Hand Gesture Recognition with a novel IR Time-of-Flight Range Camera – A pilot study

Pia Breuer[1], Christian Eckes[2], and Stefan Müller[3]

[1] Universität Siegen
pbreuer@informatik.uni-siegen.de
[2] Fraunhofer Institute Intelligente Analysis- and Informationsystems IAIS
christian.eckes@iais.fraunhofer.de
[3] Universität Koblenz-Landau
stefanm@uni-koblenz.de

Abstract We present a gesture recognition system for recognizing hand movements in near realtime. The system uses a infra-red time-of-flight range camera with up to 30 Hz framerate to measure 3d-surface points captured from the hand of the user. The measured data is transformed into a cloud of 3d-points after depth keying and suppression of camera noise by median filtering. Principle component analysis (PCA) is used to obtain a first crude estimate on the location and orientation of the hand. An articulated hand model is fitted to the data to refine the first estimate. The unoptimized system is able to estimate the first 7 Degrees of Freedom of the hand within 200 ms. The reconstructed hand is visualized in AVANGO/Performer and can be utilized to implement a natural man-machine interface. The work reviews relevant publications, underlines the advantages and shortcomings of the approach and provides an outlook on future improvements.

Keywords: gesture recognition, range camera, PCA, segmentation, human computer interaction, AVANGO, SGI Performer, computer graphics, computer vision, model based matching

1 Introduction

Human computer interaction (HCI) plays a important role in many applications today. It exists the desire to realize more natural forms of interaction between humans and machines. One of these forms is the interaction via gestures, e.g. by recognizing natural hand- or full-body movements. Many researchers have focused on measuring the articulation of the whole body neglecting the hand posture since full-body posture is in itself already a complex task, even when problems due to clothing, self-occlusion and non-rigid tissue can be avoided. On the other hand, the human hand has more that 24 degrees-of-freedom in finger/thumb articulation and forms a very difficult problem on its own. Solutions to both problems have been developed but they violate an important aspect in HCI, which is called “come as you are”. This paradigm is one of the most influential goals in HCI but also the most difficult one to achieve: The intentions of the human user must be recognized effortlessly and non-invasive, e.g. without attaching any special hardware to the human body. Data gloves, suits, inertial tracking devices and optical markers attached to the hand, face or body of the human user must be avoided.

Hence, we present in this paper an approach for non-invasive gesture recognition in real time. We focus on making the hand and its movements useful for human computer interaction. We regard hand gesture recognition as a composition of two separate problems, namely reconstruction and recognition. Reconstruction deals with the problem of inferring the various degrees-of-freedom of the hand from sensor data, e.g. by matching and aligning domain knowledge in form of an articulated 3d-model with sensed data. The second step, recognition, classifies the gesture as belonging to an instance stemming from a predefined set of relevant gestures. There exist many well-known methods for classifying time-dependent data stemming from the reconstruction, for instance methods based on hidden Markov models (HMM) are the most common ones. We consider the problem of recognition as solved (or at least, as easier) and focus in our study on the problem of reconstruction, instead.

We decided to investigate the usefulness of the new Swissranger SR2 miniature time-of-flight range camera for gesture recognition, as this new hardware exhibits significant advantages over more traditional type of camera sensors: It is able to deliver range data in real time, namely with a framerate of 30 Hz without the need to solve the correspondence problem, as it is mandatory in depth estimation by means of stereo triangulation. Furthermore no special background or skin color model is required for the segmentation of the hand, so it can be used at many different places. The camera is based on active IR illumination and is therefore more robust against illumination changes than an optical hand localization based on skin color would be. However, despite these advantages it remains to be investigated in this study how useful the Swissranger SR2 really is for gesture recognition, where its weaknesses and limits are and how these problems might be investigated by further studies.

After reviewing some relevant previous work (Sect. 2) this paper presents the working principle of the range camera (Sect. 3.1). This is followed by explaining in detail how the camera is calibrated (Sect. 3.1). The following sections present our approach to reduce sensor noise (Sect. 3.1), how the hand can be segmented from the background (Sect. 3.1) and we show how a first estimation of the hand pose can be obtained from a PCA (Sect. 3.2). The next sections deal with the model based fine-matching (Sect. 3.2), present the results (Sect. 4) and conclude this paper with a final discussion. Let us now start with Sect. 2 by looking at some related work.

2 Previous work

A common method for real-time range-scanning is the use of structured-light. Hall-Holt and Rusinkiewicz [1] use a standard video camera and a DLP projector to produce range images at 60 Hz. Infra-red structured light pattern together with stereo reconstruction are used by Ypsilos et al. [2].

One of the first who worked with range data for hand gesture recognition were Malassiotis et al. [3]: 3D information was acquired following a structured light approach with a frame rate of twelve image pairs per second. But they did not use range data exclusively, since the hand is segmented from the background by means of skin color segmentation. Bray et al. [4] applied a similar method by using also structured light to generate depth maps in combination with a skin color model to eliminate the background. Perrin et al. [5] worked with a target-oriented laser beam and two mirrors to perform 3D tracking. The authors determine distance measurements by using the absolute value of the reflected light but the system cannot be used in real time.

Using IR range data from a time-of-flight camera in gesture recognition is relatively new. One of the few authors following this approach are Liu and Fujimura [6]. Their system works in real-time, but it uses the range data only for segmentation to cut out the region nearest to the camera. The model parameters are estimated by doing shape comparisons on basis of 2D images.

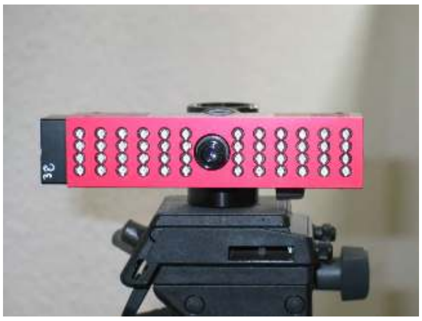
Our work is also motivated by the project "Virtual Cyberglove" of the Computer Graphics Group of the University of Bonn (cp. [7] and [8]). This project aims at estimating global position and all degrees of freedom of a hand (e.g. of the back of the hand) by using multiple video cameras.

A careful selection of the hand model is also relevant for efficient hand gesture recognition. We use the model developed by [9] which aims at achieving an anatomically correct reconstruction and animation of human hand.

3 Gesture recognition with the Swissranger SR2

3.1 The Swissranger ToF camera

Hardware, ToF principle The infra-red time-of-flight range camera Swissranger SR2 was developed by CSEM ([10] and [11]). It does not only measure local brightness, but a complete distance map. Both range images and grey scale images are delivered via USB 2.0 interface to the host PC.



resolution	max. 124x160*
depth resolution	up to 5mm
wavelength	870nm
luminance	800mW (optical)
max. distance	7,5m
frame rate	up to 30fps**
lense	f = 8mm, F/# = 1.4, M 12x0.5
proportions	135(w)x45(h)x32(d)
weight	0.2kg

* depending on ROI
** depending on camera settings

Figure 1. Swissranger SR2 and its technical specification (taken from [12])

The emitter sends out a radio-frequency modulated light field, which is reflected by the scene and sensed by the camera. The distances between the camera and the scene objects are computed by using the time-of-flight principle, e.g. by measuring the time the light needs for reaching the objects and becoming reflected back to the sensor of the camera.

$$TOF = \frac{2D}{c} \quad \text{with speed of light } c = 3 \cdot 10^8 m/s$$

Instead of emitting impulses, the SR2 sends out continuously modulated light waves with a modulation frequency of 20 MHz. Demodulation is done by sampling. The incoming signal gets sampled four times per period (each sample shifted by 90°), whereby the signal can be clearly reconstructed. Every sampling corresponds to an integration of the photo generated electrons over the relevant part of the period. To reduce the signal-to-noise rate the integration values are summed up over many periods. So not the time is measured but the phase shifting between emitted and detected wave. The ambiguity distance is $7.5m$, before a phase shift of 2π is reached.

Hence, the distance is computed according to

$$D = L \cdot \frac{\varphi}{2 \cdot \pi} \quad with \quad L = \frac{c}{2 \cdot f_m}$$

$\varphi = \arctan\left(\frac{c(\tau_3)-c(\tau_1)}{c(\tau_0)-c(\tau_2)}\right)$ phase shift
L: unambiguous distance range
c: speed of light
f_m: (radiofrequency) RF modulation frequency

This algorithm is called four buckets or four steps method because of sampling four times each period. More detailed information about the camera can be found in [13] and [14].

Calibation A comprehensive calibration turned out to be very difficult (cp. [15] and [16]). Gut [17] even talks about the necessity of multiple calibration. We decided to do it as follows: To analyze the output of the camera, we made a experimental setup composed of a container with fixed wheels and two books placed at different distances. We established a z value of 8000 from the camera corresponding to a range of 1000mm.

To convert the pixel values (u,v) to a position in the real world depending on the range, we use the following formula:

$$\boldsymbol{x} \equiv \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} u \\ v \\ 0 \end{pmatrix} + \begin{pmatrix} -u \\ -v \\ f \end{pmatrix} \cdot \frac{d}{\sqrt{u^2 + v^2 + f^2}} \tag{1}$$

$(x\ y\ z)^T$: 3D coordinates
$(u\ v)^T$: camera coordinates
f: focal length
d: measured range

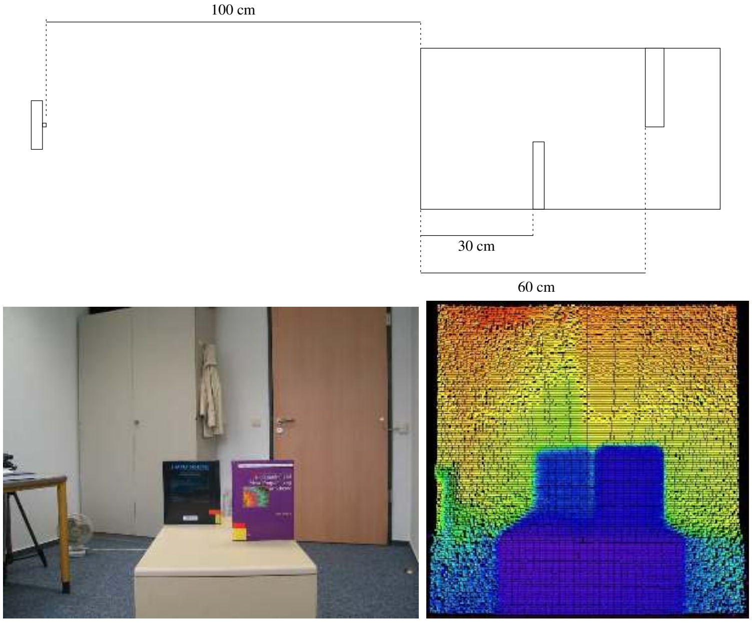


Figure 2. experimental setup (here at a range of 1.0m)

It can be derived from Fig. 4 (taken from [15]) via the theorem on intersecting lines.

For efficiency reasons the values of the square-root are computed only once and stored in a look up image with the size of the sensor. It depends only on the (u,v) positions and the focal length, which do not change over the time.

Noise suppression There is a significant amount of noise present in the sensor data, as Fig. 3 reveals. We are able to reduce speckle noise significantly by applying a median filter to the range image. This method, however, cannot suppress the most annoying reason for incorrect measurements, which give rise to the so called "tail of a comet": The surface near the object border seem to meld with the background, precise measurements near object borders become impossible. One reason for this systematic errors lies in the limited resolution of the sensor chip: Each pixel has to average over background and object at the borders because both photons from the background and photons from the object are collected by the same pixel. Moreover, the incident angle reaches $\pi/2$ near the object borders and results in a significant reduction of the number of reflected photons.

Depth segmentation Arm and hand are segregated from the background by simple depth keying, e.g. we define a region of interest (ROI) in depth and discard any measurements which do not fall within the predefined range of depth.

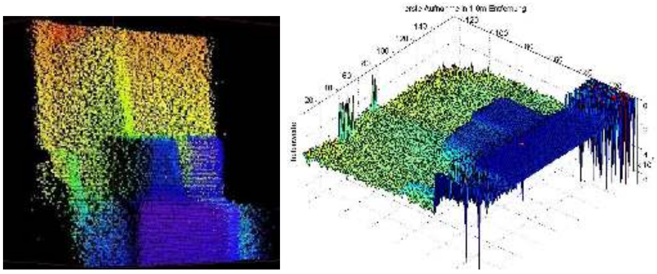


Figure 3. visualization of the measured data (here at a range of 1.0m)**:** representation with SR2-Tool (left) with MatLab (right)

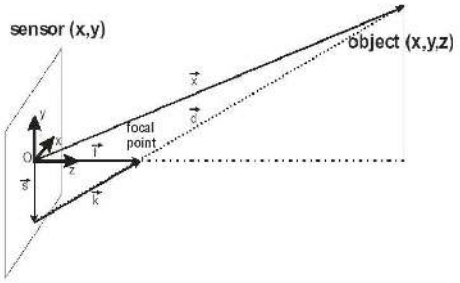


Figure 4. model to determine the coordinates [15]

The depth range defines also a working point for our recognition system as it marks the optimal trade off between two conflicting properties: The hand should cover as much area of the camera sensor as possible, but any overexposure must be avoided. Overexposure results in serious errors as it impairs the depth measurements of the complete scene significantly. The last image in Fig. 5 shows the effect: The background in the lower area seems to be nearer to the camera compared to the initial state (image at the upper middle), but has in fact not moved at all. Due to this effect, there exist an optimal distance between hand and camera for which the depth measurements works best. For these purposes we scrutinized the correlation between distance, sensor coverage and overexposure and analyzed measurements of a hand at different distances. Fig. 5 shows a selection of the results.

Overexposure can also be reduced by decreasing the integration time. We use a depth range between 0.5m and 1.0m, which correspond to z-values of the camera between 4000 and 8000. We solve the segmentation problem by performing connected component analysis and selecting the largest remaining connected sub-segment as input to the reconstruction methods.

3.2 Measuring hand geometry

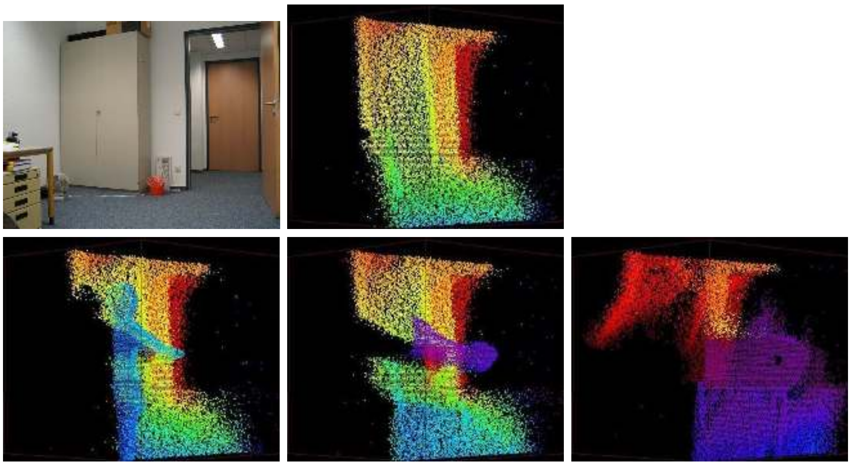


Figure 5. selection of the results visualized with the SR2-Tool: The first row shows the initial state (middle: visualization of the data as points; left: photo of the scene for comparison). The second row shows the data when the hand is located at distances of 2.0m, 1.0m and 0.5m, respectively. The left image shows the hand at a very large distance. The right image shows the hand at a very short distance where overexposure cannot be avoided. The image at the middle shows the optimal trade-off which is used as the working point of the system.

Coarse analysis A first crude estimation of the hand is obtained by fitting an ellipsoid into the data points which form a point cloud after the segmentation. We perform a principal component analysis to compute the position and the principal axes of the ellipsoid: We calculate the average/mean of the point cloud and compute the covariance matrix. Eigenvectors and eigenvalues of this matrix form the principal axes with their corresponding dimensions.

From [18]: Given a sequence $x = x_1, \ldots, x_n \in R^m$ of m-dimensional data points the $m \times m$ matrix

$$\mathrm{CovMat}(x) = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x}) \cdot (x_i - \bar{x})^T$$

denotes the *covariance matrix* of x. The vector $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$ is the mean vector of the different components of $x_1, \ldots, x_n$.
Because of working in the 3D space, every vector x consists of three values, the pixel position (x,y) in the image and the measured distance. Thus the covariance matrix looks like this:

$$\mathrm{CovMat} = \begin{pmatrix} \mathrm{Cov}_{x,x} & \mathrm{Cov}_{x,y} & \mathrm{Cov}_{x,z} \\ \mathrm{Cov}_{y,x} & \mathrm{Cov}_{y,y} & \mathrm{Cov}_{y,z} \\ \mathrm{Cov}_{z,x} & \mathrm{Cov}_{z,y} & \mathrm{Cov}_{z,z} \end{pmatrix}$$

in which $\mathrm{Cov}_{a,b} = \frac{1}{n} \sum_{i=1}^{n} (a_i - \bar{a}) \cdot (b_i - \bar{b})^T$ the covariances are.

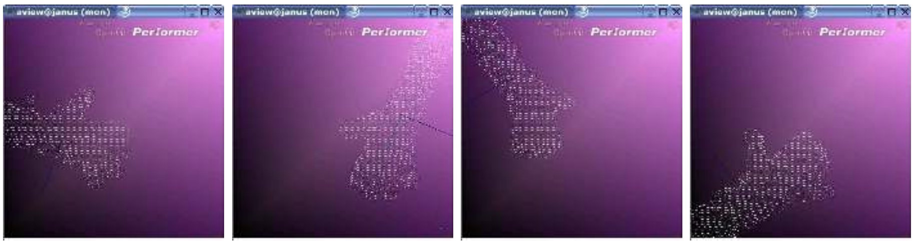


Figure 6. point cloud with principal axes

Figure 6 shows the point cloud with its principal axes visualized in AVANGO/ Performer. The processing performed so far can already be used for a simple and crude gesture interface: By tracking the center of mass of the human hand cloud, the user might control a pointer or an object on the screen in which the principal axes can also provide the direction. Think about playing darts: The center of mass should mark the position to drop and the largest of the principles axis marks the direction the dart is thrown away.

Fine matching The center of mass and the principal axes of the point cloud provide a good estimate for fitting an articulated hand model to the point cloud, as the visualization of the results have revealed. The aim of this first fine matching is to determine translation, rotation and scaling of the model accordingly. A skeleton of three dimensional hand model should be placed into the point cloud so that the distance of the surface points to the measured data is minimized. For these purposes we used a three dimensional hand model. It consists of 1,515 vertices and 3,000 triangles and corresponds to a human hand with anatomical structure. It was build for real time animation using a physically based muscle simulation and elastic skin attributes. Mrs. Irene Albrecht [9] made it available to us for this work. In our work we do not use the muscle model, we use only the triangle mesh and the vertices respectively.

The next step in our gesture recognition systems is the fitting process which aims at minimizing the distance between model and point cloud based on the sum of point-to-point distances. In this case, the Hausdorff distance may be used as a metric. On the other hand, metrics based on pure pairwise distances between point clouds do not take benefit from the more explicit surface properties of the hand model since it “only” tries to position the point clouds together as near as possible (similar to ICP registration). Alternatively, one might compare the surfaces/visual hulls between model and reconstructed surface patches by minimizing the volumne between surface patches directly since the model also consist of a triangle mesh.

More formally, the model must be placed at a good starting position inside of the point cloud to provide an adequate starting point for registration. Thus we look at the solution of the correspondence problem, e.g. matching the model data to the sensor data. The correspondences are needed to be able to compute the absolute distances between model points and point cloud the algorithm tries to minimize. We define the following cost function for the sum of minimal square distances, in which we first aim at parameterization of the first 7 DoFs:

$$K(\boldsymbol{\alpha}, t) := \sum_{<i,j>} \left[\left[\boldsymbol{x}_i^{(P)}(t) - \boldsymbol{x}_j^{(M)}(\boldsymbol{\alpha}) \right]^2 M_{ij} \right]$$

$\boldsymbol{\alpha} = (t_x, t_y, t_z, \phi_x, \phi_y, \phi_z, s)$: vector including the possible DoFs (three for translation, three for rotation and one for scaling).
$\boldsymbol{x}_i^{(P)}(t)$: ith point of the point cloud at time t
$\boldsymbol{x}_j^{(M)}(\boldsymbol{\alpha})$: jth model point after executing all of the transformations in $\boldsymbol{\alpha}$
$M_{ij} \in \{0, 1\}$: this matrix represents the correspondences between 3D point i and model vertex j; is there found a 3D point i being nearest to the considered model point j, the matrix gets the value 1 at the position ij; the pairs with the smallest distances are defined by these correspondences.

The system aims at finding a minimum of this cost function depending on the parameters $\boldsymbol{\alpha}$. However, one problem complicates the minimization procedure, the self-occlusion of the model: not all points of the model are visible for the camera: A point of a model is regarded as visible if it is not placed outside the field of view and the normal of the corresponding triangle does not point away from the camera. Hence, the articulation must discard (self)-occluded surface points automatically. The PCA deliveres the rough position of the point cloud, which correspond as the rough position of the skeleton. First the skeleton is moved to the measured center of mass (global translation) and rotated to become aligned to the second principal axis. By calculating the "length" of the object within the field of view (expansion of the point cloud along the second principal axis) using the bounding box, the model can be moved along this axis (the forearm) accordingly, so that the finger tips are at the border of the point cloud. Finally the model has to be scaled because of the perspective distortion and distance from the camera sensor.

So we pursued an iterative approach during optimization: With every iteration the scaling factor of the model gets heightened or lowered by 0.1 until no more scaling causes decreasing of the point distances, i.e. the cost function gets minimized. Visibility of each point is classified using the above mentioned criteria at each iteration step – only visible model points contribute to the cost function. On the other hand, the less vertices are visible, the less are used in the cost function and may cause a lower sum of square distances. To make the cost function invariant to the total number of vertices, we normalize all distances by dividing the cost by the number of visible point pairs. However, as the estimation of the DoF must be statistically significant, we use an some additional constrains: If less than one third of the model points are visible the reconstruction is stopped, as we must avoid situations in which an incorrectly upscaled hand model with minimal visible points still fits a part of the point cloud, since only a part of the hand model matches with the sensor data.

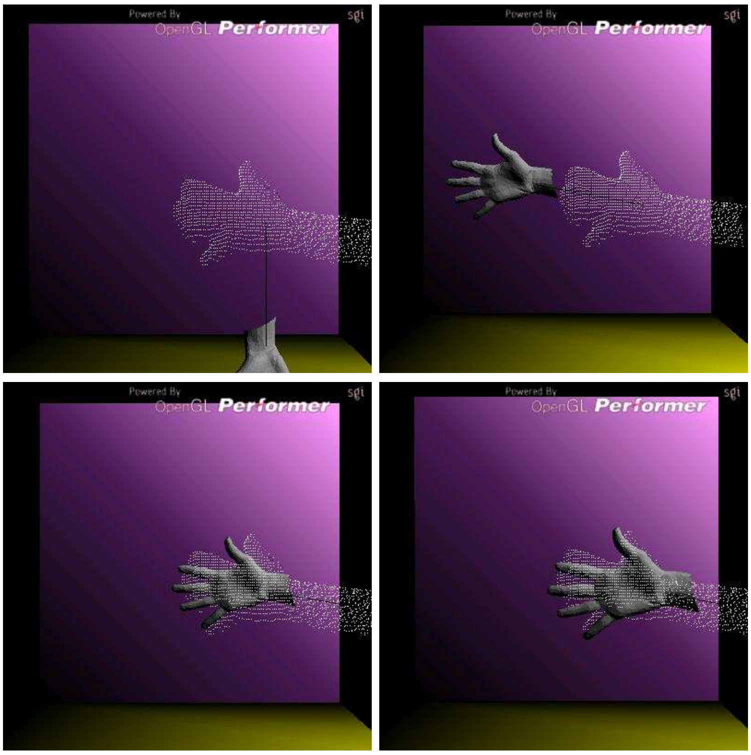


Figure 7. work flow upper left: moved to the center of mass; upper right: rotated on the second principal axis; lower left: moved into the point cloud; lower right: scaled

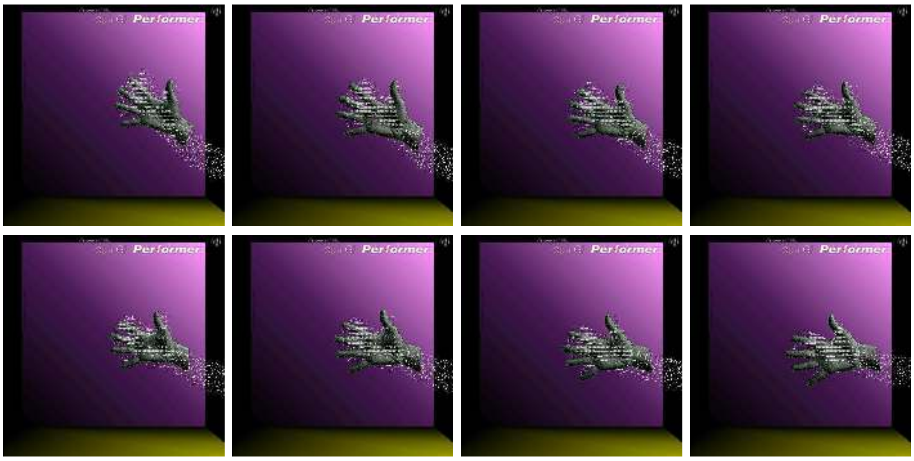


Figure 8. image sequence 1 (frontal view): shows a small movement downward

4 Results

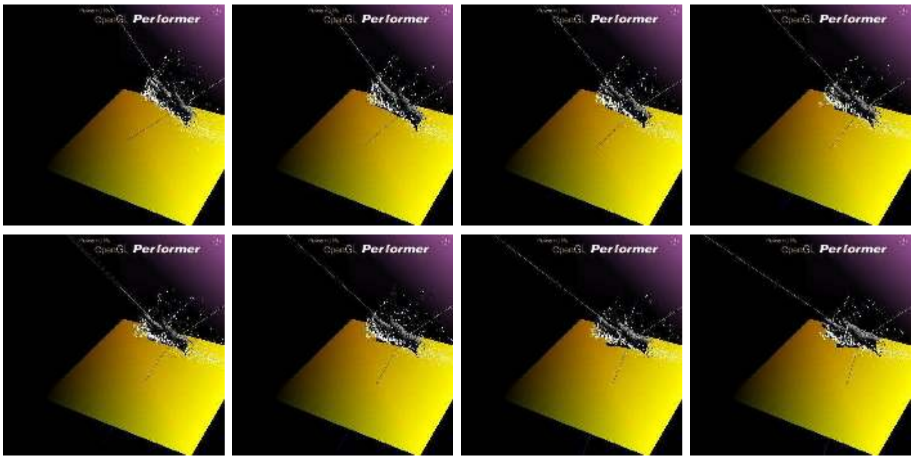


Figure 9. image sequence 1 (side view): shows the same movement as Fig. 8 from another perspective

Figs. 8 and 9 document a felicitous recognition. The hand model is well fitted into the point cloud and follows it as the frontal and side view demonstrate. Fig. 10 shows an more problematic recognition: the algorithm has difficulty in detecting the hand moving into the scene. The entry orientation has been detected wrong in image 1. Image 2 shows how the algorithm tries to fit the model into the small point cloud. The usage of the median distance had not the intended effect, as also image 9 and 11 reveal. Moreover, the scaling is also incorrect. The remaining images show how the model is fitted into the point cloud correctly.

We achieved a recognition of seven DoFs within a frame rate of 2-3fps, using a 3.00GHz Intel Pentium 4 with 2GByte RAM. However, the algorithms not being optimized for speed and efficiency. One of the main objectives of this paper was to investigate gesture recognition with the new special hardware, the Swissranger SR2, and to summarize the advantages and disadvantages of such an approach.

The primary advantage of the special hardware is to be able to receive images with the depth information of the scene with considerable high frame rates. Furthermore there is no special background necessary. Worst-case problems in video-based segmentation, e.g. "blue pullover in front of a blue wall" or "a hand in front of a skin colored background" loose their horror. On the other hand, if there is a lot of sun/light the range camera gets interfered. One can observe two fundamental effects: On the one hand the intensive background light generates a lot of additional electrons which causes higher photon shot noise. Thereby it influences the preciseness of the measurement negatively. And on the other hand

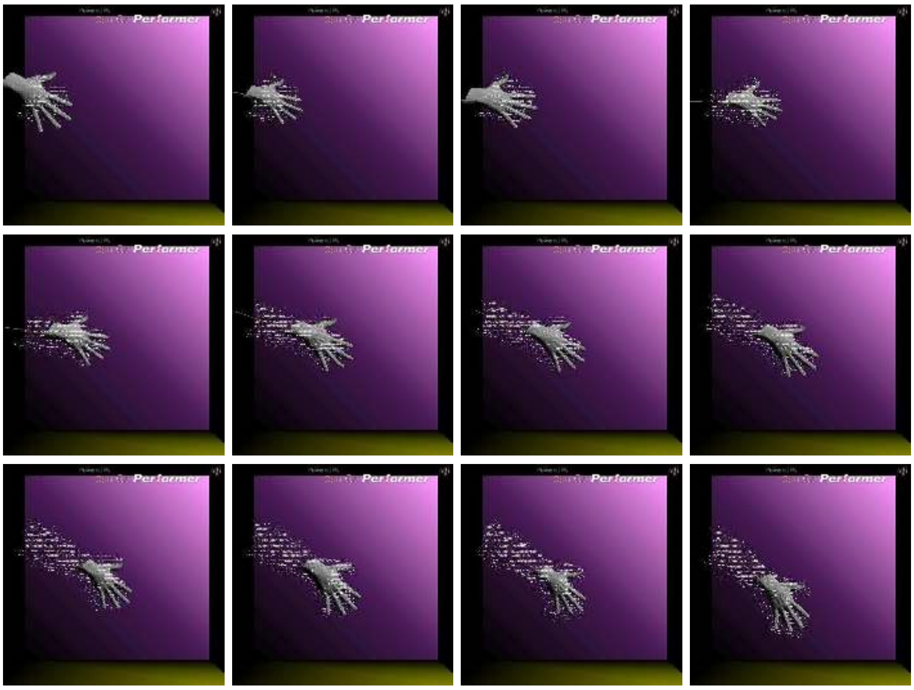


Figure 10. image sequence 2: shows the incoming of the hand into the field of view and the following movement downwards

the electrons generated by the background light over-saturate the pixel sensor and thereby completely overload their capacity.

Despite these rather technical advantages and shortcomings due the special hards, the correspondences search remains a considerable difficult problem for which efficient and robust methods are rare. Two problems have also be kept in mind:

1. *visibility and occlusion:* The problem of self-occlusion has to be considered, since the range camera delivers $2\frac{1}{2}$D data (by sampling surface points from the cameras point of view) but the model and the objects are an real 3-dimensional objects. Moreover, the objects may only be partly visible due to the limited frustrum volume and the depth segmentation. On this account these points become extraneous to the ”marriage” of the model with the point cloud because the camera also does only see the front of the scene (quasi $2\frac{1}{2}$D). Only a correct classification as invisible points may help which can only be performed after correct registration – an classical “chicken and egg problem”, which we have tried to solve by using iterations.
2. *tail of a comet:* The “tail of a comet” produces many problematic 3D-points which cannot be matched correctly with any model. We have circumvented the problem by using some ad-hoc constraints - first principal axis points to

the depth, second axis specifies the direction of the forearm and the third axis determines the "thickness" of the arm - to minimize its influence on the recognition. However, it is our believe that a more careful analysis will replace our constrains in the future.

Last but not least, it is important to use a good model. The model currently used is qualified because of its anatomical structure and its real time capability. It can be animated and fitted into the point cloud.

5 Conclusion and Outlook

5.1 Conclusion

We have described a pilot study of doing hand gesture recognition with a novel IR time-of-flight range camera. After calibration and noise reduction to get useful measured data we defined a region of interest and interpreted the remaining data as a point cloud. This approach gave us the opportunity to do a principal component analysis on it, which returned the principal axes of the point cloud being a clue for the precise positioning of the model. Visualization of the intermediate results showed how good PCA works on the measured data, i.e. how good the clue is. After showing some already possible ways for human-computer interaction we continued with the fine matching. A realistic 3D hand model was fitted into the point cloud. The correspondence problem and classification of visibility have been analyzed and were solved in an iterated algorithm aiming at minimizing a cost function based on point-to-point distances. Finally we identified global translation, rotation and scaling to fit the model into the data at the best possible rate. The system was able to recognize 7DoFs of a human hand with 2-3 Hz framerate. This is a promising result and defines a road map for further research in the future.

5.2 Outlook

Current work aims at improving the robustness of the system against out-liners by registering the model to the sensor point clouds with more robust algorithms, such as Random Sample Consensus (RANSAC). The introduction of a tracking mode, e.g. by adding a particle filter to the system might significantly decrease the processing time. The fast tracking mode might refine the hand model in a highly reduced search space and should support significant higher framerate. Another promising field of research lies in exploiting the currently discarded IR intensity measurements. Moreover, data stemming from additional calibrated high-resolution video cameras may further increase the precision of the reconstructed hand parameters. Image processing algorithms and sensor fusion methods may combine the depth, IR and sensor data from the visible acquisition hardware in a consistent manner.

Last but not least, the new model of the 3D-camera, the Swissranger 3000, has recently been made available by CSEM. The new hardware supports to set the integration time for each pixel individually. Hence, smaller distances between camera and the human hand without over-saturation of the photo integrators become feasible and may result in an increased sample density. Thereby, more detail of the hand becomes visible and might also increase the number of recognizable DoF significantly.

6 Acknowedgement

This work was funded by the German research project "Virtual Human" under the grand 01IMB01.

References

1. Hall-Holt, O., Rusinkiewicz, S.: Stripe boundary codes for real-time structured-light range scanning of moving objects. In: Eighth International Conference on Computer Vision (ICCV). (2001)
2. Ypsilos, I.A., Hilton, A., Rowe, S.: Video-rate capture of dynamic face shape and appearance. fgr (2004) 117
3. Malassiotis, S., Tsalakanidou, F., Mavridis, N., Giagourta, V., Grammalidis, N., Strintzis, M.G.: A face and gesture recognition system based on an active stereo sensor. In: Proceedings 2001 ICIP, Thessaloniki, Greece, 7-10 Oct. 2001, vol.3. (2001) 955–8
4. Bray, M., Koller-Meier, E., van Gool, L.: Smart particle filtering for 3d hand tracking. In: Proceedings. FGR2004, Seoul, South Korea, 17-19 May 2004. (2004) 675–80
5. Perrin, S., Cassinelli, A., Ishikawa, M.: Gesture recognition using laser-based tracking system. In: Proceedings. FGR2004, Seoul, South Korea, 17-19 May 2004, IEEE Comput. Soc (2004) 541–6
6. Liu, X., Fujimura, K.: Hand gesture recognition using depth data. In: Proceedings. FGR2004, Seoul, South Korea, 17-19 May 2004, IEEE Comput. Soc (2004) 529–34
7. : computer graphik, universität bonn. http://cg.cs.uni-bonn.de/ (2005)
8. Bendels, G.H., Kahlesz, F., Klein, R.: Towards the next generation of 3d content creation. In: Proceedings. AVI '04. (2004) 283–89
9. Albrecht, I., Haber, J., Seidel, H.P.: Construction and animation of anatomically based human hand models. In: SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer animation. (2003) 98–109
10. : Swiss center for electronics and microtechnology. www.csem.ch/ (2005)
11. : Swissranger sr-2 miniature time of flight camera. www.swissranger.ch/ (2005)
12. CSEM: Swiss ranger sr-2 datasheet. www.swissranger.ch/ pdf/SR-2_DataSheet.pdf (2005)
13. Oggier, T., Michael Lehmann, Rolf Kaufmann, M.S., Richter, M., Metzler, P., Lang, G., Lustenberger, F., Blanc, N.: An all-solid-state optical range camera for 3d real-time imaging with sub-centimeter depth resolution (swissranger). In: SPIE, conference on optical system design, St. Etienne, September 2003. (2003)
14. Oggier, T., Büttgen, B., Lustenberger, F., Becker, G., Rüegg, B., Hodac, A.: Swissranger sr3000 and first experiences based on miniaturized 3d-tof cameras. In: Proceedings, 1st Range Imaging Research Day, September 8/9, 2005 at ETH Zurich Switzerland. (2005) 97–108
15. Kahlmann, T., Ingensand, H.: Calibration and improvements of the high-resolution range-imaging camera swissranger. In: Proceedings of SPIE Vol. 5665; Videometrics VIII; 16-20 January 2005, San Jose, California, USA. (2005)
16. Kahlmann, T., Ingensand, H.: Range imaging sensor properties and calibration. In: Proceedings, 1st Range Imaging Research Day, September 8/9, 2005 at ETH Zurich Switzerland. (2005) 71–80
17. Gut, O.: Untersuchungen des 3d-sensors swissranger. Master's thesis, ETH Zürich (2004)
18. Dreiseitl, S.: Skriptum zu mat3 ws2004/05 (2005)