# Facial Recognition in Objects Placed in Noisy Background (RNN/CNN)

This report is based on the experiment conducted in the paper “*Simple Learned Weighted Sums of Inferior Temporal Neuronal Firing Rates Accurately Predict Human Core Object Recognition Performance*”. Where a collection of Pov-Ray objects was placed in a random noisy background that would not match the object in theme, Neural networks were then tasked to identify which image contains a face.

To replicate the experiment, the Pov-Ray objects were taken from the online collection [[1]](#_[1]_https://colah.github.io/posts/2)while the backgrounds were using those that were already available by default in the Pov-Ray program. There were extremely little facial objects in the collection online and thus the manipulation of the camera angle was used to increase the number of pictures that can be taken on the same object. This also introduces variety to the training data, though this is only beneficial to objects that look different from different angles, as compared to a basketball object aside from the shadows. This report followed the tutorial [[2]](#_[2]_https://stackoverflow.com/quest) on handling of pictures which are kept on Google Drive and used on Google Colab platform.

The original pictures created through the Pov-Ray program were fixed at the resolution of 1920 by 1080 pixels, however the RAM provided on the platform of 25GB was exhausted and the neural network could not be trained. Therefore the pictures had to be rescaled down to 25% of the original resolution, as working with 50% resolution still exhausted the RAM. Through the Google Colab platform, a GPU was used on this experiment which greatly reduce training time, with the exception of the RNN neural network that was trained with the TPU (Tensor Processing Unit). Where the RNN required multiple hours per epoch on the GPU however only required an hour per epoch on the TPU.

The training data after been fed and read into the platform, it had to be shuffled otherwise the positioning will be taken advantage of by the neural network and is an unwanted behaviour. With the limited training data, a smaller number for epoch and batch size was considered to prevent overfitting.

The first model as the control was a simple single layer Dense network which was simply unable to make any real predictions aside. Making a blind guess of all 1s or 0s would result in 50% accuracy. Where the test data was generated with 12 non-face pictures and 12 face pictures for a total of 24 test pictures that are generated with an unseen background.

The second model was the RNN network of a single layer of GRU units, despite having only 4 GRU units the training time was extremely long compared to previous experiments in NLP. This might be attributed to the input data of image, which have already been reduced to 25% of the original. As such, only 2 runs were made on the RNN where the latter run had achieved 70.83% accuracy. An important note is that the loss function for both runs were on the decline, so that might possible mean that the RNN can continue learning or it will start overfitting. Unfortunately, overfitting could not be tested as running the model would require extended period of training time.

The third model was using a convolution model of 1-dimension, which would be view the input data similar to the RNN as the spatial information is not retained. The validation and training accuracy peaked fairly early on the 2nd epoch, however on hindsight it is due to overfitting. As the training data is heavily biased towards non-face pictures, then the neural network is incentivised to predict all as 0 which is non-face. The fourth model was also using the convolution model of 2-dimension, however also performed the same.

Thereafter, some informal tests were conducted to probe at a possible reason as to why the CNN were not performing as expected. In fact, they performed as poor as the control, where no learning has seemed to taken place. As seen in the raw prediction results from the sigmoid dense node, all were flat 0 values.

It is later found that having an additional pooling layer, increasing kernel size to (5,5) and reducing filters seemed to have increased accuracy slightly. Possible interpretations might have been the pooled image was still too noisy, the CNN focused on the wrong pattern or overfitting has occurred.

The fifth model was then a dual layer convolution with reduced filters and dual layer pooling. Occasionally, the model was performed extremely well with accuracy of 95.83% (only one mistake in prediction) as the model was also able to get a perfect 0% with the predictions all wrong. The CNN model was at this point, learning however it seemed to learn different patterns which will lead to various results. Including the variations, this made the model unstable, despite setting seed all over the script.

It is found that by adding a third layer of convolution with reduced filters helped stabilise the model, as it was able to achieve accuracy of more than 95% more reliably. It would seem that the this addition has helped the CNN focus on the right pattern to help identify facial recognition as the accuracy does not dip back below the 50% mark.

# Further Actions

The training data for this experiment was extremely limited and that might have likely constrained this experiment to a notable degree. A more robust test data would have been able to test the neural networks better.

The elevated cost of training the RNN has also prevented hyper tuning and as such, the RNN performance could be considered muted this time round. However, in terms of cost-efficiency the RNN does not deliver. Although the GPU has made the training of the CNN much cheaper, the final CNN model seemed to be able to deliver consistent remarkable results over the RNN.

At this point, this experiment seems to support that CNN is the optimal neural network structure to tackle spatial recognition task. As compared to RNN, however as seen through the hyper tuning phase of the CNN, the spatial data had to be processed to an optimal degree or risk overfitting. If the RNN were to have access to the processed data of the CNN, it may be possible to combine the benefits of both networks.

If the direction into spatial data were to continue, a better object collection would be required to substantiate the training data creation. Otherwise, the neural networks that have not been exposed to a realistic amount of variation is unable to confidently project predictions on unseen objects or backgrounds, making extrapolation extremely dangerous and thereby limiting the practical usage of said neural network.

# References

## [1] <http://objects.povworld.org/>

## [2] <https://www.analyticsvidhya.com/blog/2019/01/build-image-classification-model-10-minutes/>

## 