Sign Language Recognition Based on Deep Learning

Aurora

Abstract: For sign language recognition, this paper introduces a model of CNN, and an application has been implemented using this model. Meanwhile, this paper compared the SVM model and CNN model. The accuracy of CNN model reaches 99.77%. In four kernels of SVM model, the accuracy of linear kernel is also quite high (99.48%). The result is that the CNN model performs better on the dataset of sign language MNIST.

Keywords: Sign Language; Support Vector Machine; Convolutional Neural Network; Machine Learning; Deep learning

1 INTRODUCTION

Sign language [1] is a kind of language which is primary used for the deaf and hard of hearing. Also sign language is used for those who are unable to physical speak. Everyone who want to communicate with the deaf has to use sign language. There are so many kinds of sign language in the world. Different places have different sign language for the sign language always comes from natural language. Thus, sign language is an important way to communicate not only between the deaf but also between the deaf and the non-deaf.

With the development of technology, computer vision has becoming an effective way to provide help in people's daily life. So many applications have been developed in almost every field of industry, education, agriculture and business. As the hardware has been improving increasingly, there are lots of intelligent algorithm coming out, like Support Vector Machine, Deep Neural Network, which have been used in the applications of smart home, e-learning, robot and so on.

In this paper, we applied computer vision technology into sign language recognition by using different model (Support Vector Machine and Deep Neural Network) to classify the alphabets of sign language. We only focus on American Sign Language which is primary language for many deaf in North America. The dataset of this paper is sign language MNIST which includes 34627 images. These images are about 24 alphabets of sign language. We divided it into two datasets: one is training dataset and the other is testing dataset. Then we trained two models (Support Vector Machine and Deep Neural Network) online on the Kaggle website and compared the result. In order to get the best hyperparameter, we did the validation by using 90% of total dataset to train and 5% of total dataset to test. In the Support Vector Machine model, we compared four kernel functions (linear kernel, Polynomial kernel, Gaussian radial basis kernel and Sigmoid kernel), and the accuracy of linear kernel is 99.48%. That means the model is linear. In the Deep Neural Network model, we used 3-layer convolutional neural network and got accuracy of 99.77% which is quite high. The reason may be that the images in this dataset are taken in the same condition, like same place, same brightness. Meanwhile, we designed and implemented an application of sign language recognition which could capture the image from the camera of computer and classify the image.

2 DATA SET

Sign language MNIST [2] is a dataset used by researcher who want to build the model for sign language alphabet classification. It comes from the MNIST database (Modified National Institute of Standards and Technology database) which is a large database of handwritten digits used for image-based machine learning methods. Same with MNIST, Sign language MNIST has 28×28 pixel (784 total) per sample. Instead of black and white image in MNIST, the image in sign language MNIST are all colorful. It does not include all 26 alphabets, for the letter J and Z requires motion. The dataset includes 34627 images and each image has a label (0-8, 9-24). There is no 9 and no 25, for we exclude the letter J and letter Z. The all images in dataset are saved in csv file. Each row represents a single image. The first column is label (the number of class that this image is). From the second column to 785th column, there are 784 pixels. Each pixel has grayscale values between 0-255 for the image are converted to gray. These images show multiple persons repeating the gesture against different backgrounds. In order to enlarge the quantity, the images were modified by 3 degrees rotation and +/-15% brightness. Thus, this dataset is very suitable for intelligent algorithm building.

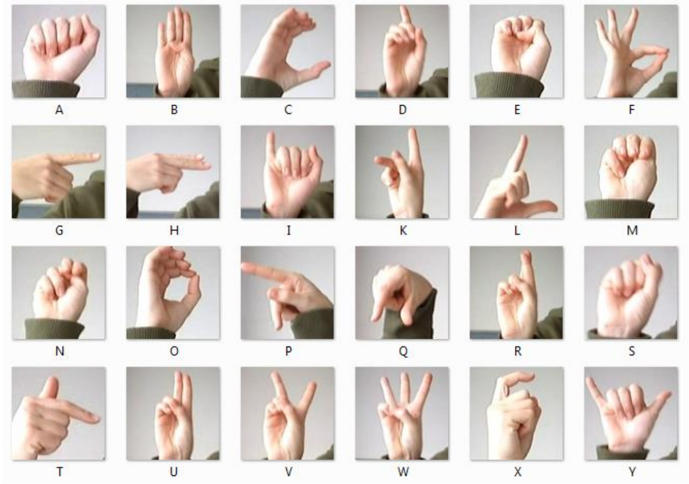


Figure 1: Samples of database

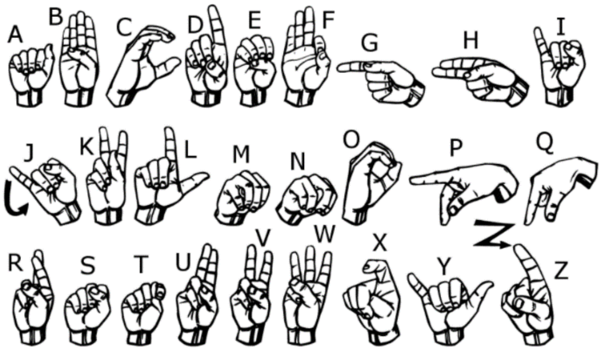


Figure 2: The alphabets of sign language

3 SUPPORT VECTOR MACHINE AND CONVOLUTIONAL NEURAL NETWORK

In recent years, there are so many machine learning algorithms coming out. The most popular two is Support Vector Machine (SVM) and Deep Neural Network (DNN). And Convolutional Neural Network (CNN) is a kind of DNN.

3.1 Support Vector Machine

In order to solve the non-linear problem, Support Vector Machine is a way to map the data into high dimension to classified by hyperplane. A kernel function is used to mapping. There are 4 main kernel function[3], linear kernel, polynomial kernel, gaussian radial basis kernel, sigmoid kernel. The formulas of these are as follow:

Linear kernel:

$$k(x_i, x_j) = (x_i \cdot x_j) \tag{1}$$

Polynomial kernel:

$$k(x_i, x_j) = (x_i \cdot x_j + 1)^d \tag{2}$$

Gaussian radial basis kernel:

$$k(x_i, x_j) = e^{(-\gamma \|x_i - x_j\|^2)} \tag{3}$$

Sigmoid kernel:

$$k(x_i, x_j) = tanh(\alpha x^T y + c) \tag{4}$$

In this paper, we tried these four kernels and got four accuracy respectively.

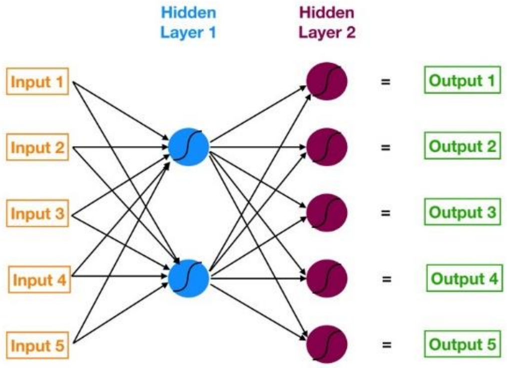


Figure 3: The structure of Neural Network

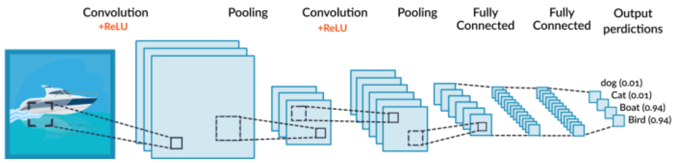


Figure 4: The structure of Convolutional Neural Network

3.2 Convolutional Neural Network

Deep Neural Network (DNN) is a highly effective way for image classification, especially convolutional neural network which is also called CNN. Figure 3 is the structure of Neural Network which includes two hidden layers. Figure 4 [4] is the structure of Convolutional Neural Network which has seven layers. In each convolutional layer, 3×3 matrix is used to get convolved feature. The aim of this is to get high level feature such as edges. About the pooling layer, it returns the maximum value or average value from the portion of the image covered by the matrix. The whole CNN structure includes several pairs of convolutional layers and pooling layers. More layers better are. But more layers more complex the structure is. So, it is necessary to trade off.

4 THE APPLICATION OF SIGN LANGUAGE RECOGNITION

We designed and implemented the application of sign language recognition. There are two parts, training model part, and real time recognition part. The flow chart of training model is as figure 5.

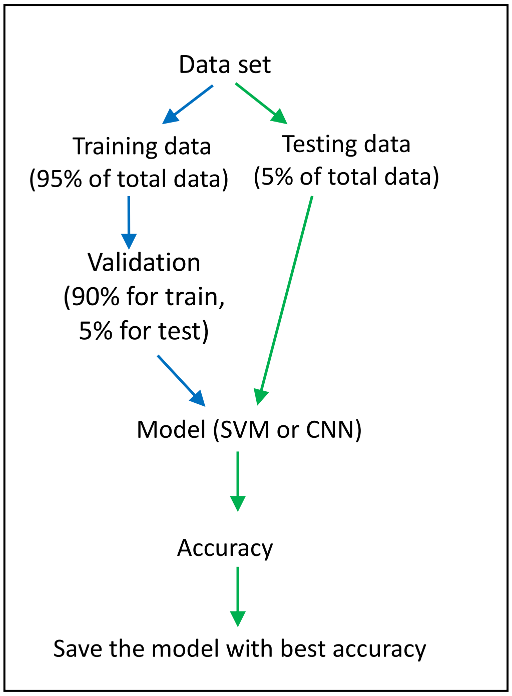


Figure 5: The flow chart of training model

In the training model part, the dataset is split into two sets, one is training dataset, the other is testing dataset. Training data is used to train model and testing data is used to do predict to get the accuracy so that the best model is the one has the highest accuracy. In model training, validation has been done by 90% of total data (31164 images) for train and 5% of total data (1731 images) for test. It is used to optimize the parameter of model. Because of the limit of computer hardware, we did this part on Kaggle website by Sklearn and Keras, and training the model online using GPU. Obviously, using GPU is much faster (almost 10 times) than training in local. The dataset has been saved in csv file. That means the image has been converted into gray and resized to 28×28 already. The figure 6 shows a sample which is resized to 28×28 pixel. The figure 7 shows that the count of training data for each class is almost same. That means it is balance between each class

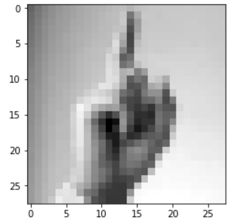


Figure 6: A sample after processing

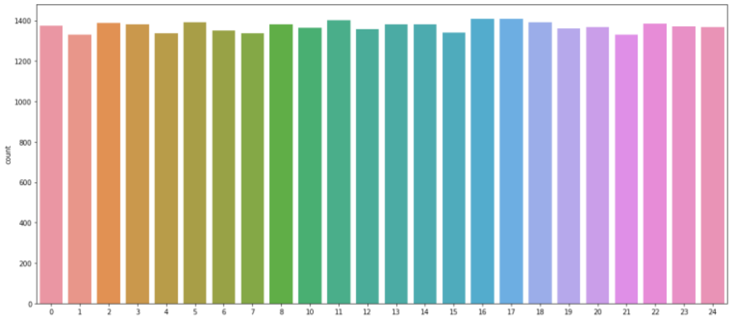


Figure 7: The count of training data for each class

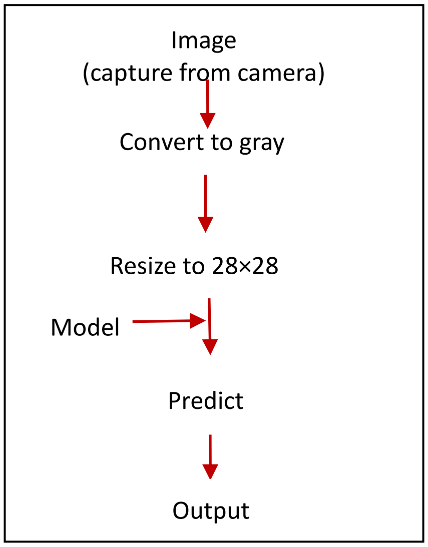


Figure 8: The real time recognition application

The other part of application is real time recognition by using OpenCV [5]. Flow chart shows as figure 8. First, the camera of computer is opened after starting up the application and begin to capture the image. Then the image is converted to gray and resized to 28×28. At last, the image data is predicted by the model of the HDF5 file. The output of the predict is the number of the class that this image belongs to, and the application shows the alphabet according to this number. The figure 9 shows the UI of this application.

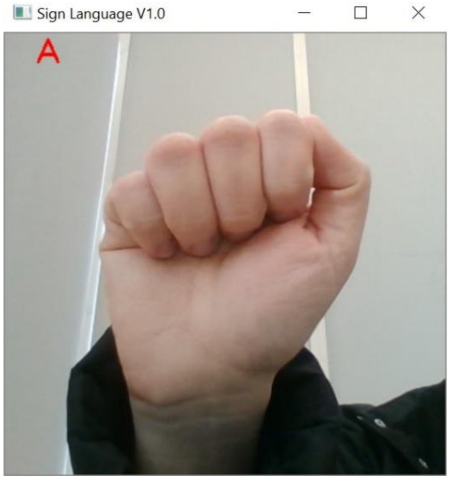


Figure 9: The UI of the application

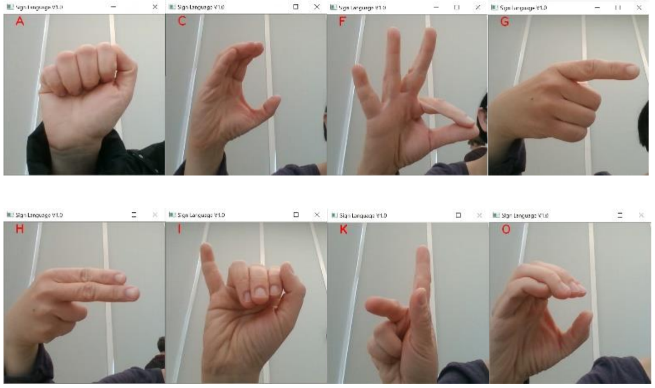


Figure 10: Samples classified successfully

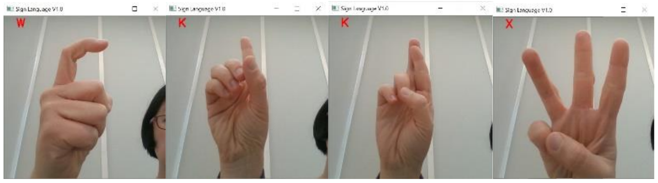


Figure 11: Samples classified unsuccessfully

The figure 10 shows some samples which the application classified successfully. Also, the figure 11 shows some samples classified unsuccessfully. This may be because the gestures of the hand. The tester could not use sign language so well, and the position of hand does not correct. Meanwhile, the brightness and scale of image are different with the dataset. So, sometime, it is a little harder to recognition.

5 SIGN LANGUAGE RECOGNITION BASED ON DIFFERENT MODEL

This part shows the model training in this application. As the SVM and CNN are both effective models, we build both models using sign language MNIST. Comparing different kernel function of SVM, the accuracy of SVM model are different. Meanwhile, CNN model shows excellent performance.

5.1 Sign Language Recognition Based on SVM

As it is mentioned in part 3, there are four kinds of kernel functions of SVM [6]. We build the SVM model with different kernels. The results show as table 1. Most of these four kernels perform well, especially the linear kernel reaches the accuracy of 99.48%. That means this dataset should be classified by linear. But about the sigmoid kernel, the accuracy is only 47.58%. As the structure of SVM with sigmoid kernel is similar with neural network, the neural network may not perform well on this dataset. The accuracy of other two kernels (polynomial kernel and RBF kernel) are reasonable.

Table 1: Accuracy with different kernel

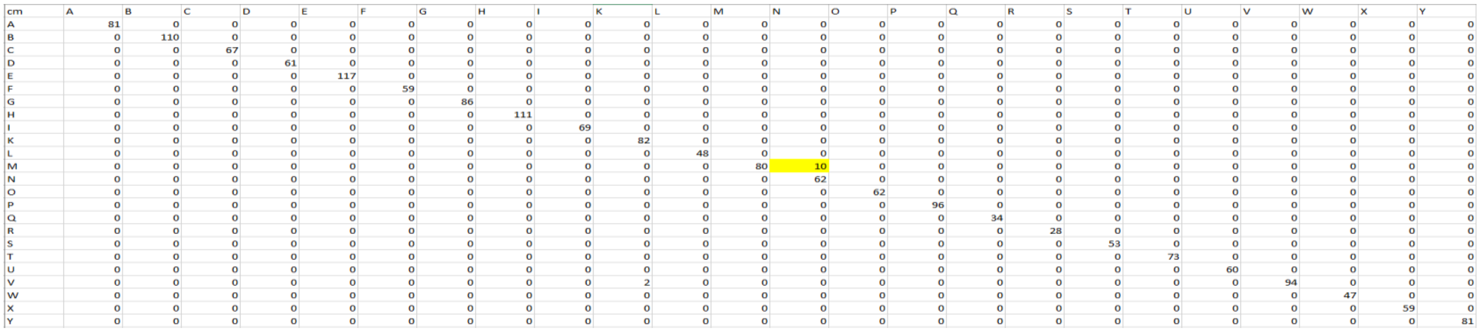
	Accuracy
Linear kernel	99.48%
Polynomial kernel	80.25%
RBF kernel	94.11%
Sigmoid kernel	47.58%

The confusion matrix of SVM model with linear kernel is as table 2. It shows good result. Only 10 images of ‘M’ are recognized as ‘N’. Other 1730 images are all recognized correctly.

5.2 Sign Language Recognition Based on Deep Learning

CNN is a kind of Deep Neural Network, which is good at in image analysis. We used this to build another model [7]. This CNN model includes 10 layers which perform pretty good. The accuracy reaches as high as 99.77%

Table 2: confusion matrix of SVM model with linear kernel



cm	A	B	C	D	E	F	G	H	I	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
A	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B	0	110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	0	67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	0	0	0	61	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	117	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	86	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	0	111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I	0	0	0	0	0	0	0	0	69	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K	0	0	0	0	0	0	0	0	0	82	0	0	0	0	0	0	0	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	0	0	0	48	0	0	0	0	0	0	0	0	0	0	0	0	0
M	0	0	0	0	0	0	0	0	0	0	0	80	10	0	0	0	0	0	0	0	0	0	0	0
N	0	0	0	0	0	0	0	0	0	0	0	0	62	0	0	0	0	0	0	0	0	0	0	0
O	0	0	0	0	0	0	0	0	0	0	0	0	0	62	0	0	0	0	0	0	0	0	0	0
P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	96	0	0	0	0	0	0	0	0	0
Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	34	0	0	0	0	0	0	0	0
R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	0	0	0	0	0	0
S	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	53	0	0	0	0	0	0
T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	73	0	0	0	0	0
U	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0
V	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	94	0	0	0
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	47	0	0
X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	81

The figure 12 shows loss value of both training and validation. They converge quickly as epoch increasing, and they are almost equal at last. That means this model are not over-fitting.

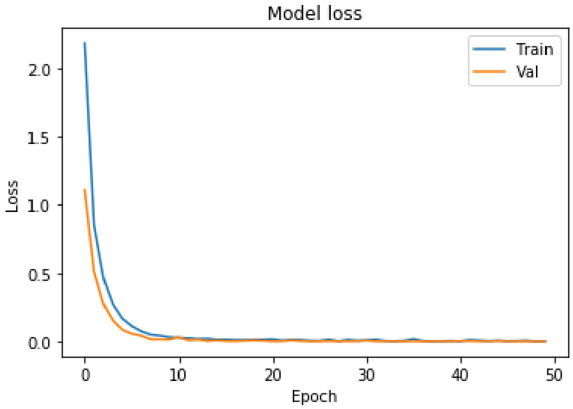


Figure 12: Loss value of CNN model with epoch increasing

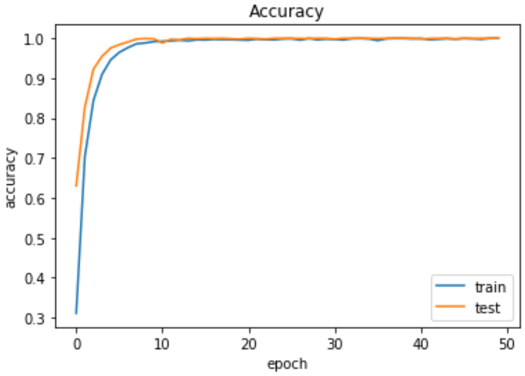


Figure 13: Accuracy of the CNN model with epoch increasing

Also, as the figure 13 shows, the training accuracy and testing accuracy are both converging quickly, and almost equal at last.

5.3 Result Analysis

There are some different between SVM model and CNN model. By training CNN model for 10 times, we got the accuracy from 96.59% to 99.77%, which means the accuracy swing around 99%, as the figure 14 shows. Otherwise, the accuracy of SVM model is always the same no matter how many times the model is trained. This is because the CNN depends on initial value. Each time, different initial values lead to different network. Thus, the accuracy is different.

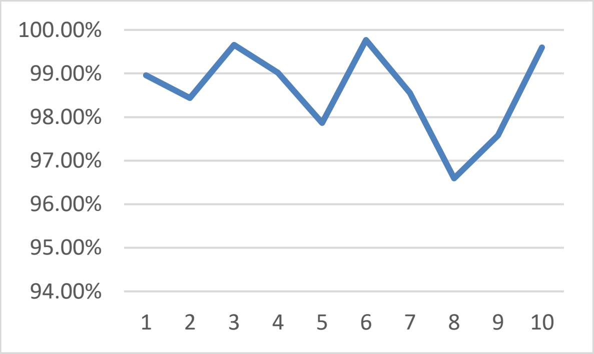


Figure 14: The accuracy of training CNN 10 times

6 CONCLUSIONS

Helping people to communicate with the deaf and assisting people who want to learn sign language, we design and implement an application of real time sign language recognition. We build and compare two models using SVM and CNN. The result shows the CNN model is better than SVM model, although in SVM model linear kernel also perform well.

REFERENCE

[1]https://en.wikipedia.org/wiki/Sign_language

[2]https://www.kaggle.com/datamunge/sign-language-mnist

[3]https://data-flair.training/blogs/svm-kernel-functions/

[4]https://missinglink.ai/guides/convolutional-neural-networks/convolutional-neural-network-tutorial-basic-advanced/

[5]https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_tutorials.html

[6]https://www.kaggle.com/auroraw/kernel-signlanguagesvm

[7]https://www.kaggle.com/auroraw/kernel-signlanguagedl