Sign Language Numeral Gestures Recognition Using Convolutional Neural Network

Ivan Gruber[1,2](✉), Dmitry Ryumin[3], Marek Hrúz[1,2], and Alexey Karpov[3]

[1] Faculty of Applied Sciences, Department of Cybernetics,
UWB, Pilsen, Czech Republic
grubiv@kky.zcu.cz
[2] Faculty of Applied Sciences, NTIS, UWB, Pilsen, Czech Republic
mhruz@ntis.zcu.cz
[3] SPIIRAS, St. Petersburg, Russia
dl_03.03.1991@mail.ru, karpov@iias.spb.su

Abstract. This paper presents usage of convolutional neural network for classification of sign language numeral gestures. For requirements of this research, we created a new dataset of these gestures. The dataset was recorded via Kinect v2 device and it consists of recordings of 18 different people. Only depth data-stream was used in our research. For a classification task, there was utilized classic VGG16 architecture and its results were compared with chosen baseline method and other tested architectures. Our experiment on classification showed the great potential of neural networks for this task. We reached recognition accuracy 86.45%, which is by more than 34% better result than chosen baseline method.

Keywords: Sign language · Image recognition · Classification
Neural network · Assistive robot · Computer vision

1 Introduction

The task of increasing the level of automation and robotization in all spheres of human activity is one of the keys in the modern information society. In this connection, scientists and leaders of developed countries, as well as developing countries, in cooperation with world scientific centers and companies pay attention to technologies for an effective, natural and universal interaction of a person with computers and robots.

Currently, interactive information systems are used in the areas of social services, medicine, education, robotics, the military industry, community service centers, to interact with people in various situations. In addition, robotic assistants are finding more and more widespread which are simple and intuitive in use. Compared to industrial robots that are only able to repeat predetermined tasks, robot-assistants are aimed at interacting with people in the performance

© Springer Nature Switzerland AG 2018
A. Ronzhin et al. (Eds.): ICR 2018, LNAI 11097, pp. 70–77, 2018.
https://doi.org/10.1007/978-3-319-99582-3_8

of tasks. In this case, many classical interfaces are not enough. Instead, more intuitive and natural approaches for human interfaces are needed (speech [2], gestural [3], multimodal [1,4–6], etc.). For example, gestures can transmit simple commands to a robot that will carry unambiguous meaning and are effective at some distance from the robot and in noisy conditions when speech is ineffective.

It is also known that deaf people have limited capabilities when communicating with the hearing. Therefore, there is necessity to develop recognition of sign language technologies for deaf people. In addition to large world companies, national research centers are also working in this direction. Scientists from the American Institute of Robotics at Carnegie Mellon University are working on a system that can analyze the language of the body and gestures up to the point of the fingers [7]. A number of researchers rightly point out that serious differences in the semantic-syntactic structure of written and sign languages do not yet allow an unambiguous translation of the sign languages. Therefore, there are currently no fully automatic sign language translation systems. To create a complete model, it is necessary to make a semantic analysis of written phrases, and this is still possible only at a superficial level because of imperfections in text analysis algorithms and knowledge bases.

At present, Microsoft provides a tool in the form of a sensor-rangefinder Kinect for the development of systems with the possibility of recognizing the sign language [8,9], which allows us to obtain a three-dimensional video stream of information in the form of a depth map or a three-dimensional cloud of points. MS Kinect 2.0 provides simultaneous detection and automatic tracking of up to 6 people at the distance of 1.2–3.5 meters from the sensor. In the software, a virtual model of human's body is presented as a 3D skeleton of 25 points.

The paper is organized as follows: in Sect. 2 we introduce used dataset; in Sect. 3 we presented used processing methods and discuss software implementation details; in Sect. 4 we describe the experiment and show obtained results; and finally in Sect. 5 we draw a conclusion and outline our future research.

2 Dataset

In this paper, we use our own dataset of numeral hand gestures. We recorded 10 gestures of a hand performing numbers from American Sign Language. These gestures are, to some extent, universal and many other sign languages use them. We recorded 18 people performing the gestures with 5 repetitions using a commercial depth sensor Kinect v2. For the purpose of this research, we use only the depth data-stream. Each repetition of a gesture consists of a movement of the hand into the performing space, where the hand stops and a static gesture representing a number from zero to nine is shown. To obtain only the frames with the gesturing static hand we implemented our own semi-automatic labeling algorithm. Since Kinect provides us with a skeletal model of a human it is easy to follow the movement of the hand by tracking a joint representing the palm of the hand. Some time synchronization is needed but the position of the joints changes linearly between consecutive frames and thus the proper position of the palm joint in the time of depth map acquisition is easily interpolated. The palm joint location is considered as a center of a 3D box containing the hand. Since Kinect uses orthographic projection in the depth axis the depth of the 3D box is always constant and has been chosen to be 200 mm. However, the xy-axes use projective transformation and thus the size of the 3D box in this image plane has to be adapted according to the depth of the palm joint. We use the same size of the box in both the x and y axis computed using the formula:

$$M = \frac{\alpha \cdot \mathrm{depth}_{\mathrm{max}}}{\mathrm{depth}}, \tag{1}$$

where M is the size of the box in pixels, $\mathrm{depth}_{\mathrm{max}}$ is the maximal depth of the capturing device (in our case 8000), depth is the measured depth in the palm joint location, and α is a scale coefficient, which we experimentally chose equal 15. All the 3D boxes are resized to 96 × 96 pixels and the depth in the box is normalized from 0 to 1. These resulting hand depth images are manually labeled as either one of the numeral gestures or as a non-informative gesture simply named background. Furthermore, if the performer used his/her left hand for gesturing the resulting hand depth image was flipped.

Next, the hand depth images were augmented to help with the training of the neural network. We used random translation and planar rotation to obtain the final dataset. Each hand image was translated four times by a randomly selected 2D vector representing the planar translation. The numbers were drawn from a uniform distribution in an interval $[-12; 12]$ px. The rotation was performed three times by a randomly selected angle from the interval $\pm 20^{\circ}$. In total, the dataset consists of 130843 depth images of hands. Some examples of the dataset are shown in Fig. 1.

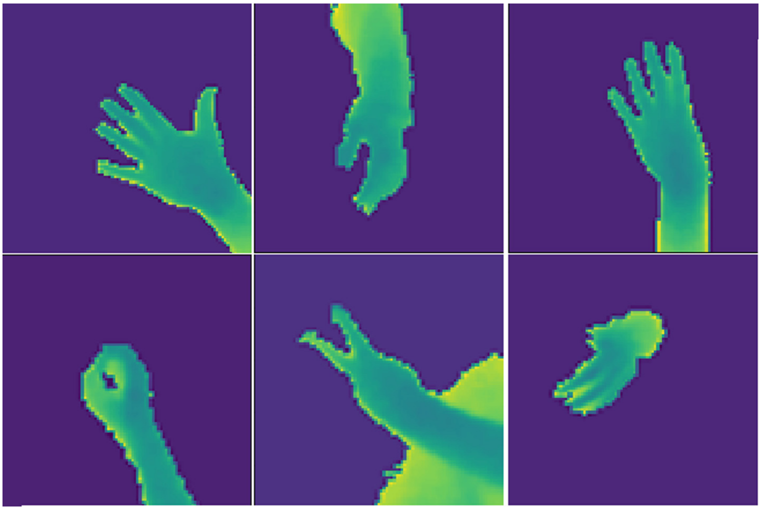

Fig. 1. Example of the dataset. From top left to the bottom right: gesture for no. 5, background, no. 4, no. 0, no. 2, and background again.

3 Methods

Due to the neural networks improvements since 2012 [10], most hand-crafted feature descriptors in image classification, if enough data available, become inferior in comparison with machine-learned ones. In this paper we tested two approaches on the task of numeral gesture classification.

First, we calculated Histogram of Gradients (HoGs) [11] for all the data. Each HoG's cell had 16×16 pixels and each block had 3×3 cells. With this settings we obtained feature vector with dimension of 1152 for each image.

These HoGs were used to train standard Support Vector Machine (SVM) [12] classifier with RBF kernel. This setup is used as our baseline method.

Second, we trained convolutional neural network with modified VGG16 architecture [13]. This architecture belongs to the golden standard among neural network architectures used for image classification, especially for tasks with a lower amount of training data. The exact network configuration we used is shown in Table 1.

Table 1. Modified VGG16 architecture.

Layer	Properties	Activation fcn
Conv1a	3×3, stride 1, filters 64	ReLU
Conv1b	3×3, stride 1, filters 64	ReLU
Max pool	2×2, stride 2	
Conv2a	3×3, stride 1, filters 128	ReLU
Conv2b	3×3, stride 1, filters 128	ReLU
Max pool	2×2, stride 2	
Conv3a	3×3, stride 1, filters 256	ReLU
Conv3b	3×3, stride 1, filters 256	ReLU
Max pool	2×2, stride 2	
Conv4a	3×3, stride 1, filters 512	ReLU
Conv4b	3×3, stride 1, filters 512	ReLU
Max pool	2×2, stride 2	
Conv5a	3×3, stride 1, filters 512	ReLU
Conv5b	3×3, stride 1, filters 512	ReLU
Max pool	2×2, stride 2	
Fully-connected1	1024D	ReLU
Dropout	dropout rate 0.5	
Fully-connected2	11D	SoftMax

4 Experiments and Results

In our experiment, we evaluate the performance of methods on the classification task of numeral gestures, i.e. we want to classify the input image into one of 11 classes (10 numerals and background).

Due to the amount of data, we use cross-validation with 10 different cross-validation settings. For each of them, our dataset was split into two subsets - train set, and test set, where each test set contained data from 4 speakers and train set rest of them.

As a benchmark method SVM classifier trained on HoGs with dimension of 1152 was used. The average recognition accuracy among all the cross-validation settings was 52.31% ± 3.51% on the test data.

Table 2. Comparison of the recognition accuracy results from individual cross-validations (CVs).

CV split	Accuracy, %
No. 1	83.37
No. 2	83.72
No. 3	92.04
No. 4	87.01
No. 5	88.11
No. 6	83.74
No. 7	88.85
No. 8	84.80
No. 9	84.16
No. 10	88.73

For neural network architecture, we come out from VGG16 architecture, however, we cut one of the fully-connected layers entirely and the second one was resized from 4096 to 1024, i.e. this layer provides feature vector with size 1024, which is comparable with the dimension of used HoG descriptor.

The neural network was trained with 20 epochs with mini-batch size 64 and with initial learning rate = 10^{-3}. The learning rate was decreased after 10 epoch to 10^{-4}. For updating network parameters standard SGD optimization with momentum = 0.9 and weight decay = 5×10^{-4} was used. As a loss function, standard Softmax loss was used. Neural network was implemented in Python using Keras deep learning library [14]. The average recognition accuracy among all the cross-validation setting was 86.45% ± 2.93%, which is by more than 34% better than used baseline method. The results from the individual cross-validations can be found in Table 2.

The results show us, that not each cross-validation is equally difficult. This phenomenon is probably caused by the different ability of each speaker to perform numeral gestures properly. Further, it can be caused by inconsistency during labeling among our annotators. You can see some examples of misclassification in Fig. 2.

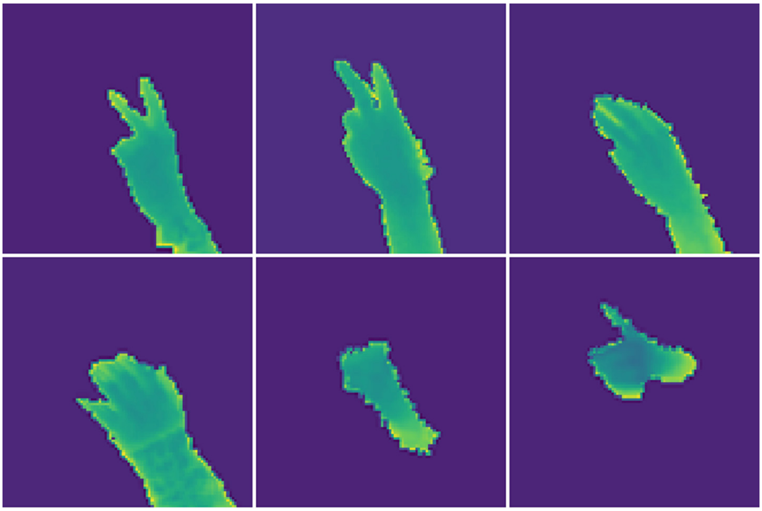

Fig. 2. Examples of misclassification. From the top row left to right: classified as 3 instead 2, classified as 7 instead 2, classified as background instead 3. Bottom row: classified as background instead 5, classified as background instead 6, classified as background instead 7. Last two are examples of wrong labels in our dataset.

We also tested some other neural network architectures during our initial experiments. All of them were tested only on cross-validation split number 1 with the same training settings as our modified VGG16. For comprehensive comparison see Table 3. CNN3 × 32 is a simple architecture with three convolutional layers, whereas each of them has 32 filters with kernel size 3 × 3, and two fully-connected layers (one with size 1024 and the last one with size 11 as a classification layer). CNN3 × 32b is almost the same architecture, however, the number of filters of the second convolution is doubled and the third one is quadrupled. CNN3 + 5 + 7 has three convolutions and 2 fully-connected layers again, however, each convolution has different size of the kernel (3, 5, and 7 respectively). All of the convolutional layers have 32 kernels again. Last tested architecture CNN3 + 5 + 7b utilizes the same approach as CNN3 × 32b, e.g. the number of kernels in convolutions is appropriately increased.

Overall, the experiment shows the superiority of the approach utilizing a neural network and machine-learned features over the classic HoG+SVM approach. Moreover, we reached very promising results, which show us a great potential of neural networks for gesture and sign language recognition.

Table 3. Comparison of baseline method, modified VGG16 and other tested architectures in terms of recognition accuracy.

Method	Accuracy, %
HoG+SVM	50.12
VGG16_1024	**83.72**
CNN3 × 32	71.23
CNN3 × 32b	73.18
CNN3 + 5 + 7	74.42
CNN3 + 5 + 7b	75.11

5 Conclusion and Future Work

Sign language recognition and gesture recognition is very demanded task in the modern world. We believe it is essential for next generation of robotic assistants, as well as an assistive tool for deaf people. In this paper, we show the great potential of the usage of neural networks for this task. Moreover, we reach very promising recognition results on our own dataset of sign language numeral gestures. We believe that with some minor modification of our neural network architecture, with more augmentations, and with bigger training set, we can reach flawless results.

In our future research, we would like to extend our dataset with recordings from more speakers. Additionally, we would like to add some other important sign language gestures.

Acknowledgments. This work is financially supported by the Ministry of Education and Science of the Russian Federation, agreement No. 14.616.21.0095 (reference RFMEFI61618X0095) and the Ministry of Education of the Czech Republic, project No. LTARF18017. The work has been also supported by the grant of the University of West Bohemia, project No. SGS-2016-039. Moreover, access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum provided under the programme "Projects of Large Research, Development, and Innovations Infrastructures" (CESNET LM2015042), is greatly appreciated.

References

1. Ivanko, D.V., Karpov, A.A.: An analysis of perspectives for using high-speed cameras in processing dynamic video information. SPIIRAS Proc. **44**(1), 98–113 (2016). https://doi.org/10.15622/sp.44.7
2. Karpov, A., Kipyatkova, I., Zelezny, M.: Automatic technologies for processing spoken sign languages. Procedia Comput. Sci. **81**, 201–207 (2016)
3. Ryumin, D., Karpov, A.A.: Towards automatic recognition of sign language gestures using kinect 2.0. In: Antona, M., Stephanidis, C. (eds.) UAHCI 2017. LNCS, vol. 10278, pp. 89–101. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58703-5_7

4. Karpov, A., Krnoul, Z., Zelezny, M., Ronzhin, A.: Multimodal synthesizer for Russian and Czech sign languages and audio-visual speech. In: Stephanidis, C., Antona, M. (eds.) UAHCI 2013. LNCS, vol. 8009, pp. 520–529. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39188-0_56
5. Karpov, A., Ronzhin, A.: A universal assistive technology with multimodal input and multimedia output interfaces. In: Stephanidis, C., Antona, M. (eds.) UAHCI 2014. LNCS, vol. 8513, pp. 369–378. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07437-5_35
6. Ivanko, D., et al.: Using a high-speed video camera for robust audio-visual speech recognition in acoustically noisy conditions. In: Karpov, A., Potapova, R., Mporas, I. (eds.) SPECOM 2017. LNCS (LNAI), vol. 10458, pp. 757–766. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66429-3_76
7. Cao, Z., Simon, T., Wei, S.-E., Sheikh, Y.: Realtime multi-Person 2D pose estimation using part affinity fields. In: CVPR (2017)
8. Shibata, H., Nishimura, H., Tanaka, H.: Basic investigation for improvement of sign language recognition using classification scheme. In: Yamamoto, S. (ed.) HIMI 2016. LNCS, vol. 9734, pp. 563–574. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-40349-6_55
9. Guo, X., Yang, T.: Gesture recognition based on HMM-FNN model using a Kinect. J. Multimodal User Interfaces **11**, 1–7 (2016)
10. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proceedings of Advances in Neural Information Processing, pp. 1106–1114 (2012)
11. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), pp. 886–893 (2005)
12. Hearst, M.A.: Support vector machines. IEEE Intell. Syst. **13**, 18–28 (1998)
13. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR (2014)
14. Chollet, F.: Keras (2015). https://keras.io