Action Recognition in a Wearable Assistance System

Marc Hanheide, Nils Hofemann, and Gerhard Sagerer
Bielefeld University, Faculty of Technology,
P.O. Box 100131, 33501 Bielefeld, Germany
Email: {mhanheid,nhofeman,sagerer}@techfak.uni-bielefeld.de

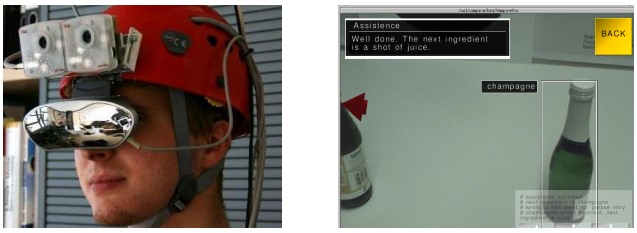
Abstract

Enabling artificial systems to recognize human actions is a requisite to develop intelligent assistance systems that are able to instruct and supervise users in accomplishing tasks. In order to enable an assistance system to be wearable, head-mounted cameras allow to perceive a scene visually from a user's perspective. But realizing action recognition without any static sensors causes special challenges. The movement of the camera is directly related to the user's head motion and not controlled by the system. In this paper we present how a trajectory-based action recognition can be combined with object recognition, visual tracking, and a background motion compensation to be applicable in such a wearable assistance system. The suitability of our approach is proved by user studies in an object manipulation scenario.

1. Introduction

Manipulations of objects and their utilization to accomplish certain tasks play a major role in everyday human life. Thus, computer systems that are able to classify and interpret manipulative gestures are subject to research for quite a long time and play a major role in different scientific fields like human-machine interaction, interactive learning, and intelligent assistance, to mention only a few. Especially the latter – research on artificial systems that can assist in object manipulation tasks – potentiate a great variety of applications. A system that is able to supervise a user in performing a task can help to detect failures and assist in complex constructions in terms of context-dependent instructions.

The recognition of object manipulations or other human activities from visual cues is a well studied subject in recent research. Most approaches found in literature analyze the trajectory of the manipulating hand to classify activities. The trajectory is acquired by segmenting and tracking skin color regions or special markers attached to the hand. Most approaches for visual trajectory-based classification either use a static camera for, e.g., surveillance [10] or interaction [9], or have direct control of their visual sensors [8]. In previous works we already presented how hand trajectories captured by a static camera can be combined with contextual information about the manipulated objects to allow a robust classification of human's actions with a probabilistic framework [4].



(a) AR gear with cameras. (b) View of the user.

Figure 1. The wearable assistance system.

In this paper we present an extension and improvement of this approach and its integration in a *wearable* assistance system. As a challenge in this particular use case of a wearable system, activities are recognized based solely on video input captured from the user's perspective using a head-mounted camera without any additional sensory. Thus, the system has no control of the camera's motion itself. Accordingly, we introduce a novel combination of visual tracking with a background motion compensation and a probabilistic classification framework for the recognition of actions in a wearable assistance system.

The paper is structured as follows: In section 2 our approach for action recognition on a wearable platform is presented and involved algorithms are outlined. After presenting some results on visual tracking accuracy and system performance in section 3 the paper concludes with a summary and an outlook.

2. System Architecture and Implementation

The action recognition presented in this paper is developed as part of an integrated cognitive assistance system dealing with one-handed object manipulations [15]. This

0-7695-2521-0/06/$20.00 (c) 2006 IEEE

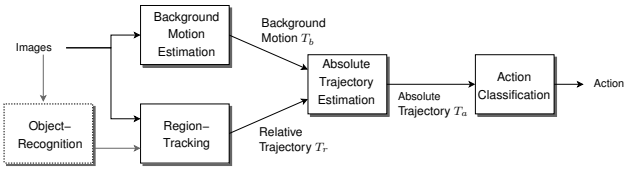


Figure 2. Architectural sketch of the system.

system utilizes an augmented reality setup (AR gear) as depicted in Fig. 1(a) which comprises all (vision) sensors to perceive the scene from a user's perspective. The two helmet-mounted cameras capture images at a rate of $15Hz$. By redisplaying the captured images in the display of the user, a video see-through loop is realized (Fig. 1(b)). The image stream of the left camera is used as monocular visual input for the action recognition presented in the paper.

Enabling action recognition on a wearable device implies special requirements. In the proposed system manipulation of objects are classified based on the contextual information about the object's type and its trajectory when being manipulated. It is assumed that due to the absence of any static cameras, the user has to keep the manipulated object in focus of his field of view. This appears to be a reasonable constraint since humans usually follow their hand to verify the manipulation consecutively especially when carrying out more complex tasks.

To realize action recognition based on these assumptions, the conceptual system architecture shown in Fig. 2 has been designed. Objects focused by the user are recognized by a module for view-based object segmentation and recognition [6] (displayed shaded in Fig. 2), which however is not subject to this paper in detail. Its functionality is to provide a label and the bounding box of the object which is used to initialize the visual *Region Tracking* module. As the scene is perceived via the head-mounted camera only the trajectory acquired by this visual tracking component computes the *relative* motion T_r of the object with respect to the user's view. To obtain the *absolute* trajectory T_a it is necessary to estimate the user's or camera's motion T_b, respectively. The responsible module for *Background Motion Estimation* is detailed in section 2.2. The module *Absolute Trajectory Estimation* computes the absolute trajectory T_a by compensating T_b.

2.1 Visual Tracking

Action recognition in the presented approach is based on trajectories of manipulated objects. To obtain this trajectories visual tracking techniques are applied. Several different approaches can be found in literature, each having different benefits and drawbacks [3]. For the task of online action recognition, two different algorithm have been integrated

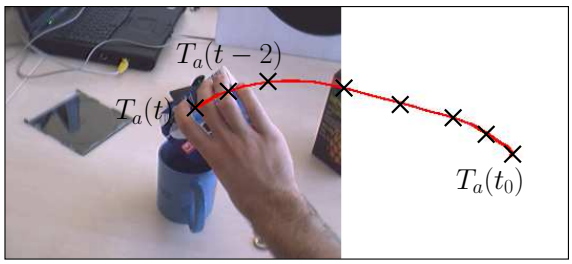


Figure 3. Frame $\mathcal{I}_t$ with absolute trajectory.

and evaluated that are able to perform in (soft) real-time.

Hyperplane Tracking The first realized tracking algorithm utilizing the approach of *hyperplanes* [5] (a template-based technique) allows tracking according to different motion models. It can be designed to either track only translational object motions as well as more complex motion model like affine or projective transformations. Basically, this approach assumes the change of intensities I of a set of pixels in the tracked region $\mathcal{R}$ to be explained by a transformation F at a given time t. In the training phase of the algorithm a linear Transformation A_h which is independent of t is estimated. This can be used to compute the motion parameters

$$\boldsymbol{\delta T_r} \quad = \quad \boldsymbol{A_h}\left(\boldsymbol{I}(\boldsymbol{F}(\boldsymbol{\mathcal{R}}, \boldsymbol{T_r}), \boldsymbol{t}) - \boldsymbol{I}(\boldsymbol{F}(\boldsymbol{\mathcal{R}}, \boldsymbol{T_r^*}), \boldsymbol{t})\right)$$

from the difference of pixel intensities. Tracking based on this approach is fast and allows to track also rotating objects, but requires the tracked object to be textured and is quite sensitive to occlusion.

Kernel-based Tracking To overcome the limited robustness of the Hyperplane Tracker, another approach which is much more robust to occlusion and requires no time consuming training phase has been integrated. The kernel-based tracking proposed by Comaniciu et al. [2] uses color histograms as features to compute the similarity between the reference model and a candidate model. The mean shift algorithm is applied for an iterative minimization of this distance by optimizing the motion parameters. The approach is admittedly restricted to translational motion models, but provides a very good tradeoff of high accuracy and robustness.

2.2 Background Motion Model

As mentioned before, computing the absolute trajectory requires the estimation of the user's motion to subtract it from the relative trajectory obtained by the visual tracking module. Various approaches for the estimation of three-dimensional egomotion from image sequences are known in the literature [13, 14]. We utilize an approach to track the background according to an affine motion model of the global image based on tracked patches [12, 16]. For each

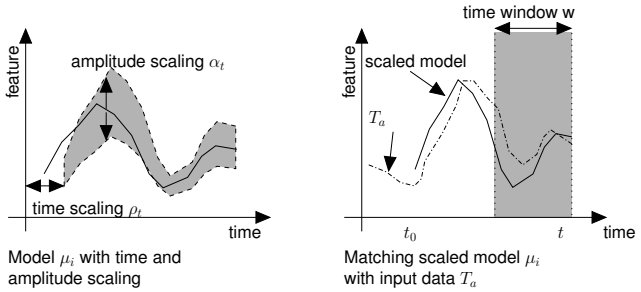


Figure 4. Scaling and matching of a model μ

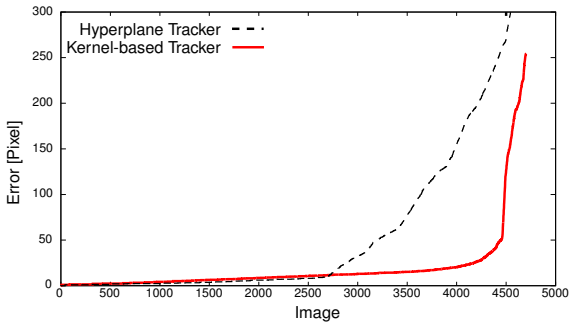


Figure 5. Absolute tracking errors.

of these patches the affine motion estimation is computed and *Least Median of Squares (LMedS)* regression [11] is applied to estimate the global background transformation $\boldsymbol{T_b}$. LMedS copes well with outliers and thus allows to compute the global image movement neglecting local deviations caused by the performed action itself.

Figure 3 displays the estimated absolute trajectory of a "pouring" action. Note that the complete trajectory is not visible in the current frame $\boldsymbol{\mathcal{I}_t}$, but has been estimated using the background motion model.

2.3. Activity Recognition

For the recognition of actions the fast and robust Condensation-based Trajectory Recognition method (CTR) described by Black and Jepson [1] is applied. This approach is an extension of the Condensation algorithm by Isard and Blake [7]. For details of the algorithm used and manipulations in context of objects refer to [4].

This Particle Filtering algorithm compares several models $\boldsymbol{\mu_i}$ – each describing an action – with the observed feature vector $\boldsymbol{T_a(t)}$ using a sample set of size $\boldsymbol{N}$. In this notion, $\boldsymbol{\mu_i(\phi)}$ denotes the model feature vector at time step $\boldsymbol{\phi = t - t_0}$. A model is propagated starting at $\boldsymbol{t_0}$ and evolves over time. To cope with variance in the execution of the action the model $\boldsymbol{\mu_i}$ is scaled in time $\boldsymbol{\alpha_i}$ and amplitude $\boldsymbol{\rho_i}$ as shown in Fig. 4(a). The quality of the match of such a scaled model – the sample $\boldsymbol{n}$ with its parameter vector $\mathbf{s}^{\boldsymbol{n}}$ – and the measurement $\boldsymbol{T_a(t)}$ is expressed in a weight $\boldsymbol{\pi_t^{(n)}}$ (see Fig. 4(b)). The temporal characteristics of an action is included by using the data in a time window $\boldsymbol{w}$ of several previous steps.

Recognition of actions is performed by calculating the end probability $\boldsymbol{p_{end}(\mu_i)}$ for each model by summing up the weights $\boldsymbol{\pi_t^{(n)}}$ of all samples with a matching position in the last 10% of the trajectory length. A model is recognized when the threshold for the end probability $\boldsymbol{p_{end}(\mu_i)}$ for this model is reached. Hereby, an implicit rejection is given as only complete actions are detected.

The characteristics of actions manipulating objects is the movement of the object as well as its orientation. Depending on the features offered by the tracking algorithm the CTR uses the velocity $\boldsymbol{\Delta x}$ and $\boldsymbol{\Delta y}$ of the object or its rotation $\boldsymbol{\Delta\gamma}$ additionally. The stochastic approach allows robust and fast recognition even on noisy input data.

3. Evaluation

Evaluation of the system follows a two-face strategy. On the one hand, the different tracking approaches have been evaluated in terms of accuracy and robustness. On the other hand, we conducted user studies in a natural scenario.

Absolute Tracking In order to evaluate the different tracking modules we created synthetic image sequences containing several patches of different objects that are moved around on a background image at different speed in a controlled manner. Additionally, the background image is shifted randomly to simulate the user's movement. By means of this, ground-truth data is available to evaluate the absolute trajectory estimation separately. Fig. 5 presents results of this quantitative evaluation study. Tracking errors are measured in euclidian distance in pixel deviation for each image (640x480 pixel) of the sequences and displayed in ascending order in the diagram. It can be seen that both trackers are quite accurate in most cases, but the hyperplane tracker appears to be much less robust. As expected, for allowing a more complex motion model (including rotation and scaling) the price of reduced robustness has to be paid.

Bartender Assistant The recognition performance of the classification system is evaluated as part of the integrated system – the bartender assistant, described in more detail in [15]. The system's task is to instruct a user wearing the AR gear to mix beverages. The intelligent assistant supervises the user and inform him or her in case of errors. It thus has to classify the actions the user is performing. For evaluation purposes, three different typical actions have been chosen: "pouring", "moving", and "shaking". For the evaluation of the action classification sub-system all errors caused by other modules like, for instance faulty object recognition results have been neglected. The results for the action recognition using the kernel-based tracker is displayed in Table 1, those for the hyperplane tracker in Table 2. For each action the total number of actions 'n',

action	n	c	s	d	i	ER
pour left	9	9	-	-	-	0.0
pour right	8	7	-	1	1	25.0
move left	8	5	3	-	-	37.5
move right	6	3	1	2	-	50.0
shake	9	8	-	1	-	11.1
sum	40	32	4	4	1	
%		80.0	10.0	10.0	2.5	25.0

Table 1. Results using kernel-based tracker

action	n	c	s	d	i	ER
pour left	9	9	-	-	1	11.1
pour right	8	7	-	1	-	12.5
move left	8	7	-	1	-	12.5
move right	6	4	2	-	1	50.0
shake	-	-	-	-	-	-
sum	31	26	2	2	2	
%		83.8	6.4	6.4	6.4	19.1

Table 2. Results using hyperplane tracker

the correct recognized ones 'c', and substitutions 's', deletions 'd', and insertions 'i' are listed. Based on these values the Error Rate $\boldsymbol{ER} = \frac{(s+d+i)}{n}$ is calculated. Due to the high velocity of the object and occlusion by the user's hand during the "shake" action, a robust tracking with the hyperplane tracker could not be achieved (see Table 2). Using the kernel-based tracker causes problems in detecting "move" actions and in distinguishing these from "pour" actions. Here, the rotation as additional feature reduces the number of deletions and especially the number of substitutions, e.g, when a "pour left" action was detected instead of "move left". The training for the models was done on one action per model performed by one user. The good recognition results applying only this small training set show the good generalization ability of the system.

4. Conclusion and Outlook

We presented an unique approach enabling a wearable assistance system to recognize human actions from a head-mounted camera. The evaluation proved the proposed combination of local visual object tracking and global camera motion compensation to be appropriate for the classification of actions using an enhanced condensation algorithm. Both object tracking approaches yielded good recognition performance, but have different applicability constraints. Hyperplane tracking allows tracking according to more complex motion models, which on the one hand allowed better discrimination of actions being quite similar in the translational motion. On the other hand, the algorithm was inapplicable for really fast motion and a high degree of occlusion occurring for instance in "shake" actions. Thus, a future extension of our system will be to combine both tracking algorithms to stabilize the translational tracking, but also to provide information about object rotations. The proposed approach provides an avenue towards wearable assistance system that help users in accomplishing everyday task.

References

[1] M. J. Black and A. D. Jepson. A probabilistic framework for matching temporal trajectories: Condensation-based recognition of gestures and expressions. In *Proc. European Conf. on Computer Vision*, volume 1, pages 909–924, 1998.

[2] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *Pattern Analysis and Machine Intelligence*, 25:564–575, 2003.

[3] B. Deutsch, C. Graessl, F. Bajramovic, and J. Denzler. A comparative evaluation of template and histogram based 2-d tracking algorithms. In *Proc. Pattern Recognition Symposium (DAGM)*, Heidelberg, 2005. Springer.

[4] J. Fritsch, N. Hofemann, and G. Sagerer. Combining sensory and symbolic data for manipulative gesture recognition. In *Proc. Int. Conf. on Pattern Recognition*, number 3, pages 930–933, Cambridge, United Kingdom, 2004. IEEE.

[5] C. Gräßl, T. Zinßer, and H. Niemann. Efficient Hyperplane Tracking by Intelligent Region Selection. In *Proc. IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 51–55, 2004.

[6] G. Heidemann, H. Bekel, I. Bax, and H. Ritter. Interactive online learning. *Pattern Recognition and Image Analysis*, 15(1):55–58, 2005.

[7] M. Isard and A. Blake. A mixed-state condensation tracker with automatic model-switching. In *Proc. of 6th Int. Conf. Computer Vision*, pages 107 – 112, 1998.

[8] Y. Nagai. Learning to comprehend deictic gestures in robots and human infants. In *Proc. of the 2005 IEEE Int. Workshop on Robot and Human Interactive Communication*, pages 217–222, 2005.

[9] V. Pavlovic, R. Sharma, and T. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *PAMI*, 19:677–695, 1997.

[10] P. Ribeiro and J. Santos-Victor. Human activities recognition from video: modeling, feature selection and classification architecture. In *Proc. Workshop on Human Activity Recognition and Modelling*, pages 61–70, Oxford, Sept. 2005.

[11] P. Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79:871–880, 1984.

[12] J. Shi and C. Tomasi. Good features to track. In *Proc. Comuter Vision and Pattern Recognition*, 1994.

[13] T. Tian, C. Tomasi, and D. Heeger. Comparison of approaches to egomotion computation. In *Proc. Computer Vision and Pattern Recognition*, 1996.

[14] F. Woelk and R. Koch. Robust monocular detection of independent motion by a moving observer. In *Proc. Int. Workshop on Complex Motion*, 2004.

[15] S. Wrede, M. Hanheide, S. Wachsmuth, and G. Sagerer. Integration and coordination in a cognitive vision system. In *Proc. Int. Conf. on Vision Systems*. IEEE, 2006.

[16] T. Zinßer, C. Gräßl, and H. Niemann. Efficient feature tracking for long video sequences. In *Pattern Recognition, 26th DAGM Symposium*, pages 326–333. Springer, 2004.