PyGmI – Creation and evaluation of a Portable Gestural Interface

Matthias Schwaller
DIVA Group
University of Fribourg
1700 Fribourg, Switzerland
matthias.schwaller@unifr.ch

Denis Lalanne
DIVA Group
University of Fribourg
1700 Fribourg, Switzerland
denis.lalanne@unifr.ch

Omar Abou Khaled
University of Applied Sciences of Western Switzerland, Fribourg
1700 Fribourg, Switzerland
omar.aboukhaled@hefr.ch

ABSTRACT
The Portable Gestural Interface PyGmI, which we implemented, is a smart tool to interact with a system via simple hand gestures. The user wears some color markers on his fingers and a webcam on his chest. The implemented prototype permits to visualize and navigate into presentation files, thanks to a tiny projector fixed on the user's belt. The gesture recognition uses color segmentation, tracking and the Gesture and Activity Recognition Toolkit (GART). This article presents PyGmI, its setup, the designed gestures, the recognition modules, an application using it and finally an evaluation.

Author Keywords
Gestural Interaction, Portable User Interface.

ACM Classification Keywords
H5.2. Information interfaces and presentation: Input devices and strategies; Interaction styles; Natural language.

INTRODUCTION
Nowadays, laptops and smart phones are indispensable. These devices are not only used for work in the office but also to get information on the way. When a user would like to visualize information, he has to take the device out and navigate in the menus. The idea of this project is to simplify these kinds of tasks. Therefore we developed a smart tool which displays the information in front of the user either on a wall or on another object and the user can navigate using hand gestures. Since the tool is wearable, the user does not have to take it out to use it. A big challenge and limiting factor of wearable computing is the power supply [9]. This is especially a challenge if not only a PC is used but also a camera and a projector. For this reason an ultra mobile personal computer (UMPC) with a smart portable projector can be used. Both, the projector and the UMPC have their own batteries. A webcam was chosen for the camera since it uses a USB cable as a power supply. As a result, the prototype does not need any power cables, which makes the handling much more user-friendly.

We are planning to investigate in gesture recognition in our research lab. Therefore the PyGmI is our first step in a long process of interacting with hand gestures. In this project we use color finger markers. In the future, we would like to extend our system so that it does not need the help of markers.

The remainder of this paper is structured as follows: First we give an overview of some related work. Next we show the setup of the PyGmI prototype followed by the implemented gestures. Then, the gesture recognition and the prototype with an evaluation are presented, followed by conclusions.

RELATED WORK
This work was inspired by the concept described in the sixthsense project [6], which does not provide any information about how gestures are recognized or information about its reliability or evaluation. Our work implements a gesture recognizer which fulfills and extends sixthsense concept by allowing gestures to be used to exchange digital information.

The project Kinect (former Natal), form Microsoft Research[1] is one of the most popular gesture recognition systems even if it is not yet available at the moment of writing this paper. The Kinect system can be used to play on a video console without touching a controller. Contrary to Kinect, PyGmI does not track the whole body and favors precision for the pointing. Löchtefield et al. [3] developed the system ShelfTorchlight which helps users in search for book in a library or a product in the supermarket. The prototype uses a mobile phone and a mobile projector. This project uses a small projector like the PyGmI to display the information but it has no gestural interface. Argyros et al. [1] proposed a vision-based mouse. Their system recognizes 2D and 3D hand gestures to manipulate a mouse

[1] http://en.wikipedia.org/wiki/Kinect

NordiCHI 2010, October 16–20, 2010, Reykjavik, Iceland.
 ISBN: 978-1-60558-934-3...$5.00.

and they use one or two potentially movable webcams. A great advantage of their system is that they do not need any additional finger markers. Contrary to PyGmI, their system implements only gestures that can control mouse events.

SETUP OF PYGMI

The user communicates with the device via hand gestures. To do this, the user wears some color markers on his index fingers and the thumbs in order to facilitate detection of the fingers. These 4 fingers are the most important fingers for gestures. The PyGmI in use can be seen in figure 1.

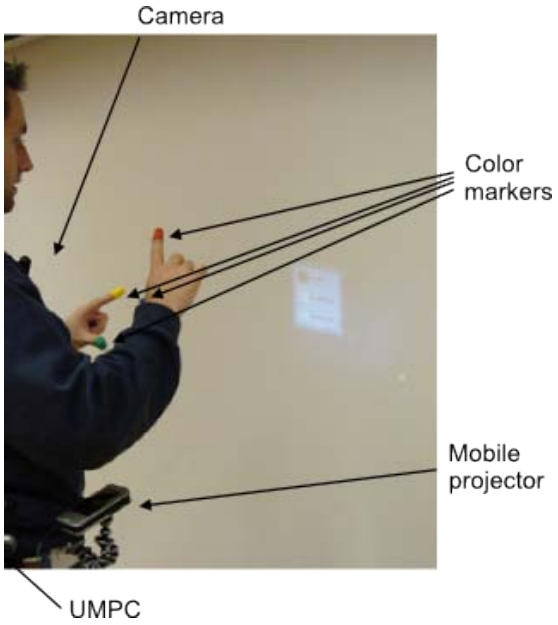

Figure 1. The Portable Gestural Interface in use

The PyGmI has to see what the user sees. For this reason the camera has to be placed close to the eyes of the user. One possibility is to place the camera on the head of the user. However, for the gesture recognition it is simpler if the camera is not moving that much, which are the reasons why we placed the camera on the chest close to the throat. Another disadvantage of placing the camera on the head is that the head may be slightly turned to the side and so the hands are not in the camera range.

To make information visible to the user a small projector is used. This allows using the tool either to augment the environment of the user by projecting on everyday objects or to use it on a wall to replicate a normal PC. Nobuchika et al. [8] analyzed several possibilities of where to put the projector. They advise using the lumbar mounted version. But, there are many other possibilities. Another would be to wear the beamer on the shoulder. This placement would be very unstable because the hand gestures also involve moving the shoulder. A further possibility is to wear the beamer on the chest, either flat on the chest with a mirror to change the projection direction or straight in front of the user. Since the chest mounted version can give shadows on the projected image we have chosen the lumbar mounted version.

HAND GESTURES

This section presents the gestures which are implemented for the PyGmI prototype. The gestures are intended to be used with the prototype for presentation visualization in PowerPoint. The first gesture is a deictic gesture while the other 5 gestures are metaphorical gestures. In the implementation process the difference is that for the deictic gesture there is no need for segmentation and identification (see figure 5). Since the system is made to be generic, it is easy to extend the system with additional gestures. The gestures we implemented have been chosen in order to fulfill the three dimensions presented by Lenman et al. [4]. First the gestures have to be easy to learn and remember ("cognitive aspects"). Second, the gestures have to be easy to perform ("articulatory aspects") and finally, the gestures have to be appropriate for practical use and fulfill the state-of-the-art of technology ("technological aspects").

The **gesture click** presented in figure 2 permits simply to click on a file to open it. The gesture can be either executed by the right hand or by the left hand. When the user wishes to grab an object he can use both hands and execute two click gestures at the same time. Note that a click contains a press down and a release. To move an object the user clicks down, moves the object and releases it.

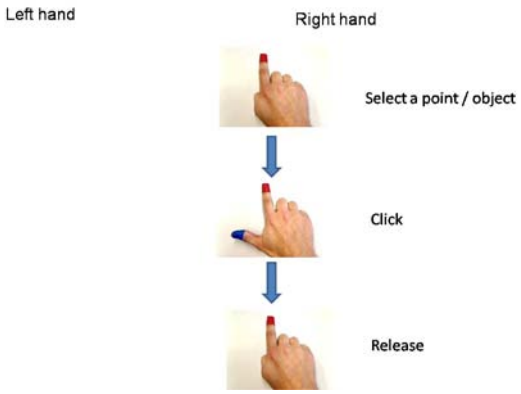

Figure 2. Gesture click

The **gesture open,** presented in figure 3, permits to put the PowerPoint in full screen mode.

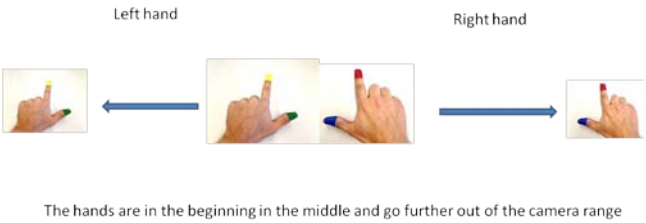

Figure 3. Gesture open

To close the full screen mode or the PowerPoint, the **gesture close** is used. It is the same gesture as "open" but in the inverse sense.

The two **gestures next and prev** are used to navigate in the slides. To perform these gestures the right thumb has to be in the camera range. The left index finger has to move at the same time from the left to the right in order to produce a "next" gesture and from right to left to produce a "prev" gesture.

To send a file to another person the **gesture send** is used. The user has to first "grab" the icon of the desired file and further to slide it out of the screen on the right side. First, the file has to be selected, in order to know which file has to be sent. Contrary to a drag and drop, the file has to be selected before the gesture. The reason for the selecting is due to the simplicity of the gesture recognition, since just the gesture is detected and not the location of the gesture.

Figure 4. The start of the send gesture

GESTURE RECOGNITION

The PyGmI is built in such a manner that it can be used with several applications. All the gestures detected are sent to registered applications. Therefore, the PyGmI includes an API which permits a third party application to listen for these gestures. In order to recognize gestures different steps and tasks have to be done (see figure 5).

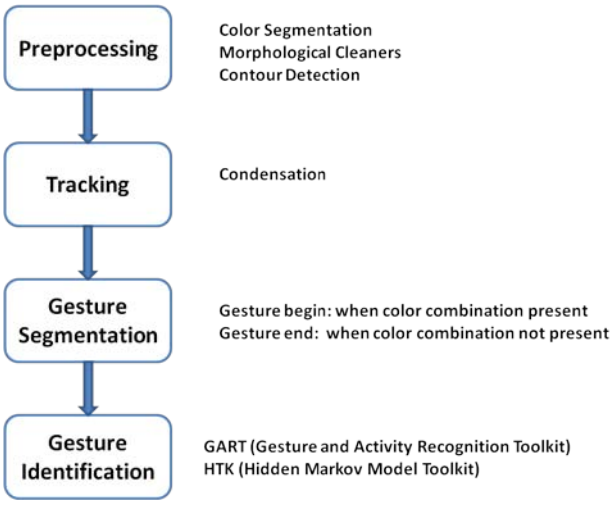

Figure 5. Overview of the Gesture Recognition

In order to be able to start the gesture recognition a **preprocessing** (see figure 5) is necessary. In the first part of the preprocessing the image is segmented in different colors. Second, a cleaning of the images is necessary to eliminate some irregularities in the edges of the color rectangles, through the morphological cleaning operators opening and closing. As a third step, the contours of the color rectangles are created. Since the color markers, which the user wears to make the gestures, are rectangular, the simplest and most efficient way is to detect rectangles. It is then possible to find the center of these rectangles. The coordinates of the centers are later used as input for the tracking and gesture identification.

The **tracking** part permits to track the different color rectangles and to predict their locations. The algorithm which is used for this is the Condensation algorithm [2]. This step is necessary to be able to predict, for instance, the location of a color marker if there is some light variation and when for one frame the color marker cannot be detected.

Since, the camera takes nonstop images, a way has to be found to detect when a gesture starts and when a gesture finishes. In other words there has to be a **gesture segmentation**. For this prototype the start is considered only when a color combination is detected. As soon as the color combination disappears, the gesture is considered to be finished. It is possible for instance, that in just one image a color has disappeared because of the light conditions. Therefore, a color is considered to be lost only after 5 frames. For the iterations before, the predictions of the tracking are considered as the locations.

The **gesture identification** part of the gesture recognition requires training of the different gestures in order to recognize them. Once a gesture is detected and identified PyGmI sends an event to inform the registered applications. For this training phase nine people performed the gestures on several backgrounds. There were 31 videos per gesture which were used for the training.

The system works with 20 fps and with a resolution of about 320 x 240. For the entire gesture recognition the libraries Emgu CV and OpenCV are used. The library used for the gesture identification is the Gesture and Activity Recognition Toolkit (GART) [5]. GART, which uses the Hidden Markov Model Toolkit (HTK), detects mouse gestures. For this project it was extended to 8 coordinates (2 per color). A limiting problem of GART is that the options to configure are only static (the same for each object of GART) and final (cannot be modified). They use a fixed number of HMMs which cannot be changed by the programmer. GART is very comfortable to start with gesture recognition but not flexible, which is why we will develop our own classifier in the future.

RESULTS AND EVALUATION

The gesture recognition rate was measured with the toolkit GART. 70% of the video recordings of gestures were used for the training and the other 30% for testing. We did several tests with numbers of gestures and also with the numbers of skip states, which can be configured in GART. The first test on the number of gestures was done with 14 videos per gesture. All of these videos were made by the

same person. We used 3, 4 and 5 gestures, which is the number of metaphorical gestures used by PyGmI. Surprisingly the maximum recognition rate was achieved with 5 gestures and was 85.94%. The second test was done with 31 videos per gesture made by 9 different people and here we evaluated the number of skip states. The number of gestures trained was 5. The number of skip states which was evaluated was going from 1 to 5 and the best rate was achieved by 4 and 5 skip states on which the recognition rate was 73.91%. In the second test the gestures were made by different people, which explain the lower recognition rate compared to the first test. We could not compare our results with the sixthsense project since they did not publish their results.

The PyGmI was integrated in an existing framework and a demonstration application was developed. For the demonstration application the framework InterFace [7] was chosen. This framework already includes several modalities and thus the integration of PyGmI was quite simple. The demonstration application permits to visualize a presentation using the portable projector. The metaphorical hand gestures permit the navigation in the slides, opening presentations in full screen and closing presentations. Finally, the presentations can also be sent to another device via a gesture. The deictic hand gesture allows navigating in a file browser and the use of the other plug-ins of the framework. Several people tested PyGmI. They found the system quite easy to use for scrolling through a presentation. No controlled user experiment has been done at the time of writing.

CONCLUSIONS AND FUTURE WORK

Since the user has to make his gestures in front of the camera it is possible that only a part of the gesture is in the camera range. Therefore, the user has to train at the beginning in order to be able to use the system. An extension which has to be made is the feedback on the projection. This will improve the usability since the user will be able to see if his fingers (color markers) are in the camera range.

An improvement also has to be done on the hardware part of the system. At the moment a UMPC is used which is worn on a belt. In the future, it would be nice to use a smart phone which can be stored in the pocket of the user's pants. The projector which was bought at the beginning of the project was one of the first portable projectors. Its disadvantages are that it is not bright enough to use in daylight and that the autonomy is only one hour.

Since the PyGmI is our first attempt to solve gesture recognition using image processing we plan to improve the gesture recognition in several ways. First, we intend to create our own classifier since our recognition rate is not close enough to 100 and GART not flexible enough. Second, we plan to capture 3D gestures using a time of flight camera (TOF) which will free users from color markers. Third, we plan to recognize gestures as a way to interact with a large vertical surface. Therefore, we will rethink the needed gestures, which is why we recently did a wizard of oz experiment in order to identify which gestures are best for users. Once several systems have been developed we will be able to compare them. The aim will be to create an efficient system to detect ergonomic hand gestures without any help of markers.

REFERENCES

1. ARGYROS, A. A. AND LOURAKIS, M. I. A. 2006. Vision-based interpretation of hand gestures for remote control of a computer mouse. In *Computer Vision in Human-Computer Interaction*. Springer-Verlag, 40–51.
2. ISARD, M. AND BLAKE, A. 1998. CONDENSATION - Conditional Density Propagation for Visual Tracking. *Int. J. Comput. Vision 29,* 1, 5–28.
3. LÖCHTEFELD, M., GEHRING, S., SCHÖNING, J., AND KRÜGER, A. 2010. ShelfTorchlight: Augmenting a Shelf using a Camera Projector Unit. In *Adjunct Proceedings of the Eighth International Conference on Pervasive Computing*.
4. LENMAN, S., BRETZNER, L., AND THURESSON, B. 2002. Using marking menus to develop command sets for computer vision based hand gesture interfaces. In *NordiCHI '02: Proceedings of the second Nordic conference on Human-computer interaction*. ACM, New York, NY, USA, 239–242.
5. LYONS, K., BRASHEAR, H., WESTEYN, T. L., KIM, J. S., AND STARNER, T. 2007. GART: The gesture and activity recognition toolkit. In *HCI (3)* (2008-12-04), J. A. Jacko, Ed. Lecture Notes in Computer Science, vol. 4552. Springer, 718–727.
6. MISTRY, P., MAES, P., AND CHANG, L. 2009. WUW - Wear Ur World: A Wearable Gestural Interface. In *CHI '09: Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*. ACM, New York, NY, USA, 4111–4116.
7. MUGELLINI, E., ABOU KHALED, O., PIERROZ, S., CARRINO, S., AND CHABBI DRISSI, H. 2009. Generic Framework for Transforming Everyday Objects into Interactive Surfaces. In *Proceedings of the 13th International Conference on Human-Computer Interaction. Part III*. Springer-Verlag, Berlin, Heidelberg, 473–482.
8. SAKATA, N., KONISHI, T., AND NISHIDA, S. 2009. Mobile Interfaces Using Body Worn Projector and Camera. In *VMR '09: Proceedings of the 3rd International Conference on Virtual and Mixed Reality*. Springer-Verlag, Berlin, Heidelberg, 106–113.
9. STARNER, T. 2001. The Challenges of Wearable Computing: Part 1. *IEEE Micro 21,* 4, 44–52.