Segmentation and Tracking for Vision Based Human Robot Interaction

Salman Valibeik and Guang-Zhong Yang
Institute of Bio-Medical Engineering, Imperial College London
{valibeik, g.z.yang}@imperial.ac.uk

Abstract

Vision based Human Robot Interaction (HRI) in a crowded scene is a challenging research problem. The aim of this paper is to provide a reliable framework for simple gesture recognition for robotic navigation under partial occlusion and varying illumination conditions. The proposed method combines hand motion segmentation and skin colour detection for gesture recognition. Motion clustering based on Least Median Square Error (LMedS) followed by Kalman filtering and HMM gesture detection has been used. Experimental results have shown that the method can successfully restore the motion field that allows accurate, dominant affine motion detection for consistent gesture estimation.

1. Introduction

Vision based Human-Robot Interaction (HRI) has made great strides recently toward practical implementations. HRI provides major advantages by reducing navigation difficulties, allowing seamless and natural interaction between humans and robots. In a home or office environment, HRI is desirable in many applications, enabling robots to act in a way that is similar to humans and to provide assistance in a range of daily activities [1].

In order to design an effective HRI system, it is necessary to understand how Human-Human Interaction (HHI) takes place. HHI can be complex when two or more people are involved. Most of the inherent parameters involved rely on human perception, where intention and high-level cognitive processes play an essential role in determining our expressions and ways of interaction.[2].

Although a machine's ability can be enhanced to some extent by modelling the cognitive influences [3-5], there are other parameters that are difficult to predict due to the inherent ambiguities involved. Cognitive decision making does not always produce an observable reaction, thus making the task more difficult. Therefore, physical aspects of perception are widely used instead. These are the interactions that are currently used for robotic guidance and HRI. Recently, there has been much progress in the physical aspect of interaction due to advances in sensing technology. Sensory interaction is mainly divided into four categories: *auditory, visual, sensation* and *smell*. Our main focus in this paper is to convey visual information in a scene based on computer vision.

Recognising trained gestures, including partial or whole body movements, is desirable for gesture recognition. For both conditions, a silhouette needs to be extracted first. For example, Gavrila *et al* [6] detect a richer set of texture features trained by neural networks as a candidate region and Seong-Whan [7] employed depth information obtained by a stereo camera to recover the whole body silhouette. Since recovering body posture is not always possible, many researchers have relied on tracking changes in certain key parts of the human body, such as the face and hands. Face recognition is a significant area of research which is beyond the scope of this paper, but a comprehensive overview can be found in [8]. Hand gesture is of particular interest due to its dynamic nature, which can be used as an intuitive tool for HRI. However, due to pose variability and shape complexity, hand gesture recognition remains a difficult task, particularly in a crowded environment. Hanning *et al* [9] employed modified Histogram of Oriented Gradients (HOG) as a feature for each gesture in recognising Chinese signals. Zhu *et al* [10] applied recognition of continuous dynamic hand gestures by combining an appearance model with motion trajectory. Skin colour detection has been used in many applications to extract silhouette regions which mostly consist of the face and hands [11, 12]. There are two major problems with existing techniques. The first is that it assumes the availability of large size

978-0-7695-3496-1/08 $25.00 © 2008 IEEE
DOI 10.1109/WIIAT.2008.285

silhouettes, and the second relies on a relatively static and controlled environment in terms of lighting, which is not practical for real applications. In a crowded scene, however, the presence of multiple people, changing illumination and occlusion need to be carefully addressed.

The aim of this paper is to present a robust tracking method based on a combination of skin colour and motion extraction followed by a restoration scheme for hand gesture recognition suitable for HRI. It provides accurate tracking and gesture recognition results in the presence of occlusion and varying illumination. It also allows the detection of gestures in a crowded environment, thus making it suitable for general robotic navigation. In the following sections, the theoretical details of the gesture recognition scheme and its application to HRI are provided.

2. Gesture Recognition

2.1 Local temporal information extraction

In order to extract regions of interest for gesture recognition (in this case mainly skin coloured objects in the image), a skin database containing about 1 million skin colour pixels is gathered. The next stage of the process consists of fitting parametric models such as Gaussian to the training data, where the resulting model, which consists of means and deviations of a single fitted Gaussian model, is then used to classify each pixel in the image. By pruning out most of the background and extracting the connected regions, silhouettes suitable for gesture recognition are derived. To this end, a connected component labelling algorithm is employed. Connected labelling assigns different labels to connected skin-coloured regions in the image. However, merely obtaining skin coloured objects is not sufficient on its own due to the possibility of occlusion with other objects of a similar colour. In order to resolve this issue, motion information has been incorporated, where the moving skin coloured objects are tracked. In order to extract motion, the optical flow method by Black and Anandan [13] is used as a baseline implementation due to its accuracy and speed. However, the accuracy of the method generally suffers from depth discontinuity, illumination variation, occlusion and un-textured areas in the scene. In order to extract accurate motion out of a dense motion field, affine motion estimation has been conducted. In the proposed method, motion field is restored and the dominant affine motion is derived for each moving region.

2.2 LMedS motion restoration

Most affine motion segmentation techniques use least Square (LS) error measurement to obtain dominant affine representation, θ. This consists of 6 parameters $\theta = \{a_0, \cdots, a_5\}$ over $N \times N$ non-overlapping cells called seed blocks [14, 15]. The velocity, $V_{x,y}$ in the x and y direction of each point in each seed block is examined by following distortion assignment:

$$\gamma = \arg\min \sum_{x,y\in R} r^2(V_{x,y}, V_\theta) \qquad (1)$$

where γ reaches the minimum when the calculated dominant affine motion, V_θ, in each block matches with the real motion. However, this is not the case, as motions are distorted and seed blocks contain multiple motions. This is mainly due to object boundaries or inaccurate motion extraction as a result of noise and illumination conditions. Although most techniques reject the outliers by measuring the average residual between dominant affine motion and extracted motion in each seed block, it gives poor segmentation results when noise or seed blocks are located at object boundaries. In this paper, a robust Least Median Square Error, LMedS, based on the formulations proposed by Rousseeuw and Leroy [16] is used. The adapted LMedS method restores the motion distribution after affine motion segmentation so that it can be used reliably for gesture recognition. The base line LMedS implementation is as follows:

$$\hat{\theta} = \arg\min_{\tilde{\theta}_j} \left\{ \sum_{x,y\in R} median(r^2(V_{x,y}, V_{\tilde{\theta}_j}) \mid j = 1,2,\ldots,o \right\} \qquad (2)$$

where R indicates each seed block and $\tilde{\theta}_j$ is the temporary affine parameters obtained by j trials and selecting p number of points in the seed block (p in this case is 6). When o number of trials has been made, the dominant affine parameter is the minimum median, $\hat{M}$, among $M_1, M_2, \ldots, M_o$. As discussed in [16], the computation time of such a naive process can be high. An alternative approach is to take fewer samples. However, to obtain accurate estimates, it would have to contain correct measurements. In addition, when R is small, median based statistical inferences may converge to inaccurate solutions. Rousseeuw and Leroy [16] proposed to use median based computation to remove the data points that jeopardise the residual estimation. Subsequently, the LS approach can be employed to obtain robust estimations. In order to reject outliers at residual estimation, a suitable threshold is essential. The initial threshold based on a normalised absolute median deviation σ^0, is obtained as follows:

$$\sigma^0 = 1.4836(1 + 5/(R-p))\sqrt{M_{\min}} \qquad (3)$$

where the correlation factor $1.4826 = 1/(0.75\Phi^{-1})$ ensures $med_i |z_i| / \Phi^{-1}(0.75)$ is a consistent estimator of σ when $|z_i|$ is normally distributed as $N(0,\sigma^2)$ and $1 + 5/(R-p))$ is a scale factor obtained empirically to compensate for the correlation factor when the sampling number is small.

$$w_R = \begin{cases} 1 & if \ \left|r_i / \sigma^0\right| < 2.5) \\ 0 & else \end{cases} \qquad (4)$$

The initial weight factor, w_R, is obtained by uniform residual factor, r_i / σ^0, for each pixel. The Final normalised median deviation is obtained by the initial weight estimate in Eq. (4).

$$\tilde{\sigma} = \sqrt{\left(\sum_{i=0}^{R} w_i r_i^2\right) / \left(\sum_{i=0}^{R} w_i - p\right)} \qquad (5)$$

The final weighting factor is obtained by substituting $\tilde{\sigma}$ estimation from Eq. (5) to σ^0 in Eq. (4). The robust LS approach, along with the weighting factor, is used to calculate the affine parameters after robust outlier detection. In order to verify the accuracy of the results, 1500 different patches from real sequences have been gathered with low variation (an example is shown in Fig. 2) between calculated affine parameters by LS and motion in each patch as ground truth data. Different levels of noise have been introduced with changing variance to evaluate the robustness of the proposed technique. As illustrated in Fig. 3, the proposed LMedS can cope well with up to 50% distortion in the data.

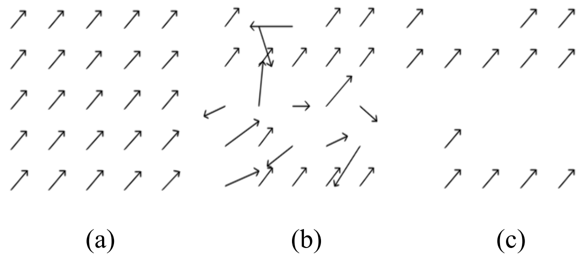


Fig. 2. (a) Illustrates the 5x5 vector window with low distortion which is used to calculate the ground truth data, (b) shows the distorted vector by 50% and (c) demonstrates the restored vector field by proposed LMedS technique.

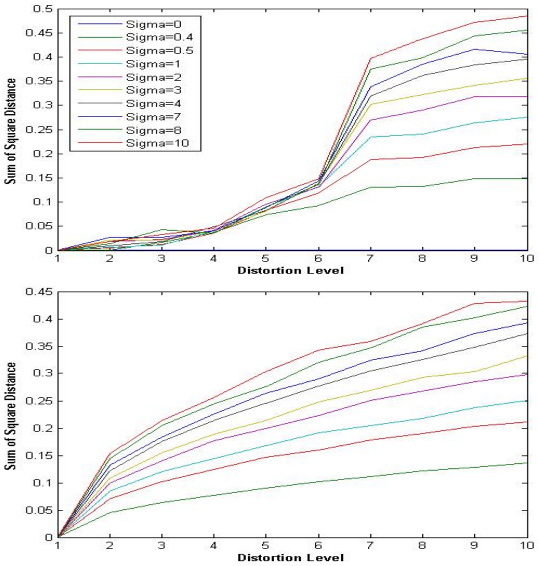


Fig. 3. The Average SSD error estimate between the ground truth data versus estimated affine motions by LS at the bottom and by LMedS at the top. Each distortion level is equivalent to 10% noise in data and 'Sigma' is noise variance for each vector.

2.3 Object tracking and update

In robotic navigation, tracking non-rigid objects is a challenging problem in computer vision. A good strategy must be able to handle occlusion, varying illuminations, as well as rapid movements. In this study, a Kalman filter has been used to track the moving skin coloured objects in an environment. At each step, the measurements are updated by a combination of normalised distance ratio, the size of tracked silhouette and dominant affine motion as in Eq. (6), where L_D is the normalisation factor, defined as the longest distance in the image, and L_a is the affine normalisation factor. In Eq. (6), T corresponds to the tracked silhouette and M is the measurements which are made in each sequence. L_w and L_a are the width and height normalisation factors, respectively. T_{affine} is the predicted affine motion model and M_{affine} is the measured affine motion model. T_w and T_h are tracked silhouette width and height, respectively. In Eq. (6),

$$\begin{aligned} p(T \mid M) = {} & w_1\left(1 - \frac{\sum SSD(T_{xy}, M_{xy})}{L_D}\right) \\ & + w_2\left(1 - \frac{\sum SSD(T_{affine}, M_{affine})}{L_a}\right) + w_3\left(1 - \frac{T_w}{L_w}\right) \\ & + w_4\left(1 - \frac{T_h}{L_h}\right) \end{aligned} \qquad (6)$$

$w_i (i = 1...4)$ are weighting factors and $\sum_{i=1}^{4} w_i = 1$.

Since the position and width are generally more stable, higher weights are assigned to positions and size rather than affine trajectory which can change rapidly. Consequently, if the highest likelihood described in Eq. (6) is above a pre-specified threshold, 0.8 in our case, the measurement has been updated. Otherwise, it is assumed that new objects are appearing in the environment and being added to the Kalman Tracker. The remaining blobs are removed if not updated after 5 consecutive intervals. To compare the proposed tracking method, the popular camshaft method from Intel OpenCv has been employed and the results of tracking are illustrated in Fig. 4. In this scene, the face of the person walking in the scene from behind has been detected as part of the moving skin coloured object. Since the camshaft applies colour information for tracking, when similar coloured objects occlude with each other, the region grows, which is the case in Fig. 4(b). However, the proposed tracker compensates for this fact by considering motion information for each region.

2.4 Hidden Markov Model Gesture Recognition

For gesture recognition, each silhouette motion trajectory has been trained on a 4 state left-right Hidden Markov Model (HMM). To this end, 7 subjects had been asked to perform different gestures. The main gestures considered in this study are used for directing the robot to move left, right, backward, forward and goodbye commands in addition to waving (hello), which is used to grab the robot's attention. Each gesture has been pre-trained, producing 6 models for the final gesture recognition vocabulary.

A total number of 372 gestures have been used in the test phase. These gestures have been tested with the presence of multiple people, increasing the level of distraction. Each subject has been asked to perform gestures at the same time as other people are moving in the environment. These include full and partial occlusions and rapid as well as slow movements. To populate gesture recognition results, $\frac{True\ positive}{True\ positive + False\ negative + False\ positive}$ is used as it is illustrated in Table 1. This indicates that accuracy decreases as the number of people increase due to possible occlusion and possible illumination changes. Different types of occlusion have been considered, which include motion occlusions and similar colour occlusion. Example results are illustrated in Figs. 5, 6, 7 and 8 and further explained in the figure captions. All the tracked objects are colour coded and recognised gestures are displayed in the same colour as the gesture being extracted.

TABLE I
GESTURE RECOGNITION RESULTS

Number of people	Wave (hello)	Turn Right	Turn Left	Back-ward	For-ward	Good bye
1	92%	90%	91%	88%	84%	90%
2	80%	85%	83%	80%	81%	88%
3	73%	78%	77%	75%	80%	76%
4	71%	73%	75%	70%	75%	72%

3. Conclusions and future work

In this work, a robust motion estimation and outlier detection for calculating dominant affine motion based on LMedS has been proposed. Analytical results indicate the practical value of this method for outlier detection, which improves tracking and gesture recognition in the presence of occlusion and different illumination conditions. We have demonstrated the use of the derived motion field for detecting simple gestures in a crowded environment. The current limitation of the method occurs when distraction and occlusion levels exceed a certain limit. One potential approach to solving this is to incorporate 3D depth information to further improve the robustness of the technique. Future challenges include increasing the complexity of the scenes and incorporating gaze and head orientation to better understand intension and attention of the user in order to maintain active communication.

References

[1] J. A. Adams and M. Skubic, "Introduction to the Special Issue on Human Robot Interaction," *Systems, Man and Cybernetics, Part A, IEEE Transactions on,* vol. 35, pp. 433-437, 2005.

[2] R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz, and J. G. Taylor, "Emotion recognition in human-computer interaction," *Signal Processing Magazine, IEEE,* vol. 18, pp. 32-80, 2001.

[3] K. Hyoung-Rock, L. KangWoo, and K. Dong-Soo, "Emotional interaction model for a service robot," in *Robot and Human Interactive Communication, 2005. ROMAN 2005. IEEE International Workshop on*, 2005, pp. 672-678.

[4] R. W. Picard, E. Vyzas, and J. Healey, "Toward machine emotional intelligence: analysis of affective physiological state," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 23, pp. 1175-1191, 2001.

[5] L. Vidrascu and L. Devillers, "Annotation and detection of blended emotions in real human-human dialogs recorded in a call center," in *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, 2005, p. 4 pp.

[6] D. M. Gavrila and J. Giebel, "Shape-based pedestrian detection and tracking," *Intelligent Vehicle Symposium, 2002. IEEE,* vol. 1, pp. 8-14 vol.1, 2002.

[7] L. Seong-Whan, "Automatic gesture recognition for intelligent human-robot interaction," 2006, pp. 645-650.

[8] Y. Ming-Hsuan, D. J. Kriegman, and N. Ahuja, "Detecting faces in images: a survey," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 24, pp. 34-58, 2002.

[9] Z. Hanning, D. J. Lin, and T. S. Huang, "Static Hand Gesture Recognition based on Local Orientation Histogram Feature Distribution Model," in *Computer Vision and Pattern Recognition Workshop, 2004 Conference on*, 2004, pp. 161-161.

[10] Y. Zhu, H. Ren , G. Xu , and X. Lin "Toward real-time human-computer interaction with continuous dynamic hand gestures," in *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, 2000, pp. 544-549.

[11] D. Chai and K. N. Ngan, "Face segmentation using skin-color map in videophone applications," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 9, pp. 551-564, 1999.

[12] D. Tsishkou, M. Hammami, and C. Liming, "Face detection in video using combined data-mining and histogram based skin-color model," 2003, pp. 500-503 Vol.1.

[13] M. J. Black and P. Anandan, "A framework for the robust estimation of optical flow," in *Computer Vision, 1993. Proceedings., Fourth International Conference on*, 1993, pp. 231-236.

[14] G. D. Borshukov, G. Bozdagi, Y. Altunbasak, and A. M. Tekalp, "Motion segmentation by multistage affine classification," *Image Processing, IEEE Transactions on,* vol. 6, pp. 1591-1594, 1997.

[15] J. Y. A. Wang and E. H. Adelson, "Representing moving images with layers," *Image Processing, IEEE Transactions on,* vol. 3, pp. 625-638, 1994.

[16] L. Peter J. Rousseeuw, "Robust regression and outlier detection," 1987.

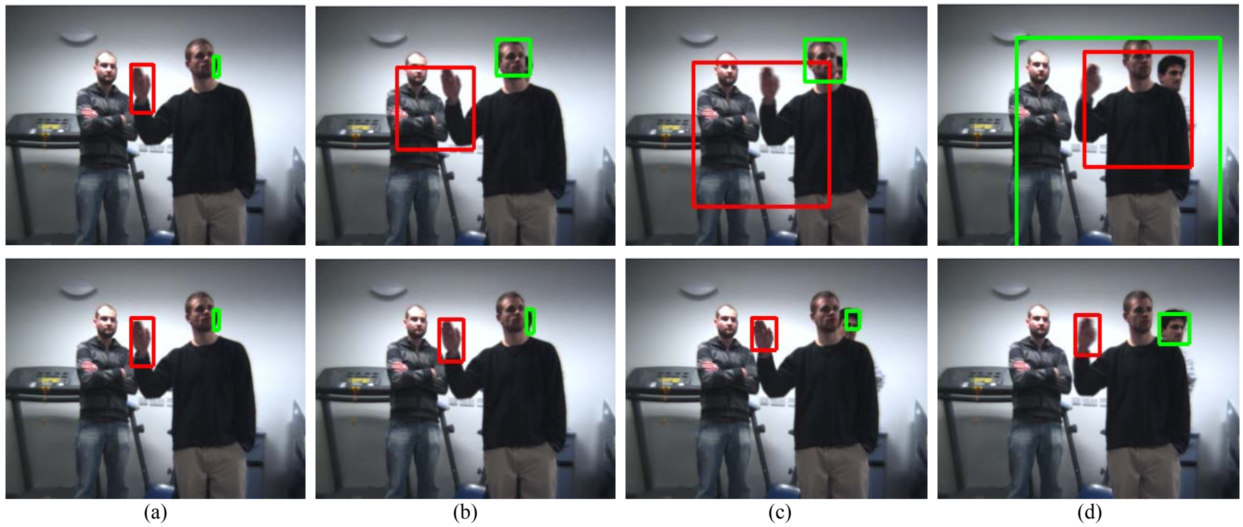
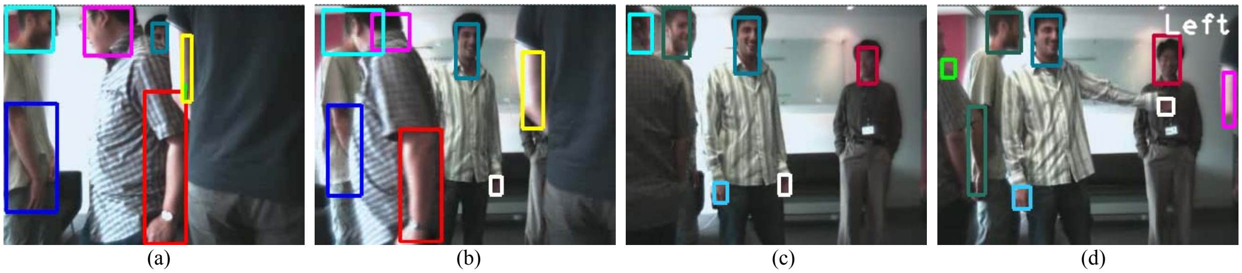
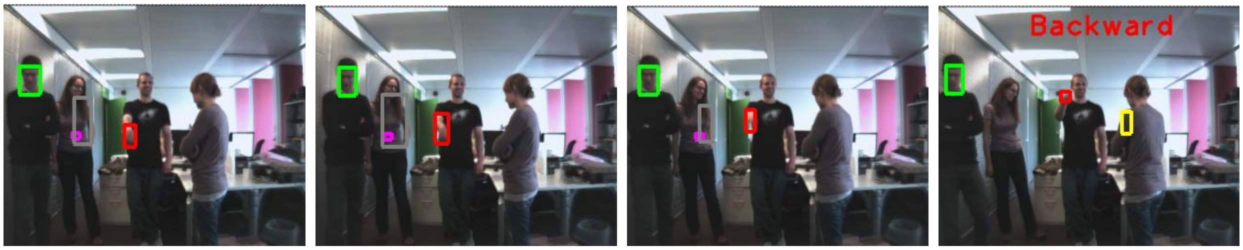

(a) (b) (c) (d)

Fig. 4. The top row demonstrates the consecutively tracked moving skin regions based on the Camshift algorithm while in the bottom row shows the result of the proposed method. In this figure, different colours indicate the different objects being tracked at each time. Column (a) shows the initial frame where each region is assigned; (b-d) illustrate the frame numbers 8, 28 and, 55. On the top row, it can be seen that camshaft fails whereas proposed method, on the bottom row, successfully tracks the objects.



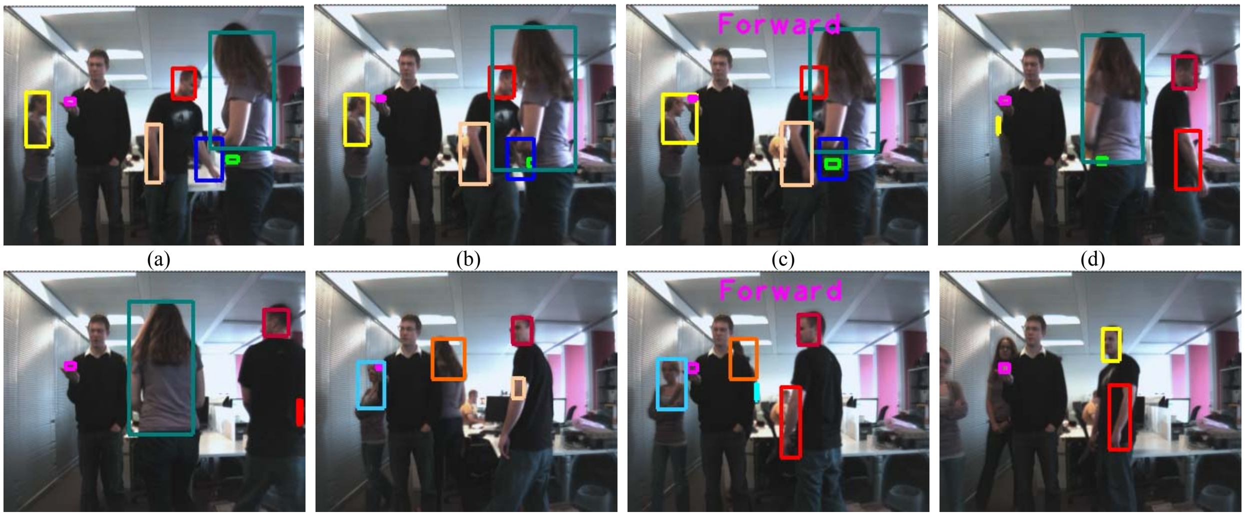
(a) (b) (c) (d)

Fig. 5. Image sequence demonstrating how a "move towards left" command is recognized in the presence of multiple people.



(a) (b) (c) (d)

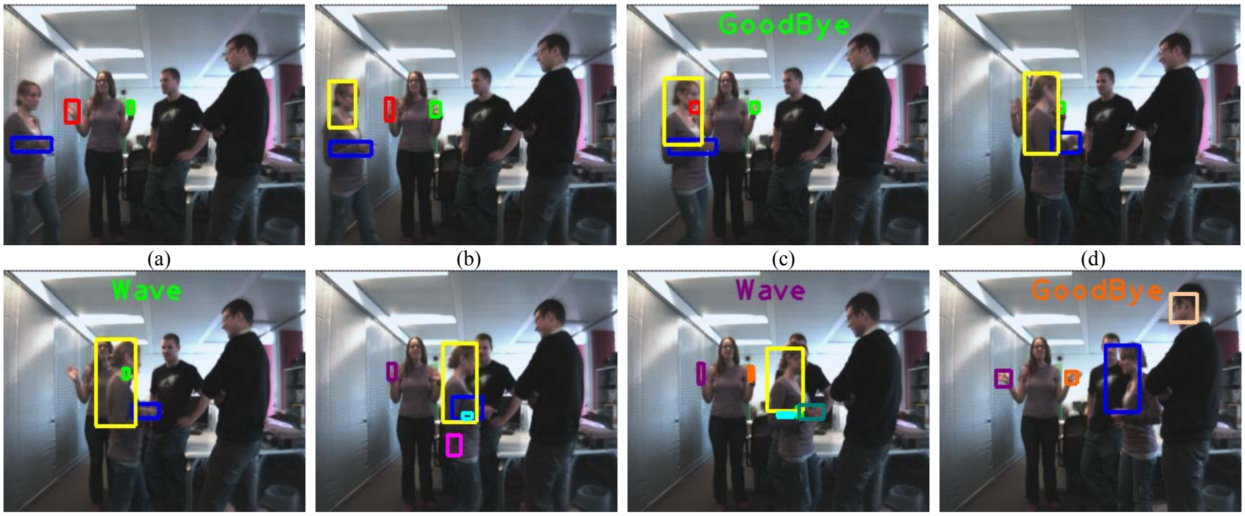
Fig. 6. The backward command consists of repeated vertical movements as depicted by the red blob. This has been detected successfully in (d).



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 7. Image sequence showing the subtle "move forward" command is detected in the presence of multiple people in (c) and (g). The hand performing forward command in purple blob is recovered from partial occlusion caused by the other person, as indicated by yellow and blue blobs, walking behind (c) and (g) respectively.



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 8. Image sequences showing the detection of the 'Goodbye' command which consists of waving two hands at the same time. This has been detected successfully despite another person creating distraction in (c) and (h). The red blob in (d), which is part of the hand gesture, disappears due to other person walking in front, consequently the green blob is detected as waving in (e). A new blob in purple and orange is assigned to the person's hand in (f) and (g). In (g), the waving command is detected due to the lack of motion trajectory in the orange blob caused by earlier occlusion. This has been detected successfully in (h).