Lars Lehmann*, Christian Wiede, and Gangolf Hirtz

Individual Avatar Feedback Creation for Assisted Motion Control

<https://doi.org/10.1515/cdbme-2019-0155>

Abstract: In Medical Training Therapy (MTT), the precise execution of the training exercises is of decisive importance for the success of the therapy. Currently, a therapist has to treat up to 15 patients simultaneously on an outpatient basis. We propose, an assistance system, can evaluate both quantity and quality of the movement performed using a target-oriented model and give recommendations for action directly to the patient by means of feedback. An avatar in traffic light colours signals in which body region an error has occurred. The individualisation of the underlying three-dimensional avatar increases the willingness of the patients to participate in the exercises without the supervision of therapist. The monitored error frequency was decreased by 50 % by the assistance system.

Keywords: Visualisation, Feedback Control, Assisted Motion Control, Motion Error Detection

1 Introduction

In MTT, up to 15 patients are supervised simultaneously for by one trained therapist [5]. Consequently, it is not guaranteed that caregivers will have a constant overview of the entire group at all times. This lack of control can have a devastating effect on the therapy, because in the unobserved moments movement errors can occur in the patient, which can have an extremely negative effect on the treatment. It is necessary to support the overstrained caregivers be a self-sufficient, markerless, assistance system that monitors the patient's movements and immediately alerts them to any movement errors.

We propose a assistance system which uses a depth sensor to extract a skeleton of the patient and to evaluate its motion sequence with the help of artificial intelligence. Furthermore, it is necessary to present the determined data to the patient in a suitable manner. The feedback has to be clear and immediately understandable and has to be realtime like the evaluation. Our approach includes a real-time individual wireframe model mesh animation and the subsequent coloring of detected motion errors.

***Corresponding author: Lars Lehmann**, Chemnitz University, Reichenhainer Str. 70, Chemnitz, Germany, e-mail: lars.lehmann@etit.tu-chemnitz.de

Christian Wiede, Chemnitz University, Chemnitz, Germany
Gangolf Hirtz, Chemnitz University, Chemnitz, Germany

2 Related Work

Due to demographic change and the insufficient number of movement therapists, attempts are being made to counter this problem with the help of intelligent detectors. It is helpful to introduce a visual system to motivate people to train [11][12][10]. Therefore, this work is dedicated to the development of such feedback, with a particular focus on the appropriate model representation for the transmission of supervised movements. Velloso et al. [11] uses markers to perform a motion analysis. The implementation of the visualization is limited to the text output in a realtime 2D image. These hints indicating actions already turned out to be useful automation for motion instructions. An extended work of this approach is introduced by Kwon and Cross [10]. Hereby, the system is extended by a visual component. A skeleton is visualized on which the user can orient himself when executing his exercises. The system guides as an expert through a given training pattern and analyzes the movement. Depending on the evaluation of the error, the skeleton is colored in the traffic light colors presenting the error. Another rehabilitation system with motion control is presented by Zhao et al. [12]. Here a 3D realtime model with a standard game development platform [3] is used for the presentation. The model embodies the test persons and helps them to get involved in the therapy. The movement becomes more motivating and thus more successful. The approaches described aim to increase the success of the therapy and support the automated recognition of movements. For our method, we combine all of these approaches and extend them at certain points. Especially the representation of the animation figure and the error coloring segmentation is improved. The significant enhancement of details and individualism in the representation of freely selectable whole-body realtime animation models is investigated and evaluated.

3 Methods

In order to realize a colored wireframe skeleton animation, more degrees of freedom are needed than a standard game development platform, like Unity [3], provides. Therefore, the complete graphic pipeline has to be used. A CPU is used for the backup of the static 3D model data and the matching calculation of the incoming skeleton data. For a sectionally col-

ored 3D model animation the GPU has to be used, otherwise it could not be realized in real time [4].

3.1 Static Animation Calculations

A rigged and skinned 3D model [7] is required for realtime animation. This model consists of a multi-bone skeleton and is connected to a vertex-based wireframe model. Each vertex is influenced by one or more bones, whose weight attraction to the vertex is stored in a suitable animation-supporting file format [2] [1]. In order to calculate a change in the model surface caused by skeletal animation, we need to process the static reference model. It must be known where the bones are located in the rigged model [6] and how the chain of the skeletal transformation is constructed. A transformation matrix for each node is calculated once over all bones. This matrix denotes the translation from the child node to its parent nodes down to the root node. Whenever the trailing and individually selectable vertex model is changed, this computation reflecting the original state is processed again as far as the underlying skeleton changes in the initial pose.

3.2 Static Segmentation Calculations for Error Coloring

The motion errors valuated by the system are visualized in red, yellow, and green. These traffic light colors indicate the urgency of the error, in which red has the highest priority. This requires a determination of the bone belonging to the defect. The bone affiliation of a vertex does not change in the course of the animation. Therefore, the following calculations only need to be performed once and are then permanently available for this 3D model. The connection of the weight to the bone is decisive for the determination of the affiliation. There is a possibility that the vertex is affected by eight bones. It should be noted that the weights are uniformly distributed over the entire bone. At the edge the values are low and then increase continuously to reach their maximum value in the middle and then drop again towards the edge. A linear distribution would be much better suited for a suitable colouring, because the error-related coloring can be applied to the bone. The intersection point H of a vertex G perpendicular angle γ to the bone with the bone straight AB line, with the difference Δ between these vectors, is calculated (see Figure 1) as follows:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{\Delta} = \left(\begin{pmatrix} x \\ y \\ z \end{pmatrix}_B - \begin{pmatrix} x \\ y \\ z \end{pmatrix}_A \right) \quad (1)$$

$$\Delta_U = \frac{(G.x - A.x) * \Delta.x + (G.y - A.y) * \Delta.y + (G.z - A.z) * \Delta.z}{\Delta.x^2 + \Delta.y^2 + \Delta.z^2} \quad (2)$$

$$\begin{pmatrix} A.x + \Delta_U * \Delta.x \\ A.y + \Delta_U * \Delta.y \\ A.z + \Delta_U * \Delta.z \end{pmatrix}_H \quad (3)$$

The affiliation results from the index of the bone attribute b_a . Subsequently the result is divided by the total length of the bone to obtain a percentage, the desired linear distribution, for subsequent staining.

Fig. 1: Detection of the green vertices belonging to the bone

However, there are also points which lie outside the bone line, because the automatic weight distribution also interpolates beyond the edge. These outliers must be eliminated to ensure the clear allocation of the vertices surfaces to the bone. By stretching a plane orthogonal to the straight line and comparing on which side the vertices to be considered are located. As shown in Figure 1 an exclusion can occur. The plane points E or F can be determined by inserting a straight line perpendicular angle α to the straight line at the starting point A of the bone section. This straight line is now rotated by a $90^\circ \beta$ with respect to the skeleton axis.

To determine the third point E of the plane, 2 points of the plane are needed. These are given by C and D . The difference Δ_{CD} between the two is multiplied by a matrix r_{CD} . This matrix is obtained from a four-dimensional identity matrix I which is rotated by 90° around the normal orthogonal to the surface, the previously determined Δ . The angle itself can accept all values except 0° and 180° , because the point to be calculated E would be on the same line CD .

$$\begin{pmatrix} a_{11} & \cdots & a_{14} \\ \vdots & \ddots & \vdots \\ a_{41} & \cdots & a_{44} \end{pmatrix}_{r_{CD}} = \text{rot} \left(I, 90^\circ, \begin{pmatrix} x \\ y \\ z \end{pmatrix}_{\Delta} \right) \quad (4)$$

The resulting crossed elbow line points CD and EF are used to determine the necessary tree reference points for a straight line equation. Of course this auxiliary plane has to be inserted also at the other end B of the bone line. In order to determine whether the vertex belongs to the bone, it is now sufficient to determine on which side of the straight line it is located. If the green vertex of interest lies between the spanned planes, it belongs to the bone, else if it does not fulfill this condition, it is an red outlier.

Fig. 2: Remeshing clear edge scheme: possible intersections of a triangle mesh with a plane line

If a clear, straight edge is also required for bone segmentation, the surface at the seams must be recalculated. For this the sorted out vertices and their neighborhood are needed to calculate their triangle intersections with the inserted plane. The possibilities are shown in Figure 2. The first two green and blue colored triangles are the new triangles to be created from the intersections. The green triangles are part of the bone line and the blue triangles are outside. Nevertheless, the blue triangle positions must also be recalculated, since they are related to the green ones and otherwise there would be an unwanted distortion of the mesh at those points. The other possible intersections between the auxiliary plane and the observation triangle shown in Figure 2 do not need to be recalculated.

In addition, each recalculated vertex of the mesh also requires a new normal which is formed from the average of the neighboring standards connected by triangles. To support the animation for the newly added vertices, the new bone weights are as well calculated. For this, the average weights of the vertices of the neighboring triangles are formed and stored as a new weight vector. The newly calculated values are inserted in the existing static lists.

3.3 Dynamic Animation Calculations

Following calculation is performed whenever the positions of the underlying skeleton nodes change. Therefore, the connections of the bones must be sufficiently known, so that the incoming data can be compared with the underlying skeleton reference and transformation changes can be adjusted immediately [9]. Since these calculations are only skeleton data, they can also be performed dynamically on the CPU without any noticeable latency in the display. Every incoming skeleton bone for which a 3x3 transformation matrix r has to be created. For this goal a 3D vector td is needed which represents the change of direction with respect to the reference and is the quotient of the difference vector s_{Δ} between the beginning and the end of the bone and the corresponding length l . The final transformation matrix r of each bone is then calculated by a vector rotation between the statically available rigged bone direction vector d_a and the newly calculated transformation direction td of the current skeletal bone.

3.4 Dynamic Hardware Accelerated Vertex Animations

The animation of the surface of the underlying wireframe model is realized in the fragment shader of the graphics card and is calculated for each vertex in each render frame [8]. Firstly, it has to be determined how many weighted bone neighborhoods w_i there are for each single vertex. For this purpose, the bone index array b_a is iterated and its number determined. The eight iterations result from the assumption that there could be 8 active bones for a vertex at the same time. Now the weight sum w_s is determined which acts as loop counter for the following calculations.

$$w_s = \sum_{i=0}^{w_i} w_a[i] \quad (5)$$

Then the same value w_s is used and the corresponding index b_i is taken from the statically available bone index field b_a . The change in bone rotation is represented in a 3D vector r_{Δ} . For this the previously calculated rotation matrix r is multiplied by the difference vector p_{Δ} . This vector is the difference between the position attribute p and the start vector of a bone from the static mesh.

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{r_{\Delta}} = \left(\begin{pmatrix} a_{11} & \cdots & a_{13} \\ \vdots & \ddots & \vdots \\ a_{31} & \cdots & a_{33} \end{pmatrix}_{r[b_i]} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix}_{p_{\Delta}} \right) \quad (6)$$

In order to complete the determination of the vertex change p_n at the surface, the determination of the weight factor w is decisive. For this, the neighborhood weights w_i is iterated and within the loop the quotient of the weight index w_a and the sum of weights w_s is calculated. The new and final position of the shifted vector p_n is the sum of the addition of the static skeleton nodes $t_{ja}[b_i]$, the previously calculated modified rotation vector r_Δ and the subsequent multiplication with the weight factor w .

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{p_n} = \sum_{i=0}^{w_i} \left(\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{t_{ja}[b_i]} + \begin{pmatrix} x \\ y \\ z \end{pmatrix}_{r_\Delta} \right) w \quad (7)$$

4 Results and Discussion

An user acceptance study was conducted to evaluate how the visual feedback system encouraged patients to perform their movements properly and how the design was received. The system was tested in a clinical facility for patients with a hip implantat. 15 test persons completed a three-week rehabilitation program and trained with the assistance system under observation by therapists. Subsequently, the patients were asked to complete a standardized questionnaire comprising twelve questions with five possible answers each. The system was evaluated predominantly positively in the survey. The evaluation of the movement feedback, which was important for the question of the paper, and the indications for autocorrection were mostly understandable and helpful [5].

In addition, the error frequencies over the period of therapy were evaluated. The aim was to determine whether the patient's movements could be improved during the training period with the help of the assistance system. This also shows that the error frequency monitored by the assistance system was decreasing by more than 50 % [5].

5 Conclusion and Future Work

A visual feedback was created for the movement therapy, in which it is possible to select an individual avatar for the display. The movements were analyzed and evaluated. Errors were colored according to their importance in traffic light colors on the previously calculated bone segments. This allowed the test person to see immediately where the error occurred. In order to evaluate the assisted therapy system, a user study was carried out with subjects who evaluated the system. The

results were positive and the user acceptance high. The independent evaluation of the recorded training results showed a noticeable gain in health-promoting movements.

In future the integration of real scanned individual characters would promote immersion. Of course, the creation of the models would have to be feasible as a sideline and without too much time expenditure.

The monitoring of the control should also extend to other body regions. Both the hands and the face could be integrated into the motion monitoring. In addition, the number of reference movements would have to be increased by implementing a system of suitable recording of these. Thereby, the therapists could store a new motion sequence at any time.

The integration of the complete system into a virtual reality environment has once again increased immersion. The limitation that is associated with a fixed location feedback system would be abolished. It would then no longer be necessary to realign the display device for body rotated exercise changes.

References

- [1] Arnaud R, Barnes MC. COLLADA Book.; 2006.
- [2] Murray JD, VanRyper W. Encyclopedia of graphics file formats; 1996.
- [3] Smitt M, "Unity 5.x Cookbook: More than 100 solutions to build amazing 2D and 3D games with Unity"; 2015
- [4] Wolff D, "OpenGL 4.0 Shading Language Cookbook"; 2011
- [5] Assistierte Bewegungskontrolle in der Rehabilitation durch intelligente Sensortechnologie.; 2018.
- [6] Le BH. Example-based Rigging and Realtime Animation of Characters with Linear Skinning Models. University of Houston: 2014.
- [7] Dalmau DS. Core Techniques and Algorithms in Game Programming. Chapter 15. Character Animation; 2004:447-83.
- [8] Movania MM. OpenGL Development Cookbook. Chapter 8. Skeletal and Physically-based Simulation on the GPU; 2013.
- [9] Mukundan R. Advanced Methods in Computer Graphics: With examples in OpenGL. Chapter 4. Skeletal Animation; 2012:53-76.
- [10] E. Velloso, A. Bulling, and H. Gellersen, "Motionma: motion modelling and analysis by demonstration," in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2013: 1309–1318.
- [11] D.Y. Kwon and M. Cross, "Combining Body Sensors and VisualSensors for Motion Training," Proc. ACM SIGCHI, pp. 94-101, 2005.
- [12] W. Zhao, H. Feng, R. Lun, D. D. Espy, and M. Reinthal, "A kinect-based rehabilitation exercise monitoring and guidance systems," in Proceedings of the 5th IEEE International Conference on Software Engineering and Service Science. IEEE, 2014: 762–765.
- [13] Lösche C, Nitzsche N, Maiwald C, Richter J, Lehmann L, Wiede C, Weigert M, Schulz H. Validation of an assistance system for motion analysis. Dtsch Z Sportmed. 2019.