

AMOBAs: A Database System for Annotating Captured Human Movements

Stefan Grünvogel, Jens Piesk,
Stephan Schwichtenberg
Laboratory for Mixed Realities
at the Academy of Media Arts Cologne
Am Coloneum 1, Gebäude C1
50829 Köln, Germany
{gruenvogel,piesk,schwichtenberg}
@lmr.khm.de

Gregor Büchel
University of Applied Science Cologne
Betzdorfer Str. 2, 50678 Köln, Germany
gregor.buechel@fh-koeln.de

Abstract

For the management of a large number of motion data for humanoid virtual actors we propose to use a database system to store and retrieve motion data with additional information. We show key questions lying behind the different classification schemes for human motion. Having obtained these criteria, we deduce the requirements a database has to fulfil for coping different classification schemes. We describe the AMOBA database system which allows to store and retrieve human movement data with an expandable set of classification schemes. To implement classification schemes we annotate movements with specific characteristics.

1. Introduction

Using databases in the field of computer animation is a promising idea which was explored in different conceptual settings. Farenc et. al. [7] presented a virtual humanoid animation in a web-based application querying a database in order to get more information about the environment. Steinback [19] uses a database for human sign language and Tanco [20] for synthesising human motion with the help of statistical models.

Since the creation of good animation data is a highly time and manpower consuming process, there are several approaches for reusing motion data of a certain character by other characters while keeping the characteristic of the motion. Gleicher [8] for example uses spacetime constraints displacement mappings for retargetting and Hodgins [10] uses dynamic simulation.

Keeping these techniques in mind the idea of using a database for storing animation sequences and remapping

the data onto different characters is nearby. Akanksha et. al. [1] propose a database for storing and retrieving motion data. They use a set of metadata for describing the kinematic properties and some simple style information (e.g. 'brisk') of the motions. Thus they are able to reuse the animations of one geometric model for another geometric model.

In order to search for some special kind of motion in a set of animation sequences the set of simple style informations has to be expanded. Our approach is to describe human movements by an expandable set of different classification schemes. The animation sequences in the database are annotated with respect to these classification schemes. Motions can then be retrieved from the database by describing the motions by one or several classification schemes. For example we can search for all motions having a 'happy' style and being an iconic gesture.

The structure of the paper is as follows. In Chapter 2 we introduce the questions we encounter if we want to consider different classification schemes for human movement. We consider criterions in terms of three questions for specifying classification schemes for human movement. These questions result in requirements on a motion database which we state and discuss in Chapter 3. We implemented a database prototype, the AMOBA database (Annotated Motion Database), which respects these requirements. A description of the system architecture, the Entity-Relationship Model and the Annotations of the AMOBA database is given in Chapter 4.

2. Classification Schemes

There are several different systems for the characterization and classification of human movements. Behind these schemes lie the following questions:

2.1. What is Classified?

A given motion can be described with the help of the data defining the movement of the skeleton (which we get e. g. by motion capturing), but also by additional knowledge that comes from some other data source. By watching an animated body a Laban Movement Analyst can get the appropriate Shape and Effort parameters of the movement (cf. Laban [13], [14], Liwei Zhao [21]). Psychologists can evaluate in experiments the effect of nonverbal behaviour of a virtual character to persons watching the character acting (cf. Badler and Allbeck [2], Bente et. al. [3]). Furthermore if motion captured data and an appropriate skeleton is provided one can calculate the speed of the limbs. But there is also other data which might be important and which is not available by analysing the kinematics of the skeleton. Electromyograms of the lower extremities or the centre of pressure for each foot of a walking human might serve as an example (cf. Elbe et. al. [6], Pearson [17]). Enumeration of the total of movements belonging to a special class is often used in motion capturing sessions for computer games (cf. Keines [12], Menache [15]). Thus there may be the classes *locomotion* and *fighting* for the protagonist of a third person shooter game. Each class itself has subclasses of movement. Locomotion for example has the subclasses *run*, *walk*, *crawl* and for fighting it is *boxing*, *kicking*, *shooting*. These subclasses itself hold several distinct motions, which can be characterized further may be identified by their name.

2.2. Which Process is used for Classification?

It is clear that each categorization scheme uses its specific process for acquiring the respective data. These can be divided into three categories: acquiring the data automatically from the recorded movement of the skeleton, acquiring additional data from separate data sources or classification by an expert. Examples for automatic data acquisition can be found in human gesture recognition. The methods range from template matching (e.g. Zimmerman et. al. [22]) over statistical classification (e.g. Rubine [18]) up to neural networks (e.g. Murakami and Taguchi [16]). Electromyograms are used by Elbe et. al. [6] and Pearson [17]. An example for the acquisition of data from an expert would be a Laban Movement Analyst, who describes Laban's Shape and Effort of a given motion (cf. [13], [14], [21]). But there are also first experiments for retrieving such data automatically. Camurri [4] analysed the expressivity in dance in Laban's Term of Effort by measuring the changes of the Kinesphere, i. e. the expansion and contraction of the space surrounding the dancer. The chosen method of data acquisition also has an impact on the type of the data for the characterization: words, sin-

gle values, ranges between 0 and 1, trajectories etc.

2.3. Which Movements are Classified?

Not every characterization scheme can classify all human movements. In the presentation [11] for example there is a "review of the tasks an infantry man must do" (an enumeration of several movements and stances) - which are certainly not all possible movements a human being is able to do. Other classification schemes are applicable to every human movement, for example the Laban Movement Analysis.

3. Requirements of a Motion Database

Taking into account the above properties of classification schemes, a motion database should fulfill the following requirements when describing recorded human motion.

1. The database should be able to handle several classification schemes.
2. The set of available classification schemes should be extensible.
3. The database should provide a sufficient vocabulary for each classification scheme.
4. The database should handle incomplete characterizations.

We will now discuss the above requirements.

Requirement 1 means that it is possible to describe a human motion in different classification schemes in the motion database. An optional feature would be to use similarities between classifications schemes for retrieving animation sequences. If two classification schemes describe the same phenomena by two different measures, it should be possible to map the values of one scheme to the other. An example for this would be to map the description of Laban's Effort Elements into kinematic description of the limbs. This was already explored by Chi [5] for the synthesis of expressive arm movements.

The second requirement means, that the database should not be restricted to a given set of classification schemes, but also maintains the possibility to introduce new ones while the database is in use.

The provision of a sufficient vocabulary constrains the user to classify a new movement according to some given basic categories, and frees him to expand the classification at a lower level in the classification hierarchy. Example: Think of a classification of human locomotion by making a list of all possible locomotions. A simple idea would be to classify the movements by the names of the motions files.

If we have walking motions then two different users could name the motions different for example "walk", "walk forward", "forward" etc. which makes it difficult to find again the motion in the database later, because there may be some hundred more walk motions in the database. A better idea would be to choose within the classification scheme "locomotion" the subclasses "run", "walk", "crawl". Now the different "walk" movements can be put into the "walk" subclass and the user is forced to choose this given categorization scheme. Later a further classification of a given motion is open to the user. He can now specify a motion which represents a human walking straight forward with the name "forward".

It should be possible to put a motion into the database without characterize the motion in all available characterization schemes of the database, as indicated by Requirement 4. This is due to the fact, that often no appropriate expert is available who can classify the movement in a specific classification scheme.

4. The Motion Database AMOBA

4.1. The System Environment

For the implementation of the database scheme of AMOBA we have used the relational database *MySQL*, running under Linux. The animations for a special character are synthesized by our real-time Animation-Engine. The Animation-Engine uses the concept of Motion Model (cf. Grassia [9]) for controlling movements with high level parameters. The Animation Engine has an interface to the AMOBA database for retrieving animation sequences which are used for synthesizing complex movements by blending them together. It also retrieves additional annotation data (such as length of clip, which bodyparts are used etc.) of the animation sequences which are used for synthesizing the movements. The connection to the database is realized with the ODBC interface of *MySQL*. There also exists a second interface for *DEPAC*, a proprietary experimental platform for evaluation of animated characters (cf. Bente et. al. [3]) which is used for editing annotations of motions.

4.2. The Entity-Relationship Diagram

The AMOBA database has three principal components: *Character*, *Motion Data* and *Annotation*. Their connection is shown in Figure 1 as a simplified Entity-Relationship diagram.

Since we want to store motion data for different virtual humans the entity *Character* defines a hierarchy of *Joints* which together form the skeleton of the *Character*. Note that we only consider articulated figures. For each joint we

add also the information to which *Body Part Group* it belongs. Examples for a *Body Part Group* could be the left arm, the right arm, the left foot etc. This information is essential if one later wants to emphasize, that a movement is strongly coupled with certain body parts. The arms of a human being are for example more important for a pointing gesture than the feet.

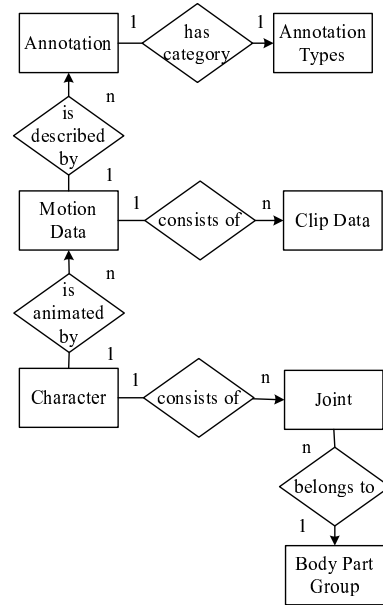


Figure 1. Simplified Entity-Relationship Diagram of the AMOBA Database

The *Frame* table keeps motion data for a certain *Character*. Motion data consists of translation and rotation values for the joints of the skeleton of a character. We preferred to store the rotation data as quaternions to avoid introducing the limitations associated to Euler angles sequences when the original data is free from such problems (e.g. when motion capture data or animation is provided in 3x3 matrix or quaternion formats). Up to now we only store animation data which was created by motion capturing. The *Motion Data* table connects the motion data in *Frame* with the classification of the motion.

For storing an expandable set of classifications of a motion we introduce the *Annotation* entity (cf. Table 1). Beside the primary key it holds the key for another annotation which can describe a given motion even further. Thus the Requirements 1 and 2 of Section 3 are fulfilled. For holding actual annotation data we have the *description* field for textual values and the *value* field for holding numeric values. In order to represent complex datatypes (like time sequence, quaternions or bodypart) we have to sequentialize the complex data types to these atomic *description* and *value* fields.

Field Name	Description
annotation_key	Index for Primary Key
next	Index for Secondary Key
annotation_identifier	Annotation Type as Foreign Key
description	Free Text for Annotation Type
value	Value of the Annotation Type

Table 1. The Annotation Entity Table

Field Name	Description
annotation_identifier	Index of Primary Key
category	Identifier for the Category
subcategory	Identifier for the Subcategory

Table 2. The Annotation Type Entity Table

Thus if we want to add a new datatype we do not have to add a new SQL-table specialized for this new data type into the database and can hold all data types in one table.

Because we also want to ensure a faithful vocabulary (Requirement 3) each annotation belongs to a *category* and a *subcategory* (cf. Table 2). These have to be set if a new categorization scheme is added to the database system. We decided not to introduce subcategories of subcategories (and so forth) to prevent flooding the database with numerous descriptions. Thus the interface of the database forces the user to think of a sufficient choice of subcategories. But the user has the freedom for further diversification under the level of subcategories.

At the moment we have the following database entries for Annotation Types: Locomotion (with subcategories running, walking etc.), Action (with subcategories take, grasp etc.), Emotional (with subcategories fear, interest etc.), Gestures (with subcategories iconic, emblematic etc.) and Effect (the effect of the animation on the viewer).

A special kind of *Annotation Type* is *Technical*, which supplies the database with different data types such as integer, double, string, spacevector, quaternion, bodypart, time-sequence etc. As mentioned above, these complex data types are realized as subcategories of the *Technical* annotation and sequentialized into the atomic fields *description* and *value* of the *Annotation*

Further research is needed to find and elaborate the right description and annotation schemes for specific interaction and production scenarios.

4.3. The Annotation Interface

The *Annotation* entity is part of the AMOBA data base system. Both the applications Animation Engine and DE-PAC are getting access to the database by the Annotation Interface. This is implemented as a C++ class, which serves as abstraction to the *Annotation* entity in the database. It

communicates with AMOBA by the ODBC interface. To query the database for motions with a certain characteristic, an annotation which is created by the user is searched in the database. The search result is weighted by the Best-Match Algorithm, i.e. the motion which has an annotation with the most correspondences to the user-created annotation is chosen. An additional parameter can be used as the minimal percentage of coincidence of the user input with the result. A value of 100% corresponds to an Exact-Match-Algorithm.

5. Conclusion and Future Work

We have shown different questions which lie behind the classification schemes of human movement. Four requirements for a database of human motions are stated. We proposed the prototype data base system AMOBA for storing and retrieving human motions with different classification schemes.

Further work will include finding and elaborating the right description and annotation schemes for specific interaction and production scenarios.

AMOBA will be used e.g. for an Augmented Reality System (mqube: www.projekt-mqube.de) and Nonverbal Communication Research using Conversational Virtual Actors (EMBASSI: www.embassi.de). Here the usability of the AMOBA system will be evaluated. Further research has to be done also in the field of weighting query results according to their relevance and the mapping between different characterizations.

6. Acknowledgment

The authors thank the reviewers for their comments on the first draft of the paper.

This work was supported by the BMBF grant 01 IR A04 C (M^3 -Eine mobile Multi-User Mixed Reality Umgebung) and the BMBF grant 01 IL 904 L (EMBASSI - Elektronische multimediale Bedien- und Service-Assistance).

References

- [1] Akanksha, Z. Huang, B. Prabhakaran, and C. R. Ruiz. Reusing motions and models in animations. In J. A. Jorge, N. M. Correia, H. Jones, and M. B. Kamegai, editors, *Multimedia 2001*, pages 21–32, Wien, 2001. Springer-Verlag.
- [2] N. Badler and J. Allbeck. Towards behavioral consistency in animated agents. In N. Magnenat-Thalmann and D. Thalmann, editors, *Deformable Avatars*, pages 191–205. Kluwer Academic Publishers, 2001.
- [3] G. Bente, N. Krämer, G. Trogemann, J. Piesk, and O. Fischer. An integrated approach towards the generation and

- evaluation of nonverbal behavior in face-to-face like interface agents. In *Proceedings of the Interactive Intelligence Assistance & Mobil Computing Workshop, Rostock-Warnemünde, Germany*, November 2000.
- [4] A. Camurri and R. Trocca. Analysis of expressivity in movement and dance. In *Proceedings CIM -2000, L'Aquila, AIMI*, 2000.
- [5] D. M. Chi. *A Motion Control Scheme for Animating Expressive Arm Movements*. PhD thesis, University of Pennsylvania, 1999.
- [6] R. J. Elble, C. Moody, K. Leffler, and R. Sinha. The initiation of normal walking. *Movement Disorders*, 9(2):139–146, 1994.
- [7] N. Farenc, C. Babski, F. Garat, and D. Thalmann. Database and virtual human representation on the WEB. In *Advances in Databases and Multimedia for the New Century -A Swiss-Japanese Perspective-*, pages 61–68. World Scientific, 1999.
- [8] M. Gleicher. Retargetting motion to new characters. In *Proceedings of SIGGRAPH 98*, Computer Graphics Annual Conference Series, 1998.
- [9] F. S. Grassia. *Believable Automatically Synthesized Motion by Knowledge-Enhanced Motion Transformation*. PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, 2000.
- [10] J. Hodgins and N. Pollard. Adapting simulated behaviors for ne characters. In *Proc. ACM SIGGRAPH '97*, pages 153–162, Los Angeles, 1997.
- [11] S. C. Jacobsen and M. Olivier. Exoskeleton for human performance augmentation (EHPA). In *Presentation at DARPA Workshop*, Washington, March 2000.
- [12] M. Keines. Planning and directing motion capture for games. http://www.gamasutra.com/features/20000119/kines_01.htm.
- [13] R. v. Laban. *The Mastery of Movement on Stage*. Macdonald & Evans, London, 1971.
- [14] R. v. Laban and F. Lawrence. *Effort*. Macdonald & Evans, Plymouth, second edition, 1974.
- [15] A. Menache. *Understanding Motion Capture for Computer Animation and Video Games*. Academic Press, San Diego, 2000.
- [16] K. Murakami and H. Taguchi. Gesture recognition using recurrent neural networks. In *Proceedings of the CHI'91*, pages 237–242. ACM Press, 1991.
- [17] K. Pearson. The control of human walking. *Scientific American*, 235(6):72–86, 1976.
- [18] D. Rubine. Specifying gestures by examples. *Computer Graphics*, 25(4):329–337, 1991.
- [19] C. Steinback. Capri: A database for computer generated fingerspelling. In *CSUN'S Sixteenth Annual International Conference "Technology and Persons with Disabilities"*, March 2001.
- [20] L. M. Tanco and A. Hilton. Realistic synthesis of novel human movement from a database of motion capture examples. In *Proc. IEEE Workshop on Human Motion*, 2000.
- [21] L. Zhao. *Synthesis and Aquisition of Laban Movement Analysis Qualitative Parameters for Communicative Gestures*. PhD thesis, Computer and Information Department, University of Pennsylvania, 2001.
- [22] T. G. Zimmerman, J. Lanier, C. Blanchard, S. Bryson, and Y. Harvill. A hand gesture interface device. In *Graphics Interface '87 (CHI+GI '87)*, pages 189–192, 1987.