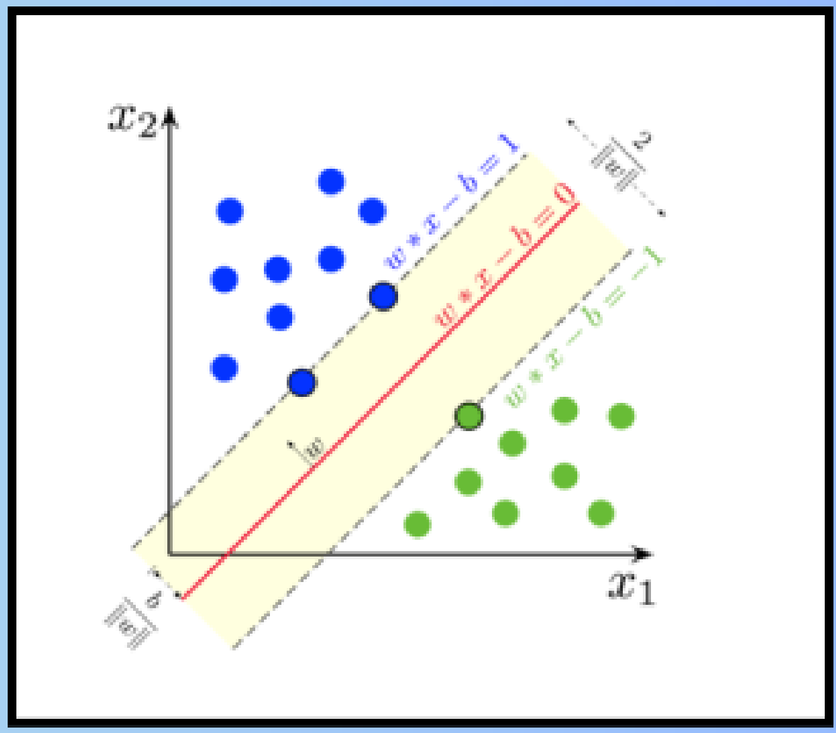
Human Activity Recognition using Kernelized SVMs and CNNs

By:-

Arjit Singh Arora 2021452

Barneet Singh MT23028

Aman Sharma 2021010



Submitted To:

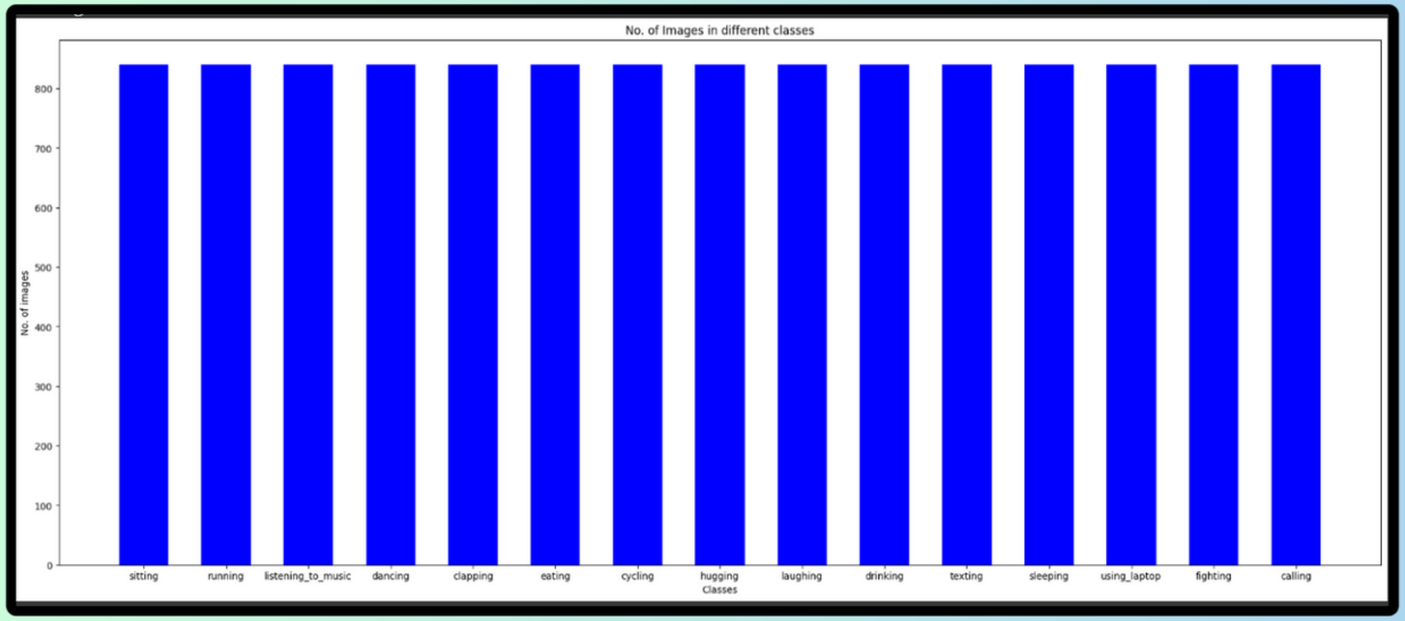
Dr. Vinayak Abrol

Problem Statement

Build an Image Classification Model using SVMs/CNNs that classifies to which class of activity a human is performing

About the Dataset

- The dataset features 15 different classes of Human Activities.
- The dataset contains about 15k+ labelled images including the test images.
- Created a Train-Val-Test split of (70:15:15) of the data

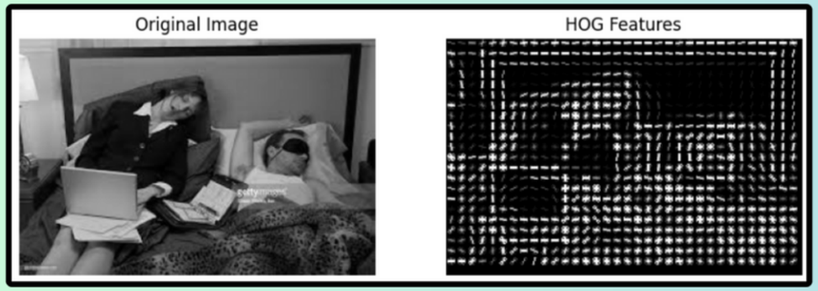


All the 15 classes in the HAR dataset have the same number of images, hence, there will be no class imbalance based on the no. of images per class

About the Dataset

Classical ML based approaches

EDA and Data Preprocessing

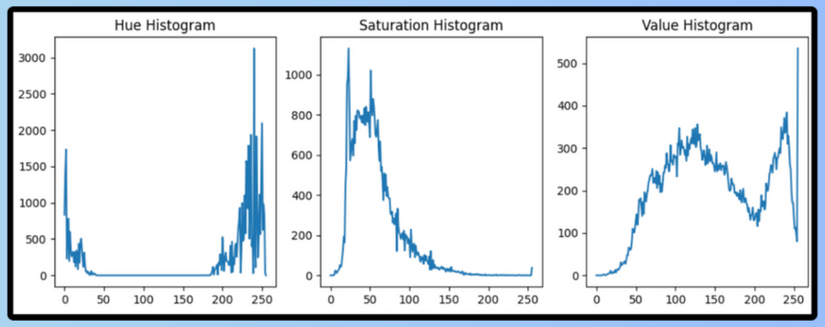


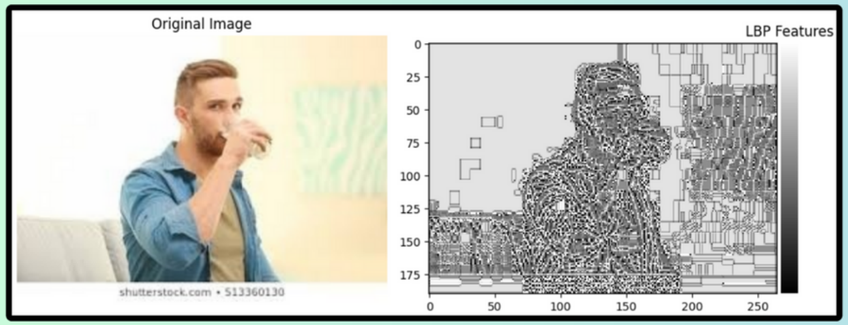
HISTOGRAM OF ORIENTED GRADIENTS

HOG, is a feature descriptor which is used in for the purpose of object detection. The technique counts occurrences of gradient orientation in the localized portion of an image. For the regions of the image it generates histograms using the magnitude and orientations of the gradient.

HSV FEATURES

The HSV features are well-suited for color analysis and feature extraction. By separating color information into hue, saturation, and value components, you can perform tasks like color-based object detection and tracking. This can be done by using colour histograms.

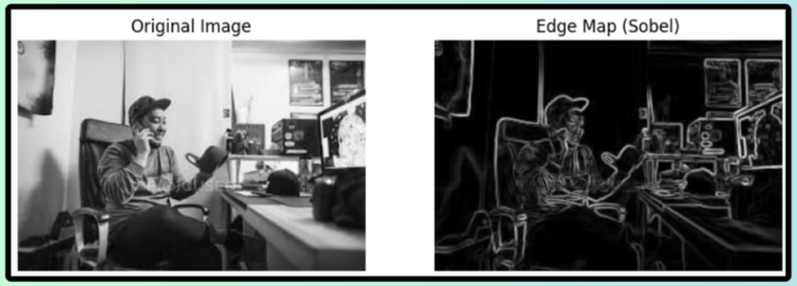




LOCAL BINARY PATTERNS

LBPs compute a local representation of texture. This is constructed by looking at points surrounding a central point and testing whether the surrounding points are greater than or less than the central point (i.e. gives a binary result).

EDA and Data Preprocessing - II

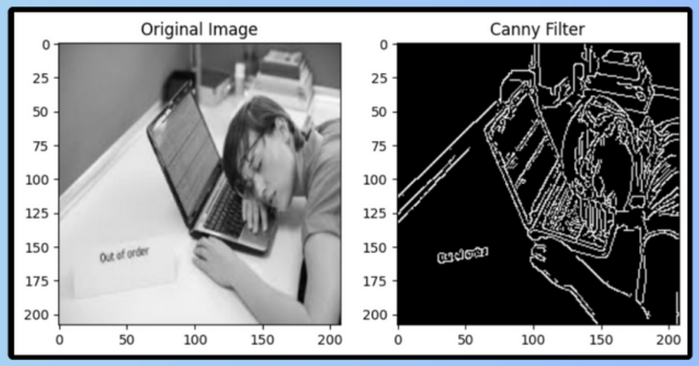


SOBELS FILTER

Sobel's Edge detection filter is used to detect edges in the images . These can be converted to statistical features like mean, median etc. or flattened to convert into feature vectors.

CANNY FILTER

The Canny edge detector is an edge detection operator that uses a multi-stage algorithm to extract useful structural information from different vision objects and dramatically reduce the amount of data to be processed .



SIFT FEATURES

The SIFT features are local and based on the appearance of the object at particular interest points, and are invariant to image scale and rotation. They are also robust to changes in illumination, noise, and minor changes in viewpoint.

EDA and Data Preprocessing - III

GAUSSIAN AND BILATERAL FILTER FOR NOISE REMOVAL

The Gaussian filter and bilateral filter are both commonly employed for noise removal in image processing, but they differ in their approaches and outcomes. The Gaussian filter, uniformly blurs an image, effectively reducing noise but potentially sacrificing edge details. On the other hand, the bilateral filter, excels in preserving edges by considering both spatial proximity and intensity differences.

OTHER TECHNIQUES

- **Min-max scaling:** For a constant scale as SVMs are sensitive to the scale of the features
- **Template Matching / Patch Similarity:** It finds predefined patterns in images, aiding in object recognition and feature localization.
- **Contours:** It identify edges and shapes in images, pivotal for object detection, recognition, and boundary analysis.
- **Median filters:** Has high capablity of reducing noise , due to which edges got lost.

METHODOLOGY AND RESULTS

1) HOG + HSV + LBP features

We extracted HOG features from **grayscale images** and concatenated them with HSV features from **coloured images** . After this we joined them with LBP features to get a total of around **4500 features** . This combined feature matrix was passed to the SVM classifier where we used **Polynomial kernel with a degree of 6** for optimal results .

- Min- max scaler boosted accuracy by 4%

Accuracy on Cross validation: 35%

2) Bilateral Filter (for noise removal), Sobel filter (for BG seperation) then HOG + HSV + LBP features

Here we reduced all the extra noise and then used sobel filter to highlight only imp edges in the image. **All HOG feature extraction done on this was expected to increase accuracy by a lot , because of less noise** . However, the accuracy reduced .

Accuracy on Cross validation : 30%

METHODOLOGY AND RESULTS -II

3) Ensemble of HOG , HSV

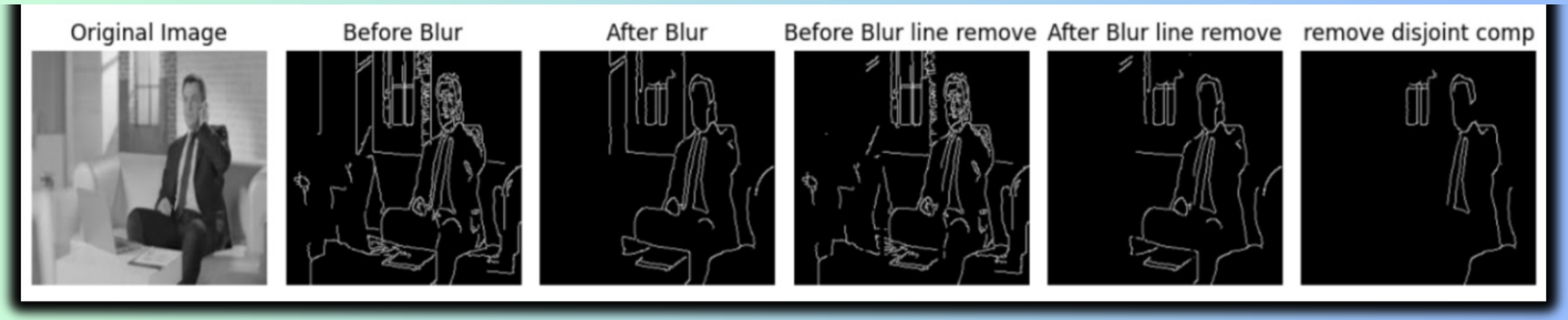
This time we created an ensemble of HOG and HSV features . 2 SVMs was trained on scaled versions of the two independently. Then at the time of classification we calculated the confidence score of the two SVMs the one which gave better was finally chosen for prediction.

Accuracy on Cross validation: 31%

4) SIFT FEATURES + HSV + LBP

SIFT, HSV, and LBP combined for richer image analysis, leveraging keypoints, color details, and local textures. Their fusion enhanced model understanding for more nuanced image interpretation. We tried with only one row of SIFT features for each image because it was giving extra features for each iteration.

Accuracy on Cross validation : 25 %



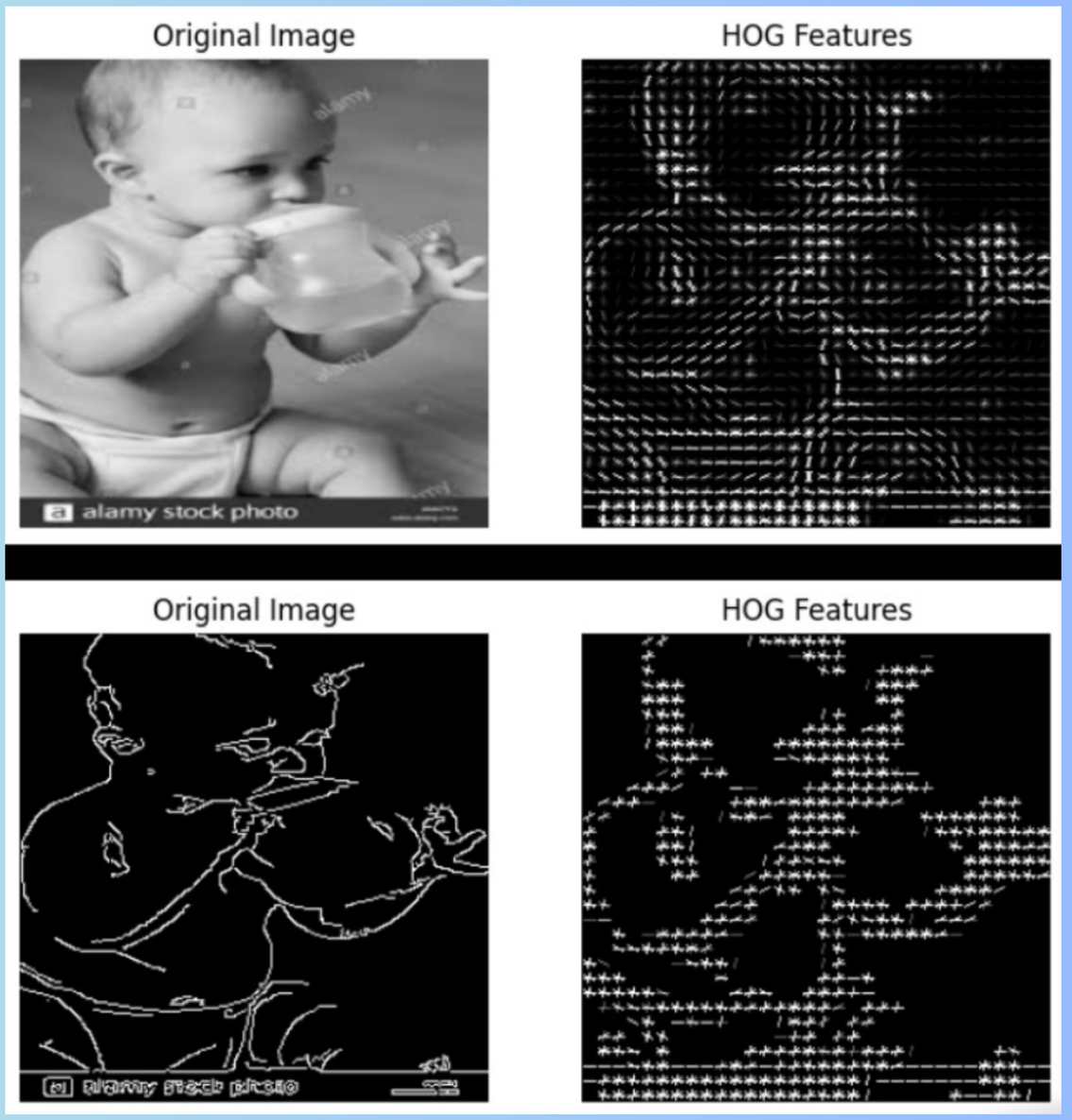
METHODOLOGY AND RESULTS -III

Results on Different Kernels Used (for the best model)

- Linear Kernel : 28%
- RBF Kernel (Gaussian) : 34%
- Polynomial Kernel (deg-6) : **35**%
- Sigmoid Kernel : 14%

Other Techniques Used

- Increased the No. of HOG features and used mean, var to reduce dimensionality (PCA not working as expected)
- Z-score Normalization



PROBLEMS

- **Noise in the dataset:** Traditional feature extraction methods involve handcrafted operations on the image. While these methods (filters and thresholds) may capture certain local features, they do not perform **noise removal in a learned and adaptive way** as they are manual. **Ties in the dataset added to the woes.**

- **Imperfect Feature extraction:** Handcrafted techniques like HOG, HSV, and LBP **have limited capacity to represent and generalize complex and high-dimensional patterns inherent in human activities**.

- Thus, traditional ML techniques and **Non-Deep learning-based** feature engineering do not lead to high accuracy in this task. **Existing analysis** also showed a majority of DL techniques (like OpenPose, CNNs, and RNNs being used to solve this problem. **CNNs** are capable of learning hierarchical representations of data and reducing noise inherently due to multiple layers.

Now we look at CNNs for improving accuracy

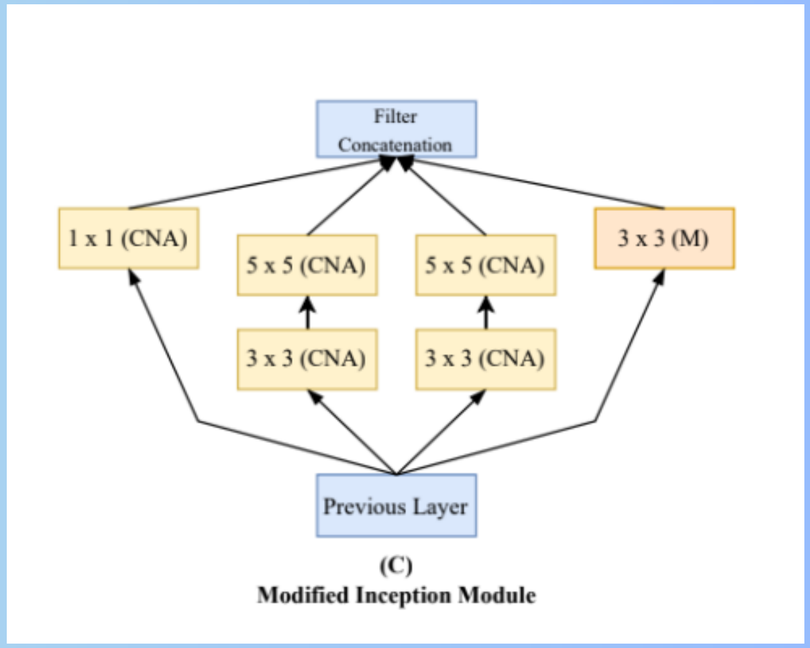
Deep Learning based Techniques

METHODOLOGY AND RESULTS

1) Custom CNN based architecture (using modified version of INCEPTION BLOCKS)

We used **6 such inception blocks** followed by a **global avg pooling** at the end and then a **fully connected layer for final classification.** After each inception block we did a **maxpool having stride=2** (for downsampling the image) . The no. of channels were doubled after each block starting from **32** and all the way upto **512.** The Activation function used was **ReLU** followed by a **Cross Entropy Loss**

- **Batch Size:** 32
- **No. of Epochs:** 40
- **LR:** 0.001
- **Image size:** 160x160
- **Optimizer:** Adam



(C)
Modified Inception Module

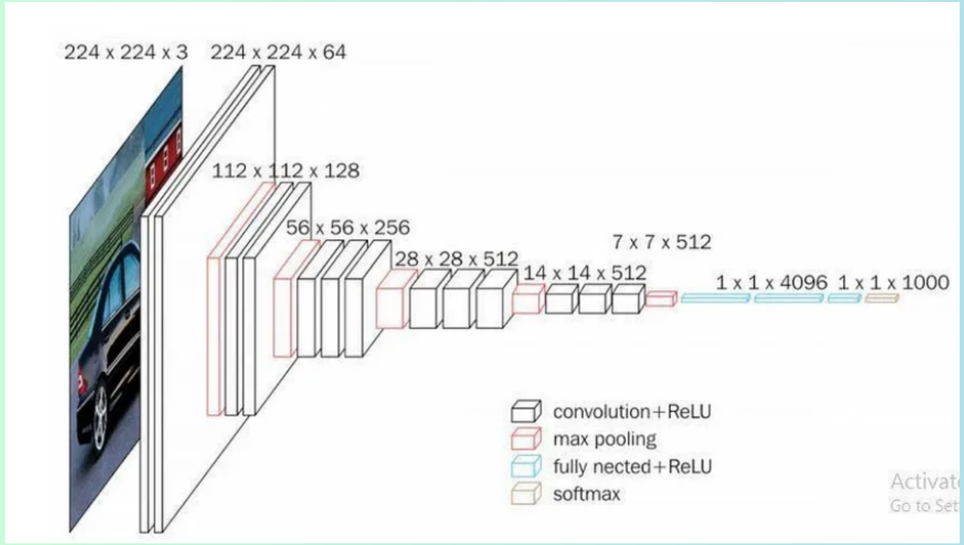
CNA: Convolution , Batch Normalization and Activation

Accuracy on test split: 56%

METHODOLOGY AND RESULTS

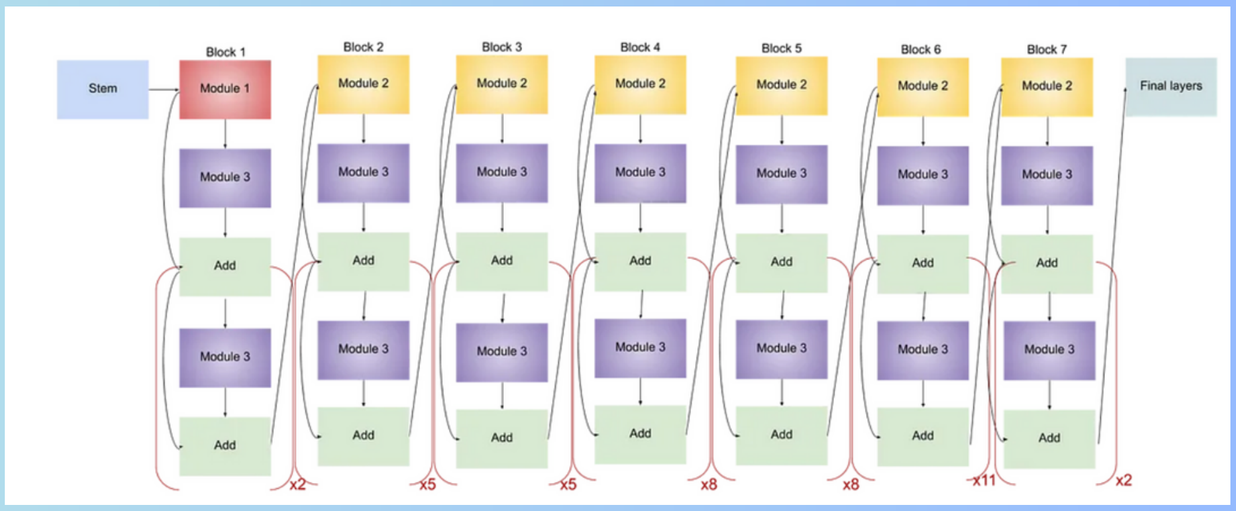
The following Pre-Trained models were fine tuned on the dataset by using a flattening layer and 2 Dense layers

2) Pre-Trained VGG -16



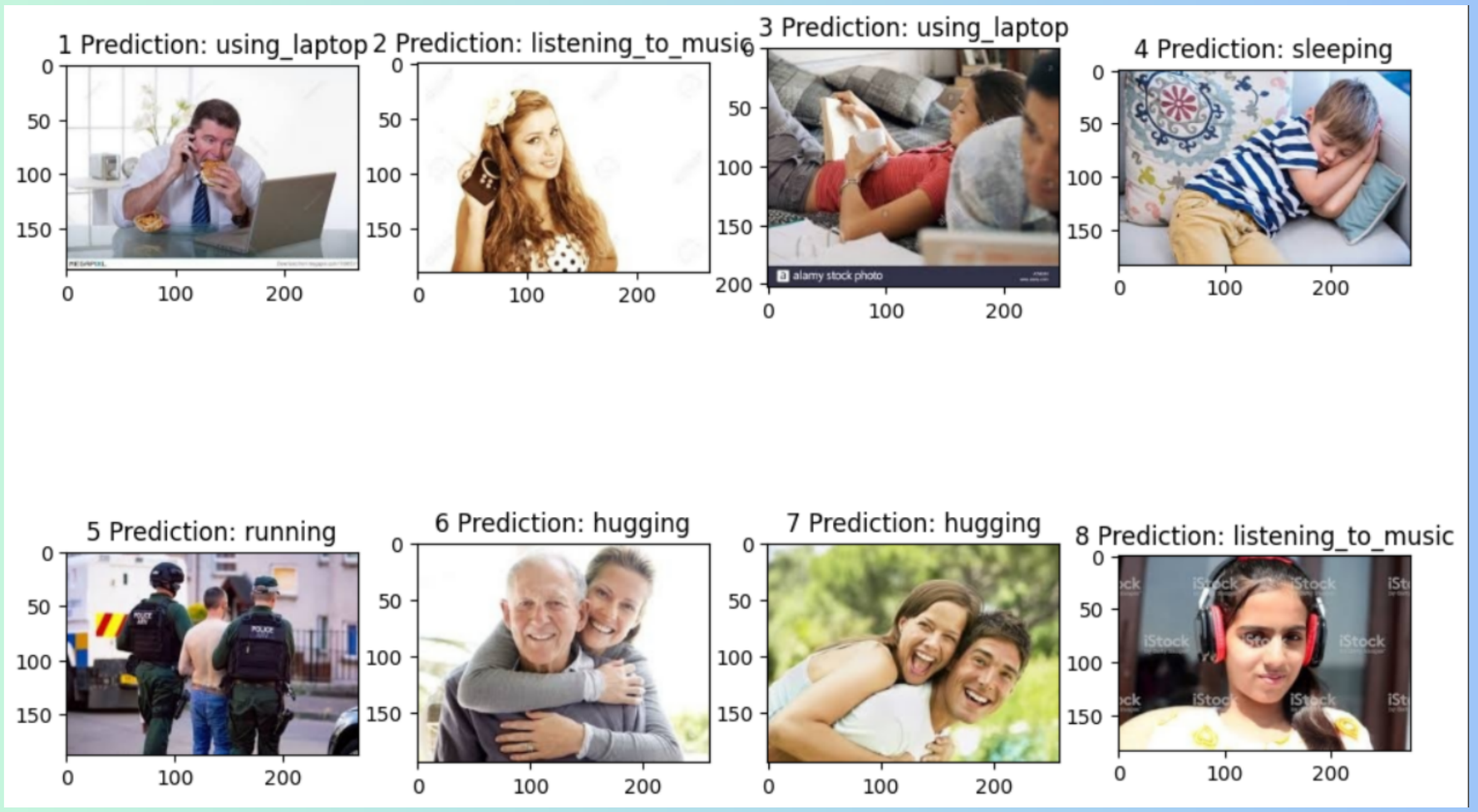
Accuracy on test split: 53%

3) Pre-Trained Efficient Net B7



Accuracy on val split ~ 70%

SOME FINAL PREDICTION RESULTS



THANK YOU !