**Experiment**

In tis section, the proposed method will be evaluated, which is done through training phase. There are two main process in training phase which is training and inference. Aforesaid that input from every neuron is multiplied by its weight and added with its bias to be an input to the activation function. In training phase, these parameters are adjusted so that the final output is same as expected value.

Before further discuss on training phase, the terms and data used for the training phase should be first comprehend. This data can be image, text, sound, time series and video and called data set \cite{datatype}. The data set will be split into two set, one for training process and another one for inference process.

The data set used for the proposed model is a set of images from \cite{dataset} that contain image of closed eyes and 37751 images of open eyes. In total 76408 images, where 80% of it will be used for training process to find the right value for the weight used in proposed neural network and 20 % for inference process to evaluate the trained proposed model. Since the proposed model used transfer learning method, the implemented part from inceprionV3 is not trained.

To handle the huge amount of input, the images are split into batches. The batch size is the number of samples that are passed to the network at once \cite{batch}. This is to increase the speed of traing because the number of errors that need to be stored is the number of the batch size instead of the error for each images \cite{batchreasons}.

To increase the accuracy of the model, the data set will be passthrough the model a few times. This iteration is called epoch \cite{epoch}. The number of epochs applied for training the proposed model is 5 and the batch size is 8. So, the number of batches for each epoch is 7640.

The proposed method is design using python environment and the API used for the model are from TensorFlow\footnote{}. The training phase was executed on computer with 64-bit operating system, x64-based processor, Intel(R) Core(TM) i7-1065G7, 1.50GHz and 16 GB RAM.

Training a network means nothing more than solving a complex optimization problem \cite{advance}. At first, the value of weight and bias for every neuron is randomly assigned. After that, an image is given to the model as an input. The final output value is compared with an expected value, where the difference value or known as loss value is recorded. The loss values are used by loss function to improve the value of the weight in the way that the loss value can be reduced in the future. The lost function that is used in proposed method is categorical cross entropy. In training process, it could happen that the model is “overfitting”. Means that the model has high accuracy on trained data but low accuracy in real environment. To reduce the overfitting, the learning rate is reduced. The algorithm used to achieve that is called optimizer. For the proposed model, the optimizer used is Adam \footnote{}.

The trained model is then proceed to the inference process where the model accuracy is evaluated, where the remaining data set is used. The result from the whole training phase is shown in ``Fig. \ref{epochaccuracy}’’, where the model have accuracy of 95% on final epoch. The blue line is the value during training and the orange line is the value during validation or inference. The loss on each epoch is visualized in ``Fig. \ref{epochloss}’’. ``Fig. \ref{iterationaccuracy}’’ and ``Fig. \ref{iterationloss}’’ visualize the accuracy and loss for every iteration.

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated