

An Interactive Algorithm for Virtual Patient Positioning

J. Casafranca¹, A. Sújara¹, M. García¹

¹URJC Universidad Rey Juan Carlos, España
GMRV Grupor de Modelado y Realidad Virtual

Abstract

In the last years, Virtual Reality medical simulators are gaining importance, training new physicians in a safe environment. In order to improve the trainees' skills, these applications let them perform a specific medical procedure in different virtual patients with different anatomical variation. Typically, virtual patients are generated from medical imaging data sets, such as MRI (magnetic resonance imaging), CT (computed tomography) or US (ultra sound). Usually, these data are generally captured in specific subject position. This pose is different from the subjects' position required in the simulated specific medical procedure. This paper proposes a novel technique that allows adapting the virtual patient anatomy to any desired pose. Our algorithm follows a geometrically based approach with the intention of: (i) being independent of a proper mechanical description of the tissues which is rarely available and (ii) keeping our user interface running at interactive rates. We adapted the skeletal animation workflow to deal with internal anatomy models. Most of the stages of this workflow have been automated. The user intervention has been limited to the interactive pose selection process. Additionally, in order to refine the solutions provided by our geometric approach, we have designed an optimization phase to achieve more appealing results.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

1. Introduction

Hoy en día, los entrenadores basados en Realidad Virtual (RV) están ganando cada vez más popularidad. Dentro de este grupo de herramientas destacan los simuladores médicos [UGF*09, BAG*14]. La importancia de estas aplicaciones es evidente, especialmente en aquellos procedimientos médicos que requieren destrezas no cognitivas que han de adquirirse con la práctica. Los simuladores basados en realidad virtual proporcionan un entorno seguro donde entrenar a la nuevas promociones de médicos. Las mejoras en el rendimiento de los computadores actuales, el desarrollo de los dispositivos de Entrada/Salida y el avance en las técnicas de simulación basada en física en entornos interactivos permiten una transferencia efectiva de destrezas del mundo virtual al mundo real [Sey08, DWB*14, BKTN14].

La nueva generación de simuladores médicos no solo mejorará la calidad de la simulación sino que permitirá a los futuros médicos entrenar con datos anatómicos de pacientes reales [WAVH*12, VLS*13]. A partir de datos procedentes de técnicas, tales como MRI (imagen por resonancia magnética), CT (tomografía computerizada) o US (imagen capturada por ultrasonido), se podrán obtener modelos virtuales es-

pecíficos de cada paciente. Esta aproximación no está exenta de problemas: (i) a la hora de realizar muchos procedimientos médicos no se cuenta con este tipo de datos específicos de cada paciente, (ii) las técnicas de imagen médica actuales no capturan adecuadamente todos los tejidos del paciente, (iii) no es fácil extraer las propiedades mecánicas de los distintos tejidos a partir de estos datos. Por este motivo, proyectos como RASimAs (grant agreement no. 610425) optan por crear una base de datos de pacientes virtuales que recojan las variaciones anatómicas típicas.

Generalmente, las distintas técnicas de imagen médica existentes capturan datos en posiciones específicas del paciente, haciendo necesario adaptar los modelos de los pacientes virtuales, obtenidos a partir de estos datos, a la pose en la que se realizará el procedimiento médico que se vaya a simular. En este artículo, se describe un algoritmo que permite adaptar fácilmente distintos pacientes virtuales a la pose deseada, respetando los siguientes requisitos:

- La intervención del usuario se limitará a seleccionar la pose deseada.
- Dicha selección se realizará de forma interactiva.
- No se dispone de información anatómica completa. Los

tejidos modelados pueden variar arbitrariamente de un paciente virtual a otro.

- No se dispone de una descripción del comportamiento mecánico de las estructuras anatómicas modeladas.

Con el objetivo de garantizar estos requisitos, se optó por diseñar un algoritmo geométrico. Dicho algoritmo calcula de forma novedosa la influencia del movimiento de los huesos en el resto de estructuras anatómicas. En los siguientes apartados, se describen las técnicas que se han diseñado y adaptado con el fin de adecuar el flujo de trabajo de la animación esquelética a transformar las estructuras anatómicas internas de un modelo virtual. La técnica propuesta realiza la mayor parte de las etapas de forma automática en pre-proceso, permitiendo que el proceso de selección de poses se realice de forma interactiva. En post-proceso, una etapa de optimización final refina el resultado obtenido en la etapa interactiva, solventando las limitaciones de los algoritmos puramente geométricos.

Cabe destacar que, aunque nuestro objetivo es aplicar esta técnica a la creación de modelos de pacientes virtuales en poses específicas, el sistema aquí propuesto puede aplicarse a otros entornos que requieran visualizar la anatomía interna de distintos caracteres virtuales, por ejemplo, en aplicaciones destinadas a la educación, a la animación cinematográfica y efectos especiales...

2. Antecedentes

A día de hoy, la animación esquelética es la técnica más utilizada a la hora de animar el modelo virtual de un carácter articulado en el campo de los gráficos por computador. Esta técnica permite transferir las poses y movimientos de un esqueleto virtual a los vértices que componen la malla superficial del personaje que se desea animar. El número de grados de libertad del esqueleto virtual es muy inferior al número de vértices de la maya, simplificando la selección de poses. El proceso de animación se divide en cuatro etapas: (i) *rigging*, (ii) pesado, (iii) selección de la pose y (iv) *skinning*. En la etapa de *rigging*, se crea el esqueleto virtual. La etapa de pesado asocia el movimiento de los vértices de la malla superficial con el movimiento de uno o varios huesos del esqueleto virtual. Posteriormente, se definen las posiciones clave de la animación en el esqueleto virtual y, por último, se transfieren las poses seleccionadas al modelo que se desea animar (fase de *skinning*). De forma general, las tres primeras etapas se realizan de forma manual o semi-automática, dependiendo el resultado final de las dotes artísticas del usuario encargado de llevarlas a cabo. En nuestro sistema, el operador estará encargado exclusivamente de seleccionar la pose del paciente virtual.

Si bien es cierto que en la bibliografía pueden encontrarse varios trabajos orientados a automatizar la etapa de *rigging* [BTST11, PS12, MAF10], no es fácil encontrar técnicas que automaticen la etapa de pesado. Una excepción son

los trabajos de Baran et Popovic [BP07], en los que se proponen algoritmos que permiten automatizar ambas etapas. El algoritmo de *rigging* que se presenta en su publicación adapta un esqueleto virtual pre-construido al modelo virtual del personaje. Dicho proceso de adaptación, al igual que el de la mayoría de los algoritmos anteriormente citados, no tiene en cuenta la estructura ósea del personaje virtual De forma similar a la técnica que planteamos en este artículo, Baran et Popovic [BP07] utilizan la ecuación de difusión de Laplace para calcular la influencia de cada hueso sobre los vértices del modelo.

Entre las técnicas diseñadas con objeto de automatizar la fase de animación destacan las técnicas de *retargeting* como por ejemplo las que se describen en [CK99, HRE*08]. La idea principal que subyace detrás de estas técnicas es adaptar movimientos pregrabados a esqueletos virtuales específicos. La selección de poses también puede simplificarse utilizando técnicas basadas en cinemática inversa [Par07].

La etapa de *skinning* es una etapa automática. Entre las principales aportaciones en esta área, destaca el artículo seminal de Magnenat-Thalmann et al [MTLT88]. Actualmente, la técnica de *skinning* más utilizada es el *Linear Blending Skinning* (LBS). Esta técnica se ha descrito en múltiples artículos, como por ejemplo en [LCF00]. El algoritmo se basa en calcular la posición de los vértices de la malla, interpolando linealmente las matrices de transformación asociadas a los huesos del esqueleto virtual. A pesar de los problemas derivados de interpolar linealmente rotaciones definidas a partir de matrices (*candy-wrapper effect*, pérdida de volumen...), esta técnica es la más extendida gracias a que puede implementarse de forma eficiente en las arquitecturas gráficas modernas. Kavan et al en [KCvO08] solventan parte de los problemas derivados del LBS, con una pérdida de eficiencia mínima, representando las transformaciones de los huesos virtuales mediante cuaterniones duales. Esta técnica recibe el nombre de *Dual Quaternion Skinning* (DQS). La técnica de DQS, a pesar de solventar de forma efectiva el problema del *candy-wrapper effect* y prevenir la pérdida de volumen, no funciona bien en todos los casos. En [LLS*13], se propone una técnica que solventa los problemas DQS a la hora de trabajar con escalados y cizallados. En [CMT05, LT08, Hej04, LCF00], también se describen técnicas de *skinning* que intentan solventar las limitaciones de LBS, aunque en la mayoría de los casos el coste computacional es elevado y acarrear otras limitaciones. En [LT08], las articulaciones se definen por medio de *splines*, mejorando el resultado final pero dificultando la creación de un esqueleto virtual de forma automática.

Dejando a un lado la animación esquelética clásica, existen otras alternativas que podría adaptarse a nuestro problema. En los últimos años han proliferado los modelos esqueleto-musculares, los modelos biomecánicos y las técnicas basadas en la simulación física [LGK*12]. Patterson et al [PMS12] simulan de forma eficiente el comportamien-

to de los músculos, pero su técnica requiere una etapa manual larga y tediosa, conducida generalmente por un experto. La mayoría de los modelos biomecánicos se crean de forma manual o supervisada para cada anatomía específica, incluso algunos de los más detallados se centran en regiones concretas del cuerpo humano [LST09]. Fan et al proponen, en [FLP14], una técnica de animación musculoesquelética que simula físicamente el comportamiento de los músculos preservando su volumen. El modelo anatómico puede construirse de forma automática a partir de imágenes médicas, pero necesita información de la forma del músculo en reposo y de la forma del músculo deformado. Actualmente, existen otras técnicas capaces de obtener modelos esqueleto-musculares, partiendo directamente de imágenes médicas [BAGD07, GRP10, SSC*09]. En la fase de construcción de los distintos modelos mecánicos, los métodos anteriores requieren información detallada del sujeto procedente de imágenes médicas. Esta información no está siempre disponible. Además de forma general, modelos anteriores se centran principalmente en simular el comportamiento de los músculos y huesos obviando el resto de estructuras anatómicas.

Por último, destacar el trabajo reciente de Ali et al [AHLG*13]. En esta publicación, se plantea una técnica que permite transferir la anatomía interna, en diferentes poses, de un personaje virtual a otro. Los autores se centran en el proceso de transferencia y no en el de animación. Aun así, esta técnica permite realizar el proceso de pesado en el modelo de origen y transferir los pesos a los modelos finales, permitiendo al artista animar estos modelos. La principal limitación de esta técnica es la poca variabilidad anatómica que es capaz de generar. Los distintos tejidos ocupan las mismas posiciones relativas en todos los modelos. Nuestra técnica deberá poder enfrentarse a una diversidad anatómica mayor.

3. Selección de poses

Tal y como se ha explicado en la introducción, en este artículo se propone una técnica que permite adaptar la información anatómica de un paciente o personaje virtual a una determinada posición. Puesto que el proceso de selección de poses ha de ser supervisado por un artista, se ha prestado especial atención a la eficiencia del algoritmo de forma que dicho proceso pueda realizarse de forma interactiva. Las etapas más costosas del algoritmo pueden realizarse antes de comenzar a manipular el modelo en una etapa de pre-proceso.

En este trabajo se han diseñado y adaptado algoritmos que permiten adecuar la animación esquelética a la animación de los tejidos internos de un modelo virtual, limitando la intervención del usuario a la selección de poses. A diferencia de como ocurre en la animación esquelética clásica, nuestro esqueleto virtual no modifica directamente la información geométrica del modelo virtual, en su lugar define un campo de deformaciones en el interior del modelo que se utilizará para posteriormente trasladar los tejidos internos de dicho

modelo. Si bien es cierto que nuestra técnica puede trabajar con representaciones anatómicas incompletas, se ha de disponer, al menos, de la siguiente información: la superficie de la piel, los tejidos óseos y el punto de rotación de las distintas articulaciones. El proceso de selección de poses se divide en las siguientes etapas (ver Fig. 1):

- **Volumetrización:** en un primer paso se volumetriza el interior del modelo virtual utilizando una malla de tetraedros. Sobre esta malla se definirá el campo de desplazamientos asociado al movimiento de los huesos.
- **Pesado:** se calcula de forma automática la influencia de los huesos del esqueleto sobre los vértices de la malla de tetraedros.
- **Mapeado:** en esta etapa se asignan los tejidos virtuales a la malla de tetraedros obtenida en el paso anterior.
- **Selección de poses:** un operador selecciona las distintas poses transfiriéndose automáticamente el movimiento del esqueleto a la malla de tetraedros y posteriormente a los tejidos del modelo.
- **Optimización:** nuestro sistema permite refinar la solución anterior utilizando un algoritmo de simulación basado en la física que resuelve el problema estático.

Las fases de *volumetrización*, *pesado*, *mapeado* y *optimización* son las más costosas desde un punto de vista computacional. Por este motivo, las tres primeras etapas se ejecutarán en pre-proceso una vez por cada paciente virtual. Por otro lado, la fase de *optimización* se ejecutará, si el usuario lo desea, una vez por cada una de las poses seleccionadas.

3.1. Volumetrización

En esta primera etapa, se discretiza el interior del modelo virtual utilizando una malla de tetraedros. La citada malla se utilizará para calcular de forma automática y eficiente la influencia de cada hueso (ver Sec. 3.2), estimar un campo de desplazamientos (ver Sec.3.4) y transferir este campo al modelo virtual (ver Sec. 3.3).

Con el fin de poder controlar el proceso de discretización, la malla de tetraedros no se calcula directamente a partir del modelo virtual. En su lugar, en un primer paso se genera una imagen 3D a partir de los modelos de la piel y los huesos. Esta imagen permite simplificar el etiquetado de los huesos en la malla de tetraedros. Además, tal y como se explicará más adelante mejora la robustez de esta fase.

Las dimensiones de la imagen 3D dependen del tamaño de la caja contenedora del modelo virtual y del tamaño de máximo de vóxel seleccionado. En los casos probados, se ha escogido el tamaño máximo de vóxel de forma que las dimensiones de la imagen 3D sean siempre inferiores a 250x700x120. El proceso de voxelización comienza marcando los vóxeles que colisionan con la piel (Fig. 2.a). En un segundo paso, utilizando la técnica descrita en [SHS03], se etiquetan los vóxeles interiores del modelo. Después, se eliminan los vóxeles etiquetados como piel (Fig. 2.b). Este

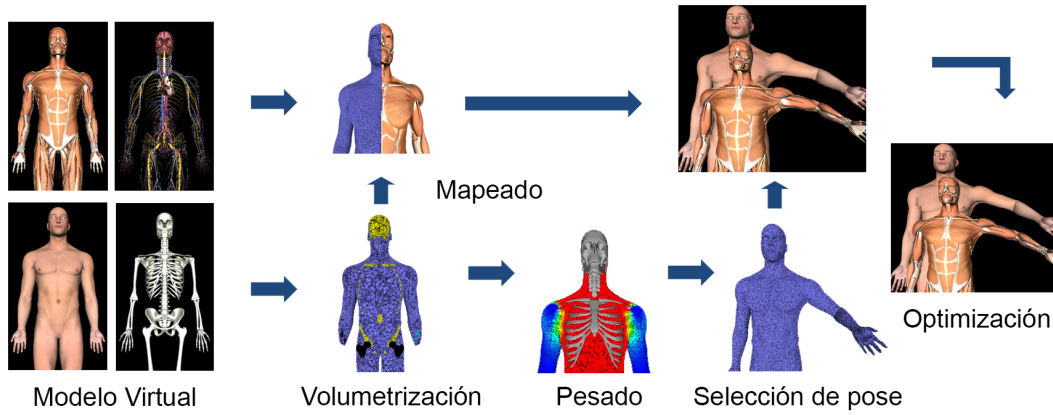


Figure 1: Visión general del algoritmo.

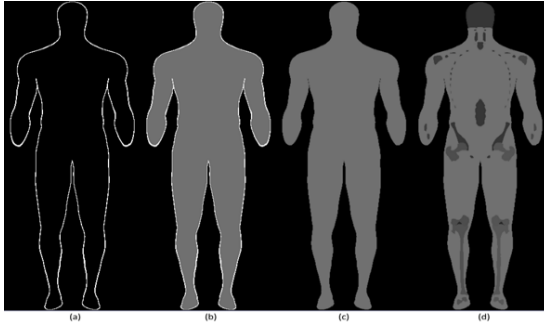


Figure 2: Cortes frontales de la imagen volumétrica en sus distintas etapas de creación.

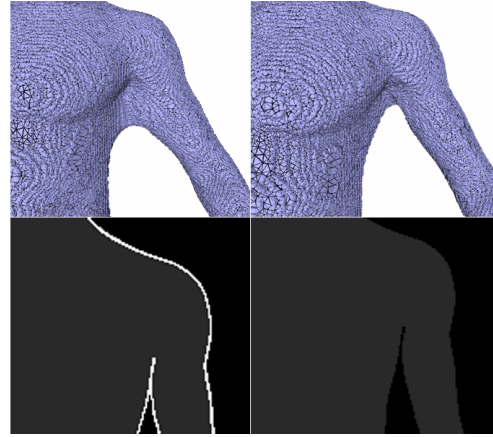


Figure 3: Resultado del proceso de tetraedrización, sin eliminar (izquierda) y eliminando la piel del modelo (derecha).

paso resalta las fronteras del modelo impidiendo que regiones disconexas aparezcan conectadas en la malla de tetraedros (Fig. 3), solventando uno de los principales problemas de la etapa de *pesado* (ver Sec. 3.2). En el siguiente paso se marcan los vóxeles que colisionan con la superficie de los huesos y utilizando [SHS03], de nuevo se etiquetan los vóxeles que componen las distintas estructuras óseas del modelo (Fig. 2.c y 2.d).

Una vez se ha generado la imagen 3D, ésta se usa para crear una malla de tetraedros (ver [JAYB14]). A la hora de configurar el algoritmo que creará la malla de tetraedros, hay que llegar a un compromiso entre precisión y eficiencia. Las representaciones con un número de tetraedros elevado son más precisas pero menos eficientes. La tetraedrización se realiza de modo que las mallas generadas tengan una mayor resolución en las fronteras de la piel y los huesos y menor en el resto de zonas. En los casos probados se ha mantenido siempre el número de tetraedros de la malla por debajo de $3,5 \times 10^6$ y el número de vértices, por debajo de 8×10^5 .

La salida de esta etapa es una malla de tetraedros con las distintas estructuras óseas etiquetadas (ver Fig. 4).

3.2. Pesado

El campo de desplazamientos en el interior de tetraedro de la malla se calcula interpolando el desplazamiento de cada uno de sus vértices. El movimiento de los vértices de la malla de tetraedros está asociado al movimiento de uno o varios huesos. Esta etapa calcula como influyen los distintos huesos en el movimiento de cada uno de los vértices de la malla volumétrica. El peso $w_{i,j}$ pondera la influencia del hueso j en el vértice i y debe cumplir las siguientes condiciones:

$$w_{i,j} \geq 0 \quad \forall i \in V \wedge \forall j \in B \quad (1)$$

$$1 = \sum_{j \in B} w_{i,j} \quad \forall i \in V \quad (2)$$

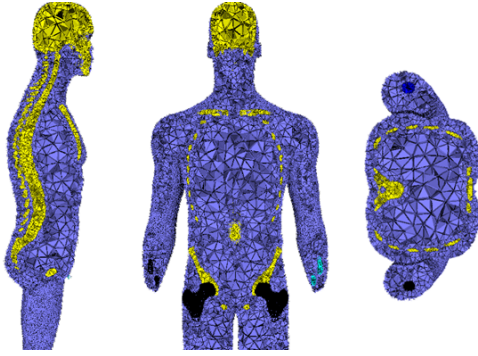


Figure 4: Distintos cortes que muestran el resultado del proceso de tetraedrización. Los huesos están etiquetados en diferente color.

donde V es el conjunto de vértices de la malla de tetraedros y B es el conjunto de huesos del esqueleto virtual.

Baran et Popovic en [BP07] explican las propiedades que deben de cumplir los pesos en un sistema de animación esquelética: las transiciones entre las áreas de influencia de distintos huesos deben de ser suaves y los pesos deben de ser independientes de la resolución y de la topología de la malla. Para garantizar estas propiedades, proponen el uso de la ecuación de difusión de Laplace. La influencia de los distintos huesos se propaga por el modelo del mismo modo en que lo haría la temperatura. La idea detrás del algoritmo que se propone en este artículo es muy similar pero hemos adaptado la formulación y la resolución, por no ser directamente aplicable a nuestro problema: (i) su algoritmo está diseñado para trabajar con mallas superficiales y (ii) las mallas deben de contener completamente al esqueleto virtual. Por otro lado, la formulación utilizada por Baran et Popovic obliga a resolver de forma completa un sistema lineal por cada hueso. En nuestro método, también tenemos que resolver un sistema de ecuaciones lineales por hueso, pero en nuestra formulación la matriz de coeficientes de los distintos sistemas permanece constante y solo varía el vector de términos independientes. Esto nos permite calcular la factorización Cholesky [GJ*10] de dicha matriz solo una vez y utilizarla para resolver los sistemas lineales de cada hueso.

La influencia W_j de un hueso j es 1 en su volumen y 0 en el resto de huesos, siendo estas las condiciones de contorno de nuestro sistema. En el equilibrio estático, el resto de puntos del volumen del modelo deberán verificar la siguiente ecuación:

$$\nabla^2 W_j = 0. \quad (3)$$

La ecuación anterior se discretiza para su resolución aplicando el Método de los Elementos Finitos (FEM) en la malla de tetraedros calculada previamente y utilizando las coord-

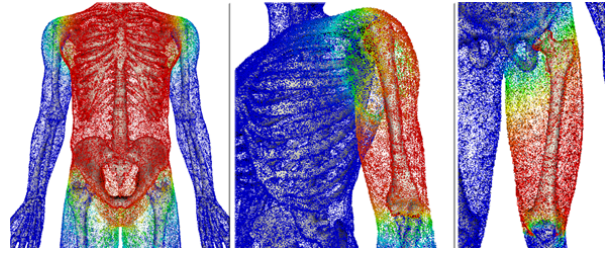


Figure 5: Zona de influencia del torso, del húmero y del fémur.

nadas baricéntricas [Cox89] de los tetraedros como función de forma (para más información ver [LNS04]). Finalmente, la ecuación 3 queda discretizada de la siguiente forma:

$$\mathbf{A}\mathbf{W}_j = \mathbf{b}_j \quad (4)$$

\mathbf{A} es la matriz de coeficientes del sistema y es constante para todos los huesos. \mathbf{b}_j es el vector de coeficientes y depende de las condiciones de contorno. \mathbf{W}_j es el vector de incógnitas y devuelve la influencia del hueso j en los vértices de la malla de tetraedros.

Por último, destacar que es fácilmente demostrable que nuestra formulación del problema garantiza que se cumplan las ecuaciones 1 y 2. El principio del mínimo (el valor mínimo de la función solo puede alcanzarse en el contorno de la misma) garantiza que se cumple la ecuación 1. Por otro lado, si consideramos $\mathbf{b} = \sum_{j=0}^n \mathbf{b}_j$ y $\mathbf{W} = \sum_{j=0}^n \mathbf{W}_j$, donde n es el número de huesos del modelo, el sistema $\mathbf{A}\mathbf{W} = \mathbf{b}$ equivale a calcular la temperatura suponiendo que todos los puntos en el contorno tienen un valor de 1. En este caso, el principio del máximo y el principio del mínimo garantizan que todos los elementos de \mathbf{W} tomarán un valor de 1, quedando probado que se cumple la condición impuesta en la ecuación 2.

La figura 5 muestra la influencia de los huesos del torso, húmero y el fémur en algunos vértices de la malla. Las zonas rojizas muestran valores cercanos a 1 y las azuladas, cercanos a 0.

3.3. Mapeado

En la siguiente sección se explicará cómo se traslada el campo de desplazamientos definido en el interior de la malla de tetraedros a las mallas que representan las distintas estructuras anatómicas. Con el fin de mejorar el rendimiento, los vértices de las mallas anatómicas se ligan con un elemento de la malla volumétrica (un tetraedro) en pre-proceso.

Desde el punto de vista matemático, determinar si un punto está dentro de un tetraedro es trivial. Desde un punto de vista computacional, el enfoque más simple tiene una complejidad $O(n^2)$. Para acelerar el proceso, la malla de tetraedros se almacena en una tabla de tipo *Spatial Hash*

[THM*03]. Si un vértice del modelo anatómico no está en el interior de un tetraedro, se utiliza la tabla hash para buscar iterativamente el tetraedro más cercano.” En una malla compuesta de 2584115 tetraedros, el mapeado de un modelo virtual con 1277325 vértices, de los cuales el 4.6% no se encuentra en el interior de la malla de tetraedros, se reduce de 4 horas 55 minutos a 36.35 segundos.

3.4. Selección interactiva de poses

En esta etapa, el usuario seleccionará la pose del esqueleto de forma interactiva y automáticamente se trasladará la pose al modelo virtual. A día de hoy, nuestra implementación permite transformar el esqueleto virtual utilizando cinemática directa, cargando una pose pregrabada o cargando una animación pregrabada. Técnicas como la cinemática inversa [Par07], la transferencia de poses descritas en [CK99] podrían aplicarse de forma directa en nuestro sistema.

Cada modificación del esqueleto se traslada primero a la malla de tetraedros y posteriormente a los tejidos del modelo virtual. Teniendo pre-calculada la influencia de los huesos del modelo virtual, los vértices de la malla de tetraedros pueden transformarse utilizando prácticamente cualquier técnica de *skinning*. Hemos decidido utilizar la técnica propuesta por Kavan *et al* [KCvO08] (DQS) puesto que solventa los principales inconvenientes de las técnicas basadas en LBS, prácticamente con su mismo rendimiento y pudiendo implementarse de forma eficiente en las arquitecturas gráficas modernas. De hecho, en nuestro sistema este paso se ejecuta en la tarjeta gráfica del computador.

El campo de desplazamiento en los puntos interiores a un tetraedro se puede calcular interpolando el valor del desplazamiento de sus vértices. En nuestra implementación, interpolamos utilizando las coordenadas baricéntricas de cada tetraedro. Dado que estamos interpolando linealmente, el campo vectorial resultante es continuo pero no así su primera derivada. La derivada del desplazamiento es constante en el interior del tetraedro, pudiendo calcularse una matriz constante para cada tetraedro con la transformación afín de todos sus puntos interiores (ver [MG04] para más información). Una vez calculada esta matriz, se utilizará para transformar la posición en reposo de los vértices de los distintos tejidos del modelo virtual asociados un determinado tetraedro, obteniendo la posición final deseada. Tanto el cálculo de la matriz de transformación de cada tetraedro como la transformación de los vértices de los tejidos virtuales se ejecutan en la tarjeta gráfica del computador.

3.5. Optimización

La salida de la etapa anterior calcula poses visualmente realistas en la mayoría de los casos. La fase *skinning* basada en DQS solventa de forma efectiva los problemas de *candy-wrapper* típicos de las técnicas basadas en LBS e impide de forma efectiva la pérdida de volumen de la malla. Aun

así, en algunas circunstancias puede producir un aumento no deseado de volumen. Nuestro sistema permite refinar la solución inicial simulando el comportamiento físico del modelo virtual. Es importante destacar que el objetivo de esta etapa no es obtener un resultado preciso. Tal y como se ha explicado anteriormente, ni se dispone de modelos anatómicos completos, ni de descripciones que permitan modelar el comportamiento mecánico de los distintos tejidos. El objetivo es mejorar el aspecto visual, mejorando la conservación de volumen.

El modelo mecánico utilizado se caracteriza por: plantear las ecuaciones del equilibrio estático, emplear el tensor de deformaciones de *Cauchy* y emplear un modelo de elasticidad isotrópico, homogéneo y lineal. Con objeto de solventar los problemas derivados de utilizar el tensor de deformaciones de *Cauchy*, las ecuaciones del modelo se resolverán utilizando la formulación co-rotacional del *Método de los Elementos Finitos*. La discretización espacial utilizada será la calculada en el paso de *Volumetrización*. Las condiciones de contorno necesarias para resolver el problema estático vienen definidas por las posiciones de los huesos. Puede encontrarse información adicional en [MG04].

El modelo elástico utilizado puede controlarse con dos parámetros: el *ratio de Poisson* y el *módulo de Young*. El *ratio de Poisson* controla la conservación de volumen y para garantizar dicha conservación se debe de tomar un valor de 0.5 (el valor real utilizado es ligeramente inferior a 0.5 para asegurar la estabilidad numérica). Se ha escogido el módulo de *Young* intentado mejorar la estabilidad del sistema. Para ello, se han analizado distintas matrices de coeficientes del sistema, obtenidas a partir de distintos valores del módulo de *Young*, quedándonos con el valor que daba como resultado la matriz de coeficientes con menor número condicionante.

La formulación co-rotacional necesita calcular las fuerzas internas (derivadas de la deformación) en una configuración no rotada, después las fuerzas se rotarán a la configuración final [MG04]. Para ello, necesita calcular la rotación final de los elementos de la malla de tetraedros. Dado que *a priori* no se conoce esta rotación, el problema estático se suele resolver de forma iterativa. La rotación de los elementos de la malla de tetraedros en la solución inicial empleada (ver Sec. 3.4) se acerca a la rotación final, reduciéndose significativamente el número de iteraciones requeridas para llegar a la solución final. En la mayor parte de los casos probados, con una iteración se obtenían resultados prácticamente indistinguibles de los resultados obtenidos con más iteraciones.

El modelo matemático planteado requiere resolver un sistema de ecuaciones lineales para conocer los desplazamientos de los vértices de la malla de tetraedros. Estos desplazamientos se utilizarán, en un segundo paso, para calcular la transformación que se debe aplicar a los vértices de un tejido virtual asociados a un determinado tetraedro (ver Sec. 3.4). La matriz de coeficientes del sistema lineal es dispersa, simétrica y definida positiva. De forma habitual, este tipo

Table 1: Modelos utilizados en los distintos test realizados. La dos primeras columnas indican la complejidad de la malla de tetraedros (número de tetraedros y número de vértices respectivamente) generada a partir del modelo virtual. La última columna indica el número de vértices del modelo virtual.

Modelo	Tetraedros	Nodos	Vértices
ZM	$2,57 \times 10^6$	553412	1477864
ZF	$2,31 \times 10^6$	492646	1466840
A	$3,41 \times 10^6$	767173	567493

de sistemas de ecuaciones se resuelven utilizando el método del gradiente conjugado [PTVF07]. La convergencia de estos métodos se acelera con el uso de pre-condicionadores (ver [HES03]) y utilizando como solución inicial una cercana a la solución final. En este trabajo, utilizamos como solución inicial la calculada en la sección 3.4, además de emplear un preconditionador de Jacobi.

4. Resultados

En este apartado se muestran los resultados obtenidos (ver figuras 6, 7 y 8) y se describen las pruebas que se han diseñado para validar la técnica propuesta en este artículo. En nuestra experimentación se han utilizado los anatómicos masculino (ZM) y femenino (ZF) de *ZygoteBody*TM [Zyg] y el modelo masculino (A) de *Anatomium*TM [Ana]. La tabla 1 resume las características de estos modelos. Las animaciones utilizadas se han descargado del *Carnegie Mellon University Motion Capture Database* [CMU]. Todos los test se han ejecutado en un PC con procesador Intel®i7-4820K @ 3.7GHz, con 16GB de RAM y con una tarjeta gráfica NVIDIA®GeForce GTX 770.

La figura 6 compara los resultados obtenidos utilizando la fase de *skinning* LBS con los obtenidos utilizando DQS. La pérdida de volumen en la imagen superior derecha (LBS) respecto de la posición en reposo es evidente. Se observa como DQS corrige esta pérdida de volumen. En la figura 7 se presenta un caso en el que DQS tiende a ganar volumen y como nuestro paso de optimización (FEM) hace frente a esta situación. Nuestra técnica define un campo de deformaciones en el interior del modelo, dotando a nuestra técnica de una gran flexibilidad a la hora de modificar cualquier estructura anatómica interna. De hecho, tal y como puede verse en la figura 8, nuestro sistema puede extenderse fácilmente para trabajar con datos volumétricos. En este caso, los vóxeles del modelo volumétrico deformado, se mapean sobre los vóxeles de la imagen volumétrica en reposo, utilizando el campo de deformaciones calculado sobre la malla de tetraedros. Hemos comprobado que si este proceso se realiza en GPU, pueden obtenerse tasas de refresco interactivas.

Principalmente por su alto contenido en agua, de forma general, los tejidos orgánicos se consideran incompresibles.

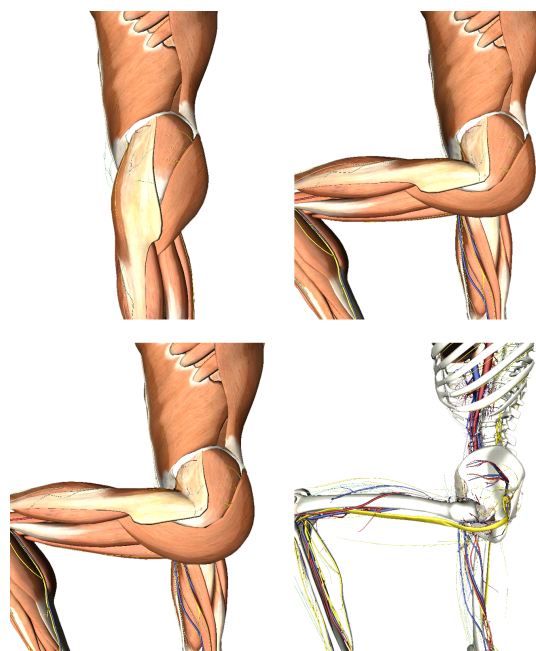


Figure 6: La imagen superior izquierda muestra el modelo en reposo, la imagen superior derecha muestra la pose obtenida con LBS, la imagen inferior izquierda muestra la misma pose obtenida con DQS y la imagen inferior derecha muestra el resultado obtenido con DQS en otros tejidos.

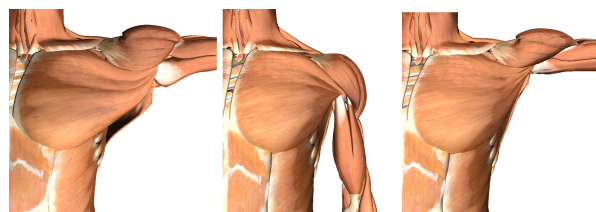


Figure 7: La imagen central muestra el modelo en reposo, la imagen de la izquierda muestra la pose obtenida con DQS y la imagen de la derecha muestra la pose obtenida después de la etapa de optimización.

De este modo, la conservación del volumen del modelo virtual es una propiedad deseable a la hora de animar tejidos orgánicos. Por este motivo, hemos comparado el comportamiento de tres técnicas de *skinning*: LBS, DQS y nuestra optimización (FEM). Para ello, hemos utilizado una animación de un ciclo de caminado [CMU]. Los resultados para los tres modelos de la tabla 1 están disponibles en la figura 9. Puede observarse como la etapa de optimización corrige los problemas de pérdida de volumen de LBS y los problemas de incremento de volumen DQS. También puede observarse que en situaciones en las que se tiende a perder volumen, DQS funciona mejor que LBS, mientras que en situaciones donde

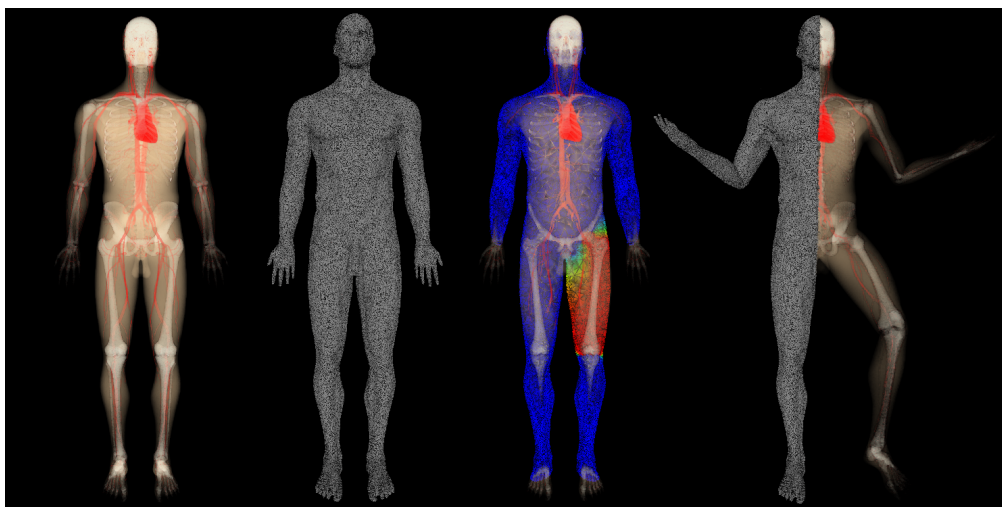


Figure 8: La primera imagen por la izquierda muestra la imagen volumétrica original. A partir de esta imagen se calcula una malla de tetraedros (segunda imagen por la izquierda). Después, se calcula la influencia de cada hueso. La segunda imagen por la derecha muestra el área de influencia del fémur de la pierna derecha. Finalmente, se transfiere la deformación de la malla de tetraedros a la imagen volumétrica (primera imagen por la derecha).

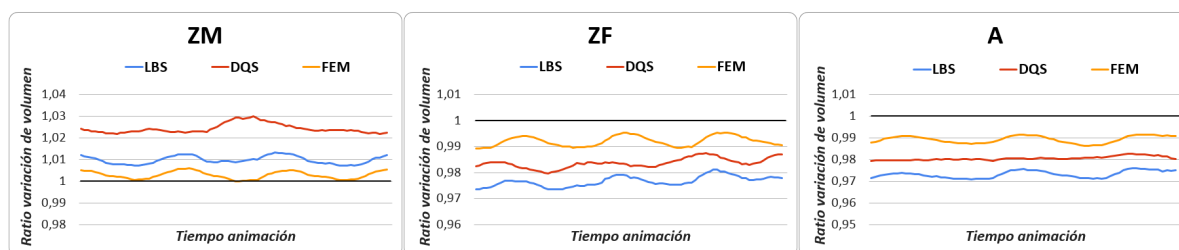


Figure 9: Ratio de conservación de volumen de las distintas alternativas de skinning probadas en los modelos descritos en la tabla 1 durante un ciclo de caminado.

se tiende a ganar volumen, LBS puede funcionar mejor que DQS.

Por último, hemos evaluado el rendimiento computacional en la etapa de pre-proceso (ver tabla 2) y en la etapa de selección de poses (ver tabla 3). En los tests realizados, el tiempo empleado en la etapa de pre-proceso es siempre inferior a 5 minutos. La etapa de selección de poses se ha dividido en dos fases: la fase interactiva y la fase de optimización. La fase interactiva incluye el *skinning* y la transferencia de la pose de la malla de tetraedros al modelo virtual. Tal y como se observa en la tabla 3, la etapa interactiva no tiene problemas para alcanzar los requisitos de interactividad requeridos. Tanto el tiempo medio empleado, como el tiempo pico durante la animación de un ciclo de caminado son siempre inferiores a 30ms. Por otro lado, la etapa de optimización no puede ejecutarse en tiempo real. Por este motivo, dicha etapa se ejecuta en post-proceso. Cabe destacar que la

Table 2: Rendimiento de la fase de pre-proceso. Tiempo en milisegundos invertido en cada una de las sub-etapas del pre-proceso de los 3 modelos utilizados. La columna rotulada como Vol. muestra los tiempos de la etapa de Volumetrización.

Modelo	Vol.	Pesado	Mapeo	Total
ZM	67641	8924	80997	157562
ZF	60047	7195	24159	91401
A	103300	17884	123104	244288

etapa de post-proceso no supera los 35s en los experimentos realizados.

5. Conclusiones y trabajos futuros

En los últimos años, la simulación esqueleto-muscular ha avanzado significativamente en el campo de los gráficos por

Table 3: Rendimiento de la etapa de selección de poses. Tiempos en milisegundos. Los tiempos medios y pico de la fase de skinning se han medido en la animación de un ciclo de caminado.

Modelo	Etapa interactiva		Optimización
	Medio	Pico	
ZM	10.48	28	20614
ZF	10.29	20	17271
A	5.56	11	30255

computador y se está haciendo un gran esfuerzo en desarrollar técnicas que permitan crear estos modelos directamente a partir de datos de pacientes reales. Una de las principales limitaciones de este tipo de técnicas es que no siempre se dispone de datos requeridos de un determinado paciente.

En este artículo se presenta una técnica que permite adecuar la anatomía interna de un modelo virtual a la pose deseada. Esta técnica extiende el paradigma de la animación esquelética a los tejidos internos de un modelo virtual. Destaca por ser independiente de las estructuras anatómicas modeladas y no necesita conocer el comportamiento dinámico de dichas estructuras. La técnica es tan flexible que puede utilizarse para animar cualquier imagen volumétrica (como, por ejemplo, datos procedentes de MRI), siempre y cuando el tejido óseo esté correctamente segmentado.

Nuestra técnica propone una aproximación geométrica al problema. Esta aproximación plantea una solución heurística. Por este motivo, esta técnica no es adecuada para planificación quirúrgica. Pero sí podría aplicarse al campo del entrenamiento, donde el resultado no tiene por que corresponderse con un paciente real. Basta con que la posición de las estructuras anatómicas sea coherente con algún grupo tipo. En el futuro, comprobaremos si las destrezas adquiridas en este tipo de pacientes pueden transferirse al mundo real. Nuestro trabajo también se adecuaría a aplicaciones en el campo del ocio (videojuegos, cine...), donde un resultado plausible es suficiente.

Asimismo, para mitigar las limitaciones de nuestra fase de *skinning*, se ha implementado una etapa de optimización en post-proceso. Esta etapa hace uso de un modelo matemático basado en la mecánica de medios continuos. A pesar de ello, este modelo no tiene en cuenta las propiedades mecánicas de los tejidos manipulados y se utiliza exclusivamente para garantizar la conservación de volumen. Estamos estudiando la posibilidad de mejorar el resultado de nuestra simulación añadiendo información mecánica de los tejidos, incluyendo información de nuevas estructuras anatómicas en la malla volumétrica y utilizando un modelo matemático más preciso.

El rendimiento de la fase de *skinning* permite alcanzar los requisitos de interactividad requeridos. Por el contrario, las etapas de pre-proceso y post-proceso no son interactivas. A

pesar, de ello son suficientemente rápidas como para no representar un hándicap a la hora de utilizar la herramienta. La etapa de pre-proceso puede realizarse una única vez por modelo, mientras que la etapa de post-proceso se realiza una vez por pose. Por este motivo, actualmente estamos trabajando en mejorar la eficiencia del post-proceso. Creemos que una implementación de esta etapa en el procesador gráfico del sistema permitiría ejecutarla interactivamente.

Tal y como se ha explicado en la introducción, el objetivo principal de este trabajo es adaptar la anatomía de distintos pacientes tipo a poses requeridas en distintos procedimientos quirúrgicos. En un futuro próximo se estudiará si los modelos obtenidos con esta técnica permiten una transferencia adecuada de destrezas del entorno simulado al entorno real.

6. Agradecimientos

Este trabajo ha sido parcialmente financiado por las siguientes entidades: el Ministerio Español de Economía y Competitividad (a través del proyecto Cajal Blue-Brain miembro de la iniciativa Blue Brain dirigida por el EPFL y el proyecto TIN2014-57481-C2-2-R) y la Comisión Europea (a través de los proyectos HBP: Human Brain Project - FET Flagship HBP604102 y RaSimAS: Regional Anaesthesia Simulator and Assistant - FP7-ICT 610425). También queremos agradecer al *Carnegie Mellon University* por darnos la oportunidad de utilizar sus datos de captura de movimiento.

References

- [AHLG*13] ALI-HAMADI D., LIU T., GILLES B., KAVAN L., FAURE F., PALOMBI O., CANI M.-P.: Anatomy transfer. *ACM Trans. Graph.* 32, 6 (Nov. 2013), 188:1–188:8. 3
- [Ana] ANATOMIUM: 21st century solutions ltd. <http://www.anatomium.com/>. 7
- [BAG*14] BAYONA S., AKHTAR K., GUPTA C., EMERY R. J., DODDS A. L., BELLO F.: Assessing performance in shoulder arthroscopy: The imperial global arthroscopy rating scale (igars). *The Journal of Bone & Joint Surgery* 96, 13 (2014), e112. 1
- [BAGD07] BLEMKER S. S., ASAKAWA D. S., GOLD G. E., DELP S. L.: Image-based musculoskeletal modeling: Applications, advances, and future opportunities. *Journal of magnetic resonance imaging* 25, 2 (2007), 441–451. 3
- [BKTN14] BUCKLEY C. E., KAVANAGH D. O., TRAYNOR O., NEARY P. C.: Is the skillset obtained in surgical simulation transferable to the operating theatre? *The American Journal of Surgery* 207, 1 (2014), 146–157. 1
- [BP07] BARAN I., POPOVIĆ J.: Automatic rigging and animation of 3d characters. *ACM Trans. Graph.* 26, 3 (July 2007). 2, 5
- [BTST11] BHARAJ G., THORMÄHLEN T., SEIDEL H.-P., THEOBALT C.: Automatically rigging multi-component characters. *Comp. Graph. Forum (Proc. Eurographics 2012)* 30, 2 (2011). 2
- [CK99] CHOI K.-J., KO H.-S.: On-line motion retargetting. *Journal of Visualization and Computer Animation* 11 (1999), 223–235. 2, 6
- [CMT05] CORDIER F., MAGNENAT-THALMANN N.: A data-driven approach for real-time clothes simulation. *Computer Graphics Forum* 24, 2 (2005), 173–183. 2

- [CMU] CMUMCD: Carnegie mellon university motion capture database. <http://mocap.cs.cmu.edu/>. 7
- [Cox89] COXETER H.: *Introduction to Geometry*. Wiley Classics Library. Wiley, 1989, ch. Barycentric Coordinates. 5
- [DWB*14] DAWE S. R., WINDSOR J. A., BROEDERS J. A., CREGAN P. C., HEWETT P. J., MADDERN G. J.: A systematic review of surgical skills transfer after simulation-based training: laparoscopic cholecystectomy and endoscopy. *Annals of surgery* 259, 2 (2014), 236–248. 1
- [FLP14] FAN Y., LITVEN J., PAI D. K.: Active volumetric musculoskeletal systems. *ACM Trans. Graph.* 33, 4 (July 2014), 152:1–152:9. 3
- [GJ*10] GUENNEBAUD G., JACOB B., ET AL.: Eigen v3. <http://eigen.tuxfamily.org>, 2010. 5
- [GRP10] GILLES B., REVERET L., PAI D.: Creating and animating subject-specific anatomical models. *Computer Graphics Forum* 29, 8 (2010), 2340–2351. 3
- [Hej04] HEJL J.: *Game Programming Gems 4*. GAME PROGRAMMING GEMS SERIES. Charles River Media, 2004, ch. Hardware skinning with quaternions. 2
- [HES03] HAUTH M., ETZMUSS O., STRASSER W.: Analysis of numerical methods for the simulation of deformable models. *The Visual Computer* 19, 7–8 (2003), 581–600. 7
- [HRE*08] HECKER C., RAABE B., ENSLOW R. W., DEWEESE J., MAYNARD J., VAN PROOIJEN K.: Real-time motion retargeting to highly varied user-created morphologies. *ACM Trans. Graph.* 27, 3 (Aug. 2008), 27:1–27:11. 2
- [JAYB14] JAMIN C., ALLIEZ P., YVINEC M., BOISSONNAT J.-D.: *CGALmesh: a Generic Framework for Delaunay Mesh Generation*. Research Report RR-8256, INRIA, Jan. 2014. 4
- [KCvO08] KAVAN L., COLLINS S., ŽÁRA J., O’SULLIVAN C.: Geometric skinning with approximate dual quaternion blending. *ACM Trans. Graph.* 27, 4 (Nov. 2008), 105:1–105:23. 2, 6
- [LCF00] LEWIS J. P., CORDNER M., FONG N.: Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2000), SIGGRAPH ’00, ACM Press/Addison-Wesley Publishing Co., pp. 165–172. 2
- [LGK*12] LEE D., GLUECK M., KHAN A., FIUME E., JACKSON K.: Modeling and simulation of skeletal muscle for computer graphics: A survey. *Foundations and Trends® in Computer Graphics and Vision*, 7 (2012), 229. 2
- [LLS*13] LEE G. S., LIN A., SCHILLER M., PETERS S., MCLAUGHLIN M., HANNER F.: Enhanced dual quaternion skinning for production use. In *ACM SIGGRAPH 2013 Talks* (New York, NY, USA, 2013), SIGGRAPH ’13, ACM, pp. 9:1–9:1. 2
- [LNS04] LEWIS R., NITHIARASU P., SEETHARAMU K.: *Fundamentals of the Finite Element Method for Heat and Fluid Flow*. Wiley, 2004. 5
- [LST09] LEE S.-H., SIFAKIS E., TERZOPOULOS D.: Comprehensive biomechanical modeling and simulation of the upper body. *ACM Trans. Graph.* 28, 4 (Sept. 2009), 99:1–99:17. 3
- [LT08] LEE S.-H., TERZOPOULOS D.: Spline joints for multi-body dynamics. *ACM Trans. Graph.* 27, 3 (Aug. 2008), 22:1–22:8. 2
- [MAF10] MILLER C., ARIKAN O., FUSSELL D.: Frankenrigs: Building character rigs from multiple sources. In *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, 2010), I3D ’10, ACM, pp. 31–38. 2
- [MG04] MÜLLER M., GROSS M.: Interactive virtual materials. In *Proceedings of Graphics Interface 2004* (School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2004), GI ’04, Canadian Human-Computer Communications Society, pp. 239–246. 6
- [MTLT88] MAGNENAT-THALMANN N., LAPERRIÈRE R., THALMANN D.: Joint-dependent local deformations for hand animation and object grasping. In *Proceedings on Graphics Interface ’88* (Toronto, Ont., Canada, Canada, 1988), Canadian Information Processing Society, pp. 26–33. 2
- [Par07] PARENT R.: *Computer Animation, Second Edition: Algorithms and Techniques*, 2nd ed. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2007. 2, 6
- [PMS12] PATTERSON T., MITCHELL N., SIFAKIS E.: Simulation of complex nonlinear elastic bodies using lattice deformers. *ACM Trans. Graph.* 31, 6 (Nov. 2012), 197:1–197:10. 2
- [PS12] PANTUWONG N., SUGIMOTO M.: A novel template-based automatic rigging algorithm for articulated-character animation. *Computer Animation and Virtual Worlds* 23, 2 (2012), 125–141. 2
- [PTVF07] PRESS W. H., TEUKOLSKY S. A., VETTERLING W. T., FLANNERY B. P.: *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, 3 ed. Cambridge University Press, New York, NY, USA, 2007. 7
- [Sey08] SEYMOUR N. E.: Vr to or: a review of the evidence that virtual reality simulation improves operating room performance. *World journal of surgery* 32, 2 (2008), 182–188. 1
- [SHS03] SUZUKI K., HORIBA I., SUGIE N.: Linear-time connected-component labeling based on sequential local operations. *Comput. Vis. Image Underst.* 89, 1 (Jan. 2003), 1–23. 3, 4
- [SSC*09] SCHMID J., SANDHOLM A., CHUNG F., THALMANN D., DELINGETTE H., MAGNENAT-THALMANN N.: Musculoskeletal simulation model generation from mri data sets and motion capture data. In *Recent advances in the 3D Physiological Human*. Springer, 2009, pp. 3–19. 3
- [THM*03] TESCHNER M., HEIDELBERGER B., MÜLLER M., POMERANETS D., GROSS M.: Optimized spatial hashing for collision detection of deformable objects. *Proceedings of Vision, Modeling, Visualization VMV’03* (2003), 47–54. 6
- [UGF*09] ULLRICH S., GROTTKE O., FRIED E., FROMMEN T., LIAO W., ROSSAINT R., KUHLEN T., DESERNO T. M.: An intersubject variable regional anesthesia simulator with a virtual patient architecture. *International journal of computer assisted radiology and surgery* 4, 6 (2009), 561–570. 1
- [VLS*13] VOTTA E., LE T. B., STEVANELLA M., FUSINI L., CAIANI E. G., REDAELLI A., SOTIROPOULOS F.: Toward patient-specific simulations of cardiac valves: state-of-the-art and future directions. *Journal of biomechanics* 46, 2 (2013), 217–228. 1
- [WAVH*12] WILLAERT W., AGGARWAL R., VAN HERZEELE I., CHESHIRE N., VERMASSSEN F.: Recent advancements in medical simulation: Patient-specific virtual reality simulation. *World Journal of Surgery* 36, 7 (2012), 1703–1712. 1
- [Zyg] ZYGOTE BODY: Zygote media group. <https://www.zygotebody.com/>. 7