REAL TIME GESTURE RECOGNITION IN 3D SPACE USING SELECTED CLASSIFIERS

Łukasz Gadomer[1], Marcin Skoczylas[2]

Bialystok University of Technology, Faculty of Informatics, ul. Wiejska 45A, 15-315 Bialystok, Poland
[1]E-mail: m.skoczylas@pb.edu.pl
[2]E-mail: lukasg7@gmail.com

Abstract:
In this paper, authors propose a solution to track gestures of hands in 3-dimensional space that can be inserted into a CAVE3D environment. Idea of gestures recognition system is described and the results of research made on a recorded gesture data. In this study three selected classifiers to resolve this problem have been tested and results compared.

Keywords: 3-dimensional space; gesture recognition; CAVE environment

INTRODUCTION

In this paper, authors propose a solution to track gestures of hands in 3-dimensional space that can be inserted into a CAVE3D environment. Solution is based on the use of gestures for this purpose. A system consists of two main parts: a tool for gesture recognition and a graphical environment in which the user can see objects moved in three dimensional space and control them using gestures. Authors describe the idea of recognition system and present the results of research made on a recorded gesture data which contains 12 gestures, 80 repeats each, which gives summary 960 instances of gestures. In this study three selected classifiers to resolve this problem have been tested and results compared.

1. RELATED WORK

The problem of gesture recognition, especially in three dimensional space, is well known. There exist many other works related to this research where authors show different ideas of solving this problem. Some of them concentrate only on body parts motion (mainly hands), other use also different kind of information about gesture, for example received from images (like hand posture and shape). In [1] authors were using theory of random propagation and formulated gesture recognition problem as an $\ell 1 \ell 1$-minimalization. Their system operated on 3-axis accelerometer. Similar accelerometer was used by authors in [2]. It was transmitting collected signals to personal computer by a bluetooth protocol. Gesture data was reduced to 8-bit numbers and recognized using algorithm based on sign sequence and template matching. Description of successful attempt of using genetic programming to recognize gestures was presented in [3]. A position invariant gesture recognition real-time algorithm based on dynamic time wrapping was described in [4]. Another idea of gesture recognition was using bayesian networks. Method proposed in [5] is using such networks and is based not only on dynamic hand moves, but also hand posture.

A popular way of obtaining gesture data is using depth images. Such kind of images are possible to obtain for example from Microsoft Kinect sensor [6], which also was used as a data collecting device in research presented in this paper. Many authors also used

the same sensor in their works. In [7], authors were using depth images to detect gestures, which were classified by decision forest made of several multiclass support vector machine algorithms. Another popular way of performing gesture recognition with Kinect is using Hidden Markov Model. Authors in [8] checked how it works and tried to improve HMM-based solution. In [9] another HMM-based method of recognizing single hand gestures in three dimensional space is described.

In our publication we present our novel idea of solving the problem of gestures recognition.

2. GESTURE RECOGNITION METHOD

2.1. Requirements

To understand what are the requirements of a proper gesture recognition let's describe a hypothetical scenario. The user is at the center of a region of some virtual representation of three-dimensional space – this can be for example a CAVE3D environment. In front of the user and in his left and right, big screens are located that view images, resembling a 3D space (see Figure 1). The user can perform gestures that can be interpreted by a gesture recognizer to achieve predefined actions. Such an action can be for example a new object creation. When such gesture is recognized, the user should see a new object on the front screen. User can also perform other actions, for example he can move an existing object using different gesture, drop the object by another one, and so on. After taking a look at such scenarios, there are several basic requirements of a gesture recognition algorithm:

- gesture recognition should be performed in a real time,
- ability to recognize at least a few different gestures,
- ability to define a new gesture,
- ability to learn defined gestures.

Authors noticed that the problematic part of such approach is the need of a real time processing of detected gestures. The user should be able to perform gesture at any time, without worrying whether the program will recognize it or is now processing previous data.

The tool inserted into CAVE3D environment should also allow recognition of different gestures. For example, there should exist at least gestures to: create, drop or release, move, delete objects, but also control the space anchor point (where the user is positioned). Also, we can consider creating a number of other gestures to achieve more advanced tasks, for example modifications of objects shape (bending, scaling) and many others.

In overall, depending on requirements, the user should be able to define his own set of gestures that will be responsible for specific functionalities. This can be achieved by allowing the user to record a series of gestures and marking them individually by a defined label. Recorded gestures are the training data set for classifiers.

Fig. 1. Cave3D, source: http://www.bialystokonline.pl/gfx_artykuly/201211/67014.jpg, 18 Feb. 2014

2.2. Gesture tracking equipment

For the purpose of gesture tracking, authors decided to utilize a well-known Microsoft's solution called Kinect [10]. Kinect has a camera for detecting the field depth, which provides support for the third dimension – camera captures gray scale images of the field, in which the intensity parameter defines depth. Kinect has also the ability to track human body skeleton, so that fact allowed authors to focus on the gesture recognition algorithm rather than image analysis.[1]

Motion detection with the help of the Kinect controller is based on the detection of characteristic points of a whole person body in its range. Despite this, au-

[1] Note, that Kinect has also a very basic and limited set of gestures that can be automatically recognized. That feature was not used in this work.

thors limited gesture tracking to the points representing the user's hands only.

2.3. Data representation.

The data has to be represented in a normalized way that can be properly interpreted by the classifier. This representation should be resistant to minor differences between successive repetitions of gestures, the distance of the user from the camera, etc. Therefore, it is not useful to record position of the hand directly.

Considering above, the authors decided to store a relative representation of successive recorded samples, not direct positions. Each sample consists of three numbers - dimensions: X, Y, Z. Each subsequent sample is the distance from the previous sample within each dimension. Subsequent samples represent the direction and speed of movement of the hand in three directions in relation to the previous recorded sample. As a result, the entire gesture consists of a set of values (see Fig. 2).

$X_2 - X_1$	$Y_2 - Y_1$	$Z_2 - Z_1$	$X_3 - X_2$	$Y_3 - Y_2$	$Z_3 - Z_2$	•••	$X_n - X_{(n-1)}$	$Y_n - Y_{(n-1)}$	$Z_n - Z_{(n-1)}$

Fig. 2. Representation of a recorded gesture consisting of n samples, source: by author

2.4. Implementation of gesture recognition in real time.

To achieve gesture recognition in real-time an author's own method of detection of the beginning and end of the gesture was utilized. The method is based on the assumption that it is not relevant when the user begins to perform gesture - what counts is the moment when there's an end of its execution. Termination of the gesture occurs when the user pauses the movement of the hand for a set number of recorded samples. The detected gesture is a set of samples (a motion) performed just before the acquisition had stopped.

To solve the problem of recognition authors decided to utilize classifiers. Selection of the classifier is crucial, considering that not only the accuracy can change, but also speed of learning and recognition processes. Three classifiers have been evaluated in this research to find one that fits best in our experimental requirements:

- Artificial feed-forward neural network with back-propagation learning (Multilayer Perceptron, MLP) [11]

In this work, authors used a network, where the input represented by samples of a given gesture (and therefore the length of the input is equal to 3 (dimensions) times the number of samples). Network consists of one hidden layer having a length

$$\frac{\text{the length of the input layer} + \text{the length of the output layer}}{2}$$

$$\frac{\text{the length of the input layer} + \text{the length of the output layer}}{2}$$

and the output layer with a length equal to the number of possible gestures to recognize. Each output represents one class of the gestures, a number in the range of $[0,1][0,1]$that represents validity of associated gesture to that output. Gesture is considered recognized, when value of the output is close to the value of 1.

The neural net is known to properly learn non-linear solutions on ill-defined problems, but the training procedure becomes slow on large datasets.

- Support Vector Machine, SVM [12]

SVM maps input space into high-dimensional feature space constructing a hyperplane that divides two classes of objects. Function that performs that mapping is known as a SVM kernel function. The training example set $\{(\overrightarrow{x_1}, y_1) \dots (\overrightarrow{x_k}, y_k)\}\{(\overrightarrow{x_1}, y_1) \dots (\overrightarrow{x_k}, y_k)\}$ represents input space and $y \in \{-1, 1\} y \in \{-1, 1\}$ represents two classes, is mapped into feature space in which the mapped training examples are linearly separable. Divided space can be used for classification of new data objects and the best separation is achieved by a hyperplane that has the largest distance to the nearest training points of all classes. Generally speaking, maximize:

$$W(\alpha) = \sum_{i=1}^{k} \alpha_i - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{k} \alpha_i y_i \alpha_j y_j K(\overrightarrow{x_i}, \overrightarrow{x_j})$$

subject to

$$\sum_{i=1}^{k} \alpha_i y_i = 0, \alpha_i \in [0, C], i = 1, \dots, k$$

Where $K(\overrightarrow{x_i}, \overrightarrow{x_j}) K(\overrightarrow{x_i}, \overrightarrow{x_j})$ is a dot-product SVM-kernel function. Many different kernel functions were proposed in the past: linear, polynomial, sigmoid and the Radial Basis Function (RBF): $K(\vec{x}, \vec{x'}) = e^{-\gamma \|\vec{x} - \vec{x'}\|^2}$ $K(\vec{x}, \vec{x'}) = e^{-\gamma \|\vec{x} - \vec{x'}\|^2}$ are most popular.

Authors utilized a multi-class SVM [13] for the purpose of multiple gestures recognition, thus the output of the classifier is a single integer in the range $[0, number\ of\ gestures - 1]$ $[0, number\ of\ gestures - 1]$ that represents the gesture recognized.

- Radial Basis Network [14]

RBN differs from standard neural network in its hidden layers structure. Radial basis function network has only one hidden layer which uses radial basis func-

Tab. 1. Sample research result: SVM, linear kernel, without normalization.

Gesture	O	\	/	(	)	8	|	~	^	<	-	>
O	98,09	0,00	0,00	0,00	0,00	1,66	0,00	0,00	0,00	0,26	0,00	0,00
\	0,00	99,61	0,00	0,00	0,00	0,00	0,39	0,00	0,00	0,00	0,00	0,00
/	0,00	0,00	91,85	2,19	0,24	0,00	5,60	0,00	0,00	0,00	0,12	0,00
(	0,00	1,23	0,12	84,94	0,00	0,00	3,70	0,00	0,00	10,00	0,00	0,00
)	0,00	0,00	4,19	0,00	88,18	0,00	2,59	0,00	0,00	0,00	0,00	5,05
8	5,36	0,00	0,00	0,12	0,00	92,52	0,00	2,00	0,00	0,00	0,00	0,00
|	0,00	1,94	4,52	10,06	3,35	0,00	76,65	0,00	0,13	1,94	1,16	0,26
~	0,00	1,52	0,00	0,00	0,00	0,00	0,38	92,25	5,84	0,00	0,00	0,00
^	0,00	3,49	0,36	2,53	0,36	0,00	0,48	1,08	90,49	0,24	0,96	0,00
<	0,00	2,34	0,99	7,52	0,37	0,86	4,32	0,00	0,00	83,60	0,00	0,00
-	0,13	0,00	7,29	0,00	0,26	0,00	0,13	0,00	0,00	0,00	91,82	0,38
>	1,24	0,00	1,74	0,00	15,65	1,49	0,87	0,00	0,00	0,00	2,98	76,02

source: prepared by authors

Tab. 2. Results of measurements obtained using selected classifiers

Classifier	SVM	NN	RBN
Mean recognition accuracy	91,74	90,65	82,92
Mean standard deviation of the accuracy	5,89	5,86	8,27
Calculations Time	274,69	5193,10	376,88

source: prepared by authors

tions as activation functions. The input layer and the output layer structure is designed the same way as neural network. It also uses the same kind of normalization and gives results in the same format.

3. RESULTS AND DISCUSSION

For accuracy testing purposes and to achieve best results in gestures recognition problem, authors performed several experiments with collected gestures data. These include 12 gestures recorded as values in format described in section 3.3. The study checked the effectiveness of the classifiers, primarily to identify accuracy in different gestures recognition, as well as speed of calculations. Collected data were randomly divided into training data (80%) and testing data (20%). Classifiers have been taught using training data, then testing data have been classified. Such operation was repeated 50 times, then averaged.

Example results of the experiments are presented in Tab. 1. Each column represents different gesture and the rows contain respective recognition accuracies as percentages. The numbers on the main diagonal of the table show correct identification of the gesture, others show the recognition error.

Results are summarized in Tab. 2.

On the basis of these observations, authors conclude that for the given problem of classification of gestures the best results are obtained using the SVM classifier. SVM gave the best results in the shortest possible time and was characterized by a low diversity of the results achieved in subsequent repetitions of the test.

CONCLUSIONS

The gesture recognition can be achieved for insertion into the CAVE3D system using a Kinect device and method described in this paper. Gestures can be successfully recognized using classifiers. Selection of the appropriate classifier to solve the problem of gestures recognition is very crucial. Based on studies presented in this paper it can be concluded that the decision should fall on the SVM classifier. It should be emphasized however, that the results could be slightly different for a different set of gestures or other selected classifiers parameters, but taking into account the specific nature of the problem and carefully conducted study by authors, the result of them can be considered as representative for a given research problem.

REFERENCES

1. **Akl A., Chen Feng, Valaee S. (2011)**, *A Novel Accelerometer-Based Gesture Recognition System*, "Signal Processing", IEEE Transactions on, vol.59, no.12, pp.6197-6205.
2. **Ruize Xu, Shengli Zhou, Li W.J. (2012)**, *MEMS Accelerometer Based Nonspecific-User Hand Gesture Recognition*, "Sensors Journal", IEEE, vol.12, no.5, pp.1166-1173.
3. **Li Liu, Ling Shao (2013)**, *Synthesis of spatio-temporal descriptors for dynamic hand gesture recognition using genetic programming,* in: *Automatic Face and Gesture Recognition (FG), 10th IEEE International Conference and Workshops on*, vol., no., pp.1-7, 22-26 April.
4. **Bodiroza S., Doisy G., Hafner V.V. (2013)**, *Position-invariant, real-time gesture recognition based on dynamic time warping*, "Human-Robot Interaction" (HRI)*, 8th ACM/IEEE International Conference on*, vol., no., pp.87,88, 3-6 March.
5. **Shiravandi S., Rahmati M., Mahmoudi F. (2013)**, *Hand gestures recognition using dynamic Bayesian networks*, in: *AI & Robotics and 5th RoboCup Iran Open International Symposium (RIOS), 2013 3rd Joint Conference of* , vol., no., pp.1,6, 8-8 April.
6. **Suarez J., Murphy R.R. (2012)**, *Hand gesture recognition with depth images: A review*. "RO-MAN", IEEE , vol., no., pp.411-417, 9-13 Sept.
7. **Miranda L., Vieira T., Martinez D., Lewiner T., Vieira A.W., Campos M.F.M. (2012)**, *Real-Time Gesture Recognition from Depth Data through Key Poses Learning and Decision Forests*, in: *Graphics, Patterns and Images (SIBGRAPI), 2012 25th SIBGRAPI Conference on* , vol., no., pp. 268-275, 22-25 Aug. 2012.
8. **Youwen Wang, Cheng Yang, Xiaoyu Wu, Shengmiao Xu, Hui Li (2012)**, *Kinect Based Dynamic Hand Gesture Recognition Algorithm Research*, in: *Intelligent Human-Machine Systems and Cybernetics (IHMSC), 4th International Conference on*, vol.1, pp.274-279, 26-27 Aug.
9. **Zhong Yang, Yi Li, Weidong Chen, Yang Zheng (2012)**, *Dynamic hand gesture recognition using hidden Markov models*, in: *Computer Science & Education (ICCSE), 2012 7th International Conference on*, pp.360-365, 14-17 July.
10. http://www.microsoft.com/en-us/kinectforwindows/, 12 Feb. 2014.
11. Autor ?? (1995), *Introduction to artificial neural networks*, in: *Electronic Technology Directions to the Year 2000, 1995. Proceedings.*, pp. 36-62, 23-25 May.
12. **Burges C. J. C. (1998)**, *A tutorial on support vector machines for pattern recognition*, Data Min. Knowl. Discov. 2, pp. 121–167.
13. **Lee Y., Lin Y., Wahba G. (2001)**, *Multicategory Support Vector Machines*, "Computing Science and Statistic", 33.
14. **Daqi G., Chen Mingming, Li Yongli (2005)**, *A single-layer radial basis function network classifier and its applications*, in: *Neural Networks, 2005. IJCNN '05. Proceedings. 2005 IEEE International Joint Conference on* , vol.2, no., pp.1045-1050 vol. 2, 31 July-4 Aug.
15. http://www.bialystokonline.pl/gfx_artykuly/201211/67014.jpg, 18 Feb. 2014

Acknowledgments: This work was supported by the MB/WI/3/2012 and S/WI/1/2013.