

# 3D Action Recognition in an Industrial Environment

Markus Hahn<sup>1</sup>, Lars Krüger<sup>1</sup>, Christian Wöhler<sup>1,2</sup>, and Franz Kummert<sup>2</sup>

**Abstract** In this study we introduce a method for 3D trajectory based recognition of and discrimination between different working actions. The 3D pose of the human hand-forearm limb is tracked over time with a two-hypothesis tracking framework based on the Shape Flow algorithm. A sequence of working actions is recognised with a particle filter based non-stationary Hidden Markov Model framework, relying on the spatial context and a classification of the observed 3D trajectories using the Levenshtein Distance on Trajectories as a measure for the similarity between the observed trajectories and a set of reference trajectories. An experimental evaluation is performed on 20 real-world test sequences acquired from different viewpoints in an industrial working environment. The action-specific recognition rates of our system correspond to more than 90%. The actions are recognised with a delay of typically some tenths of a second. Our system is able to detect disturbances, i.e. interruptions of the sequence of working actions, by entering a safety mode, and it returns to the regular mode as soon as the working actions continue.

## 1 Introduction

Today, industrial production processes in car manufacturing worldwide are characterised by either fully automated production sequences carried out solely by industrial robots or fully manual assembly steps where only humans work together on the same task. Up to now, close collaboration between humans and machines, especially industrial robots, is very limited and usually not possible due to safety concerns. Industrial production processes can increase efficiency by establishing a close collaboration of humans and machines exploiting their unique capabilities. A

---

<sup>1</sup>Daimler AG, Group Research and Advanced Engineering  
P. O. Box 2360, D-89013 Ulm, Germany

<sup>2</sup>Applied Informatics, Faculty of Technology, Bielefeld University  
Universitätsstraße 25, D-33615 Bielefeld, Germany

safe interaction between humans and industrial robots requires vision methods for 3D pose estimation, tracking, and recognition of the motion of both human body parts and robot parts.

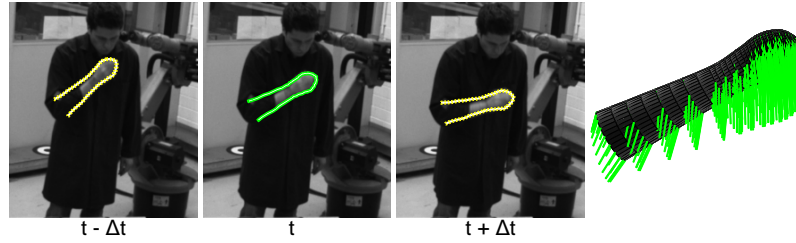
Previous work in the field of human motion capture and recognition is extensive. Moeslund et al. [12] give a detailed introduction and overview. Bobick and Davis [2] provide another good introduction. They classify human motion using a temporal template representation from a set of consecutive background-subtracted images. A drawback of this approach is its dependence on the viewpoint.

Li et al. [11] use Hidden Markov Models (HMMs) to classify hand trajectories of manipulative actions and take into account the object context. In [3] the motion of head and hand features is used to recognise Tai Chi gestures by HMMs. Head and hand are tracked with a realtime stereo blob tracking algorithm. HMMs are used in many other gesture recognition systems due to their ability to probabilistically represent the variations of the training data. Dynamic Bayesian Networks (DBN) generalise HMMs and are able to consider several random variables [13]. In [14] two-person interactions are recognised with a DBN. The system has three abstraction levels. On the first level, human body parts are detected using a Bayesian network. On the second level, DBNs are used to model the actions of a single person. On the highest level, the results from the second level are used to identify the interactions between individuals.

A well known approach to gesture recognition is the method by Black and Jepson [1], who present an extension of the CONDENSATION algorithm and model gestures as temporal trajectories of the velocity of the tracked hands. They perform a fixed size linear template matching weighted by the observation densities. Fritsch et al. [5] extend their work by incorporation of situational and spatial context. Both approaches merely rely on 2D data. Hofemann [8] extends the work in [5] to 3D data by using a 3D body tracking system. The features used for recognition are the radial and vertical velocities of the hand with respect to the torso.

Croitoru et al. [4] present a non-iterative 3D trajectory matching framework which is invariant to translation, rotation, and scale. They introduce a pose normalisation approach which is based on physical principles, incorporating spatial and temporal aspects of trajectory data. They apply their system to 3D trajectories for which the beginning and the end is known. This is a drawback for applications processing a continuous data stream, since the beginning and end of an action are often not known in advance.

This paper addresses the problem of tracking and recognising the motion of human body parts in a working environment, which is a precondition for a close collaboration between human workers and industrial robots. Our 3D tracking and recognition system consists of three main components: the camera system, the model-based 3D tracking system, and the trajectory-based recognition system. As an imaging system we use a calibrated small-baseline trinocular camera sensor similar to that of the SafetyEYE protection system ([www.safetyeye.com](http://www.safetyeye.com)) which is used in our production processes to protect human workers.



**Fig. 1** Example of a spatio-temporal 3D pose estimation result.

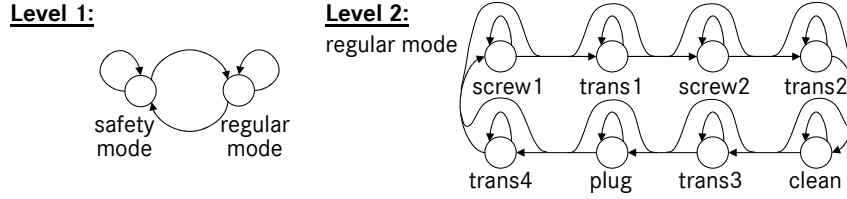
## 2 The 3D Tracking System

We rely on the spatio-temporal 3D pose estimation and tracking system introduced in [7], which is based on the Shape Flow algorithm. A spatio-temporal 3D model of the human hand-forearm limb is used as shown in Fig. 1 (right), made up by a kinematic chain connecting the two rigid elements forearm and hand. The model consists of five truncated cones and one complete cone. The Shape Flow algorithm fits the parametric curve to multiple calibrated images by separating the grey value statistics on both sides of the projected curve. Fig. 1 illustrates a correspondingly obtained spatio-temporal 3D pose estimation result. At time step  $t$  the projected contour of the estimated 3D pose is shown as a green curve, while the images at time steps  $t \pm \Delta t$  depict the projected contours inferred from the temporal pose derivative as yellow curves. To start tracking, a coarse initialisation of the model parameters at the first time step is required. In the tracking system we apply two instances of the Shape Flow algorithm. A winner-takes-all component selects the best-fitting spatio-temporal model at each time step using different criteria. A more detailed description is given in [7].

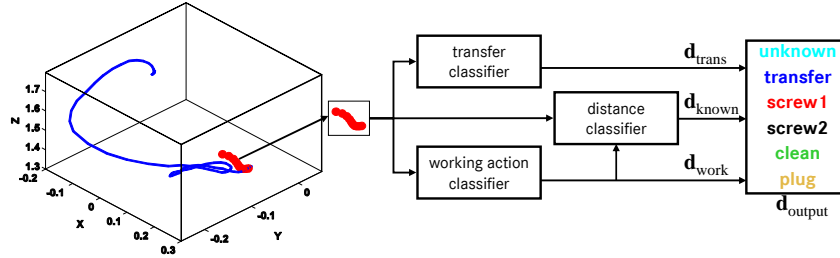
## 3 Recognition System

The working action recognition system is based on a 3D trajectory classification and matching approach. The tracking stage yields a continuous data stream of the 3D pose of the tracked hand-forearm limb. Our trajectories are given by the 3D motion of the wrist point. The cyclic sequence of working actions in an engine assembly scenario is known to our system. However, it may be interrupted by “unknown” motion patterns. The beginning and the end of a trajectory are not known a priori, which is different from the setting regarded in [4]. To allow an online action recognition in the continuous data stream, we apply a sliding window approach which enables the system to perform a recognition of and discrimination between human motion patterns.

Due to the fact that our system is designed for safe human–robot interaction, we implemented a recognition stage with two levels (Fig. 2). At the first level, the decision is made whether the human worker performs a known working action (regular



**Fig. 2** The two-level architecture of the safety system.



**Fig. 3** Classifiers and smoothed 3D input trajectory. The values in the current sliding window are marked in red, all others in blue.

mode) or an unknown motion (safety mode) based on a set of trajectory classifiers (cf. Section 3.1). In the safety mode (level 1), the system may prepare to slow down or halt the industrial robot. The regular mode (level 2) defines the cyclic working process performed by the human worker. It is implemented as a HMM in which the state is continuously estimated by a particle filter, where the particle weights are computed with the trajectory classifiers described in Section 3.1 and a trajectory matching approach based on the Levenshtein Distance on Trajectories (LDT) measure [6].

### 3.1 Trajectory Classifiers

The state of system level 1 according to Fig. 2 is determined by a set of classifiers based on features extracted from the trajectory data in the sliding window. In a pre-processing step, noise and outliers in the trajectory data are reduced by applying a Kalman Filter based smoothing procedure [4]. The kinematic model of the Kalman Filter is a constant-velocity model. Fig. 3 (left) depicts the smoothed 3D trajectory (blue) and the 3D points (red) in the current sliding window of size  $W = 8$  time steps. We apply two second-order polynomial classifiers and a Mahalanobis distance classifier [15] in a hierarchical manner (cf. Fig. 3). We found experimentally that a polynomial classifier is favourable for classifying transfer motion patterns and working actions based on small training sets (cf. Section 4), while distance-based classifiers turned out to be too restrictive for these tasks when applied to our large test set. The Mahalanobis distance classifier was used for post-processing the out-

puts of the polynomial classifiers. At time step  $t$  the current input trajectory  $\mathbf{Z}_t$  with

$$\mathbf{Z}_t = [(X_{(t-W+1)}, Y_{(t-W+1)}, Z_{(t-W+1)})^T, \dots, (X_t, Y_t, Z_t)^T], \quad (1)$$

consisting of the last  $W$  elements of the continuous data stream, is used to extract features to which the set of classifiers is applied as described in the following.

Motion patterns occurring between two working actions (here denoted by “transfer motion”) are recognised by a polynomial classifier using two features, (i) the travelled distance along the trajectory and (ii) the maximum angle between two consecutive motion direction vectors in the sliding window. The output discriminant vector  $\mathbf{d}_{\text{trans}}$  of this classifier consists of the two classes “transfer motion” and “no transfer motion”. Normalisation yields the transfer discriminant vector  $\tilde{\mathbf{d}}_{\text{trans}}$  of unit length with components in the interval  $[0, 1]$ .

Since it is known where the worker has to tighten a screw or to fit a plug, the second polynomial classifier is used for recognising working actions by incorporating spatial context for the actions “tighten screw 1”, “tighten screw 2”, “clean” and “plug”. These 3D positions are constant across the sequence and are obtained based on the known 3D pose of the engine. As features, the polynomial classifier for working actions uses the minimum distance in the sliding window to the 3D position of (i) screw 1, (ii) screw 2, (iii) the area to be cleaned, and (iv) the position where to fit the plug. Normalisation yields the working action discriminant vector  $\tilde{\mathbf{d}}_{\text{work}}$ .

The Mahalanobis distance classifier is applied to the result of the polynomial classifier for working actions and decides whether the recognised working action is a known one, since such motion patterns can only occur close to the 3D object associated with that action. The classifier applies a winner-takes-all approach to the discriminant vector  $\tilde{\mathbf{d}}_{\text{work}}$  and performs a comparison between the training data and the measured distance to the 3D object associated with the winner class. The result is the normalised discriminant vector  $\tilde{\mathbf{d}}_{\text{known}}$ .

Based on the normalised discriminant vectors of the three classifiers, which are given by

$$\tilde{\mathbf{d}}_{\text{trans}} = \begin{pmatrix} \tilde{d}_{\text{trans}} \\ 1 - \tilde{d}_{\text{trans}} \end{pmatrix}, \quad \tilde{\mathbf{d}}_{\text{known}} = \begin{pmatrix} \tilde{d}_{\text{known}} \\ 1 - \tilde{d}_{\text{known}} \end{pmatrix}, \quad \tilde{\mathbf{d}}_{\text{work}} = \begin{pmatrix} \tilde{d}_{\text{screw1}} \\ \tilde{d}_{\text{screw2}} \\ \tilde{d}_{\text{clean}} \\ \tilde{d}_{\text{plug}} \end{pmatrix}, \quad (2)$$

an overall discriminant vector  $\mathbf{d}_{\text{output}}$  for the six classes “unknown”, “transfer”, “screw 1”, “screw 2”, “clean”, and “plug” is determined, which is given by

$$\mathbf{d}_{\text{output}} = \begin{pmatrix} d_{\text{unknown}} \\ d_{\text{trans}} \\ d_{\text{screw1}} \\ d_{\text{screw2}} \\ d_{\text{clean}} \\ d_{\text{plug}} \end{pmatrix} = \begin{pmatrix} 1 - \tilde{d}_{\text{known}} \\ \tilde{d}_{\text{known}} \cdot \tilde{d}_{\text{trans}} \\ \tilde{d}_{\text{known}} \cdot (1 - \tilde{d}_{\text{trans}}) \cdot \tilde{d}_{\text{screw1}} \\ \tilde{d}_{\text{known}} \cdot (1 - \tilde{d}_{\text{trans}}) \cdot \tilde{d}_{\text{screw2}} \\ \tilde{d}_{\text{known}} \cdot (1 - \tilde{d}_{\text{trans}}) \cdot \tilde{d}_{\text{clean}} \\ \tilde{d}_{\text{known}} \cdot (1 - \tilde{d}_{\text{trans}}) \cdot \tilde{d}_{\text{plug}} \end{pmatrix}. \quad (3)$$

Finally, normalisation of  $\mathbf{d}_{\text{output}}$  to unit length yields the classification vector  $\tilde{\mathbf{d}}_{\text{output}}$ .

### 3.2 Recognition of the sequence of working actions

The decision whether the system is in safety mode or in regular mode is made based on the first discriminant value  $\tilde{d}_{\text{unknown}}$  of the normalised output discriminant vector  $\tilde{\mathbf{d}}_{\text{output}}$  and the matching accuracy of the particle weights in system level 2, where the observed trajectories are analysed with respect to the occurrence of known working actions.

Similar to [11] we apply a particle filter based non-stationary HMM matching in order to recognise the sequence of working actions. The HMM of system level 2 (Fig. 2) is derived from the known cyclic working task, defined by a parameter set  $\lambda = (S, A, B, \Pi)$ :

- $S = \{q_1, \dots, q_n\}$ , the set of hidden states;
- $A = \{a_{ij,t} | a_{ij,t} = P(q_t = s_j | q_{t-1} = s_i)\}$ , non-stationary (time-dependent) transition probability from state  $s_i$  to  $s_j$ ;
- $B = \{b_{i,k} | b_{i,k} = P(o_t = v_k | q_t = s_i)\}$ , probability of observing the visible state  $v_k$  given the hidden state  $s_i$ ;
- $\Pi = \{\pi_i | \pi_i = P(q_1 = s_i)\}$ , initial probability of state  $s_i$ .

We assigned a set of reference trajectories to each hidden state  $\{q_1, \dots, q_n\}$  based on the associated working action. Our system relies on a small number of reference trajectories which are defined by manually labelled training sequences. To cope with different working speeds, the defined reference trajectories are scaled in the temporal domain (from  $-20\%$  to  $+20\%$  of the total trajectory length).

Similar to [11] the CONDENSATION algorithm [9] is used to estimate the state of the HMM based on temporal propagation of a set of  $N$  weighted particles:

$$\left\{ (\mathbf{s}_t^{(1)}, w_t^{(1)}), \dots, (\mathbf{s}_t^{(N)}, w_t^{(N)}) \right\} \quad \text{with} \quad \mathbf{s}_t^{(i)} = \left\{ q_t^{(i)}, \phi_t^{(i)} \right\}. \quad (4)$$

The particle  $\mathbf{s}_t^{(i)}$  contains the hidden state  $q_t^{(i)}$  and the current phase  $\phi_t^{(i)}$  in this hidden state, where the phase indicates the fraction by which the working action has been completed. The resampling step reallocates a certain fraction of the particles with regard to the predefined initial distribution  $\Pi$ . The weight  $w_t^{(i)}$  of a particle is calculated according to  $w_t^{(i)} = p(o_t | \mathbf{s}_t^{(i)}) / \sum_{j=1}^N p(o_t | \mathbf{s}_t^{(j)})$ , where  $p(o_t | \mathbf{s}_t^{(i)})$  is the observation probability  $o_t$  given the hidden state  $q_t^{(i)}$  and its phase  $\phi_t^{(i)}$ . The propagation of the weighted particles over time consists of a prediction, selection, and update step.

**Select:** Selection of  $N - M$  particles  $\mathbf{s}_{t-1}^{(i)}$  according to their respective weight  $w_{t-1}^{(i)}$  and random distribution of  $M$  new particles over all other states in the HMM.

**Predict:** The current state of each particle  $\mathbf{s}_t^{(i)}$  is predicted based on the selected particles, the HMM structure (Fig. 2), and the current phase  $\phi_t^{(i)}$ . The transition probabilities  $A$  are not stationary but depend on the current phase  $\phi_t^{(i)}$  of the particle. The phase is always restricted to the interval  $[0, 1]$ . A high phase value indicates that the reference trajectories are almost traversed and that there is an increased probability to proceed to the next state.

**Update:** In the update step, the weights of the predicted particles are computed based on the discriminant vector  $\tilde{\mathbf{d}}_{\text{output}}$  derived from the classifiers and a trajectory matching based on the LDT measure [6]. To compute the weight  $w_t^{(i)}$  of a particle  $\mathbf{s}_t^{(i)}$ , the 3D data  $\mathbf{Z}_t$  in the current sliding window are matched with the current sub-trajectory of all reference trajectories of the hidden state  $q_t^{(i)}$ . The current sub-trajectory in a hypothesis trajectory is defined by its phase  $\phi_t^{(i)}$  and length  $W$ . The weight is given by the LDT measure of the best matching reference trajectory multiplied by the discriminant value in  $\tilde{\mathbf{d}}_{\text{output}}$  associated with the corresponding action class of the hidden state  $q_t^{(i)}$ . It is possible that the same working action is performed at different positions in 3D space, e.g. tightening a screw at different locations of the engine. Hence, the trajectories are normalised w.r.t. translation, rotation, and scaling. For this purpose we apply the quaternion-based approach introduced in [10].

The set of weighted particles yields a likelihood at each time step for being in the specific working action states of the HMM (cf. Fig. 4 (bottom) for an example sequence).

## 4 Experimental Evaluation

The system is evaluated by analysing 20 trinocular real-world test sequences acquired from different viewpoints. The time interval between subsequent image triples acquired with our small-baseline trinocular camera sensor amounts to  $\Delta t = 71$  ms. These sequences contain working actions performed by eight different test persons in front of a complex cluttered working environment. Each sequence contains at least 300 image triples. The distance of the test persons to the camera system amounts to 2.2–3.3 m. Each sequence contains the working actions listed in Table 1, where all actions are performed with the right hand. All sequences were robustly and accurately tracked at all time steps with the spatio-temporal 3D tracking system described in Section 2.

Only two sequences, each comprising 400 image triples, in which the working actions are performed by two different individuals, are used for training the system. These two individuals (teachers) are well trained while all other test persons are less well trained since they were shown the actions by the teachers only once in advance. This teacher-based approach is motivated by our application scenario, in which workers are generally trained by only a few experts.

**Table 1** Recognition results on our set of 20 test sequences.

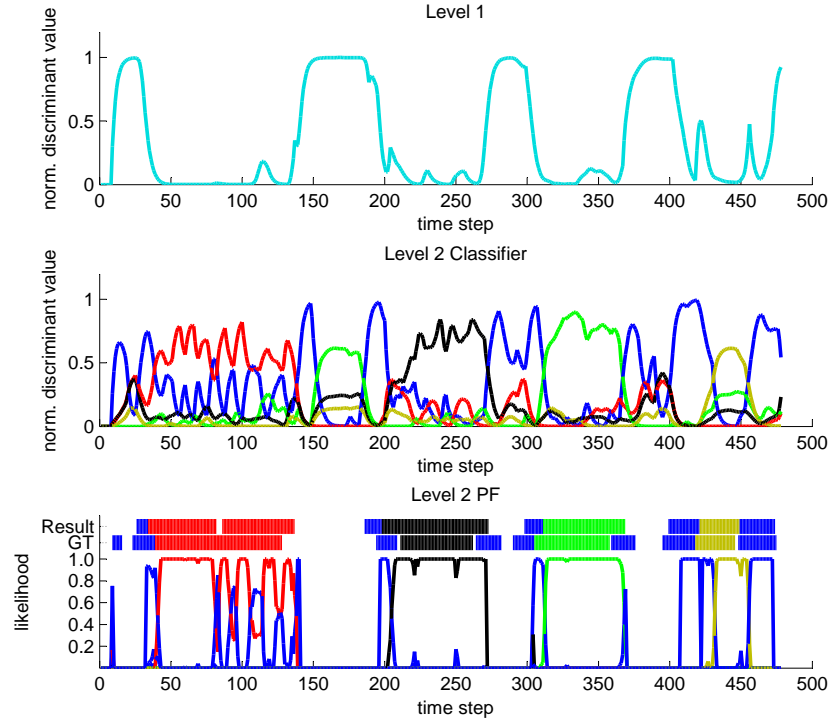
	tightening screw1	tightening screw2	cleaning	plugging
Total [#]	26	23	28	32
Correct [#]	24	21	27	30
Duplicate [#]	1	2	9	0
Deletion [#]	2	2	1	2
Substitution [#]	0	0	0	0
Insertion [#]	0	2	1	1
Recognition rate [%]	92.3	91.3	96.4	93.8
Word error rate [%]	7.7	17.4	7.1	9.4
Delay begin, mean [ms]	216	777	1102	364
Delay begin, std [ms]	902	702	1753	666
Delay end, mean [ms]	442	246	-1180	239
Delay end, std [ms]	1076	588	1500	319

We assigned ground truth labels manually to all images of the training and test sequences. All results are obtained with a total number of  $N = 500$  particles and  $M = 100$  uniformly distributed particles. The computation time of our Matlab implementation of the recognition system is around 1 frame per second on a Core 2 Duo with 2.4GHz.

Table 1 shows that the system achieves average action recognition rates of more than 90% on the test sequences. The relatively large number of duplicates for the cleaning action is due to the erroneous recognition of short transfer phases during these actions as a result of fast motion. The recognition errors can be ascribed to tracking inaccuracies and to motion patterns that differ in space and time from the trained motion patterns. Higher recognition rates may be achieved by using more training sequences, since the scaling in the temporal domain of the reference trajectories is not necessarily able to cope with the observed variations of the motion patterns. The average word error rate, which is defined as the sum of insertions, deletions, and substitutions, divided by the total number of test patterns, amounts to about 10%. Our recognition rates are similar to those reported by Croitoru et al. [4] and Fritsch et al. [5]. Segmented 3D trajectories are used in [4], which is different from our approach, while the method described in [5] relies on 2D data and is not independent of the viewpoint. A precise and stable 3D tracking is essential for our approach since the 3D positions associated with the working actions are separated from each other by only a few decimetres.

On the average, our system recognises the working actions with a delay of several tenths of a second when compared to the manually labelled ground truth, except for the cleaning action which is typically recognised by about one second earlier (negative mean delay). The standard deviations of the delays are typically comparable to or larger than their mean values. One should keep in mind, however, that our manually assigned labels are not necessarily perfectly accurate.

Beyond the recognition of working actions, our system is able to recognise disturbances, occurring e.g. when the worker interrupts the sequence of working actions by blowing his nose. The system then enters the safety mode (indicated by high values of  $\hat{d}_{\text{unknown}}$  in Fig. 4 (top)) and returns to the regular mode as soon as the working actions are continued.



**Fig. 4** Recognition of working actions for an example sequence. Top: Normalised classifier output  $\tilde{d}_{\text{unknown}}$ . Middle: Last five components of the output  $\tilde{\mathbf{d}}_{\text{output}}$  (red: screw 1; black: screw 2; green: clean; brown: plug; blue: transfer). Bottom: Final action recognition result compared to ground truth (GT).

## 5 Summary and Conclusion

In this study we have introduced a method for 3D trajectory based recognition of and discrimination between different working actions. The 3D pose of the human hand-forearm limb has been tracked over time with a two-hypothesis tracking framework based on the Shape Flow algorithm. Sequences of working actions have been recognised with a particle filter based non-stationary HMM framework, relying on the spatial context and a classification of the observed 3D trajectories using the LDT as a measure for the similarity between the observed trajectories and a set of reference trajectories. An experimental evaluation has been performed on 20 real-world test sequences acquired from different viewpoints in an industrial working environment. The action-specific recognition rates of our system correspond to more than 90%, where the actions have been recognised with a delay of typically some tenths of a second. Our system is able to detect disturbances, i.e. interruptions of the sequence of working actions, by entering a safety mode, and it returns to the regular mode as soon as the working actions continue.

An extension of the recognition system to more actions is straightforward. The HMM then needs to be extended by adding new working actions, the classifiers need to be re-trained, and the number of particles should be increased, since the required number of particles scales linearly with the number of actions. Future work may involve online learning of reference trajectories and the definition of a more complex interaction scenario with several human workers and industrial robots. In a such a scenario the usage of more complex DBNs would be appropriate.

## References

1. M. J. Black and A. D. Jepson. A probabilistic framework for matching temporal trajectories: Condensation-based recognition of gestures and expressions. In *ECCV '98: Proc. of the 5th European Conf. on Computer Vision-Volume I*, pages 909–924, 1998.
2. A. F. Bobick and J. W. Davis. The recognition of human movement using temporal templates. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(3):257–267, 2001.
3. L. W. Campbell, D. A. Becker, A. Azarbayejani, A. F. Bobick, and A. Pentland. Invariant features for 3-d gesture recognition. In *FG '96: Proc. of the 2nd Int. Conf. on Automatic Face and Gesture Recognition (FG '96)*, 1996.
4. A. Croitoru, P. Agouris, and A. Stefanidis. 3d trajectory matching by pose normalization. In *GIS '05: Proc. of the 13th annual ACM international workshop on Geographic information systems*, pages 153–162, 2005.
5. 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, 2004.
6. M. Hahn, L. Krüger, and C. Wöhler. 3d action recognition and long-term prediction of human motion. In A. Gasteratos, M. Vincze, and J. Tsotsos, editors, *Proc. Int. Conf. on Computer Vision Systems, Santorini, Greece.*, pages 23–32, 2008.
7. M. Hahn, L. Krüger, and C. Wöhler. Spatio-temporal 3d pose estimation and tracking of human body parts using the shape flow algorithm. In *Proc. Int. Conf. on Pattern Recognition, Tampa, USA*, 2008.
8. N. Hofemann. *Videobasierte Handlungserkennung für die natürliche Mensch-Maschine-Interaktion*. Dissertation, Universität Bielefeld, Technische Fakultät, 2007.
9. M. Isard and A. Blake. Condensation—conditional density propagation for visual tracking. *Int. J. Comput. Vision*, 29(1):5–28, 1998.
10. S. K. Kearsley. On the orthogonal transformation used for structural comparisons. *Acta Cryst.*, A45:208–210, 1989.
11. Z. Li, J. Fritsch, S. Wachsmuth, and G. Sagerer. An object-oriented approach using a top-down and bottom-up process for manipulative action recognition. In *DAGM06*, volume 4174 of *Lecture Notes in Computer Science*, pages 212–221, 2006.
12. T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126, 2006.
13. K. P. Murphy. *Dynamic bayesian networks: representation, inference and learning*. PhD thesis, 2002. Chair-Russell, Stuart.
14. S. Park. A hierarchical bayesian network for event recognition of human actions and interactions. *Multimedia Systems, Sp. Iss. on Video Surveillance*, 10(2):164–179, 2004.
15. J. Schürmann. *Pattern classification: a unified view of statistical and neural approaches*. John Wiley & Sons, Inc., 1996.