ON-LINE ACTION RECOGNITION FROM SPARSE FEATURE FLOW

Hildegard Kuehne[1], Dirk Gehrig[2], Tanja Schultz[2] and Rainer Stiefelhagen[1]

[1] *Computer Vision for Human-Computer Interaction Lab, Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany*

[2] *Cognitive Systems Lab, Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany*

Keywords: Action Recognition, Motion Analysis, Sequence Analysis, Human Computer Interaction.

Abstract: The fast and robust recognition of human actions is an important aspect for many video-based applications in the field of human computer interaction and surveillance. Although current recognition algorithms provide more and more advanced results, their usability for on-line applications is still limited. To bridge this gap a on-line video-based action recognition system is presented that combines histograms of sparse feature point flow with an HMM-based action recognition. The usage of feature point motion is computational more efficient than the more common histograms of optical flow (HoF) by reaching a similar recognition accuracy. For recognition we use low-level action units that are modeled by Hidden-Markov-Models (HMM). They are assembled by a context free grammar to recognize complex activities. The concatenation of small action units to higher level tasks allows the robust recognition of action sequences as well as a continuous on-line evaluation of the ongoing activity. The average runtime is around 34 ms for processing one frame and around 20 ms for calculating one hypothesis for the current action. Assuming that one hypothesis per second is needed, the system can provide a mean capacity of 25 fps. The systems accuracy is compared with state of the art recognition results on a common benchmark dataset as well as with a marker-based recognition system, showing similar results for the given evaluation scenario. The presented approach can be seen as a step towards the on-line evaluation and recognition of human motion directly from video data.

1 INTRODUCTION

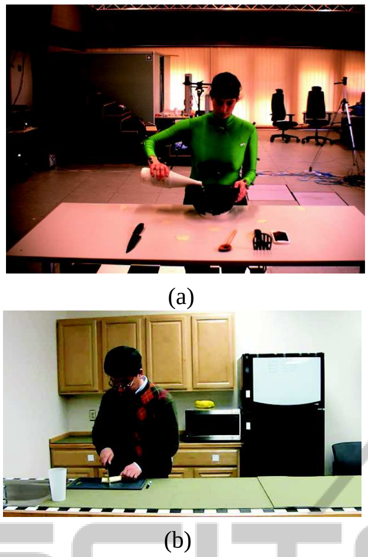
The recognition of human action is a growing field, perhaps even one of the key topics for human computer interaction and surveillance applications. It can be applied in the context of simple communicative interaction like waving or pointing, but also help to understand complex tasks and enable reasonable service, e.g. in the context of service or industry robots. One of the main goals in this field is the understanding of what the current behavior aims at and the context in which this happens. This would allow a forward-looking and anticipatory behavior and enable the support of the current task execution and the adaption to the users needs.

The following paper presents a system for the video-based recognition of complex tasks in order to allow a recognition of basic actions and to understand the intention behind. It works on-line and is able to recognize the ongoing action. The system combines three components: first, the video images are converted into global histograms of sparse feature flow. This can be seen as a valuable alternative to histograms of oriented flow (HOF), as feature based histograms can reach a similar recognition performance while being more efficient allowing on-line application as they are needed in the field of human computer interaction. The second component is the HMM-based recognition of small action units based on the feature flow histogram input. In a third step, the action units are combined by a higher level grammar that guides the concatenation of small action units into a meaningful sequence and so, the recognition of the overall task.

The here presented scenario takes place in the household domain considering typical kitchen tasks like cutting fruits or pouring a glass of water. Examples for such a setting can be seen in Figure 1. Each complex task is decomposed into action units, and a grammar has been set up that allows the combination of the action units to continuous action sequences.

We show that the recognition performance with the histograms of sparse feature flow is comparable to

Kuehne H., Gehrig D., Schultz T. and Stiefelhagen R..
ON-LINE ACTION RECOGNITION FROM SPARSE FEATURE FLOW.
DOI: 10.5220/0003861506340639
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2012), pages 634-639
ISBN: 978-989-8565-03-7


(a)

(b)

Figure 1: Example for action sequences in a kitchen scenario: a) 'Basic Kitchen Tasks' dataset: pouring water into a bowl (Gehrig et al., 2009), b) 'Activities of Daily Living' dataset: chopping a banana (Messing et al., 2009).

the one with optical flow histograms as well as with state of the art systems and that both approaches have similar recognition rates as the marker based system in the given scenario. To be able to compare the recognition performance of motion histograms with the one of marker based recognition systems, half of the performed tasks of the here presented 'Basic Kitchen Tasks' dataset were captured with video as well as with a commercial marker based motion capture system from Vicon. Additionally, we evaluated the runtime of the optical flow as well as of the feature based approach, showing that the feature based approach is fast enough to allow a on-line recognition during execution.

2 RELATED WORK

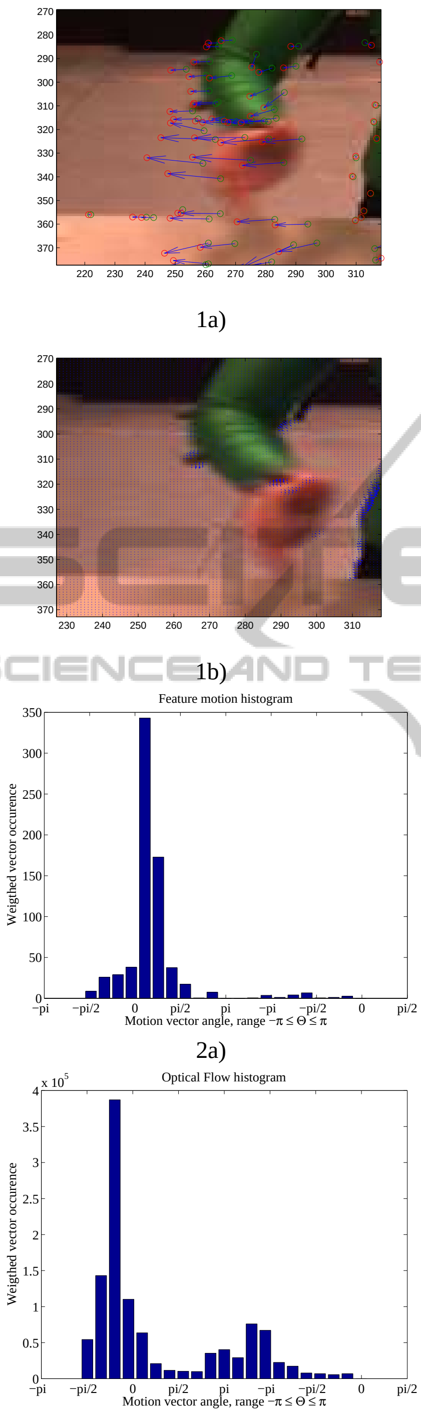
The use of global and local histograms has become a more and more important technique in the context of action recognition for a lot of different application scenarios, e.g. presented by Efros et al. (Efros et al., 2003) in the context of sports, by Marszalek et al. (Marszalek et al., 2009) for video and movie databases or by Danafar and Gheissari (Danafar and Gheissari, 2007) for surveillance applications.

Many approaches use local accumulated optical flow histograms, i.e. Lucena et al. (Lucena et al., 2009). The flow histograms are computed from a number of tiles in the region of interest. The input vector is a concatenation of the aggregated histograms. A closely related approach that is also build on tiled optical flow histograms but that focuses on the modeling of HMMs for recognition is presented by Mendoza et al. (Mendoza et al., 2009). They split the region of interest into 8 tiles and calculate optical flow histograms with 4 bins for magnitude and 8 bins for orientation for each tile. After a PCA this 256D feature vector reduces to a 32D vector which is used for recognition. For the modeling of actions they propose products of HMMs (PoHMM).

More abstract approaches are dealing with the syntactic structure of actions and tasks. The decomposition and concatenation of complex tasks has e.g., been described by Ivanov and Bobick (Ivanov and Bobick, 2000). The approach proposes the decomposition of complex action into smaller tasks and their reassembling by a higher level grammar. Following this idea, the task of action recognition is also split up into two steps. First small action units had to be recognized, e.g. by simple HMMs, then the result of this low level recognition is processed by a higher level stochastic action grammar.

3 FEATURE FLOW HISTOGRAMS

Motion information can be gained from dense optical flow fields or from tracking of feature points only. The feature tracking used in this paper is based on the Lucas-Kanade method described in (Lucas and Kanade, 1981) and (Tomasi and Kanade, 1991). The initialization and tracking of features follows the pyramidal KLT feature tracking implementation by (Koehler and Woerner, 2008). The initialization of new features is done for every frame following the algorithms of Shi and Tomasi (Shi and Tomasi, 1994). Every frame of the video sequence is represented by a global histogram of its overall motion directions without any further local information. The weighted histogram for frame t is calculated from the motion vector of the feature points of images I at time index t and $t+1$ (I_t, I_{t+1}). The motion vector $(u(\delta t), v(\delta t))$ of the feature is used to calculate the resulting motion direction θ, indicated by an angle value from $[-\pi, \pi]$ and γ defining the motion intensity. The feature motion directions are weighted with their norm values. The elements for one bin of the histogram are calculated based on the motion angle θ. As the motion angle ranges from $[-\pi, \pi]$, the vector of elements for the k-th bin $h(k)$ of a histogram with n bins can be defined as:



1a)

1b)

2a)

2b)

Figure 2: Comparison of feature motion (a) and optical flow (b) histogram: 1) example for vector plot, 2) bar plot of weighted motion histograms.

$$h(k) = \{(u,v)|\theta(u,v) \geq \frac{(k2\pi)}{n} - \pi \quad \cap \quad \theta(u,v) < \frac{((k+1)2\pi)}{n} - \pi\} \quad . \tag{1}$$

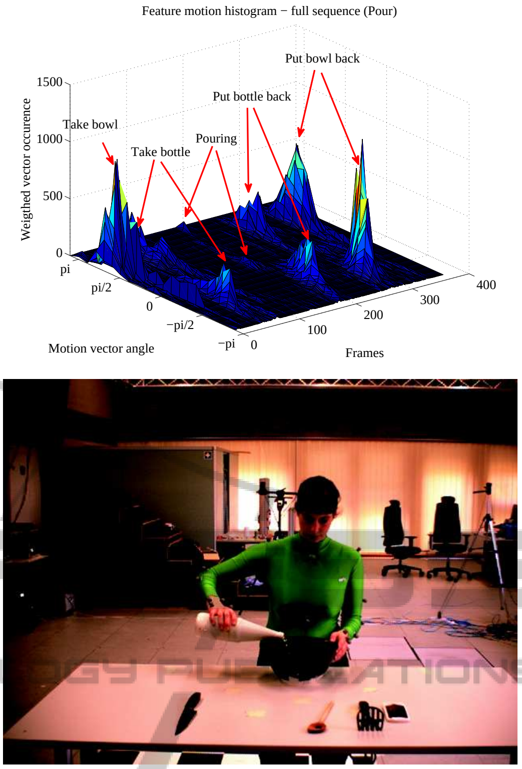


Figure 3: Example for feature motion histogram distribution for action sequence ‘Pouring’.

The number of elements in $h(k)$ is indicated by $N(h(k))$, and the elements represent the motion vectors (u,v) of the related feature points. The k-th bin for the weighted histogram is calculated from the intensity of all elements in the vector as shown in

$$H(k) = \sum_{i=1}^{N(h(k))} \gamma(h(k_i)). \tag{2}$$

Examples for the feature flow motion compared to the optical flow motion as well as the resulting histograms can be seen in Figure 2. The histograms are sampled over time resulting in a 30-dimensional input vector for the HMMs. An example for the histogram distribution over a complete action sequence can be seen in Figure 3.

4 ACTION UNITS AND GRAMMAR

Complex tasks, in this case in the household domain, usually consist of concatenated action units. If someone wants to cut vegetables, one usually has to take it, get a knife, start cutting, put the knife back etc. Action units in this context refer to a motion that is performed continuously and without interruption. So, action units are the smallest entity, which order can be changed during the execution, for example is it possible to first take the knife then the vegetables, but it could also be done the other way around. Additionally all tasks, as long as they have a meaningful aim, have to be executed in a certain order. It would not make sense to start cutting vegetables without holding a knife or without the vegetables in front. As the order in which the different tasks are executed is not random, it is possible to formulate a grammar, which has to be followed. This action grammar defines the action sequences, which are a concatenation of action units that result in a meaningful task. A example for a simplified grammar can be seen in Figure 4. This grammar describes the three idealized actions stirring, mashing and pouring in case they were always executed this way. The action that will be recognized depends on the path through the graph. The here presented tasks take place in the kitchen domain. They comprise taking kitchen utensils from a table, working with them and putting them back to their places. If a cyclic action unit like stirring or grating is involved, this action unit can be individually repeated. The action sequences and action units had been defined beforehand.

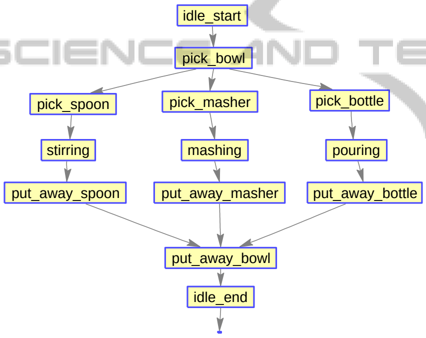


Figure 4: Sample grammar for three tasks: stirring, mashing and pouring.

5 ACTION RECOGNITION SYSTEM

The action recognition system is made up of two components. First, a low level modeling is done on the level of action units using HMMs. Second, for the recognition of action sequences, the low level HMMs are combined with a stochastic context free grammar, which controls the longer sequences of action units and also allows to solve disambiguaties at the level of action units. During the recognition of the sequences, an implicit automatic segmentation of the action sequences into action units is performed.

Our action recognition system features the one pass IBIS decoder (Soltau et al., 2001), which is part of the Janus Recognition Toolkit JRTk (Finke et al., 1997). We use this toolkit to recognize actions based on Hidden Markov Models (HMMs).

Each action unit is statistically modeled with a 4-state left-to-right HMM. Each state of the left-to-right HMM has two equally likely transitions, one to the current state, and one to the next state. The emission probabilities of the HMM states are modeled by Gaussian mixtures. The number of Gaussians per mixture is taken from cross-validation experiments. An action sequence is modeled as a sequential concatenation of action unit models.

Initialization and Training. To initialize the HMM models of the action units, we manually segmented the data into the action units. As action units are modeled by a 4-state HMM, the manually segmented data are equally divided into four sections, and a Neural Gas algorithm (Martinetz and Schulten, 1991) is applied to initialize the corresponding HMM-state and its emission probabilities. HMM training was performed featuring the Viterbi EM algorithm based on forced alignment on the unsegmented action sequences.

On-line Recognition. The on-line decoding of the systems is carried out as a time-synchronous beam search. Large beams are applied to avoid pruning errors, using a context free grammar to guide the recognition process. The context free grammar consists of 10 start symbols, one for each sequence type, leading to a sequence of terminals representing the sequence of actions units of the specific activity. The idle positions are optional and the number of repetitions for the cyclic action units is arbitrary, but at least one.

6 EVALUATION

We evaluate the recognition performance of the proposed system by applying it to two different datasets. The first dataset is the Basic Kitchen Tasks dataset [1] consisting of 10 action sequences with a total of 48 action units. Each action sequence has been recorded 20-30 times resulting in an overall of 250 action sequences samples and over 6000 action unit samples. The video data is captured with 30fps and a resolution of 640x480 px with a Prosilica GE680C camera.

[1] http://www.sfb588.uni-karlsruhe.de/bkt-dataset/

Table 1: Comparison of optical flow (HoOF) and feature flow (HoFF) on the Basic Kitchen Tasks (BKT) dataset(10 sequences / 48 action units) and the ADL dataset(10 sequences / 53 action units).

BKT dataset I	HoOF	HoFF
Sequence recog.	100.0 %	100.0 %
Unit recog.	96.7 %	96.6 %

ADL dataset	HoOF	HoFF
Sequence recog.	82.0 %	71.3 %
Unit recog.	63.5 %	55.0 %

Parallel to the video data acquisition, five of the performed action sequences are recorded with a marker based motion capture system (Vicon). Each sequence has been repeated 20 times. Overall 100 samples with over 2400 action units were recorded. Reflective markers were attached to the test persons upper body and mapped onto a kinematic model to calculated the related joint angle trajectories of the test persons motions. The system outputs a feature vector of the 24 joint angles, describing the actual pose of an upper body model. For the recognition deltas of joint angles are calculated as the input vectors. The second dataset is the University of Rochester Activities of Daily Living dataset [2]. This set also comprises 10 different tasks, which have been manually segmented using a total of 53 action units. The input feature vectors of all systems are normalized by mean subtraction and by normalizing the standard deviation to 1.

6.1 Feature Flow Recognition

To compare the recognition performance of the feature based approach with the optical flow based approach, we compute the histograms of oriented feature flow (HoFF) as well as the histograms of oriented optical flow (HoOF) for all video sequences. Both histograms consist of 30 bins corresponding to a 30 dimensional input vector for the HMMs.

For HMM action unit model training and evaluation we use a 10-fold (Basic Kitchen Tasks dataset I) / 3-fold (ADL dataset) cross-validation over all action sequences. To initialize the HMMs, hand-segmented action units of the training data are used. For training we use the training data without segmentation information. The test set is also used without any segmentation information. The given recognition results refer to the mean recognition rates over all test runs.

Both datasets were evaluated according to the recognition performance of optical flow and feature flow. For the first dataset, the sequence recognition rate is optimal for both approaches and the mean unit recognition rates rank at 96.7% for optical flow and at 96.6% for the feature flow based approach (Table 1). For the second dataset the overall recognition rate is 82.0% for optical flow and 71.3% for sparse feature flow (Table 1). Comparing those results to the recognition performance published for this dataset so far (Messing et al., 2009), it outperforms motion-based approaches without local information, which is what we need to allow flexible settings and environments.

Table 2: Comparison with marker based system for 5 sequences.

	Marker based	HoOF	HoFF
Input vector dim.	24	30	30
Gaussians per state	16	16	16
States per unit	4	4	4
Sequence recog.	100.0 %	100.0 %	100.0 %
Unit recog.	98.3 %	96.9 %	97.5 %

6.2 Comparison with a Marker based System

Five of the performed actions of the Basic Kitchen Tasks dataset were simultaneously recorded with a marker based motion capture system (Vicon). To ensure comparability, the recognition differs only in the type of input vector, while all other system components are the same for both systems.

The recognition performance while using a context free grammar is for all systems optimal. The action unit recognition rate (see Table 2), describing how many actions units were correctly recognized, is best for the marker based systems, while the optical flow based system is the worst. The problems in recognition result mainly from the mistakes in counting of cyclic motions. Action units can be overlooked, because they only consist of a few frames. Regarding the good performance of the video based systems, one has to remark that the recorded setting was optimal for vision systems, with a camera standing in front of the test person, whereas the marker based system is view point independent.

6.3 Runtime Performance

The runtime is evaluated for feature flow histograms, optical flow histograms and a CUDA-based Java implementation of feature flow histograms on a 2.83GHz Intel Core2Quad processor with 8GB RAM. For the evaluation the processing time per frame for each sequence is analyzed. It can be shown that the

[2] http://www.cs.rochester.edu/ rmessing/uradl/

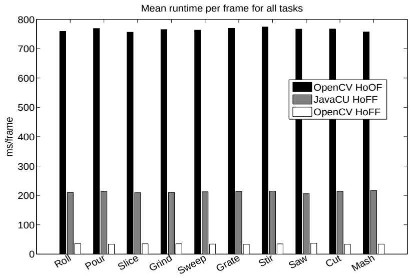


Figure 5: Runtime for optical flow compared to feature based system.

optical flow histogram calculation takes around 764 ms and the openCV based implementation of feature flow histograms needs 34 ms. It is constant for any type of sequence as can be seen in Figure 5. The runtime for the decoding ranges between 20 and 35 ms. It is done by beam search over all possible action units giving a hypothesis of the current action unit as well as the history of action units and type of sequence that has been performed. This leads to an over all processing time of the system of 25fps, which can be seen as acceptable for on-line recognition.

7 CONCLUSIONS

In this paper a system for the on-line recognition of human actions is presented. The video based action recognition techniques are qualified for the recognition of sequences of action units and complex activities. The combination of feature flow histograms and HMMs enables an on-line action recognition system to recognize human activities during their execution in a natural, unrestricted scenario. We see this as a valuable step towards an on-line action recognition that allows to adapt to the user and its needs while still being robust and scalable enough to work in a real live environment.

ACKNOWLEDGEMENTS

We thank the Insitute for Sports and Sport Science, Karlsruhe Institute of Technology (KIT), Germany for recording the marker data used in this work. This work was partially supported by the German Research Foundation (DFG) within the Collaborative Research Center SFB 588 on Humanoid Robots - Learning and Cooperating Multimodal Robots and by OSEO, French State agency for innovation, as part of the Quaero Programme.

REFERENCES

Danafar, S. and Gheissari, N. (2007). Action recognition for surveillance applications using optic flow and svm. In *ACCV*, volume 2, pages 457–466.

Efros, A. A., Berg, A. C., Mori, G., and Malik, J. (2003). Recognizing action at a distance. In *IEEE International Conference on Computer Vision*, pages 726–733, Nice, France.

Finke, M., Geutner, P., Hild, H., Kemp, T., Ries, K., and Westphal, M. (1997). The karlsruhe-verbmobil speech recognition engine. *ICASSP-97.*, 1:83–86.

Gehrig, D., Khne, H., Wrner, A., and Schultz, T. (2009). Hmm-based human motion recognition with optical flow data. In *9th IEEE-RAS International Conference on Humanoid Robots, Humanoids 2009*, Paris, France.

Ivanov, Y. A. and Bobick, A. F. (2000). Recognition of visual activities and interactions by stochastic parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:852–872.

Koehler, H. and Woerner, A. (2008). Motion-based feature tracking for articulated motion analysis. In *Workshop on Multimodal Interactions Analysis of Users a Controlled Environment, IEEE Int. Conf. on Multimodal Interfaces (ICMI 2008)*, Chania, Greece.

Lucas, B. D. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision.

Lucena, M. J., de la Blanca, N. P., Fuertes, J. M., and Marín-Jiménez, M. J. (2009). Human action recognition using optical flow accumulated local histograms. In *Iberian Conf. on Pattern Recognition and Image Analysis, IbPRIA*, pages 32–39.

Marszalek, M., Laptev, I., and Schmid, C. (2009). Actions in context. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:2929–2936.

Martinetz, T. and Schulten, K. (1991). A ”neural-gas” network learns topologies. *Artificial Neural Networks*, 1:397–402.

Mendoza, M. A., Pérez De La Blanca, N., and Marín-Jiménez, M. J. (2009). Fitting product of hmm to human motions. In *Proc. of the 13th Int. Conf. on Computer Analysis of Images and Patterns, CAIP*, pages 824–831, Berlin, Heidelberg. Springer-Verlag.

Messing, R., Pal, C., and Kautz, H. (2009). Activity recognition using the velocity histories of tracked keypoints. In *ICCV*, Washington, DC, USA. IEEE Computer Society.

Shi, J. and Tomasi, C. (1994). Good features to track. *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 593–600.

Soltau, H., Metze, F., Fügen, C., and Waibel, A. (2001). A one-pass decoder based on polymorphic linguistic context assignment. *ASRU*, pages 214–217.

Tomasi, C. and Kanade, T. (1991). Detection and tracking of point features. Technical report, International Journal of Computer Vision.