Human Activity Recognition

(using kernelized SVM)

Group 28

PROBLEM STATEMENT

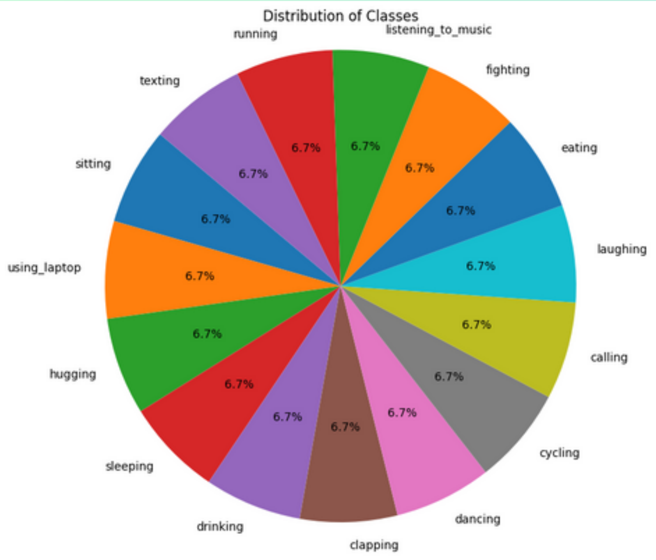
Human Activity Recognition is a critical area in the field of computer vision and machine learning, with wide-ranging applications from surveillance and security to healthcare and human-computer interaction.
The objective of this project is to develop a human activity recognition system using kernelized SVMs applied to an image dataset.

This project involves performing advanced EDA, preprocessing and feature extraction from the dataset consisting of 15 different classes of images, choosing the most suitable kernel for SVM and calculating the accuracy on the validation set to judge its performance.

EXPLORATORY DATA ANALYSIS

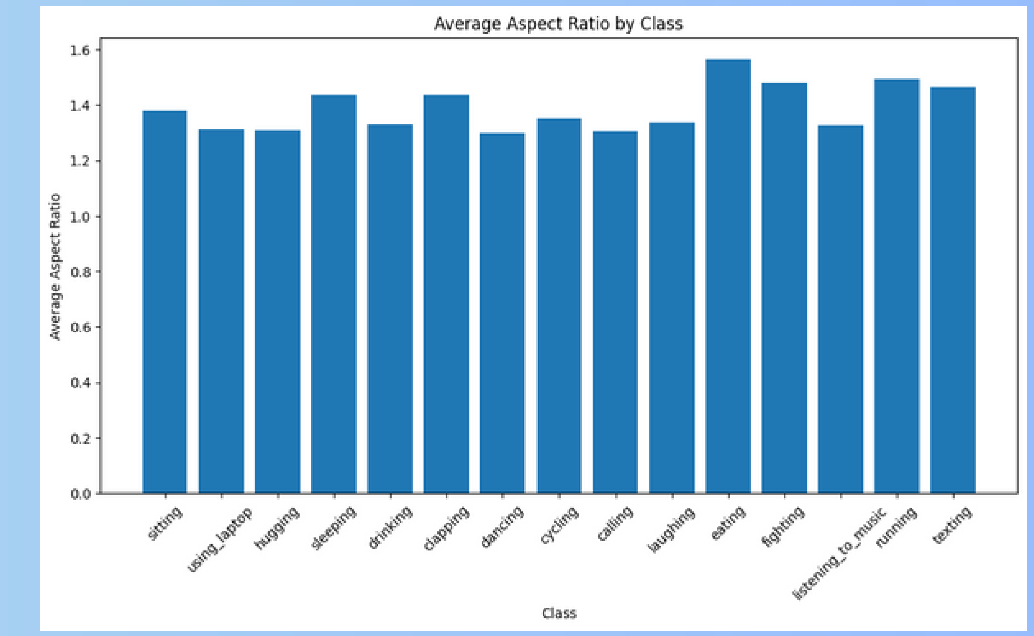
01 12600 Images in train set

840 per class x 15 classes



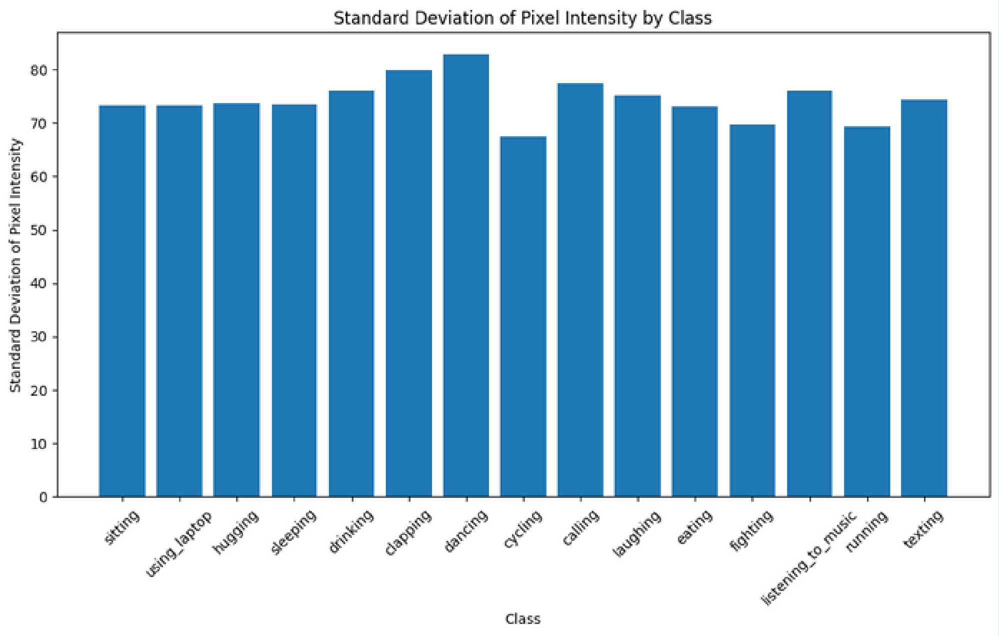
Average aspect ratio 02

Lie between 1.2 and 1.5 for all the classes



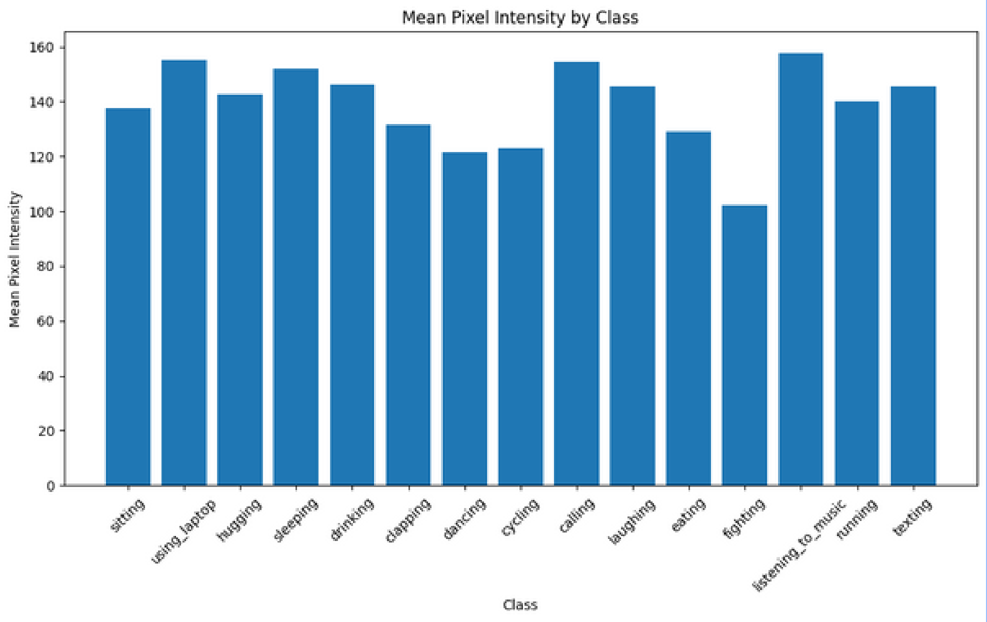
03 Standard Deviation of Pixel Intensity

Between 65 and 80



Mean Pixel Intenstiy 04

Between 100 and 150



METHODOLOGY

01

Resizing image

240x240 pixels

02

HOG

Apply Histogram of oriented gradients on our dataset

03

Canny Edge

Apply Canny edge detection

04

Feature concatenation

HOG and Canny features are combined

05

Standardization

Standardize the combined features

06

Dimensionality Reducation

Select 200 components using PCA

07

Test train splitting

Divide the training data into 70-30 components

08

Model training

OVR on SVMs with RBF kernel with probability estimates

09

Model testing

Testing the model on the test set

COMPARISON WITH EXISTING RESEARCH

Paper and Dataset	Methods/Algorithms Used	Results
Random Forest for Human Daily Activity Recognition	Random Forest (RF), ANN, KNN, LDA, Naïve Bayes, SVM	RF achieved the highest accuracy but was slower on large datasets. Accuracy: 87.16%
Human Activity Recognition Using K-Nearest Neighbor Algorithm	K-Nearest Neighbor (KNN)	Impressive testing accuracy of 90.46% with k=20 for smart factory activities.
Human Activity Recognition Using Gaussian Naïve Bayes Algorithm on sensor data	Naive Bayes(NB) and Gaussian Naïve Bayes (GNB)	Achieved an accuracy rate of 89.5%, outperforming traditional Naïve Bayes.

Our model achieved an approximate 37.5% accuracy on the validation set.

- Our approach leverages a fusion of HOG and Canny features, utilizing their strengths in capturing gradient-based and edge-based features, respectively.
- The task of Human Action Recognition is intricate, and achieving high accuracy requires fine-tuning models and feature engineering.
- On the same dataset, CNN gives the best accuracy of around 75%.

SVM (Support Vector Machine)

Support Vector Machine (SVM) is a powerful and versatile supervised machine learning algorithm used for classification and regression tasks. It works by finding the optimal hyperplane that best separates different classes in the feature space, maximizing the margin between data points of different classes.

KERNEL

Linear

Calculates the linear separation in the feature space.

Polynomial

Allows for non-linear boundaries by raising the data to a specified degree

Gaussian

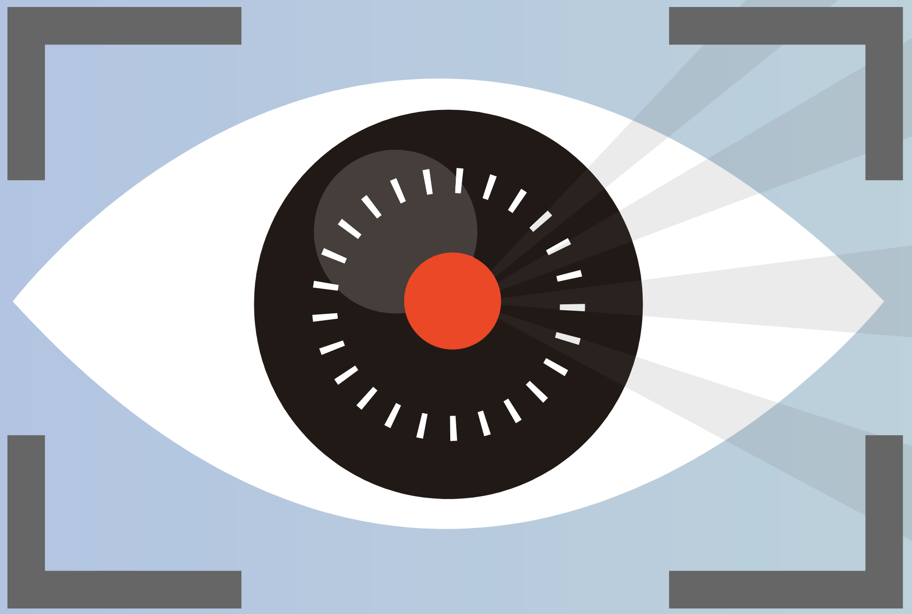
Measures the distance between points in a transformed feature space, handling complex patterns.

Laplacian

Handles non-linearity by focusing on locality in the feature space

CHALLENGES

Kernelized SVM



Overfitting

Although we achieved 99% accuracy in training data, accuracy in test data was below 40%.

Feature Engineering

Extracting meaningful and useful features from the images to prevent overfitting required a lot of research.

Parameter tuning

Choosing the right parameters for everything required a lot of experimentation.

Computation power

The lack of computing power on our side prevented us from increasing the parameters.

CONCLUSION

HOG

01

Histogram of oriented gradients is very useful in HAR through images and boosted accuracy singlehandedly.

Accuracy

02

The overall accuracy achieved is 37.5%.

Canny edge

03

Canny edge helped in detecting edges

KERNELIZED SVM

Pre processing

04

Image preprocessing like resizing, standardizing, augmentation play a crucial role.

PCA

05

PCA is important to constraint the number of parameters as each image can have a lot of parameters (pixels) and other features.

Feature Engineering

06

Since every image has many pixels, extracting meaningful features from the data is crucial to extracting maximum information using minimum features.