Hand pointing detection system for table-top visual human-machine interaction

Peter Rulić, Alen Vrečko

University of Ljubljana, Faculty of Computer and Information Science
E-mail: peter.rulic@fri.uni-lj.si alen.vrecko@fri.uni-lj.si

1 Introduction

Visual communication plays an important role in human to human communication. Recently a lot of effort has been made, to extend this natural means of human to human communication, to human-machine interaction. Visual communication between humans is generally complicated because it involves various deliberate and undeliberate gestures with different meanings. One of the basic gestures in visual human communication is pointing. Pointing would also be a convenient means for communication between human and machine, because it would enable more natural and intuitive communication with the machine. The primary purpose of pointing is the determination of salient objects or regions in a scene. To introduce communication with pointing, we need a visual system for pointing hand detection and its parameter estimation. The vision system has to meet certain requirements to be usable in scenes where human machine communication is performed. In this paper we will describe the computer vision system for pointing hand detection in table-top communication scenes. We will address the requirements of such a system, describe an approach for visual processing, explain the visual communication protocol and discuss the integration of the pointing system into a larger cognitive system frmework. We will also implement the visual system for hand pointing detection and analyse its performance.

The computer vision system for pointing hand detection is included in EU FP6 IST Cognitive Systems Integrated project: CoSy - Cognitive Systems for Cognitive Assistants.

2 Human – machine communication system

The scenes for which we wish to build a visual human – machine communication system are general table – top scenes, where multimodal interaction between a human and a robot is performed. In the table-top scene the human and the robot are manipulating the objects present on the table. The robot is considered to be a play-mate to the human in some arbitrary table interaction game. The robot camera is positioned above the scene and directed at the table, to capture images of a table ground plane. The images of the table-top scene involve the table plane and arbitrary objects on the table, which can be manipulated by the human or by the robot. The manipulation of objects on the table involves placing, moving, removing and pointing at the objects. The purpose of the system for pointing detection is the determination of salient objects on the table. Object salience is important contextual information in the human-robot information exchange and often a useful disambiguation tool. By default the salient object is the last object added to the scene. The purpose of hand pointing is to shift the focus of the human robot conversation to an arbitrary object in the scene.

The pointing is in most cases simply performed by showing a closed hand, with the index finger positioned in the pointing direction. To model the pointing hand, we focus only on human actions on the table. We categorize human actions into two groups:

– Manipulating objects on the table, by moving them, placing them, etc.
– Pointing to the objects on the table by hand.

The computer vision system for the detection of hand pointing has to detect a human hand in the image, and has to distinguish pointing hand gestures, from other gestures that are result of other activities in the scene.

We name the system for pointing hand detection HandPointer. HandPointer is a part of a larger system named PlayMate. The PlayMate system incorporates many different sub-systems to form a multimodal cognitive robotic system for high level natural interaction with a human. The communication between HandPointer and PlayMate is bi-directional and is shown in figure 1. The PlayMate system provides the HandPointer system with an an image frame and data about all objects on the table-top scene that are available for interaction. HandPointer processes the given image frame to determine the following:

– presence of a human hand in the scene,
– presence of human hand pointing,
– details of hand pointing – position of hand, pointing direction,
– object in scene, to which human is pointing.

From the human point of view, the communication with pointing is simple and intuitive. The human just points to one of the objects on the table and receives confirmation about a salient object. The pointing protocol, includes action and conformation. When the action of pointing is detected, the conformation about object being pointed at is presented. The action is

performed by the human by pointing to an object on the table. The confirmation about the pointed object and all the consequent actions are performed by the machine. The confirmation is necessary feedback to the human, for comfortable communication with machine.

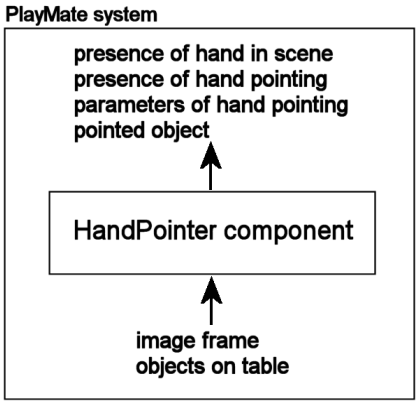

Figure 1: communication between system for pointing detection and PlayMate system.

3 Computer vision system

From the description of the human-machine communication system, we can conclude that the purpose of the computer vision system, is to detect the user's hand and recognize the pointing gesture to determine the object being pointed at. The human-machine interaction takes place in a real life environment, that is why we have to consider the following requirements:

- Robustness – the hand pointing detection should work reliably in arbitrary real life table-top environments with arbitrary types of scene illumination.
- Real time execution – the time delay between real life actions and obtained visual parameters should be unnoticeable, for smooth interaction between human and machine.
- Precise parameter estimation – the resulting parameters of the computer vision system should be estimated precisely to avoid errors in communication.
- Use of an arbitrary non-stationary color camera – the computer vision system should work with images, captured with fixed or mobile cameras.
- Ability to upgrade to new functionalities – the computer vision system should be flexible to accept different target objects, with the purpose to extend basic pointing detection capabilities.

With consideration of defined requirements, we take the approach for modeling of the pointing hand as defined in [1]. We divided the handpointing detection system into two image processing levels as shown in figure 2. The first level represents image segmentation. The second is the shape matching level. The computer vision system processes a single color image frame for desired results.

The segmentation level of the handpointing detection system is based on color context based edge detection defined in [2]. The target color for segmentation is human skin color. The segmentation is used for processing the input color image, and to extract contours of skin color, according to skin color parameters. The segmentation processing level is built of two steps. The first step is pixel based color segmentation. The second step represents edge detection on the color segmented image. With the first step, every pixel of the image is processed, to be classified in one of two color classes. These classes are a skin color class and a class of all other colors. If the color pixel value correspond to skin color, the pixel is classified as skin color pixel, othervise it is classified as a pixel of other colors. As a result of color pixel classification we get binary image, where white foreground represents skin pixels. Dark background represents pixels of other colors.

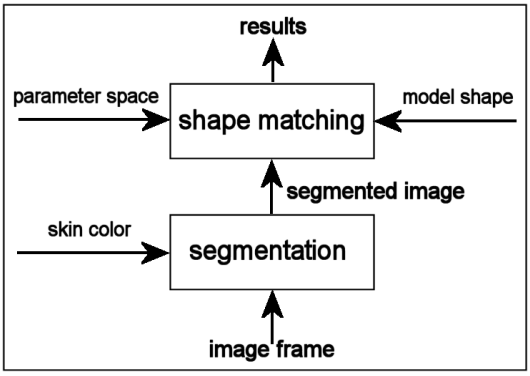

Figure 2: levels of computer vision system.

The advantage of pixel based segmentation is in its computational efficiency. Every image pixel is processed only according to its value. Its position in the image and relation to neighbouring pixels are irrelevant for classification. This is why the quality of segmentation can suffer, because the results can appear as perforated skin color segments and jagged segment edges.

The color segmented image is processed in the next segmentation step – edge detection. Detection of edges is performed by dividing the segmented binary image into 50% overlapped blocks. In blocks, the edges are approximated with a straight line. This approach reduces the effect of jagged edges and perforated segments.

The result of segmentation is a binary image, which represents contours of skin color objects. The goal of segmentation is the detection of the users hand shape in the image. Our chosen segmentation approach is based on color selective edge detection, which in contrast to general purpose non-selective edge detection such as

[3], enables detection of edges based on the target color. With this approach we wish to detect only contours of the user's hand, and to inhibit all disturbing contours in image. The main motivation for choosing such a segmentation approach, is high robustness and a computationally efficient implementation.

The second level of the computer vision system is the shape matching. We have chosen Hausdorff based, non-correspondence shape matching [4], because of its potential for high efficiency and robustness. The basic goal of the shape matching level, is to align the shape model with the segmented image in a such way, that the model elements are maximally close to the contours of the segmented image. The model is aligned to the image by setting up its parameters. In table-top scenes the pointing hand can be modeled in two dimensions, which define the table plane. The hand is present above the table-top and can be positioned in three dimensional space. Thus, we can model the pointing hand with four parameters: horizontal and vertical position, orientation and size. This combination of parameters form an affine transformation, with which the perspective transformation from the table-top scene to the image plane is approximated. For pointer only, three parameters are relevant: horizontal and vertical position and orientation. Size is not a relevant pointing parameter, but is necessary for modeling the hand. The hand moves above the table in three dimensions, that is why its image size changes when it is closer or further from the camera.

The alignment of the model shape and the contour image is performed in two steps, which form a hierarchical-genetic approach for shape maching. The first step represents testing of the shape model. The second step is used for fine alignment of model and image.

In the testing step the parameter combinations are generated by scanning the parameter space. Exhaustive scanning of parameter space would yield a high computational load. That is why the scanning is performed on down-sampled parameter space. The parameter combinations are generated with down sampled parameter instances. Every parameter combination is estimated with a metric, which measures the distance between the model and the image. The best estimated parameter combinations are taken to the next step of shape matching – alignment.

With the testing step of shape matching, we get a few roughly aligned model instances, which are fine aligned to the image in the alignment step. With alignment we obtain precise parameters of the pointing hand model, which are used for pointing.

The combination of context based edge detection with non-correspondence based shape matching, provides potential for robust and computationally efficient modeling of a pointing hand. It is also applicable for arbitrary geometric shape and arbitrary numbers of parameters. This makes this approach upgradable to new hand gestures and new functionalities in visual human – machine interaction.

4 Implementation and experiments

We have implemented a computer vision system for hand pointing detection - HandPointer as part of the Playmate system. The Playmate system is an instantiation of the CAS implemented in CAST [5]. The HandPointer has been implemented on Intel Pentium 4 3,2Ghz PC machine, with the Linux Ubuntu operating system. For implementation, the C++ programming language was used. The table-top scene was illuminated with non-standard illumination, which was a mixture of neon lighting and daylight. The images of table-top scenes were captured with a general purpose color Logitech QuickCam Pro 3000 camera, with a resolution of 640 columns and 480 rows.

We measured the frame rate for pointing hand detection, which was approximately 10 frames per second. The frame rate was high enough for real time operation. The sample images of the pointing model, input image, pointing segmentation image and resulting pointing model rendered back on the input image are shown in figure 3.

We also measured the average values of obtained pointing parameters: horizontal and vertical pointer position and orientation. The pointing precision is an importaint aspect of any pointing input device, because it determines its useability. To obtain the pointing precision, we performed a simple test. We pointed at an object with a still hand over an arbitrary period of time. Within this time period we observed pointing parameters. The number of consecutive frames captured within the pointing time was 36. The measured precision is presented with obtained average parameter values and their standard deviations over the 36 consequent frames. Quantitative results are presented in table 1.

Table 1: Results of measuring average hand pointer parameters from 36 consequent frames.

	Average value	Standard deviation
Horizontal hand pointer position in pixels	346,14	1,03
Vertical hand pointer position in pixels	142,58	3,53
Orientation of hand pointer in degrees	30,27	3,77

The average values in table 1 are hand pointing parameters: horizontal and vertical position, and orientation of pointer in image plane. Standard deviations represent the spread of hand pointing parameters around the average value. Since we obtained

parameters with still pointing hand, the standard deviation values can be interpreted as pointing precision. The ideal values for standard deviation would be 0. In this case the parameters would not change through consecutive frames. In practical cases the standard deviations are relatively small according to the image dimension.

One possible way of increasing the pointing precision is averaging values from many consecutive frames, instead of just taking values from a single frame. But this approach has a drawback of increasing the time delay for pointing action.

Figure 3a: geometry of a pointing hand model.

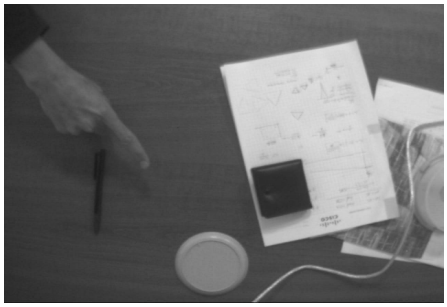

Figure 3b: input image.

Figure 3c: result of skin color selective edge detection.

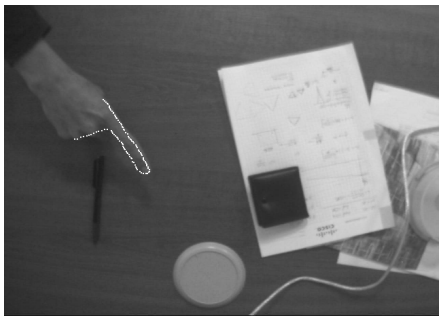

Figure 3d: result of pointing hand modeling. Detected pointing hand template is rendered on input image frame.

The performed experiment does not give us absolute insight about the pointing precision for arbitrary hand position and orientation in the scene. But it gives us a rough estimation of possible pointing precision. The obtained precision was good enough for successful pointing at arbitrarily large objects in a table-top scene, without averaging. Generally, pointing at the smaller objects is less successfull than pointing at larger ones.

5 Conclusion

We have implemented a computer vision system that can be used for human – machine interaction with hand pointing in real environment table top scenes. We have achieved satisfactory results regarding real time execution, and precision for interaction in real environments.

The main drawback of the system is robustness under changing lighting conditions. This is because the hand pointing vision system uses color as a basis for image segmentation. Changes in lighting conditions cause the changes in perceived color by camera, which influences the segmentation quality. Improvement of the system would be possible by improving the segmentation process to adapt to lighting changes.

The presented vision system is capable of modeling the pointing hand, and can be uprgraded for recognition of various hand gestures. The upgrade of the system is straightforward, by introducing corresponding geometric models to novel hand gestures.

Acknovledgement

This research has been supported by the following fund: EU FP6-004250-IP project CoSy.

6 References

[1] Peter Rulić, Alignment of geometric objects with image edge modelling, Doctoral dissertation, University of Maribor, Nov. 2006.

[2] Peter Rulić, Iztok Kramberger, Zdravko Kačič, Progressive method for color selective edge detection, *Optical Engineering*, vol. 46(3), pp. 037004-1 – 037004-10, Mar. 2007.

[3] John F. Canny, A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 8(6), pp. 679-698, Nov. 1986.

[4] Daniel P. Huttenlocher, Gregory A. Klanderman, and William J. Rucklidge, Comparing images using the Hausdorff distance, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15(9), pp. 850-863, Sep. 1993.

[5] Nick Hawes, Michael Zillich, Jeremy Wyatt, BALT & CAST: Middlevare for Cognitive Robotics, *Proceedings of IEEE RO-MAN '07*, pp 998 - 1003, Aug. 2007.