Touch- and marker-free interaction with medical software

T. Kipshagen[1*], M. Graw[1*], V. Tronnier[2], M. Bonsanto[2] and U.G. Hofmann[1]

[1] Institute for Signal Processing, University of Luebeck, Luebeck, Germany
[2] Department of Neuro-Surgery, University Hospital Schleswig-Holstein (UKSH), Campus Luebeck, Luebeck, Germany

*both authors contributed equally

We describe a touch-free input device for a medical data display application based on a hand gesture recognition system. The system is divided into the soft- and hardware for gesture recognition and a control plug-in for the open source DICOM viewer OsiriX. OsiriX is an image processing software dedicated to display data sets produced by medical imaging equipment or confocal microscopy. Hand gestures and 3D hand positions are recognized using two webcams in stereo configuration attached to a MacOS computer in a tower-like housing. Gathered pictures are subsequently transformed into interface commands controlling the OsiriX display, thus replacing completely the use of the computer's mouse.

***Keywords*- gesture recognition, touch-free interface, markerfree,**

I. INTRODUCTION

High quality interventional procedures in the surgical theatre are dependent on the availability and use of information from intraoperative imaging modalities. This can be done based on proprietary visualization tools or on open source driven projects.

OsiriX (www.osirix.org) is an open source image processing and 3D volume visualization project. It runs on MacOS X and is commonly used for planning and assisting medical procedures [9].

Unfortunately, its (and competitors') use in the surgical theatre is limited due to the need to either physically touch input devices (mouse, keyboard etc.) and thus compromise sterililty requirements or back-seat controlling the system computer by "proxy nurses". In any case, it means an unwanted interruption of the surgical workflow and additional distraction.

In the following, we propose and present a solution to this problem by interpreting and utilizing contact-free gathered hand gestures to control the OsiriX software (see Fig. 1). The use of hand gestures to control software has been subject to research in a variety of publications [5,6,7,12], but did not take the special requirements of medical procedures into account.

Our system is composed of an image processing and gesture recognition component and an OsiriX plug-in, running on an iMac (Apple, Inc.) housed in an OR usable tower chassis. The user's hand and its contour is determined automatically by image processing. Gesture specific features are extracted and used to replace the standard input device.

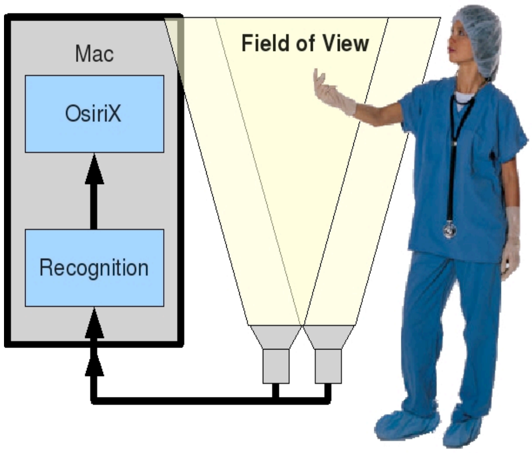

Fig. 1: Schematic drawing of our contact-free software control application.

The following report describes the project as it is currently evaluated in a neurosurgical operation theatre.

II. MATERIALS AND METHODS

A. *Hardware*

We utilize stereo-cameras to triangulate hand positions in 3D and map them to the 2D environment of the OsiriX application. To maintain a hygienically safe distance from the computer screen, two inexpensive webcams are utilized. Our current hardware setup consists of an Apple iMac with a 2.40 GHz Intel Core Duo Processor and 1.98 GB RAM and a screen-resolution of 1920x1200 pixels. Two Unibrain (Unibrain Inc., San Ramon, CA) Fire-i cameras (with a resolution of 640x480 pixels) are mounted below the screen on an adjustable camera rig. Thus, the cameras acquire pictures by looking up towards the ceiling of the operation theatre. An additional light-source may be employed to brighten up colors and is mounted below the cameras.

B. *Housing*

A 24" iMac is housed in an 190 cm * 66 cm * 16 cm wheeled aluminum chassis for easy transportation. The prototype is constructed from prefabricated *item* Building Kit System parts (item Industrietechnik GmbH, Solingen) and sealed by rubber fittings to foster hygienic cleaning of all surfaces (see Fig. 2) .

Cameras and additional illumination are mounted on an aluminum profile beam on the lower part of the tower and can be adjusted in height.

Fig. 2: WagO tower with Apple iMac running OsiriX.

C. *Gesture Recognition*

The gesture recognition component is based on the OpenCV image processing library [9].

Hand gestures are recognized in a processing loop divided into six main steps: image pre-processing, hand segmentation, post-processing, contour feature extraction, classification of extracted features and stereo-vision position calculation.

Each camera frame is processed independently, only step six requires information from both frames.

1. Pre-Processing: Gaussian filtering with 3 pixel radius removes noise from the frames. Images are rectified using the intrinsic and extrinsic camera-parameters and OpenCV's undistortion function.
2. Color-Segmentation is done assuming a Gaussian distribution of the skin-color (or color of a glove). To adjust this step for changing situations, an adaption step is performed: mean color and covariances within a calibration region are determined. To classify a pixel, the Mahalanobis distance [8, 13] to the mean value is calculated. Pixels with a small distance are classified as *hand*, pixels exhibiting a large distance are treated as *background*.
3. The segmentation result is post-processed by filtering to remove noise and close possible gaps in the tentative hand segment.
4. The arm is removed with a heuristic algorithm [2]. The contour is then sampled as a one-dimensional function. Sample points are used to calculate low-frequency Fourier descriptors [3], which define a unique feature-vector.
5. The feature vector is classified by a nearest-neighbor algorithm comparing it with reference vectors obtained from a database [11]. Thus, only specific gestures are allowed. The user can define these gestures and determine the specificity of the recognition.
6. The three-dimensional hand position is determined using the stereo-correspondence of a specified point of the hand. This can be either the center of the palm or the tip of a finger closest to the screen.

C. *Controlling OsiriX*

The plug-in receives a gesture and a position code from the gesture recognition component.

This code defines an interface event for the OsiriX application. User commands involve motion control in a 2D or 3D viewer: translation, rotation or zooming. These motions need a speed or/and direction. The hand position is used as a means to compute such parameters. If a new gesture is recognized the hand position is saved as a base position. As long as the same gesture is detected in the concurrent frame, the distance of the hand relative to the base position defines an offset. This offset is mapped to a velocity or direction parameter for the command associated with the gesture. For example, the *zooming* rate in the OsiriX viewer increases for an increasing distance of the hand to the base position. Moving the hand back to the initial position stops zooming.

Other events like selection of points in the image require an absolute hand position. For this purpose, the position code has to be transformed into screen coordinates, which requires a calibration to the specific hardware setup.

III. Results

The proposed setup with up-ward looking cameras and subsequent stereoscopic hand segmentation provided good results with and without artificial illumination as can be exemplary seen in Figs. 3 a) and b). To evaluate the results of the segmentation, we use normalized mutual information and a correlation coefficient between a manually segmented hand serving as ground truth and the automatically segmented one [1]. Combining the presented method with a simple post-processing step, we achieve a correlation of 0.94, and normalized mutual information of 0.82 between the calculated segmentation and the ground truth. A test set of 312 images was investigated.

Fig 3 a: Segmentation result, OR, artificial illumination.
a) Original image, b) manual ground truth, c) calculated segmentation

Fig 3 b: Segmentation result, OR, natural daylight
a) Original image, b) ground truth, c) calculated segmentation

Fig. 4. illustrates the currently implemented hand gestures to control the OsiriX Software:

- sliding through an image series
- rotating of 2D images and 3D reconstructions
- zooming in and out of images or reconstructions
- resetting the view.

Handedness is not an issue, since the utilized classification features are mirror invariant.

The system does not impose any special requirements on the user, lighting conditions and static backgrounds. Sleeves or gloves do not affect the gesture recognition.

The system can be extended with new gestures and is capable of close to real-time processing. Gesture classification and hand position are transmitted up to five times per second to OsiriX.

To validate the systems pointing precision, we presented 15 volunteers with the task to point their index finger at a computer generated cross-hair. Holding it at the position for 0,7 sec triggered a "mouse click" event to be analysed(www.albinoblacksheep.com/flash/shoot). Detected hand positions differed from real (screen generated) positions with less than 2 cm in 96%. Less than 1 cm of error was achieved in 50% of all cases. Thus, the recognition seems fairly exact.

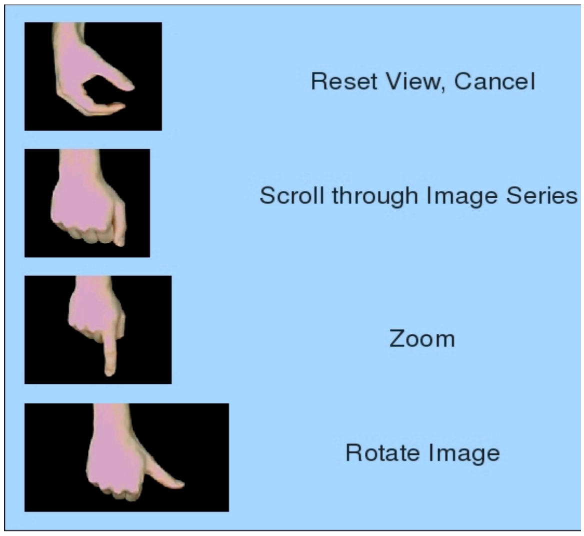

Fig 4: (left) Pictures of currently implemented and recognized gestures to control the OsiriX software. (right) OsiriX command replaced by hand gestures.

IV. Conclusions

The WagO system allows an easy to learn means of controlling the OsiriX application with hand gestures in a configuration suitable for medical settings.

Thus, a surgeon can interact with the OsiriX-software to view and manipulate medical images while performing surgery and sustaining sterility.

V. Discussion

We have presented a touch-free user interface for the OsiriX application. It is intended for surgeons to navigate image data while operating. Indeed, as the gesture recognition component is independent of the OsiriX plug-in, it can easily be integrated into other applications. The plug-in offers another advantage as it can be added to an existing OsiriX system.

Our future work will concentrate on validating its use under real surgical conditions and improving and extending its versatility.

To improve positioning we are currently experimenting with higher-resolution images and different camera positions.

Another goal is to enhance the classification of hand gestures by more sophisticated methods (e.g. Bayes classifier, neural networks) gaining a better recognition rate of hand gestures.

We plan to train further gestures and add them to the system, so the plug-in can be extended with additional commands. We are also working on recognizing dynamic gestures for an even more intuitive interaction with the computer.

VI. Acknowledgements

We acknowledge the invaluable help of the University of Lübeck's scientific metal workshop. Part of this work was funded by the Medical Faculty's grant for Biomedical Engineering.

V. References

[1] Baldi, P. *et al:* Assesing the accuracy of prediction algorithms for clasification: an overview In Bioinformatics Review, Vol. 16, No. 5, pages 412-424, Oxford University Press, 2000

[2] Deimel, B.: Entwicklung eines stabilen, videobasierten Verfahrens zur Segmentierung der Hand am Unterarm. Diploma thesis. University of Dortmund, 1998

[3] Dengsheng, Z., Goujun, L.: A Comparative Study on Shape Retrieval Using Fourier Descriptors with Different Shape Signatures. In Proc. of International Conference on Intelligent Multimedia and Distance Education. Pages 1-9, Fargo, ND, USA, June 1-3, 2001

[4] Jähne, B.: Digital Image Processing, 5th revised and extended edition. Berlin: Springer, 2002

[5] Kohler, M.: Technical Details and Ergonomical Aspects of Gesture Recognition applied in Intelligent Home Environments. Gesture and Sign-Language in Human-Computer Interaction. Lecture Notes in Artificial Intelligence 1371, pages 285-296. Berlin: Springer, 1997

[6] Licsár, A., Szirányi, T.: Dynamic Training of Hand Gesture Recognition System. ICPR 2004.Volume 4, pages: 971-974, 23-26 August, 2004

[7] Licsár, A., Szirányi, T.: Supervised training based hand gesture recognition system. Képfeldolgozók és Alakfelismerık III. Konferenciája, Domaszék, January 23-25, 2002

[8] Mahalanobis, P.C., On the generalised distance in statistics, Proceedings of the National Institute of Science of India 12, pages 49-55, 1936

[9] Open Source Computer Vision Library. http://www.intel.com/technology/computing/opencv

[10] OsiriX Medical Imaging Software. http://homepage.mac.com/rossetantoine/osirix/Index2.html

[11] Paliwal, K., Ramasubramanian, V.: Modified K−Means Algorithm for Vector Quantizer Design. IEEE Transactions on Image Processing, 1964

[12] Stark, M., Kohler, M.: ZYKLOP: Ein System für den gestenbasierten Dialog mit Anwendungsprogrammen. Modeling Virtual Worlds - Distributed Graphics. Berlin: Infix, 1995

[13] Vezhnevets, V., Sazonov, V., and Andreeva, A., A survey on pixel-based skin color detection techniques, in Proc. Graphicon-2003, September 2003.

Author: Ulrich Hofmann
Institute: University of Lübeck
Street: Ratzeburger Allee 160
City: 23562 Lübeck
Country: Germany
Email: hofmann@isip.uni-luebeck.de