Recognizing Static Sign Language Gestures using CNN

Tim Hadler
Dept. Computer Science and Software Engineering
University of Canterbury
Christchurch, New Zealand
tch118@uclive.ac.nz

Professor Richard Green
Dept. Computer Science and Software Engineering
University of Canterbury
Christchurch, New Zealand
richard.green@canterbury.ac.nz

***Abstract*—This paper proposes a method to recognize static sign language gestures from an image, using a convolutional neural network (CNN). The method involves pre-processing the image before passing it to the classifying CNN. The pre-processing includes applying Gaussian blur, color segmentation, contour extraction, and morphological operations. The method testing resulted in an overall accuracy of 93.3%.**

Keywords—Cross Entropy Loss, epochs, ANN, CNN, morphology

I. Introduction

Recognizing objects, gestures and body language are simple tasks for humans. Information gathered from the eyes, is passed through a complex network of neurons to decode and relay information on what is being looked at. For computers, recognizing objects and gestures through a camera is a much more complex task. There are several methods in the literature that aim to complete this task, such as artificial neural networks [14], convolutional neural networks (CNN), support vector machines, nearest neighbours [15], graphs, and distributed locally linear embeddings. Using neural networks to train a computer program to recognize objects is now very common and growing in popularity. Neural networks are mathematical models based on the human brain. In this paper, a method is proposed to recognize static hand gestures of the American Sign Language, using simple image processing techniques, and a pre-trained CNN.

II. Background

A. Artificial Neural Networks

An artificial neural network (ANN) is a computing structure based on the topology of the human brain, commonly used to automatically classify objects and features. Artificial neural networks consist of nodes, or 'neurons', arranged in layers that transmit information through the network. Figure 1 shows the basic structure of an ANN. Data to be classified is passed into the network through the input layer, is processed through the inner 'hidden' layers, and the result emerges in the output layer. A network with multiple hidden layers is known as a deep artificial network, or a multilayer perceptron (MLP). A multilayer perceptron can solve nonlinearly separable problems [5].

Convolutional neural networks (CNNs) are the most widely used deep learning structure for image classification. CNNs generally consist of convolution layers, pooling layers,

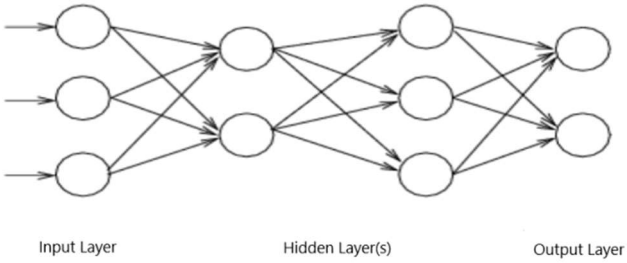

Figure 1: General ANN architecture, adapted from [7]

and fully connected layers. Figure 2 shows a model for a CNN.

Convolution layers are where the features of the input image are extracted. Unlike in regular MLP layers, neurons in a convolution layer are arranged in feature maps. Each neuron in the feature map is connected to a sub-set of the neurons in the previous layer by trained weights. The inputs are convolved with the weights to produce the feature map. Each feature map has a set of trained weights that is used for each neuron in that feature map. Each feature map in the same layer will have a different set of weights to extract different features from the image. Since convolution can result in negative values, it is common to multiply the result by the ReLu activation function [8]. If the convolution result is negative, the ReLu activation function replaces the result with zero.

The purpose of pooling layers is to reduce the spatial resolution of the feature map and the sensitivity of the output to distortions and translations in the input [9]. Once the features of the image have been extracted, their exact location does not matter, just their relative location to other features.

B. Transfer Learning

The main disadvantage of ANNs is that they require a lot of data to train. It is common to have hundreds or thousands of images to train a CNN to recognize objects. Transfer learning is a method that utilizes a network that has already been trained for a particular problem, and re-training the last one or two layers to enable it to solve a different problem. There are many CNN models available online that have been trained on large datasets to classify certain objects. The hidden layers of these models have already been trained to extract important features from an input image. Transfer learning is

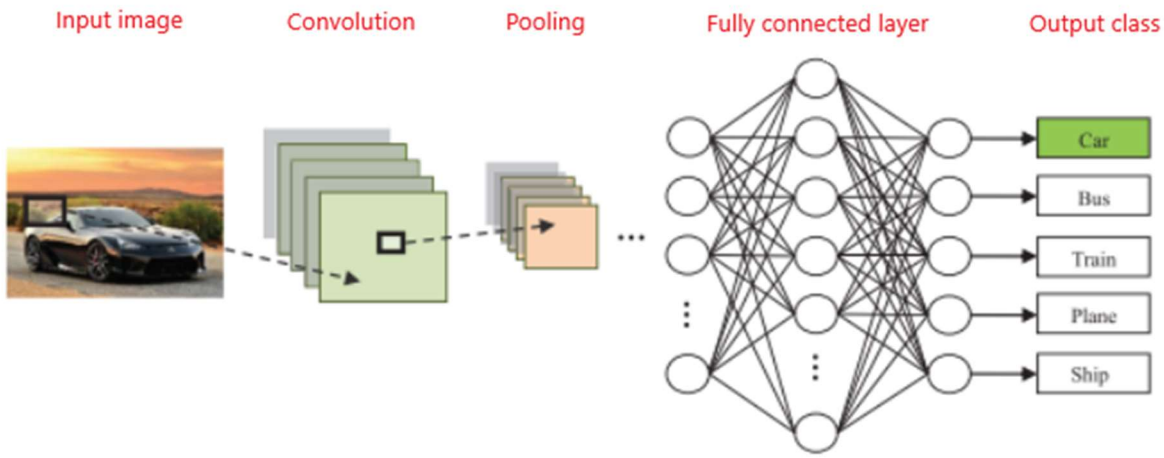

Figure 2: General architecture of a CNN, adapted from [6]

the process of copying a pre-trained model, removing the output layer, adding a new output layer appropriate for the new classification problem, and training only this new output layer to classify the extracted features. This significantly reduces the amount of training, and data needed.

C. CNN Training

Training a CNN model to classify images involves training it on datasets where the class of each image is known. AS the model predicts what class each image belongs to, the error is measured and minimized by adjusting the model's parameters. The error is measured as a loss function. Common loss functions are mean squared error loss, mean absolute error loss, and cross entropy loss.

It is common to split the training datasets into two categories; train and validate. It is useful to see how the model performs on images that it is not being trained on. This gives a better idea of how the model will perform outside of training. The validation set is like a test set, where the model's parameters are not updated after prediction.

Training a model occurs over several epochs. An epoch is a single iteration of training where the model is trained on a subset of the train dataset and tested on a subset of the validation dataset.

D. Segmentaton

Segmentation is used to separate objects and characteristics in an image [3] [4]. Segmentation can be used to separate lines, colors, discontinuities, textures, points etc. In gesture recognition, the only part of the image of interest is the hand, all other features of the image can be ignored. Two types of segmentation relevant for gesture recognition are line segmentation, and color segmentation [9].

E. Morphology

Morphological filters are commonly used in image processing to remove or highlight features from a segmentation. Relevant morphological operators include opening and closing. Opening removes thin lines and noise from an image, closing removes holes [11].

F. Static Hand Gesture Recognition Based on Convolutional Neural Networks

This paper [1] proposes a method of identifying static hand gestures from images using a convolutional neural network. The proposed method includes an image processing stage with the following operations: color segmentation using an MLP network, morphological operations, contour generation and polygonal approximation. After the image is processed, a binary image is obtained and a logical AND operation is performed on the original image. The resulting image is then used to train the classifying CNN.

The segmentation of the image was done by an MLP network trained to perform skin color segmentation. The network architecture has two hidden layers, with 5 and 10 neurons respectfully. The network took a single pixel as an input, with three neurons in the input layer to represent each RGB tone. The output layer has one neuron to classify the pixel as skin or non-skin. The MLP was trained using 5 images.

After segmentation, to remove noise and holes within the hand region, morphological erosion and closing operations are applied. First the erosion is applied using a horizontal line structuring element with the size of 9 pixels. Second the closing operation is applied using a square element with dimension 13-pixels. Finally, a polygonal approximation of the contours is applied to remove noise.

One issue the researchers found with the processed images is that the segmentations of certain gestures were very similar, due to their similar outline. This would make it hard for the classifying CNN to distinguish between different gestures. The proposed solution is to use a binary instance of the segmented images as masks in a logical AND operation with the original image in greyscale. By doing this, the information of the palm and fingers within the gestures are kept.

G. *Using a Deep Learning CNN to Translate the New Zealand Sign Language Alphabet to Text*

This paper [2] proposes a method for using a CNN to translate the static hand gestures of the New Zealand Sign Language. The method includes using the pre-trained CNN Inceptionv3, and using transfer learning to modify the network to classify 26 letters of the sign language. Using transfer learning on a pre-trained network is an efficient way to test research as creating and training a CNN from scratch can be time consuming.

Data sets were acquired by filming the different hand gestures within a boundary window, and each frame of the video was saved as an image. This approach of data acquisition results in the CNN model being trained on sets of images that are all taken on the same hand, and in the same lighting and camera angle. Additionally, the gestures in the images will have little variation as they were taken from one instance of performing the gesture. The boundary window was used to help find where in the image the hand gesture will be. This adds a constraint on how the image needs to be taken, if the gesture is outside this window, the model will not be able to classify it

III. METHOD

This research was done on Linux 19.1 Cinnamon OS with Intel® Core™ i7-8700 CPU processor operating at 3.20GHz. A Logitech C170 webcam was used to collect photos. Code was developed using Wing 101 IDE. Pytorch 1.5.1 was API used for implementing the CNN.

The method proposed involves pre-processing the image before passing it to the neural network to classify. The processing stage includes the following operations: Gaussian blur, color segmentation, morphological open and close, and contour generation. In this paper, two databases were used. A self-obtained dataset containing 5 images of each of the 26 gestures, and a public dataset in the literature [9]. Transfer learning was used on VGG16, a pre-trained image classification network trained on the ImageNet dataset.

The online dataset contained 1815 images of American Sign Language gestures. There was 5 sets of different hands, and multiple photos at different angles and in different lighting conditions. The self-obtained data set contained 10 images of each gesture, at different angles.

A. *Pre-processing*

Each stage of the image processing is shown in Figure 3. Color segmentation – To decrease the amount of information passed to the classifier and to increase accuracy, color segmentation is used to extract the hand from the image. Gaussian blur with kernel size (19x19) is applied to reduce noise, then a color threshold is applied to create a binary mask of the image. Morphological operations are then applied to reduce noise and clarify the shape of the hand. The open operation has a kernel size of (5x5) and 4 iterations, the close operation has a kernel size of 3 with 3 iterations.

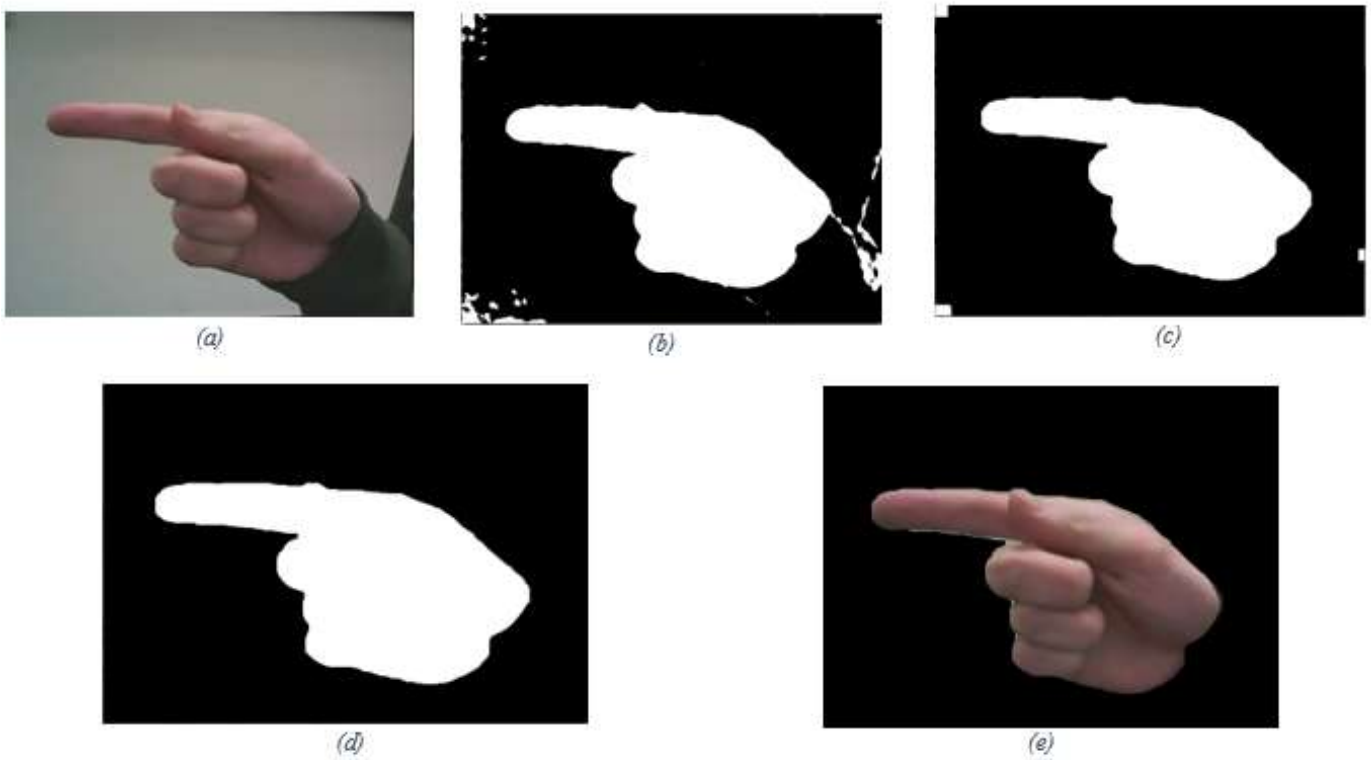

Figure 3: Pre-processing stages. (a) Original image, (b) After color thresholding, (c) after morphology, (d) after largest contour, (e) after bitwise mask with (a)

The final segmentation is obtained by finding and filling the largest contour. To retain hand features like finger position, nails, and lines, a binary and operation is applied to the final mask and original image. The image is then cropped around the contour.

B. Classifier

The original VGG16 model has 13 convolution layers with the ReLu function applied to their outputs, 5 max-pooling layers, and 3 fully connected layers with ReLu. The output layer is soft-max for classifying. The VGG16 model is shown in Figure 4.

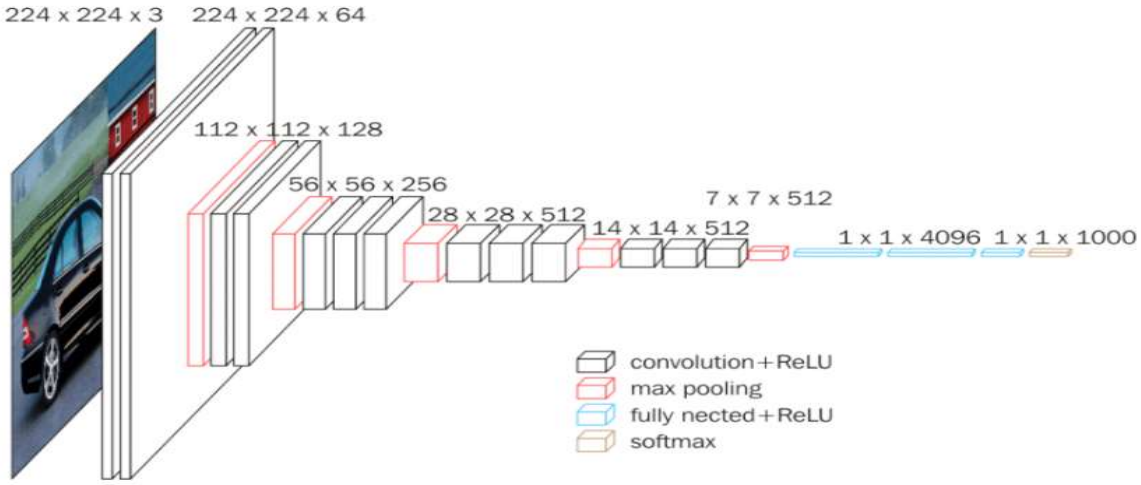

Figure 3: VGG12 Classifier Model [10]

Transfer Learning was applied to VGG16 to train it to classify 26 gesture classes. The output layer was removed, and another layer was added with 1000 neurons, as well as a new output layer with 26 neurons. The parameters for the rest of the model were frozen, so they were not re-trained.

The model was trained on both datasets, over 10 epochs, the loss function used was Cross Entropy Loss. Of the training data 10% was kept for validating during training.

C. Testing

The performance of the method was tested on another self-obtained dataset, with 10 images of each gesture. The photos were taken with different backgrounds, lighting conditions, and at different orientations than the images the classifier was trained on.

IV. Results

A. CNN Training

Throughout training the model, the accuracy after each epoch was recorded along with the model's parameters. A graph of the model's accuracy throughout training is shown in Figure 5. The model's accuracy reaches 99.6% after the fifth epoch, the remaining 5 epochs increase the accuracy to 99.9%. The model was reloaded with the parameters after the fifth epoch to avoid over-fitting.

As well as training, the model was validated on a test set of images in each epoch. Figure 6 shows the model achieving 100% accuracy in epoch 5.

B. Testing the method

The results for testing the model on a second self-obtained dataset of 5 images for each gesture is shown in Table 1.

Table 1: Model testing results

Letter	Accuracy
A	1.0
B	1.0
C	1.0
D	0.8
E	1.0
F	1.0
G	1.0
H	0.8
I	1.0
J	1.0
K	1.0
L	1.0
M	0.6
N	1.0

O	1.0
P	1.0
Q	1.0
R	1.0
S	0.8
T	0.6
U	1.0
V	1.0
W	1.0
X	0.8
Y	1.0
Z	0.2

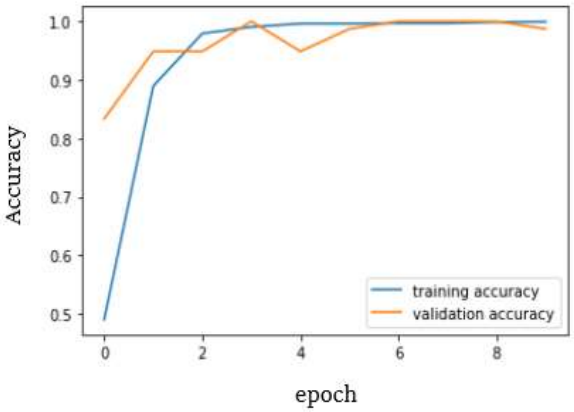

Figure 4: CNN accuracy during training

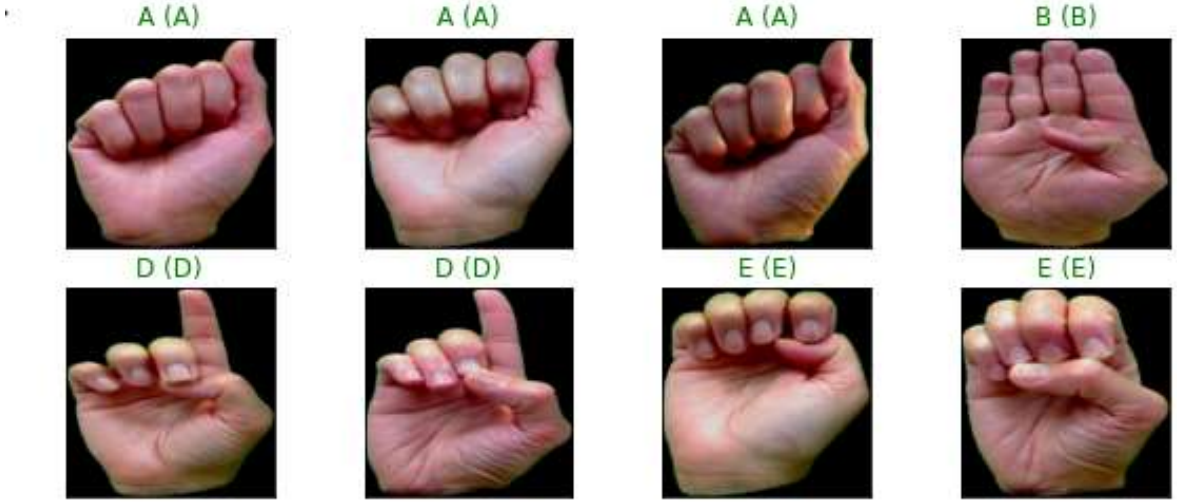

Figure 6: Results from validation test during epoch 5 (Green for correctly predicted class, Red for incorrect)

C. *Limitations*

This method uses color segmentation to extract the hand from an image. This requires that the color ranges selected to extract from the images are matched to the skin color of the user. The background of the image needs to be ensured to be a sufficiently different color than the user hand or it will be included in the processed image to be classified.

V. Conclusion

The method proposed in this paper aims to improve upon previous years work at recognizing static sign language letters using a CNN. The requirement that the hand gesture needs to be performed within a bounding box was removed. By extracting the hand with color segmentation, the gesture can be recognized anywhere in an image.

After the hand is extracted, the background is removed by masking the original image. This improved the classifiers accuracy at classifying images with different backgrounds.

The CNN had overall an accuracy of 93.3% on the testing dataset. The results for the letters J and Z were excluded as they are dynamic gestures.

References

.

[1] RF Pinto, CD Borges, AM Almeida and IC Paula, "Static Hand Gesture Recognition Based on Convolutional Neural Networks", University Federal do Cear´a, 2019.

[2] Kane Findlay and Richard Green, "Using a Deep Learning CNN to Translate the New Zealand Sign Language Alphabet to Text", Computer Vision Lab, University of Canterbury, 2019.

[3] R. C. Gonzalez and R. E. Woods, "Color Image Processing" in *Digital Image Processing*, 3rd edition, 2006, pp. 331 – 339.

[4] R. Szeliski, "Segmentation" in *Computer Vision: Algorithms and Aplications*, 2010, pp. 235 – 270.

[5] LM Belue, "Literature Review" in *Multilayer Perceptrons for Classification*, 1992, pp. 9 –20.

[6] W Rawat and Z Wang, "Overview of CNN Architecture" in *Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review*, 2017, pp. 3 – 7.

[7] B Póczos, "Feedforward Neural Networks" in *Advanced Introduction to Machine Learning, 2017, pg. 14.*

[8] AF Agarap, “The Model” in *Deep Learning using Rectified Linear Units (ReLU)*, 2018, pg. 2.

[9] CY Lee, PW Gallagher and Z Tu, “Generalizing Pooling Operations” in *Generalizing Pooling Functions in Convolutional Neural Networks: Mixed, Gated, and Tree*, 2016, pp. 2 – 3.

[10] S Yuheng and Y Hao, *Image Segmentation Algorithm Overview*, 2017.

[11] A. L. C. Barczak, N. H. Reyes, M. Abastillas, A. Piccio, andT. Susnjak, “A new 2D static hand gesture colour imagedataset for ASL gestures,” Research Letters in theInformationandMathematicalSciences, vol. 15, no. 4356, pp. 12–20, 2011.

[12] NeuroHive https://neurohive.io/en/popular-networks/vgg16/ cnn model

[13] Dr. Romik Chatterjee “Image Processing Fundamentals” https://en.wikipedia.org/wiki/Mathematical_morphology [June. 24, 2020

[14] N. A. Ming-Hsuan Yang and M. Tabb, “Extraction of 2dmotion trajectories and its application to hand gesture rec-ognition,”IEEETransactionsonPatternAnalysisandMachineIntelligence, vol. 24, no. 8, pp. 1061–1074, 2002.

[15] . V. den Bergh, D. Carton, R. De Nijs et al., “Real-time 3Dhand gesture interaction with a robot for understanding di-rections from humans,” inProceedings of the 2011 20th IEEEInternational Symposium on Robot and Human InteractiveCommunication(RO-MAN), pp. 357–362, Atlanta, GA, USA,July 2011.