Retrieval, Recognition and Reconstruction of Quadruped Motions

Björn Krüger[1], Hashim Yasin[1], Rebeka Zsoldos[2] and Andreas Weber[1]
[1]*Multimedia, Simulation and Virtual Reality Group, Department for Computer Science, Bonn University, Bonn, Germany*
[2]*Movement Science Group Vienna, Clinical Department of Small Animals and Horses, University of Veterinary Medicine Vienna, Vienna, Austria*

Keywords: Motion Retrieval, Motion Reconstruction, Action Recognition.

Abstract: Different techniques have been developed for capturing and retrieval, action recognition and video based reconstruction of human motion data in the past years. In this paper, we focus on how these techniques can be adapted to handle quadruped motion capture data and which new applications may appear. We discuss some particularities that must be considered during large animal motion capture. For retrieval, we derive suitable feature sets from quadrupeds motion capture data to perform fast searches for similar motions. Based on the retrieval techniques, the action recognition can be performed on the input motion capture sequences as well as on input video streams. We further present a data-driven approach to reconstruct quadruped motions from video data.

1 INTRODUCTION

Motion capturing of human motions has become a standard technique in data-driven computer animation. Several systems are nowadays available in all price categories, starting from consumer electronics (e.g. Kinect, WiiMote) up to professional optical systems like Vicon or Giant. All these technologies have their strengths and weaknesses, an overview is given in (Moeslund et al., 2006).

This increasing amount of motion capture data allows for new applications not only in computer animation and human computer interaction but also in sport sciences, medicine and biomechanics. On the one hand, motion analysis in horses became a powerful tool to record movement patterns during gait and other exercises. 3-D motion capture data of horses are mainly used in research to broaden knowledge and understanding of clinical conditions and treatment (Hobbs et al., 2010). On the other hand, quadruped data can be interesting for games, if we consider animation of non-humanoid characters (Vögele et al., 2012).

While motion capturing of animals have already got increasing attention in clinical environments but most of the techniques in computer animation are developed to handle human data. To cover this gap, we adapt several well known techniques from computer animation to work with quadrupedal motion capture data, and report on according series of experiments in this work.

2 RELATED WORK

We cover the closest related works for various areas in this section. For further details we refer to the cited publications and the references within. A good overview on previous works in computer animation, dealing with quadruped motions is given in a STAReport (Skrba et al., 2008).

Retrieval. The increasing amount of available motion capture data requires all data-driven methods to make use of efficient motion retrieval strategies. So called *Match Webs* to index motion capture databases are introduced in (Kovar and Gleicher, 2004). This method has quadratic complexity in the size of the motion capture database, since a local distance matrix has to be computed comparing each pair of frames. The same complexity holds for the computation of a *neighbor graph* structure (Chai and Hodgins, 2005). Boolean features are introduced (Müller et al., 2005), to segment human motion capture data. Krüger et al. present a fast method to search for numerically similar poses and extends pose matching to motion matching by employing a so called *lazy neighborhood graph* (Krüger et al., 2010).

Krüger B., Yasin H., Zsoldos R. and Weber A..
Retrieval, Recognition and Reconstruction of Quadruped Motions.
DOI: 10.5220/0004710903560363
In *Proceedings of the 9th International Conference on Computer Graphics Theory and Applications* (GRAPP-2014), pages 356-363
ISBN: 978-989-758-002-4

Action Recognition. In the field of action recognition, a wide variety of techniques was developed, depending on different available types of input signals.

Using video sequences as input, Bobick et al. employ *temporal templates* based on static vector-images (Bobick et al., 2001). Here, the vector value at each point is a function of the motion properties at the corresponding spatial location in an image sequence. Schuldt et al. use local space-time features in combination with *Support Vector Machine* (SVM) classifier for action recognition (Schuldt et al., 2004). Arikan et al. use an interactively guided SVM to annotate entire motion capture database in an interactive process. Their approach works well on a small motion capture database of American football motions used in their paper (Arikan et al., 2003).

Motion Reconstruction from Video. A variety of solutions has been investigated by computer vision community for 3D motion reconstruction like construction of statistical human pose models — transforming 2D silhouettes and contours into 3D pose and motion (Elgammal and su Lee, 2004); modeling of motion priors — some prior knowledge from motion captured databases utilized for 3D reconstruction; and physics based modeling for video based human motions (Wei and Chai, 2010). In motion priors modeling, the prior knowledge from mocap database is sometimes embedded into implementation of some constraints. Rosenhahn et al. employ geometric prior information about the movement pattern in markerless pose tracking process (Rosenhahn et al., 2008). Most of the work regarding reconstruction from video sequences has been done on human motion like (Yasin et al., 2013).

Less work has been done in case of reconstruction of quadruped species motion: Huang et al. synthesize motion sequences, driven by photographs of horses (Huang et al., 2013). In (Wilhelms and Van Gelder, 2003), the authors make use of contour detection techniques and fit a 3D model into the extracted 2D contours. For slow motions and simple backgrounds, this technique works satisfyingly. Considerable user interaction was needed for sequences that contain more complex motions. Based on PCA on binary input images, the authors of (Favreau et al., 2004) extract parameters to extract 3D motion sequences. We are not aware of any data-driven method in this context, since there is a vital need to record systematic quadruped motion capture databases as well.

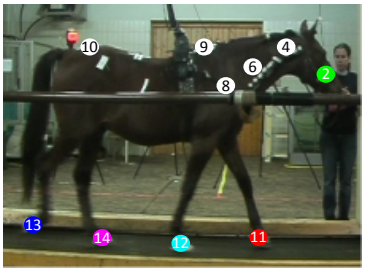

Figure 1: Marker setup on horse performing different gait sequences on treadmills. The colored markers show that markers which are selected to develop feature sets.

3 QUADRUPED MOCAP DATA

In this section, we present some details on the recording environment of our motion capture data. In this paper, we use three-dimensional kinematic data captured from five mature horses.

3.1 Marker Setup

For recording, retro flective skin markers are attached to each horse using adhesive tape. Marker setups can be varied meeting the measurements for various purposes. In a basic motion capture set-up of horses, generally seven markers are required to capture the whole body motion. The first marker is normally placed on the head, then two on the trunk and the four on the hooves. However, the number of markers needs to be increased, when the measurement purpose is more complex and requires more details. In addition, marker setups can vary between subjects due to the size differences. In our case, markers are placed on the head (left and right crista facialis), on the highest point of the withers, sacrum and lateral side of each hoof to identify motion cycles. Since mocap data used in this work are originally recorded in a clinical setup where research is focused on the neck movement in different types of gait, additional markers are attached along the vertebrae of the horses neck. An overview on used markers in this work is given in Fig. 1.

3.2 Motion Capture Dataset

Under the recording conditions described above, motion sequences of five animals are recorded. Each animal performed at least three trials (each 10 s) of the two motion styles *walk* and *trot*.

Thus, we have a database containing 30 motion trials, that include varying numbers of motion cycles. The total amount of data sums up to 36,000 frames, sampled at 120 Hz which corresponds to five minutes of motion capture data. We denote this full database

as $\mathcal{DB}_{\text{quad}}$. For our experiments, we work with various downsampled versions of this data set. If this is the case, the upper index denotes the sampling frequency.

It is important to note that due to the relatively sparse marker setup, 14 in the discussed setup, compared to 42 in the HDM05 (Müller et al., 2007) database, it is not possible to fit a suitable skeleton to the recorded marker data. Thus, we are featuring the data by their 3D marker positions only.

4 MOTION RETRIEVAL

The search for similar motion segments in a possibly annotated, database is a crucial step in all data-driven methods. We have decided to adapt the technique from Krüger et al. (Krüger et al., 2010) due to the following reasons: The technique can be easily parameterized with arbitrary feature sets. Comparisons of pose based versus motion based similarity searches are possible with the same framework.

4.1 3D Feature Sets

In (Krüger et al., 2010), the authors conclude that a feature set called $\mathcal{F}_{\text{e15}}$ is the one of choice, because of its simple computation, low dimension, while still meaningful in describing human poses. This feature set describes the positions of hands, feet and the head in the body's root coordinate system. We adapt this feature set to a feature set $\mathcal{F}_{\text{M15}}^{\text{quad}}$ where the positions of the four feet and the head are used to describe a pose in a relative coordinate system.

While in skeleton representations of human motion data, the root node is located between the hips, other choices are possible for quadruped data. We consider the marker *Withers* (No. 9 in our marker set) to be root marker. This choice is motivated by the observation, that the root node in human representations is very close to the whole body's center of mass. For quadrupeds, the center of mass is more close to the forelegs (Nauwelaerts et al., 2009). Thus, we obtain a more characteristic normalized representation of the poses.

We perform the pose normalization directly on marker data: All positions are given relative to the *Withers* marker, after rotating all marker positions around the y axis such that the *Sacrum* marker (No. 10) is moving in the x-y-plane.

We also consider feature sets including the velocities and accelerations of the markers. The idea is to have a similarity search that might be based on inputs from other sensor types, such as acceleration sensors. These sensors have been used to reconstruct full body poses of human motions (Tautges et al., 2011) from a sparse sensor setup. The feature sets developed on the basis of three dimensional information are described in Table 1.

4.2 kNN Search

We distinguish between two types of $k$nn search: First, similar poses have to be found in a motion capture database. After computing feature sets for all frames of the motion database, the k nearest neighbors for a new pose can efficiently be retrieved by searching a kd-tree. Second, we search for the k most similar motion sequences compared to an example motion sequence. In this case, a technique called *lazy neighborhood graph* (LNG) can be applied (Krüger et al., 2010).

4.3 2D Feature Sets

To reconstruct motion sequences from 2D input either derived from mocap data or video data, we need to search the database for similar poses based on 2D feature sets extracted from input signals. Hence, we introduce 2D feature sets which are derived from mocap data and from video data as well. If these feature sets are comparable, we are able to solve the cross-modal retrieval scenario between 2D input signals and the motion capture database. In this context, we sample feature sets from the database at as many as needed viewing directions to find similar poses without having any information about the actual viewing direction of the camera.

Motion Capture Data. From 3D feature sets $\mathcal{F}_{\text{M15}}^{\text{quad}}$, we have extracted 2D feature sets $\mathcal{F}_{\text{2D10}}^{\text{quad}}$ by orthographic projection on 2D plane at different viewing directions — azimuth angles (0-350) with step size 10 degrees and elevation angles (0-90) with step size 10 degrees. We translate these 2D feature sets so that they locate their origin at center of mass in order to be comparable with later described video based 2D feature sets. On the basis of these 2D feature sets, we are able to search for pose based nearest neighbors within a kd-tree. We extract 2D feature sets from input motion at specified azimuth and elevation angles in a similar fashion as described above. When experiments are performed on the bases of mocap data instead of video data, we call this data as *synthetic* input data.

Video Data. In case of video data, 2D feature sets $\mathcal{F}_{\text{vid10}}^{\text{quad}}$ are detected and tracked with the help of MSER

Table 1: Details of different types of feature sets used in this paper.

Feature sets	Type	Description of the feature sets
$\mathcal{F}^{\text{quad}}_{\text{M15}}$	3D	Normalized positions of the hooves markers and the head markers.
$\mathcal{F}^{\text{quad}}_{\text{M15vel}}$	3D	Derived velocities of the hooves markers and the head markers.
$\mathcal{F}^{\text{quad}}_{\text{M15acc}}$	3D	Derived accelerations of the hooves markers and the head markers.
$\mathcal{F}^{\text{quad}}_{\text{2D10}}$	2D	Normalized positions of the hooves and the head markers in 2D plane extracted from 3D information of synthetic data.
$\mathcal{F}^{\text{quad}}_{\text{vid10}}$	2D	Normalized positions of the hooves and the head markers in 2D image plane extracted from video data.

and SURF feature detection techniques. We have used the same feature detection technique as in (Yasin et al., 2013) and refer to this work for further details. In this paper, we only deal with intrinsic camera parameters and have discarded the extrinsic camera parameters, i.e. translation and orientation information. In order to get intrinsic camera parameters, scaling factor and focal lengths along x-axis and y-axis f_x and f_y respectively in pixel related units have been extracted from the 2D and 3D information of a couple of frames where motion capture and video data are captured synchronously. We use 3D information of few frames for camera calibration purpose only. To be comparable with 2D feature sets $\mathcal{F}^{\text{quad}}_{\text{2D10}}$ derived from mocap data, we normalize the video 2D feature sets $\mathcal{F}^{\text{quad}}_{\text{vid10}}$ by translating them to their center of mass. The two dimensional feature sets extracted either from motion capture data or video data have been represented in Table 1.

5 ACTION RECOGNITION

Identifying specific actions in an unknown stream of incoming motion data is still a current strand of research, even for human motion sequences. Basically all techniques developed for human data might be tried on quadruped data, too. Since only a limited amount of motion capture data is available, that can be used for training and cross validation steps, many sophisticated machine learning techniques can not be applied here.

We propose to use a modified k-nearest neighbor voting, that considers the temporal evolution of the regarded motion sequence. Instead of using the nearest neighbors obtained by a similarity search for voting directly, we consider poses for voting only, that are ending poses of a path through a lazy neighborhood graph (LNG) as described in Section 4. This lazy neighborhood graph can be parameterized with the width of the preceding window of frames. Thus, poses are regarded as similar, if the preceding window of frames is similar to the preceding query frames, too.

Considering the temporal evolution of a motion segment, makes the $k$nn voting more robust, due to the following reasons: First, poses that are numerically similar, but intersecting the actual sequence of poses from another direction will be filtered out. Second, if a query from a motion class is not reflected by the database and is given as input, k nearest neighbors can be returned from all motion classes. In such noisy neighborhoods it is unlikely that a connected path of the needed length is found. Thus, no neighbors will be returned and the risk of wrong classifications decreases rapidly.

6 RECONSTRUCTION FROM VIDEO

3D motion reconstruction from monocular video is an ill-posed problem and for a 3D reconstruction we obtain the missing information from pre-existing knowledge available in the mocap database. To this end, we retrieve nearest neighbors from the database, as described in the motion retrieval section. In the spirit of Chai and Hodgins, we perform a data-driven energy minimization, based on these nearest neighbors (Chai and Hodgins, 2005). We adapt their problem formulation and modify it according to the situation when we have neither skeletal data nor joints' orientation information and only make use of a single camera instead of two cameras. As we lack information in our mocap database like joints' orientation data and skeleton representation, we perform pose reconstruction by employing joints' positional configuration.

6.1 Online Motion Reconstruction

We consider reconstruction of motion as energy minimization problem and to solve this we have utilized three energy expressions, pose energy, control energy and smoothness energy,

$$E_{\text{rec}} = argmin(\alpha E_p + \beta E_c + \gamma E_s) \qquad (1)$$

Where α, β and γ are the associated weights for pose energy, control energy and smoothness energy respectively and are considered as user defined constants. Each energy term is normalized by its number of elements N.

Pose Energy. This term minimizes the 3D positional configuration of reconstructed pose and 3D

pose derived from the k-nearest neighbors in low dimensional PCA space. It eliminates back and forward movements of the performing horse and forces the reconstructed pose according to the the nearest neighbors—prior information available in mocap database. Mathematically,

$$E_p = \| \frac{1}{\sqrt{N_t}} (P_t^r - M_t)^T C^{-1} (P_t^r - M_t) \|^2 \qquad (2)$$

Where P_t^r is the reconstructed pose, M_t is the mean vector of $k$nn-examples at frame t, The C^{-1} is the inverse of the covariance matrix and $(P_t^r - M_t)^T$ is the transpose of the difference between reconstructed pose and the mean vector.

Control Energy. It computes deviation between 2D projection of 3D feature sets of hooves and head of the *reconstructed pose* $P_t^{r,2D}$ and 2D feature sets of the *estimated pose* $P_t^{e,2D}$ obtained from video or synthetic input example at current frame t,

$$E_c = [\frac{1}{\sqrt{N_t}} (P_t^{r,2D} - P_t^{e,2D})] \qquad (3)$$

In case of video example, we perform first the process of normalization so that both sides coordinates becomes comparable but in case of synthetic examples we need not to normalize the pose first.

Smoothness Energy. It enforces the smoothness of the reconstructed pose and eliminates the jerkiness or jittering effects. For that purpose, we utilize previous two reconstructed frames information as,

$$E_s = [\frac{1}{\sqrt{N_t}} (P_t^r - 2P_{t-1}^r + P_{t-2}^r)] \qquad (4)$$

Where P_t^r, P_{t-1}^r and P_{t-2}^r are the reconstructed poses at frames t, $t-1$ and $t-2$ respectively.

7 RESULTS

7.1 Similarity Searches

We now report on the experiments performed for logical and numerical similarity searches. In the small database, k nearest neighbors are retrieved relatively fast: The construction of the kd-tree on database $\mathcal{DB}_{\text{quad}}^{30\text{Hz}}$ is done in less then eight milliseconds, searching for 256 similar poses took 0.6 milliseconds in average.

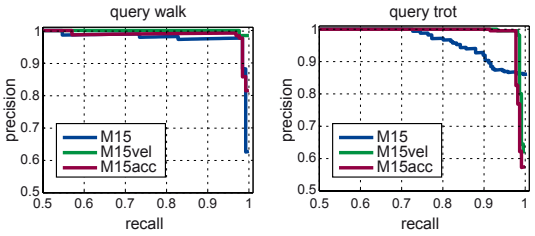

Figure 2: Precision-recall diagrams comparing feature sets based on position ($\mathcal{F}_{\text{M15}}^{\text{quad}}$), velocity ($\mathcal{F}_{\text{M15vel}}^{\text{quad}}$) and acceleration ($\mathcal{F}_{\text{M15acc}}^{\text{quad}}$) information. We show results for one representative query motion cycle of both motion classes walk and trot.

Numerical. For numerical similarity search, we have searched for similar motion cycles using the LNG. To come up with precision-recall diagrams, we extend the local pose neighborhood until all motion cycles of the query class are returned as match. We perform this experiment with all feature sets based on 3D information: $\mathcal{F}_{\text{M15}}^{\text{quad}}$,$\mathcal{F}_{\text{M15vel}}^{\text{quad}}$ and $\mathcal{F}_{\text{M15acc}}^{\text{quad}}$. Figure 2 shows the according diagrams for a representing walk and trot motion cycle. For the walking query, we obtain a high precision value (97%) up to a recall from 97%. For all feature sets, the precision drops for the last few matches only. In contrast for the trot motion cycle, more mismatches are returned when we use the position based feature set $\mathcal{F}_{\text{M15}}^{\text{quad}}$. Using derived feature sets $\mathcal{F}_{\text{M15vel}}^{\text{quad}}$ and $\mathcal{F}_{\text{M15acc}}^{\text{quad}}$, we obtain much better results. This behavior can be explained by a closer look on the motion classes: In both classes, walk and trot, the marker positions are not sufficiently distinct, while velocities and accelerations are.

Logical. Kovar and Gleicher (Kovar and Gleicher, 2004) have introduced the concept of logical similarity searches. Here, the retrieved matches of a query motion segment are used as new queries in new iteration of the searching process. New segments are retrieved until no new ones are found. We repeat this experiment with motion cycles from both classes. To this end, we restrict the number of nearest neighbors to 256, to ensure that no false positives are returned for a query motion cycle. In both cases, walk and trot, this retrieval scenario finds nearly all motion cycles without any mismatches. Figure 3 shows the numbers of new found motion cycles per iteration. The algorithm converges after four iterations in both cases.

7.2 Recognition of Activities

In this section, we focus on the results of our action recognition approach. We will first compare the simple $k$nn method versus the proposed LNG based vari-

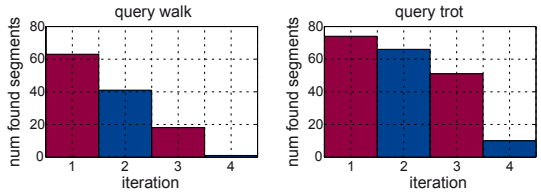

Figure 3: Number of found motion segments per iteration of the logical similarity search. We show results for one representative query motion cycle of the two motion classes walk and trot.

ant on motion capture data. Later we show some results obtained from video data.

Mocap Data. To investigate the difference between *k*nn voting per frame and the LNG based voting, we have performed experiments with the same representative motion cycles as in the previous section. Figure 4 shows the results for the simple *k*nn voting. The horizontal axis of this Figure describes a time line given in frames, while the vertical axis shows all motion classes represented by our database. We search for 256 nearest neighbors and counted per frame to which motion class these neighbors belong. This number of found nearest neighbors per frame is color-coded from white (no neighbor in this class) to black (256 neighbors in this class). Consequently, these graphs show a per frame confusion of the neighborhoods obtained with the respective method.

For the walking example, most of the neighbors belong to the walking class. Considering the trot motion cycle much more confusion between the two motion classes occurs. Still the majority belongs to the correct motion class in all frames, but this indicates, that this method is getting unstable, already in this simple scenario taking into account only two motion classes.

In comparison to these investigations, Figure 5 shows the results for the modified version with LNG. We keep the window length, taken into account for the graph construction, ten frames for these examples. Here, results for both example motions show a similar structure: No mislabeling are found in both cases, in return the number of retrieved neighbor poses from the LNG paths are much lower after the first couple of frames. For the first frame the result is the same, as for the direct *k*nn voting: The path LNG has a length of one frame only. With increasing path lengths the number of neighbors that can be connected with the graph drops down to approximately 50 paths per frame when the full window length is reached.

Video Data. In this case, we have tracked and extracted the feature set $\mathcal{F}_{\text{vid10}}^{\text{quad}}$for video sequence for each motion class. We search the 256 nearest neighbors in database $\mathcal{DB}_{\text{quad}}^{25\text{Hz}}$ and compute the according LNG paths. All considered frames are classified correctly, the results are shown in the supplemental video.

7.3 Video based Reconstructions

In order to elaborate the performance of proposed reconstruction framework, first we have sampled our database at different viewing directions for *k*d-tree construction as mentioned earlier in section 4. We have evaluated the effectiveness of the proposed methodology on variety of input examples like motion capture based, so called *synthetic* and video examples.

Mocap based Examples. We have testified our approach on walk and trot motions at different viewing directions for synthetic data. To this end, we have selected the range for viewing directions like azimuth angles from 0 to 180 degrees with 5 degrees step size (0-5-180) assuming that the same results would be executed for the second half of the circle. Similarly, for elevation angles, we start from 0 degree up to 60 degree with step size 10 degree (0-10-60), considering the fact that near to 90 degree or top view, the body of the performing horse becomes the hinderance in capturing the full detailed motion of the hooves.

For performance check, we have computed average reconstruction error between original motion and the reconstructed motion by calculating the average Euclidian distance in centimeters between them. We present this average reconstruction error with a graph as shown in Figure 6, where along x-axis azimuth angles (0-5-180) and along y-axis elevation angles (0-10-60) have been plotted, while the reconstruction error is color-coded from blue (low error) to red (high error).

For walk motion, it has been noticed that at *side view* either it is left side or right side, lowest reconstruction error is achieved: Here, the hooves of the horse are best visible. In *front view*, some lack of motion information appears which results in higher reconstruction error. We also observe that when the viewing direction moves to *top view*, the reconstruction error also increases 6(a).

Like for walk, the *side view* of the horse in trot motion gives detailed information due to which reconstruction error is reduced as compared to other views. The nearly same pattern of results like in walk, appears in case of *front view* and *top view*. Some dif-

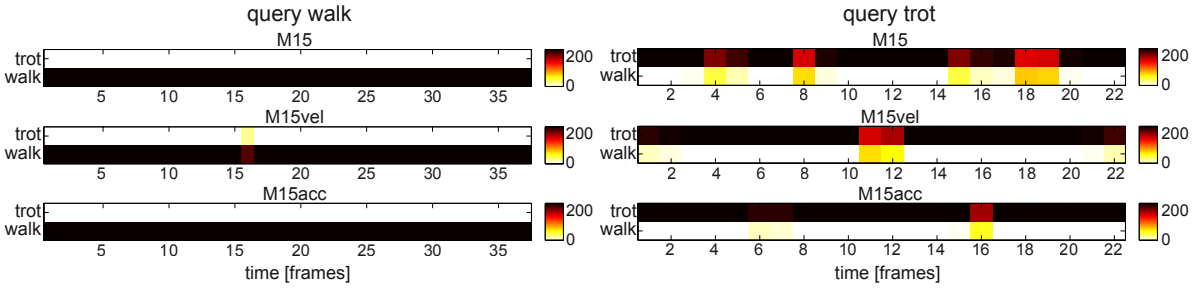

Figure 4: Visualization of the number of nearest neighbors per frame for walk and trot motion cycles. This Figure shows the results for the feature sets based on 3D information.

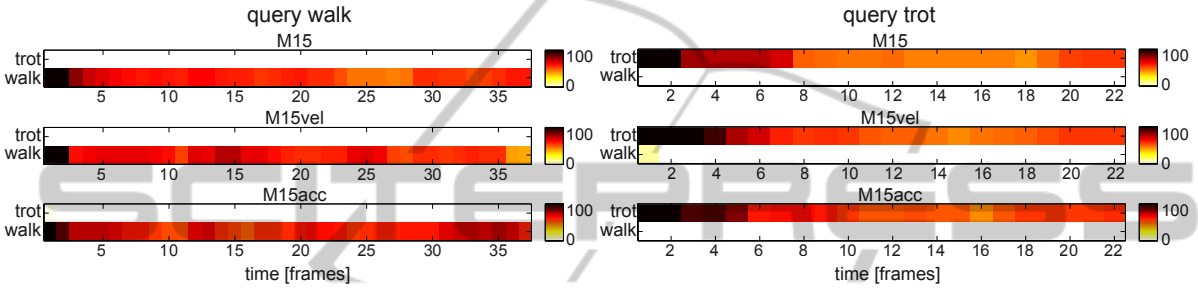

Figure 5: Visualization of nearest neighbors according to LNG paths per frame for walk and trot motion cycles. This Figure shows the results for the feature sets based on 3D information.

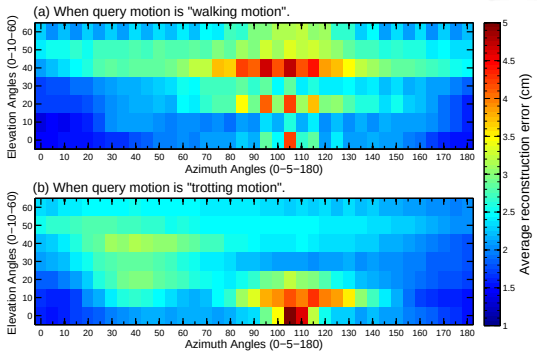

Figure 6: Average reconstruction errors for walking and trotting motions, with different viewing directions — azimuth angles (0-5-180) with step size 5 degree and elevation angles (0-10-60) with step size 10 degree.

ference is due to the reason that in trotting motions the positions hooves and head are different compared to walk. As a result, both *front view* and the *top view* execute less reconstruction error as compared to walking motions, see Figure 6(b).

Video Examples. We have checked our reconstruction approach on real videos of walk and trot motions. 2D feature sets $\mathcal{F}_{\text{vid10}}^{\text{quad}}$, which have been extracted from the input video, are given as input to the system. By performing different experiments, we have found that reconstruction results depend mainly on how accurate 2D feature sets $\mathcal{F}_{\text{vid10}}^{\text{quad}}$ are extracted from video data.

The challenges during feature extraction are: blurring effects, occlusion, illumination factors etc. which create hinderance in detection, tracking and extraction of feature sets from video data. Trot motions have more occlusions and blurring effects as compared to walk motions. Due to these factors, we can not detect or track the feature sets correctly for all frames. Thus, we have annotated key-frames in the video. Some results can be seen in supplementary video.

8 CONCLUSIONS AND FUTURE WORK

In this work, we have transferred techniques, developed for human motion data, to motion data from quadrupeds. For motion retrieval, the basic technique can be used without modifications, after defining suitable feature sets. Considering the results of the action recognition experiments, we show that extending the *k*nn search by a temporal component, even a simple approach can lead to good results. In the area of motion reconstruction from video data, we have adapted a technique that is known to work on human motion sequences. For the motion styles, represented by our database, we obtain satisfying results in both cases, synthetic 2D input query motions and real input video sequences. We are aware that all results have been

achieved on a relatively small database. Nevertheless, we believe that the results presented here show how quadruped data can be used in the context of data-driven animation. Therefore, one of the most important steps for further work is the creation of an enlarged motion capture database, to cover more types of gait and other typical exercises. With such type of data at hand, more sophisticated techniques for action recognition might be applied and compared in a reasonable manner. The motion reconstruction scenario might be extended to more complex scenarios, we have presented here results on a static camera with an object that is moving on a treadmill only. A reconstruction of motion sequences from a riding theater based on single video is one of our future goals. In this scenario, the types of motions are less restricted as compared to the current treadmill scenario. The skeleton representation of quadruped might be computed and helpful in the process of full body quadruped motion reconstruction. Another possible strand of research is the reconstruction of full-body movements based on accelerometer readings in outdoor scenarios. Another important step is the recording of other species in order to derive more general models of quadruped motion from such kind of data.

REFERENCES

Arikan, O., Forsyth, D. A., and O'Brien, J. F. (2003). Motion synthesis from annotations. *ACM Trans. Graph.*, 22:402–408.

Bobick, A. F., Davis, J. W., Society, I. C., and Society, I. C. (2001). The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:257–267.

Chai, J. and Hodgins, J. K. (2005). Performance animation from low-dimensional control signals. *ACM Trans. Graph.*, 24(3):686–696.

Elgammal, A. and su Lee, C. (2004). Inferring 3d body pose from silhouettes using activity manifold learning. In *In CVPR*, pages 681–688.

Favreau, L., Reveret, L., Depraz, C., and Cani, M.-P. (2004). Animal gaits from video. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, SCA '04, pages 277–286, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

Hobbs, S. J., Levine, D., Richards, J., Clayton, H., Tate, J., and Walker, R. (2010). Motion analysis and its use in equine practice and research. *Wiener Tierärztliche Monatsschrift*, 97:55–64.

Huang, T.-C., Huang, Y.-J., and Lin, W.-C. (2013). Real-time horse gait synthesis. *Computer Animation and Virtual Worlds*, 24(2):87–95.

Kovar, L. and Gleicher, M. (2004). Automated extraction and parameterization of motions in large data sets. *ACM Transactions on Graphics*, 23(3):559–568. SIGGRAPH 2004.

Krüger, B., Tautges, J., Weber, A., and Zinke, A. (2010). Fast local and global similarity searches in large motion capture databases. In *2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '10, pages 1–10, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

Moeslund, T. B., Hilton, A., and Krüger, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Comput. Vis. Image Underst.*, 104(2):90–126.

Müller, M., Röder, T., and Clausen, M. (2005). Efficient content-based retrieval of motion capture data. *ACM Trans. Graph.*, 24:677–685.

Müller, M., Röder, T., Clausen, M., Eberhardt, B., Krüger, B., and Weber, A. (2007). Documentation mocap database hdm05. Technical Report CG-2007-2, Universität Bonn.

Nauwelaerts, S., Kaiser, L., Malinowski, R., and Clayton, H. M. (2009). Effects of trunk deformation on trunk center of mass mechanical energy estimates in the moving horse, Equus caballus. *Journal of biomechanics*, 42(3):308–11.

Rosenhahn, B., Schmaltz, C., Brox, T., Weickert, J., and Seidel, H.-P. (2008). Staying well grounded in markerless motion capture. In *Proceedings of the 30th DAGM symposium on Pattern Recognition*, pages 385–395, Berlin, Heidelberg. Springer-Verlag.

Schuldt, C., Laptev, I., and Caputo, B. (2004). Recognizing human actions: A local svm approach. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 3 - Volume 03*, ICPR '04, pages 32–36, Washington, DC, USA. IEEE Computer Society.

Skrba, L., Reveret, L., Hetroy, F., Cani, M.-P., and O'Sullivan, C. (2008). Quadruped animation.

Tautges, J., Zinke, A., Krüger, B., Baumann, J., Weber, A., Helten, T., Müller, M., Seidel, H.-P., and Eberhardt, B. (2011). Motion reconstruction using sparse accelerometer data. *ACM Trans. Graph.*, 30(3):18:1–18:12.

Vögele, A., Hermann, M., Krüger, B., and Klein, R. (2012). Interactive steering of mesh animations. In *2012 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*.

Wei, X. and Chai, J. (2010). Videomocap: modeling physically realistic human motion from monocular video sequences. *ACM Trans. Graph.*, 29(4):42:1–42:10.

Wilhelms, J. and Van Gelder, A. (2003). Combining vision and computer graphics for video motion capture. *The Visual Computer*.

Yasin, H., Krüger, B., and Weber, A. (2013). Model based full body human motion reconstruction from video data. In *6th International Conference on Computer Vision / Computer Graphics Collaboration Techniques and Applications (MIRAGE 2013)*.