2017 IEEE International Conference on Robotics and Automation (ICRA)
Singapore, May 29 - June 3, 2017

Cyrillic Manual Alphabet Recognition in RGB and RGB-D Data for Sign Language Interpreting Robotic System (SLIRS)

Nazgul Tazhigaliyeva[1], Nazerke Kalidolda[1], Alfarabi Imashev[2], Shynggys Islam[2], Kairat Aitpayev[2], German I. Parisi[3], Anara Sandygulova[1*]

Abstract— **Deaf-mute communities around the world experience a need in effective human-robot interaction system that would act as an interpreter in public places such as banks, hospitals, or police stations. The focus of this work is to address the challenges presented to hearing-impaired people by developing an interpreting robotic system required for effective communication in public places. To this end, we utilize a previously developed neural network-based learning architecture to recognize Cyrillic manual alphabet, which is used for fingerspelling in Kazakhstan. In order to train and test the performance of the recognition system, we collected four datasets comprising of static and motion RGB and RGB-D data of 33 manual gestures. After applying them to standard machine learning algorithms as well as to our previously developed learning-based method, we achieved an average accuracy of 93% for a complete alphabet recognition by modeling motion depth data.**

I. Introduction

Hearing-impaired people around the world communicate via a sign language, which uses gestures to express meaning and intent that include hand-shapes, arms and body, facial expressions and lip-patterns [1]. Similar to spoken languages, each country or region has its own sign language of varying grammar and rules, leading to a few hundreds of sign languages that exist today [2]. In addition, many deaf-mute people are not able to understand a written spoken language.

This paper describes the SLIRS project, which aims to develop an interpreting robotic system of a sign language tailored for Kazakhstan. Having consulted our local hearing-impaired community on their needs, SLIRS's first priority is to develop an interpreting system of the sign language vocabulary required for effective communication in Centers for Public Services (CPS).

Since SLIRS's first objective is to automatically recognize sign language utilizing multi-sensory input from mono and depth cameras embedded on the robot, the focus of this paper's work is to employ RGB and RGB-D data for fingerspelling recognition. Fingerspelling is mainly used to spell proper nouns, scientific and foreign borrowed terms, and other words lacking a sign representation. To this end, we utilized our previously developed method [3] and applied it to a new application of recognizing 33 manual gestures of Cyrillic alphabet. Although the learning approach is an incremental modification of the learning algorithm proposed by [4] (2015), where we just modify the labeling function to return the specific letter value, this paper reports the results that outperform standard machine learning approaches of an improved accuracy of 93% (previous performance was 77.2% [5]) for Cyrillic manual alphabet recognition, which serves as the first milestone in the progress of the SLIRS project.

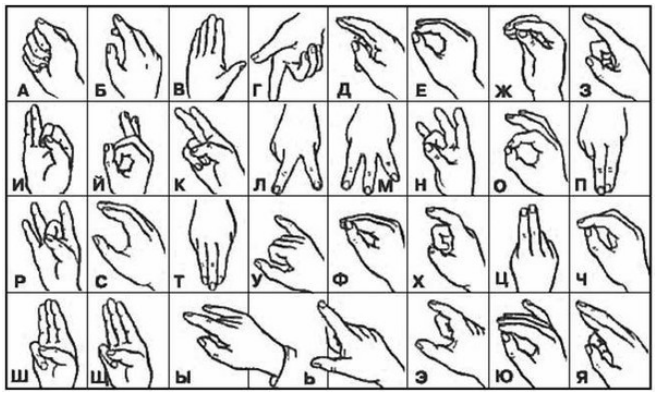


Fig. 1. Sign language of Cyrillic alphabet

To underpin such approach, we collected and annotated four datasets[1] of 33 manual gestures of the Cyrillic alphabet (Fig. 1). In summary, the contributions of this paper are:

- a fully annotated dataset of RGB static data for 33 manual gestures of four professional signers;
- a fully annotated dataset of RGB motion data for 33 manual gestures of four professional signers;
- a fully annotated dataset of RGB-D static data for 33 manual gestures of 30 healthy adults;
- a fully annotated dataset of RGB-D motion data for 33 manual gestures of 30 healthy adults;
- a novel concept of the SLIRS project, to which we subsequently apply our previously developed neural network-based classification algorithm that achieves fairly good classification results.

First, a brief review of the related work is provided. It is followed by the descriptions of our datasets, methodology

[1]Nazgul Tazhigaliyeva, Nazerke Kalidolda and Anara Sandygulova are with the School of Science and Technology, Nazarbayev University, 53 Kabanbay Batyr Avenue, Astana, Kazakhstan {nazgul.tazhigaliyeva, nazerke.kalidolda, anara.sandygulova}@nu.edu.kz

[2]Alfarabi Imashev, Shynggys Islam, and Kairat Aitpayev are with National Laboratory Astana (NLA), 53 Kabanbay Batyr Avenue, Astana, Kazakhstan {alfarabi.imashev, sislam, kairat.aitpayev}@nu.edu.kz

[3]German I. Parisi is with Knowledge Technology Group, Department of Informatics, University of Hamburg, Hamburg, Germany german.parisi@gmail.com

*Corresponding author: anara.sandygulova@nu.edu.kz

[1]https://goo.gl/A718Zv

and discussion of results.

II. Related Work

Research and development on sign language analysis and recognition started with wearable devices such as gloves with sensors and trackers, colored gloves or colored fingers [6]. In contrast, vision-based systems provide a natural way of communicating for deaf people, however it still remains to be a challenging problem for effective hand detection, segmentation, and tracking [2]. Robust gesture recognition is an essential objective for any sign language interpreting system. This section presents related research efforts that have been carried out to address this objective.

According to [7], the main components of any type of sign recognition system include image acquisition, hand localization, pose estimation and gesture classification. Once the image is acquired from a depth camera, it is processed using various *hand localization* methods. The problem of segmentation has been addressed by using depth thresholding for hand isolation [8], [9] and by placing bounds on the number of pixels inspected in the area of detected hand [10], [11]. Some techniques involve predicting the hand location by relating it to other body parts [12].

Temporal and spatial information of hands makes the *hand tracking* possible, which in turn leads to dynamic gesture recognition. The research community most often uses the NITE body tracking middleware in combination with the Kinect SDK [13], [14].

As the next step, various classification algorithms are used to categorize a particular sign or hand gesture. These algorithms take segmented hand images and their tracked trajectories as input and make a prediction.

Malassiotis *et al.* (2002) [15] developed gesture classification of 20 letters from the Greek Sign Language, primarily of numbers from 0-9. Feris *et al.* (2005) [16] addressed the problem of finger occlusion that arise in fingerspelling by introducing a small modification to the capture setup. Keskin *et al.* (2013) [17] utilized *the object recognition by parts* approach to recognize 10 digits of American Sign Language (ASL).

Apart from recognition, sign language and gesture recognition has been utilized for robot control and human-robot interaction. Singh *et al.* (2012) [18] proposed a new approach for robust automated real-time robot control tool using Indian Sign Language. Their technique combines feed forward back propagation neural network (FNN) and Hidden Markov Model (HMM) to deal with dynamic sign language recognition, learning and interpretation of continuous signals. The algorithm was integrated with the HOAP-2 humanoid robot generated in WEBOTS. The proposed system achieved 95.34% of recognition and interpretation accuracy of 21 gestures offline.

Sohn *et al.* (2013) [19] presented a 2-Tier control for a human-robot collaborative tasks. The adult-sized humanoid robot, Hubo, whose lower and upper bodies are controlled separately using data from MoCap and sign language accordingly. The sign language gestures were recorded offline using the MoCap motion capturing system and evaluated by the Monte Carlo learning agent. As a result, the Hubo humanoid robot successfully assisted the human operator in object carrying task.

Luis-Pérez *et al.* (2011) [20] utilized a set of Mexican Sign Language to make the robot perform specific tasks. The system recognizes and interprets 23 signs of the alphabet with the accuracy of 95.8%. Sugiuchi *et al.* (2002) [21] exploited the sign language to control the multi-fingered Dual robot hand in human mimetic approach and to perform paper cutting and chopstick handling. However, the authors report that despite that the sign language interpretation software system could be effectively implemented, there were major limitations in hardware.

Child-size humanoid robots have been exploited for demonstration of signs and other significant components of sign language such as facial expressions and mimicry [22]. Screen avatars have also been used to interpret written English text into ASL [23], [24].

This paper does not focus on advancing the state of the art in hand segmentation, localization and/or tracking, but contributes with its approach of using previously developed gesture classification method to classify our RGB and RGB-D datasets.

III. Data Collection

A. RGB datasets

Two types of RGB (2D) datasets were collected: single-instance and multi-instance datasets.

RGB videos were recorded by four experienced sign language interpreters, 3 female and 1 male. The first interpreter experiences diminished hearing and studied in a specialized institution for people with hearing impairments. The second interpreter is from a family with a deaf parent. The third one is a professional sign language interpreter on local television. The forth interpreter has deaf parents and currently is a professional sign language interpreter in mass media.

Videos were recorded using the Logitech c310 web camera (HD 720p). Video recording took place during the day in standard lightning conditions. The background of the shooting space was of homogeneous green colour and well-lit by the daylight. Collected dataset was processed to the size of 320x240 pixels. Additionally, for the fingerspelling process to look more natural, we generated a vocabulary of short words (4-6 letters). The interpreters were then asked to spell these words using sign language alphabet.

There is a special Kazakh sign language corpus that contains Kazakh language fingerspelling configurations recorded from different angles. The corpus was specifically recorded for Computer Vision and Machine Learning purposes. The whole process of data collection has been done using the corpus as it has build-in video recording and annotation tools. All data used in this work are available through the corpus website[2].

[2]http://kslc.kz

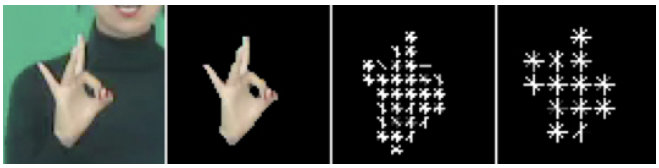


Fig. 2. Process of feature extraction: a) original image; b) segmented hand using graph cut method; the HOG descriptor for the masked hand image with the cell size c) 6x6 and d) 10x10 pixels.

TABLE I

Number of letters recorded for each of the four sign language interpreters (singer IDs are in presented in the first column). There are 33 letters in total.

1: А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й
28	17	17	14	20	33	17	11	18	22	16
К	Л	М	Н	О	П	Р	С	Т	У	Ф
10	22	22	25	28	12	14	20	23	24	9
Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
10	12	14	12	14	15	14	20	14	16	21
2: А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й
57	35	35	29	41	67	35	23	37	45	33
К	Л	М	Н	О	П	Р	С	Т	У	Ф
20	44	45	51	57	25	29	40	47	50	18
Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
21	25	28	27	29	31	29	43	29	31	43
3: А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й
28	17	16	13	20	33	16	11	18	22	16
К	Л	М	Н	О	П	Р	С	Т	У	Ф
10	22	20	25	28	12	14	19	23	25	9
Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
10	12	14	13	14	15	21	14	14	16	21
4: А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й
10	12	11	10	12	23	13	6	15	17	9
К	Л	М	Н	О	П	Р	С	Т	У	Ф
5	11	14	18	20	7	6	14	15	14	7
Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
9	9	7	8	11	12	11	16	10	11	15

Hand detection and feature extraction was performed using the state-of-the-art approach presented by Buehler *et al.* [25]. The skin probability algorithm [26] was used as the first step towards hand detection. Then, skin color elements were labeled and the closest ones to the center of the image were taken as the most probable location of the hand. Finally, "graph cuts" algorithm was applied in order to segment hand pixels from the background where each pixel of the image is considered as a graph node [27]. The feature extraction was done using Histogram of Oriented Gradients (HOG) [28]. The HOG descriptor was similar to [25] with 4 orientation bins, a cell size of 10x10 pixels and block size of 1 cell (see Fig. 2).

The entire dataset of segmented hands consists of 73688 images for relational (i.e. multi-instance) and 2518 images for propositional (i.e. single-instance) sign language alphabet. The size of each image is 60x60 pixels. This has been made for the convenience of feature extraction. The letters "щ" and "й" were not included in single-instance dataset for the reason that they have the same configuration as "ш" and "и" respectively. The amount of letters that were recorded by four sign language interpreters is provided in Table 1.

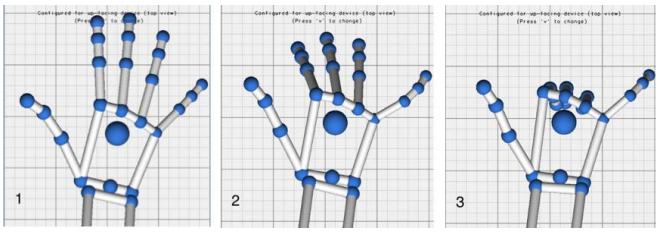


Fig. 3. Fingerspelling in process: 1. Palm initial position 2. Transition between the initial position and the letter 3. Actual letter

B. RGB-D datasets

Similarly, two representations of depth dataset were collected using the Leap Motion sensor. The multi-instance relational dataset was obtained by the method of appending consecutive frames of 3D information of the same letter and the same participant to represent one class. In the single-instance dataset a class is represented by only a single frame of 3D data, letter and participant IDs. For multi-instance dataset the motion data was saved.

The datasets were collected on a regular day for 30 people aged between 20 and 30 years old. The Leap Motion sensor was used to track and record manual gestures data. The participants were invited to the classroom to perform hand gestures in front of the Leap Motion sensor. The sensor was placed horizontally with the X axis pointing to the right, Y axis pointing at the participant and Z axis pointing downwards as opposed to traditional position where Y and Z axes are pointing up and at the participant respectively. More precisely, the sensor was attached to the monitor of the recording computer. The computer was placed at the level appropriate for the participant to perform gestures while standing. No other external modifications were applied to the Leap sensor. Also, throughout the experiment the technical specifications of the sensor were left as received from the manufacturer. Each session involved one participant at a time. Each volunteer was asked to repeat the Cyrillic sign language alphabet letters one by one as it was shown on the video tutorial taken from the official sign language interpreting website[3].

The participant was asked to perform a hand movement starting from the initial position to the one when the letter was clearly distinguished. An example of the manual motion is presented in Figure 3. In single-instance dataset case, the recording of the palm used in a single frame of the letter itself (the 3rd step of Figure 3).

The data were obtained by the method of recording coordinates of the participant's hand and movement of each finger of this hand, precisely the x, y, and z coordinates of each finger joint, orientation and direction of the palm, direction of fingers, frame and hand translation for dynamic gestures. The Leap Motion Visualizer was used for hand and finger tracking purposes. Each letter shown by the participant was recorded in the corresponding .csv file. Each .csv file contained 500 frames of data on average and held data about

[3]http://www.surdo.kz

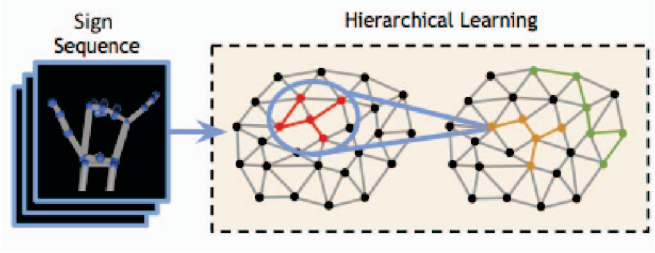


Fig. 4. Hierarchical architecture with self-organizing neural learning (GWR networks). Learning is carried out by training a higher-level network with neuron activation trajectories from a lower level network trained on hand gesture sequences.

the transition from the initial position of the palm to the position when the letter could be clearly seen (Fig. 3). The initial position of the palm was needed to ensure robust tracking of fingers by the Leap Motion sensor.

A Cyrillic alphabet contains 33 letters. Participants without physical disabilities were invited for data collection. At the end of the experiment the entire dataset was processed to contain an equal number of attributes for each frame.

IV. Learning Architecture

The learning architecture consists of 2 hierarchically arranged self-organizing neural networks (Fig. 4). The use of hierarchical self-organization has been shown to be an efficient and effective method for recognizing human motion [4]. This method is consistent with neurophysiological findings that have identified a specialized area for the visual processing of complex motion in the brain in a hierarchical fashion [29]. From a computational perspective, self-organization is an unsupervised learning mechanism that allows to learn representations of the input by iteratively obtaining a non-linear projection of the feature space [30].

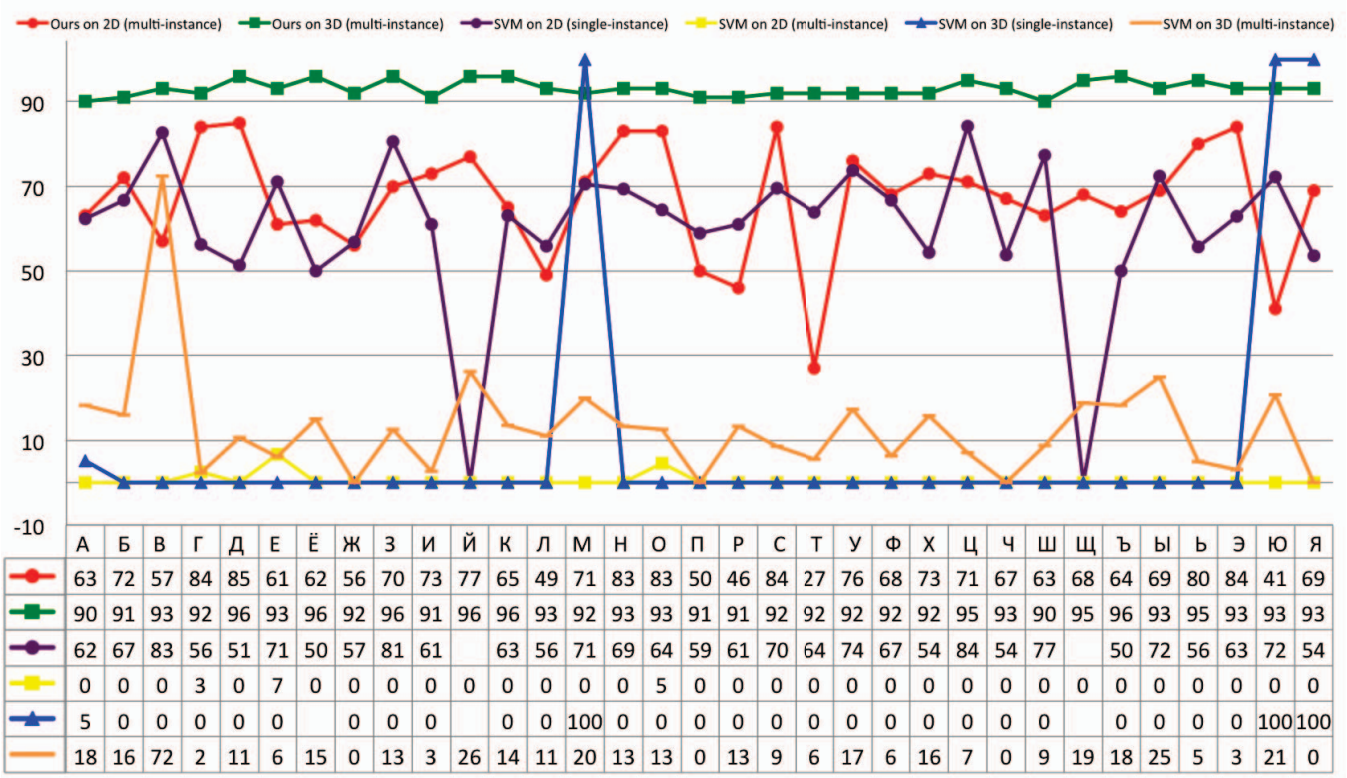
The learning model consists of Growing When Required (GWR) networks [31] that iteratively obtain generalized representations of sensory inputs and learn inherent spatio-temporal dependencies. The GWR network is composed of a set of neurons and their associated weight vectors $\mathbf{w}_j$ linked by a set of edges. The activity of a neuron is computed as a function of the distance (usually the Euclidean distance) between the input and its weight vector. During the training, the network dynamically changes its topological structure to better match the input space following competitive Hebbian learning [32].

V. Experiments & Results

We firstly obtain classification results on RGB and RGB-D data from various machine learning approaches provided by Weka machine learning tool [33] in order to find a baseline needed for performance evaluation of our GWR-based Hierarchical Self-Organizing Learning approach. Specifically, we performed the following experiments:

- First, it should be noted that the dimensions of all datasets were reduced down to 2-10 components using Principal Component Analysis (PCA).
- Secondly, various single-instance supervised learning approaches were applied to our RGB (2D) and RGB-D (3D) datasets. Among different existing approaches provided by Weka machine learning tool [33], Support Vector Machines (SVM) accessible under weka.classifiers.functions.SMO produced the best results with 10-fold cross-validation: 64.68% for single-instance RGB data and 5.99% for single-instance RGB-D data.
- Then, Support Vector Machines for Multiple-Instance Learning (miSVM) [34], which are available through Weka machine learning tool [33], was used as a baseline for evaluation of our classification method. Here the classifier deals with sequences of frames for each letter. The Multi-Instance Learning Kit (MILK)[4] was downloaded as an additional tool for Weka and used on our multi-instance RGB and RGB-D datasets. The tool includes the algorithm, which implements Stuart Andrews' mi_SVM (Maximum pattern Margin Formulation of MIL) and applies weka.classifiers.functions.SMO to solve multiple instances problem. The algorithm attaches a label to each instance in the bag that corresponds to its class label. After, it computes SVM solution using SMO for all instances in positive bags. The algorithm then reassigns the class label for each instance in the positive bag according to the SVM result. It keeps iterating until labels do not change anymore. Overall, classification using multiple-instance SVM 10-fold cross validation showed an accuracy of 4.6% for multi-instance RGB (2D) data and 12.86% for multi-instance RGB-D (3D) dataset.
- Finally, we tested the performance of our previously developed neural-based method on our dynamic data (multi-instance datasets). In the current work, the goal of the hierarchical learning architecture is to process and classify action sequences in terms of the sign language alphabet. For this purpose, we extended the unsupervised GWR-based learning of the higher level network to attach labels to trained neurons. In this case, the network is trained with the motion sequences in an unsupervised fashion while attaching the labels of the input $\lambda(\mathbf{x}_t)$, i.e. letter to best-matching neurons. In contrast to the previous approaches using GWR-based associative learning [4], in this approach each label has a letter value, so that new samples can be processed through the hierarchy and can return the label values of the best-matching sequence. For GWR learning, we used the following training parameters: insertion threshold $a_T = 0.50$, learning rates $\epsilon_b = 0.3$, and $\epsilon_n = 0.006$, $\kappa = 0.5$, maximum $a_{max} = 5$, firing counter parameters $\eta_0 = 1$, $\tau_b = 0.3$, $\tau_n = 0.1$, firing threshold $\eta_T = 0.01$. A thorough discussion of training parameters was presented in [31]. For testing we performed 3-fold cross validation. The classification

[4]http://www.cs.waikato.ac.nz/ ml/milk/



	А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
	63	72	57	84	85	61	62	56	70	73	77	65	49	71	83	83	50	46	84	27	76	68	73	71	67	63	68	64	69	80	84	41	69
	90	91	93	92	96	93	96	92	96	91	96	96	93	92	93	93	91	91	92	92	92	92	92	95	93	90	95	96	93	95	93	93	93
	62	67	83	56	51	71	50	57	81	61		63	56	71	69	64	59	61	70	64	74	67	54	84	54	77		50	72	56	63	72	54
	0	0	0	3	0	7	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	0		0	0	0		0	0	100	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	100	100
	18	16	72	2	11	6	15	0	13	3	26	14	11	20	13	13	0	13	9	6	17	6	16	7	0	9	19	18	25	5	3	21	0

Fig. 5. Recognition results of RGB and RGB-D data of each classifier for each letter. There are 33 letters in total.

accuracy of our algorithm for each letter for RGB and RGB-D data is presented in Figure 5. Our approach achieves an average classification accuracy of 67% for 2D data and 93% for 3D data in classifying Cyrillic manual alphabet. It is an improvement in comparison to our previously reported performance of 77.2% on a depth dataset collected with 10 people detailed in [5].

As can be seen from the cross-comparison table of all obtained results (Table II), single-instance data (both RGB and RGB-D) outperforms multi-instance data for SVM algorithms. Average accuracy for single-instance RGB data is 64.68%. However, our approach performs slightly better on multi-instance RGB data (67%). But the best result is achieved on multi-instance RGB-D data (93%) in comparison to the results obtained from both single-instance and multi-instance data on SVM.

TABLE II
COMPARISON OF RESULTS

Average Accuracy (%)	SVM	OUR METHOD
2D data (single-instance)	64.68%	-
2D data (multi-instance)	4.6%	67%
3D data (single-instance)	5.99%	-
3D data (multi-instance)	12.86%	93%

VI. Conclusion and Future Work

In this paper we utilized our previously developed method and applied it to a new application of human-robot interaction i.e. recognition of Cyrillic fingerspelling consisting of 33 manual gestures. To this end, a depth camera is used to gather a dataset of 3D motion data in real-world settings and a mono camera is used to gather a dataset of 2D data of the professional signers in order to train supervised learning algorithms. Although the utilized learning method was previously used within a different application, obtained results motivate our future work to continue modeling the hand motion based on relevant metrics from each fingerspelling gesture. Our method is shown to outperform the standard machine learning methods. Also, the results presented confirm that the modality of motion depth data provide more accurate results than any other data type. Deriving from this work are four fully annotated single-instance and multi-instance RGB and RGB-D datasets. In addition, our contribution of classifying 33 gestures of the Cyrillic manual alphabet is also in a larger number of classified gestures than previously reported research on fingerspelling recognition. Future work will include collecting a larger dataset for training and overall development of a real-time interpreting autonomy for the robots to be deployed in public spaces.

ACKNOWLEDGMENT

The authors would like to thank the stuff of the community fund “Yunsz Alem” (Silent World) for the provided help with the project. This work was funded by the School of Science and Technology, Nazarbayev University, Kazakhstan.

REFERENCES

[1] M. Tolba and A. Elons, "Recent developments in sign language recognition systems," in *Computer Engineering & Systems (ICCES), 2013 8th International Conference on*. IEEE, 2013, pp. xxxvi–xlii.

[2] O. Aran, "Vision based sign language recognition: modeling and recognizing isolated signs with manual and non-manual components," Ph.D. dissertation, Citeseer, 2008.

[3] A. Sandygulova, Y. Absattar, D. Doszhan, and G. I. Parisi, "Child-centred motion-based age and gender estimation with neural network learning," in *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[4] G. I. Parisi, C. Weber, and S. Wermter, "Self-organizing neural integration of pose-motion features for human action recognition," *Frontiers in Neurorobotics*, vol. 9, no. 3, 2015. [Online]. Available: http://www.frontiersin.org/neurorobotics/10.3389/fnbot.2015.00003/abstract

[5] N. Tazhigaliyeva, Y. Nurgabulov, G. I. Parisi, and A. Sandygulova, "Slirs: Sign language interpreting system for human-robot interaction," in *2016 AAAI Fall Symposium Series*, 2016.

[6] A. K. Sahoo, G. S. Mishra, and K. K. Ravulakollu, "Sign language recognition: State of the art," *ARPN Journal of Engineering and Applied Sciences*, vol. 9, no. 2, pp. 116–134, 2014.

[7] J. Suarez and R. R. Murphy, "Hand gesture recognition with depth images: A review," in *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2012, pp. 411–417.

[8] Z. Mo and U. Neumann, "Real-time hand pose recognition using low-resolution depth images." in *CVPR (2)*, 2006, pp. 1499–1505.

[9] P. Breuer, C. Eckes, and S. Müller, "Hand gesture recognition with a novel ir time-of-flight range camera–a pilot study," in *International Conference on Computer Vision/Computer Graphics Collaboration Techniques and Applications*. Springer, 2007, pp. 247–260.

[10] Z. Li and R. Jarvis, "Real time hand gesture recognition using a range camera," in *Australasian Conference on Robotics and Automation*, 2009, pp. 21–27.

[11] F. Klompmaker, K. Nebe, and A. Fast, "dsensingni: a framework for advanced tangible interaction using a depth camera," in *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction*. ACM, 2012, pp. 217–224.

[12] K. Fujimura and X. Liu, "Sign recognition using depth image streams," in *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*. IEEE, 2006, pp. 381–386.

[13] M. Ronchetti and M. Avancini, "Using kinect to emulate an interactive whiteboard," *MS in Computer Science, University of Trento*, 2011.

[14] C. Bellmore, R. Ptucha, and A. Savakis, "Interactive display using depth and rgb sensors for face and gesture control," in *Image Processing Workshop (WNYIPW), 2011 IEEE Western New York*. IEEE, 2011, pp. 1–4.

[15] S. Malassiotis, N. Aifanti, and M. G. Strintzis, "A gesture recognition system using 3d data," in *3D Data Processing Visualization and Transmission, 2002. Proceedings. First International Symposium on*. IEEE, 2002, pp. 190–193.

[16] R. Feris, M. Turk, R. Raskar, K.-H. Tan, and G. Ohashi, "Recognition of isolated fingerspelling gestures using depth edges," in *Real-Time Vision for Human-Computer Interaction*. Springer, 2005, pp. 43–56.

[17] C. Keskin, F. Kıraç, Y. E. Kara, and L. Akarun, "Real time hand pose estimation using depth sensors," in *Consumer Depth Cameras for Computer Vision*. Springer, 2013, pp. 119–137.

[18] S. Singh, A. Jain, and D. Kumar, "Recognizing and interpreting sign language gesture for human robot interaction," *International Journal of Computer Applications*, vol. 52, no. 11, 2012.

[19] K. Sohn, Y. Kim, and P. Oh, "2-tier control of a humanoid robot and use of sign language learned by monte carlo method," in *2013 IEEE RO-MAN*. IEEE, 2013, pp. 547–552.

[20] F. E. Luis-Pérez, F. Trujillo-Romero, and W. Martínez-Velazco, "Control of a service robot using the mexican sign language," in *Mexican International Conference on Artificial Intelligence*. Springer, 2011, pp. 419–430.

[21] H. Sugiuchi, T. Morino, and M. Terauchi, "Execution and description of dexterous hand task by using multi-finger dual robot hand system-realization of japanese sign language," in *Intelligent Control, 2002. Proceedings of the 2002 IEEE International Symposium on*. IEEE, 2002, pp. 544–548.

[22] P. Uluer, N. Akalın, and H. Köse, "A new robotic platform for sign language tutoring," *International Journal of Social Robotics*, vol. 7, no. 5, pp. 571–585, 2015.

[23] M. Huenerfauth, "A multi-path architecture for machine translation of english text into american sign language animation," in *Proceedings of the Student Research Workshop at HLT-NAACL 2004*. Association for Computational Linguistics, 2004, pp. 25–30.

[24] M. Kipp, A. Heloir, and Q. Nguyen, "Sign language avatars: Animation and comprehensibility," in *International Workshop on Intelligent Virtual Agents*. Springer, 2011, pp. 113–126.

[25] P. Buehler, A. Zisserman, and M. Everingham, "Learning sign language by watching tv (using weakly aligned subtitles)," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 2961–2968.

[26] J. Alon, V. Athitsos, Q. Yuan, and S. Sclaroff, "A unified framework for gesture recognition and spatiotemporal gesture segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 9, pp. 1685–1699, 2009.

[27] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 1. IEEE, 2001, pp. 105–112.

[28] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1. IEEE, 2005, pp. 886–893.

[29] D. Perrett, E. Rolls, and W. Caan, "Visual neurones responsive to faces in the monkey temporal cortex," *Experimental brain research*, vol. 47, no. 3, pp. 329–342, 1982.

[30] T. Kohonen, *Self-organization and associative memory*. Springer Science & Business Media, 2012, vol. 8.

[31] S. Marsland, J. Shapiro, and U. Nehmzow, "A self-organising network that grows when required," *Neural Networks*, vol. 15, no. 8, pp. 1041–1058, 2002.

[32] T. Martinetz, "Competitive hebbian learning rule forms perfectly topology preserving maps," in *ICANN'93*. Springer, 1993, pp. 427–434.

[33] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *ACM SIGKDD explorations newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[34] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Advances in Neural Information Processing Systems 15*. MIT Press, 2003, pp. 561–568.