American Sign Language Recognition: Algorithm and Experimentation System

Martyna Lagozna, Milosz Bialczak, Iwona Pozniak-Koszalka, Leszek Koszalka(✉), and Andrzej Kasprzak

Department of Systems and Computer Networks,
Wroclaw University of Science and Technology, Wroclaw, Poland
m.lagozna@gmail.com, milosz.bialczak@gmail.com,
{iwona.pozniak-koszalka, leszek.koszalka,
andrzej.kasprzak}@pwr.edu.pl

Abstract. The objective of this paper is an attempt to take part in the fast development of computer science, especially machine learning, in the field of finding the tools for supporting the disabled people. Speech-and–hearing impaired people are the part of our society and it would be a great convenience both for them and for speaking people, who would have an opportunity for a better communication using computer technology. In this paper, the results of the research concerning recognition of sign language have been provided. The research includes experimenting with the images transformations and the usage of different learning and feature detecting algorithms to obtain the best quality of signs recognition. In addition, the recommendations based on results of experiments can be applied in practical issues, for example, in determining hands rotations used in sign language, which can improve the accuracy of recognition.

Keywords: Sign language · Recognition · Machine learning · Algorithm · Experimentation system

1 Introduction

Nowadays, using computer technology in solving medical issues became a common practice. Fast development and the growing popularity of machine learning approach allowed for creating technologically advanced applications and complex algorithms, which are aimed at helping people. Sign language recognition is an interesting and a significant issue strictly connected with both machine learning and medical technology. People with disability hearing loss account for 5% of the whole society [1]. In the past, these people had no technical possibilities to facilitate their lives and communication ways. Now, it is still challenging but more possible, so our motivation is the contribution in solving this issue.

In this work, we concentrate on experiments with different features extraction and learning algorithms, paying a particular attention to image processing. Moreover, the scope of this work includes also checking the impact of background and hands rotation on the received accuracy basing on the usage of two completely different data sets.

N. T. Nguyen et al. (Eds.): ICCCI 2019, LNAI 11683, pp. 676–685, 2019.
https://doi.org/10.1007/978-3-030-28377-3_56

The main objective is implementing an own algorithm taking into account that at the current stage of research we concentrate on single letters from American Sign Language (ASL) alphabet. One of the crucial assumptions, considered in this work, is recognition of images taken with average-quality camera. The algorithm, which has been implemented, might be available for everyone, for example, by a build-in camera in mobile phone or computer. The algorithm should be given an image on input and the result should be returned as recognized letter on string type. Therefore, we consider the following problems:

- Finding crucial features on the picture.
- Emphasizing hand features.
- Finding satisfying way of learning the algorithm.

The paper is organized as follows. Section 2 is devoted to related work. The problem is formulated in Sect. 3. The proposed algorithm is described in Sect. 4. This section is divided on four subsections: preprocessing, features extraction, post-processing, and learning. The experimentation system is presented in Sect. 5, including experiment design with data sets used. The obtained results of experiments and comments are presented in Sect. 6. In Sect. 7, appear the conclusion and plans for further research in the area.

2 Related Work

In the last decade there were many of researches in the field of sign language recognition. In very important work [2] was shown that gesture recognition gives an opportunity for interacting with machine without any mechanical devices. The authors of this paper have proposed the recognition system, which understands human gestures. Another interesting method for the recognition of human gesture is presented in [3]. The method is based on neural network and stochastic computing. However, the authors confirmed that most of the technologies in the area of human gesture recognition was power consuming.

In the last years, Microsoft Kinect sensors give the possibility for lower cost of Human-Computer-Interaction (HCI). It allows for proposing a new sensible solutions, e.g., in [4], a novel multi-sensor fusion framework was proposed; in [5] a Novel CNNs with multiview and fusion for American Sign Language recognition; in [6], an attempt of hand gesture recognition, using the SIFT algorithm is described. The authors of this paper showed that SIFT algorithm is working good with the standard ASL database but also with home-made database.

The paper [7] has reported the interesting idea of three - stage translation system. The first stage is responsible for the communication with the neuromorphic camera DVS sensors. The second stage regards feature extraction of the events generated by the DVS – it gives the opportunity for a presentation of the digital image processing algorithms developed in software. The third stage consists in the classification of the ASL alphabet basing on the implemented artificial neural network.

The very important work [8], in our opinion, is focused on the recognition of single letters of American alphabet. The authors of this paper used a hierarchical mode-seeking method for localization hand joint positions under kinematic constraints. They also used special Microsoft Kinect camera what allowed them to achieve high accuracy.

3 Problem Formulation

The objective of this work is to create an algorithm, which can recognize static gestures of single letters from American Sign Language alphabet. The single task is a classification of hand's photos:

Given: the images of gestures on photos.

To find: the recognized letters on string type.

Such that: the index of performance defined by (1) is of the highest possible accuracy. The accuracy is the index of performance expressed by the formula (1).

$$\text{Accuracy} = (N_c)/(N) \tag{1}$$

where Nc is the number of the corrected recognitions, N is the total number of recognitions.

For a single classification task the Support Vector Machine - a set of methods for supervised learning [9] has been used. In this case, the decision function is defined by the following formula:

$$sgn\left(\sum_{i=1}^{n} y_i \alpha_i K(x_i, x) + \rho\right)$$

where x and y are training vectors, $x_i \in \mathrm{R}^{\mathrm{p}}, i = 1, 2\ldots, n, y \in \{1, -1\}^n$ and $K(x_i, x_j)$ is the kernel [10].

4 Algorithm

As it was mentioned, creating an algorithm requires taking into consideration the following tasks: (i) Finding crucial features on the picture; (ii) Emphasizing of hand features; (iii) Finding satisfying way of learning the algorithm.

The structure of the recognition algorithm concerns step by step performing. In Fig. 1, the proposed model in the form of the general data flow scheme is presented.

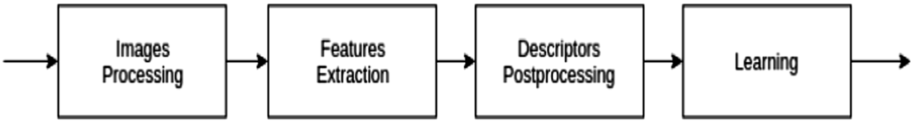

Fig. 1. Data flow model.

4.1 Preprocessing

In the first step, the appropriate preparation of the images should be made. We have applied two methods of the image modification, including Gaussian Blur for reducing image noise and Anisotropic Filtering for sharpening the main shape of the gesture. These methods and their properties are presented in [11] and [12], respectively. Using of these methods allows avoiding recognition of excess points and enhancing quality of the considered image. These methods and their properties are presented in [11] and [12], respectively.

In Fig. 2, the pre-processing data flow is shown in the form of block-scheme.

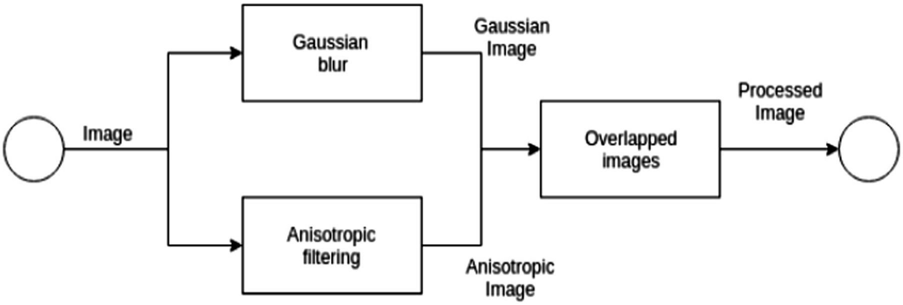

Fig. 2. Model of preprocessing.

4.2 Features Extraction

The following tree different features extractors (available in *ski-image* library [10]) have been utilized.

- CENSURE (The Center Surround Extrema) extractor, which is characterized by fast computing time [13]. However, this extractor is able to find usually only key points but not descriptors [14].
- BRIEF (The Binary Robust Independent Elementary Features) extractor is an efficient feature point descriptor [15]. Unfortunately, it is unable to sensible dealing with images rotation.
- ORB (The Oriented FAST and Rotated BRIEF) extractor is a combination of the modified BRIEF extractor (presented in [16]) and FAST key point feature detector (described in [17]). This extractor allows for images rotation and possesses ability to fast computing.

4.3 Post-processing

Between steps of features extraction and classifier learning, the extracted data are processed and improved to become more reliable. This step, called post-processing, is presented in the form of the block-scheme in Fig. 3.

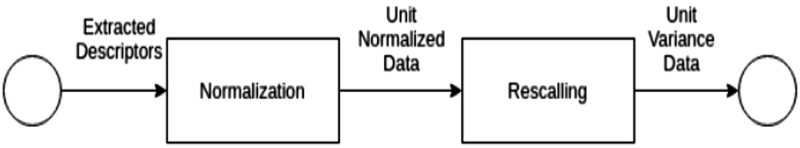

Fig. 3. Model of post-processing data flow.

Two processes have been implemented:

- NORMALIZE, which rescales the vector for each sample to ensure the unit norm, independently of the distribution of the samples [18].
- STANDARD SCALER, which can remove mean and scale the data to unit variance.

However, the outliers have an influence when computing the empirical mean and standard deviation which shrink the range of the feature values. Standard Scaler unfortunately cannot guarantee balanced feature scales in the presence of outliers [18].

4.4 Learning

Support Vector Machines (SVMs) are a set of related methods for supervised learning, applicable to both classification and regression problems. A SVM classifiers create a maximum-margin hyperplane that lies in a transformed input space and splits the example classes, while maximizing the distance to the nearest cleanly split examples. The parameters of the solution hyperplane are derived from a quadratic programming optimization problem [9]. In the paper, the following methods are used:

- VECTOR – CLASSIFIER, which learns with use of vector of descriptors retrieved from the picture and the information about the correct letter on the image. This method has a significant disadvantage. Vectors received by the classifier must have exactly the same length which as a consequence, extorts cutting the longer ones and ignoring the ones that are too short.
- POINTS – CLASSIFIER, which learns by processing descriptor after descriptor and the information about related letter from the picture. This way of learning makes the algorithm independent from the data vector length.
- COMBINED - CLASSIFIER, when the outputs from both vector and points classifiers are taken into consideration. This classifier learns by taking those outputs and the information about correct answer.

5 Experimentation System

The input of the experimentation system is an image. The output states the result returned as recognized letter on string type.

5.1 Datasets

Two different data sets have been used. The first one has been downloaded from the website of Silesian University of Technology [19]. An example of data base element is shown in Fig. 4.

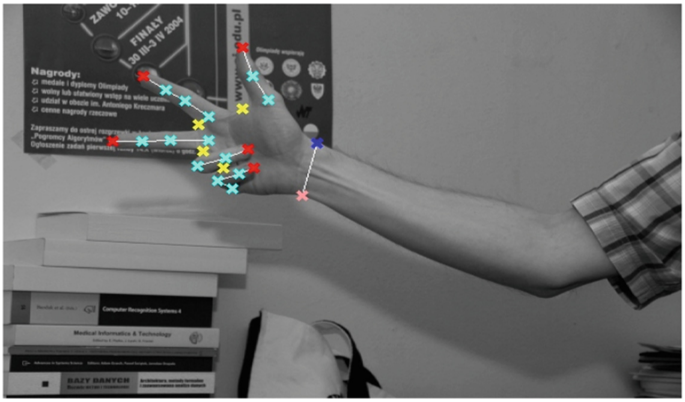
Nagrody:
FINAŁY
30 III-3 IV 2004
Computer Recognition Systems 4
Medical Informatics & Technology
BAZY DANYCH

Fig. 4. An element of database with hand gestures [19].

The data base consists of 899 images of gestures from American alphabet from A to Y excluding J. Letters ‘Z’ and ‘J’ are moving gestures and it is impossible to show them on a single picture. This data set contains images with uncontrolled background and lightning conditions, different angle of hand rotations form observer perspective and different resolution. Images are oriented both vertically and horizontally.

The second data set used in the experiment is a self-made set of images which has been created in possibly similar lightning condition. On each image only hands at uniform, plain background has been shown. An example is shown in Fig. 5.

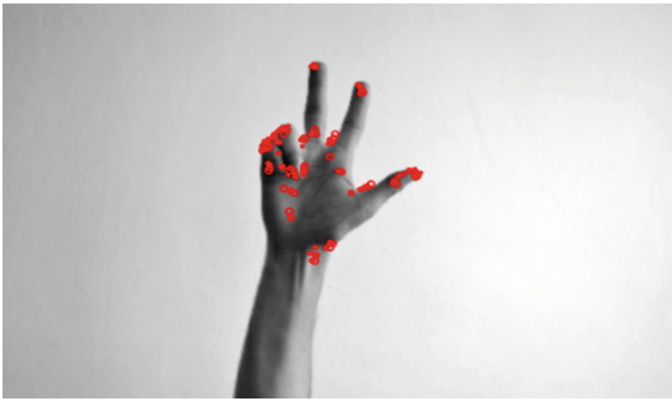

Fig. 5. An element of our own database with hand gestures.

Each image has exactly the same height, resolution and all of them are oriented horizontally. The set contains images of gestures of all American sign alphabet also with ‘J’ and ‘Z’ excluded.

5.2 Process of Experiment

During the research the successive steps have been added to the algorithm. At first the model has been learned by pure descriptors. Next anisotropic filtering has been added. Then Standard Scaler and Gaussian Blur have been implemented. The last step was adding normalization to the output.

The above described steps have been applied to three different features extractors: BRIEF, CENSURE and ORB. On each extractor three methods of learning has been tested: by points, vectors and combined. As a metric for evaluation of classification the accuracy defined by Eq. (1) has been used.

6 Research

6.1 Complex Experiment

Table 1 contains the results of the accuracy (the percentage values of correct recognition) at every step of the algorithm. These results were obtained for the complex experiment composed of single experiments. For simple experiment we fixed the considered various elements of preprocessing, features extraction, learning, and post-processing. In some cases, we decided to abandon taking into account some features extractors because of their unsatisfying accuracy in comparison to the others (see elements in Table 1 with “no data”).

Table 1. Results when adding next steps to the algorithm.

Features extractions	Learning	Descriptors	Anisotropic filtering	Standard Scaler	Gaussian blur	Normalize
ORB	Points	51.72%	60.34%	91.38%	94.84%	96.55%
ORB	Combined	13.79%	13.79%	12.07%	13.79%	12.07%
ORB	Vector	13.79%	13.79%	13.79%	22.41%	25.86%
CENSURE	Points	20.69%	22.41%	15.52%	12.07%	–
CENSURE	Combined	13.79%	18.97%	15.52%	5.17%	–
CENSURE	Vector	8.61%	1.72%	0.00%	1.72%	–
BRIEF	Points	15.52%	–	–	–	–
BRIEF	Combined	15.52%	–	–	–	–
BRIEF	Vector	0.00%	–	–	–	–

It can be observed in Table 1, that among all used features extractors ORB occurred to be the most effective for the considered problem. Probably, its ability to dealing with rotations had a big impact on the obtained results.

Therefore, the aim of the next experiments has been collecting more detailed information about elements (versions of preprocessing, learning and post-processing) connected with ORB features extraction.

6.2 Experiment for ORB Testing

In Fig. 6, the detailed results of experiments for ORB features extraction are shown.

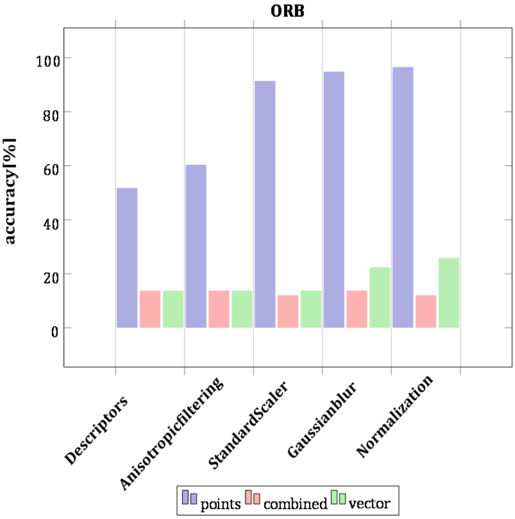

Fig. 6. Influence of versions of different steps of the algorithm with ORB.

The best method of learning the classifier occurred to be learning by points. Furthermore, successively added algorithms at following steps significantly improved the result. Adding Standard Scaler was a crucial step and had considerable impact on the final accuracy.

We can be satisfied with the performance of the algorithm for gestures on the plain background. In case of adding distorting background, e.g., check shirt (we tested some cases), the performing of algorithm should be improved.

In Table 2, are shown the results produced by the final (all steps) version of the algorithm tested on three different data sets (mentioned in Sect. 5.1):

Set #1 - The set of the numbers (digits) prepared by us.

Set #2 – The set of the letters prepared by us.

Set #3 - The set of images of gestures regarding alphabet taken from [19].

Table 2. Results when adding next steps to the algorithm

Dataset	ORB		
	Points	Combined	Vector
Set #1	96.55%	12.07%	25.86%
Set #2	72.90%	3.23%	12.26%
Set #3	27.36%	3.48%	2.49%

It may be observed once more that learning by using points classifier leads to better accuracy of the recognition.

7 Final Remarks

7.1 Conclusion

In the version, presented in this paper, the recognition of static gestures on plain background seems to work fine. To sum up, we managed to implement an algorithm which can pose a good basis to continue extended research.

7.2 Plans for Further Research

In the nearest future we plan to focus on further developing the algorithm to reach the improvement of accuracy. Our main goal is to ensure the possibilities of experiments:

- For recognition of moving signs.
- For recognition with special classifier only for mistaken gestures.

We should also concentrate on designing and implementing the extended experimentation system which allows making multi-stage experiments in automatic manners, following the rules described in our works [20] and [21].

Acknowledgement. This work was supported by the statutory funds of the Department of Systems and Computer Networks, Wroclaw University of Science and Technology, Wroclaw, Poland.

References

1. Deafness and hearing loss. World Health Organization. http://www.who.int/newsroom/fact-sheets/detail/deafness-and-hearing-loss. Accessed Nov 2018
2. Sridevj, K., Sundarambal, M., Muralidharan, K., Rathinadurai, J.L.: FPGA implementation of hand gesture recognition system using neural networks. In: 2017 11th International Conference on Intelligent Systems and Control (ISCO), January 2017
3. Wang, X., Chen, W., Ji, Y., Ran, F.: Gesture recognition based on parallel hardware neural network implemented with stochastic logics. In: 2016 International Conference on Audio, Language and Image Processing, July 2016

4. Kumar, P., Gauba, H., Roy, P., Dogra, D.P.: Coupled HMM-based multi-sensor data fusion for sign language recognition. Pattern Recogn. Lett. **86**, 1–8 (2017)
5. Tao, W., Leu, M.C., Yin, Z.: American sign language alphabet recognition using convolutional neural networks with multi-view augmentation and inference fusion. Eng. Appl. Artif. Intell. **76**, 202–213 (2018)
6. Nicholai, M.: Alphabet recognition of ASL: a hand gesture recognition approach using SIFT algorithm. Int. J. Artif. Intell. Appl. **4**(1), 105–115 (2013)
7. Rivera-Acosta, M., Ortega-Cisneros, S., Dominguez, J.R., Sandoval, F.: American sign language alphabet recognition using a neuromorphic sensor and an artificial network. Sensors **17**(10), 2176 (2014)
8. Dong, C., Leu, M.C., Yin, Z.: American sign language alphabet recognition using microsoft kinect. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 44–52 (2015)
9. Shmilovici, A.: Data Mining and Knowledge Discovery Handbook, pp. 231–247. Springer, USA (2005)
10. https://scikitlearn.org/stable/modules/svm.html. Accessed Jan 2019
11. Flusser, J., Farokhi, S., Hoschl, C., Zitova, B., Pedone, M.: Recognition of images degraded by Gaussian blur. IEEE Trans. Image Process. **25**(2), 790–806 (2016)
12. Yang, G.Z., Burger, P., Firmin, D.N., Underwood, S.R.: Structure adaptive anisotropic image filtering (1995)
13. Schmidt, A., Kraft, M., Fularz, M., Domagala, Z.: Comparative assessment of point feature detectors and descriptors in the context of robot navigation. J. Autom. Mob. Rob. Intell. Syst. **7**, 11–20 (2013)
14. Lee, P., Timmaraju, A.S.: Learning binary descriptors from images (2012)
15. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: BRIEF: binary robust independent elementary features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15561-1_56
16. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: an efficient alternative to SIFT or SURF, November 2011
17. Rosten, E., Drummond, T.: Machine Learning for High-Speed Corner Detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 430–443. Springer, Heidelberg (2006). https://doi.org/10.1007/11744023_34
18. http://scikitlearn.org/stable/auto_examples/preprocessing/plot_all_scaling.html. Accessed Dec 2018
19. http://sun.aei.polsl.pl/~mkawulok/gestures/. Accessed Jan 2019
20. Bogalinski, P., Davies, D., Koszalka, L., Pozniak-Koszalka, I., Kasprzak, A.: Evaluation of strip nesting algorithms: an experimentation system for the practical users. J. Intell. Fuzzy Syst. **27**(2), 611–623 (2014)
21. Hudziak, M., Pozniak-Koszalka, I., Koszalka, L., Kasprzak, A.: Multi-agent pathfinding in the crowded environment with obstacles: algorithms and experimentation system. J. Intell. Fuzzy Syst. **32**(2), 1561–1578 (2017)