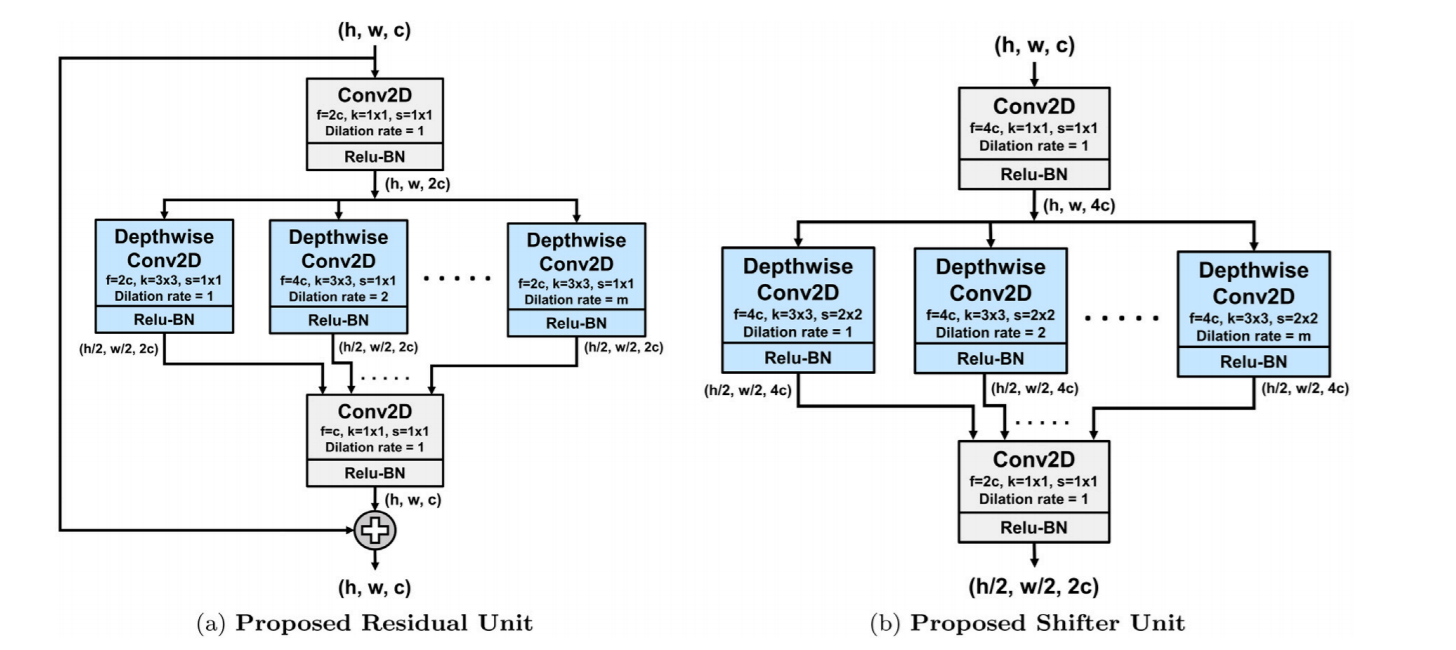
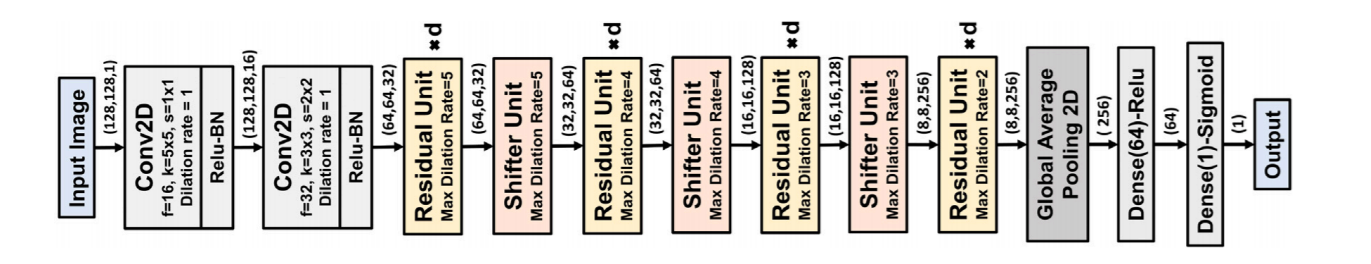


Dilated Convolution for different dilation rates with kernel size (𝟑 × 𝟑) are encompassing different receptive areas. With increased dilation rate, the receptive area also gets bigger, though kernel size is kept unchanged.



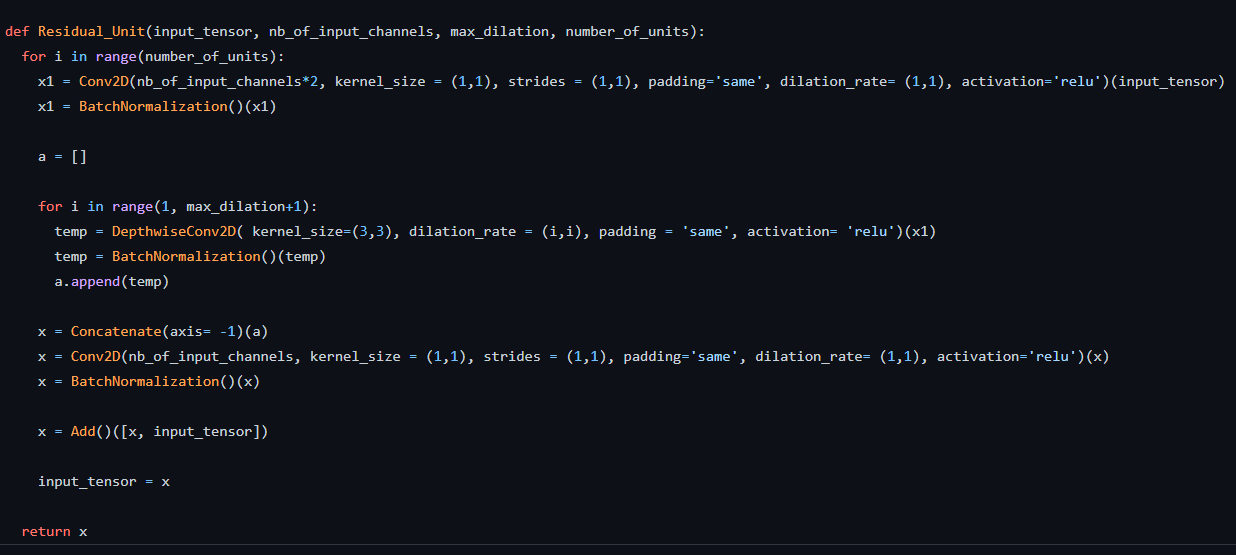
Proposed structural units. Here, 𝐡, 𝐰, and 𝐜 denote the height, width and no. of channels of the feature map, respectively, while ‘𝐤’ stands for kernel size, ‘𝐬’ for strides and ‘𝐟’ for number of filters in the convolution. In depthwise convolution, dilation rate will be varied from 𝟏 to ‘𝐦’.



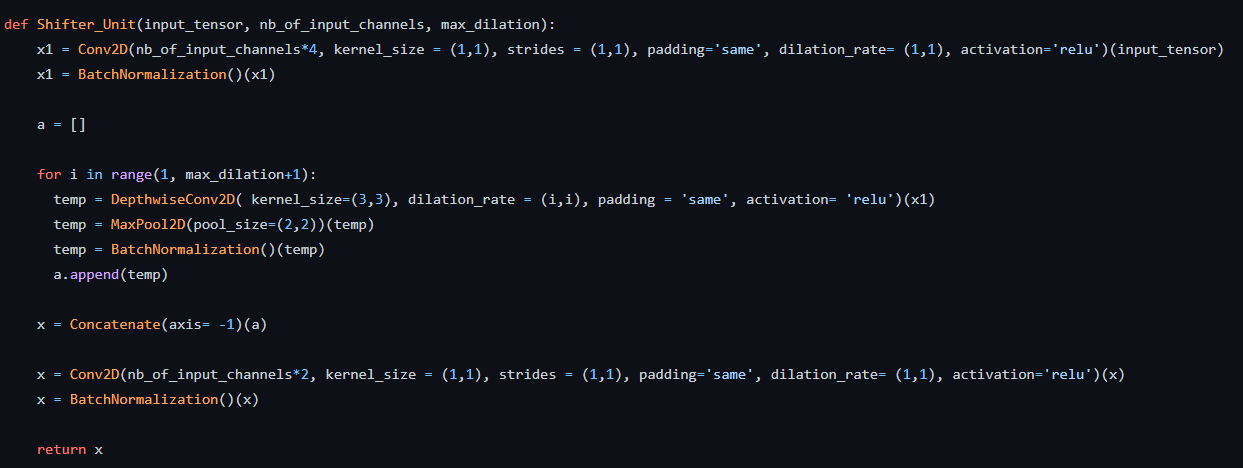
Schematic of the Proposed CovXNet architecture optimized for input shape (𝟏𝟐𝟖, 𝟏𝟐𝟖, 𝟏). Each residual unit is replicated for ‘𝐝’ times.

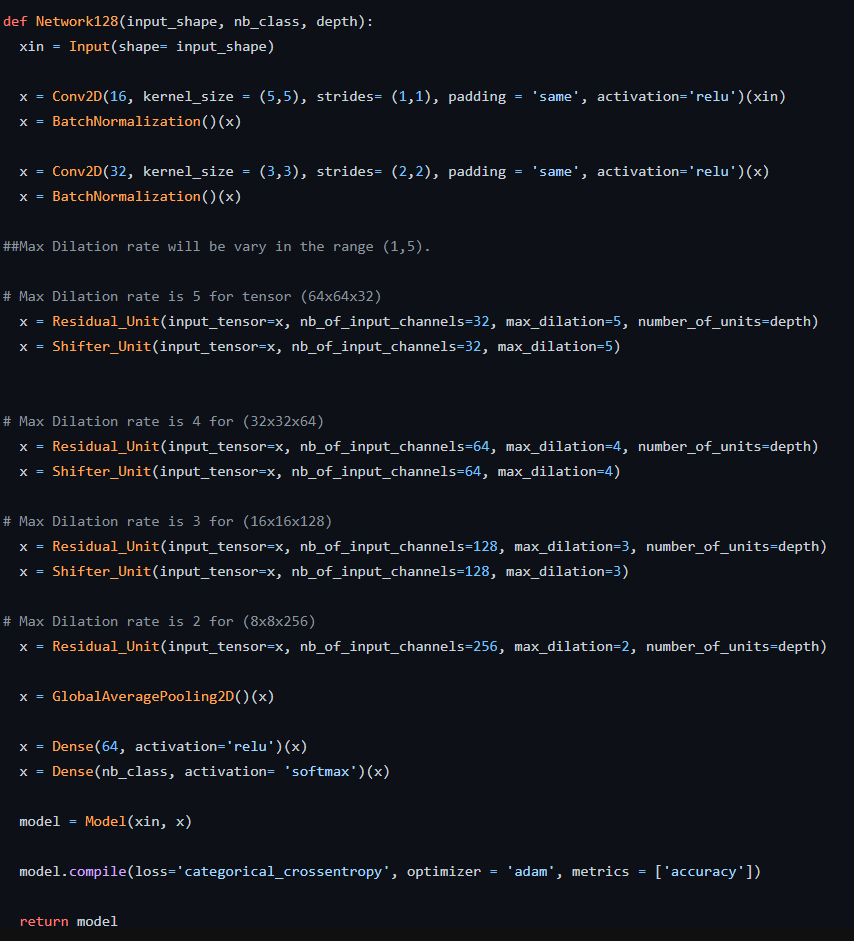
The residual and shifter units are the main building blocks of the proposed CovXNet architecture, as shown in Fig. 4. Firstly, the input image undergoes convolutions with broader kernels to process the information with the larger receptive area. The following convolution introduces some dimensional transformation. Afterward, it passes through a series of residual units. Depth of this stack of residual learning (𝑑) can be increased to produce a deeper network. Shifter units are incorporated in between such stacks to introduce dimensional transformation to generalize the extracted the information further. However, the maximum dilation rate (𝑚) of each residual unit is determined based on the dimension of the input feature map. For processing larger features, 𝑚 is set to be higher to increase the maximum receptive area of the residual unit accordingly to encompass more variations in the extracted features. Finally, the processed feature map passes through global average pooling followed by some densely connected layers before providing final prediction. Moreover, the rectified linear unit (ReLU) is instigated after each convolution for non-linear activation with batch normalization to make the convergence faster.

**RESIDUAL UNIT ->**



**SHIFTER UNIT ->**



**PROPOSED ARCHITECTURE ->**

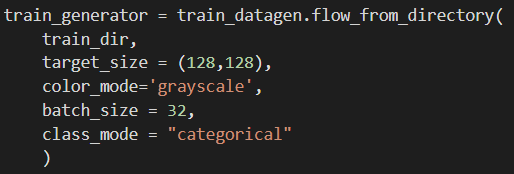


**CLASSES ->**

****

**TRAIN AND TEST SPLIT ->**

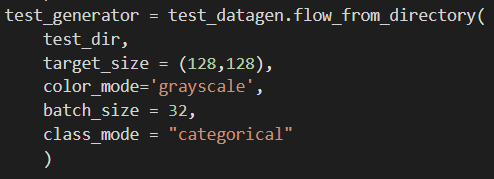
**TRAIN ->**





2 classes are forged and genuine in train folder

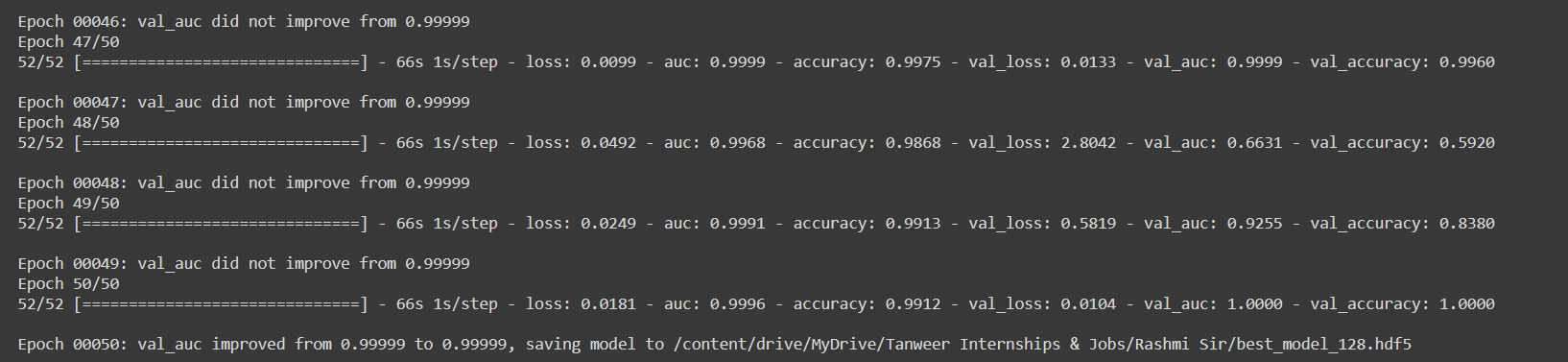
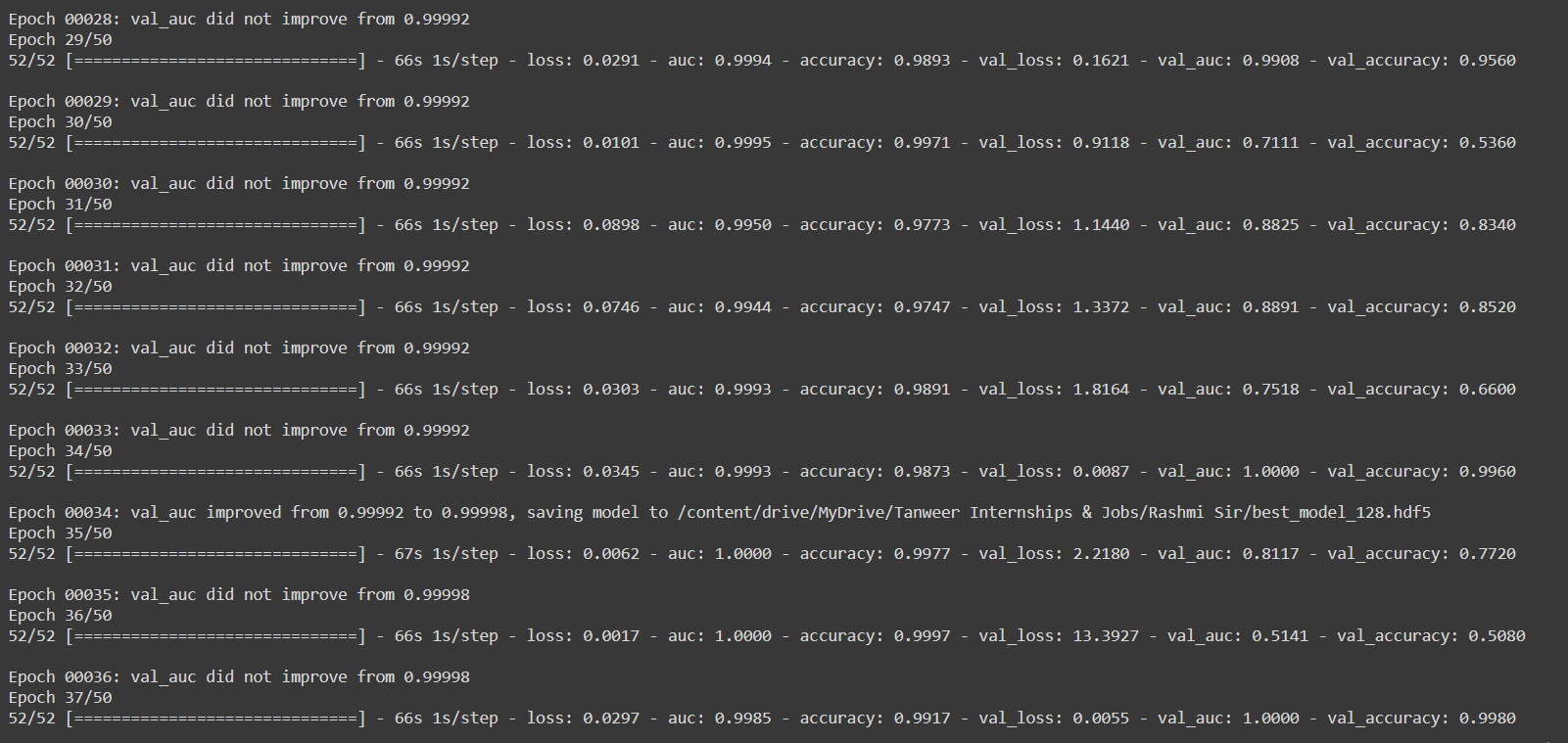
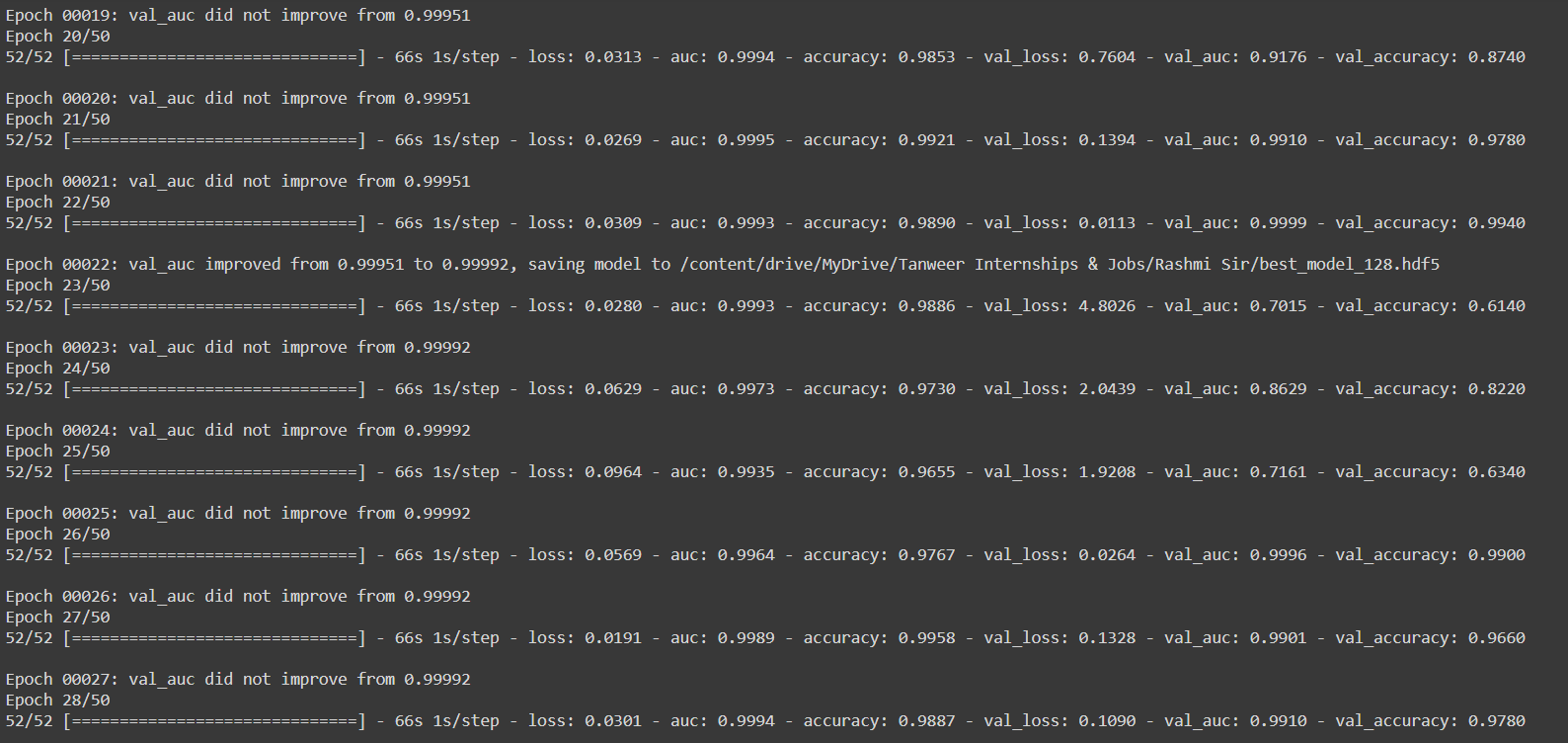
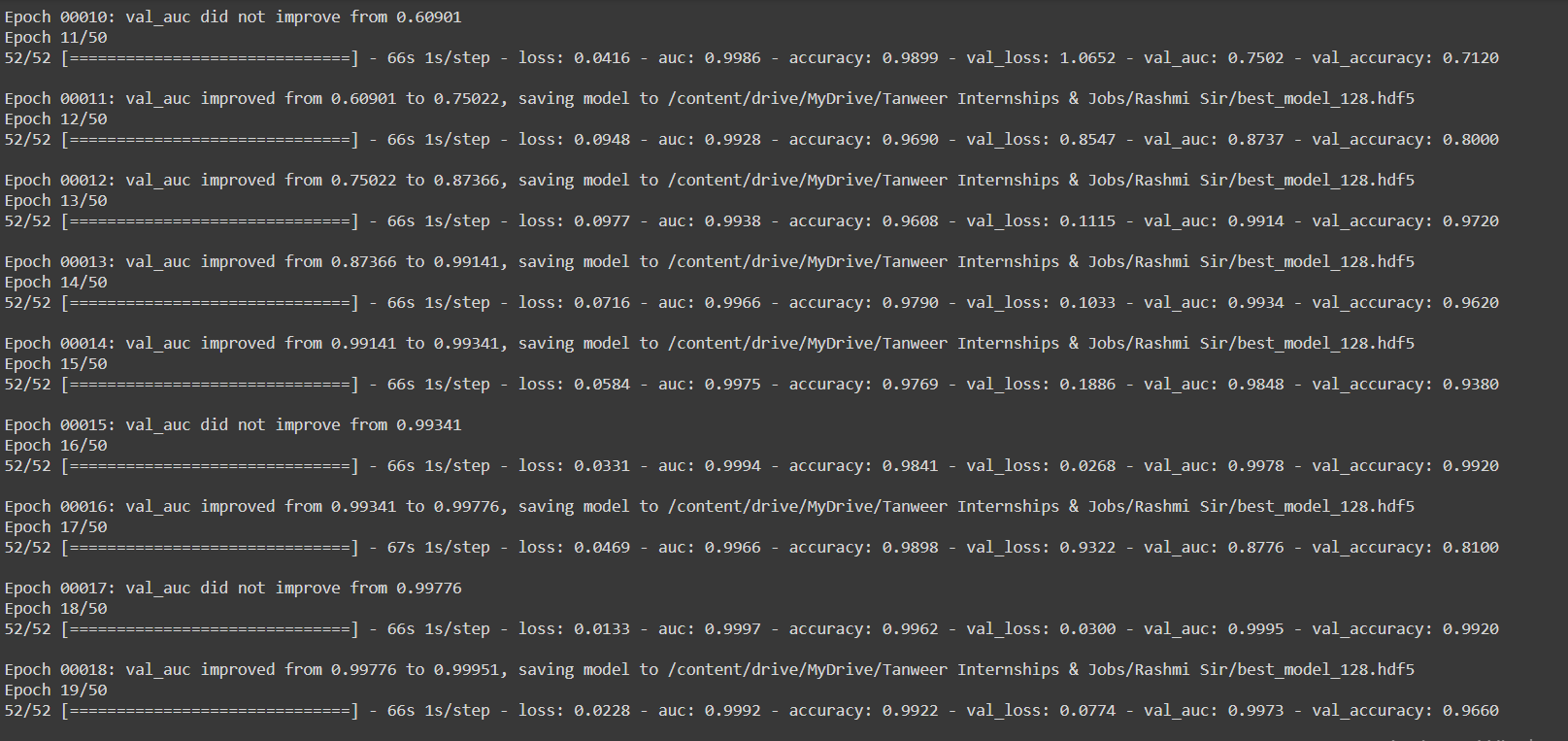
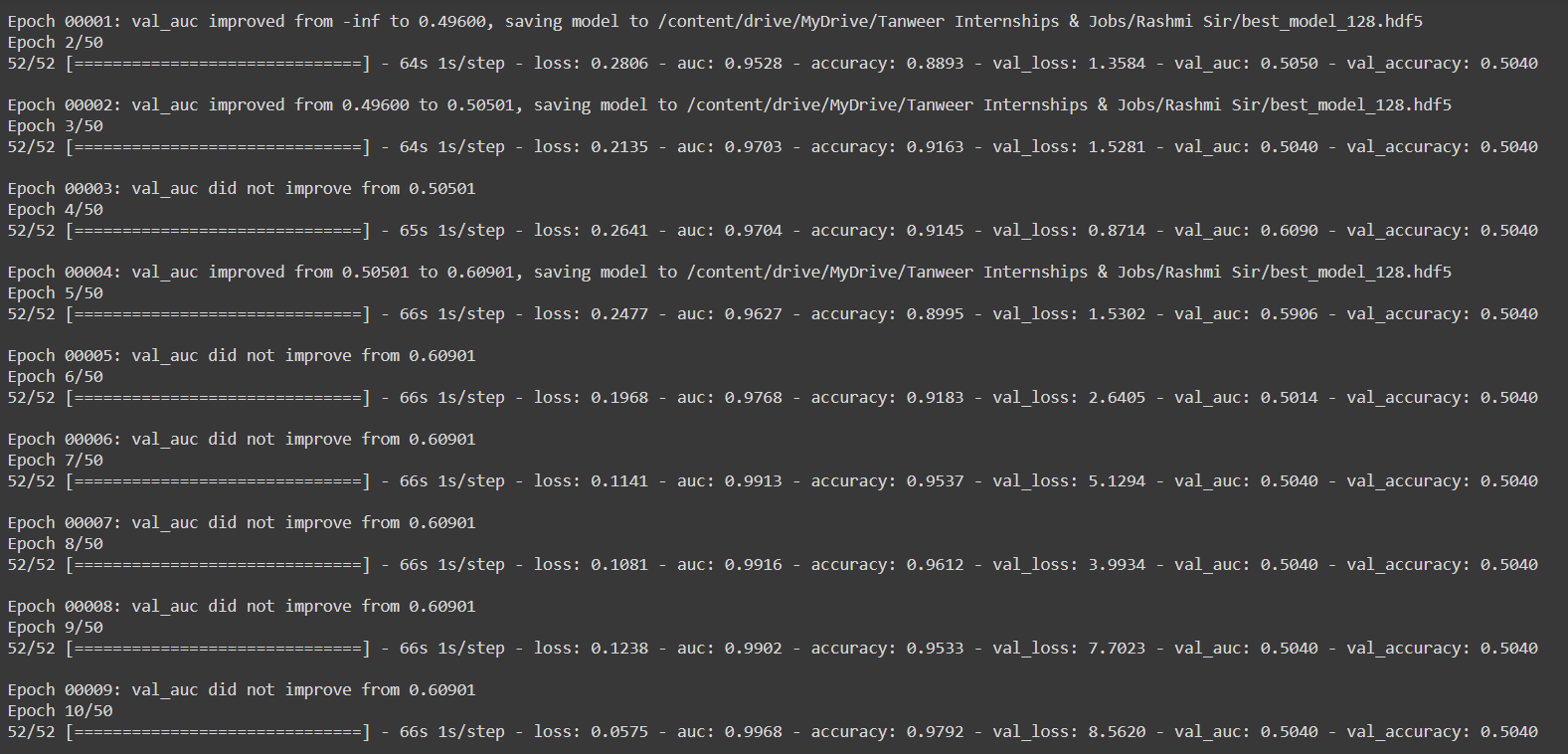
**TEST ->**

****

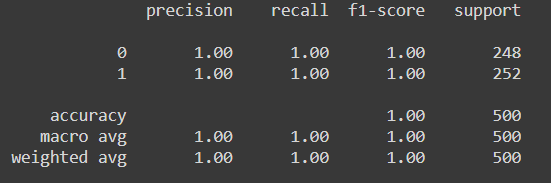
****

2 classes are forged and genuine in test folder

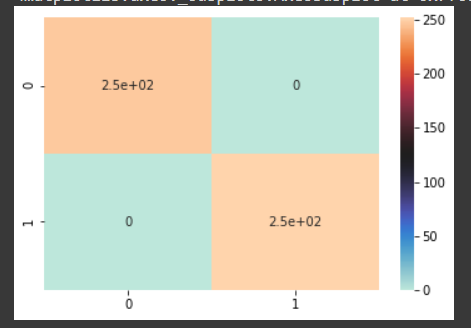
**TRAINING LOGS ->**

****

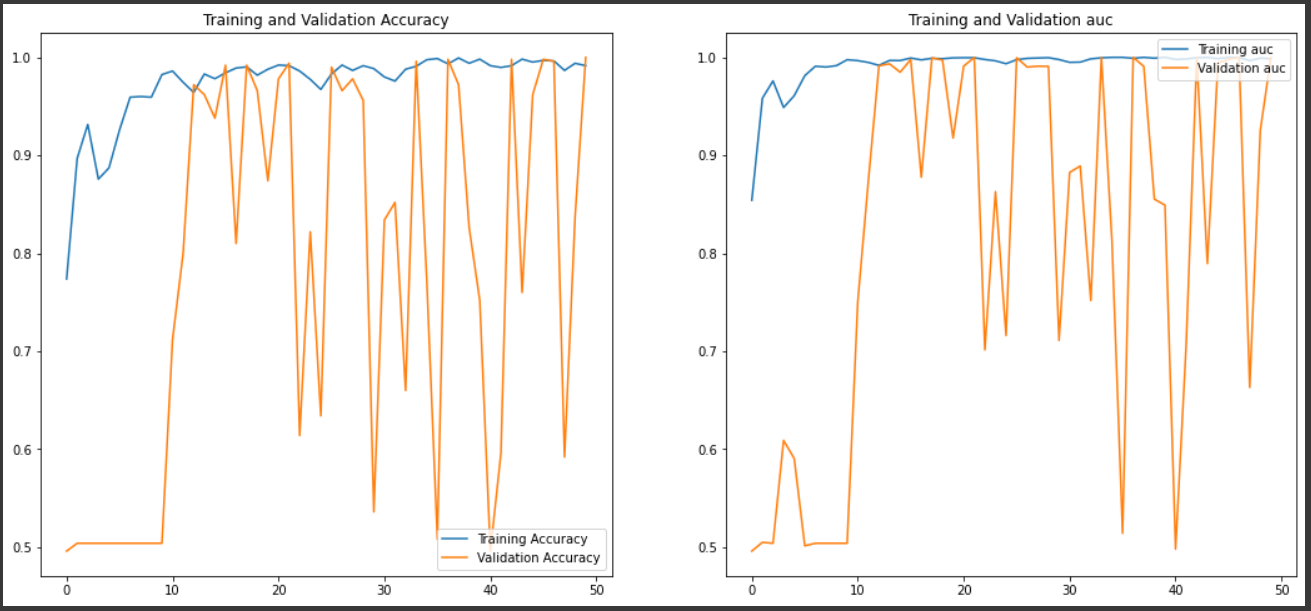
**CLASSIFICATION REPORT ->**

****

**CONFUSION MATRIX ->**

****

**PLOT->**

****