Fingerspelling Recognition Using Synthetic Images and Deep Transfer Learning

Nguyen Tu Nam[1], Shinji Sako[1], Bogdan Kwolek[2]

[1]Dept. of Computer Science, Nagoya Institute of Technology
466-8555, Gokiso-cho, Showa-ku Nagoya, Japan
t.nguyen.269@nitech.jp, s.sako@nitech.ac.jp

[2]AGH University of Science and Technology
30 Mickiewicza Av., 30-059 Krakow, Poland
bkw@agh.edu.pl

Abstract

Although gesture recognition has been intensely studied for decades, it is still a challenging research topic due to problems posed by background complexity, occlusion, viewpoint, lighting changes, the deformable and articulated nature of hands, etc. Numerous studies have shown that extending the training dataset with real images about synthetic images improves the recognition accuracy. However, little work is devoted to demonstrate what improvements in recognition can be achieved thanks to transferring the style onto synthetically generated images from the real gestures. In this paper, we propose a novel method for Japanese fingerspelling recognition using both real and synthetic images generated on the basis of a 3D hand model. We propose to employ a neural style transfer to include information from real images onto synthetically generated dataset. We demonstrate experimentally that neural style transfer and discriminative layer training applied to training deep neural models allow obtaining considerable gains in the recognition accuracy.

Keywords: Fingerspelling recognition, Transfer learning, Neural style transfer, Synthetic images.

1. INTRODUCTION

Sign language is an indispensable method for communication between Deaf and Hard of Hearing (DHH) people with each other and with normal people. Most countries have own spoken/written languages as well as own sign languages (i.e. Vietnamese Sign Language, American Sign Language, Indian Sign Language). Some of them also have sign language dialects. The number of people who understand the sign language is not high. In order to narrow the gap in communication by sign language, researchers are developing intermediary systems to make communication between such people easier. The systems act as translators, recognize sign languages and translate them to text or speech.

Recognition performance of Japanese sign language (JSL) is still too low. The hand gesture recognition is an important task that also should be considerably enhanced. It is one of the most difficult tasks in human-computer interaction, not only studied in context of sign language recognition, but it is also extensively studied in robotics [1, 2], virtual reality [3, 4], etc. There are two ways to recognize hand gestures. One way is sensor-based, for instance based on data glove [5] and another one is vision-based [6]. Although sensor-based approaches can achieve high performance, they affects user convenience (user has to wear a glove) and the nature of interactions between computers and humans. While the first approach is difficult to deploy widely due to the high price of equipment, the vision-based approach has garnered more attention from researchers in recent years.

Japanese fingerspelling (or yubimoji in Japanese) is a part of JSL and represents the letters of the Japanese alphabet. Similar to American Sign Language (ASL), it is performed with a single hand - also known as one-handed alphabet (some other sign languages use the two-handed alphabet, for example, British Sign Language). Fingerspelling is often used to express words borrowed from other languages, people names, place names, etc., where there is no way to express them using sign language. Two components that make up Japanese fingerspelling are static fingerspelling and dynamic fingerspelling. In this paper, we focus on recognition of static fingerspelling on RGB images.

Developing static Japanese fingerspelling recognition system is still challenging, even with recent advances in computer vision. The major difficulties are as follows:

Lack of data: Available data in Internet repositories and other public resources are very limited.

Complexity: Japanese sign language consist of 41 static signs, 4 dynamic signs and 4 diacritics representing the phonetic syllables. Another difficulty is influence of lighting conditions and backgrounds. Variety of backgrounds, lighting conditions (as well as skin color) greatly affects the performance of hand segmentation and hand detection.

Diversity: The same gesture is usually performed in different times, even by the same performer. It can also be different in shape due to high degree of freedom of articulations. Viewing angle of the camera is also an important factor that influences the recognition performance. Figure 1 presents some examples of three Japanese fingerspelling signs: /ka/, /so/, /ha/. We can see that shape of the hand is very different when the camera view changes.

Figure 1: Influence of viewing angle of the camera on the acquired hand shape

Ambiguity: One can be easily confused when recognizing similar gestures due to flexibility of our fingers and the viewing angle. Let us consider the example of /hi/ and /nu/ gestures (see Figure 2). If the curvature of the index finger is not clear, we may mistake the /nu/ gesture for /hi/ gesture at the particular view angle.

Figure 2: Example of /hi/ and /nu/ signs

Motivated by the shortcomings presented above, in this paper we propose a robust end-to-end method for static Japanese fingerspelling recognition on RGB images. Based on dataset introduced in [7], we have approached the problem in several ways. At the beginning, in order to balance data (reduce the impact of imbalanced data on classification results), we expanded the real dataset using synthetic images, which were rendered on the basis of a 3D hand model with views from multiple cameras. Numerous studies have shown that synthetic images enhance the recognition performance. For instance, Ros *et al.* [8] used synthetic images to improve semantic segmentation of urban scenes. Lugaresi *et al.* [9] utilized synthetic hand images to boost performance in keypoint regression task. However, there are not many studies that show what recognition accuracy can be achieved in case of the use of synthetic images only. Another issue that has not been addressed until now is what accuracies can be achieved using only synthetic images in hand gesture recognition task. One of the problems in approaches relying on synthetically generated images on the basis of 3D hand models is that texture diversity is insufficient. Such lack of rich texture diversity adversely affects the recognition performance on the test subset. While increasing the diversity of the texture needs a lot of manual work like adjusting lighting, changing the texture of the hands, the neural style transfer can help us create new synthetic images of fingerspelling automatically with color, brightness, and contrast as on real images. Afterwards, we combined real images with synthetic images, in which we included information from real images using neural style transfer. Finally, using transfer learning approach we performed experiments on fingerspelling recognition. Figure 3 shows the pipeline of the proposed system.

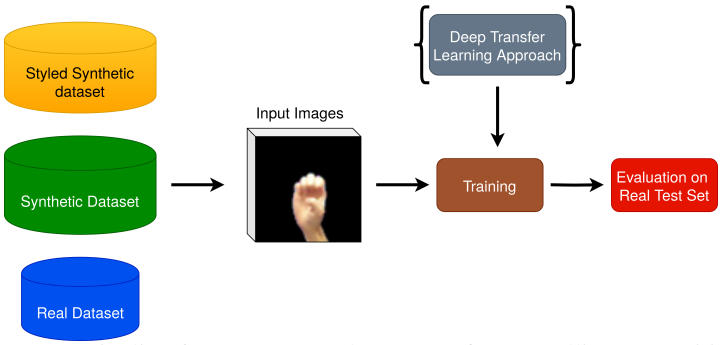


Figure 3: Pipeline for our proposed Japanese finger spelling recognition

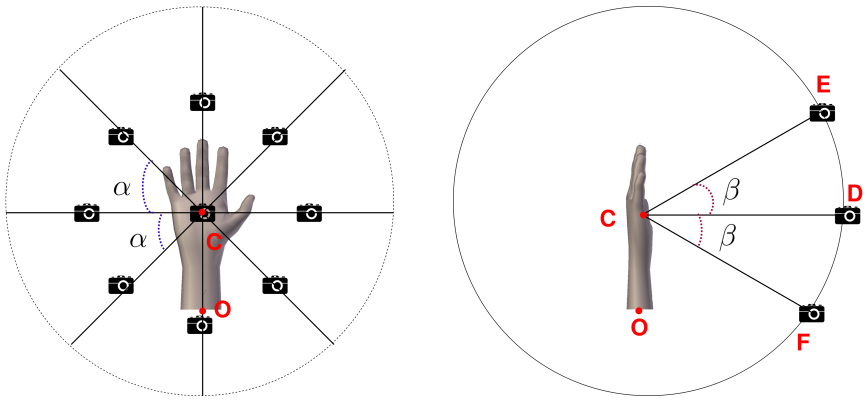
2. RELATED WORKS

Hand gestures/fingerspellings recognition is a task that is still extensively studied due not enough performance that is required by real applications. Recently, a large scale dataset for gesture recognition in the wild has been proposed [10]. Shi *et al.* proposed an approach for continuous ASL fingerspelling recognition task, reducing the need for detection or segmentation steps. However, to the best of our knowledge there is no sufficiently large dataset for Japanese fingerspelling recognition that permits the training deep neural networks with generalization sufficient for real applications. The most closely related work is a work done by Mukai *et al.* [11], who used a classification tree and Support Vector Machine (SVM) to recognize 41 static Japanese finger spelling signs. The evaluations were done on 287 images that have simple background, whereas classification tree was utilized to recognize easy signs, and SVM was used for more difficult nine signs. The classification accuracy was equal to 86%. Machacon *et al.* [12] employed data glove for data acquisition and multilayer perceptron model for recognition of static Japanese fingerspelling signs. However, they successfully recognized only 18 signs of 41 signs. Usually, in order to cope with not enough amount of training data, transfer learning is applied for training deep models for fingerspelling recognition. For example, in ASL fingerspeling recognition such an approach has been employed in [13].

3. METHODOLOGY

3.1 3D Hand model and rendering

A 3D hand model has been used to render synthetic finger spellings images and to extend real hand dataset that has been introduced in [7]. Based on @3DHaupt's rigged hand model, we used the Blender software to modify the 3D model and synthesize 41 fingerspelling signs. In order to acquire 3D hand images from different angles, we added nine cameras into the model. Figure 4 illustrates the location of cameras in front and left side of viewing angles. At the beginning, we



(a) Front view (b) Left side view

Figure 4: Illustrations the location of hand model and nine cameras

created a circle parallel to the xOy plane, where the center was a point on the palm (called C). Then, we created two circles with the same center C and the same radius as the first circle. Those two circles intersect the first circle with angles equal to 45 and 90 degrees, respectively. Afterwards, we selected the intersection of the three circles at the position D in front of the hand model and placed a camera. Finally, on each circle, we placed two cameras at two points E and F, where where $\beta = \widehat{ECD} = \widehat{FCD} = 30°$. All cameras point toward the point C.

A python script has been developed to automatically render the synthetic images. By using skeletal animation technique, for each of 41 fingerspelling signs, we created three actions. Each action is created on the basis of start and end postures. The number of frames for each action (corresponding to given hand posture) can be changed in the python script. For example, if we create an action with eight frames/postures, we only need to create the starting and ending postures, whereas the remaining 6 frames will be automatically generated by the Blender. In this way, we can minimize the manual work and still create a large amount of required hand postures. During preparing the dataset we changed the number of frames per action $n_{frame} = 2, 4, 8, 16$ and 32. The number of rendered synthetic images N can be determined on the basis of the following formula:

$$N = n_{sign} * n_{camera} * n_{action} * n_{frame} \tag{1}$$

where n_{sign} is number of static fingerspelling sign, n_{camera} stands for number of cameras, whereas n_{action} is number of actions per sign. All rendered images of size 400x400 have black background and have been stored in the *.png* format. Sample rendered images (/a/ gesture) in view of the employed cameras are shown in Figure 5.

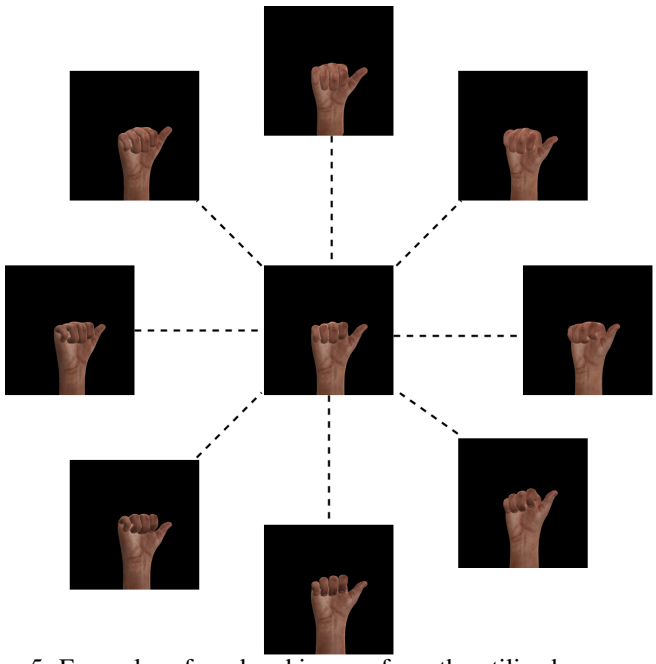


Figure 5: Examples of rendered images from the utilized camera setup

3.2 Deep Transfer Learning

Over the last few years, deep learning has emerged as the state-of-the-art in artificial intelligence [14]. The potential of deep learning and Convolutional Neural Networks (CNNs) has been successfully utilized in many challenging fields. Some promising results were obtained in sign language/gesture recognition [15, 16]. However, deep learning approaches require considerable amount of data to properly train the models. The lack of data of sufficient amount is a common problem in deep learning. Several promising techniques to address the lack of data have been proposed, and one of the most noticeable methods is deep transfer learning.

The idea behind transfer learning is based on that humans use knowledge learned in the past and then apply/transfer it to a new field or in new environment. Unlike traditional machine learning that learns each task independently, and does not consider knowledge from other domains to improve its generalization, transfer learning attempts to reuse knowledge $\mathcal{K}$

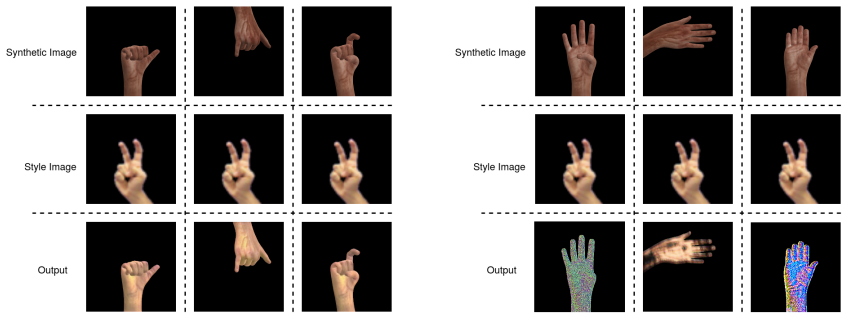
learned from existing source domain $\mathcal{S}_\mathcal{D}$, when encounter a new issue/topic $\mathcal{T}_\mathcal{D}$ (target domain) [17]. In most cases (and in our case) the size of $\mathcal{S}_\mathcal{D} \gg$ size of $\mathcal{T}_\mathcal{D}$. Usually, training deep network with millions of parameters on a large scale dataset (i.e ImageNet [18] - 1000 classes, 1.2M images for training, 50K images for validation, 100K images for testing) often tooks few days/weeks depending on the employed GPU/TPU. After the learning is completed, we gain the knowledge $\mathcal{K}$ using state-of-the-art deep learning models with the learned weights (also known as pre-trained models). We will use such knowledge to apply it to our problem, in which the amount of labeled data is limited (size of $\mathcal{S}_\mathcal{D} \gg$ size of $\mathcal{T}_\mathcal{D}$).

CNNs can learn hierarchical representation of features, whereas different layers extract different information from the input. First layers learn low-level features, such as edges and corners that are not specific to the task, while deeper layers extract high-level features such as objects, complex textures [19] that learn specific features depending on the tasks. In order to apply deep transfer learning to our problem, we will keep first layers and weights of a pre-trained model, replace last few layers and add own layers (initialized with random weights). Such approach allows us dealing with small dataset and to minimize the risk of overfitting.

3.3 Neural Style Transfer

Synthetic images obtained from 3D hand model have an unique texture. This reduces the diversity of hand data. To generate fingerspelling images similar to real fingerspelling images and increase the diversity of data, we propose to use Neural Style Transfer (NST). NST was proposed by Gatys *et al.* in 2016 [20] and it allows redrawing a photograph in the style of any arbitrary painting without artist's knowledge. The idea behind NST is based on capability of deep models to learn hierarchical representation of features (as we mentioned in Section 3.2), owing to which we can extract representations of the content and the style of images. The main idea of NST is as follows. We have three images: style image, content image and input image (usually would be a random noise image or a copy of the content image). We take a pre-trained network to extract style S_s from style image, content C_c from content image and both style S_i, content C_i from input image. The input image will be transformed to minimize both content loss $\mathcal{L}_c = \|C_c - C_i\|^2$ and style loss $\mathcal{L}_s = \|S_s - S_i\|^2$.

Our NST uses a pretrained VGG 19-layer model with batch normalization *vvgg19_bn* [21]. All images are resized to 512x512. Nine images randomly selected from the training part of real hand dataset act as style images. Each image in the synthetic dataset will randomly take one of nine style images to perform the training and to generate a new image. To remove unexpected pixels in generated images, we used opencv to create masks from each synthetic image (made from 3D hand), and apply *bitwise AND* operation on masks-generated images. While the majority of images generated by NST have "good" hand texture, some of generated images produced "not good" texture. Figure 6 shows some examples of "good" outputs and "not good" outputs, where the style image contained the /ro/ sign.



(a) Good outputs (b) Not good outputs

Figure 6: Some examples of NST with style images containing /ro/ sign

4. DATASET

Real Dataset: Japanese fingerspelling dataset [7] contains real hand gestures performed by ten people, and has been recorded by an RGB camera. The dataset is divided into 2 parts: training set (nine people) and person independent test set (one person). All images in the dataset are color images (black background) of size 64x64, belonging to 41 classes (corresponding to 41 signs). The discussed dataset is unbalanced.

Synthetic dataset: As indicated above, in the synthetic dataset the number of frames per action is equal $n_{frame} = 2, 4, 8, 16, 32$. In the following experiments, we use the synthetic dataset with $n_{frame} = 8$. All images in the dataset are color images (black background) of size 400x400 and belong to 41 classes (corresponding to 41 signs). Each class has the same number of images, equal to 216. The size of synthetic dataset is $N = 8856$ (results from (1)).

Styled synthetic dataset: After applying NST to the whole image from the synthetic dataset with $n_{frame} = 8$ (made from 3D hand model), we got a styled synthetic dataset which is the same size as synthetic dataset. All images in the dataset are color images (with black background) of size 512x512.

5. EXPERIMENTAL RESULTS

We performed five experiments. All experiments were evaluated on test subset of the real dataset. In the first one, the training data was the real sub-dataset (Real). The training data in the 2nd experiment was a combination of training subset of the real dataset and the synthetic dataset (Real + Synth). A combination of training subset of the real dataset and the styled synthetic dataset (Real + Style Synth) has been evaluated in 3rd experiment. In 4th (Synth) and 5th (Styled Synth) experiment we simulated a scenario in which we have in disposal only the synthetic data and/or styled data for the training.

Three pre-trained models: densenet161, vgg19_bn, resnet18 have been used to perform training of the models for the classification of fingerspellings. We kept first layers of selected pre-trained models, which have weights learned on the large scale ImageNet dataset. Then we added some layers such as AdaptiveConcatPool2d, flatten layer, dropout and linear one (out_features = 41 corresponding to 41 classes) at the output with weights initialized randomly. Input images were resized to resolution of 64x64x3 pixels. In the training phase, instead of freezing some layers and training remaining layers, we used a technique called *discriminative layer training*. In the discussed technique models can be trained with different learning rates on each layer group. Since first layers have learned general features, during fine-tuning the model, the learning rates for such layers should be small, while last layers require larger learning rates. Figure 7 illustrates the idea of discriminative layer training, where learning rates $a \leq b \leq c$. During the training, an online data augmentation

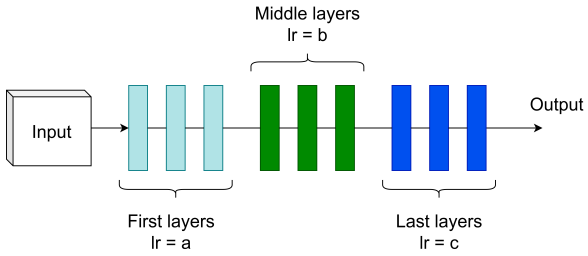


Figure 7: Discriminative layer training

has been used. Batch size was set to 128 and number of epochs was set to 10. After finishing the training, Test Time Augmentation (TTA) on the test subset has been executed. We applied different transformations to each image from the test subset. Then model made a prediction for each augmented image and returned a final prediction, which is an ensemble of those predictions. TTA may not be the solution for real-time applications due to additional transformations on image to make predictions. However, with additional cost the model can perform better predictions.

As we can observe in Tab. 1, a promising accuracy can be achieved on real images in the person independent scenario. Extending the real dataset about synthetic images of fingerspellings, which were obtained by posing the 3D models in the requested hand articulations leads to significant improvement of the classification accuracy. We demonstrated experimentally that combining real images with synthetic images, whose style was transferred from the real images leads to further improvement of the classification accuracy, see results in 3rd row in Tab. 1. Comparing results in the last two rows, we can observe that the synthetic images with style transferred from the real images permit to achieve far better classification accuracy in comparison to experiment in which only synthetic images were employed. Thanks to very faithful reproduction of gestures by our 3D models, and in particular thanks to the expansion of the real dataset with model (expert) realization of gestures, the recognition rates achieved on the basis of synthetic images are high.

Table 1: Classification performance in five scenarios.

	densenet161		vgg19_bn		resnet18	
	Accuracy	TTA	Accuracy	TTA	Accuracy	TTA
Real	0.9050	0.9240	0.9257	0.9344	0.8826	0.8843
Real + Synth	0.9413	0.9482	0.9465	0.9430	0.9119	0.9102
Real + Styled Synth	**0.9585**	**0.9551**	**0.9603**	**0.9620**	**0.9275**	**0.9292**
Synth	0.7288	0.6839	0.6442	0.6183	0.5320	0.4922
Styled Synth	0.7945	0.8117	0.7599	0.7807	0.7288	0.7375

6. CONCLUSIONS

In this paper, we proposed a novel approach for static finger spelling recognition on RGB images. To deal with data imbalance in real fingerspelling dataset [7], a 3D hand model and a python script to automate the rendering have been created, and few thousands synthetic images rendered on the basis of 3D hand model were added to the dataset. Deep transfer learning method has been used to support training the deep model. In addition, neural style transfer method has been utilized to transfer style of the real images onto the synthetic images. We demonstrated experimentally that including information in the synthetic images from the real images by the use of the neural style transfer techniques leads to better classification rates. Thanks to such additional information the recognition performance is better than 95% in the person independent scenario. In scenarios in which we have only synthetic images for the training and a limited amount of real hand images (9 style images for each gesture), promising classification rates can be achieved. Future work includes experiments on the synthetic dataset with different number of frames.

ACKNOWLEDGMENTS

This work was partially supported by Polish National Science Center(NCN) under research grant 2017/27/B/ST6/01743.

REFERENCES

[1] J. L. Raheja, R. Shyam, U. Kumar, and P. B. Prasad, "Real-time robotic hand control using hand gestures," in *2010 Second International Conference on Machine Learning and Computing*, pp. 12–16, 2010.

[2] M. Van den Bergh, D. Carton, R. De Nijs, N. Mitsou, C. Landsiedel, K. Kuehnlenz, D. Wollherr, L. Van Gool, and M. Buss, "Real-time 3d hand gesture interaction with a robot for understanding directions from humans," in *2011 RO-MAN*, pp. 357–362, 2011.

[3] S. S. Rautaray and A. Agrawal, "Interaction with virtual game through hand gesture recognition," in *2011 International Conference on Multimedia, Signal Processing and Communication Technologies*, pp. 244–247, 2011.

[4] C. Khundam, "First person movement control with palm normal and hand gesture interaction in virtual reality," in *2015 12th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 2015.

[5] M. A. Ahmed, B. B. Zaidan, A. A. Zaidan, M. M. Salih, and M. M. B. Lakulu, "A review on systems-based sensory gloves for sign language recognition state of the art between 2007 and 2017," *Sensors (Basel, Switzerland)* (7), 2018.

[6] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: a survey," *Artificial Intelligence Review* **43**, pp. 1–54, Jan 2015.

[7] N. T. Nguen, S. Sako, and B. Kwolek, "Deep cnn-based recognition of JSL finger spelling," in *Hybrid Artificial Intelligent Systems*, pp. 602–613, Springer International Publishing, (Cham), 2019.

[8] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *CVPR*, June 2016.

[9] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C. Chang, M. G. Yong, J. Lee, W. Chang, W. Hua, M. Georg, and M. Grundmann, "Mediapipe: A framework for building perception pipelines," *CoRR* **abs/1906.08172**, 2019.

[10] B. Shi, A. M. D. Rio, J. Keane, D. Brentari, G. Shakhnarovich, and K. Livescu, "Fingerspelling recognition in the wild with iterative visual attention," 2019.

[11] N. Mukai, N. Harada, and Y. Chang, "Japanese fingerspelling recognition based on classification tree and machine learning," in *2017 Nicograph International (NicoInt)*, pp. 19–24, June 2017.

[12] H. T. C. Machacon and S. Shiga, "Recognition of japanese finger spelling gestures using neural networks," *Journal of Medical Engineering & Technology* **34**(4), pp. 254–260, 2010.

[13] B. Kang, S. Tripathi, and T. Q. Nguyen, "Real-time sign language fingerspelling recognition using convolutional neural networks from depth map," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, 2015.

[14] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: A brief review," *Computational Intelligence and Neuroscience* , p. 7068349, 2018.

[15] P. Molchanov, S. Gupta, K. Kim, and J. Kautz, "Hand gesture recognition with 3d convolutional neural networks," in *CVPR Workshops*, June 2015.

[16] J. C. Núñez, R. Cabido, J. J. Pantrigo, A. S. Montemayor, and J. F. Vélez, "Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition," *Pattern Recognition* , 2018.

[17] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering* **22**, pp. 1345–1359, Oct 2010.

[18] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[19] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Advances in Neural Information Processing Systems 27*, Curran Associates, Inc., 2014.

[20] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *CVPR*, 2016.

[21] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv e-prints* , p. arXiv:1409.1556, Sep 2014.