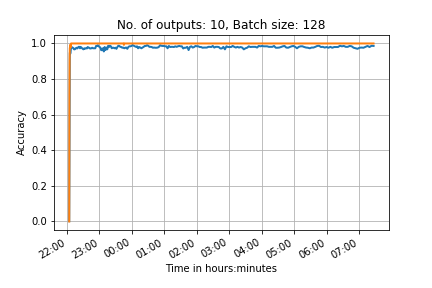
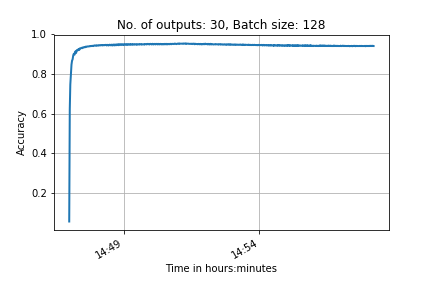
**Weekly report**

*Work on transfer learning*

Successfully transferred from a neural network with ten outputs to a network with 30 outputs yielding 95% accuracy. This must now be compared with training for 30 outputs from scratch.

Below is the training graph for training on ten outputs for the usual CNN structure. In addition is the training graph for retraining the last layer.





*Work on Neural network class*

Neural networks are now easy to build using the class that Seb finished. The class has been successfully implemented numerous times.

*Work on Generative Adversarial Networks (GANs)*

Using example code online built a running program of GAN for our Chinese characters.

Unfortunately it doesn’t seem to be minimising the loss created by the generator, and the loss by the discriminator falls at first but then plateaus when it reaches the generators loss. Also, it takes a long time to run because it has so many convolution outputs and layers. For now we can decrease the number of outputs, but we have to keep the layers the same (to ensure it de-convolves well).

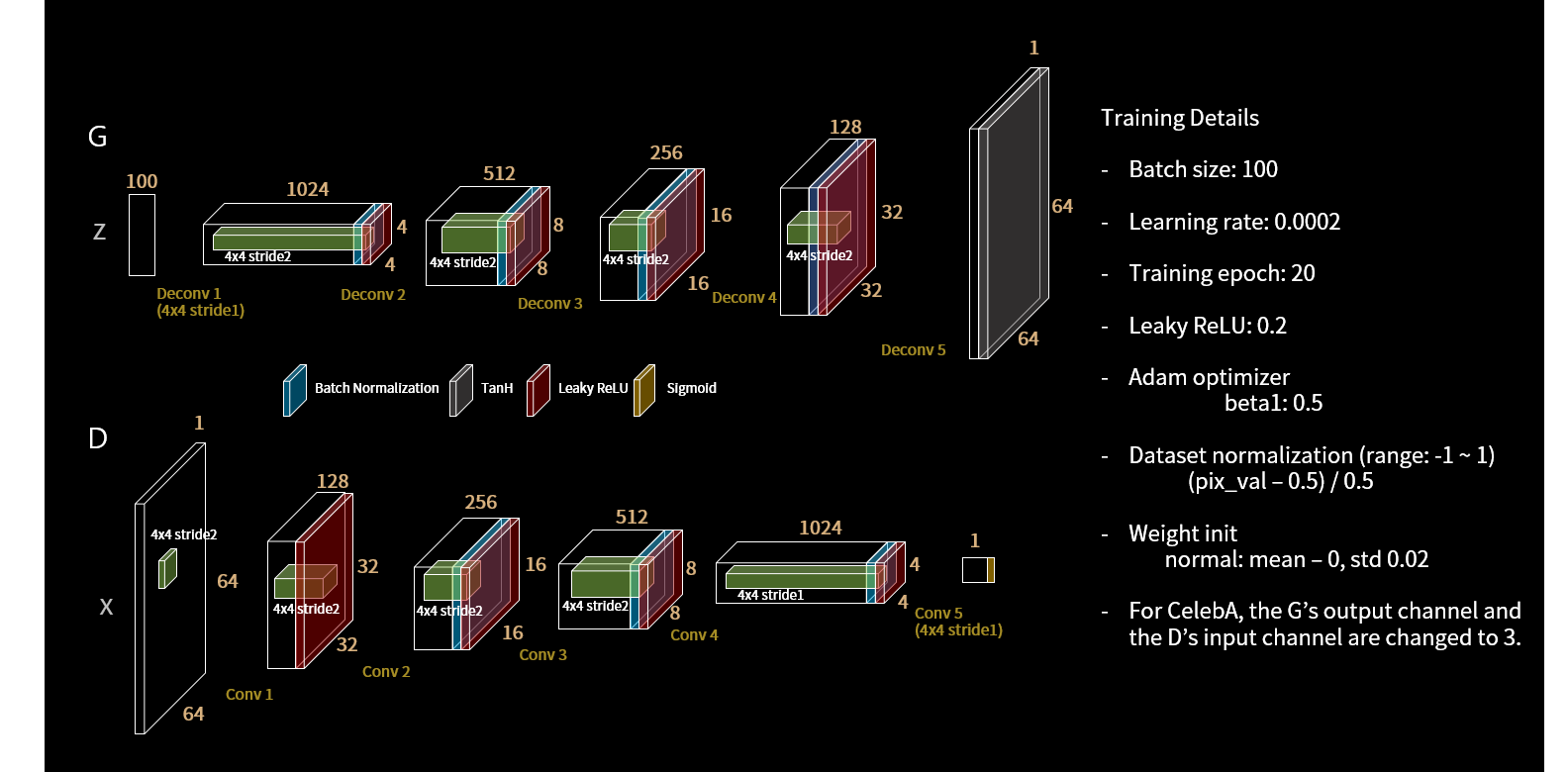


Figure 1: Model of the GAN we are using. This is called a DC-GAN

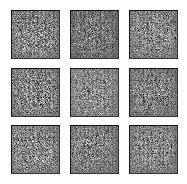


Figure 2: Example of random images created by the GAN. If all goes well these will turn into Chinese characters once the GAN is trained

Elliot mentioned we can use transfer learning for the discriminator part of the GAN. This is a really interesting point and requires research.

**Action points for the next week**

1. Carry on working on visualising the feature maps of our neural network for 100 outputs .

2. Train a network for 30 outputs to compare with transfer learning

3. train a network for 100 outputs and transfer learn to 3866 outputs

4. Get the GAN code to work and generate some Chinese characters using it

5. Figure out how to do ‘vector image addition’ using the GAN to turn calligraphy writing into handwriting-esque characters

6. Segment the entire Baotu Spring poem

7. Figure out how to incorporate both training and test data for the GAN