Multi-user vision interface based on range imaging

Tobias K. Kohoutek*
ETH Zurich – Institute of Geodesy and Photogrammetry
Wolfgang-Pauli-Strasse 15, 8052 Zurich, Switzerland

ABSTRACT

For common computer interaction the mouse is established as a standard device. The recognition of freehand 3D-interaction has already been implemented by detecting the fingertips and the eyes of the users. This application is based on the stereo-photogrammetry approach with two webcams. Attempts with a single webcam have been performed as well to quit the synchronisation of two video streams. Using the range-imaging technology the user can move in front of the display from 30 cm up to the maximal ranging distance that is supported by the camera. The body, especially head and hand, can be detected in 3D within the operating range and an additional gesture-analysis tool is able to interpret the commands of the user. With this approach, the computer mouse is not needed anymore. The main topic of this paper is the multi-user interaction. Operating the computer at the same time with several users is not supported by the actual operating systems. The simultaneous detection of several users and their hands in 3D was achieved. A fast switching between the users to control the computer in turns is explained.

Keywords: 3D-tracking, real-time, range imaging, human computer interaction, multi-user interface

1. INTRODUCTION

Computer screens and displays have become larger in the past, also digital projectors have been used much more often. This has inspired research to use other input devices then keyboard and mouse. Since the early 1990's much research has been carried out in Human Computer Interaction (HCI). According to Erol et al.1, the human hand is the most effective interaction tool for HCI. Gorodnichy et al.2 carried out research for completely hands-free interaction that requires motion measurement and tracking of various human body parts. Most research in computer vision uses stereo photogrammetric approaches1,3,4 or the single camera (webcam, digital video cam) approach5,6. On the other hand there are some less known methods. Oka et al.7 were using an infrared camera to detect areas close to the temperature of the human body. Banker et al.8 designed a 3D computer mouse while using an ultrasonic transmitter. The active triangulation principle by Malassiotis & Strintzis9 is based on the coded light approach for 3D data acquisition.

Besides the detection of the users head a detection of the users “pointing device”, the hand/fingertip, is required in many applications. Nickel & Stiefelhagen10 explain the importance of pointing device detections with: “Humans perform the arm towards a pointing target in the communication with others to mark a specific object, location or direction.” The 3D geometry and the gesture in natural arm movements has to be detected and the pointing direction has to be estimated. Consequently several problems have to be solved to implement a virtual mouse.

The acquisition and distance measurement is realized for each individual pixel by a range imaging camera exploiting the time-of-flight (TOF) principle. Objects in a scene reflect the emitted light pulses back to the camera, where their time of arrival is measured. A range imaging camera combines the benefits of single and multi camera systems. Range imaging enables to measure the user's position in 3D using only one camera.

The second step is to create a multi-user interface by upgrading the existing image processing algorithms from a one user to a multi user system independent from the users' movements. However, it is not possible to operate with multi users at the same time. Due to the limitation of the operating systems (MS Windows, Mac OS, Linux) only one pointing device can be implemented. The advantage is to switch among the several users who were detected during image processing.

* kohoutek@geod.baug.ethz.ch, www.geometh.ethz.ch

Proc. of SPIE Vol. 7341 73410M-1
2. RANGE IMAGING

While using a single camera, the distance between the user and the camera/display is determined easily by the users head size\textsuperscript{11}. In contrast, approximation of the distance between the users hand and the screen/camera remains a different task. Stereo or multi vision photogrammetry acquires a 3D-model but analyses and matching of multiple video streams is needed. Common methods to measure 3D are stereo triangulation, sheet of light triangulation, structured light projection or interferometry. Range imaging however uses the time-of-flight principle.

There are two different ways to measure distances using TOF. One method is the time measurement of a laser pulse, that is reflected at an object. This principle is used by lasercanners. Another type of TOF is working with a modulated infrared light source to measure the phase delay as it is done in this work. The accuracy of the distance measurement is the limiting factor in HCI for computer vision.

Range imaging cameras provide real time distance data at video frame rates up to 50 frames per second. They acquire an amplitude image and a range image. The local brightness as well as the distances for every pixel is coded in 16 bit gray values. For each pixel the distance is measured directly by calculating the phase shift between the emitted and reflected signal. The phase map and finally a complete distance map can be acquired by detecting the phase delay between both signals\textsuperscript{12} as it is shown in Fig. 1a). By sampling this signal the three unknown parameters of the modulated signal in Fig. 1b), the amplitude $A$, the offset $I$ and the phase $\varphi$ can be determined by the following equations (1) to (3), where $m_1$...$m_4$ are the measured phase delays.

$$\varphi = \arctan \left( \frac{m_4 - m_2}{m_1 - m_3} \right)$$

$$A = \sqrt{\frac{(m_3 - m_1)^2 + (m_4 - m_2)^2}{2}}$$

$$I = \frac{m_1 + m_2 + m_3 + m_4}{4}$$

With (4) the distance $D$ and the achievable depth resolution $\Delta D$ can be calculated.

$$D = \frac{D_{\text{max}} \cdot \varphi}{2\pi} \text{ with } \Delta D = \frac{D_{\text{max}}}{\sqrt{8}} \frac{\sqrt{I}}{2A}$$

The maximal operating range of the camera is represented by $D_{\text{max}}$. Every object which is further away will be shown in a wrong gray value due to the repetition of the modulated signal. State of the art range imaging cameras operate with a wavelength of 850 nm and a modulated frequency of 25...30 MHz. The current array size dithers between 60 x 60 up to 205 x 205 pixels.
3. VIRTUAL TOUCHSCREEN

In a first step the camera position has to be related to a large screen. Usually, the camera is placed on the top of the screen. Fig. 2. shows that a view frustum between the user and the display is flexible while the user moves in the operating distance of the range imaging camera. While using this technology, the frustum will be adjusted to the user's body and get re-adjusted with the user's movement in real time. A virtual touch screen appears in front of the user by pointing to the display. The virtual screen will always be within the frustum and is adjusted to the users arm length.

3.1 Hand-Head Line Model

People tend to look towards an object and use one finger to point in the direction of the object when they show it to somebody else. The extension of the finger and the eyes are in the line of gaze with the object, shown in Fig. 3. The same happens here with the display and the user. All objects on the screen are not touchable from the distance but at the fingertip the virtual display will be placed. This is done by checking the range image for the closest point to the camera by histogram evaluation. Remember, that the distance is stored in a 16 bit gray value image. While the fingertip will always be the closest point to the camera, the gray value will be an extreme of the image values. The minimal gray value presents the fingertip of the user.

Pointing to the four corners of the screen is one possibility to calibrate the system. This way the view frustum becomes adjusted to the users arm length. Because of the range imaging technology the user is now able to move forward or backward, left and right in front of the display. By pointing towards the display the mouse pointer is now driven by the fingertip. The fingertip interaction, left click, right click and mouse wheel scrolling are parts of the gesture recognition but be beyond the scope of this work. There are several papers deal with gesture recognition.

Fig. 3: Hand-head line model (modified from 11)

$P_e = (X_e, Y_e, Z_e)$

$P_o = (X_o, Y_o, Z_o)$

$P_p = (X_p, Y_p, Z_p)$

$P_e = \text{point at the eye}$

$P_o = \text{a point at the object}$

$l_g = \text{line of gaze (from eye to object)}$

$l_r = \text{line of ray (e.g. arm direction)}$

$P_p = \text{pointing marker (e.g. fingertip)}$
4. MULTI-USER INTERFACE

As previously mentioned, the detection and tracking of a single user in front of a camera has been achieved already in many different works. The challenge here is to generate multi-user interface functionality by 3D image processing. The idea for multiple users on a computer screen originates from a blackboard where several users can write or draw simultaneously. This work will show the possibility of detecting and tracking multiple users in 3D. The used range imaging camera SR-4000 is developed by MESA® Imaging. Its pixel array size is 176 x 144 working with a wavelength of 850 nm and a modulated frequency of 30 MHz. The operating range with standard settings is 0.3 to 5.0 meters.

The complete source code is written in MS Visual Studio 2008 C++ enhanced to the offered library files for camera acquisition and image processing. In a first step a real time face tracking with scale invariance could be realized. Therefore the Haar-Face detection method was used. The Haar-Like feature classifier is given by the Open Source Computer Vision Library OpenCV18. This algorithm uses a face template to match it with the users face. The mimic of the user has no influence to the template matching. Not yet implemented is a real time tracking of the eyes such as Savas9 provides. Tracking the eyes is important when the users are close to the display (<1.5 m). In this case, the head-line is established between eyes and hand. Actually, the mid-eye position and the fingertip build the line of gaze.

Normally one computer is operated by a single user who only needs one keyboard and one mouse to control the system. Currently a second pointing device is not supported by the operating systems of today. A second course can not appear on the screen by simply connecting a second mouse with the computer. It is only possible to move one screen course with both devices. In a multi-user application there are at least two users (A and B). The face of user A has to be detected as well as the one of user B. Also both mid-eye positions have to be calculated. Each position has his own view frustum towards the display. By detecting the fingertips of A and B there will be four lines of gaze. Of course, only two of them are correct. Fingertip of user A has to be connected with the mid-eye position of user A. The same has to be done for user B. If fingertip A is combined with mid-eye B or vice versa then a wrong object in- or outside the view frustums is pointed at. The correct combination between the hands and the heads of user A and B is given by the bodies of the users. A fingertip is always connected to a hand, that is connected to an arm, that is connected to the body where the user's head is is on the top. A 3D body tracking20 is necessary to fix the users lines of gaze. All necessary information can be obtained from the range image.

The screen course is calculated where the line of gaze of a user intersects with the display. Assuming that user A is controlling the operating system, the arm and fingertip of user B should not be raised. If user B wants to control the system now he has to rise up his arm not before user A brought his arm down. This order is crucial to pass the pointing device on user B. The reason for that order is that both users do not have to stay in the same distance to the display. If user A would be closer to the camera he would stay in front of user B, even when user B would lift his arm. The fingertip detection is based on the definition of distance regions. The minimal distance obtained by the minimal gray value in the range image represents the finger of the user. If user A is closer to the display the minimal gray value is represented by the body of user A. To solve this problem a threshold is used. A body is represented by more pixels then a hand or fingertip. The solution is to count the pixels close to the camera. If there are more pixels then the defined threshold then the detected object can not be the fingertip. The algorithm has to start again and search for the next points close to the camera. While head and hand of user B are detected he will be the new system operator.

5. RESULTS

It can be observed from Fig. 4 that the distance information is captured by the range imaging camera for each pixel. The closest point to the camera is the pointing finger towards the screen. The detection algorithm was successful. The face detection performs best between 130 cm to 400 cm. If the user is too close to the camera too much reflected light gets on the sensor, and if too far away, the face is to small for the detection. Fig. 5 shows, that the face detection algorithm lost the tracking of a head when the user turned around because the used algorithm is not rotation invariant and the template for the matching requires a face to be oriented towards the front. This phenomenon occurred for larger distances even under small rotations. Solving this problem will be a part of future works.
Fig. 4: 3D scene (mirror-inverted) and top view to the same scene

Fig. 5: face detection in distance of 3 m
faces towards the camera (left), one face with small rotation (right)
6. CONCLUSION

A set of experiments with real-time head and hand tracking was performed. Volunteers were asked to move their heads in a natural way and point one by one towards the display. The fingertip was found successfully for every user. The screen cursor was set to the position where the line of gaze intersects with the display.

Another challenge is the gesture recognition with the range imaging camera. For instance, a fast forward-backward motion could be implemented as a click. In order to enable this, the distance measurement accuracy in the range image has to be increased in order to adjust of the view frustum more accurately. However, a simultaneous interaction of multiple users could not be realized because of the limitation of pointing devices by the operating systems. A fast switching between the users has been found to be the only possibility to allow several users operating the same computer simultaneously.

REFERENCES