**Introduction**

As the popularity of digital cameras is increasing, many have a huge collection of digital pictures. Even the photographers who take pictures in ceremonies like weddings, birthday parties and anniversary celebrations confess that almost 40% of the images taken by them, have a less quality of pictures. One of the reasons for such poor quality of pictures is blurriness in the images. Therefore, Blur detection on these images helps the photographers or users to classify the images so that one can move all the blurred images into a separate folder so that their efforts in separating out the images becomes less.

Blur detection is more challenging as it is hard to distinguish the blur type, blur level and blur setting . It segments the blurred parts of an image accurately. It is an important application in salient object detection, image restoration, defocus magnification, deblurring, blur segmentation and so on. According to the cause of Blurness,the blur is divided into 2 types i.e., out-focus and motion blur. Blur detection is a baseline for many applications as mentioned above. Usually, Blur is purposefully added by the photographers or the cameraman to add effects to the image. This skill is very common by optical imaging systems. All these hand-crafted feature-based methods are convenient and effective.

There are many methods in the field of machine learning where we can detect blur in a given image and classify the image as blur or clear image. Some of the methods are using variance of Laplacian using OpenCV library in python, using MLPCLassifier of sklearn library, implementation of Convolutional neural networks using TensorFlow and keras libraries in python. These methods give our desired output in a single floating value point which represents the class i.e., blur or clear.[5] These methods differ in giving the accurate results.Though,some methods can give good and appropriate results, other methods can give an inappropriate result while some may give accurate results based upon how we work in each method.

**Terminology**

**Related work**

In the beginning, we have used variance of Laplacian using OpenCV where we get the threshold using that we get a blur rate to classify the image. Using this we have got a less accuracy ~62% then using the Sklearn library available, we have used MLP classifier to train our model with that trained model we have got an accuracy of ~78%. And upon analyzing the things we came to know that there are some corner cases such as image of a plan wall, image of a picture taken in portrait mode. These were wrongly classified. Then we have decided to implement Convolutional Neural Networks using Keras and TensorFlow which we found as the simple and easiest technique. It gives the output as a single floating-point value which represents the class of the image i.e., blur or clear.

We have used Convutional neural networks on a pretrained model i.e., Unet for implementing blur detection on the images. The Unet Architecture is shown in figure.

A CNN-trained model based on a dataset which contains training and evaluation directories in which there exists other two directories namely Naturally-blurred, artificially-blurred and undistorted image directories in training set and naturally-blurred and artificially-blurred directories under evaluation set of Certh image blur dataset.

Our dataset uses two types of blurs i.e., natural blur and artificial blur to classify the images. When there is any disturbance in the movement of the camera while clicking the picture, the image obtained is said to be naturally blurred. If the lens of the camera is not focused, then the image obtained is said to be artificially blurred. If the image is clear without any disturbance, then it is said to be undistorted.

As we are giving any kind of images without any specifications, we may get some specific use cases where the system may give an unpredicted answer. Some of those use cases include   
1) An empty wall which should be interpreted as a clear image,   
2) An image taken in Portrait mode from an iPhone which should be portraited as a clear image,  
3) An image which contains 2 ppl standing diagonally in which a person standing away is clear whereas another person is blurred. This scene should give us a blur verdict,

These are some of the use-cases which are implemented in real life