A visualization framework for the analysis of neuromuscular simulations

Nicolas Pronost · Anders Sandholm · Daniel Thalmann

Abstract We present a visualization framework for exploring and analyzing data sets from biomechanical and neuromuscular simulations. These data sets describe versatile information related to the different stages of a motion analysis. In studying these data using a 3D visualization approach, interactive exploring is important, especially for supporting spatial analysis. Moreover, as these data contain many various but related elements, numerical analysis of neuromuscular simulations is complicated. Visualization techniques enhance the analysis process, thus improving the effectiveness of the experiments. Our approach allows convenient definitions of relationships between numerical data sets and 3D objects. Scientific simulation data sets appropriate for this style of analysis are present everywhere motion analysis is performed and are predominant in many clinical works. In this paper, we outline the functionalities of the framework as well as applications embedded within the OpenSim simulation platform. These functionalities form an effective approach specifically designed for the investigation of neuromuscular simulations. This claim is supported by evaluation experiments where the framework was used to analyze gaits and crouch motions.

Keywords Scientific visualization · Biomechanics and neuromuscular simulation · Motion analysis · OpenSim software

N. Pronost (✉) · A. Sandholm · D. Thalmann
Virtual Reality Lab, École Polytechnique Fédérale de Lausanne,
Lausanne, Switzerland
e-mail: nicolas@cs.uu.nl

N. Pronost
Games and Virtual Worlds, Utrecht University, Utrecht,
The Netherlands

1 Introduction

Effective visualization of biomechanical and neuromuscular simulations is a complex problem, especially as it involves many different types of data in large quantities. The increasing use of neuromuscular models in biomechanics and motion analysis shows the need to explore and investigate frameworks. Several important visualization challenges arise from working with the data sets collected with such technologies.

The first challenge in visualization and analysis of neuromuscular data sets is to highlight the spatial relationships that are present. Motion analysis aims at producing simulated data sets, but spatial components are also needed in order to relate them within the space-time context. Thus, effective 3D spatial understanding is an important feature of visualization systems appropriate for such data.

The second challenge in the analysis of neuromuscular simulations lies in the fact that simulations contain several very different stages, each using and producing their own data sets. Such versatile information is impossible to explore in a single specific way. Trade-off between data-driven information visualization and generic scientific visualization is then necessary.

Creating visual representations of simulated results that support clinically relevant conclusions is also challenging. The most common method for depicting time-dependent output is the two-dimensional plot, which can only depict one parameter at a time, even though data are often correlated in three or more dimensions. For complex research questions this may result in a large number of plots. For instance, muscular control estimation, which is often the final simulation stage, may work on several hundreds elements [1]. Moreover, each element may contain more than one type of information to analyze (activation, force, fiber

length) and each of them can be relevant for the evaluation of the simulation. We believe that plots are in such cases only functional when exact numerical values and relations are required, which is not always the case during the evaluation process where alternative representations may be more informative.

This inspired us to create a framework for the exploration and the analysis of complex and versatile information from neuromuscular simulations. Our system is further differentiated by the fact that it easily defines relationships between the data and visual representations directly within the simulation environment. We tried, in collaboration with biomechanics researchers, to propose a framework that is generic enough to cope with a number types of useful evaluative information. We propose visualization tasks for image-based data sets, multi-dimensional numerical values, external geometry based simulations and multi-simulation comparison.

The remainder of this paper is structured as follows. In Sect. 2 we provide some background on the nature of the data and scientific application discussed in this work. In Sect. 3 we discuss existing literature on biomechanical data visualization. In the subsequent section we describe our visualization framework in detail, followed by more specifics of the driving applicative experiment. Finally, we discuss the contributions of our system and conclude with a prospect on future work.

2 Background in application area and data

The framework presented here is likely to apply to exploration and analysis of neuromuscular simulations of interest to biomechanics communities and clinicians. The example application driving the work in this paper concerns the full process of motion analysis from the subject-specific model creation to the multiple evaluation of muscular activities.

Today the field of neuromuscular simulations is used to understand the underlying dynamics of living beings' movement, from gait research, treatment of patient with gait problem, to the teaching of physicians and the development of ergonomic furniture. During the last years several platforms have been developed, from commercial tools [1, 2] to open-source based solutions [3] and pure in-house solutions. The expansion of this field has also allowed the accessibility to neuromuscular models that are capable to describe different levels of complexity [2, 4]. This development of detailed models and simulation tools has given researchers and physicians powerful tools to create advanced simulations and even to execute different what-if scenarios or to evaluate different simulation results before the physical treatment has even started. The standard process to fulfill a simulation starts from the model that is scaled to the motion data.

Then inverse kinematics and inverse dynamics can sequentially be applied. Finally, muscle control is performed and usually compared to electromyography recording. Each step is crucial, and cross-validations and clinical relevant observations are ideally performed to confirm the intermediate results. Research questions require analysis of versatile and numerous data. For example: How much the model differs from the actual studied human being? How does the muscular activation change over time and how does it fit the actual electromyography recording? What feedback do we get from simulations at other scales? How sensitive is a specific parameter to the simulation?

Biomechanics researchers and clinicians begin their study by collecting data from experimental trials in labs. In our application, Magnetic Resonance Imaging (MRI) of the lower limb was first realized on several subjects. In close collaboration with radiologists, adequate protocols for the imaging of soft and bony tissues were defined (1.5T Philips Medical Systems machine, axial 2D T1 Turbo Spin Echo, TR/TE = 578/18 ms, FOV/FA = 40 cm/90°, matrix/resolution = $512 \times 512/0.78 \times 0.78$ mm, thickness: 2 mm). These images were registered and segmented [5] to produce a musculoskeletal model of the subjects. Then, standing, walking and crouching motions were acquired using eight ProReflex Qualisys IR cameras (200 Hz) synchronized with two AMTI ground reaction force plates (2000 Hz) and eight Noraxon EMG sensors (2000 Hz). The OpenSim simulation platform [3] was used as neuromuscular platform to analyze the motions. The MotionLab toolbox [6] was used to preprocess these data and to prepare each step of the neuromuscular simulations. The FEBio platform [7] was used to create muscular deformation simulations using the subject-specific segmented data and analyzed motions. All these data constitute the source for the visualization framework presented in this paper.

3 Related work

Much of the research on biomechanical analysis finds its origins in gait analysis, the study of locomotion [8]. Many improvements to the standard 2D plot originate from this field. For example, by adding a third dimension to the plots [9] or a color coding [10], additional parameters can be visualized. Such techniques may although unnecessarily complicate the interpretation of the data and are not suitable for complex and versatile information visualization as well as non-plot-based viewable data like images and volumes.

Solutions to visualize multiple sequences and relationships between them have then been proposed. For example, Keefe et al. [11] present a multi-view framework that explicitly supports visualization of multiple related motion sequences. Similarly Krekel et al. [12] target the visualization of the relationships between multiple connected joints

by combining interactive filtering and multiple visualization techniques. These approaches are especially tailored for the visualization of multi-simulations and complex correlated data sets, but are not suitable for the visualization of versatile biomechanical information.

Several systems for 3D visualization of biomechanical motion data have been developed and now benefit from animated and interactively controlled 3D graphics. A primary function of these systems is providing a view of anatomical features (e.g. bones, muscles) positioned appropriately in 3D space. Almost all of these systems also support some form of motion playback, often with some interactive support for adjusting camera parameters and playback speed. Beyond simply replaying experimentally captured data, 3D visualization systems also provide for visualization of derived data. For example, computing and viewing the position of helical axes to describe the motion of one bone relative to another is a technique that is gaining popularity within the biomechanics community [13, 14] providing insight into the rotation and translational components of a complex motion. Other examples of 3D visualization of biomedical data include applying color maps to bone surfaces [15] or muscle surfaces [16], and drawing 3D data glyphs and tracer curves to indicate paths of anatomical features [17]. Our framework employs such techniques as our investigation and is not on the introduction of novel visualization techniques but rather on how to employ them to allow exploration and analysis of neuromuscular simulations directly embedded within the simulation platform.

However, as researchers in information visualization have pointed out, 2D visualizations can also be effective tools for multi-dimensional data analysis, especially for identifying trends over time. This is one of the reasons why we integrate our framework into an existing simulation platform that already possesses these functionalities. Integrating 3D and 2D visualization techniques is particularly appealing because each brings a unique strength that compensates for the other's weakness. Analyzing motion trends using only 3D visualization tools imposes a high cognitive load on the user, since analysis often requires comparisons between multiple detailed motions that the user must keep in his working memory [18]. On the other hand, when analyzing motion using only 2D visualizations, the abstract representations of the motion data do not provide the necessary context for the user to understand the 3D structure of the object or its movement in space. We believe that neuromuscular simulation analysis can benefit from such 2D/3D complementarity visualizations, and we hope that this paper contributes to the development of such framework.

4 Simulation visualization framework

The main function of the visualization framework is to create relationships between data from neuromuscular simulations and visual representations. These relationships are defined by an external formal description that is editable and initially read by the system. Each visualization has its own set of parameters. Users just have to edit the description file to match their simulations. We describe in the two following sections the concept of the framework and the functionalities that have been implemented so far.

4.1 Framework concept

The framework has been developed around two major components: the tasks and the performers (see Fig. 1).

The *tasks* define which data need to be studied, like marker errors or muscle activations. These values correspond to inputs retrieved from a storage file of a simulation stage or directly referenced in the description file. A task knows how to use the information provided and how to execute the visualization process. A task can be static, called *visual task* where the data do not evolve over time, or time-dependent, called *animated task* where the view is updated with the current motion state. Tasks are launched and ended from the GUI. Each task uses a performer to affect the view by modifying current components (like meshes' properties) or adding new elements (like streamlines).

The *performers* execute the visualization. The tasks tell the performers which input(s) they are submitting to them. This is how performers expose which inputs are compatible

Fig. 1 Concept of our visualization framework. Tasks and performers are the major components involved in the visualization process
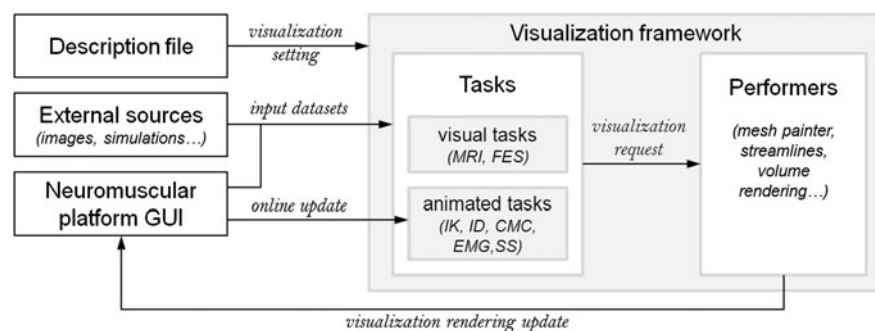
with to the GUI. The user chooses whether to enable the performer and make parameterizations. For example, the mesh painter performer knows that it will receive a scalar information, have to convert it into color and opacity coding on a 3D geometry and finally render it.

Tasks and performers are entities decoupled from the neuromuscular simulations for two reasons. The first one is that we wanted to have the possibility to visualize data that are not directly linked to a neuromuscular simulation stage (such as the *MRI* and the *FES* tasks, see Sect. 4.2). The second reason is that we wanted to be able to post-process the visualization, i.e. to load off-line the data to visualize in the GUI. This is required as users usually first run their simulations without any graphics feedback to speed up the process.

4.2 Functionalities

In this section we describe the functionalities of the framework which currently provides seven visualization tasks. Each task has practical applications in the neuromuscular analysis of motions.

4.2.1 Magnetic resonance imaging

One of the first step is to scale a generic neuromuscular model to the current subject or patient. Several models are available in the literature [2, 19, 20] that are usually created as a mean subject from cadaveric data. Scaling a generic model requires to locate anatomical landmarks on the living subject. The uncertainty of such localizations plus the subject-specific anatomical properties imply modeling errors than are impossible to reduce without further information on the subject. To overcome this problem, visualization of magnetic resonance images (MRI) can be proposed. For example, atlas-based [21] and segmentation-based [22]

modeling techniques of action lines allow the creation of scaled models that better correspond to the subject-specific data.

In this task we propose to visualize and manipulate MRI within the scaled model view (see Fig. 2). Three orthogonal slicing planes are created and the user can move them interactively inside the image volume. The window/level of the images can also be adjusted. In order to align up the posture of the neuromuscular model and the posture of the subject during the acquisition, a marker set describing anatomical landmarks in the MRI volume is used. An inverse kinematics process is then executed to align up the posture of the model.

4.2.2 Inverse kinematics

As each sequential step of a neuromuscular simulation relies on the results of the previous step, the first ones influence greatly the overall results. After scaling, the next step consists in tracking the kinematics of the motion. Indeed, the next steps need to solve the laws of motion that rely on the angle-based expression of the motion, not on marker trajectories in space. This process is usually realized by an inverse kinematics (IK) solver, where degrees of freedom (DoF) are evaluated to minimize the weighted distances between virtual markers attached to the body parts and experimental markers from motion capture. The parameterization of the cost functions and the choice of the solver depends on the motion and subject that are analyzed. The resulting errors between the experimental markers and the virtual markers during the tracking are often critical in the evaluation of a simulation [23]. To have a method to visualize quickly these errors will enable to point out the best and worth markers and so improve the simulation in a more interactive way.

In this task, our framework computes the errors between the experimental markers and the virtual markers associ-
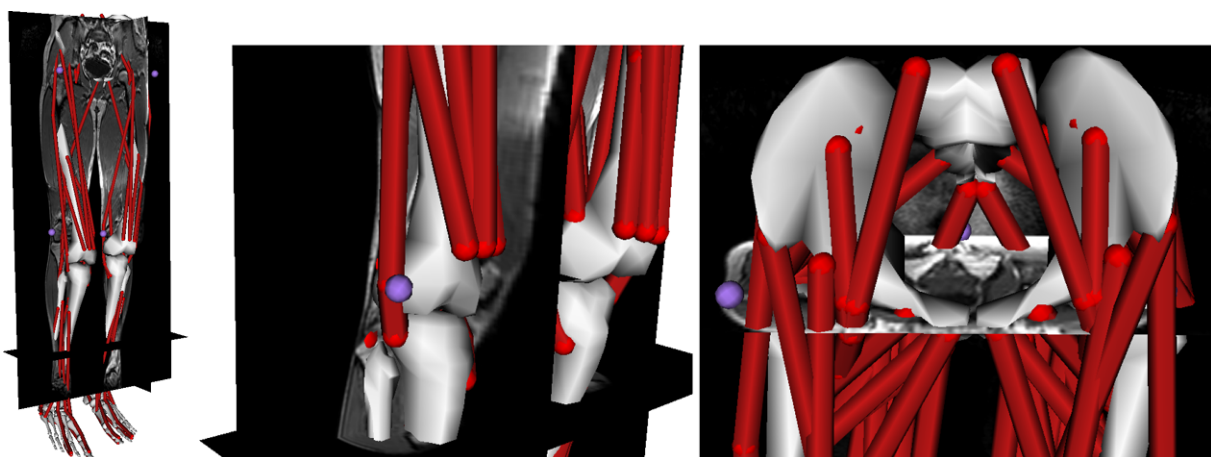

Fig. 2 The *MRI* task. 3D exploration of MRI and model scaling analysis are possible using this task. *Left*: a general view of the lower limb; *middle*: a close-up view of the knee area; *right*: a close-up view of the pelvis area
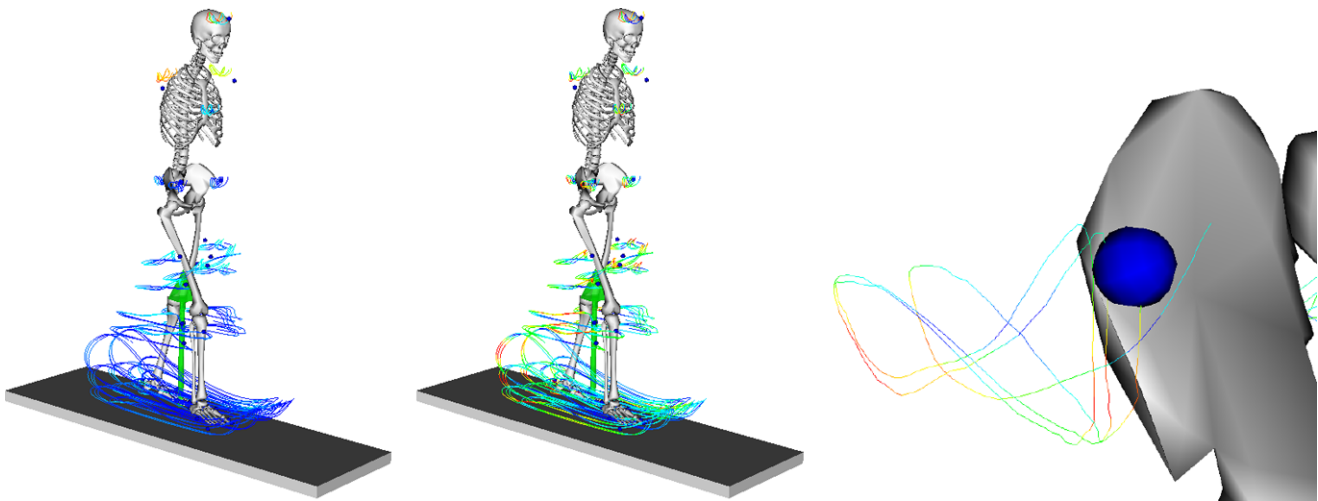
Fig. 3 The *IK* task. IK tracking analysis is possible using this task, here on a gait motion. *Left*: global marker errors are displayed so the comparison between markers is facilitated; *middle*: local marker errors are displayed so the comparison over time of each marker is facilitated; *right*: a close-up view on the streamline representing the tracking of a marker on the pelvis

ated with a motion and display them using a streamline performer. A streamline indicates the path followed by a virtual marker while its colors represent the error to the experimental marker trajectory (see Fig. 3). The user can manipulate online how many segments the streamlines will contain, from 1 (average error over the whole motion) to the number of frames (producing one color per frame). The user can also choose to display the local error for each specific marker. Then each streamline (i.e. each marker) will have its own minimum and maximum values.

4.2.3 Inverse dynamics

At some point, researchers and clinicians consider the dynamics of the motion. Forces and moments actuating the human body will be investigated. Inverse dynamics methods are then employed. The principle consists in solving the second fundamental law of motion according to the external forces applied on the human body, and the mass, inertia and angular acceleration of its body parts. Visualizing forces and moments over time using plots can be tedious when a user wants at the same time to know the corresponding motion phase (typically in the gait cycle), or when the number of data (related to the number of DoF) is large.

In this task, we visualize forces and moments directly on the model while the motion is played, so the user can save time to analyze simulation results from an inverse dynamics analysis. We demonstrate here the visualization of moments around rotational DoF by arrows-glyphs directly in the model view (see Fig. 4). The direction of an arrow lies along the rotational axis of the corresponding DoF and its size represents the magnitude of the moment. Its location is the center of the corresponding joint. A user-defined scale

factor indicates the ratio between the size of the arrows (in meter) and the magnitude of the moment (in N·m). The user can also interactively (de)select the visualized moments.

4.2.4 Computer muscle control

The most valuable stage of a neuromuscular simulation is the computation of the muscular activities. Usually either static optimization or computer muscle control algorithms are used for that purpose. A full-body model can integrate several hundreds of muscles [1] composed themselves of several action lines. A thorough 2D plot investigation of resulting data sets requires time that users can most of the time not afford. 3D visualization provides a suitable interface for the exploration and the analysis of these data sets [16].

In this task, our framework can display muscular forces or activations on 3D surfaces using color and opacity properties. Some simulation platforms [3] allow the visualization of activations directly on action line representations. As the data are related to muscles, a better way would be to visualize the results on the muscle geometries (see Fig. 5). These geometries can typically be generated from MRI-based segmented data sets. To associate the muscle geometries to the corresponding activations or forces, our framework defines assignments in the configuration file. The goal of the assignments is to create relationships between 3D geometries of body parts of the current model and action lines that exist in the model. In each assignment, several action lines can be assigned to several geometries. When several geometries are specified, their visualization properties will be duplicated. When several action lines are specified, the user can define weights. Let a be the activation value for the muscle M the user want to visualize, $a_1, \ldots, a_n$ the activation values of the action lines modeling M, and $w_1, \ldots, w_n$ their
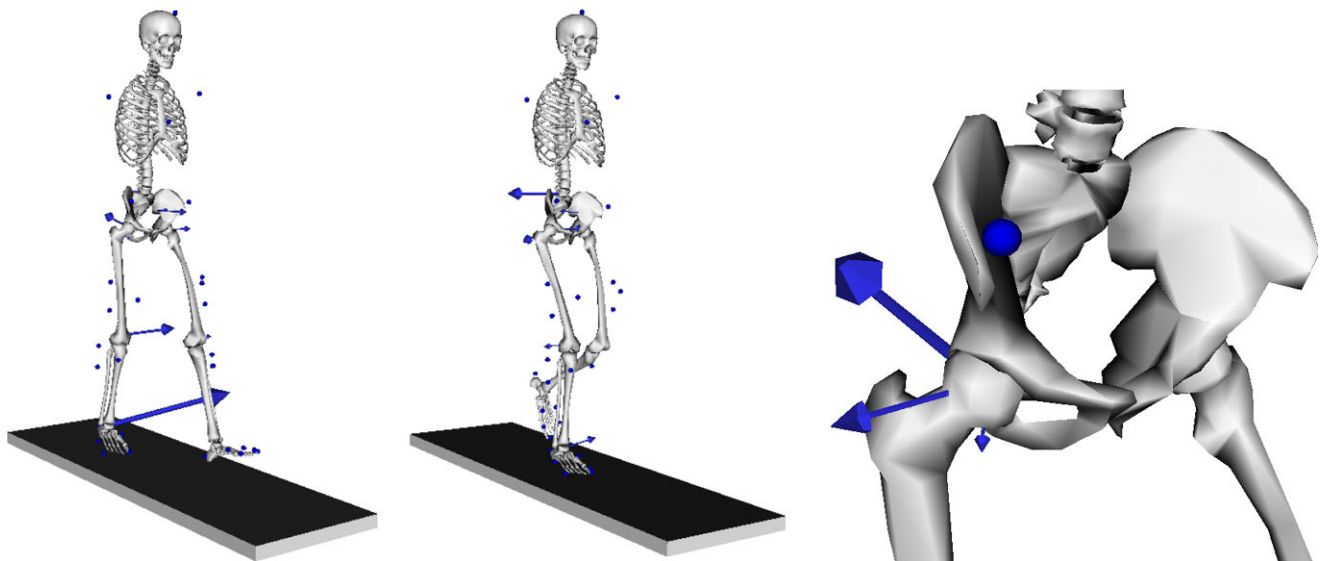
Fig. 4 The *ID* task. Forces and moments are visualized. Direction and magnitude of the glyphs give intuitive information on the dynamics of the motion. *Left* and *middle*: two full-body moment visualizations at different gait time; *right*: a close-up view on the three moments around the right hip during stance phase
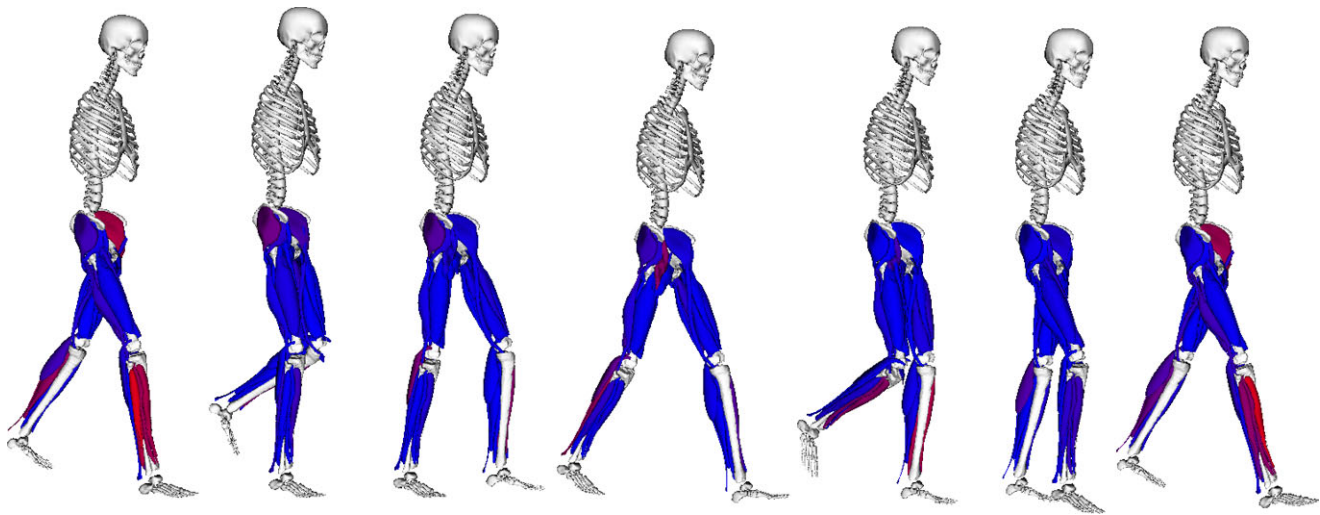

Fig. 5 The *CMC* task. Action line activations are visualized on segmented muscle meshes during a gait cycle. The evaluation of areas of activity allows the intuitive biofeedback of neuromuscular simulations

weights, with $\sum_i^n w_i = 1$. Then, our framework calculates $a = \sum_i^n w_i \times a_i$. If no weight is given, a default weight of $\frac{1}{n}$ is used for each of the n action lines. Either color or opacity can then be used to visualize the activation or force of each muscle. The color and opacity maps can also be edited by the user.

4.2.5 Electromyography validation

Electromyography (EMG) is a method for recording and evaluating the muscular activity. It detects the electrical potential generated by muscle cells when these are activated or at rest. There are two main kinds of EMG, surface EMG (sEMG) and needle EMG which measures the muscle potential inside a muscle. EMG is used in several applications, such as clinical use in diagnosis of neurological and neuromuscular problems. In neuromuscular simulations, EMG signals can be used either to drive a simulation or as validation data where the simulated muscular excitation patterns have to be correlated to the captured EMG signal. EMG is one of the few methods that provide a window on muscle activity and hence muscle force production during functional movements (see [24] for a review on recording and processing sEMG data).

Fig. 6 The *EMG* task. Muscle activation patterns are compared to actual EMG signals. From left to right (synchronized on the same crouch time): the *exact* method shows the small direct differences in *blue* and the larger in *red*; the *derivative* method uses the same principle on the first derivative values; the *sign* function shows the differences in the trends (*blue*: wrongly negative, *white*: same trend, *red*: wrongly positive); the *threshold* function shows if muscular activities are underestimated (*blue*), overestimated (*red*) or in the same range (*white*) than the EMG

In this task, EMG data are preprocessed and then compared to simulated muscle activation patterns. The framework applies the following manipulations. First, the peak value from maximum voluntary contraction trials are used in order to normalize the EMG signal of each recorded muscle. Then the signals can optionally be rectified (mean offset correction plus absolute valuation), filtered (low-pass at a user-defined frequency) and re-sampled (at a user-defined time step). Finally, the comparison with the simulated muscle activation is realized. The framework provides four frame-by-frame comparison functions.

- The *exact* function compares directly the processed value of each EMG with its corresponding action line(s).
- The *derivative* function compares the first derivative of the EMG and activation values.
- The *sign* function compares the sign of the first derivative of the EMG and activation values.
- The *threshold* function defines whether or not EMG and activation values are above a common threshold.

These functions return a value that is used to map colors on the corresponding meshes (see Fig. 6). Color maps and relationships between signals and geometries are defined as in Sect. 4.2.4.

4.2.6 Finite element simulation

As we have previously discussed, computational methods for simulating motion have primarily used muscle-driven multi-body dynamics, in which neuromuscular control is optimized. Such simulations generally represent joints and soft-tissue as simple kinematic or elastic elements for computational efficiency. These assumptions limit applications in studies such as ligament injury or osteoarthritis, where local tissue loading must be predicted. Conversely, tissue can be simulated using the finite element (FE) method, but this does not represent the effects of whole body dynamics and neuromuscular control. Coupling the two domains would overcome these limitations and allow prediction of movement strategies guided by tissue stresses. Researchers have already demonstrated benefits of such approaches on foot modeling in gait analysis [25] and on limited sets of lower limb muscles during contraction [26].

This visualization task aims at visualizing finite element simulation results directly inside the neuromuscular platform. For example, for users running static optimization or computer muscle control to estimate muscle forces and using them as input for FE simulations, this task provides a visual feedback of their simulations. In the current state of the framework, the task can manipulate only 3D descriptions of the geometries of objects over time. No other data (pressure, temperature, strain, stress) are imported. The task so displays the geometrical modifications and colors the cells according to these modifications (see Fig. 7).

4.2.7 Sensitivity study

An ideal neuromuscular model should use subject-specific kinematical and muscle parameters in order to accurately estimate postures and muscle forces. However, because subject-specific properties are difficult to measure and development of novel simulations is time-consuming, generic parameters are used in most of simulation studies even when the muscle parameters of neuromuscular disorder subjects are quite different from the generic ones. As a result, it is
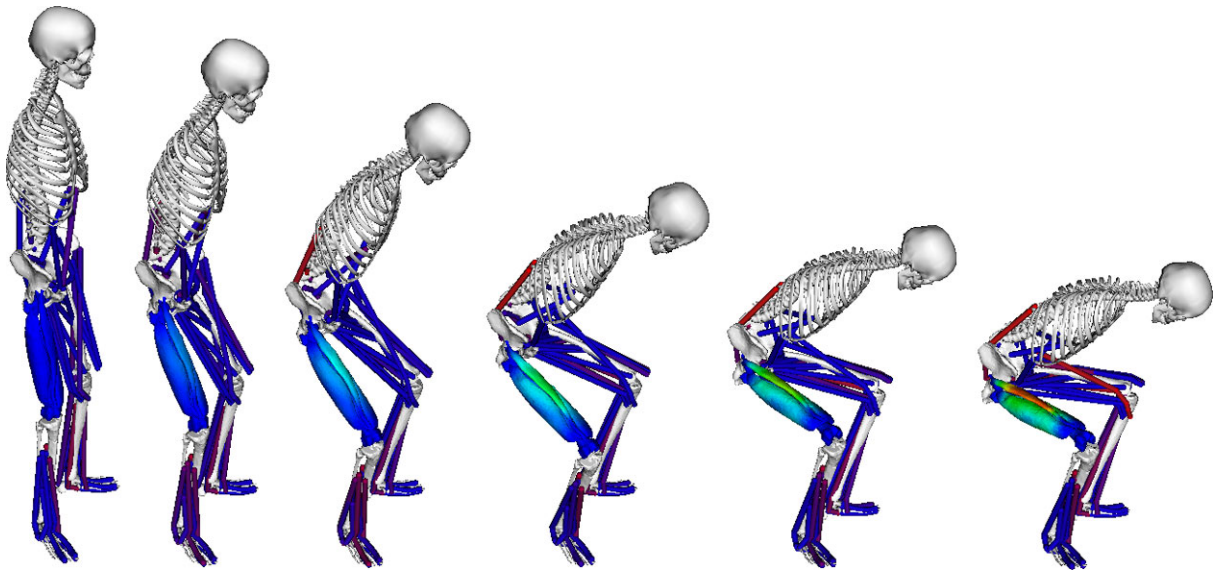
Fig. 7 The *FES* task. Multi-scale simulations are visualized simultaneously, allowing the analysis of interactions, for example between tissue, movement and internal activity. A FE simulation of the deformation of *vastus* muscles is shown concurrently with the neuromuscular simulation that has been used to set up the FE simulation
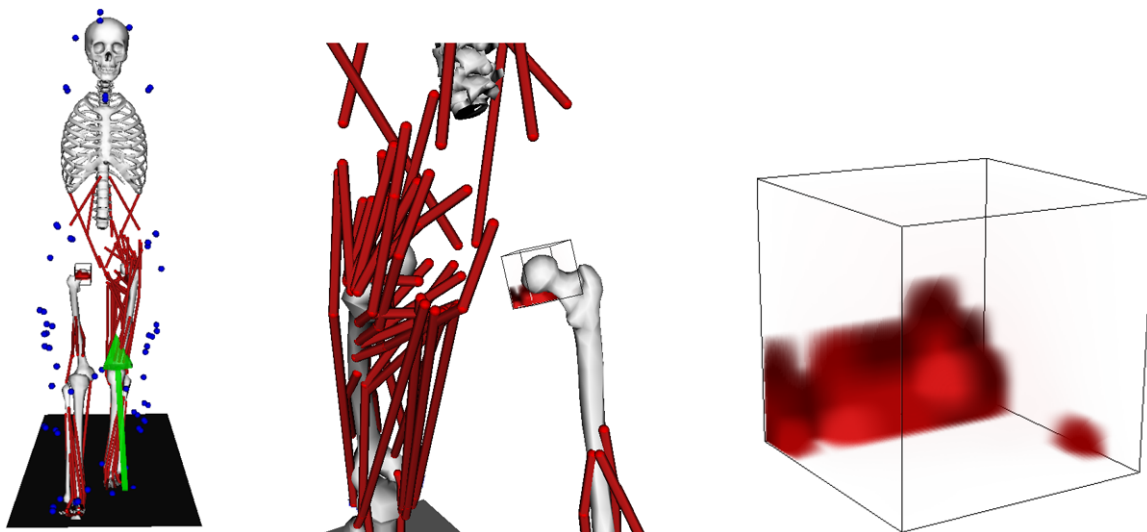

Fig. 8 The *SS* task. Volume rendering is used to visualize multi-simulation data sets especially designed for sensitivity studies. This figure presents hip flexors activities according to the right HJC location during gait. *Left*: the global view of the neuromuscular model with the 3D volume; *middle*: a close-up view on the right hip; *right*: a 3D volume view, each voxel corresponds to a simulated HJC position and the *red-opacity scale* represents the flexors forces

necessary to evaluate the influence of using generic parameters on neuromuscular models. Researchers and clinicians develop then sensitivity studies in order to introduce perturbations in the model. For example, the localization of the subject-specific hip joint center (HJC) is difficult as it is far from palpable bony landmarks. Errors in HJC localization that are as small as 20 mm may lead to substantial inaccuracies in hip kinematics and moments calculations [27] and moments arms of muscles that cross the hip [28]. A study

on muscle properties sensitivity for gait simulation suggests that it is especially important to use accurate values of tendon slack length and optimal fiber length for ankle plantar-flexors and knee extensors [29].

This task is especially designed to visualize results of sensitivity studies. Volume rendering is used to display quantitative information according to a 3-dimensional perturbation of the simulation (model or computational method). The framework first loads the set of simulations

and stores them according to their locations in the user's 3D volume. Different options are available, such as the definition of missing data, the size of the voxels or the color map. Then the framework shows the 3D volume in the model view using a ray tracing method that sums up the values of one (or more) specific parameter from the simulations (see Fig. 8).

5 Application to neuromuscular simulation

An evaluation experiment was performed on neuromuscular simulations of gait and crouch motions for several subjects (see data acquisition in Sect. 2). This section describes how the visualization framework can be used to analyze resulting data and present feedback from domain scientists.

5.1 Exploring and analyzing the data sets

As raised in Sect. 4.2, the implemented functionalities provide a large range of analysis possibilities.

Figure 3 shows for example clearly that the upper body markers are unworthily tracked while large errors also appear on feet markers at gait toe-off. These types of information are useful to adjust weights and algorithms during the IK stage. Interaction between computation and tracking estimation is so faster and more intuitive.

The concurrent use of *CMC* (Fig. 5) and *EMG* (Fig. 6) tasks demonstrates that researchers will have the possibility to point out efficiently model related and simulation related problems. For example, we can see here that *vastus lateralis* and *gastrocnemius* muscles might have problems in their activation patterns. Muscle properties and simulation parameters can then be updated in that direction.

From Fig. 8 we can observe easily that hip flexors require the production of larger forces to actuate the reference gait motion when the hip joint center is moved anteriorly and inferiorly. Such interpretation would have been much more difficult and time-consuming without the help of the visualization task.

5.2 Feedback from domain scientists

While we have yet to perform extended analysis of the use of this framework by biomechanics researchers and clinicians, we have collected some initial feedback from the OpenSim users community. The first free public version of the framework has been made available in may 2010, and has been tested by more than forty users in the following few months. The framework is available as an OpenSim plug-in at the following URL: https://simtk.org/home/visutoolopensim/.

Feedback was positive, especially for the *IK* and *CMC* tasks which were the most used and where the benefits were highlighted. *EMG* and *SS* tasks have been made available only recently. The feedback reinforces the importance of developing 3D visualization task for the investigating of neuromuscular data.

6 Conclusion

In this paper, we have presented a framework to create scientific visualizations of neuromuscular data. The visualizations are tightly integrated within the neuromuscular graphical user interface to ease interactions between data visualizations and simulations. Versatile information related to the different stages of a neuromuscular analysis can be explored and analyzed in a common framework. It has currently the ability to perform visualization of MR images, inverse kinematics errors, inverse dynamics torques, muscle activity, validation against EMG signals, FE simulations and sensitivity studies using adequate visual representations.

The visualization framework has worked very early with the concept of task (describing what users want to visualize) and performer (describing how users want to perform the visualization). Future work includes the development of additional tasks and performers according to additional biomechanics researchers and clinicians requirements. Video recording rendering and more concurrent multi-scale simulations visualizations are some examples. The comparative study of our framework usability on other simulation platforms than OpenSim will also be investigated.

The framework was also designed to allow an easy extension of its functionalities. To create a new mode of visualization, a new task and/or a new performer has to be created. Class factories are available to create them intuitively. We so hope future developers will use and extend existing tasks and performers to create new visualization functionalities targeting their needs.

References

1. Damsgaard, M., Rasmussen, J., Christensen, S.T., Surma, E., de Zee, M.: Analysis of musculoskeletal systems in the anybody modeling system. Simul. Model. Pract. Theory **14**(8), 1100–1111 (2006)
2. Delp, S.L., Loan, J.P., Hoy, M.G., Zajac, F.E., Topp, E.L., Rosen, J.M.: An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. IEEE Trans. Biomed. Eng. **37**(8), 757–767 (1990)
3. Delp, S.L., Anderson, F.C., Arnold, A.S., Loan, P., Habib, A., John, C.T., Guendelman, E., Thelen, D.G.: OpenSim: Open-source software to create and analyze dynamic simulations of movement. IEEE Trans. Biomed. Eng. **54**, 2007 (1940–1950)

4. Klein Horsman, M.D., Koopman, H.F.J.M., van der Helm, F.C.T., Poliacu Prosé, L., Veeger, H.E.J.: Morphological muscle and joint parameters for musculoskeletal modelling of the lower extremity. Clin. Biomech. **22**(2), 239–247 (2007)

5. Schmid, J., Magnenat-Thalmann, N.: MRI Bone segmentation using deformable models and shape priors. In: Metaxas, D., Axel, L., Szekely, G., Fichtinger, G. (eds.) MICCAI. LNCS, vol. 5241, pp. 119–126. Springer, Berlin (2008)

6. Sandholm, A., Pronost, N., Thalmann, D.: MotionLab: A MATLAB toolbox for extracting and processing experimental motion capture data for neuromuscular simulations. In: Proceedings of the Second 3D Physiological Human Workshop (3DPH) 2009, Lecture Notes in Computer Science, vol. 5903 (2009)

7. FEBio platform: http://mrl.sci.utah.edu/software/febio, accessed in September 2010

8. Whittle, M.: Gait Analysis: an Introduction, 4th ed. Butterworth-Heinemann, Oxford (2006)

9. Manal, K., Chang, C.-C., Hamill, J., Stanhope, S.J.: A three-dimensional data visualization technique for reporting movement pattern deviations. J. Biomech. **38**(11), 2151–2156 (2005)

10. Manal, K., Stanhope, S.J.: A novel method for displaying gait and clinical movement analysis data. Gait Posture **20**(2), 222–226 (2004)

11. Keefe, D., Ewert, M., Ribarsky, W., Chang, R.: Interactive coordinated multiple-view visualization of biomechanical motion data. IEEE Trans. Vis. Comput. Graph. **15**(6), 1383–1390 (2009)

12. Krekel, P.R., Valstar, E.R., de Groot, J., Post, F.H., Nelissen, R.G.H.H., Botha, C.P.: Visual analysis of multi-joint kinematic data. Comput. Graph. Forum **29**(3), 1123–1132 (2010)

13. Keefe, D.F., O'Brien, T.M., Baier, D.B., Gatesy, S.M., Brainerd, E.L., Laidlaw, D.H.: Exploratory visualization of animal kinematics using instantaneous helical axes. Comput. Graph. Forum **27**(8), 863–870 (2008)

14. van Sint Jan, S.L., Clapworthy, G.J., Rooze, M.: Visualization of combined motions in human joints. IEEE Comput. Graph. Appl. **18**, 10–14 (1998)

15. Marai, G.E., Laidlaw, D.H., Andrews, S., Grimm, C.M., Crisco, J.J.: Estimating joint contact areas and ligament lengths from bone kinematics and surfaces. IEEE Trans. Biomed. Eng. **51**, 790–799 (2004)

16. van den Bogert, A., Geijtenbeek, T., Even-Zohar, O.: Real-time estimation of muscle forces from inverse dynamics. In: Annual Meeting of the American Society of Biomechanics (NACOB) (2008)

17. Chen, J., Forsberg, A., Swartz, S., Laidlaw, D.H.: Interactive multiple scale small multiples. IEEE Visualization 2007 Poster Compendium (2007)

18. Robertson, G., Fernandez, R., Fisher, D., Lee, B., Stasko, J.: Effectiveness of animation in trend visualization. IEEE Trans. Vis. Comput. Graphics **14**(6), 1325–1332 (2008)

19. Holzbaur, K.R.S., Murray, W.M., Delp, S.L.: A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control. Ann. Biomed. Eng. **33**, 829–840 (2005)

20. Arnold, E., Ward, S., Lieber, R., Delp, S.L.: A model of the lower limb for analysis of human movement. Ann. Biomed. Eng. **38**, 269–279 (2010)

21. Scheys, L., Loeckx, D., Spaepen, A., Suetens, P., Jonkers, I.: Atlas-based non-rigid image registration to automatically define line-of-action muscle models: a validation study. J. Biomech. **42**(5), 565–572 (2009)

22. Arnold, A.S., Salinas, S., Hakawa, D.J., Delp, S.L.: Accuracy of muscle moment arms estimated from MRI-based musculoskeletal models of the lower extremity. Comput. Aided Surg. **5**(2), 108–119 (2000)

23. Della Croce, U., Cappozzo, A., Kerrigan, D.C.: Pelvis and lower limb anatomical landmark calibration precision and its propagation to bone geometry and joint angles. Med. Biolog. Eng. Comput. **37**(2), 155–161 (1999)

24. Staudenmann, D., Roeleveld, K., Stegeman, D.F., van Dieen, J.H.: Methodological aspects of SEMG recordings for force estimation—a tutorial and review. J. Electromyogr. Kinesiol. **20**(3), 375–387 (2010)

25. Halloran, J.P., Ackermann, M., Erdemir, A., van den Bogert, A.J.: Concurrent musculoskeletal dynamics and finite element analysis predicts altered gait patterns to reduce foot tissue loading. J. Biomech. (2010)

26. Maurice, X., Sandholm, A., Pronost, N., Boulic, R., Thalmann, D.: A subject-specific software solution for the modeling and the visualization of muscles deformations. Vis. Comput. **25**(9), 835–842 (2009)

27. Stagni, R., Leardini, A., Cappozzo, A., Benedetti, M.G., Cappello, A.: Effects of hip joint centre mislocation on gait analysis results. J. Biomech. **33**(11), 1479–1487 (2000)

28. Delp, S.L., Maloney, W.: Effects of hip center location on the moment-generating capacity of the muscles. J. Biomech. **26**(4–5), 485–499 (1993)

29. Xiao, M., Higginson, J.: Sensitivity of estimated muscle force in forward simulation of normal walking. J. Appl. Biomech. **26**(2), 142–149 (2010)

Nicolas Pronost is Assistant Professor at Utrecht University (The Netherlands) in the Games and Virtual Worlds research group. He studied Computer Science at the University of Rennes 1 (France), where he obtained his Ph.D. in 2006 and worked within the SIAMES/Bunraku project of the INRIA/IRISA Laboratory. Then he was successively a postdoctoral researcher at the State Key Lab of CAD & CG at Zheijiang University (China), and a postdoctoral researcher (Marie Curie Experienced Researcher) in the VRLab at EPFL (Switzerland). The main topics of his work are the analysis and the synthesis of human motions for biomechanical simulation and animation.

Anders Sandholm is a Ph.D. candidate and a Marie Curie Early Stage Researcher supervised by Professor Daniel Thalmann, VRLab at Ecole Polytechnique Fédérale de Lausanne, Switzerland. After his Master of Science aimed toward media technology and visualization, Anders worked as an IT consultant in Stockholm, before joining the eHealth Institute in Kalmar and The Programming Environments Laboratory, Linkopings University. During this year Anders' main research was to evaluate if and how an object-oriented language could be used in medical simulators that use haptic devices or accurate tracking devices.

Daniel Thalmann is Professor and Director of The Virtual Reality Lab (VRLab) at EPFL, Switzerland. He is a pioneer in research on virtual humans. His current research interests include real-time virtual humans in virtual reality, networked virtual environments, artificial life, and multimedia. Daniel Thalmann has been Professor at the University of Montreal and Visiting Professor/Researcher at CERN, University of Nebraska, University of Tokyo, and Institute of System Science in Singapore. He received his Ph.D. in Computer Science in 1977 from the University of Geneva and an Honorary Doctorate (Honoris Causa) from University Paul-Sabatier in Toulouse, France, in 2003.