# Heart Lab Test

### Covid patient chest Xray

**Abstract**

This test is based on the chest radiograph of a covid patient. In this test, I divided the existing pictures into a test data set and a training data set, and then converted the pictures into a data set in the form of features for analysis. Then I used the open-source residual net algorithm for analysis. By the way, I create an interface to display my test result, in order to show a clear result for our customer.

**Reade-me file**

In this section, I would like to introduce my project as a read-me file format. I would like to explain every file clearly and show my idea for processing this test.

**Main.py:** This file is for PYQT interface. I invoked a package for PYQT, to display my final result clearly. In this file, I defined some parameters about PYQT and made the final result displayed better.

**Mian-GUI:** In this file, I defined some button’s functions for the interface, like start button or close button.

**Resnet50:** As for this file, it’s an open-source algorithm come from GitHub. I collected the experience and processing my project by using it. Through this file, I can generate a file with the suffix h5, and invoke this file directly in subsequent training and testing.

**train-resnet.py:** This file is for training. I defined the data set as two classifications and I transferred the image datasets as the digital one. I used the CNN model to train on the existing data set and got a model-ResNet50-final.h5 file.

**Test-image:** This file is for testing. According to training file, I tested the dataset, in order to compare the result.

**Utils.py:** This file is for pre-processing. I separate the dataset and pre-processed the image data set. I set up some parameters for the image data.

**Summary**

In the process of this test, I first analyzed the features of the pictures and converted the features into an array form. So that the computer can analyze our pictures. Secondly, through the training of the ResNet50 model, I obtained a file with model features. In the following steps, I can test the remaining data sets by calling this file. Finally, I wrote a window function to show the results that I got.

In this section, we must first figure out why we should use the CNN model to process image information. First, there are two points. The first point is that the CNN model has local relevance. In simple words, an image can be divided into individual pixels, and these pixels are related. If the order of these pixels is disrupted, then these images have no meaning. The second point is space invariance. CNN shallow network can extract low-level features. As the network deepens, low-level features will combine to form high-level features.

Why do we choose to use the ResNet model? The reason is because this model has a feature-skip connection. The ResNet model can skip each layer of CNN connections, so that our high-level connections are correlated with low-level connections, and good results can be obtained. We can train some more complex data sets through this model. By reducing the accumulation of layers, we can achieve better results.

For this test, I got a very good score, more than 90%. This shows that in this test, our results using ResNet are very good. However, due to the low-price comparison of the existing data sets, we are prone to overfitting results. I choose to use the ResNet50 model to reduce the occurrence of such problems. In addition, if the analysis of the characteristics of the image is not in place, the problem of unsatisfactory results is prone to occur. Therefore, for the analysis of image problems, we must not only clarify the characteristics of the image, but also choose the right adaptive algorithm model. Only in this way can we get a better result.

All in all, although this test is a challenge for me, I enjoined it very much. I really appreciate the test that can help me to gain more experience. Thank you very much!