Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 1071–1074
ISBN 978-83-60810-51-4

Gaussian Hand Gesture Recognition Based MobilityDevice Controller

Vidas Raudonis
Kaunas University of Technology,
Studentu 48A-303, Kaunas,
Lithuania
Email: vidas.raudonis@ktu.lt

Rytis Maskeliunas
Kaunas University of Technology,
Studentu 48A-303, Kaunas,
Lithuania
Email: rytis.maskeliunas@ktu.lt

Paulius Lengvenis
Kaunas University of Technology,
Studentu 48A-303, Kaunas,
Lithuania
Email: paulius.lengvenis@ktu.lt

***Abstract*—The development and investigation of alternative mobility device control is presented in this work. The system uses 2D visual information, which is acquired from an ordinary web-cam, and controls the electrical drives of the mobility device by tracking and recognizing the gestures of the hand. Hand tracking is achieved by using an algorithm, which combines two methods: a statistical Gaussian method and a discrete Fourier transformation. Proposed algorithm is adaptive and flexible allowing utilizing unique gesture commands which depend on person's motor abilities. Experimental investigation proves the stable robustness, performance and high accuracy of the proposed mobility controller.**

I. Introduction

HAND gesture recognition methods combined with machine vision, virtual reality, device control, etc. could serve as a natural way of interaction especially for the users of sign based languages (i.e. deaf). A successful development of reasonably intuitive interfaces, e.g. wheelchair controlled by hand gesture instructions as in [1] can further reduce the link between a human and technology.

The technical implementation can range from HMMs [2] as a gesture is a continuous motion on a sequential time series to pseudo 3D HMMs [3], utilization of Freeman Chain Code [4] to the analysis of hand shape parameters [5], differences in image entropy [6] and feature extraction methods from support Vector machines [7] to multi-layer perceptron [8]. Most allow achieving more than 90% recognition accuracy but also built to serve a specific task in specific environment using predefined postures for users to control devices [9] thus introducing a problem of forcing the users to learn and familiarize with the system of gestures. Modern processors and mini controller allow building this functionality directly into video capture devices–smart cameras [10].

Our approach utilizes the goal of no end-user costs by using standard equipment (e.g. built-in webcams) and combines two methods: a statistical Gaussian method and discrete Fourier transformation. The proposed algorithm is adaptive and flexible enough allowing utilizing unique gesture commands each depending and configurable to a specific person's motor abilities.

This research was funded by a grant (No. MIP-037/2011) from the Research Council of Lithuania.

II. Hand Contour Extraction

Hand gesture recognition is based on a well-known background subtraction method that highlights the difference between a real time image (**I**) and a background (**B**). The output of the subtraction procedure is the mapping image (**M**) in which the region of the foreground objects are extracted. Background subtraction is widely used to detect moving object from static cameras. It is usually regarded as one of the most important steps in applications such as traffic monitoring, human motion capture, recognition and video surveillance, etc. [11].

In our case the foreground object is the picture of a hand taken using an ordinary web-camera. The abstract illustration of the hand extraction is shown in the figure 1. The background image is modeled during the training stage by capturing a static image that does not contain the objects of interest. The image is segmented in N regions by color. Each region is described by the Gaussian distribution function.

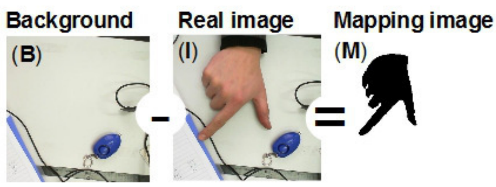

Fig. 1 The abstract background subtraction procedure

Gaussian mixture model is a robust background subtraction method and is widely used ever since it was proposed [12], [13]. The Gaussian background model describes the color distribution of the image region. The distribution is evaluated computing mean value μ_c of the color channel c and covariance matrix K_i between color channels of the ith region in the image. The parameters are computed as follows:

$$\mu_c(u,v) = \frac{1}{N}\sum_i^N B_c(u,v,i) \tag{1}$$

$$K = \begin{bmatrix} V_{11} & V_{21} & V_{31} \\ V_{12} & V_{22} & V_{32} \\ V_{13} & V_{23} & V_{33} \end{bmatrix}, V_{ij} = cov(C_i, C_j) \tag{2}$$

where V_{ij} is the covariance value between two color channels and $\boldsymbol{C} = [\boldsymbol{R}\ \boldsymbol{G}\ \boldsymbol{B}]$ is the matrix of the pixels values. Each region is described by Gaussian distribution function using formulas below.

$$\Psi = \exp[(\boldsymbol{c} - \boldsymbol{\mu}_i)^T \boldsymbol{K}_i^{-1}(\boldsymbol{c} - \boldsymbol{\mu}_i)] \quad (3)$$

$$p_i(\boldsymbol{c}) = \frac{1}{(2\pi)^{\frac{l}{2}}\sqrt{\det(\boldsymbol{K}_i)}} \cdot \Psi \quad (4)$$

where $c = [r\ g\ b]$, r, g, b – pixels values, $l = 3$ is the number of the color channels. Any new image pixel is recognized as background pixel if it satisfies the condition (5), otherwise pixels belongs to the foreground object.

$$M(u,v) = \begin{cases} 0 & if\ \sum_{i=1}^{M} p_i < \theta \\ 1 & otherwise \end{cases} \quad (5)$$

where $M(u,v)$ is mapping matrix. The resulting mapping image is filtered from the noise and the user's hand contour is extracted. We assume that the background does not vary in indoor condition and hence it is captured a-priori in the beginning of the system usage. The background model is not updated during the usage of the mobility controller.

III. Gesture Recognition and Mobility Control Algorithm

Proposed mobility controller uses ordinary web-camera which is mounted on the mobility device (electrical wheelchair) using flexible holder. The camera is directed in such way that it captures whole user hand (see figure 2). There is no matter in what background the gesturing is carried out. Our algorithm can process the background of a t-shirt, a wooden desk and other materials. The acquired hand images can be processed on ordinary laptop or tablet computer using standard low resolution built-in webcams (i.e. in a normal usage position - screen angled at a user).

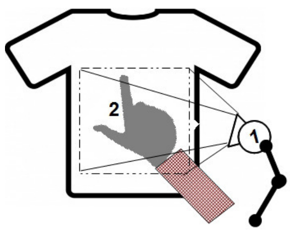

Fig. 2 Controllers web-cam position: α – web-cam's capturing region horizontal to xy plane; 1 – vertically down directed web-cam; 2 – user's hand. Hand movement directions are horizontal to α plane

Four motion control commands (γ) are used in the work, i.e., go "FORWARD", "BACKWARD", "LEFT" and "RIGHT". Possible hand gesture examples are shown in the figure 3. Each control command is identified by hand gesture and it is recognized using coefficients of discrete Fourier transformation (DFT). A unique hand gesture must be selected for each command, type of which depends on the user's motor-abilities and needs.

Fig. 3 Possible variations of hand gestures used in this work

DFT is applied on the normalized x and y coordinates of the hand contour, which are obtained after background subtraction procedure. The coordinates of the contour are selected in clockwise direction as it shown in the figure 4. The scale invariant gesture recognition is achieved using normalized data. The contour coordinates are normalized into interval from 0 to 1 according formulas (6) and (7).

$$x_i' = [x - min_k(x)]/[max_k(x) - min_k(x)] \quad (6)$$

$$y_i' = [y - min_k(y)]/[max_k(y) - min_k(y)] \quad (7)$$

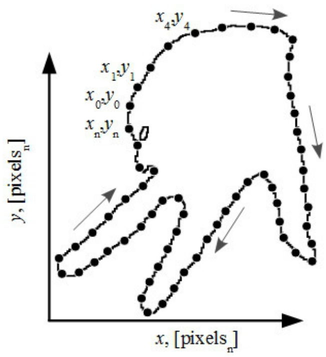

Fig. 4 The selection of the object contour coordinates for DFT

Discrete Fourier transformation of the contour coordinates x, y is expressed using formulas (8) and (9).

$$x_k(f) = \sum_{k=0}^{n-1} x_k exp[-(2\pi i/n)jk] \quad (8)$$

$$y_k(f) = \sum_{k=0}^{n-1} y_k exp[-(2\pi i/n)jk] \quad (9)$$

where $j=0,\dots,n-1$, $x(f)$ and $y(f)$ is harmonics of the frequency spectrum, n – is the total number of the contour coordinates, i – imaginary unit. The frequency components $x(f)$ and $y(f)$ of the Fourier transformation are used as discriminating features for classification purpose.

The standard Euclidean distance is computed between i^{th} template coefficients $\mathbf{F}^i$ and new coefficients $\mathbf{F}$ obtained during real time processing. The similarity measure is expressed according to formula (10).

$$\varepsilon_i = \sqrt{\sum_{k=0}^{n}\left(F_k - F_k^i\right)^2} \quad (10)$$

where k is index of the element in $\mathbf{F}$ vector.

The classification of the gesture class is achieved by searching for a minimal value between all estimated similarity measures ($\boldsymbol{\varepsilon}$). The gesture is recognized as γ class when estimated similarity value satisfies the condition (11).

$$\gamma = \begin{cases} i & \min_i(\varepsilon) < \theta_\varepsilon \\ 0 & otherwise \end{cases} \tag{11}$$

where θ_ε is the threshold value of the similarity measure.

Proposed mobility control algorithm is shown in figure 5. The algorithm begins from the training stage, in which the static background is evaluated. After calculations represented in section 2, Gaussian background mixture model is formed. When the model is formed, picture with a hand in front of a camera is captured and regarding to Gaussians background mixture model, pixels are classified in two regions: first region – background pixels, second – foreground object (user's hand region) pixels. After extracting hand region, the frequency coefficients are estimated from the contour coordinates. The coefficients are stored in the systems memory and they are used in the usage stage as a template assigned to certain motion command.

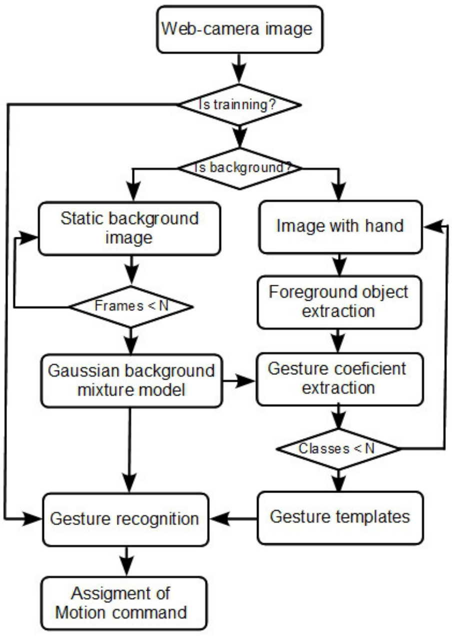

Fig. 5 The algorithm of the hand gesture based control

During system usage, the hand region is extracted from real time images based on Gaussian background model. The frequency coefficients are estimated from the resulting mapping image and they are compared with stored templates. In principle the system recognizes the gesture itself based on features of a frequency domain.

The performance and effectiveness of the proposed algorithm depends on several parameters, i.e., background pixel threshold θ and similarity threshold θ_ε. Proper selection of thresholds makes system more robust to the noise and more accurate. More about experimental research on the control algorithm is explained in the next section.

IV. Experimental Evaluation and Results

Experimental investigation involves few types of research: first accuracy evaluation of the system and secondly recognition rate of the control command. 10 persons were participating in the experiments. They were asked to train the system for themselves using individual hand gestures. Afterwards, during the testing stage, a user was asked to repeat one gesture at the time that was shown to him in a random manner. The contour coordinates of the hand gesture and the processing rate was recorded and analyzed. About 500 data samples per participant were collected during a testing stage.

The functional relationship between a command recognition accuracy and a similarity threshold level is shown in the figure 6. Relationship curve shows that the highest recognition value 98% ± 0.3% is achieved, when $\theta_\varepsilon = 160$. Lowest similarity threshold value ensures that no misclassified control command is accepted as appropriate user generated command. However, such threshold value requires more effort from the user to replicate the gesture as closes as possible to desired template gesture.

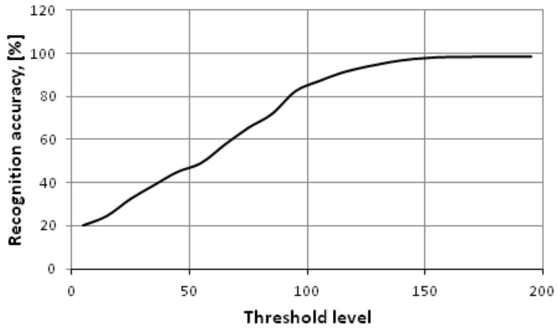

Fig. 6 The functional relationship between overall command recognition accuracy and similarity threshold value

The functional relationship between each control command and similarity threshold value is shown in the figure 7. Each curve is marked with a different marker and represents the recognition accuracy of the certain command. Highest recognition accuracy of 100% is reached when the command "STOP" is classified. There is no difference what kind of θ_ε should be used, because this command is generated when there are no foreground objects on the background. Highest threshold value 160 is required for the command "RIGHT" where accuracy was 93% ± 1.2%. The recognition accuracy of 98% ± 0.5% was reached for all other control commands

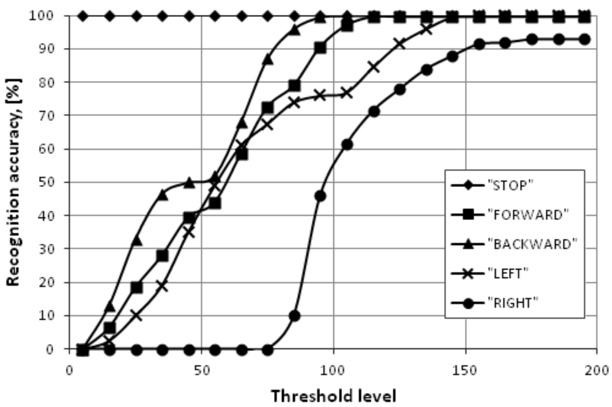

Fig. 7 The functional relationship between each control command and similarity threshold value

using the same threshold value. The lower recognition accuracy of the command "RIGHT" can be explained be the form of gesture. Sometimes the user was frustrating to replicate exact copy of template form.

Another analytical research was performed on the gathered experiment data, from which it was interesting to know how many frequency components are needed to achieve proper accuracy rate. The results have shown that more than three first frequency coefficients do not give higher than 98% recognition rate. It is explained by computing influence ration (see fig. 8). It is noticed, that first component has higher weight comparing other components. Therefore classification decision is made according to first 3 harmonics.

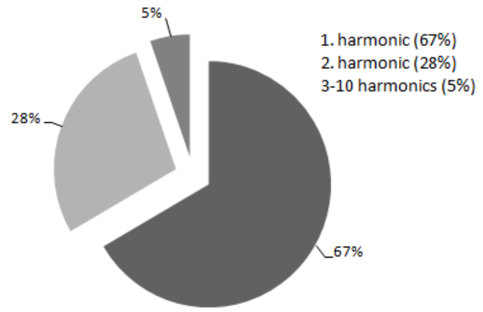

Fig. 8 Given trajectory and acquired during experimental testing

All experiment participants were asked to repeat the template hand gesture that was chosen randomly and displayed on screen. Template gesture was changed when a system correctly recognized the gesture shown by a user. The task execution time was recorded and analyzed after experiment. The functional relationship between task execution time and participant is shown in the figure 9.

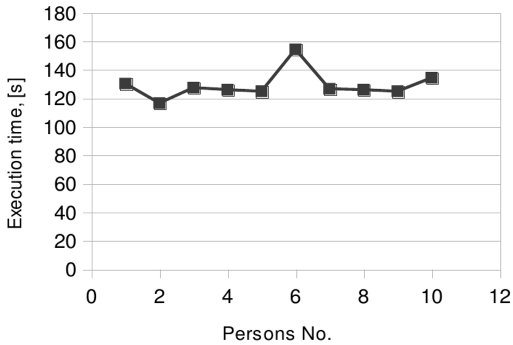

Fig. 9 Functional relationship between task execution time and participant number

The average task execution time of 128±6.3 seconds was reached during the experiments. It was noticed, that users learnt to work with the system and were able to increase not only the command generation rate but also the recognition accuracy. The control algorithm is able to process in an average of 10 frames per second. Such processing rate is sufficient to control electrical mobility device in a relatively slow speed.

V. Conclusions

A mobility control algorithm using a statistical Gaussian method enabled to automatically select gesture commands during training stage. Our system differentiates itself from the others as it does not count the number of fingers, but evaluates the shape of a hand (a whole gesture) in space. Experimental results showed that algorithm can perform accurately and fast. It was noticed, that users learned to work with the system and were able to increase not only the command generation rate but also the recognition accuracy. The control algorithm was able to process in average 10 frames per second. Such processing rate is sufficient to control electrical mobility device in relatively slow speed.

The overall recognition accuracy of ~98% was reached for the analyzed control commands using the same threshold value. There was no difference what kind of similarity threshold should be used, because the commands were generated when there were no foreground objects on the background. The lower recognition accuracy of some commands can be explained by the form of gesture (more so of a difficulty). Sometimes the user frustration (training and adaption problem) was notice while replicating exact copy of template form. Overall the system accuracy could be further improved by scale and rotation invariant features, which should be further evaluated based on physiological hand properties.

References

[1] R. Posada-Gomez et al., "A Hands Gesture System Of Control For An Intelligent Wheelchair," *in Proc. Electrical and Electronics Engineering*, ICEEE 2007, 2007, pp. 68-71

[2] Byung-Woo Min, Ho-Sub Yoon, Jung Soh, Yun-Mo Yang, T.Ejima, "Hand gesture recognition using hidden Markov models Systems," *in Proc. Man, and Cybernetics*, vol. 5, 1997, pp. 4232-4235.

[3] Nguyen Dang Binh, T. Ejima, "Real-Time hand Gesture Recognition Using Pseudo 3-D Hidden Markov Model," *in Proc. Cognitive Informatics, ICCI' 2006*, 2006, pp. 820-824.

[4] Juanfang Xu, Shouqian Sun, Yuanwu Shi, Zhanxun Dong, "Implementation of Digital Chime-bell Interaction System Driven by Hand Gesture," *in proc. Pervasive Computing and Applications, ICPCA 2008*, vol. 1, 2008, pp. 264-268.

[5] Doe-Hyung Lee, Kwang-Seok Hong, "A Hand gesture recognition system based on difference image entropy," *in Proc. Advanced Information Management and Service (IMS) – 2010*, 2010, pp. 410-413.

[6] M. Panwar, "Hand gesture recognition based on shape parameters," *in Proc. Computing, Communication and Applications (ICCCA) – 2012*, 2012, pp. 1-6.

[7] N.H. Dardas, N.D. Georganas, "Real-Time Hand Gesture Detection and Recognition Using Bag-of-Features and Support Vector Machine Techniques," *iEEE Trans. Instrumentation and Measurement*, vol. 60, iss. 11, pp. 3592-3607, 2011.

[8] Chenglong Yu, Xuan Wang, Hejiao Huang, Jianping Shen, Kun Wu, "Vision-Based Hand Gesture Recognition Using Combinational Features," *in Proc. Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP) – 2010*, 2010, pp. 543-546.

[9] Wen-Hui Chen, Yu-Hong Lin, Shun-Jie Yang, "A generic framework for the design of visual-based gesture control interface," *in Proc. Industrial Electronics and Applications (ICIEA) – 2010*, 2010, pp. 1522-1525.

[10] Z. Zivkovic et al., "Toward low latency gesture control using smart camera network," *in Proc. Computer Vision and Pattern Recognition Workshops CVPRW '08*, 2008, pp. 1-8.

[11] M. Izadi, P. Saeedi, "Robust region-based background subtraction and shadow removing using color and gradient information," *in Proc. 19th International Conference on Pattern Recognition (ICPR)*, 2008, pp. 1-5.

[12] D.-S. Lee. "Effective gaussian mixture learning for video background subtraction". *iEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, pp. 827-832, 2005.

[13] T. Bouwmans, F. E. Baf, B. Vachon, "Background modeling using mixture of Gaussian for foreground detection: A survey," *Recent Patents on Computer Science*, vol. 1(3), pp. 219-237, 2008.