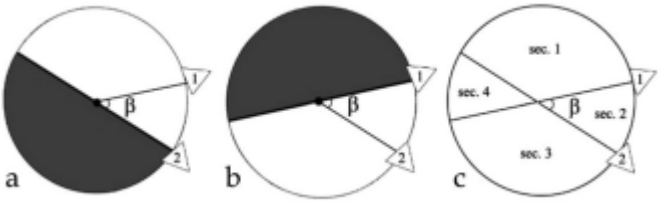


Figura 1. Oclusión

En la imagen anterior, observamos que la visibilidad del punto sólo cambia cuando el plano cruza el vector de la vista de una cámara. Notamos también que si existen n cámaras, sus vectores de vista generan 2n regiones (es decir, intervalos de ángulos) de interés; por ejemplo, en la imagen, dos cámaras dividen el círculo centrado en el punto de interés en 4 partes.

Además de esto, consideramos una restricción de triangulación para un par de cámaras: aunque ambas sean capaces de ver el punto de interés; sólo se toman en cuenta si el ángulo entre sus vectores de vista está entre 40° y 140°. Esto es porque si el ángulo es muy grande o muy pequeño, el error numérico al intentar triangular el punto es también muy grande. Asimismo, se descartan las cámaras si el punto está mas allá de el rango efectivo de estas.

La métrica anterior se puede calcular, en pseudocódigo, con el siguiente algoritmo:

```
#n - cantidad de camaras
#C[1..n] - posiciones de las cámaras
#S[1..2n] - las longitudes de las regiones que generan
los vectores de vista de las camaras
#Cf[] - posiciones de cámaras en C frente al oclusor
#flag - es verdadera si en la región,
el punto es visible por al menos dos cámaras
Q - suma de las regiones en las que no es visible el punto.
por al menos dos cámaras.
-----------------------------------------------------
calcular S
para i = 1 hasta 2n:
    calcular Cf a partir de C y S[i]
    flag = false
    para j = 1 hasta longitud(Cf):
           para k = j + 1 to longitud(Cf):
```

```
                    si Cf[j] y Cf[k] satisfacen la
                    restricción de triangulación:
                           flag = true
                           break
          si flag = false:
                 Q += S[i]
regresa S[i]
```

Recocido simulado

Para aplicar la métrica anterior en el cálculo de posiciones óptimas, se utiliza el método de recocido simulado. Este es un método muy general que, de manera heurística, busca una aproximación al mínimo o máximo global de una función.

Este algoritmo considera una temperatura. Se inicia en una estado arbitrario; y se consideran varios estados siguientes válidos. Si alguno de estas mejora la función que queremos minimizar (o maximizar), cambiamos a este estado. De otra manera, aunque el siguiente estado empeore la función, con cierta probabilidad que depende de la temperatura, se cambia a este estado.

La temperatura está definida de tal manera que, entre más alta, es más probable que se tome una nueva posición. Así, se inicia con una temperatura alta, la cual va disminuyendo conforme pasa el tiempo. El objetivo de esta es “escapar” mínimos locales; es decir, estados que no son óptimos, pero que están rodeados de estados peores.

Volviendo al problema de posicionamiento de cámaras, consideramos los estados como el conjunto de posiciones de las cámaras, y la función a minimizar será una modificación sobre la *métrica de error basada en oclusión*. Dado un estado, tiene sentido considerar a los siguientes estados posibles como aquellos que cambian la posición de exactamente una cámara.

La primera modificación necesaria es considerar múltiples puntos. Supongamos que estos son p1,…,pm. Entonces denotemos Qi como la métrica descrita antes, con el punto pi como objetivo. Nos interesa minimizar

$$f = \sum_{i=1}^{m} Q_i.$$

Esto aún genera un problema. ¿Por qué? La métrica original no incentiva el poner una cámara en una región que no contenga ninguna cámara: al pasar de 0 cámaras a 1 cámara en la región, esta región sigue sin contener al menos un par de cámaras, así que su longitud se sigue considerando. Más aún, este cambio empeora la métrica si la región de la cual se toma la cámara a mover contiene sólo un par de cámaras.

Para esto definimos θ como un *penalty* asignado a puntos que no son visibles por un par de cámaras. Consideremos un punto pi, y Ci como el conjunto de puntos cuyos vectores de vista ven a pi. Entonces, definimos el valor Ei como la nueva métrica, dada por:

$$E_i = \begin{cases} 360 + \theta & |C_i| = 0 \\ Q_i & |C_i| > 0. \end{cases}$$

Finalmente, queremos minimizar f, redefinida como

$$f = \sum_{i=1}^{m} E_i,$$

usando recocido simulado. El paper original sugiere usar θ=360 para obtener mejores resultados.

El sistema de Motion Capture en CIMAT - Guía de uso

El procedimiento anterior, aunque interesante en teoría, no es aplicado en CIMAT en la actualidad. Para el sistema de CIMAT, se cuenta con la siguiente guía:

Calibración

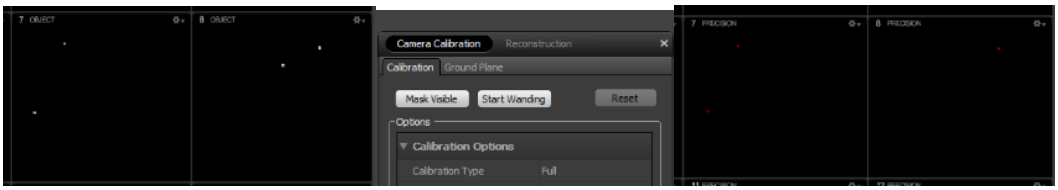
Se utiliza el software Motion, de Optitrack; a través de este se hará la calibración.

Preparar el ambiente

Se deben bloquear o remover los objetos que puedan interferir con las cámaras; ya sea ventanas abiertas, superficies reflejantes, luces infrarojas, marcadores, etc.

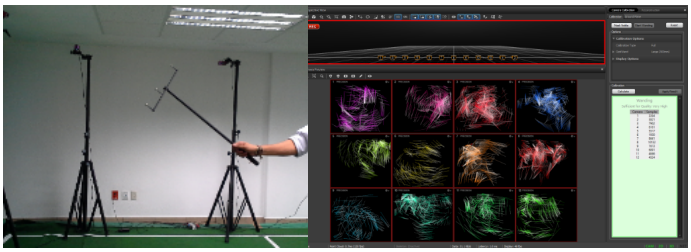
Camera masking

Se cubren las fuentes de luz restantes; incluyendo la interferencia de las cámaras entre sí. Esto se puede hacer con la opción de Auto-Masking, “Block-Visible”. Esta esconde de manera automática los puntos brillantes. Otra opción es cubrirlas manualmente con herramientas de selección.



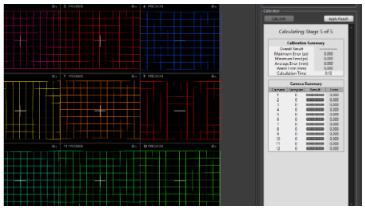
Wanding

Se debe presionar el botón “Start Wanding”; el motor de calibración comenzará a grabar samples cuando detecte la varita de calibración. Se debe cubrir con los marcadores de la varita todo el volumen que detecten las cámaras. El motor de calibración se mostrará verde cuando tenga suficientes samples. Es recomendable seguir el proceso hasta cubrir suficiente área.



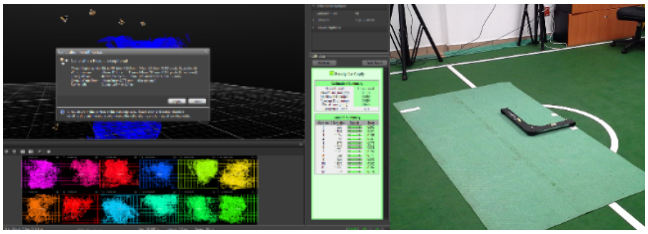
Calculation

Tras hacer el “Wanding”, el proceso de cálculo se hace presionando el botón “Calculate” del panel “Calibration”. Solo es necesario esperar a que el proceso converja a una solución, pero se puede dejar continuar para obtener calibración más precisa. Se puede monitorear el proceso con el visor 3D, que se ve como sigue:



Aplicar los resultados

Solo es necesario presionar el botón “Apply Results”. Saldrá un aviso de guardar los resultados del “Wanding”. Tras guardar, es posible escoger el plano del suelo con el artefacto que determina el plano Z.



Verificar los resultados

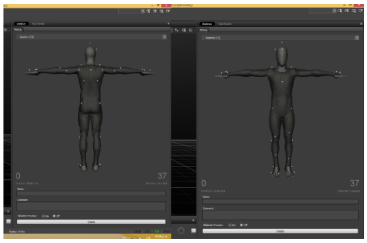
Ejecutar la herramienta “Volume Accuracy Tool” para verificar la calidad de la calibración. Se mide la desviación entre las longitudes que se midieron y las reales de la varita de calibración.

Sesión de grabación

Suit Up

Ponerse el traje de la talla adecuada, lo más ajustado posible; esto previene que los marcadores se muevan mucho respecto al cuerpo.

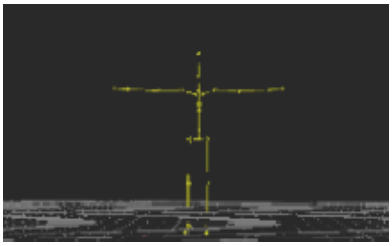
Los marcadores se deben colocar en lugares específicos (Marker Sets). El estándar, “Baseline” consiste en 37 marcadores y se puede ver en “Views”->”Sekeletons”->”Choose Markerset”.



Definir un esqueleto

Hacer clic en “Layout”->”Create”. El actor para el cual se vaya a definir el esqueleto se debe colocar en el centro del volumen de grabación con los marcadores de el Marker Set adecuado.

El actor se debe colocar en “posición T” frente a las cámaras, y se mostrará un modelo cuando el actor este en la posición correcta.



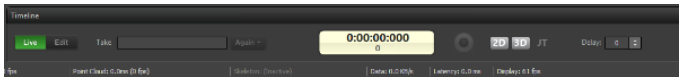
Grabar datos

Tras calibrar y definir esqueletos, seleccionamos “Layout”-> “Capture” para acceder a las opciones de grabación. Seleccionar el botón rojo de grabación y comenzar a capturar los datos 3D.

El panel de timeline funciona de manera similar que un software de grabación de audio o una cámara de video. Con el botón comenzamos a grabar en una nueva toma. El panel (botón de comenzar a grabar, detener, etc) es similar a cualquier otro software de grabación.

En la sección “Take” se da el nombre a la toma. También se muestra información como el tiempo grabado, la latencia y los “frames per second”.

Si el esqueleto pierde la forma, se puede calibrar posicionandose de nuevo en posición de T.



Sesión de edición

Abrir un proyecto existente y seleccionar “Layout”->”Edit”. Escoger alguna toma (“Takes”) para obtener la información de marcadores de esta.

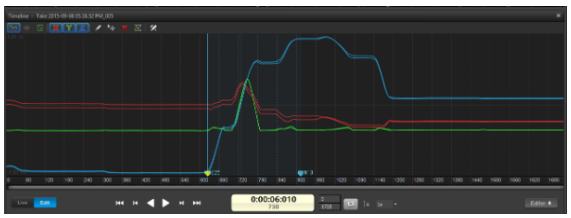
En el panel “Project”, la sección “Assets” muestra los esqueletos y cuerpos rígidos de la toma. Al seleccionar algún esqueleto, encontramos sus marcadores, y se pueden seleccionar para Editar.

Eliminar marcadores de ruido

Para cada esqueleto y cuerpo rígido se deben eliminar los marcadores que aparezcan como “Undefined”.

Seleccionar animación deseada

En el editor de “timeline” se selecciona la sección de grabación para trabajar, como se ve en la imagen:



Fill gaps

Seleccionamos marcadores con “gaps” (huecos) en sus gráficas; estos se van a llenar escogiendo con alguna interpolación.

Esto se puede hacer en “Fill Gaps” del panel “Edit Tools”. Seleccionamos el tamaño máximo de los gaps que llenaremos con una interpolación, así como el tipo de interpolación a usar (ej. Max. Gap Size = 10) que utilizaremos (ej. Interpolation = Cubic). Con la opción “Fill Selected” se llenan los gaps seleccionados.

Modelos

Encontrando los puntos en 3 dimensiones

Una vez que tenemos grabado el video a través de las cámaras, debemos encontrar las coordenadas 3D de los marcadores en base a los datos obtenidos. Para poder reconstruir la posición de un punto, este debe ser visto por al menos 2 cámaras, ya que en otro caso, no tenemos información de las 3 dimensiones. Si C1 es la posición de la cámara 1, C2 la de la segunda cámara. Estos datos son conocidos por como calibramos el sistema. Y de las imágenes podemos calcular los valores I1 y I2 que serian la posición del marcador en el plano relativo de la cámara 1 y 2 respectivamente. Una vez que tenemos estos datos definidos. Si P es la posición del marcador, entonces podemos escribir las ecuaciones.

$$C_1 + k_1(I_1 - C_1) = P\, C_2 + k_2(I_2 - C_2) = P$$

donde k1 y k2 son incógnitas, estas dos ecuaciones se pueden igualar para obtener

$$C_1 + k_1(I_1 - C_1) = C_2 + k_2(I_2 - C_2)$$

que es una ecuación con 2 incógnitas, pero recordando que son valores en R3, tenemos un sistema de 3 ecuaciones con 2 incógnitas, por lo que es posible resolverlo para encontrar la posición del marcador, con los datos.Sin embargo para datos reales, la ecuación no siempre tiene solución, ya que habrá siempre ruido que evite tener datos exactos, por lo que se encuentran dos puntos P1 y P2 que cumplen

$$C_1 + k_1(I_1 - C_1) = P_1\, C_2 + k_2(I_2 - C_2) = P_2\, (P_2 - P_1)\dot{(}I_1 - C_1) = 0\, (P_2 - P_1)\dot{(}I_2 - C_2) = 0$$

Las primeras dos condiciones nos dicen que P1 esta en la recta en la que la cámara 1 dice que esta el marcador, y que P2 en la recta en la cual la cámara 2 ve el marcador 2. Las ultimas dos condiciones nos piden que la recta que une al punto P1 y P2 sea perpendicular a las dos rectas en la que las cámaras ven el marcador. Esto nos da los puntos mas cercanos en estas rectas, una vez que encontramos estos, podemos usar el punto medio entre P1 y P2 como la posición del marcador.

Otro problema común, cuando hay varias marcadores, es saber definir cuales que marcador visto por una cámara corresponde al mismo marcador visto por otra cámara, para esto podemos checar la diferencia entre los puntos P1 y P2 generados, y si es muy grande, esto es un indicador de que no se esta observando el mismo marcador.

Guardando la información

Una vez que se ha obtenido los datos de la posición de los marcadores, y se ha usado para generar la información acerca del esqueleto y su posición a través del tiempo. debemos guardar esta información de alguna manera.

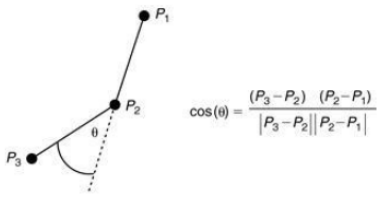
El formato mas común para esto es el formato BVH(Biovision Hierarchy), en el que se guarda primero el formato del esqueleto, esto se hace, definiendo primero una raíz, o punto de referencia, el cual suele ser ubicado en la cadera del esqueleto, a partir de el se definen las articulaciones en forma de árbol, cada articulación tiene una articulación padre, y se define su distancia original hacia su padre, lo que nos da una posición original, así como un tamaño para el hueso entre ambas articulaciones.

Una vez que tenemos definido el esqueleto, guardamos la información acerca del movimiento. de la siguiente manera. Primero definimos la cantidad de frames o imágenes que conformaran nuestra animación, y después definimos el tiempo que en segundos que ocurrirá entre cada frame. Después por cada frame guardamos información acerca de la posición del esqueleto en ese momento. Estos datos serán la posición de la raíz, y además para cada articulación, guardamos su rotación en los 3 ejes, Con esto podemos inferir la posición de cada uno de los huesos, simplemente concatenando las matrices de transformación de cada articulación con la de su padre.

Ajuste del esqueleto

Una vez que los movimientos de los marcadores se ve razonable, el siguiente paso es asignarle una estructura de esqueleto para que sea controlado por el movimiento digitalizado, sin embargo, algunos problemas surgen, por ejemplo al usar los marcadores directamente para especificar la posición es que por el ruido, el suavizado e imprecisiones, las distancias entre las articulaciones del esqueleto no se van a mantener durante el tiempo lo que puede causar un efecto de que el esqueleto “patine” o que atraviese el suelo.

Una razón de esto es que los marcadores no están exactamente en las articulaciones de la persona, si no afuera en la superficie, por lo que el punto digitalizado está desplazado de la articulación. Este desplazamiento puede ser calculado con la normal al plano formado por 3 marcadores consecutivos, consideremos el codo, si la muñeca tiene dos marcadores (usualmente para capturar la rotación) entonces la posición real de la muñeca puede ser interpolada a partir de esto y los marcadores muñeca-codo-hombro pueden ser usados para calcular la normal y así obtener la posición real del codo desplazando por una distancia previamente calculada (dependiendo del marcador y de la persona usando los marcadores) en dirección de la normal. El problema de esta solución es que cuando el brazo está extendido los marcadores son casi colineales. Ahora que las posiciones de las articulaciones son más consistentes con el esqueleto, ya pueden ser utilizadas para controlarlo. Para evitar tener coordenadas globales en el espacio se calculan las rotaciones de las articulaciones, por ejemplo, en un esqueleto jerárquico si se han capturado 3 marcadores consecutivos entonces el tercero es usado para calcular su rotación relativa a los dos anteriores



Aplicaciones

Si bien las aplicaciones del mocap más populares son alrededor del entretenimiento, su uso es de gran impacto en una gran variedad de disciplinas, desde darle movimiento a un personaje de un videojuego o película hasta ayudar en el análisis de la seguridad de un automóvil al capturar y estudiar el comportamiento de un maniquí de prueba tras un choque. Entre sus aplicaciones se encuentran

- **Entretenimiento** En el cine y la televisión, el mocap en conjunto con técnicas de VFX han sido de mucha utilidad para tener una mayor libertad creativa, además de ahorrar tiempo a los animadores y dinero a la producción. Se usa para animar personajes principales, multitudes, objetos, ciertos animales (como perros en los populares juego Assassin's Creed y Call of Duty).
- **Medicina** En esta área la captura de movimiento también es llamada 3D biological measuring o 3D motion analysis y se utiliza para generar datos biomecánicos que sirven para el análisis del caminar y tiene varias aplicaciones ortopédicas, como la mecánica articular, el análisis de la columna vertebral, el diseño de prótesis y la medicina deportiva.
- **Deporte** El mocap tiene un rol importante a la hora del análisis para mejorar el rendimiento de los atletas, pues les proporciona un modelo 3D a diferencia del video convencional en el cual se puede estudiar desde todos los ángulos e incluso comparar con la técnica de otros atletas.
- **Ingeniería** En la ingeniería tiene un uso importante a la hora del diseño, seguridad y pruebas de ergonomía, al simular la interacción humana con los modelos diseñados los ingenieros son capaces de determinar la calidad del modelo antes de construir prototipos costosos.
- **Ley** Mocap puede ser usado para generar videos sobre la reconstrucción de eventos y podrían llegar a ser usados como prueba en los juicios siempre que cumpla con los lineamientos locales y federales. Un ejemplo de esta aplicación es el video producido por Failure Analysis al recrear los eventos en el asesinato de Nicole Brown Simpson and Ronald Goldman durante el juicio del exjugador de fútbol americano O.J. Simpson.
- Seguridad y defensa La captura de movimiento ha tenido uso militar desde hace muchos años en cascos de seguimiento para operadores de vehículos de combate, en el seguimiento de la línea de visión para ayudar a apuntar armas y recibir información visual crítica. Además, se ha usado para entrenar tropas a través de la simulación y grabar sesiones de entrenamiento donde se conoce la posición exacta en el campo de entrenamiento de toda la tropa en cada instante.

Referencias

Parent, R. (2012). _Computer Animation: Algorithms and Techniques_ (3rd ed.). Morgan Kaufmann Publishers.

Menache, A. (2010). _Understanding Motion Capture for Computer Animation;Morgan Kaufmann Series in Computer Graphics_ (2nd ed.). Morgan Kaufmann Publishers.

Rahimian, P., & Kearney, J. K. (2017). Optimal Camera Placement for Motion Capture Systems. _IEEE Transactions on Visualization and Computer Graphics_, _23_(3), 1209–1221. https://doi.org/10.1109/tvcg.2016.2637334

Leung, H. (2007). _BVH motion capture data_. City University of Hong Kong - Department of Computer Science. Recuperado 18 de julio de 2022, de https://www.cs.cityu.edu.hk/~howard/Teaching/CS4185-5185-2007-SemA/Group12/BVH.html

Gleicher, M. (1999, 3 marzo). Biovision BVH. CS838 - Topics in Computer Animation. Recuperado 26 de julio de 2022, de https://research.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/BVH.html