



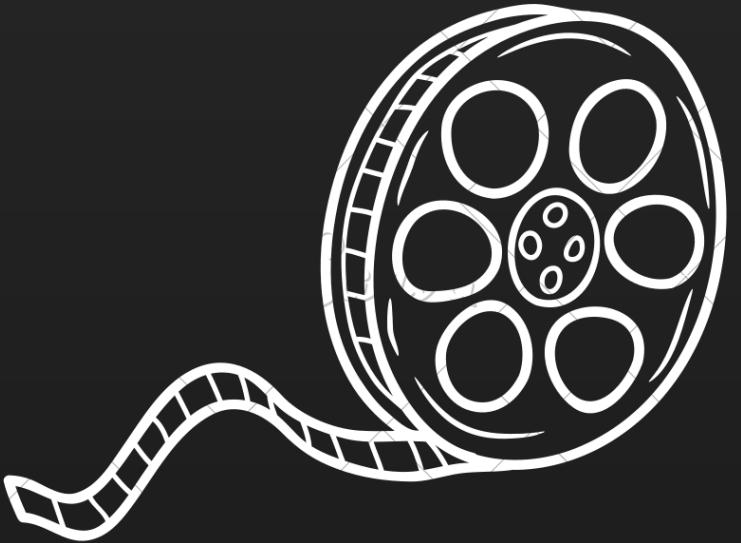
MOVIE RECOMMENDATION SYSTEM

USING EMOTION ANALYSIS

EMOFLIX

UNLEASH THE EMOTION, UNWIND THE REEL

PRESENTED BY:
ARUN SHANKAR S
GUIDE:SHERIN JOSEPH
DATE:13/08/24



INTRODUCTION

- Films have the power to captivate and inspire audiences.
- Watching films, in theaters or at home, brings people together.
- Cinema provides an outlet for stress relief, escapism, and emotional exploration.



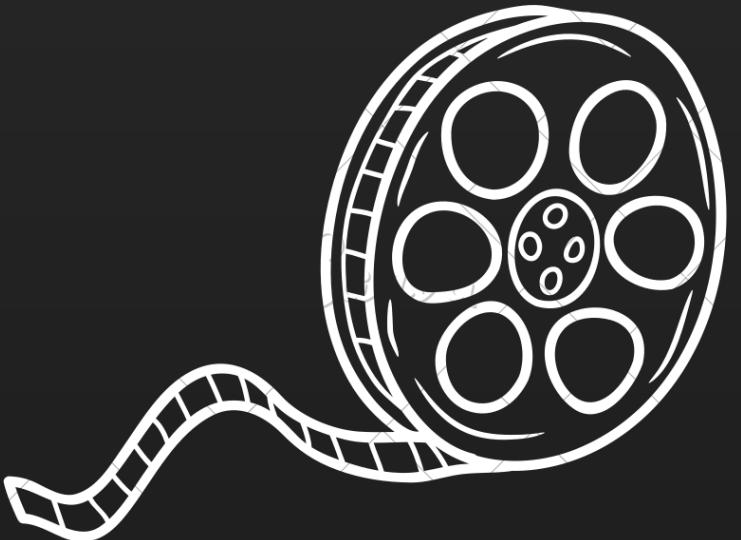


EXISTING SYSTEM CHALLENGES



Issues in Movie Selection,

- Mood Mismatch: Recommendations often ignore a user's current mood or situation.
- Overchoice: Too many options can lead to decision fatigue.



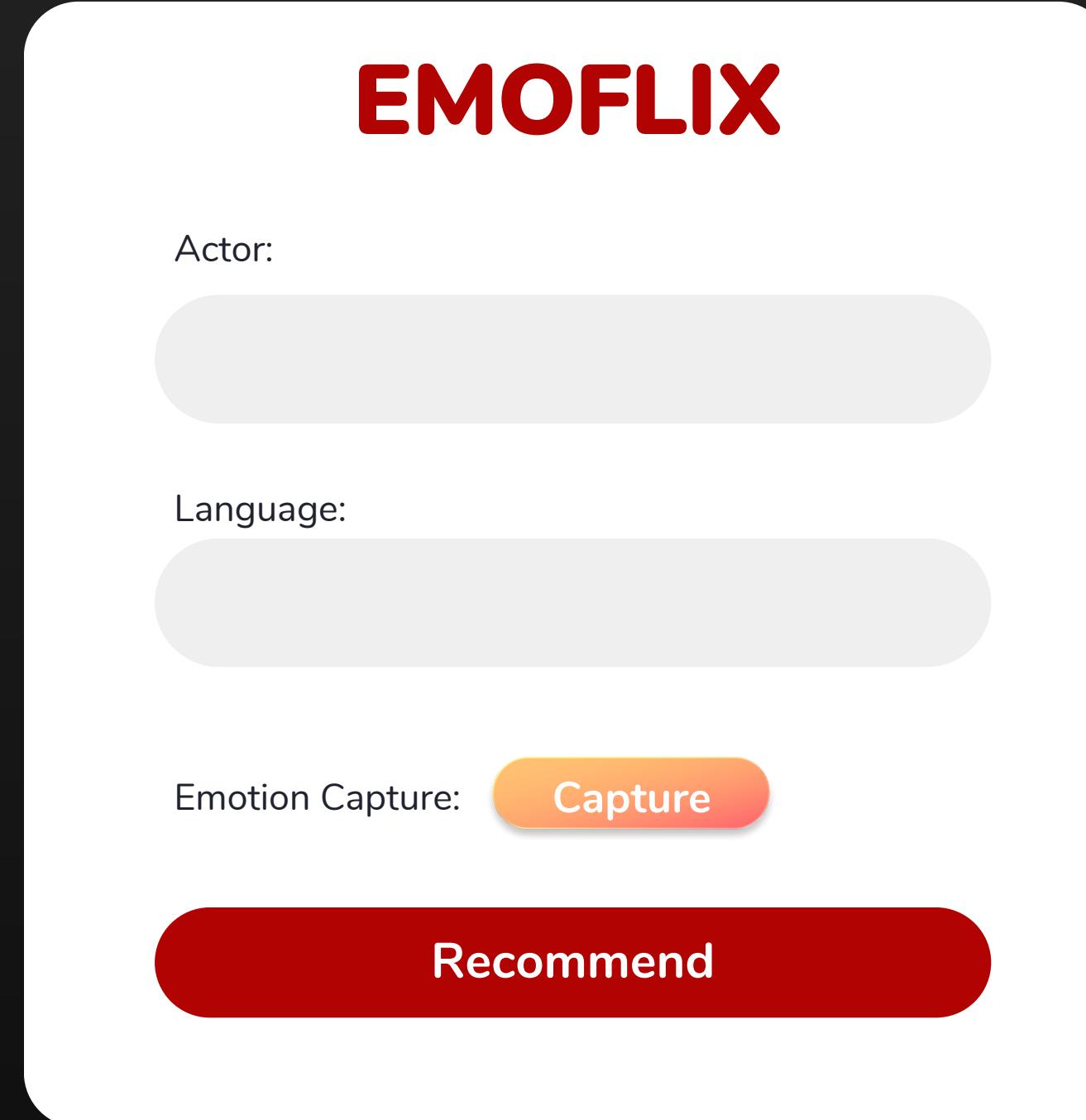
PROPOSED SYSTEM OVERVIEW

Movie Recommendations using Key Features includes,

Sentiment and Emotion Analysis: This project focuses on developing an innovative movie recommendation system designed to enhance the user experience by integrating sentiment (rating) and emotion analysis (emotion capture).

- Facial Recognition(CNN)
- Emotion Recognition
- Sentiment Analysis.
- Content-Based Filtering





01 Enter Language and Actor Preference.

02 Capture your Emotions

03 Click recommend

04 Get suggestions of movies according to your emotion.

TECHNOLOGIES USED

01

Python

The core programming language used to build and integrate different components of the project.

02

Streamlit

Creates a user-friendly interface for collecting user preferences and deals with ML aspects,

03

Pandas

Manages and processes datasets, enabling easy manipulation and analysis of the collected user inputs.

04

Matplotlib

Visualizes data trends, such as emotion detection accuracy during model training.

TECHNOLOGIES USED

05

Convolutional Neural Networks (CNNs):
detects emotions from the real-time video feed by processing the input images.

06

TensorFlow
Provides the backend framework for training and deploying the deep learning model for emotion detection.

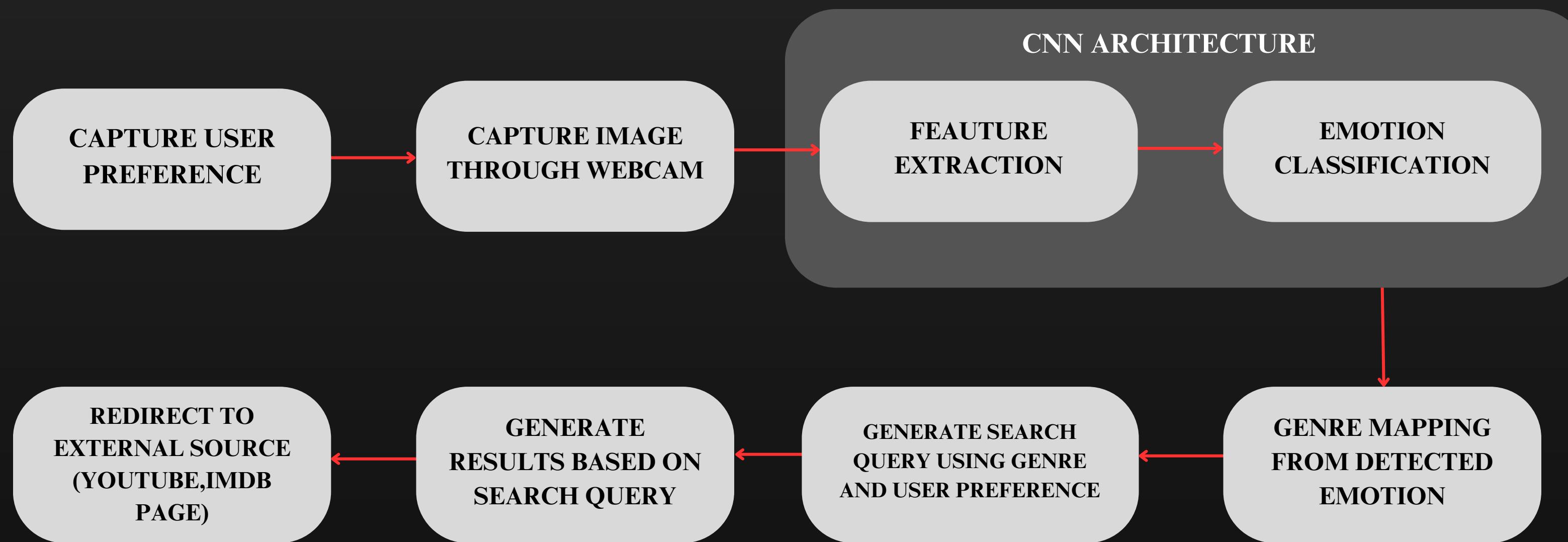
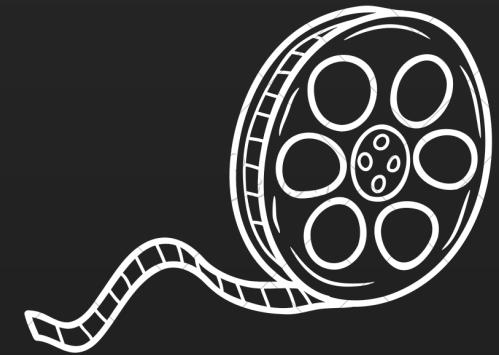
07

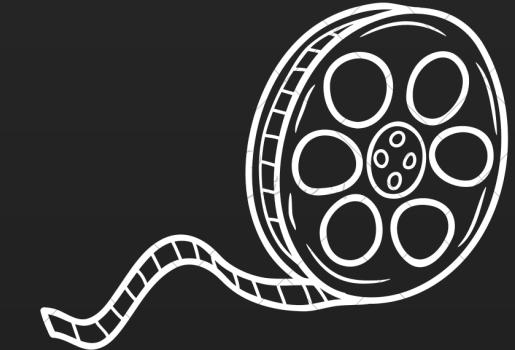
Keras
Simplifies the creation, training, and evaluation of the CNN model.

07

Numpy
Handles numerical operations and array manipulations for image processing and data handling in the project.

WORKFLOW





DESIGN MODULES

PROGRAM DESIGN

- Deep Learning Module
- User Interface Module
- Genre Mapping Module

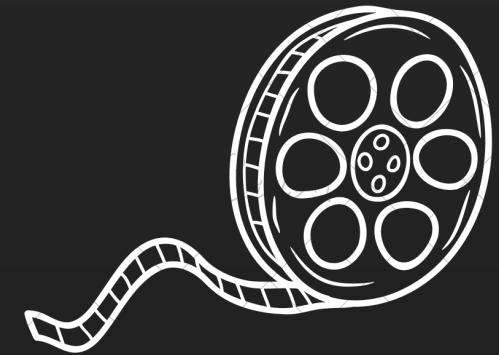
INPUT DESIGN

- Capture real-time video to detect the user's emotion.
- Text input to select or enter preferred actor and language.

OUTPUT DESIGN

- Search Query Display:
- Search Results Display

DATA PREPARATION



FER 2013 DATSET

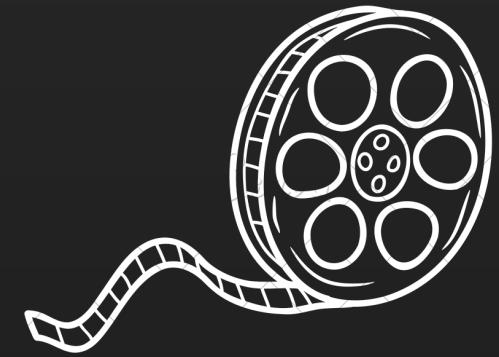
DATA VISUALIZATION



TRAINING IMAGES: 28,000
TEST IMAGES: 7,000



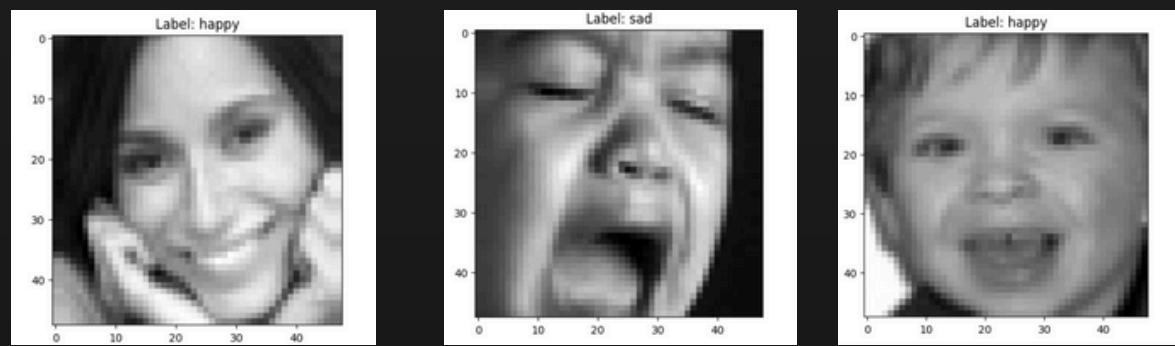
DATA PREPARATION



REMOVING DUPLICATES

```
1.jpg
section\data\train\surprise\Training_9839279.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_9851064.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_9853966.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_98737557.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_98747349.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_98767792.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_98774734.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_98828949.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_9890261.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_9909756.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_991508.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_99182527.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_99267647.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
section\data\train\surprise\Training_993811.jpg is a duplicate of D:\movie-recommendation-syst
.jpg
```

NORMALIZING IMAGES



```
train_datagen = ImageDataGenerator(
    rescale = 1./255,    ## rescale or normalize the images pixels, by dividing them 255
    shear_range = 0.2,   ## angle for shear of image in degrees
    zoom_range = 0.1,   ## for zoom in or out
    horizontal_flip = True
)
X_train = train_datagen.flow_from_directory(
    'D:\\movie-recommendation-system\\emotion_detection\\data\\train',    ## give path of training set
    target_size=(48,48),      ## target_size of image in which you want
    batch_size=32,
    color_mode = "grayscale",
    class_mode = "categorical"
)
Found 27472 images belonging to 7 classes.

test_datagen = ImageDataGenerator(rescale=1./255)
X_test = test_datagen.flow_from_directory(
    'D:\\movie-recommendation-system\\emotion_detection\\data\\test',
    target_size = (48,48),
    batch_size = 32,
    color_mode = "grayscale",
    class_mode = "categorical"
)
Found 7178 images belonging to 7 classes.
```

DATA MODELING

ANN STRUCTURE

01

Convolutional Layers:

Four layers with ReLU activations and filter sizes (32, 256, 128) for spatial feature learning.

02

MaxPooling2D:

Downsampling after each convolutional block.

03

Dropout Layers:

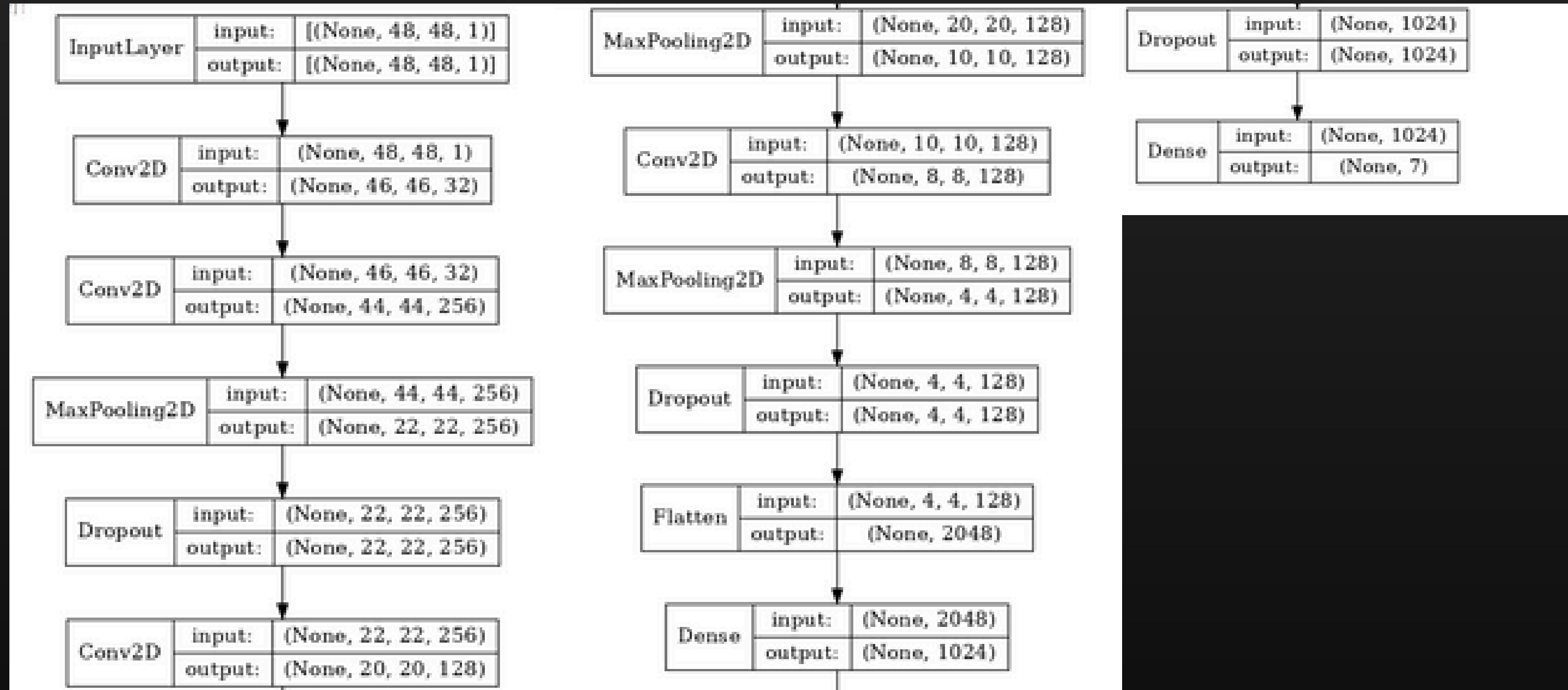
Reduce overfitting (25% and 50% rates)

04

Dense Layers:

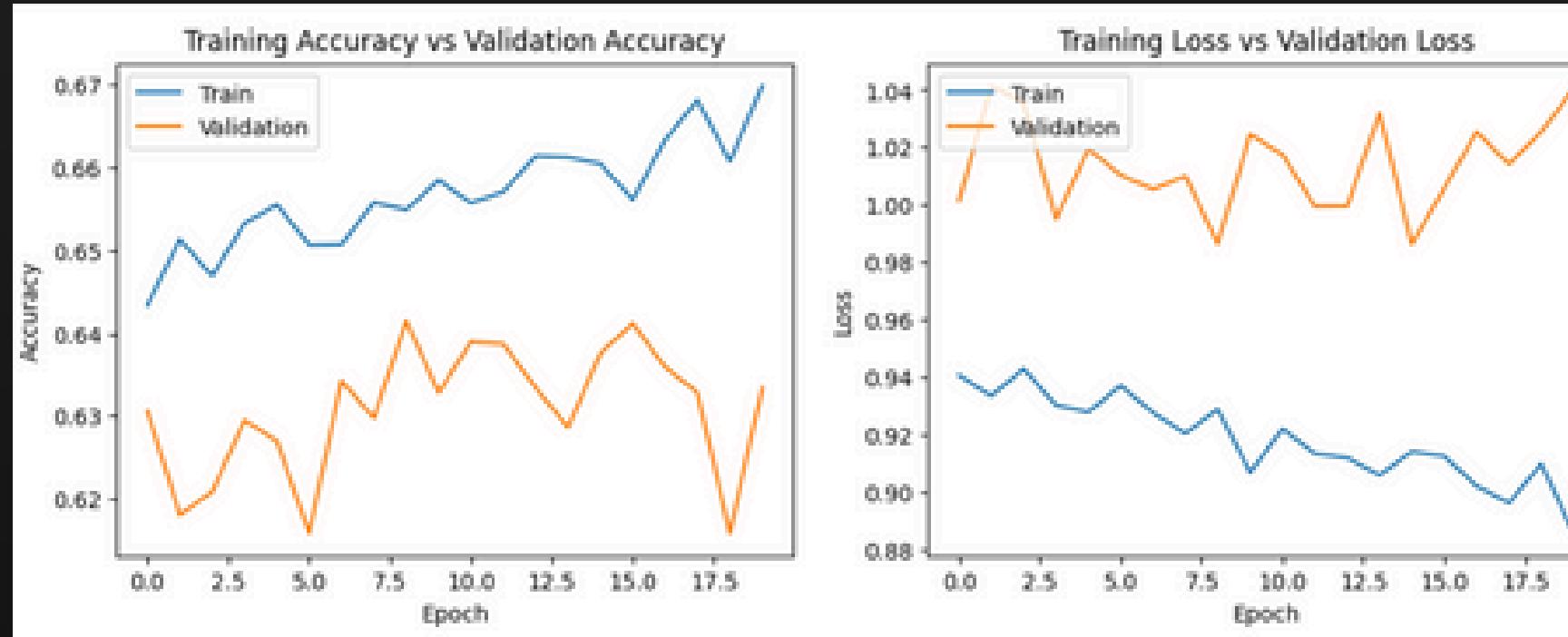
1024 units (ReLU) and output layer (7 units, softmax).

ARCHITECTURE DESIGN



MODEL EVALUATION

LEARNING CURVE



FINAL TRAIN ACCURACY = 72.06 %

VALIDATION ACCURACY = 63.60%

CONFUSION MATRIX

Confusion Matrix								
True labels	angry	disgust	fear	happy	neutral	sad	surprise	
	angry	141	14	91	225	237	136	114
	disgust	23	1	11	24	26	13	13
	fear	118	8	111	259	246	139	143
	happy	218	13	175	441	441	236	250
	neutral	165	8	138	341	275	173	133
	sad	164	7	123	341	313	154	145
	surprise	100	3	90	222	208	102	106

K - F O L D C R O S S V A L I D A T I O N

- Inorder to improve Accuracy,K-Fold Cross Validation is used.
- K-fold cross-validation splits the dataset into k subsets, trains the model on k-1 subsets, tests on the remaining one, and repeats this process k times to evaluate model performance.

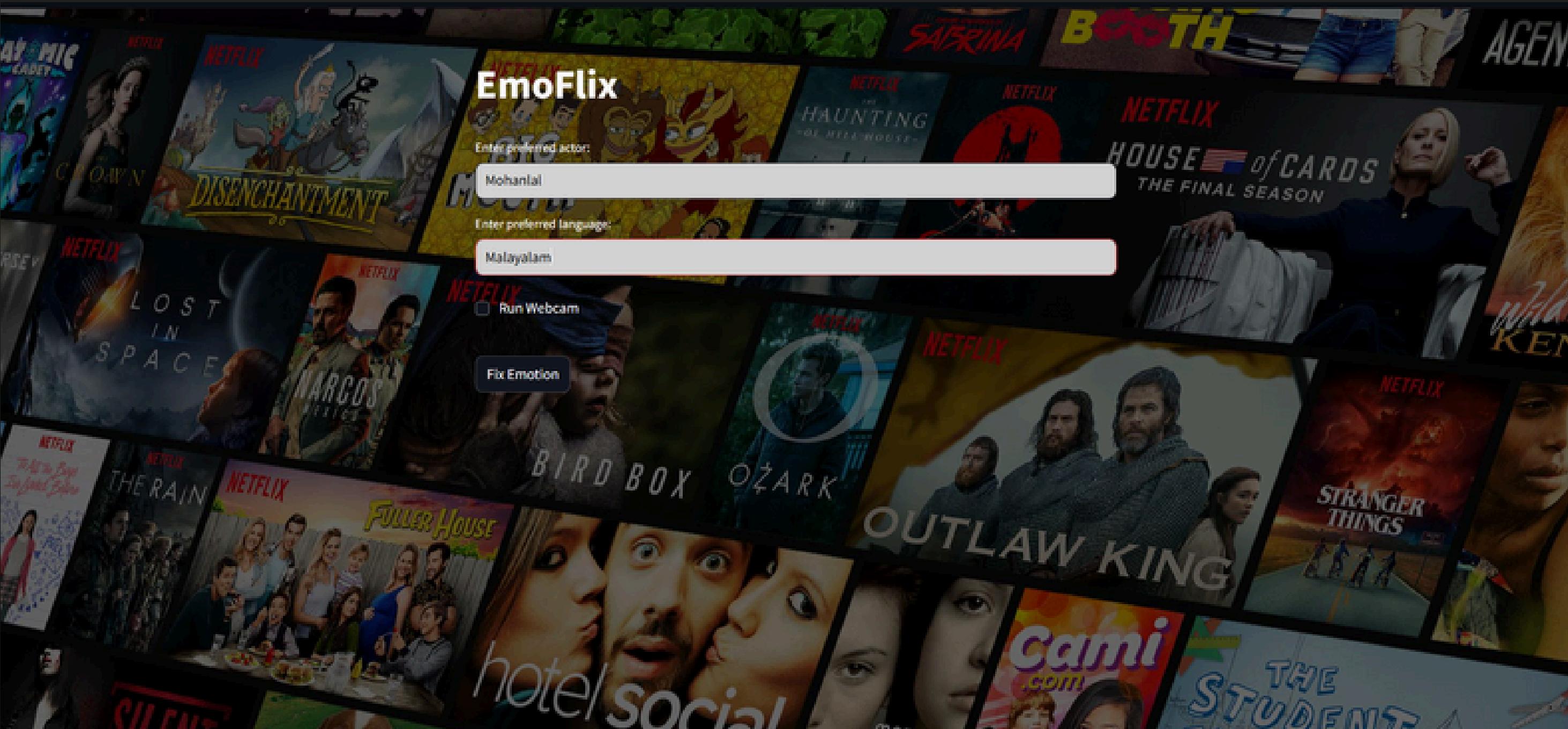
```
# Define or load your model
if fold == 0:
    # Load a pre-trained model (if it's the first fold)
    model = load_model('best_model_fold_0.h5') # Ensure this model exists before running

else:
    # Load the model from the previous fold
    model = load_model('best_model_fold_{}.h5'.format(fold - 1))
```

FINAL TRAIN ACCURACY = 88.10 %

VALIDATION ACCURACY = 74.34%

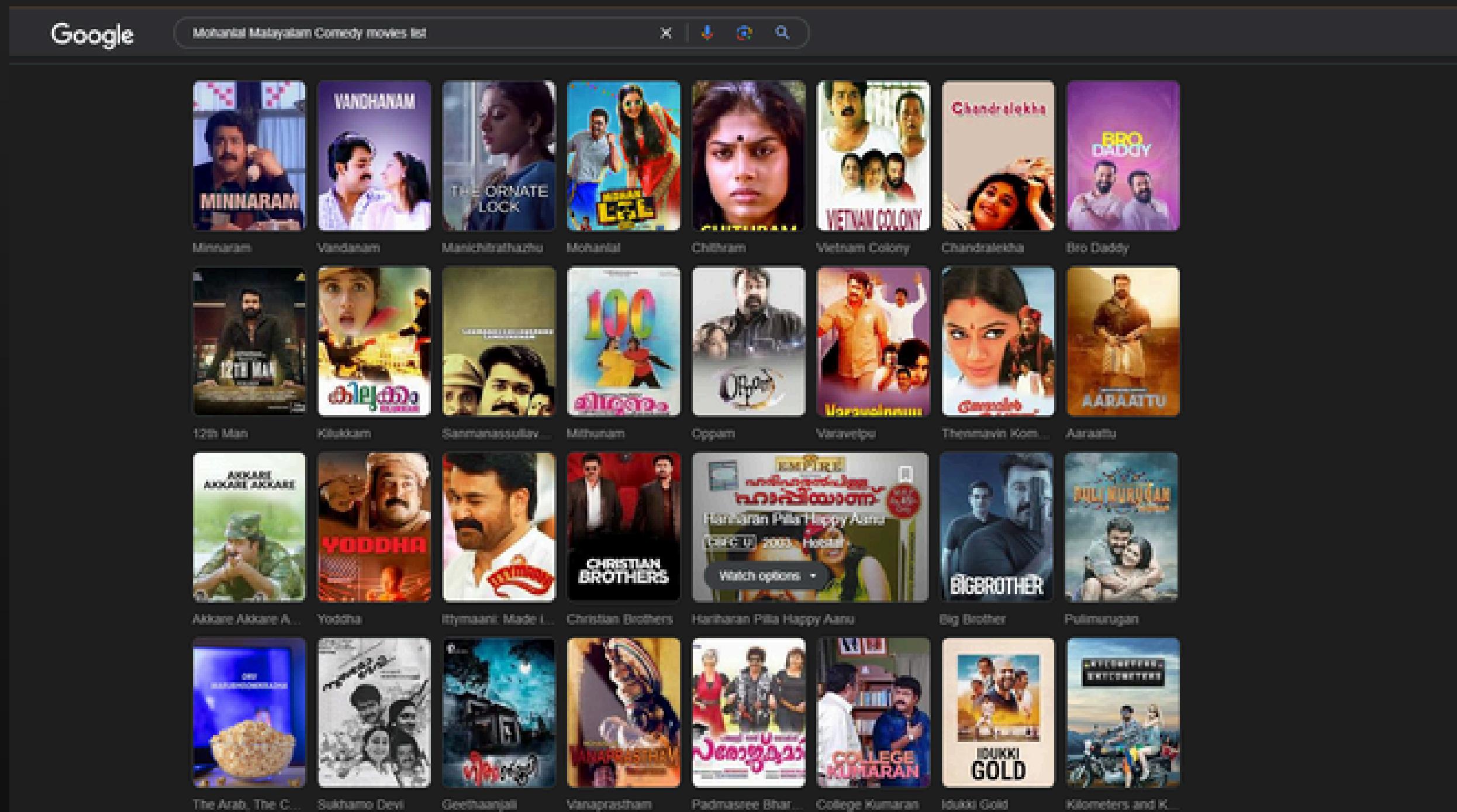
SCREENSHOTS



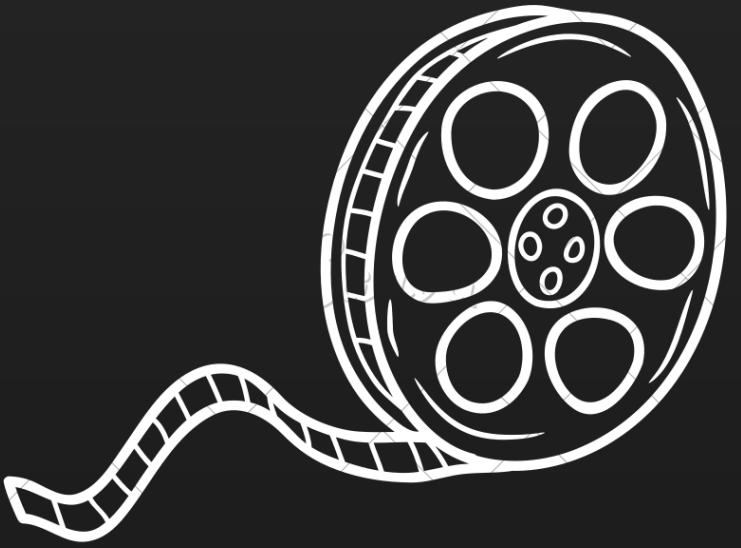
SCREENSHOTS



SCREENSHOTS



GOOGLE SEARCH PAGE



CONCLUSION

- A unique application that combines emotion detection with personalized movie recommendations.
- Simple and intuitive interface for users to input their preferences (actor and language).
- Real-time emotion detection enhances the personalization of movie suggestions.



THANK YOU