Journal of Intelligent Learning Systems and Applications, 2014, 6, 141-152
Published Online August 2014 in SciRes. http://www.scirp.org/journal/jilsa
http://dx.doi.org/10.4236/jilsa.2014.63011

Scientific Research

Image-Based Methods for Interaction with Head-Worn Worker-Assistance Systems

Frerk Saxen[1], Omer Rashid[1], Ayoub Al-Hamadi[1], Simon Adler[2], Alexa Kernchen[2], Rüdiger Mecke[2]

[1]Institute for Information Technology and Communications, Otto-von-Guericke-University, Magdeburg, Germany
[2]Fraunhofer Institute for Factory Operation and Automation (IFF), Magdeburg, Germany
Email: Frerk.Saxen@ovgu.de, Omer.Ahmad@ovgu.de, Ayoub.Al-Hamadi@ovgu.de, Adler@iff.fraunhofer.de, Alexa.Kernchen@iff.fraunhofer.de, Mecke@iff.fraunhofer.de

Received 24 May 2014; revised 28 June 2014; accepted 10 July 2014

Abstract

In this paper, a mobile assistance-system is described which supports users in performing manual working tasks in the context of assembling complex products. The assistance system contains a head-worn display for the visualization of information relevant for the workflow as well as a video camera to acquire the scene. This paper is focused on the interaction of the user with this system and describes work in progress and initial results from an industrial application scenario. We present image-based methods for robust recognition of static and dynamic hand gestures in real-time. These methods are used for an intuitive interaction with the assistance-system. The segmentation of the hand based on color information builds the basis of feature extraction for static and dynamic gestures. For the static gestures, the activation of particular sensitive regions in the camera image by the user's hand is used for interaction. An HMM classifier is used to extract dynamic gestures depending on motion parameters determined based on the optical flow in the camera image.

Keywords

Head Mounted Display, Gesture Recognition, Mobile Assistance System

1. Introduction

Despite a permanent increase of automation in industrial production, manual labor remains a key factor for the manufacturing of complex technical products (e.g. automobiles, machine tools). Considering the development in

How to cite this paper: Saxen, F., Rashid, O., Al-Hamadi, A., Adler, S., Kernchen, A. and Mecke, R. (2014) Image-Based Methods for Interaction with Head-Worn Worker-Assistance Systems. *Journal of Intelligent Learning Systems and Applications*, **6**, 141-152. http://dx.doi.org/10.4236/jilsa.2014.63011

these fields of industry, it can be stated that both the complexity and the quantity of possible product variants are rising enormously. At the same time, there are high requirements regarding the product quality and production time. Information about individual production steps as well as specific expert knowledge of the manufacturing process at hand becomes more and more important in this context. In order to meet the mentioned demands, several strategies and methods could be used. One of these possibilities is to support the employees by assistance technologies which bring information about the manufacturing process directly to the workplace. Additionally, recognition of the users intention or relevant objects in the working environment can be a significant part in assistance technologies.

In most industrial environments where hands-free working activities are required, the usage of head mounted displays (HMD) as a mobile display device can provide the worker with situation-dependent information. A system with an HMD is able to augment visual information relevant to the working task in the users field of view. Processes like order picking in a logistic warehouse or storage can be supported by an HMD as a guidance system. The potentials of HMDs are well-known [1] and their capabilities are shown in different application-scenarios [2]. On the other hand, there is a lack of user-friendliness in many systems that may lead to rejections or prejudices [3]. The need of research regarding user-based aspects were demanded early [4] but are not solved until now [5].

The interaction between the user and the mobile assistance system is an important issue in this context. Especially for industrial environments, common paradigms (e.g. interaction methods for hand-held systems) can not be used directly but need to be adapted. In this paper, we describe an assistance system for an industrial application scenario as well as approaches for an intuitive interaction with the system using image-based hand gesture recognition.

1.1. Visual Assistance System

As mentioned above, manual work as well as appropriate information is usually required in order to manufacture complex technical products under the requirements of industrial environments at hand. By using a mobile system to bring information directly to the users workplace, a better integration in the workflow can be achieved compared to stationary terminal, that provide information aside from the functionality of the workplaces. Recently, new wearable display devices (HMD) became available, which can be worn by the worker and fulfill the "hands-free" requirement.

An HMD is a wearable monitor that is positioned in the users field of view. A distinction between Video-See-Through (VST) and Optical-See-Through (OST) devices are made. VST devices capture the scene in front of the HMD by an integrated video camera. The video image is displayed on the HMD monitor. For this reason, the user sees the real world environment only via its online video, if the monitor covers the whole field of view. Additional data (e.g. work instructions) are overlaid to the video stream.

OST devices contain a semitransparent display through which the user acquires the scene. Additional data are projected via a beam splitter directly onto the semitransparent display. Therefore, a camera is not directly required and for these devices, not always present. OST devices are commonly used for augmented reality (AR) applications where virtual information can be integrated into real world scenes. OST devices are preferred for the use in industrial application scenarios [6] [7] due to the fact that the scene is directly visible and a system failure does not effect the users view. The video-based view of the VST device impairs the hand-eye coordination, which has a negative impact on the workers accuracy [6] and therefore on the quality of the work. Furthermore, VST devices have a latency between grabbing the image and displaying it onto the screen. This latency is not present in OST devices.

The information visualization on an HMD can be used as a replacement for e.g. printed instruction manuals, whereas video or animated computer graphics to present the dynamic content can be integrated. Using an HMD requires appropriate interaction devices for user control. Some devices offer additional buttons on a separate computation unit carried for instance at the user's belt. Recent devices integrate the buttons on the glass frames with the computation unit embedded. An additional input method for interaction is the usage of data gloves as well as space mice [8]. In industrial applications where the worker should not release the tools for the interaction, hands-free methods are required. Therefore, buttons for the interaction cause a lack of user-friendliness that may lead to rejections or prejudices [3]. Instead, contact-free input methods like eye movement, speech or gestures

can be taken into account. Eye tracking offers advantages in navigation but shows great disadvantages in performing selection tasks [9] and increases the visual strain. Considering the ambient noise in industrial environments, the use of speech recognition is often not feasible. Interaction through gestures seems to be the most flexible solution, if specific gestures are used that may be trained by the user in advance.

Considering virtual environments, gesture-based interactions are already used. While a sufficient detection is often not reached, special devices are used to detect the users intentions [10]. Whereas the need of research regarding user-based aspects in HMD-based applications were demanded early [4] [5], the interaction with the presented information in mobile application scenarios requires further research.

We present a visual assistance system using image processing for interaction by gesture recognition, which is the first step and it adapts to the users movement.

1.2. Gesture Recognition

Depending on concrete work task to be supported by an assistance system, there are different possibilities for choosing appropriate hand gestures. One possibility is to define the amount of interaction instructions necessary to control the system and then choosing suitable gestures for each instruction. Thereby, it has to be considered that the gestures are easy to learn as well as they may be performed intuitively by the user. On the other hand, the gestures have to be as simple as possible to be recognized robustly by an image based system.

We are using an OST HMD to present the information related to the work procedure as well as a mounted camera acquiring the scene in front of the user (**Figure 1**). The video image is used for image processing and to recognize gestures performed with one hand of the user. Hand gestures can be categorized in symbolic, iconic and deictic gestures [11]. For symbolic and iconic gestures, the shape of the user's hand in image space has a specific meaning. The position where the gesture is performed has normally no influence on the meaning of the gesture. Symbolic gestures have a direct translation into words while iconic gestures can be used to describe the properties of objects. While performing deictic gestures, the hand points to particular regions in the scene, so the position and orientation of the hand is relevant for the meaning of the gesture [12].

Hand gesture recognition systems are proposed since more than a decade. The promising application fields inspired plenty other researchers to propose new approaches (e.g. in automatic sign language recognition and human computer interaction). The upcoming of depth cameras also influenced the research in gesture recognition, covering versatile applications [13]. **Figure 2** shows a common workflow of gesture recognition systems. The image capturing, preprocessing, and segmentation step depend on the camera setup. The segmentation step includes background subtraction (*i.e.* if the camera setup is static or depth information is available) and skin segmentation (*i.e.* if color is present). In literature, researchers consider hand posture or hand gesture but there are other works dealing with both approaches [14]. The associated action to the classified gesture and the environmental perception always depends on the application. The interaction model needs the environmental perception to understand the users actions according to its environment and to change accordingly e.g. the gesture associations.

Most applications according to hand gestures deal with static cameras and therefore benefit from background modeling [13] or use depth information to segment region of interests [15]. Whether static or dynamic cameras, mono or depth information, the majority of approaches use geometric features on segmented skin areas [16]. Skin segmentation algorithm can be divided into pixel-based and region-based approaches. The region-based algorithms are only scarcely used due to time consumption and have small benefits in classification results. The pixel-based approaches can be divided into threshold-based and training-based methods. The threshold-based algorithms (e.g. [17] [18]) use explicitly defined skin regions for segmentation. The training-based methods instead use classifiers to train a model (e.g. [19] [20]). Even though training-based methods provide superior results, threshold-based algorithms are more popular because they are easy to implement and don't need any training dataset. Finding hand candidates is generally based on skin segmentation [21]. Nevertheless, incorporating additional features can improve the hand detection task [22]. The hand posture classification is often based on geometric [13] or statistical features [23] using Support Vector Machines or Artificial Neural Networks. The dynamic features for hand gesture recognition are based on hand trajectories, whereas reliable hand tracking takes a key role [24] [25].

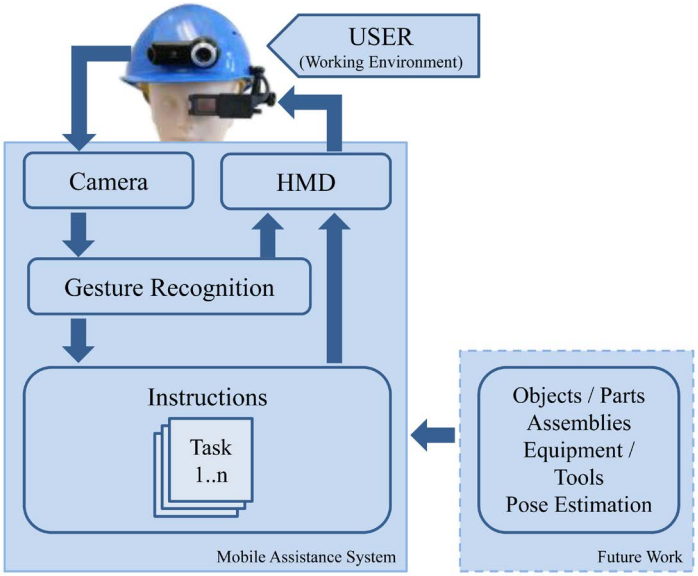

Figure 1. Concept of the mobile assistance system. The information flow is indicated by arrows.

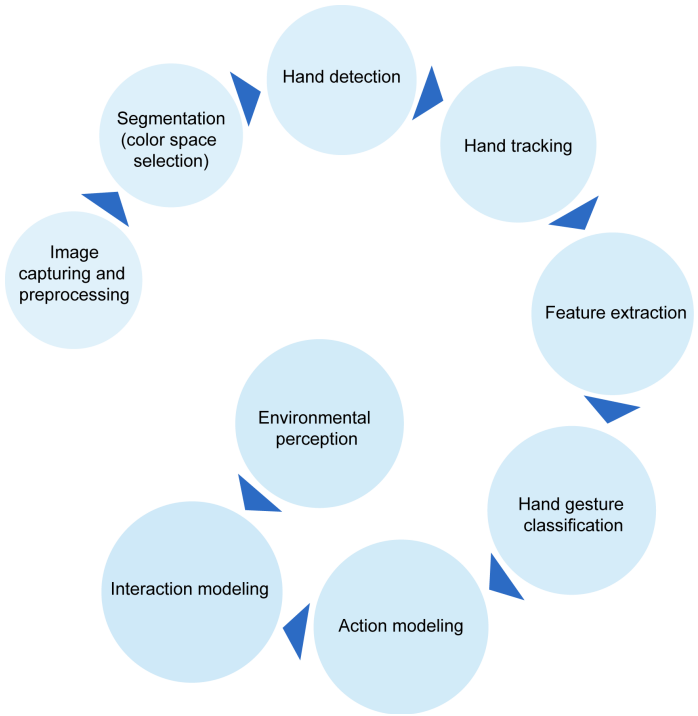

Figure 2. Work flow of gesture-based visual assistance system.

1.3. Overview

This paper is organized as follows. Section 2 explains the application scenario in detail. Section 3 describes the gesture based system for visual assistance in industrial applications. We present methods for recognition of dynamic gestures for frequently used interactions and also a method for static deictic gestures where the position of the hand in the image has a certain relation to overlaid visual elements of the graphical user interface visualized in the HMD. Section 4 concludes the advantages, difficulties and perspectives of our approach and gives an overview about future work.

2. Application Scenario

The application scenario is a visual assistance system providing procedure related information for workers in industrial applications during qualification and training (**Figure 3**). The information can be the content of the procedure (what should be done), which tools should be used, how the tools should be used (handling), and the body posture which the workers have to assume during the procedure. The tool handling and the body posture is not only important to optimize the work performance but also for health and safety at work.

The training often uses a combination of instructions from a printed manual and instructions provided by a trainer. For the printed instructions, the worker requires his hands and has to release the tools during training. This is an interruption of the working procedure and consumes additional time from the trainee to perform all procedures of an activity continuously. Supervisors have the disadvantage that the instruction of the tool handling or the procedures may vary between supervisors. Thus, a standardization of the training process could be difficult. The presentation of activities in a virtual environment could be used to present the procedures to overcome this problem, but a monitor and the interaction with the virtual environment is required which probably requires more attention than the interaction with printed instruction manuals. Therefore, in this paper we focus on a mobile application using OST HMDs (**Figure 4**) where the information is augmented in the workers field of view, so that he has not to avert his eyes from his current work. The interaction with the assistance system presented on the HMD is performed by gestures.

We use symbolic dynamic gestures for frequently used gestures. The movement of the hand to the left or right can be used to switch to the next or previous information. Movements down and up can hide or show the information visualization. For not frequent interactions we use static deictic gestures that are related to the graphical user interface of the assistance system displayed on the HMD. Changing the setup of the system, e.g. change color or contrast of the visualization, or looking up specific information which must be chosen from a menu are examples for these not frequent interactions.

3. Image-Based Gesture Recognition

In literature, static and dynamic gestures are discerned, where both require different algorithms. In most cases, static gestures (e.g. pointing) are detected by applying a skin tone detection (which is still a challenging task [20]) to identify the region of interest (ROI) before extracting and classifying features [13]. Dynamic gestures (e.g. wipe gestures) are based on an image sequence and often classified according to the motion trajectory of the hands using tracking algorithms [25]. These methods often do not or only scarcely cope with camera motion [26] and have their issues with the large variations in shape and appearance of hands (with more than 21 degrees of freedom).

Our method for dynamic gestures copes with camera motion and reduces the complexity of shape and appearance of hands for robust detection. This method has the drawback of ambiguity of some gestures resulting in a sparse gesture set.

Although stereo vision has been successfully used for gesture recognition [21] [27], the required camera systems (calibrated cameras, time of flight cameras) and the additional computation effort are not suitable for mobile systems. Therefore, in this paper our gesture based recognition system takes its observations from 2D camera images. These images are processed to retrieve the static and dynamic contents. The proposed methods can be applied for interaction with or without gloves [15], but in the following we focus on interactions without gloves. The static content is extracted by utilizing the local skin information of the hand, however, the dynamic contents are retrieved by additionally exploiting the flow information. Both, static and dynamic content build the basis for the extraction of static and dynamic gestures respectively.

Depending on the systems setup, the HMD and camera can be fixed or moveable. In an exemplary setup seen in **Figure 1**, the camera[1] used for image-based gesture recognition and the HMD[2] is attached to a helmet worn by the user. In both setups, an offset in position, orientation and field of view occurs. To improve the usability and prevent unintended interactions with the system, the gestures have to take place in the volume of the intersecting viewing frustums (**Figure 5**). The correct positioning of these frustums can be achieved via mapping methods.

In the context of the intended usage, we developed a simple mapping approach. Whereas the static gestures take place in the resulting 3D frustum of sensitive 2D regions visualized in the HMD (**Figure 5(2)**), the dynamic gesture expand to whole viewing frustum of the HMD. The three dimensional extension (depth) of the frustum

[1]Logitech Quick Cam Pro9000.
[2]LitEye LE 750A.

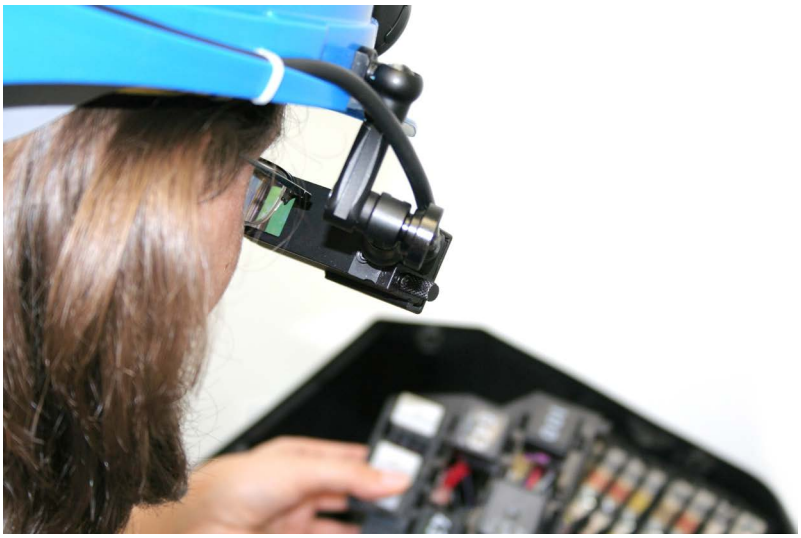

Figure 3. The user is wearing an HMD during the assembly of a control unit. Visual information are displayed in the users field of view.

Figure 4. View of the user through the HMD. The buttons (arrows) are used to toggle to the next or previous task. The left button is activated by the users hand in a predefined region.

is limited to the length of the users arm. With a known position of that frustum in the camera coordinate system, the perspective projection of the frustum in the camera image result in a mask in which the gesture detection can be done.

To obtain the 3D position, the frustum is placed on a plane (see restriction in depth, **Figure 5**), which can be tracked by an image-based method using id-marker [28]. The user manually aligns an augmented shape (visualized in the HMD) with the id-marker. With a correct alignment, it is assumed that the user has a perpendicular viewing angle onto the id-marker. The size of the augmented representation of the marker in the HMD is known via the specified distance of the user to the marker and the size of the id-marker in the real world. With the computed pose of the id-marker and the derived positions of the frustum of the region, its position within the camera image can be determined.

For applications, where the augmented information needs to be aligned more accurately with the real world, a more complex calibration process is needed. Algorithms like [29]-[31] provide solution to this problem and might be of interest in future work.

After an initial calibration of the system, changes in position between the HMD and the camera will require a new calibration. If the camera and the display are integrated in a fixed setup, the calibration needs to be performed only once.

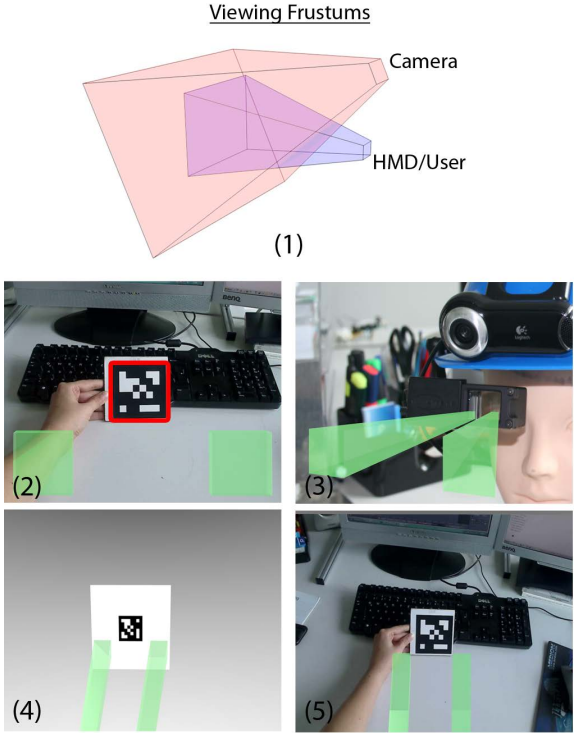

Figure 5. (1) Visualization of the viewing frustums of the camera and the HMD. Depending on the systems setup the intersections can vary largely. (2) View of the user through the HMD with the sensitive regions and the shape for the ID-Marker mapping (red rectangle). (3) 3D representation of the sensitive regions as square frustum. (4) The region frustums are positioned on the marker plane respectively to the desired layout. (5) The frustums are positioned in the camera image via the results of the mapping procedure.

3.1. Recognition of Static Gestures

Dynamic gestures are suitable for common and frequent interactions but our training application also requires that the user has to select, e.g., a menu entry or the next task, represented by sensitive image regions. Static deictic gestures are suitable for interactions with these augmented elements displayed in the HMD (**Figure 4**). The sensitive regions are visualized in the users field of view and can be activated by the users hand.

Our method for static gesture recognition is based on skin tone detection with a computational inexpensive algorithm to support the applicability on mobile devices. Limitations in memory and process speed have a significant impact on the usability of applications. Latency and update rates are of great importance for the acceptance of the system. The main approaches are to reduce the number of processing iterations over the camera image and to support parallelism of the algorithms, where parallelism requires more than one processing unit on the device.

1) *Skin tone detection*: The skin tone detection for our static gesture is using RGB (Red, Green, Blue) and HSV (Hue, Luminance and Saturation) color space. For the RGB color space, simple rules (Equation (1)) [17] are applied to the camera image.

$$\begin{aligned}&(R,G,B)\ \text{Pixel with 8 bit depth is skin if}: R>95 \text{ and } G>40 \text{ and } B>20\\ &\text{and } \max\{R,G,B\}-\min\{R,G,B\}>15 \text{ and } |R-G|>15 \text{ and } R>G \text{ and } R>B\end{aligned} \quad (1)$$

The skin detection in the HSV color space uses thresholds on all channels. A priori, pre-classified images containing skin were used to determine the threshold values. Both algorithms only use local pixel information

for the conversion between the color spaces and the skin tone detection itself. Therefore, the conversion in both color spaces and the detection can be performed in a single loop and parallel. There is a large intersection between the results of both methods. Since the static gesture detection does not rely on a precise segmentation of the users hand, an over-segmentation is preferred rather than an under-segmentation. Hence, the HSV skin detection will be performed after the RGB detection failed, in order to extend the result. Nevertheless, the results of static gesture recognition mainly depends on the skin segmentation performance [17], which is sensitive to lighting and skin-like objects.

2) *White-balancing*: Due to the dependency on specific color values of the segmentation, variation of color temperature in the camera image (changing environmental conditions and illumination) result in over- or under-segmentation. Consequently, this has a direct effect on the quality of the gesture detection which results in faulty or missing activations. By using automatic and semi-automatic white-balancing, these effects are minimized. Methods like *white world and gray world assumption* [32] are used for an automated balancing. Depending on the environmental conditions, these methods create insufficient results, when the image contains no "white" color or is overexposed. In this case, we also provide a semi-automatic white-balancing. Initially, the user is required to place a visualized rectangle above a white surface. The mean color value in this rectangle defines an offset to "real" white color. The offset is applied at run-time to the image pixel values.

3) *Activation process*: For the activation of the sensitive regions, the area covered by skin tone is determined. If the relative portion of the covered area is above a predefined threshold, the region is assumed to be touched. To reduce unintended activations, a touched region requires several consecutive image frames of a touched status to be finally activated. By allowing gaps in these consecutive frames, the activation process is more robust against potential failed detections of the user touching the region. If the region is fully activated (*i.e.* button is pressed), the associated action is performed. Equivalent, the deactivation requires a consecutive status of "not touched" of the region in order to distinguish between multiple intended activations by the user. Because of the nature of the activation process, it can be associated as loading and unloading the region. As feedback, the respective status (un-/loading level) for each frame is visualized to the user (**Figure 6**). The time intervals for loading (1 second) and unloading (1/2 second) were manually determined based on user feedback. Optimal values need to be acquired in further studies.

3.2. Recognition of Dynamic Gestures

Our dynamic gesture recognition module utilizes skin and motion information to extract the dynamic contents. In the first step, the image observations are transformed from RGB to the Improved Hue, Luminance and Saturation (IHLS) color space [20] for hand segmentation using the pixel based Bayes approach [19], trained on the ECU database [33]. The Bayes approach counts the relative frequency of skin and non-skin pixel values in the training set (resulting in a lookup table of size 256^3). For unknown data (during processing), the relative frequency of the pixel value is used to judge whether it belongs to skin or not (relative frequency of pixel value in training set is higher or below threshold $\tau = 0.243$ respectively). τ fulfills the condition for a trade-off between over- and under-segmentation by using the f-measure. In the next step, the motion estimation is performed using Farneback *et al*. [34] which computes the optical flow based on polynomial expansion. Farneback algorithm produces a 2D velocity field between two consecutive images in a dense manner and is robust to illumination changes. Moreover, to enhance the speed of optical flow computation, we use downscaled (by factor 4 for each axis) gray scale images. However, it is quite hard to distinguish between hand and camera motion. Therefore, we use the optical flow to estimate the ego-motion by considering the mean flow and its variance in border areas to find the majority motion of the image, which is assumed to be background motion and subtracted from the image flow. Taking into account that gestures are localized more often in the center of the image, the corners are weighted more to extract the background movement.

It is observed that the flow data includes richer motion information of the scene (*i.e.*, shirt sleeves are extracted while drawing the gestures). So, we extract only those dynamic contents which are conditioned to skin information and calculate the hand centroid points (x_t, y_t) which are then used to compute the gesture features.

1) *Gesture Feature extraction*: Feature extraction is an essential step for the classification, but selection of good features is always challenging. In the proposed approach, hand orientation is used as a feature and is determined between two consecutive hand centroid points (Equation (2)) when drawing the gesture path.

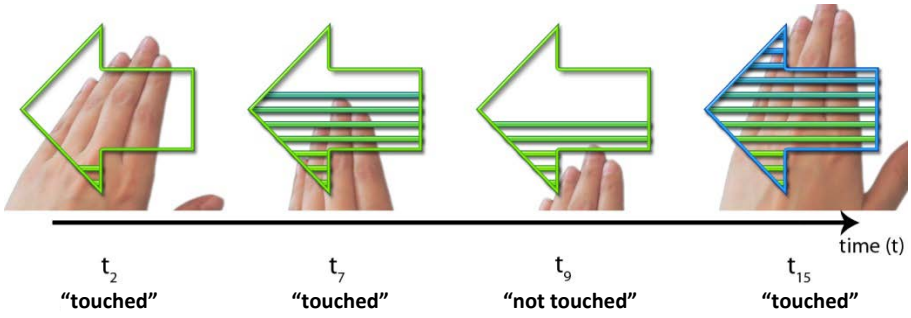

Figure 6. The buttons (arrows) fill according to the activation level. When the user touches the button (t_2) the loading starts and continues to increase the activation level (t_7). A failed detection of the user touching (t_9) will not result in the restart of the activation process, but merely in the reduction of the loading status.

$$\varphi_t = \left\lfloor \frac{\arctan\left(\frac{y_t - y_{t+1}}{x_t - x_{t+1}}\right)}{\pi/4} \right\rfloor \tag{2}$$

At each frame $t \in [1, T]$, T represents length of gesture path, and x_t and y_t are hand centroid points, the symbol φ_t is computed ranging from 0 to 7. The discrete vector φ contains the symbols of the gesture path and is used to classify with a hidden markov model (HMM).

2) *Gesture classification*: We use the Hidden Markov Model (HMM) to train and classify the gesture symbols. HMM is a generative classifier and is a mathematical model of stochastic process which produces a random chain of outcomes according to the probabilities [35]. In our approach, this stochastic process is a chain of code words (*i.e.* discrete vector φ) in HMM and its output is the classification of gesture path symbols (*i.e.* (L)eft, (R)ight, (U)p, and (D)own). In HMM, there are three different topologies namely Ergodic model, Left-Right, and Left-Right Banded (LRB). Ergodic model is the topology in which every state can be reached from every other state of the model. Left-Right topology is the one in which every state can be reached by itself or to the following states. Left-Right Banded (LRB) model is the topology in which every state can reach by itself or to the next state. In the proposed approach, we have used LRB topology with 5 states for gesture recognition. A discrete HMM is described as:

1) Set of states $S = \{s_1, s_2, \cdots, s_N\}$ where N is the number of states.

2) Set of observation symbols (discrete vector) $\varphi = \{\varphi_1, \varphi_2, \cdots, \varphi_T\}$ where T is the number of observable symbols at every state.

3) Initial probability for each state π_i, where $i = 1, 2, \cdots, N$.

$$\pi_i = \text{Prob}(s_i), \qquad \pi_i \geq 0, \qquad \sum_i \pi_i = 1 \tag{3}$$

4) Transition matrix $A = \{a_{ij}\}$ of size $N \times N$ where a_{ij} is the transition probability from state i to state j at any time instance t:

$$a_{ij} = \text{Prob}(s_t = j \mid s_{t-1} = i), \quad 1 \leq i, j \leq N,$$
$$a_{ij} \geq 0 \quad \sum_j a_{ij} = 1 \tag{4}$$

5) Observed symbols matrix $B = (b_{it})$ of size $N \times T$ where b_{it} provides us the probability of emitting symbol φ_t at state *i*:

$$b_{it} = \text{Prob}(\varphi_t \mid s_i), \quad 1 \leq i \leq N, 1 \leq t \leq T,$$
$$b_{it} \geq 0 \quad \sum_t b_{it} = 1 \tag{5}$$

There are three basic methods used inside the HMM for the training and testing phase. The first method is to use forward and backward algorithm which computes the probability of observation sequence $\text{Prob}(\varphi|\lambda)$ given the observations φ and model parameters $\lambda = (A, B, \pi)$. Further, we adjust the parameters of our model so that the likelihood $\text{Prob}(\varphi|\lambda)$ is locally maximized, which is done through Baum-Welch method. These two algorithms are used during the training phase of the HMM. Given the observation sequence φ and model parameters, the third algorithm named Viterbi is used in the testing phase which finds the optimal path for the corresponding state sequence by selecting the maximal observation probability of the gesture model. **Figure 7** shows the steps of our recognition module including the classified gesture.

3.3. Experimental Results

We have evaluated our dynamic gesture recognition system with 61 video sequences containing four different gestures from a mounted camera on a local dataset. The performed gestures taken from dataset with 12 (L)eft, 11 (R)ight, 23 (U)p, and 15 (D)own examples were detected robustly (see **Table 1**).

The dynamic gesture recognition is currently implemented for desktop computer systems. To consider the demand of mobility and to identify performance issues, the static gesture recognition has been evaluated on mobile platforms. The achieved performance meets the requirements for real-time applications. Nevertheless, on mobile devices, the camera image is commonly provided in a compressed 4:1:1 YUV color space, which may impair the skin tone detection.

4. Conclusions and Future Work

In this paper, we have presented a concept and the initial results for a visual assistance system using gestures for controlling the system and an HMD for displaying the information. Methods for static and dynamic image-based gesture recognition for hands-free interactions were described. Using the selected static and dynamic gestures, we were able to show the feasibility of our concept. The static gestures were implemented on mobile platforms to identify initial issues due to the limited hardware setup. The approach for the dynamic gesture recognition copes with camera motion. Therefore, in this paper, applicability of the common and frequently used gestures is given. Since the user generally concentrates on the interaction while performing the gesture, the camera motion exceeds its compensation limits rarely. The skin segmentation on the other hand does impair the classification performance.

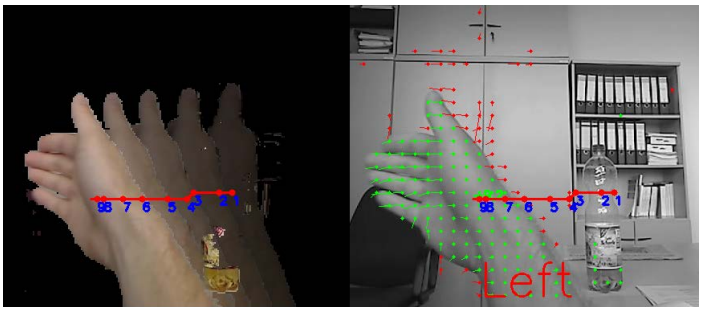

Figure 7. Segmentation, feature extraction and classification example. Left: Segmentation results with feature points of several consecutive frames of a gesture displayed as a motion history image. Right: Optical flow of skin regions (green) and non-skin regions (red) after ego-motion compensation and gesture classification result.

Table 1. Confusion matrix for 4 predefined dynamic gestures (in %).

Gesture	L	R	U	D
L	97	1	1	1
R	0	99	1	0
U	1	13	86	0
D	13	0	0	87

In the future, we will address not only aspects of user interaction but also the use of situation dependent information by considering the users actions with respect to the environment. Therefore, it is necessary to investigate the methods for the recognition of objects in the working-environment. Furthermore, the user-friendliness remains a pertinent attribute of our visual assistance system along with the real-time capability, mobility, and hands-free usage. Moreover, we will investigate the usability of different gestures according to their operability and intuitiveness. Additionally, we will evaluate the system in the real process of the workers training.

Acknowledgements

This work is a part of the project done within the Transregional Collaborative Research Centre SFB/TRR 62 "Companion-Technology for Cognitive Technical Systems" funded by the German Research Foundation (DFG); Transfer project (T01).

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this article.

References

[1] Schwerdtfeger, B., Reif, R., Gunthner, W.A., Klinker, G., Hamacher, D., Schega, L., Bockelmann, I., Doil, F. and Tümler, J. (2009) Pick-by-Vision: A First Stress Test. 8*th IEEE International Symposium on Mixed and Augmented Reality*, Orlando, 19-22 October 2009, 115-124.

[2] Conradi, J. and Alexander, T. (2008) Display Devices for Virtual Environments: Impact on Performance, Workload, and Simulator Sickness. *EGVE* 2008 *Symposium Eindhoven*, The Netherlands, 29-30 May 2008, 103-109.

[3] Smith, S.P. and Hart, J. (2006) Evaluating Distributed Cognitive Resources for Way Finding in a Desktop Virtual Environment. *Symposium on* 3*D User Interfaces* (3*DUI*), Virginia, 25-29 March 2006, 3-10.

[4] Gabbard, J.L., Swan, J.E., Hix, D., Kim, S.-J. and Fitch, G. (2007) Active Text Drawing Styles for Outdoor Augmented Reality: A User-Based Study and Design Implications. *Virtual Reality Conference (VR)*, Charlotte, 10-14 March 2007, 35-42.

[5] Tümler, J., Doil, F., Mecke, R., Paul, G., Schenk, M., Pfister, E.A., Huckauf, A., Bockelmann, I. and Roggentin, A. (2008) Mobile Augmented Reality in Industrial Applications: Approaches for Solution of User-Related Issues. 7*th IEEE/ACM International Symposium on Mixed and Augmented Reality*, Cambridge, 15-18 September 2008, 87-90.

[6] Park, M., Schmidt, L., Schlick, C. and Luczak, H. (2007) Design and Evaluation of an Augmented Reality Welding Helmet. *Human Factors and Ergonomics in Manufacturing & Service Industries*, **17**, 317-330. http://dx.doi.org/10.1002/hfm.20077

[7] Tümler, J., Scharf, C., Mecke, R.M., Schenk, M. and Paul, G. (2008) Incorporating User Preference to Represent Information for Manual Work Supported by Augmented Reality. *SVR* 2008 *Symposium*, João Pessoa, 13-16 May 2008, 196-203.

[8] Fröhlich, B., Hochstrate, J., Skuk, V. and Huckauf, A. (2006) The Globefish and the Globemouse: Two New Six Degree of Freedom Input Devices for Graphics Applications. *ACM CHI Conference on Human Factors in Computing Systems*, Montréal, 22-27 April 2006, 191-199.

[9] Huckauf, A., Urbina, M.H., Tümler, J., Doil, F. and Mecke, R. (2008) Visual Search in Head-Worn Displays. *Perception* 37 *ECVP Abstract Supplement*, **37**, 141.

[10] Kim, H., Albuquerque, G., Havemann, S. and Fellner, D.W. (2005) Tangible 3D: Hand Gesture Interaction for Immersive 3D Modeling. *EGVE* 2005 *Symposium*, Aalborg, 6-7 October 2005, 191-199.

[11] Rim, B. and Schiaratura, L. (1991) Gesture and Speech. Press Syndicate of the University of Cambridge, Ch. 7, Cambridge, 239-284.

[12] Kappas, A., Hess, U. and Scherer, K.R. (1991) Voice and Emotion. In: Feldman, R.S. and Rime, B., Eds., *Fundamentals of Nonverbal Behavior*, Cambridge University Press, Cambridge, 200-238.

[13] Hasan, M.M. and Mishra, P.K. (2012) Hand Gesture Modeling and Recognition Using Geometric Features: A Review. *Canadian Journal on Image Processing and Computer Vision*, **3**, 12-26.

[14] Elmezain, M., Al-Hamadi, A., Rashid, O. and Michaelis, B. (2009) Posture and Gesture Recognition for Human-Computer Interaction. In-Teh, Rijeka, Ch. 23, 415-440.

[15] Rashid, O., Al-Hamadi, A. and Michaelis, B. (2010) Utilizing Invariant Descriptors for Finger Spelling American Sign Language Using SVM. *Advances in Visual Computing*, **6543**, 253-263.

[16] Khan, R.Z. and Ibraheem, N.A. (2012) Hand Gesture Recognition: A Literature Review. *International Journal of Artificial Intelligence & Applications* (*IJAIA*), **3**, 161-174.

[17] Kovač, J., Peer, P. and Solina, F. (2003) Human Skin Colour Clustering for Face Detection. *International Conference on Computer as a Tool EUROCON*, **2**, 144-148.

[18] Cheddad, A., Condell, J., Curran, K. and Kevitt, P.M. (2009) A New Color Space for Skin Tone Detection. 16*th IEEE Internation Conference on Image Processing (ICIP)*, Cairo, 7-10 November 2009, 497-500.

[19] Jones, M.J. and Rehg, J.M. (1999) Statistical Color Models with Application to Skin Detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (*CVPR*), **1**, 274-280.

[20] Khan, R., Hanbury, A., Stöttinger, J. and Bais, A. (2012) Color Based Skin Classification. *Pattern Recognition Letters*, **33**, 157-163. http://dx.doi.org/10.1016/j.patrec.2011.09.032

[21] Ibraheem, N.A. and Khan, R. (2012) Survey on Various Gesture Recognition Technologies and Techniques. *International Journal of Computer Applications*, **50**, 38-44.

[22] Mittal, A., Zisserman, A. and Torr, P.H.S. (2011) Hand Detection Using Multiple Proposals. *The* 22*nd British Machine Vision Conference*, Dundee, 29 August-2 September 2011, 1-11.

[23] Rashid, O., Al-Hamadi, A., Panning, A. and Michaelis, B. (2009) Posture Recognition Using Combined Statistical and Geometrical Feature Vectors Based on SVM. *International Conference on Image, Signal and Vision Computing* (*ICISVC*), London, 26-28 August 2009, 590-597.

[24] Suk, H.I., Sin, B.K. and Lee, S.W. (2010) Hand Gesture Recognition Based on Dynamic Bayesian Network Framework. *Pattern Recognition*, **43**, 3059-3072. http://dx.doi.org/10.1016/j.patcog.2010.03.016

[25] Lange, R., Dürr, F. and Rothermel, K. (2011) Efficient Real-Time Trajectory Tracking. *The VLDB Journal*, **20**, 671-694. http://dx.doi.org/10.1007/s00778-011-0237-7

[26] Rekha, J., Bhattacharya, J. and Majumder, S. (2011) Hand Gesture Recognition for Sign Language: A New Hybrid Approach. *The* 2011 *International Conference on Image Processing, Computer Vision, and Pattern Recognition* (*IPCV*), Las Vegas, 18-21 July 2011, 80-86.

[27] Elmezain, M., Al-Hamadi, A. and Michaelis, B. (2009) A Novel System for Automatic Hand Gesture Spotting and Recognition in Stereo Color Image Sequences. *Journal of WSCG*, **17**, 89-96.

[28] Wagner, D., Langlotz, T. and Schmalstieg, D. (2008) Robust and Unobtrusive Marker Tracking on Mobile Phones. *Proceedings of the* 7*th IEEE/ACM International Symposium on Mixed and Augmented Reality*, Cambridge, 15-18 September 2008, 121-124.

[29] Fuhrmann, A., Schmalstieg, D. and Purgathofer, W. (1999) Fast Calibration for Augmented Reality. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, London, 20-22 December 1999, 166-167.

[30] Kato, H. and Billinghurst, M. (1999) Marker Tracking and HMD Calibration for a Video-Based Augmented Reality Conferencing System. 2*nd IEEE and ACM International Workshop on Augmented Reality*, San Francisco, 85-94.

[31] Tuceryan, M. and Navab, N. (2000) Single Point Active Alignment Method (SPAAM) for Optical See-Through HMD Calibration for AR. *IEEE and ACM International Symposium on Augmented Reality*, Munich, 5-6 October 2000, 149-158.

[32] Lam, E.Y. (2005) Combining Gray World and Retinex Theory for Automatic White Balancein Digital Photography. 9*th International Symposium on Consumer Electronics* (*ISCE*), Macau, 14-16 June 2005, 134-139.

[33] Phung, S.L., Bouzerdoum, A. and Chai, D. (2005) Skin Segmentation Using Color Pixel Classification: Analysis and Comparison. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**, 148-154. http://dx.doi.org/10.1109/TPAMI.2005.17

[34] Farnebäck, G. (2003) Two-Frame Motion Estimation Based on Polynomial Expansion. *Lecture Notes in Computer Science*, **2749**, 363-370.

[35] Rabiner, L.R. (1990) A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In: Waibel, A. and Lee, K.F., Eds., *Readings in Speech Recognition*, Morgan Kaufmann Publishers Inc., San Francisco, 267-296.

Scientific Research Publishing (SCIRP) is one of the largest Open Access journal publishers. It is currently publishing more than 200 open access, online, peer-reviewed journals covering a wide range of academic disciplines. SCIRP serves the worldwide academic communities and contributes to the progress and application of science with its publication.

Other selected journals from SCIRP are listed as below. Submit your manuscript to us via either submit@scirp.org or Online Submission Portal.