Scale-Independent Spatio-Temporal Statistical Shape Representations for 3d Human Action Recognition

Marco Körner[1], Daniel Haase[1] and Joachim Denzler[1]

Department of Mathematics and Computer Sciences, Chair for Computer Vision, Friedrich Schiller University of Jena, Ernst-Abbe-Platz 2, 07743 Jena, Germany

Keywords: Human Action Recognition, Manifold Learning, PCA, Shape Model

Abstract: Since depth measuring devices for real-world scenarios became available in the recent past, the use of 3d data now comes more in focus of human action recognition. We propose a scheme for representing human actions in 3d, which is designed to be invariant with respect to the actor's scale, rotation, and translation. Our approach employs Principal Component Analysis (PCA) as an exemplary technique from the domain of manifold learning. To distinguish actions regarding their execution speed, we include temporal information into our modeling scheme. Experiments performed on the CMU Motion Capture dataset shows promising recognition rates as well as its robustness with respect to noise and incorrect detection of landmarks.

1 INTRODUCTION AND RELATED WORK

In the last decades the recognition and analysis of actions and motions performed by humans have become one of the most promising fields in computer vision research and lead to a wide variety for research topics in computer vision. This family of problems aims to determine human activities automatically based on several sensor observations. A wide range of industrial as well as academic applications are based on this research, *e. g.* the interaction between humans and machines, surveillance and security, entertainment, video content retrieval as well as the research in medical and life sciences.

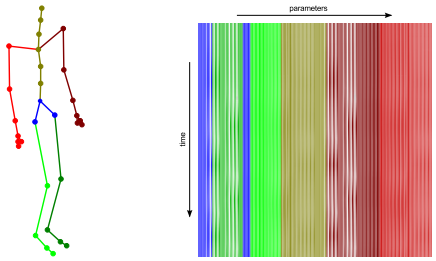
In early years of scientific interest those methods concentrated on the evaluation of 2d image sequences delivered by gray value or color cameras (Gavrila, 1999; Turaga et al., 2008; Poppe, 2010). Due to the massive amount of research those methods now achieve very good results on the standard Weizmann 2d action recognition dataset (Gorelick et al., 2007). Several of those approaches are based on the evaluation of changes in silhouettes or the extraction of interest point features in space-time volumes created by subsequent video frames (Laptev, 2005; Dollar et al., 2005). Furthermore the combination of shape and optical flow is used for action recognition (Ke et al., 2007).

In contrast to this huge amount of scientific work concerning 2d images, 3d data was not yet used in a remarkable quantity. However, the recent development of depth measuring devices such as *Time-of-Flight* (ToF) sensors or sensors based on the projection and capturing of structured light patterns make 3d data available in a fast and inexpensive way.

In this paper we present a spatio-temporal representation scheme for human actions given as sequences of 3d landmark positions which models the spatial variations in a contextual way and takes into account the temporal coherence between subsequent frames based on *manifold learning* techniques. After presenting our approach in Sec. 2 we show numerous experiments evaluated on the *CMU Motion Capture* (MoCap) dataset in Sec. 3. A summary and a brief outlook in Sec. 4 conclude this paper.

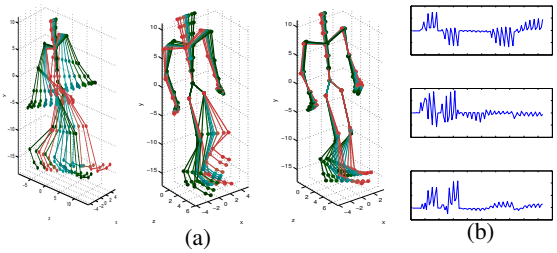
2 Statistical Shape Representation

Manifold learning techniques are widely used for classification tasks like face detection and emotion recognition (Zhang et al., 2005). For action recognition from 2d video streams the usability of *Principle Component Analysis* (PCA), and *Independent Component Analysis* (ICA) on motion silhouettes have been compared (Yamazaki et al., 2007). *Locality Preserving Projections* (LPP) were utilized in combination with a special Hausdorff distance measure on silhouettes (Wang and Suter, 2007). A comparison of further techniques for dimensionality reduction like *Locality Sensitive Discriminative Analysis* (LSDA) and *Local Spatio-Temporal Discriminant Embedding* (LSTDE) was presented in (Jia and Yeung, 2008). *Tensor PCA*



(a) (b)

Figure 1: (a) The skeleton model used in our approach with 31 joints affected by 62 degrees of freedom. (b) Data matrix for action class `walking` indicating the sequential parameter changes. The colors of the columns are corresponding to the colors in the model while the intensities illustrate the parameter values.



(a) (b)

Figure 2: The first three motion components of a `walking` action (a) and their corresponding eigenvectors (b). Colors indicate the weighting of the eigenvectors added to the mean shape (black: $w_k^t = -2\lambda_k^2$, blue: $w_k^t = 0$, red: $w_k^t = 2\lambda_k^2$). Note the anti-symmetric motion directions of the limbs in the first two components and the symmetric one in the third component.

for reducing the dimensionality of the parameter space was also investigated (Sun et al., 2011).

In the field of 3d action recognition far less work exist. *Laplacian Eigenmaps* are recently used to recognize human actions from 3d points delivered by full-body ToF scans (Schwarz et al., 2010; Schwarz et al., 2012). *Hierarchical Gaussian Process Latent Variable Modeling* (H-GPLVM) combined with *Conditional Random Fields* (CRF) was employed to model relations between limbs action classification from CMU MoCap data (Han et al., 2010).

For action recognition in 3d data a unique representation is necessary, which needs to be invariant against absolute landmark positions. While *Active Shape Models* (ASM) (Cootes et al., 1995) are massively used in facial expression classification, their main ideas are also suitable for the field of locomotion analysis (Haase and Denzler, 2011).

In the following we use a basic idea of ASMs to model and recognize human actions in 3d data.

2.1 Spatial Representation

Using a hierarchical skeleton model as shown in Fig. 1(a), any arbitrary skeleton configuration at time step $1 \leq t \leq N_\mathrm{f}$ can be parameterized as a vector of Euler angles $\boldsymbol{\theta}^t = \left(\theta_1^t, \ldots, \theta_{N_\theta}^t\right) \in \mathbb{R}^{N_\mathrm{f}}$, while N_θ is the number of joint angles. These angles are indicating the rotations of every limb wrt. the adjacent joints in any room direction limited to its number of *Degrees of Freedom* (DoF). *E. g.*, the neck joint has 3 DoF, because it can rotate in all coordinate directions, while the elbow has only 2 DoF. The model used in this paper consists of $N_\mathrm{j} = 31$ joints which yields 59 DoF and an additional global displacement $(x,y,z)_1^t$. When using ASMs, normally as a first step the landmark sets have to be aligned in terms of rotation, scale and translation using *Procrustes* analysis (Bookstein, 1997). This becomes obsolete in our scenario when normalizing the actor's skeleton in an anatomically correct fashion by setting the root rotation and translation to $\theta_1^t = \theta_2^t = \theta_3^t = x_1^t = y_1^t = z_1^t = 0, \quad 1 \leq t \leq N_\mathrm{f}$.

While angular representations tend to be ambiguous because of their periodical nature, joint rotations are projected to 3d landmark positions

$$\boldsymbol{l}^t = \underset{\boldsymbol{\theta}\to\boldsymbol{l}}{\pi}\left(\boldsymbol{\theta}^t\right) = \left((x,y,z)_1^t, \ldots, (x,y,z)_{N_\mathrm{j}}^t\right) \in \mathbb{R}^{N_\mathrm{l}}, \quad (1)$$

$1 \leq t \leq N_\mathrm{f}$, using a projection function $\underset{\boldsymbol{\theta}\to\boldsymbol{l}}{\pi} : \mathbb{R}^{N_\theta} \mapsto \mathbb{R}^{N_\mathrm{l}}$. To preserve scale invariance of our modeling, a predefined skeleton model is used for projection each time.

Combining all zero-mean skeleton configurations at every available time step yields the matrix of landmarks

$$\boldsymbol{L} = \begin{pmatrix} \boldsymbol{l}^1 - \boldsymbol{l}_\mu \\ \vdots \\ \boldsymbol{l}^{N_\mathrm{f}} - \boldsymbol{l}_\mu \end{pmatrix} \in \mathbb{R}^{N_\mathrm{f} \times N_\mathrm{l}}, \quad \boldsymbol{l}_\mu = \frac{1}{N_\mathrm{f}} \sum_{i=1}^{N_\mathrm{f}} \boldsymbol{l}^i. \quad (2)$$

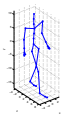
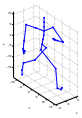
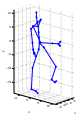
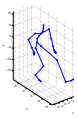
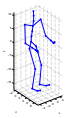
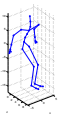
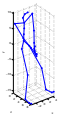
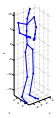
Performing *Principle Component Analysis* (PCA) on $\boldsymbol{L}$ will return its matrix

$$\boldsymbol{P}_{\boldsymbol{L}} = (\boldsymbol{v}_1^{\boldsymbol{L}} | \cdots | \boldsymbol{v}_{N_\mathrm{l}}^{\boldsymbol{L}}) \in \mathbb{R}^{N_\mathrm{l} \times N_\mathrm{l}} \quad (3)$$

of eigenvectors sorted according to their corresponding eigenvalues $\lambda_k^{\boldsymbol{L}}$ descendingly representing the importance of each data space direction. Using these eigenvectors as basis vectors, every arbitrary skeleton configuration represented by a 3d landmark coordinate set can be expressed as a linear combination $\boldsymbol{l}' = \boldsymbol{l}_\mu + \boldsymbol{P}_{\boldsymbol{L}} \boldsymbol{b}_{\boldsymbol{l}'}$ of the data matrix columns and the frame-specific *shape parameter* vector $\boldsymbol{b}_{\boldsymbol{l}'}$ added to the constant *mean shape* $\boldsymbol{l}_\mu$.

Since the amount of represented variances of landmark sequences captured by the eigenvectors decreases massively according to the evolution of their corresponding eigenvalues, the number of columns in the

Table 1: Action classes selected from CMU MoCap dataset used in our experiments.

Class	 walking	 running	 marching	 sneaking	 hopping	 jumping	 golfing	 salsa
Samples	38	28	14	15	14	9	11	30
Actors	9	9	4	5	4	3	2	2
Avg. frame number	1283	853	6426	4200	602	1325	8626	5224

eigenvector matrix $\boldsymbol{P_L}$ can be restricted to achieve a substantial reduction of dimensionality.

Fig. 2(a) shows the first three action components, while Fig. 2(b) depicts the corresponding eigenvectors of an action from class `walking`.

2.2 Integration of Temporal Context

While the previously described representation solely models linear variations of skeleton joints, the temporal evolution of configurations might contribute helpful information for the recognition and analysis of articulated actions. For this reason, our model is extended to include this temporal component.

In (Bosch et al., 2002) such a temporal modeling of periodical actions was already used to model a beating heart. This was pointed out to be a generalization of the multi-view integration approach of (Lelieveldt et al., 2003) and (Oost et al., 2006). Instead of considering a skeleton configuration at a single time step t_0 to obtain the model parameters, they regard a series of sequential time steps $t_0 < t_1 < \ldots < t_k$ or alternatively multiple views $(o_1, o_2, \ldots, o_k)$ at the same time step as a single configuration.

Applied to our problem, the provided method models the temporal evolution of skeleton configurations by appending subsequent input matrices horizontally:

$$\boldsymbol{l}^{t_0 \to t_{k_{\text{hist}}}} = \left(\boldsymbol{l}^{t_0}, \boldsymbol{l}^{t_1}, \cdots, \boldsymbol{l}^{t_{k_{\text{hist}}}}\right) \in \mathbb{R}^{(k_{\text{hist}}+1)\cdot N_{\text{l}}}, \tag{4}$$

$$\boldsymbol{L}^{t_0 \to t_{k_{\text{hist}}}} = \begin{pmatrix} \boldsymbol{l}^{t_0} & \boldsymbol{l}^{t_1} & \cdots & \boldsymbol{l}^{k_{\text{hist}}} \\ \boldsymbol{l}^{t_1} & \boldsymbol{l}^{t_2} & \cdots & \boldsymbol{l}^{t_{k_{\text{hist}}+1}} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{l}^{t_{N_{\text{f}}-k_{\text{hist}}}} & \boldsymbol{l}^{t_{(N_{\text{f}}-k_{\text{hist}})+1}} & \cdots & \boldsymbol{l}^{t_{N_{\text{f}}}} \end{pmatrix} \in \mathbb{R}^{(k_{\text{hist}}+1)\cdot N_{\text{l}} \times (N_{\text{f}}-(k_{\text{hist}}+1))}.$$

This approach allows us to distinguish between an action and its reverse counterpart as well as to classify the speed of execution.

3 EXPERIMENTS

3.1 Dataset

In order to evaluate the proposed methods, we have chosen eight different actions performed by different actors from the CMU MoCap dataset, as shown in more detail in Tab. 1. While we have selected common actions with slightly different executions like `walking`, `running`, `marching` and `sneaking` or `hopping` and `jumping`, we also took complex motions—`salsa` and `golfswinging`—into account.

3.2 Discriminability of Eigenvector Representation

When performing PCA on sequential data $\boldsymbol{L}$, the result shows the most important directions of variance in the data. For this reason, the eigenvectors $\boldsymbol{v}_k^{\boldsymbol{L}}$ corresponding to the largest eigenvalues $\lambda_k^{\boldsymbol{L}}$ are supposed to encode most of the information, while the eigenvectors corresponding to the lower eigenvalues model only minor changes in the data as well as noise.

Fig. 3 depicts the evolution of the eigenvalues for all action classes in our dataset. As can be seen, after a strong descent up to the third principal component, the eigenvalues converged strongly towards zero. After a certain component, there was no substantial contribution to the data, which became apparent at the 12^{th} eigenvalue, as indicated by the vertical line in Fig. 3. As depicted in Tab. 2, in most the cases two to three eigenvectors were sufficient to cover 90% of the variances occurred while execution of an action. Solely the action classes with high variances in all directions need more discriminability, which can be handled by increasing the number of eigenvectors. This fact can also be seen in Tab. 3, where the first three eigenvectors $\boldsymbol{v}_k^{\boldsymbol{L}}$ are shown together with their mean shapes $\boldsymbol{l}_{\mu}$.

The back projection error $\varepsilon_{\text{action}}(\boldsymbol{l}') = \|\left(\boldsymbol{l}' \cdot \boldsymbol{P}_{\boldsymbol{L}^{\text{action}}}\right) \cdot \boldsymbol{P}_{\boldsymbol{L}^{\text{action}}}^{\top} - \boldsymbol{l}'\|_2$ obtained by transforming an arbitrary skeleton configuration $\boldsymbol{l}'$ from

Table 2: Comparison of variance covering facilities of our representation scheme. While the usage of two to three eigenvectors allows to achieve 90% of the variances obtained during simple actions, more dimensions are needed to represent more complex actions.

Action Class	Amount of variance		
	90%	95%	98%
walking	2	3	4
running	2	3	5
marching	3	5	8
sneaking	3	5	8
hopping	2	5	8
jumping	3	4	6
golfing	3	3	4
salsa	8	9	12

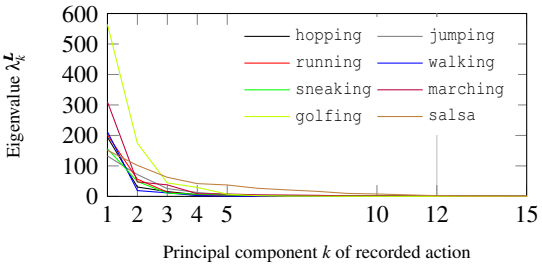


Figure 3: Evolution of eigenvalues for different action classes. Eigenvalues are decreasing massively up to the third component, while they remain static for higher-order components.

Euclidian space $\mathbb{R}^3$ into the reduced eigenspace $\mathbb{V}_{\text{action}}$ of a certain action class and back to Euclidian space, where $\boldsymbol{P}_{\boldsymbol{L}^{\text{action}}} = \left(\boldsymbol{v}_1^{\boldsymbol{L}^{\text{action}}} \middle| \cdots \middle| \boldsymbol{v}_{k_{\text{ev}}}^{\boldsymbol{L}^{\text{action}}}\right)$ is a matrix containing the eigenvectors corresponding to the first k_{ev} largest eigenvalues of $\boldsymbol{L}^{\text{action}}$, give a quantitative justification for this postulation, as can be seen in Fig. 4. As a result, the ordering of remaining eigenvectors is no longer meaningful. Therefore, they are not considered in the following classification purposes.

3.3 Feature Vector Design and Classification

In order to distinguish action classes, features have to be derived from the sequence of skeleton configurations. Using the representation described before, feature vectors $\boldsymbol{y}_{\boldsymbol{L}'} = \left(\boldsymbol{l}'_{\mu}, \boldsymbol{v}_1^{\boldsymbol{L}'}, \ldots, \boldsymbol{v}_{k_{\text{ev}}}^{\boldsymbol{L}'}\right)$ are extracted from a series $\boldsymbol{L}'$ of landmark vectors $\boldsymbol{l}'$ by concatenating its mean shape $\boldsymbol{l}'_{\mu}$ and its eigenvectors corresponding to the first k_{ev} eigenvalues.

Table 3: Comparison of the mean shapes and the first three eigenvectors of the action classes in our dataset. Note that similar actions have similar first eigenvectors and different second or third eigenvectors while different actions can already be distinguished by their first eigenvectors.

Action Class	Mean Shape	Eigenvectors		
	$\boldsymbol{l}_{\mu}$	$\boldsymbol{v}_1^{\boldsymbol{L}}$	$\boldsymbol{v}_2^{\boldsymbol{L}}$	$\boldsymbol{v}_3^{\boldsymbol{L}}$
walking				
running				
marching				
sneaking				
hopping				
jumping				
golfing				
salsa				

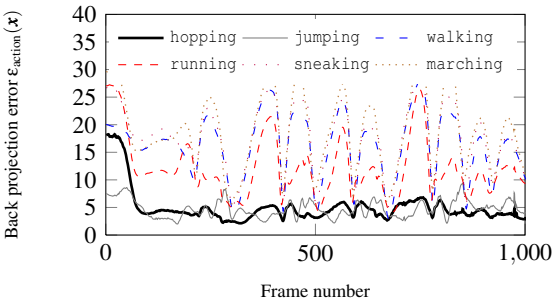
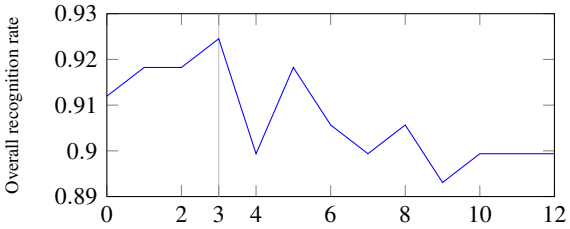


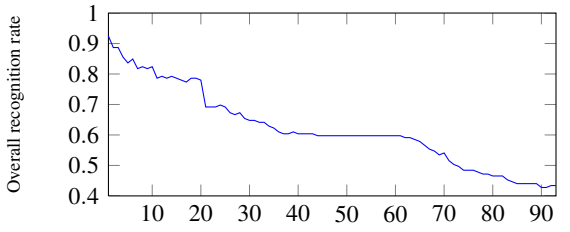
Figure 4: Back projection errors obtained by transforming skeleton configurations in every time step of a hopping sequence (thick line) from Euclidean space into action-specific eigenspaces and back to Euclidean space. Small errors suggest that the mapping is appropriate for the given action representation, while high errors are indicating poor mapping facilities.

In Fig. 5(a) one can observe that the recognition rate during classification had a maximum peak at $k_{\text{ev}} = 3$, which argues for a high degree of discriminability. This is emphasized by the vertical line in Fig. 5(a). Without using any eigenvectors, only the mean shape is taken into account during feature extraction, which leads to lower discriminability. Using more eigenvectors would cause a more exact reconstruction of the skeleton configuration and therefore a smaller discriminability due to the increased coverage of variability.

For simplicity, we used the *k Nearest Neighbor* (k-NN) framework for classification, which assigns a class label to a feature vector employing an arbitrary distance measure $d(\boldsymbol{y}_{\text{test}}, \boldsymbol{y}_{\text{target}})$ between the feature



(a) Number of eigenvectors k_{ev}



(b) Number of neighbors k

Figure 5: Effect of increasing (a) the number of eigenvectors used for building the feature vector and (b) the number of neighbors for k-NN classification on recognition rates.

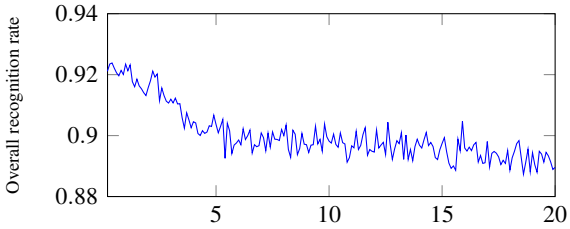
Table 4: Confusion matrix with overall recognition rates obtained by exhausting leave-one-out test on our dataset.

Training	Testing							
	walking	running	marching	sneaking	hopping	jumping	golfing	salsa
walking	**100**	0	0	0	0	0	0	0
running	33	**56**	0	11	0	0	0	0
marching	4	0	**93**	0	0	4	0	0
sneaking	0	0	0	**100**	0	0	0	0
hopping	13	7	0	0	**80**	0	0	0
jumping	0	0	7	7	0	**86**	0	0
golfing	0	0	0	0	0	0	**100**	0
salsa	0	0	3	0	0	0	0	**97**

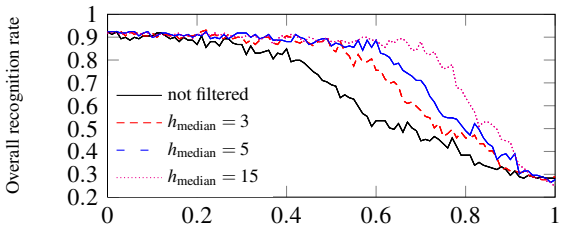
vector $\boldsymbol{y}_{\text{test}}$ and a representative prototype vector $\boldsymbol{y}_{\text{target}}$. In our experiments, we chose the Euclidean distance $d(\boldsymbol{y}_{\text{test}}, \boldsymbol{y}_{\text{target}}) = \|\boldsymbol{y}_{\text{test}} - \boldsymbol{y}_{\text{target}}\|_2$.

As can be seen in Fig. 5(b), using $k = 1$ gave the best recognition rate, while increasing the number of neighbors caused apparent worse results as well as higher computational time for classification.

Using this feature extraction scheme and the 1-NN classifier, we were able to achieve results as shown in the confusion matrix obtained by exhaustive leave-one-out test in Tab. 4. As one can see, most of the action classes in our dataset were recognized correctly in more than 80% of the cases, while 4 classes gave recognition rates of nearly 100%. Solely the action class `running` has been confused with the semantically related classes `walking` and `sneaking` due to their similar variations during execution.



(a) Gaussian noise variance σ^2_{gauss}



(b) Amount of *Salt-and-Pepper* noise p

Figure 6: Influence of increasing the strength of (a) zero-mean gaussian noise and (b) *Salt-and-Pepper* noise to the recognition rates. In order to reduce this performance drop in (b), a temporal median filter of size h_{median} was applied on the data as a preprocessing step.

3.4 Robustness to Noise

In real-world applications the input data for action classification are not ideal. Hence we modeled the influence of additive, zero-mean normally distributed, and uncorrelated *Salt-and-Pepper* noise to quantitatively evaluate the robustness of our approach.

As can be seen in Fig. 6(a), adding Gaussian noise to the input data did not negatively affect the classification results. This might be explained by the mean subtraction on the one hand and the usage of PCA on the other hand during modeling. In order to find the principal components, noise added to the data will only affect the eigenvectors corresponding to the smaller eigenvalues, while the inherent and consistent information of movement over time is still captured by the eigenvectors corresponding to the larger eigenvalues.

A similar behavior can be observed in the case of adding uncorrelated Salt-and-Pepper noise to input data. As can be seen in Fig. 6(b), while the recognition rates were decreasing with the amount of added noise, simple median filters applied to the single channels along the time dimension were able to drastically reduce these effects. It can be seen that an amount of 70% Salt-and-Pepper noise can be handled by applying a 15-frame temporal median filter which only results in a small decrease in the recognition rates.

Table 5: Effects of integrating temporal context into our model. Since the model became more distinctive regarding the execution speed of actions, integrating these temporal information affected the recognition rates slightly.

Historical Offset Δ_{hist}	Number of History Frames k_{hist}			
	1	2	3	4
5	92.45	92.45	91.82	91.19
10	93.08	92.45	91.82	92.45
15	91.82	91.19	91.82	91.82
30	90.57	89.31	92.45	89.94

3.5 Comparison to other Work

Although human action recognition was widely investigated for 2d data, there is less work available concerning the case of having access to 3d data. A similar approach to classify human actions in 3d data was taken in (Han et al., 2010), but they selected less action classes from the CMU MoCap dataset. While they distinguish only 3 action classes with small variations in execution, recognition rates of 98.29% were obtained without taking the presence of noise into account. In (Junejo et al., 2011), the same database has been used to create artificial 2d views and evaluating several distance metrics on the landmark points without modeling the shape at all. They observed recognition rates of about 90.5% in average when combining all their camera views for training and testing. The approach of (Shen et al., 2008) employed homography constraints and lead to an overall recognition rate of about 92% .

Compared to those results, our approach performs similarly (92.45%) on the same data even in the presence of noise.

3.6 Use of Temporal Context

As mentioned in Sec. 2.2, we not only model the variations of landmark transitions during a fixed time period, but also integrate the evolution of these movements by incorporating the temporal context during an action execution.

Tab. 5 shows that the integration of temporal information into the action model affects the recognition rates slightly. We tested several values for the number of history frames k_{hist} integrated to the model as well as the temporal offset $\Delta_{\text{hist}} = (t_i - t_{i-1}), 1 \leq i \leq t_{k_{\text{hist}}}$ of these frames. The observed behavior can be explained by taking into account the variability of action executions within the dataset, where, for example, one actor performs slower while another performs faster.

Although this fact is not requested in the given scenario, it would allow us to distinguish actions regarding the execution speed which can be of interest in further applications. For instance, the confusion between action classes `running` and `walking` or `sneaking` could be dissolved exploiting these temporal information.

4 SUMMARY AND OUTLOOK

We proposed a method for representing sequences of human actions while integrating spatial and temporal information into a combined model. This representation scheme was shown to be suitable for human action classification applications. Experiments performed on the CMU motion capturing dataset gave promising results which are able to compete with existing state of the art approaches.

To overcome certain false classifications, a hierarchy of single binary classifiers can be built. One can observe that similar motions are grouped into closer subtrees, while diverging actions are located in distinct subtrees.

Another field of research is the design of features used for classification. Since closely related classes tend to be confused, more sophisticated features should help to overcome this behavior.

The parameter vector $\boldsymbol{b}_{l'}$ could be used to build a self-similarity matrix instead of using the Euclidian landmark distances as proposed by (Junejo et al., 2011). More sophisticated distance measures like the angular distance in the manifold space could benefit the discriminability of the action classes.

For feeding real-world data to our approach, skeleton configurations can be extracted from frames provided by depth measuring camera devices such as Microsoft Kinect or PMD, which was recently shown to be possible in real-time (Li et al., 2010; Shotton et al., 2011). The combination of Active Shape Model based landmark detection and our proposed action representation could also be promising.

5 ACKNOWLEDGEMENTS

The data used in our experiments was obtained from `mocap.cs.cmu.edu`. The database was created with funding from NSF EIA-0196217.

REFERENCES

Bookstein, F. L. (1997). Landmark methods for forms without landmarks: Morphometrics of group differences in outline shape. *Medical Image Analysis*, 1(3):225–243.

Bosch, J., Mitchell, S., Lelieveldt, B., Nijland, F., Kamp, O., Sonka, M., and Reiber, J. (2002). Automatic segmentation of echocardiographic sequences by active appearance motion models. *IEEE Transactions on Medical Imaging*, 21(11):1374–1383.

Cootes, T. F., Taylor, C. J., Cooper, D. H., and Graham, J. (1995). Active shape models—their training and application. *Computer Vision and Image Understanding*, 61:38–59.

Dollar, P., Rabaud, V., Cottrell, G., and Belongie, S. (2005). Behavior recognition via sparse spatio-temporal features. In *Proceedings of the 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 65–72. IEEE Computer Society.

Gavrila, D. (1999). The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98.

Gorelick, L., Blank, M., Shechtman, E., Irani, M., and Basri, R. (2007). Actions as space-time shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(12):2247–2253.

Haase, D. and Denzler, J. (2011). Anatomical landmark tracking for the analysis of animal locomotion in x-ray videos using active appearance models. In Heyden, A. and Kahl, F., editors, *Image Analysis*, volume 6688 of *Lecture Notes in Computer Science*, pages 604–615. Springer Berlin / Heidelberg.

Han, L., Wu, X., Liang, W., Hou, G., and Jia, Y. (2010). Discriminative human action recognition in the learned hierarchical manifold space. *Image and Vision Computing*, 28(5):836–849.

Jia, K. and Yeung, D.-Y. (2008). Human action recognition using local spatio-temporal discriminant embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8.

Junejo, I., Dexter, E., Laptev, I., and Perez, P. (2011). View-independent action recognition from temporal self-similarities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):172–185.

Ke, Y., Sukthankar, R., and Hebert, M. (2007). Spatio-temporal shape and flow correlation for action recognition. In *Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, Visual Surveillance Workshop*, pages 1–8.

Laptev, I. (2005). On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123.

Lelieveldt, B. P. F., Üzümcü, M., van der Geest, R. J., Reiber, J. H. C., and Sonka, M. (2003). Multi-view active appearance models for consistent segmentation of multiple standard views: application to long and short-axis cardiac mr images. In *Proceedings of the 17th International Congress and Exhibition on Computer Assisted Radiology and Surgery*, pages 1141–1146.

Li, W., Zhang, Z., and Liu, Z. (2010). Action recognition based on a bag of 3d points. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 9–14.

Oost, E., Koning, G., Sonka, M., Oemrawsingh, P., Reiber, J., and Lelieveldt, B. (2006). Automated contour detection in x-ray left ventricular angiograms using multiview active appearance models and dynamic programming. *IEEE Transactions on Medical Imaging*, 25(9):1158–1171.

Poppe, R. (2010). A survey on vision-based human action recognition. *Image and Vision Computing*, 28(6):976–990.

Schwarz, L. A., Mateus, D., Castaneda, V., and Navab, N. (2010). Manifold learning for tof-based human body tracking and activity recognition. In *Proceedings of the British Machine Vision Conference*, pages 80.1–80.11. BMVA Press.

Schwarz, L. A., Mateus, D., and Navab, N. (2012). Recognizing multiple human activities and tracking full-body pose in unconstrained environments. *Pattern Recognition*, 45(1):11–23.

Shen, Y., Ashraf, N., and Foroosh, H. (2008). Action recognition based on homography constraints. In *Proceedings of the 19th International Conference on Pattern Recognition*, pages 1–4.

Shotton, J., Fitzgibbon, A. W., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. (2011). Real-time human pose recognition in parts from single depth images. In *Proceddings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1297–1304.

Sun, M.-F., Wang, S.-J., Liu, X.-H., Jia, C.-C., and Zhou, C.-G. (2011). Human action recognition using tensor principal component analysis. In *Proceedings of the 4th IEEE International Conference on Computer Science and Information Technology*, pages 487–491.

Turaga, P., Chellappa, R., Subrahmanian, V., and Udrea, O. (2008). Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1473–1488.

Wang, L. and Suter, D. (2007). Learning and matching of dynamic shape manifolds for human action recognition. *IEEE Transactions on Image Processing*, 16(6):1646–1661.

Yamazaki, M., Chen, Y.-W., and Xu, G. (2007). Human action recognition using independent component analysis. In *Intelligence Techniques in Computer Games and Simulations*.

Zhang, J., Li, S. Z., and Wang, J. (2005). Manifold learning and applications in recognition. In Tan, Y.-P., Yap, K., and Wang, L., editors, *Intelligent Multimedia Processing with Soft Computing*, volume 168 of *Studies in Fuzziness and Soft Computing*, pages 281–300. Springer-Verlag.