Real time Hand Gesture Recognition using a Range Camera

Zhi Li, Ray Jarvis
Monash University
Wellington Road Clayton, Victoria AUSTRALIA
Zhi.li@eng.monash.edu.au, Ray.Jarvis@eng.monash.edu.au

Abstract

This paper proposes a real time hand gesture recognition system. The approach uses a range camera to capture the depth data. After some pre-processing procedures, the depth data is used to segment the hand and then locate the hand in 3D space. The hand shape is classified into known categories using a chamfer matching method to measure the similarities between the candidate hand image and the hand templates in the database. The 3D hand trajectory is recognized by a Finite State Machine (FSM) method. Each gesture consists of several states. The 3D hand position determines the state transition of each gesture recognizer. Experiments show that the system performs reliably for recognizing both static hand shapes and spatial-temporal trajectories in real time.

1 Introduction

Fast and robust analysis of hand gestures has received increasingly more attention in the last two decades. Gesture recognition from a single view is important in the Human-Robot Interaction (HRI) scenario. Both static hand shape and dynamic hand trajectories are expressive in our daily life. These kinds of hand gestures are naturally preferred in HRI applications.

Various methods have been proposed to segment and track human hands. Many systems [Imagawa et al., 1998] [Hong et. al., 2000] have achieved promising performance; however, these methods only operate under strong restrictions of the environment, because they rely on simple clothing, static background, and constant lighting conditions.

We aim to achieve a real time hand gesture recognition system in a natural environment. It means we do not need homogenous clothing and should be able to perform reliably with cluttered and even non-static backgrounds.

We investigated the usefulness of a 3D range camera for our hand gesture recognition system. This hardware exhibits significant advantages over traditional cameras in the aspect of unambiguously capturing the depth data at a high frame rate, which makes the segmentation and tracking the hand in 3D space easy. Hand shapes are recognized by the Chamfer Matching method [Borgefors, 1988], and 3D trajectories are recognized using a Finite State Machine (FSM) method. Gestures are recognized in real time. This gesture recognition method has wide applications including human robot interaction, intelligent rooms, virtual reality and game control.

The remainder of this paper is organized as follows: after a brief review of the related work in section 2, we investigate the property of the range camera in section 3. Hand segmentation and its 3D position are obtained in section 4, which is the input data for the hand shape and trajectory recognition algorithms in section 5. In section 6 we present the experiments and the result. Finally, discussion, conclusion and future work are in section 7.

2 Related work

Hand detection and segmentation is an essential component for gesture recognition. Many approaches [Mo et. al., 2005] [Kjeldsen and Kender, 1996] use color information, since the skin color is a salient feature different from the background in most cases. However, these methods are not reliable under unstable illumination conditions, where, obviously, it is more challenging to extract complete hand shapes. Some methods use special colored gloves or a magnetic sensing device (data gloves) [Sturman and Zeltzer, 1994] to simplify the task, but they hinder the naturalness of daily use. The intentions of the user should be recognized effortlessly and non-invasively.

Most of the hand trajectory recognition systems deal with 2D image data. [Starner and Pentland, 1995] tracked the hand by colored glove and natural skin tone for American Sign Language recognition. Some methods use stereo vision to achieve the hand tracking in 3D space. [Nickel et al., 2004] [Stieflhagen et al., 2004] used stereo cameras to locate the hand's position in 3D in order to find the pointing direction. [Abe et al., 2000] proposed a 3D drawing system using a top-view camera and a side-view camera. Stereo vision is a popular choice for depth sensing. However, it is highly dependent on the textures of the object to find the correspondence between images, and becomes erroneous for texture-insufficient surfaces.

A time-of-flight range camera has become popular in the recent years. Although the technology is still in its early days, resulting in low resolution, noisy data etc, it has already been applied in a number of applications such as game controlling [Wang et al., 2008], upper body gesture recognition [Holte et al., 2008] [Grest et al., 2007], robot navigation [Prusak et al., 2007], and mobile human-robot teaming [Loper et al., 2009].

3 Range Camera

A 3-D range camera is employed as our apparatus. It delivers the depth data of objects in its view at every pixel at a high frame rate. The distance of 3D points is determined by a Time-of-Flight (TOF) approach using modulated infrared light. The phase shift between the reference and reflected signal is determined by a sampling and correlating method for each pixel, and then the distance is calculated by the phase shift. See [Linder and Kolb, 2007] for more details about the operational principle of the range camera technology. Figure 1 shows the range camera we adopted, which is developed by PMD Technologies GmbH.

Figure 1 PMD Range Camera

As illustrated in [Kahlmann and Ingensand, 2005], with *d* as the measured distance, the coordinate of a 3D point (see figure 2) can be calculated by equation (1).

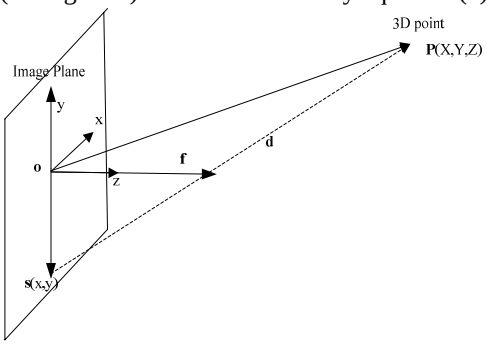


Figure 2 Geometrical illustration for coordinate evaluation [Kahlmann and Ingensand, 2005]

$$\boldsymbol{P}\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \boldsymbol{s} + \boldsymbol{d} = \begin{pmatrix} x \\ y \\ 0 \end{pmatrix} + \begin{pmatrix} -x \\ -y \\ f \end{pmatrix} \cdot \frac{d}{\sqrt{x^2+y^2+f^2}} \qquad (1)$$

P*(X,Y,Z)* is the coordinate of the 3D point, ***s****(x,y,0)* is calculated from image coordinate to Euclidean coordinate, *d* is the measured distance, and *f* is the focal length.

Assuming the camera is a simple pinhole model, in which the optical center is at the image center. We can also use equation (2) to retrieve the 3D coordinates, if we know the Field-of-View (FOV) of the camera, and ignore the difference between the distance *d* and *Z* coordinate in the depth direction, which is reasonably correct since the distance in our application is always over 1m and the FOV of the range camera is 28 degrees.

$$\mathbf{p}\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} \tan(FOV/2)\cdot Z\cdot(2x-W)/W \\ \tan(FOV/2)\cdot Z\cdot(H-2y)/H \\ d \end{pmatrix} \qquad (2)$$

In equation (2), we assume the *Z* coordinate equal the measured distance *d*, *W* is the width of the image, *H* is the height of the image. *W*, *H*, *x* and *y* are image coordinates in pixel units. In this way, we avoid the burden of calibrating the range camera. Actually, it is troublesome to calibrate the range camera, because of low-resolution and an unevenly illuminated image with a large amount of noise.

Compared to stereo vision systems, which are the traditional visual methods to capture the depth data, this device has the following advantages:

1. Illumination-invariant and color-invariant. Because it actively emits modulated infrared light, it is not influenced much by the ambient illumination condition and color of the objects in the environment.
2. Texture-independent. Because stereo vision methods are highly dependent on the texture on the objects to find the correspondence between multiple images, the range camera significantly outperforms the stereo methods in texture-insufficient regions.
3. Depth resolution is approximately 5~15mm in the distance range of 0.5 to 3 meters, variable with exposure time and reflectiveness of the surface. A comparison of the PMD range camera and stereo-vision for the task of surface reconstruction has been investigated by [Beder et al., 2007]. Their experiments show that the PMD range camera system outperforms the stereo system in terms of achievable accuracy for distance measurements.
4. Both depth and grey scale data is captured at high frame rate (15 fps).

Although having the above advantages, this range camera has some drawbacks, such as a narrow Field-of-View (28 degrees), low resolution (160x120), and the distance data contains a large amount of noise, especially near the edges of the objects, which is often referred to a 'jump boundary effect'.

3.1 Depth data processing

There is a significant amount of noise in both depth and intensity data from the sensor. Pre-processing of the depth data is essential. The standard deviation of the range depth data is found to be reciprocal to the signal amplitude [M. Frank, 2009]. We remove the "bad-pixels" first, whose amplitude of reflection is below a specified threshold. Low amplitude indicates that their depth data is inaccurate because of low signal/noise ratio. The depth values at these removed points are assigned using a linear interpolation method. Then we apply a Median filter. Speckle noise can by reduced effectively by the Median Filter. However, the most annoying noise which is known as 'jump boundary effect' can not be removed in this way. The points near the edges of the objects tend to "merge" into the background. Note the distribution of the points on the edge of the hand in figure 3. One reason for this error is the limited resolution of the sensor chip [Breuer et al., 2007].

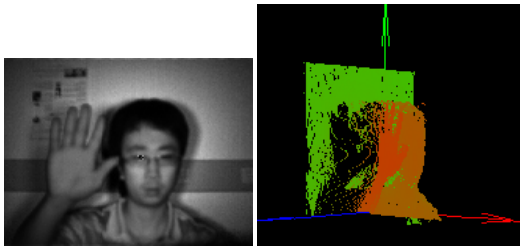


Figure 3 A grey scale image (left) and 3D points (right) from the Range Camera

Overexposure and underexposure will also result in errors in the depth data. Thus, we frequently calculate the average intensity value of the images and adjust the integration time to maintain an average grey value between 125~150.

Fig. 4 shows the effect of the pre-processing on the depth data using the above methods. The left image is the grey image from the range camera, showing the scene. The middle image is rendered using the original depth data. The right image illustrates the processing effect. We can see that the right one is less noisy. The colors represent the depth information of objects: the RGB color value is assigned as:

$$\begin{aligned} R &= (d/Maxd)\cdot 255 \\ G &= (Maxd-d)/Maxd\cdot 255 \\ B &= 0 \end{aligned} \qquad (3)$$

d is the depth value, $Maxd$ is the maximum depth of the points in the scene or a specified value.

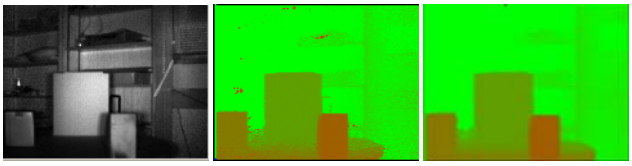
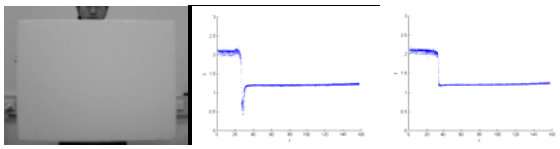


Figure 4 The effect of depth data processing

3.2 Error in regions with high velocity

Several studies have been proposed to calibrate the intrinsic parameters and depth value of the range camera [Kahlmann and Ingensand, 2005] [Reulke, 2006]. However, they only consider static scenes. Very few researches investigated the performance of the range camera with moving objects.

The "1-tap distance acquisition method" of the range camera requires that the four consecutive taps are necessary for a single distance calculation [Oggier et al., 2005]. If the distance at a pixel senses a change in this time slot, the distance calculation is falsified. This error is most serious at the edges of an object, because a distance measure could be a pixel on an object for the first two taps and the background for the last two taps. Concretely, this motion blur effect is illustrated in figure 5. In this experiment, the person moves a flat board from left to right. The depth data of the moving objects is shown in the middle image (b) and the depth data when the object is already static is shown in the right image (c). Note that this motion blur effect is different from the jump boundary effect. The depth value tends to be larger because the pixels on the border are "merging" into background. However, on this moving flat board, the depth data on the edge tends to be smaller. It is probably because of the instability of the amplitude of the reflected signal. Therefore, in order to find a relatively reliable depth data of a moving object, it is reasonable to consider the depth value at the centroid, since in most cases the depth value at the centroid varies less compared to other parts of the object.



(a) (b) (c)

Figure 5 Comparison of the depth data on a moving object and a static object

4 Hand Segmentation and Trajectory Extraction

Reliable hand segmentation from the background and other parts of body is crucial for further analysis. After successful segmentation, many techniques, such as Haar-wavelet decomposition [Jacobs, 1995] and Chamfer Distance Matching [Borgefors, 1988], can be applied to recognize the candidate hand shape, classifying it into pre-defined categories. The fineness of segmentation significantly influences the recognition result correctness.

Skin color information is traditionally used to locate and segment hand region. To reduce the effect of variable illumination conditions, the RGB color is often projected to other color spaces, which separate the color into brightness and chromatic channels, such as HSV, CIE Lab etc. However, this kind of methods does not completely eliminate the effect of the illumination disturbance.

Tracking the hand in 2D images is relatively simple. However, it is difficult to track the hand in 3D space. Stereo vision techniques are widely used for this purpose, but they often suffer from the lack of sufficient texture on the hands. Furthermore, accurate disparity computation requires high resolution images and more computation time.

In contrast to the traditional methods based on color information, hand segmentation can be achieved using only depth data. In the human-robot interaction scenario, we assume that a single person is the nearest object in the camera's view. When the person is indicating instructions by hand gestures, the hand is usually at a distance in front of the body.

Now we can simply segment the hand by a depth histogram method. Although the hand is the nearest object to the camera, we have to take the noise in depth data into account. As shown in figure 6, we use a histogram method. We put all the depth data into N bins with an interval of 10cm, and select the bin which indicates the smallest

distance and also contains sufficient number of points. Note that the bin in a red circle in figure 6 has the smallest distance value, but contains only 13 points, which indicates that it consists of noise.

After the hand distance is found, the hand shape is composed by the points whose depth values are in a range between $[d_h - \Delta d_1, d_h + \Delta d_2]$, where d_h is the hand distance, and Δd_1 and Δd_2 are specified thresholds. The hand position in image coordinate (x,y) is the center of the hand region. The depth value of the hand is further refined as follows: check the depth value in a small window of the hand, if they all fall in the foreground, then use the average value of the depth data in the window. Otherwise, adjust the position of the small window first until they all fall in the foreground. In this way, the 3D hand trajectory can be obtained.

Morphological operations i.e. erode and dilate, are applied to eliminate the noise. Then the hand shape is normalized to a uniform size and the edges are easily found from the binary image.

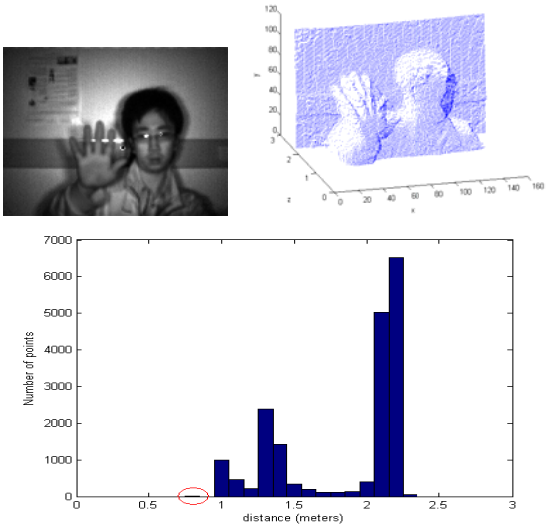


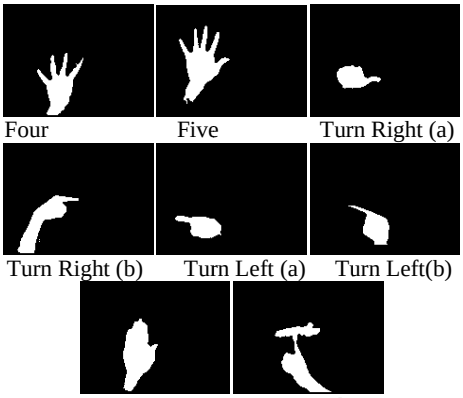
Figure 6 An example of hand gesture and the histogram of the depth data

5 Hand gesture Recognition

5.1 Hand shape analysis

For hand shape analysis, a database needs to be established. A large set of images containing various hand patterns are recorded with known labels in the training stage. Figure 7 show the samples of hand shapes that we are currently interested in. Both left and right hands could be used for gesturing, but only left hand shapes are shown here.

One Two Three



Four Five Turn Right (a)

Turn Right (b) Turn Left (a) Turn Left(b)

Stop (a) Stop (b)

Figure 7 hand shape patterns

To match a candidate hand image against our database, the chamfer distance matching method [Borgefors, 1988] is employed.

Distance transformation (DT) of the hand shape template images are calculated in advance before the test stage. DT is a reasonable approximation of the Euclidean distance. In a binary edge image, edge pixels are set to zero and non-edge pixels are set to infinity. The value of the pixel at position (i,j) at iteration step k is then computed as Equation 4 [Borgefors, 1988].

$$v_{i,j}^{k} = \min(v_{i-1,j-1}^{k-1} + 4, v_{i-1,j}^{k-1} + 3, v_{i-1,j+1}^{k-1} + 4, v_{i,j-1}^{k-1} + 3, v_{i,j}^{k-1}, v_{i,j+1}^{k-1} + 3, v_{i+1,j-1}^{k-1} + 4, v_{i+1,j}^{k-1} + 3, v_{i+1,j+1}^{k-1} + 4) \quad (4)$$

The iterations continue until there are no value changes. Figure 8 shows an example of distance transformation. The hand is first segmented out of background (left image). The edges are found and normalized (middle image), then the distance transformation is calculated (right image).

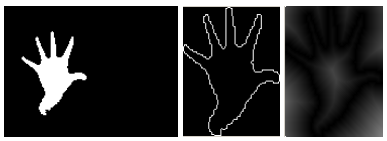


Figure 8 example of distance transformation

The matching procedure measures the similarity of two edge images. The matching score of a binary edge image I(i,j) and a distance transformation DT(i,j) is calculated as Equation 5.

$$s = \sqrt{\frac{1}{n}\sum_{i,j}(I(i,j)\cdot DT(i,j))^2} \quad (5)$$

[Borgefors, 1984] compared four different "averages" for the matching measure: median, arithmetic,

root mean square (r.m.s) and maximum. The r.m.s was found to give fewer false minima than others. The smaller the matching is, the higher the similarity is. A perfect fit will result in zero.

In order to measure the similarity between two images, e.g. a candidate image $\boldsymbol{A}$ and a template hand image $\boldsymbol{B}$ **(**the distance transformation of the template, $\boldsymbol{DT_B}$, is already obtained in advance), first, the candidate image is normalized and extracted to a binary edge image $\boldsymbol{I_A}$, then a matching score is calculated by equation 5; second, the distance transformation of the normalized candidate image $\boldsymbol{DT_A}$ is derived as well as the binary edge image of the template $\boldsymbol{I_B}$, then another matching score is calculated by equation 5. These two scores are added together as the final similarity measurement between the candidate images $\boldsymbol{A}$ and the template $\boldsymbol{B}$. The similarity measurements between the candidate and all the templates in the database are calculated and the smallest score indicates the recognized gesture.

5.2 3D hand trajectory recognition

A hand trajectory is also expressive in an interaction. Some intentions are naturally expressed by movement rather than static hand shapes.

The hand trajectory in 3D space is captured by the method in section 4. Since the hand normally has an obvious pause between a new gesture and previous one, we take it as the starting point of a gesture. The subsequent hand 3D position is relative to the starting position. In this way, the gesturer does not need to start from a particular fixed position every time.

The trajectories are recognized by a Finite State Machine (FSM) method. Each gesture is defined to be a sequence of state transitions in the spatial-temporal space.

A state S is defined as a simple vector <$\boldsymbol{u,d}$>, where $\boldsymbol{u}$ is the center of the state's space, $\boldsymbol{d}$ is the distance threshold in the *x, y, z* directions. The 3D space is divided into several cells representing different states. Note that each FSM recognizer has its own way of splitting the spatial space. A 3D position $\boldsymbol{p}(x,y,z)$ may be in the *i*th state of gesture one, while in the *j*th state of gesture two.

When a new hand position arrives, each FSM recognizer determines whether to stay at the current state or enters the next state based on the spatial parameters. A gesture is recognized if a recognizer reaches its final state [Hong et al., 2000]. Each FSM recognizer may have different number of total states, thus this method allows simple movements to be represented by fewer states and complicated gestures by more states.

This online recognition method determines a state transition when a new hand position is provided. It is different from the approaches that require complete gesture data before a recognition procedure begins.

6 Experiment and result

The hand gestures are performed at a distance of about 1.5 meters from the range camera in an indoor environment.

6.1 Hand shape recognition

We tested the ability of the system to recognize the hand shapes shown in figure 7. The hand shape templates were captured and stored in advance, and the test was performed in another day. To further test the robustness of the method, it will be tested with different people in the near future.

We noticed that the rotation of the hand both around horizontal and vertical axes may make the appearances of the hand different, which would affect chamfer distance matching result. Furthermore, in some cases, the angular separation (the fingers are sharply or narrowly separated) may also influence the matching score. Although it is difficult to achieve viewpoint independent performance because of self-occlusion from a single view, the slight rotation in space should not affect the recognition output. Therefore, in the training stage, we collected 10~18 hand shape templates for each gesture to make the recognition more robust. Each gesture is conducted with horizontal rotations of -10^o, -5^o, 0^o, 5^o and 10^o, vertical rotation of -10, 10, and several angular separations if multiple fingers are involved. Figure 9 gives a rough idea of the various appearances of the same gesture.

Figure 9 Examples of difference appearances of the same gesture

During the period when the hand is changing from one gesture to another, the shape is not recognizable, so for testing purpose, we extracted images from videos, and manually delete the hand images which can not be recognized even by human. The offline recognition confusion matrix is shown in table 1. The gestures from 1 to 11 in the table are shown in figure 7 in the same order. The overall recognition error rate is under 3%.

For the real time, online recognition, we use a stability counter. Only when the same recognition result repeats several times it will be confirmed as a recognized gesture. The online recognition rate is 98%, because of the stability counter.

Like other training-test methods, the selection of the training set is vital to recognition results. If the training set covers most of the gestures which appear in the test set, then the error rate is relatively low.

6.2 Hand trajectory recognition

Five types of hand trajectories are trained and tested using the FSM method in section 5.2, including "wave hand", "draw a triangle", "pick up", "put down" and "come here".

Several key frames of the movement from the video are shown in figure 10. The person waved his hand from left to right and then from right to left. Table 2 shows the states transition of each gesture according to the 3D hand position in figure 10. Only "wave hand" went through all its states, so it is recognized.

	1	2	3	4	5	6	7	8	9	10	11	total
1	82	0	1	0	2	2	0	0	0	0	0	87
2		76	3		2							81
3		3	64	3								70
4				57	3							60
5					59							59
6			3	1	1	45						50
7							51					51
8								47				47
9								1	59			60
10						2				68		70
11											70	70

Table 1 hand shape recognition confusion matrix

Figure 10 example of waving hand

gestures							States finished	States left
Wave hand	1	2	3	3	4	5	5	0
Draw triangle	1	1	2				2	2
Come here	1						1	4
Pick up	1						1	2
Put down	1						1	2

Table 2 states transition of each gesture in figure 10

Figure 11 shows some frames of the video in which a person moved his hand forwards and backwards to express a "come here" gesture. The states transitions of each gesture are shown in table 3. Only the third gesture went through all its states and it is recognized as "come here".

Figure 11 example of "come here"

gestures						States finished	States left
Wave hand	1					1	4
Draw triangle	1					1	3
Come here	1	2	3	4	5	5	0
Pick up	1					1	2
Put down	1					1	2

Table 3 states transition of each gesture in figure 11

The overall recognition rate of the hand trajectories is about 88%. Details for each gesture are shown in table 4.

Gestures' names	Performed times	Recognized times	Recognition Rate
Wave hand	10	9	90%
Draw triangle	10	8	80%
Come here	10	10	100%
Pick up	10	7	70%
Put down	10	10	100%

Table 4 hand trajectory recognition result

Using the method in section 5.2, simple and complicated movements can have different numbers of state transitions. It can detect the start and end point of a gesture automatically, and a gesture is not required to start at a particular position. However, the scale of the movement may affect the result.

7 Discussion and Conclusion

A real time hand gesture recognition system has been described. Taking advantage of a range camera, hand segmentation and 3D tracking becomes easy and invariant to the changes in the environment. Experiments show that the chamfer matching measurement for hand shape analysis and the FSM method for recognizing the hand trajectory achieve high recognition rates.

The main drawback of the method using only depth data in section 4 is that, when the hand and forearm are in the same depth range, the segmentation using only depth data is not able to further distinguish hand from forearm simply. In this situation, the hand positions in the images which are calculated by finding the central points of the segmented region will also be erroneous.

An ordinary web camera can be used in conjunction with the range camera to provide useful color information and higher resolution, which is also potentially useful for face detection and recognition.

First, image coordinates of the web camera and the range camera is aligned. Second, the objects in the background whose depth values are over a specified threshold is removed in the range camera. Third, the corresponding regions of the background in the web camera are removed. Last, the hand is tracked by color information from the web camera using a Particle Filter method.

We then find the corresponding hand position in the range camera in order to find its depth value. The hand is segmented from a cube whose center is the 3D hand position.

This extra web camera helps to improve the accuracy of the 3D hand position, especially in the situations mentioned above. However, it also imposes extra burden of alignment between the web camera and the range camera. Furthermore, when the person is wearing a short sleeve clothes, the color based method may locate the hand position incorrectly.

In the future, the recognition of more types of hand shapes will be included. For testing robustness, the templates will be created by one person, while hand images from several persons will be used in the test stage. The hand trajectory recognition method should be improved so that it is invariant to the scale of the movements. Gestures performed in an outdoor environment will also be tested to evaluate the capability of the range camera and the robustness of the proposed methods.

References

[Beder et al., 2007] C. Beder, B. Bartczak and R. Koch. A comparison of PMD- camera and stereo-vision for the task of surface reconstruction using Patchlets. Computer Vision and Pattern Recognition, pp 1-8, 2007.

[Borgefors, 1988] Gunilla Borgefors. Hierarchical Chamfer Matching: a Parametric Edge Matching Algorithm. IEEE transactions on systems, man, and cybernetics. Pp 849-865. 1988.

[Breuer et al., 2007] Pia Breuer, Christian Eckes, and Stefan Muller. Hand Gesture Recognition with a novel IR Time-of-Flight Range Camera – A pilot study. LNCS pp 247-260, 2007

[Frank et al., 2009] M. Frank, M. Plause, H.Rapp, U. Kothe, B. Janhne, and F.A. Hamprehct. Theoretical and experimental error analysis of continuous-wave time-of-flight range cameras. Opt. Eng. 48, 013602, 2009.

[Grest et al., 2007] Daniel Grest, Volker Krüger and Reinhard Koch· Single View Motion Tracking by Depth and Silhouette Information. Lecture Notes in Computer Science. pp.719-729, 2007

[Hong et al., 2000] Pengyu Hong, Matthew Turk and Thomas S. Huang. Gesture Modeling and Recognition using Finite State Machines. IEEE conference on Face and Gesture Recognition, 2000.

[Holte et al., 2008] M.B. Holte, T.B. Moeslund and P. Fihl. Fusion of Range and Intensity Information for View Invariant Gesture Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008.

[Imagawa et al., 1998] Kazuyuki Imagawa, Shan Lu, Seiji Igi. Color-Based Hands Tracking System for Sign Language Recognition. Proceedings of the 3rd. International Conference on Face & Gesture Recognition. pp 462-468, 1998

[Jacobs, 1995] C. E. Jacobs, A. Finkelstein, and D. H. Salesin. Fast Multiresolution Image Querying. International conference on computer graphics and interactive techniques, pp 277—286, 1995.

[Kahlmann and Ingensand, 2005] T. Kahlmann, H. Ingensand, Calibration and improvements of the high-resolution range-imaging camera SwissRanger, Proceedings of the SPIE, Vol. 5665, pp. 144-155, February, 2005.

[Kjeldsen and Kender, 1996] Rick Kjeldsen and John Kender .Toward the use of gesture in traditional user interfaces. International Conference on Automatic Face and Gesture recognition. 1996, pp 1257-1261.

[Linder and Kolb, 2007] M. Lindner and A. Kolb. Calibration of the Intensity-Related Distance Error of the PMD TOF-Camera. Proceedings of Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision, September, 2007.

[Loper et. al.2009] Matthew M. Loper, Nathan P. Koening, Sonia H. Chernova, Chris V. Jones, Odest C. Jenkins. Mobile human-robot teaming with environmental tolerance. Proceedings of the 4th ACM/IEEE international conference on Human robot interaction. Pages 157-164 . 2009

[Nickel et al., 2004] Kai Nickel, Edgar Scemann, and Rainer Stiefelhagen. 3D-Tracking of Head and Hands for Pointing Gesture Recognition in a Human-Robot Interaction Scenario." Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004.

[Oggier et al., 2005] Thierry Oggier, Mike Stamm, Matthias Schweizer, Jorn Pedersen. User manual SwissRanger 2 rev. b. Version 1.02, 2005.

[Reulke, 2006] R. Reulke. Combination of distance data with high resolution images, IEVM06, 2006

[Starner and Pentland, 1995] Thad Starner, and Alex Pentland. Real-time American Sign Language recognition from video using hidden markov models. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol 20, pp. 1371-1375, 1995.

[Stiefelhagen et al., 2004] R. Stiefelhagen, C. Fugen, P. Gieselmann, H. Holzapfel, K. Nickel and A. Waibel. Natural human-robot interaction using speech, head pose and gestures. Proceedings of 2004 IEEE/RSJ international conference on Intelligent Robots and Systems, 2004

[Sturman and Zeltzer, 1994] D.J. Sturman, and D. Zeltzer, “A Survey of Glove-Based Input”, IEEE Computer Graphics and Applications, Vol. 14, pp-30-39, 1994.

[Prusak et al., 2007] A. Prusak, O. Melnychuk, H. Roth, I. Schiller, R. Koch. Pose estimation and map building with a Time-Of-Flight-camera for robot navigation. Dynamic 3D imaging workshop, Heidelberg, Germany, 2007

[Wang et al., 2008] Using human body gestures as input for gaming via depth analysis.

[Zhenyao Mo, J. P. Lewis, Ulrich Neumann, 2005] SmartCanvas: A Gesture-Driven Intelligent Drawing Desk. IUI ’05: 10th International Conference on Intelligent User Interfaces, ACM, 2005