Multi-Task Learning of CNN For Facial Attribute Recognition

Chongyu Yuan
Simon Fraser University
chongyuy@sfu.ca

Qingwei Chen
Simon Fraser University
qingweic@sfu.ca

Xiaotong Liu
Simon Fraser University
xla246@sfu.ca

Yiming Zhang
Simon Fraser University
yza440@sfu.ca

GitHub: https://github.com/yiming-zh/CMPT419-Project

Abstract

We describe a single Convolutional Neural Network (CNN) with Multi-Task Learning (MTL) method to recognize different attributes of human faces, to be precise, given a face image, it can simultaneously deal with two different tasks: distinguish between male and female and whether they wear eyeglasses, by using only one model. In addition, we show MTL's advantages and improvement by comparing it with the traditional method, that is, using 2 independent models, each model is only responsible for one task (male/female) or (with/without glasses).

1 Introduction

CNN are now widely used in computer vision, natural language processing and other fields. However, a traditional CNN can only be responsible for one type of classification task, i.e. it has only one loss function. When we want to do multiple different classification tasks, e.g. the male-female and glasses detection tasks mentioned above, we not only want to distinguish the gender, but also recognize whether she/he wear eyeglasses, usually we need multiple models to be responsible for different tasks. But this is not the best option as these two tasks use some common features in the image from an image processing perspective, so we can put these two tasks in only one model and let them share some layers. And then, data flow into different branches, each branch is only responsible for their own tasks. To evaluate the model, there must be two loss functions that used to assess the accuracy of gender and eyeglasses identification, respectively. Refer to Sebastian's paper [1], hard parameter sharing [2] (Figure 1) is a suitable method for our needs.

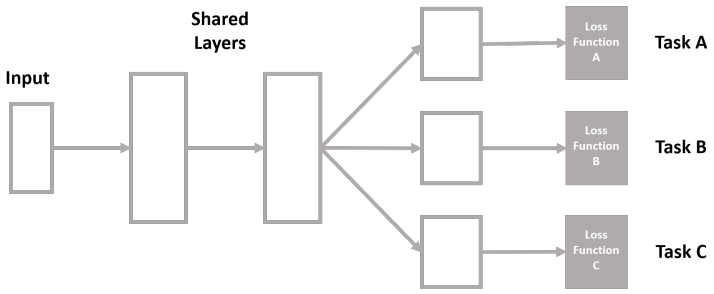


Figure 1.1: Hard parameter sharing model

This method has been proved that can improve generalisation performance on the tasks in the training set. [3]

2 Approach

2.1 Dataset

Our dataset contains a total of 7000 human facial images, the interesting thing is that most of them are generated by NVIDIA's StyleGAN [4]. This is an excellent open source AI which can generate virtual but realistic human face images.

Figure 2.1: Face images generated by StyleGAN

In addition, we also put some real faces in the training set to prevent the images generated by AI from having too many common features. The photos in the test set are all real faces. After many experiments, we found that the network trained by the training set generated by styleGan performed very well on the test set. After collecting the data, we use OpenCV to crop images to 50x50 pixels and make sure that there is only the face part in each image. Next, all images are converted to grayscale images, and the grayscale range of each pixel is converted from 0 - 255 to 0 - 1, since the facial features we want to recognize do not require color information.

2.2 Network Structure

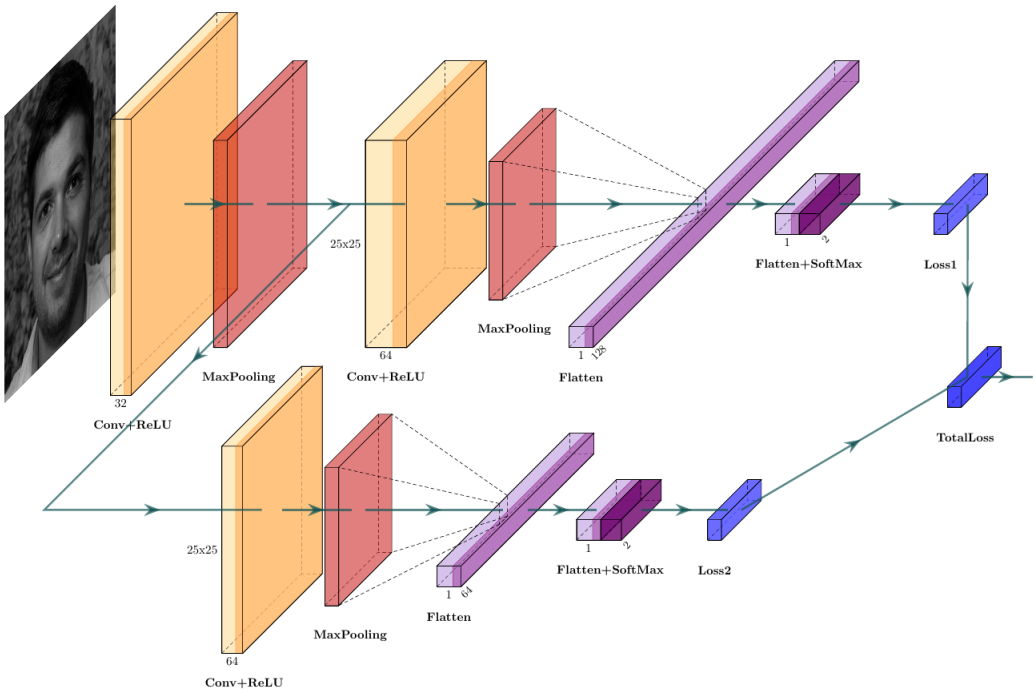


Figure 2.2: Our MTL CNN model structure

As Figure 3 showed, after a 50x50 grayscale image enters the network, it will first be convolved (using 3x3 kernels, 1 stride size and ReLU activation function) and sub-sampled, Some early features are extracted, these features can be considered as common features shared by the the image information of gender and eyeglasses (Figure 4).

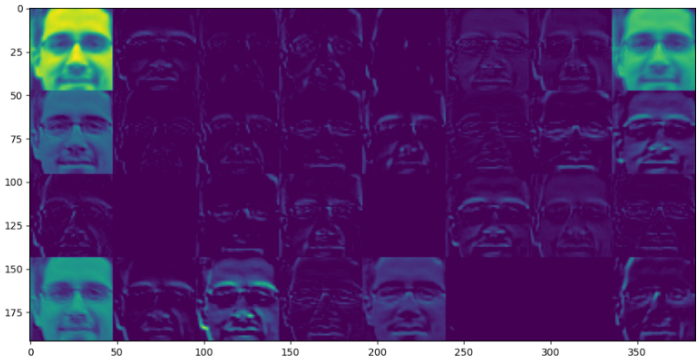


Figure 2.3: 32 feature maps from 1st convolutional layer

Then the network goes to two branches, which correspond to the two tasks of gender recognition and eyeglasses recognition respectively.

2.3 Loss Function

Each of these two classification tasks has its own independent loss function. The total loss function of the whole model is their weighted sum [5]:

$$Loss_{total} = \sum_{i=1,2} w_i * Loss_i$$

$$Loss_{i=1,2} = -\frac{1}{2}\sum_{x}[ylog(output) + (1-y)log(1-output)]$$

y is the true label of our data, output is model's actual output. If we only focus on one main task and make another task assist the main task, we should use different weights w_i. But for here we just pick $w_1 = w_2 = 1$, because two tasks are equally important for our needs. Therefore, the loss function of each branch will affect the total loss function. When backpropagating, each branch will update their independent parameters first, and then the update of the shared part parameters will be affected by both branches. To a certain extent, these two branches are not completely independent, this can reduce the risk of overfitting for each branch, which is also the reason [3] says MTL can improve generalisation performance. Another benefit is that we reduce the total amount of parameters in this way, which shortens the training time. We will show this later in comparison with the traditional network.

2.4 Training model

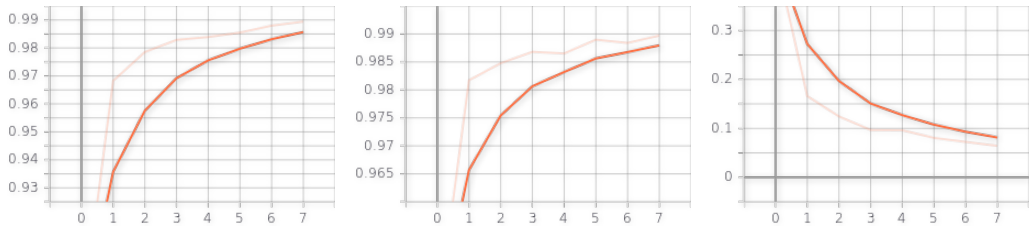
To further prevent overfitting, we use the method of early stopping, that is when the loss function of validation set stops falling in more than two epochs, we manually terminate this training. In order to accelerate the convergence of the loss function, we used a dynamic learning rate [6]. After many rounds of experiments, we found that the following formula is a better strategy to update learning rate.

$$L_i = L_{i-1}(1 - \frac{1}{Epoch_i}), L_0 = 0.001$$

Under this optimization and using 32 as the batch size, our model usually converge well in no more than 10 epochs.

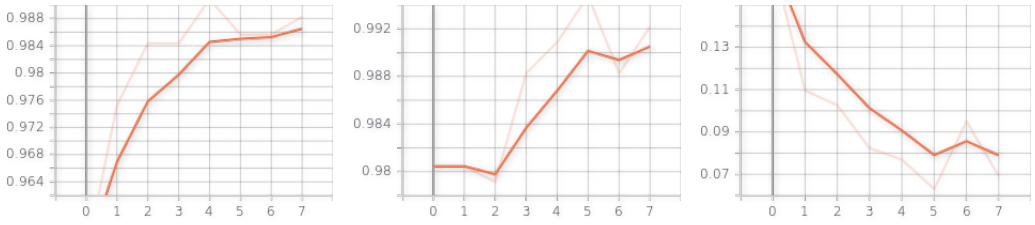
3 Result

The accuracy of this MTL CNN on the training set, validation set (with 10-fold cross-validation) and test set reached 98.89%, 97.83%, 97.77%, respectively.



(a) Accuracy of glasses recognition (b) Accuracy of gender recognition (c) Total loss function

Figure 3.1: Result on train set



(a) Accuracy of glasses recognition (b) Accuracy of gender recognition (c) Total loss function

Figure 3.2: Result on validation set

We used photos of real human faces that the model has never seen before to test, and it turns out that even if the training set is generated by StyleGan, the accuracy on real human faces is still very high.

Figure 3.3: Result on test images

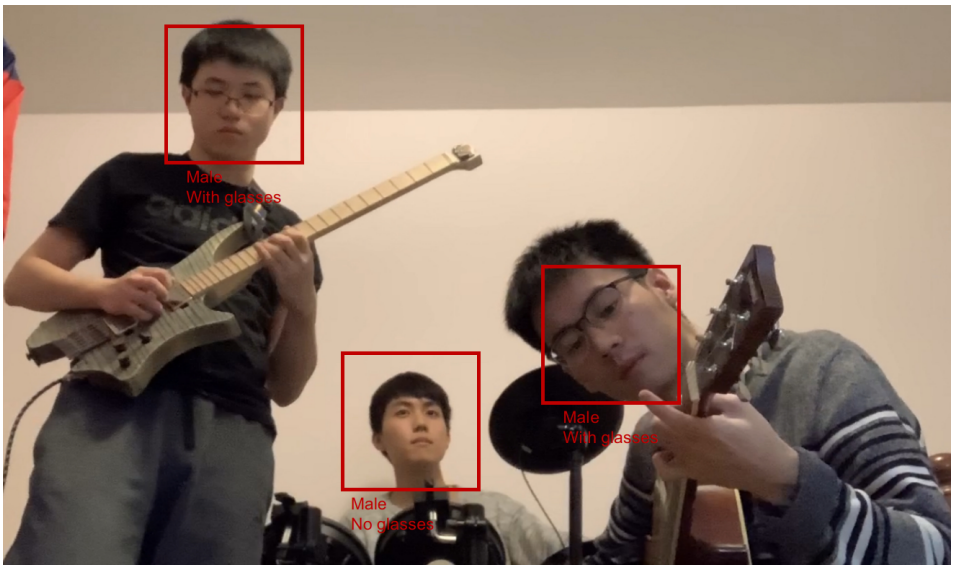


Figure 3.4: Test on our band

3.1 Comparison between MTL CNN and multiple CNNs

We also built a multiple CNNs model perform the same tasks, after ten experimental comparisons and averaging the results, we produced the following table

	MTL CNN	Multiple CNNs
Loss on train set (Avg)	0.0632	0.0509
Loss on validation set (Avg)	0.0639	0.0813
Acc on train set (Avg)	98.65%	98.64%
Acc on validation set (Avg)	98.79%	98.85%
Total trainable parameters	1524740	1525060

Figure 3.5: Comparison of two models

Since the classification tasks are quite simple and the network structure is not complicated, the difference between the two model is very small. But we can still see that the performance of MTL CNN model on the validation set is very close to the training set, which is better than multiple CNNs model. To a certain extent, this also proves that MTL can improve generalisation performance. We believe that when the task is sufficiently complex, more advantages of MTL will be reflected.

4 Conclusion

We explored the application of MTL in facial feature recognition and verified the feasibility and advantages of MTL through a simple classification task of gender and eyeglasses recognition. We described how MTL works as well. The method introduced in this project can be further extended to perform more tasks, such as inferring age and facial expression from more facial features. Even it can be applied to more fields such as self-driving cars. We found that Tesla is already using this technology [7]. The most important prerequisite is that these tasks must be logically or have related features, otherwise MTL is useless. There are also some difficulties such as which layers should be shared? This requires our further exploration.

References

[1] Ruder, Sebastian. "An overview of multi-task learning in deep neural networks." arXiv preprint arXiv:1706.05098 (2017).

[2] Caruana, Rich. "Multitask Learning: A Knowledge-Based Source of Inductive Bias ICML." Google Scholar Google Scholar Digital Library Digital Library (1993).

[3] Baxter, J. A Bayesian/Information Theoretic Model of Learning to Learn via Multiple Task Sampling. Machine Learning 28, 7–39 (1997). https://doi.org/10.1023/A:1007327622663

[4] Karras, Tero, Samuli Laine, and Timo Aila. "A style-based generator architecture for generative adversarial networks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[5] Kendall, Alex, Yarin Gal, and Roberto Cipolla. "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[6] Brownlee, J. "Understand the Impact of Learning Rate on Neural Network Performance." Mach. Learn. Mastery. https://machinelearningmastery. com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/(accessed 12.12. 19) (2019).

[7] Andrej Karpathy. Multi-Task Learning in the Wilderness (2019) https://slideslive.com/38917690