From Facial Expressions to Personalized Learning: Unveiling the Opportunities and Challenges of SVM-Based E-Learning Systems

Fatima Waseem
Software Engineering Department, Capital University of Sciences and Technology, Islamabad, Pakistan
Email: fatima.waseem@cust.edu.pk

Muhammad Saad
Department of Software Engineering, Sir Syed University of Engineering and Technology, Karachi, Pakistan
Email: msaad@ssuet.edu.pk

Muhammad Kashif Shaikh
Department of Software Engineering, Sir Syed University of Engineering and Technology, Karachi, Pakistan
Email: mkshaikh@gmail.com

ABSTRACT

Facial expression recognition (FER) using Support Vector Machines (SVMs) offers intriguing possibilities for distance education and e-learning. Imagine gauging student engagement through expressions like boredom or attentiveness, providing emotional support based on frustration, or even personalizing lectures based on individual needs. Here's where SVMs come in their effectiveness in image recognition and real-time processing make them suitable for analyzing video streams in e-learning platforms. However, challenges like privacy concerns, accuracy limitations, and technical hurdles require careful consideration. While ethically implemented FER systems could lead to improved engagement, personalized learning, and early intervention, further research is necessary to address these limitations and ensure responsible use of such technology. This research paper presents the Human Facial Expression Recognition based on Supervised Learning model (SVM) using a popular dataset Ck+. Our motive was getting best accuracy by using SVM Model with Ck+ dataset and with Fer2013 dataset and eventually succeeded but we also tried combination of Ck+ and Fer2013 dataset with SVM model for experiment. These models provided 50.53% accuracy with Fer2013, 72.61% with combination of Fer2013 and CK+ & 90.96% by using CK+ dataset, providing the most accurate results. Facial expression is an important medium of non-verbal communication. It is a rapidly growing field of research in the domain of Computer Vision and Artificial Intelligence. Since SVM performs better than other existing techniques that are used for facial recognition, therefore improves the overall efficiency of the facial expression recognition.

Keywords-- Support Vector Machine, Support Vector Classification, Facial Expression Recognition System, Fer2013, Ck+

INTRODUCTION

In the ever-evolving landscape of education, distance learning and e-learning platforms have become increasingly popular, offering flexibility and accessibility to a wider range of learners. However, according to (Xia, Ziqing et al.,2024) these virtual environments lack the immediacy and nuanced communication channels present in traditional face-to-face settings. This is where facial expression recognition (FER) emerges as a potential game-changer according to (Zhigang Yu, Yunyun Done et al.,2022). By harnessing the power of Support Vector Machines (SVMs), FER systems can analyze facial expressions within online learning platforms, unlocking a deeper understanding of students' emotional states and engagement levels. This introductory paper delves into the exciting possibilities of integrating SVM-based FER into distance education and e-learning, exploring its potential to enhance student engagement hightlighted by (Bellaj, Mohamed, et al.,2024), personalize learning experiences, and provide timely support. However, alongside these promising advancements, ethical considerations regarding privacy and potential accuracy limitations remain crucial aspects to address. With careful consideration and responsible implementation, FER systems hold the potential to revolutionize the way we learn and interact in virtual education environments. Preprocessing techniques like normalization or scaling can improve model performance by ensuring features are on a similar scale.

According to (I.Michael Revina, W.R. Sam Emmanuel, et.al., 2021) as we are living in the 21st century that is the Digital age, where everything is shifting towards automation. Our body gestures and emotions play an integral role in shifting the procedure towards automation. Human emotions have contributed to the field of Machine Learning and Artificial Intelligence and brought phenomenal changes. As a result, today's systems have become much smarter that if certain modules are integrated to the system, our system can tell what the person is thinking or feeling. Human emotional states tell how we behave for the most basic processes and difficult decisions. Our emotions guide our lives in many ways and that is visible through our facial expressions. It's understandable that human's emotional state can be used for numerous applications including better understanding of human psychology and investigating human behavior for verification process etc. Continued research has been done to enable computers to recognize facial expressions and use the embedded information in the human- machine interfaces.

According to (Pramanik, Bablu, and Sneha Singh, 2024) and (Priya, M., et al, 2024) different types of algorithms are used in Facial expression recognition for face detection and emotional state recognition by studying the faces in videos or images that are taken from embedded cameras in different devices. Facial expression recognition model can be applied to a system to assess whether a person is confident enough and fit for the job by analyzing his emotions during an interview. A human face normally produces six universal expressions and facial expression recognition uses biometric markers that helps in emotion detection in human faces.

Our project is an online donation system, where the donor and the receiver interact on a platform. The receiver uploads an ad, that is the basically an application for the donation. That application is put on hold until the user is thoroughly verified through several verification processes. One also includes an Online Interview where the user will be asked questions and will be evaluated accordingly. Our motive is to switch the process towards automation, that the system analyzes and generate outputs by itself. For that, the facial expression recognition technology is used, which by using Support Vector Machine (SVM) will identify the expressions and emotional state of the user with while giving interview. The result of the facial expression evaluation will help with the application of the user.

This research paper presents the Human Facial Expression Recognition based on Supervised Learning model (SVM) using a popular dataset Ck+. Our motive was getting best accuracy by using SVM Model with Ck+ dataset and with Fer2013 dataset and eventually succeeded but we also tried combination of Ck+ and Fer2013 dataset with SVM model for experiment. These models provided 50.53% accuracy with Fer2013, 72.61% with combination of Fer2013 and CK+ & 90.96% by using CK+ dataset, providing the most accurate results.

LITERATURE REVIEW

Facial expressions recognition has always been a complicated challenge when analyzing emotions. Number of Artificial intelligence and Machine Learning models and techniques and studies have been constructed and presented by different authors. Abdulazeez et al., (2024) and Seifu, (2022) suggested using Viola-Jones algorithm that recognizes faces. After performing image preprocessing, the features will be extracted from the facial images using the PCA approach. Then the CNN and SVM classifications are used to classify the features as happy, sad, fear, disgust etc. Yu et al., (2022) proposed a new technology for facial recognition. They proposed GoogLeNet-M network, that improves the performance of network based on streamlining the network. Secondly, they have added regularization and migration learning methods in order to improve accuracy. Their experimental results provide a recall rate of 97% and are 98% accurate.

Revina et.al., (2021) compared various FER techniques and its major contributions. Different FER techniques performances is compared on the basis of number of expressions recognized and complexity of algorithm. They compared different databases like JAFFE and CK. Using Line -based caricatures, (Cossetin, M.J. et al., 2016) suggested a static image technique for facial expressions recognition where the approach uses the geometrical and structural features of a sketched model that is matched with Line Edge Map (LEM). This approach has been assessed and has provided promising results.

Salmam, el al., (2016) used is the Gabor filter-based feature expression with the help of Learning Vector Quantization (LVQ). This technique recognizes seven different facial features form human face. It extracts features from still images, and it came into observation that LVQ better recognizes the fear expression than MLP classification. Rashid, et.al., (2016) used the LBP-TOP features and described facial expressions AdaBoost is used for the selection of the most essential features related to expressions and final recognition is done using SVM classifier. Vasanth and Nataraj, (2015) discussed the techniques to recognize the facial expressions in three levels. They used the Gabor Filter to initially extract the features of eye and mouth. Then LBP and PCA are used for reducing the dimensions of the features. Finally, the SVM model is used for classifying the expression and facial action units.

Poursaberi et al.,2012) used image ratio features that are combined with Facial Animation Parameters (FAPs) to improve the accuracy of facial expression recognition. This work primarily is done on video-based facial expression recognition. Song, D. Tao et al. (2010) used ‘salient’ distance features that are retrieved from patch-based 3D Gabor features. Its result displays high contrast recognition rate (CRR) and the performance are significantly enhanced considering muscle movement and facial elements. This approach achieves the highest CRR on JAFFE database and is one of the top performers on Cohn-Kanade (CK) database. Wang, H. et al. (2010) used Gauss-Laguerre wavelets that are highly capable in frequency extraction. Three public databases i.e., Cohn-Kanade, MMI Image and the JAFFE have validated the performance of this system.

Zhao and Pietikäinen (2009) used novel approach that is the combination of various features subsets and specialized pairwise classifiers. It considers all pairs of classes, and the dimensionality is reduced using hybrid selection strategy. Few classifiers are then trained with such pairwise subset. This system has gone through several experiments on CK, JAFFE and TFEID datasets and provided 98.07%, 99.05% and 99.63% accuracy, respectively. Bashyal, Venayagamoorthy used a model based on a geometric approach, where six distances are calculated. In addition, a decision tree is used on two databases i.e., JAFFE and COHEN. After experimenting, the results achieved 89.20% and 90.61% accuracy on JAFFE and COHEN databases respectively. Gao, M.K.H. Leung et al.,(2015) used three-staged SVM, where first stage is the binary combination for the seven expressions and is also named 21 SVMs. The K-fold test and leave-one-out validation methods were used. The experiments were carried out on JAFFE, Extended CK and Radboud databases. Afeefa Muhammed1, Ramsi Mol2 et al., (2020) proposed a method based on SVM. It also adopts CNN for image training. The SVM follows the kernel technique for data transformation. SVM performance is better than other techniques and improves the overall performance of Facial Expression Recognition system.

The growing ubiquity of facial recognition in daily life, from phone unlocking to payment systems, necessitates addressing the inherent privacy risks. Facial data, unlike passwords, is unique and unchangeable, making leaks catastrophic. This paper (Feng, Luoyin, and Xin Chen (2022)) proposes a solution: an encryption and recognition method utilizing chaotic maps to generate complex keys for image encryption. Further, a neural network ensures accurate face recognition even after encryption. The method's robustness against various attacks is thoroughly tested, demonstrating its potential to provide both convenience and strong privacy protection for facial recognition technologies.

Emotion recognition often uses singular signals like voice or video, working well in certain situations. However, as applications diversify and data grows, single-modal methods struggle to achieve high accuracy and comprehensiveness. This paper proposes using "multimodal thought" to improve accuracy. Preprocessed data from audio and video modalities are fed into specific models: "time-distributed CNNs + LSTMs" for audio and "DeepID V3 + Xception architecture" for video. These models are tested and compared to existing algorithms, demonstrating their effectiveness. Finally, a late-fusion method based on weight adaptation is proposed and implemented, combining the outputs of both models. This multimodal fusion approach significantly outperforms single-modal methods, achieving an accuracy of 84.33%, a 4% improvement (Cheng, Yongjian, et al., 2023).

This research (Majeed, Fahad, et al., 2022) explores using deep learning, specifically the YOLOv5 model, for real-time multi-face detection in surveillance systems. The challenge lies in handling diverse and difficult environmental conditions. Training on public datasets (FDDB and CFR) and a private video stream, the system achieved impressive accuracy: 93% on FDDB and 99% on the private data. This surpasses previous YOLO versions (YOLOv3 and YOLOv4) and demonstrates the effectiveness of the approach. Further validation with real-time video streams confirms the system's ability to accurately recognize multiple faces, making it a promising solution for surveillance applications.

As real-time facial recognition gains traction, researchers are working on faster and more accurate systems. This paper by Kavita, and Rajender Singh Chhillar(2023) proposes a CNN and Python-based system focused on improved face detection for applications like authentication. After analyzing various machine learning algorithms (Decision Tree, Naïve Bayes, KNN, and CNN), the authors build a system using OpenCV and Python on a celebrity face dataset. The approach involves training on 80% of the data and testing on 20%, achieving an accuracy of 89.36%. This accuracy, along with improved recall, f-score, and precision values, demonstrates the effectiveness of the proposed system compared to other machine learning techniques.

SUPPORT VECTOR MACHINE

Support Vector Machine lies in the category of Supervised Machine Learning. We are able to resolve Classification and Regression problems. Mostly, we do classification compared to regression. For classifying two-groups, we use classification algorithms of SVM. After training of model with some data, we are able to recognize or conclude new data. Here, the example below is relevant to the topic of the research paper.

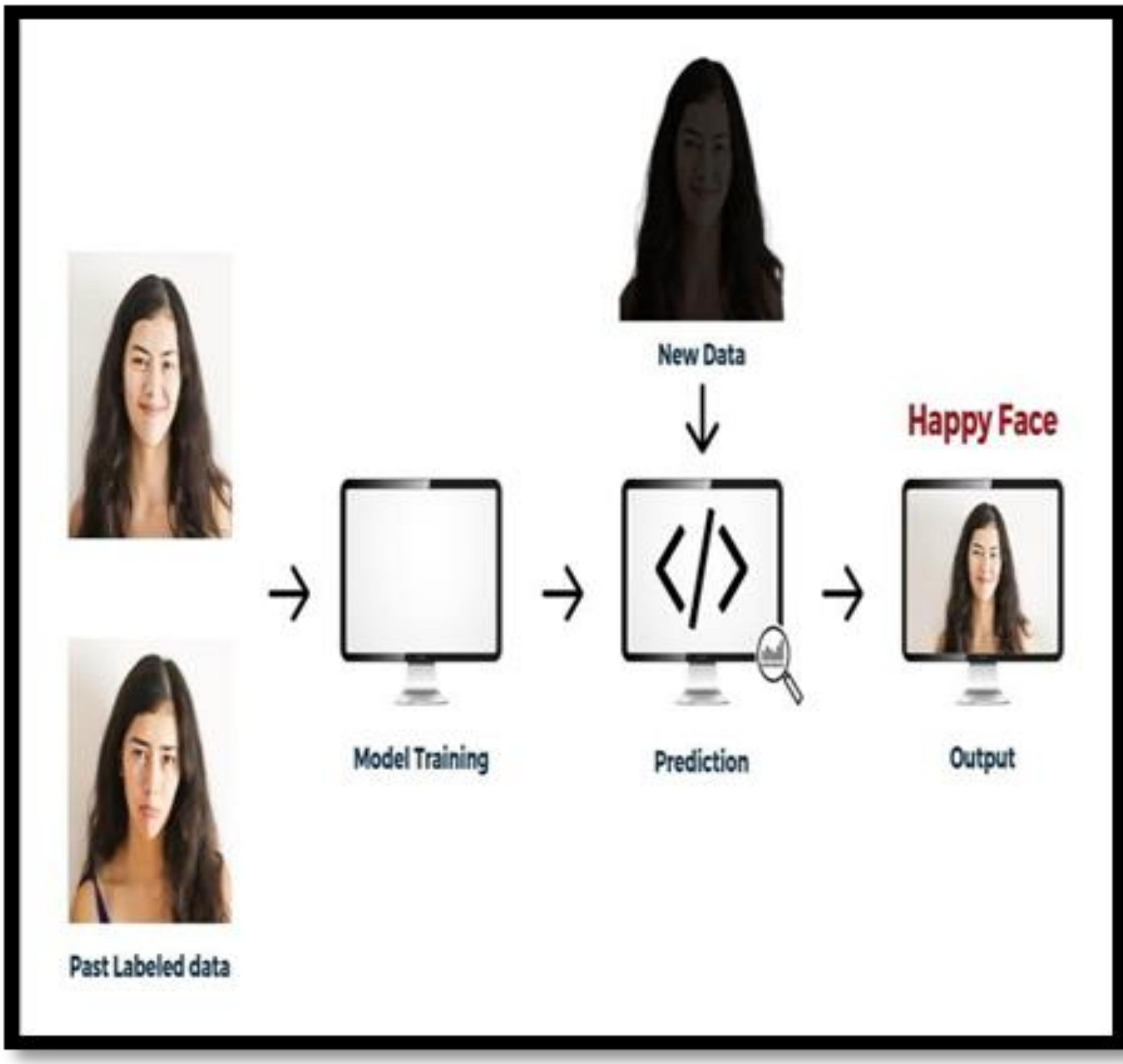


Figure 1: Facial Recognition using SVM

Figure 1 shows that, we have trained data and trained model and then we passed the new data to detect what kind of data is passed through. In simple words, in our dataset, we have data of sad and happy faces. We entered the new happy face and then our model identified the kind of data.

If we compare SVM with latest algorithms like Neural Network, SVM provides higher speed and better performance with the limited amount of data i.e. in thousands. That's why this algorithm is suitable for classification problems. In application, when the system wants to recognize hand writing, face detection, email and gene detection then SVM is the best option for doing this type of work.

WORKING OF SUPPORT VECTOR MACHINE

The working and the basics of Support Vector machine can be best recognized with an example. Suppose there are two flags; a green one and a yellow one, and there are two features in our data; x and y. A classifier is required that can classify, if pair of (x, y) coordinates are given, provide the output if whether it is green or yellow. Figure 2 shows the labeled training data is plotted on a plane:

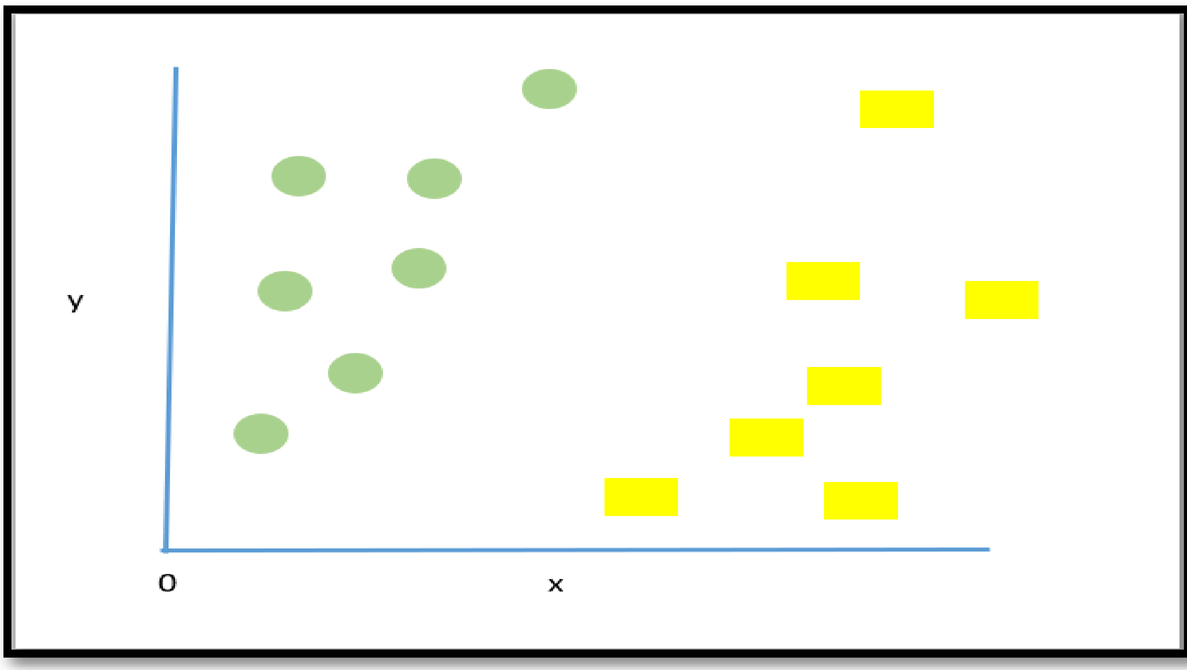


Figure 2: Our Labeled Data

SVM gathers the data points and displays a hyperplane that separates the tags the best. It is called a Decision Boundary. Anything that lies to one side of the line will be considered as green and line anything lying to the other will be classified as yellow.

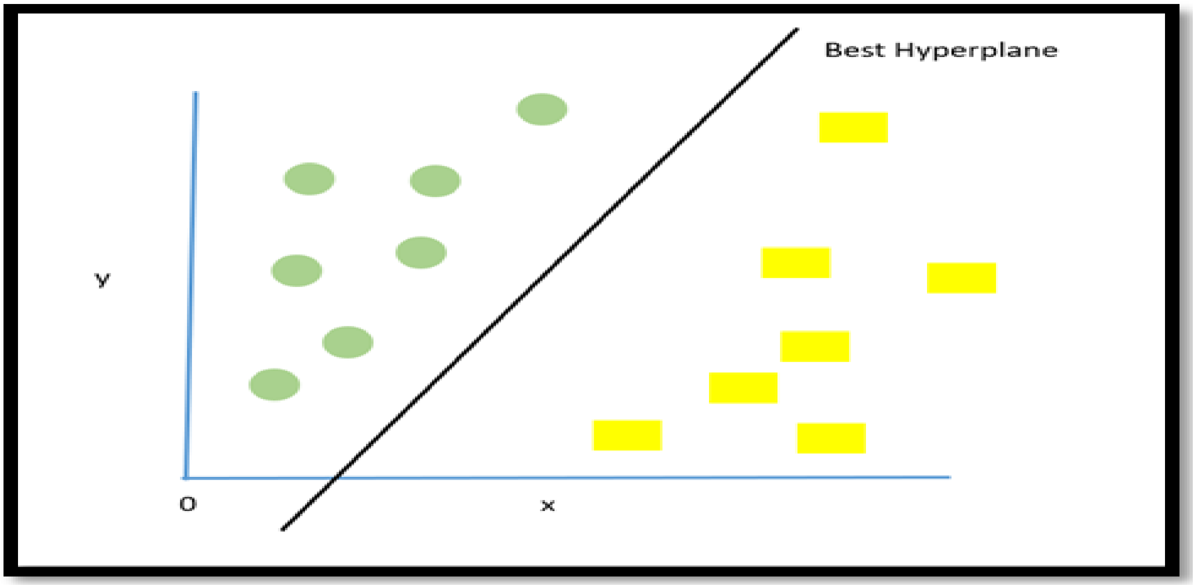


Figure 3: Straight Line is the Best Hyperplane

Figure 3 shows that simple straight line is the best hyperplane in 2D. So, the question is, what is the best hyperplane in SVM? The best one is, that have the augment margin from the both tags.

METHODOLOGY

An important part of the project is to automate the system that the system analyzes and generate results itself. For implementing the facial expression recognition, Supervised Machine Learning (SVM), a very popular and diligent model is used.

Support Vector Machine (SVM) is used in the proposed method for the Facial Expression Recognition. The work flow of the SVM model works is shown in the Figure 4.

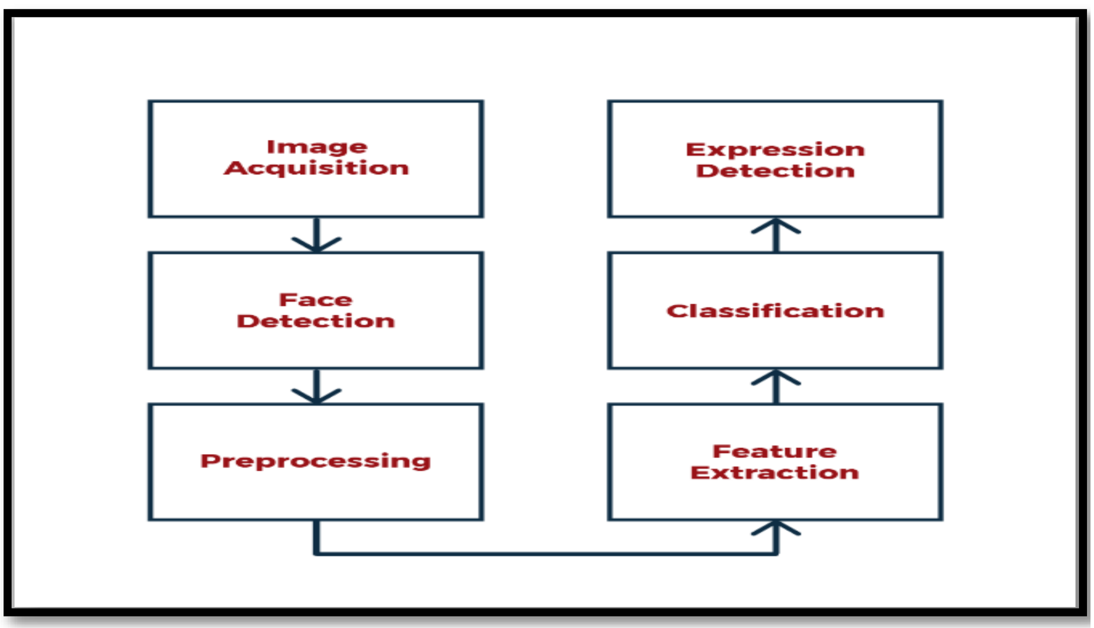


Figure 4: Facial Recognition Methodology

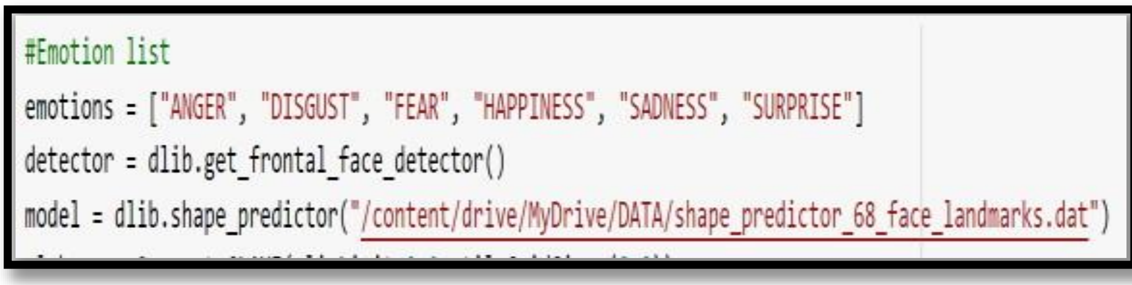
A. Image Acquisition:

In image acquisition, the images are gathered using external sources such as camera usually. These images are then taken for further processing and undergo a certain procedure.

B. Face Detection:

Face Detection uses different modules to identify the faces and generate results. Face detection is also performed using Neural Network with the help of Haar Cascade model. It provides fast processing but compromises on the accuracy of the result. On the contrary, Support Vector Machine is preferred here, using shape_predictor_68_face_landmarks model. This model generates 68 target points on the face of human. This model provides relatively higher accuracy when compared to Neural Network's Haar Cascade.

Every model is used with the help of a programming language and here, Python language is used along with its library dilib, that finds and recognizes the face in human's body. Figure 5 shows how we used this library in our model.



```
#Emotion list
emotions = ["ANGER", "DISGUST", "FEAR", "HAPPINESS", "SADNESS", "SURPRISE"]
detector = dlib.get_frontal_face_detector()
model = dlib.shape_predictor("/content/drive/MyDrive/DATA/shape_predictor_68_face_landmarks.dat")
```

Figure 5: Using dlib Library

C. Preprocessing:

During the data preprocessing, the prime focus is to provide the dataset that is relevant and clean to the model, that can be further used in the regression or classification procedure.

Data Augmentation helps during the preprocessing as it increases the amount of data by producing new data points form the same existing data. That is also done by taking images from different angles. During the preprocessing, the face forms the image is also converted into grayscale.

The model is trained on the training set of data and then the training is assessed on the portion of data kept for testing. For the optimum results, usually the split ratio of the data should be 70%-80% and 20%- 30% for training and testing respectively. In our system, for best possible results the dataset is split into 80% to 20% ratio for training and testing respectively.

D. Feature Extraction:

Feature are extracted from the face automatically since shape_predictor_68_face_landmarks model is used that generates 68 target points on the face to detect the features of human face.

E. Classification:

The dataset for the system is linear so the SVM model classifies the linear dataset into linear hyperplanes.

F. Expression Detection:

Expressions are detected and identified that which kind of expression from the human face is received to the system, whether it is happy, sad, angry or any other expression.

SUPPORT VECTOR CLASSIFICATION

In our system, we are using SVM from scikit-learn instead of using SVM from TensorFlow.

from sklearn import svm

Support Vector Classification is a class of scikit- learn SVM and it is used in the proposed model. And a library is required to execute any model, here LIBSVM is used for the implementation.

A. LIBSVM:

LIBSVM Chang, Chih-Chung, and Chih-Jen Lin.(2011) & Abdiansah et.al.,(2015)& (El Mamoun, Mamouni (2023) is an easy, effortless and efficient library for SVM classification and regression. It also provides implementation for many models and classes such as C-SVC (Support Vector Classification) Saha, Sanjib. (2023) & Takci et.al (2023) & Prentice et.al (2023), epsilon-SVR (Support Vector Regression) etc. Briefly, SVM classifier is used to train the model.

B. Kernel:

Kernel method takes input data and converts it into the processing data in the form that is required. In general, it converts training dataset so that a linear equation is formed from a non-linear decision surface with dimension spaces having large number. We have used linear kernel Sahili, Mahabba El, et.al (2023) & Cai, Difeng, Edmond Chow et.al(2023) in our system as our dataset is linear as shown in figure 6 and creates linear hyperplane.

```
>>> linear_svc = svm.SVC(kernel='linear')
>>> linear_svc.kernel
'linear'
```

Figure 6: Using Linear Kernel

VII. EXPERIMENT

The SVM model is tested with different datasets, such as CK+ and Fer2013 are used as datasets and are implemented with SVM.

A. Training Code for All Datasets:

Figure 7 displays the training code, where the SVM model is trained and evaluated with each dataset split into 80% and 20% ratio for training and testing respectively. And linear kernel is used in the model.

B. Ck+ Dataset:

The prime focus was on implementing the SVM model by using CK+ dataset to achieve the best accuracy level. And this dataset provided the best accuracy.

1) Accuracy Level:

```
Training SVM Classifier 0
Getting accuracy score -- 0
Test Accuracy:  0.9096121416526138
Mean Accuracy Value: 0.910
[[ 836    7    6    9   14   18]
 [  16  655    8   13   37    5]
 [  18   11  467   15    8   67]
 [   6    7    1 1577    9   18]
 [  17   48   12   26  848   27]
 [  15   13   40   30   15 1011]]
              precision    recall  f1-score   support

           0       0.92      0.94      0.93       890
           1       0.88      0.89      0.89       734
           2       0.87      0.80      0.83       586
           3       0.94      0.97      0.96      1618
           4       0.91      0.87      0.89       978
           5       0.88      0.90      0.89      1124

    accuracy                           0.91      5930
   macro avg       0.90      0.89      0.90      5930
weighted avg       0.91      0.91      0.91      5930

Best accuracy =  90.96121416526138 percent
SVC(C=0.01, decision_function_shape='ovo', kernel='linear', probability=True)
```

Figure7: Classification Report using CK+

Figure 8 displays the result of SVM model by using CK+ dataset and an accuracy of 90.96% is achieved using the CK+ dataset with SVM.

```
def create_model():
    accur_lin = []
    max_accur = 0
    for i in range(0,1):
        #Make sets by random sampling 80/20%
        print("Marking set %s" %i)
        X_train, y_train, X_test, y_test = make_sets()

        #Turn the training set into a numpy array for the classifier
        np_X_train = np.array(X_train)
        np_y_train = np.array(y_train)
        #train SVM
        print("Training SVM Classifier %s" %i)
        clf.fit(np_X_train, np_y_train)

        np_X_test = np.array(X_test)
        np_y_test = np.array(y_test)
        #Use score() function to get accuracy
        print("Getting accuracy score -- %s" %i)
        #npar_pred = np.array(X_test)
        pred_lin = clf.score(np_X_test, np_y_test)
        #y_pred = clf.predict(X_test)
        #Find Best Accuracy and save to file
        if pred_lin > max_accur:
            max_accur = pred_lin
            max_clf = clf
            X_test_opt = np_X_test
            y_test_opt = np_y_test
            X_train_opt = np_X_train
            y_train_opt = np_y_train
            test_pred = max_clf.predict(np_X_test)
            #train_pred = max_clf.predict(np_X_train)
            #print("Hello")
        #y_pred = clf.predict(np_X_test)
        print("Test Accuracy: ", pred_lin)
        #print(confusion_matrix(np_y_test, y_pred))
        accur_lin.append(pred_lin)  #Store accuracy in a list

    print("Mean Accuracy Value: %.3f" %np.mean(accur_lin))    #Get mea
    #test_pred = max_clf.predict(X_test_opt)
    #print(confusion_matrix(y_train_opt, train_pred))
    #print(classification_report(y_train_opt, train_pred))
    print(confusion_matrix(y_test_opt, test_pred))
    print(classification_report(y_test_opt, test_pred))

    return max_accur, max_clf
```

```
#Emotion list
emotions = ["ANGER", "DISGUST", "FEAR", "HAPPINESS", "SADNESS", "SURPRISE"]
detector = dlib.get_frontal_face_detector()
model = dlib.shape_predictor("/content/drive/MyDrive/DATA/shape_predictor_68_face_landmarks.dat")
clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8,8))
clf = SVC(C=0.01, kernel='linear', decision_function_shape='ovo', probability=True)
```

Figure 8: Training of SVM Model

C. Fer2013 Dataset:

Fer2013 dataset [22] [23] [24] was also experimented with the SVM model to compare the accuracy results with other datasets and it resulted in 50% accuracy after first training. It possibly can achieve better accuracy level with further more evaluation.

1) Accuracy Level:

```
Training SVM Classifier 0
Getting accuracy score -- 0
Test Accuracy:  0.5053671103477888
Mean Accuracy Value: 0.505
[[ 179    0   37  117  197   46   36]
 [  23    0    9   16   16    9    0]
 [  92    0   52  104  183   68   93]
 [  37    0   31 1120  115   23   46]
 [  41    0   42  107  585   59   56]
 [  66    0   33  103  243  114   27]
 [  21    0   24   71  103   10  304]]
              precision    recall  f1-score   support

           0       0.39      0.29      0.33       612
           1       0.00      0.00      0.00        73
           2       0.23      0.09      0.13       592
           3       0.68      0.82      0.74      1372
           4       0.41      0.66      0.50       890
           5       0.35      0.19      0.25       586
           6       0.54      0.57      0.56       533

    accuracy                           0.51      4658
   macro avg       0.37      0.37      0.36      4658
weighted avg       0.46      0.51      0.47      4658

Best accuracy =  50.53671103477888 percent
SVC(C=0.01, decision_function_shape='ovo', kernel='linear', probability=True)
```

Figure 9: Classification Report using Fer2013

Figure 9 displays the result of SVM model with Fer2013 dataset with 50.53% accuracy achieved.

D. Combination of Ck+ and Fer2013 Dataset:

We merged the CK+ [25] and Fer2013 dataset and trained the SVM model with this combination to check the performance result as an experiment.

1) Accuracy Level:

According to figure 10 the classification report shows that the accuracy of 72.61% has been achieved, the model can increase its performance when trained further.

The results concluded from the SVM model using Fer2013 was an experiment. The goal was to get efficient results from SVM by using CK+ model, and the efficiency in results were achieved.

```
Getting accuracy score -- 0
Test Accuracy:  0.726123728974627
Mean Accuracy Value: 0.726
[[1001   13   41  100  192   49   52]
 [  41  638   15   32    8   44   17]
 [ 104   17  564  135  179   71  147]
 [  41    6   35 2728  107   41   54]
 [  82    0   38  119  530   51   58]
 [  92   67   56  110  280  901   46]
 [  34    2   80  105   96   25 1279]]
              precision    recall  f1-score   support

           0       0.72      0.69      0.70      1448
           1       0.86      0.80      0.83       795
           2       0.68      0.46      0.55      1217
           3       0.82      0.91      0.86      3012
           4       0.38      0.60      0.47       878
           5       0.76      0.58      0.66      1552
           6       0.77      0.79      0.78      1621

    accuracy                           0.73     10523
   macro avg       0.71      0.69      0.69     10523
weighted avg       0.74      0.73      0.73     10523

Best accuracy =  72.61237289746269 percent
SVC(C=0.01, decision_function_shape='ovo', kernel='linear', probability=True)
```

Figure 10: Classification Report using combination of Fer2013 and CK+

GRAPH OF DIFFERENT ACCURACT WITH DIFFERENT DATASETS

Figure 11 shows the accuracy of the SVM models using different datasets. These models provided 50.53% accuracy with Fer2013, 72.61% with combination of Fer2013 and CK+ & 90.96% by using CK+ dataset, providing the most accurate results. Hence, the SVM model with CK+ dataset is used in our Facial recognition system. The SVM model achieved the highest accuracy of 90.96% using the CK+ dataset. This suggests that the CK+ dataset, known for its controlled expressions under controlled lighting conditions, is well-suited for training the model to recognize facial expressions accurately. The combination of FER-2013 and CK+ yielded a respectable accuracy of 72.61%. This indicates that combining datasets can potentially benefit model performance, although it might not always reach the peak achieved with a more homogeneous dataset like CK+. The SVM model achieved a lower accuracy of 50.53% on the

FER-2013 dataset. Based on the significant accuracy advantage observed with the CK+ dataset, we opted to utilize the SVM model trained on this dataset for our facial recognition system. This choice prioritizes maximizing recognition accuracy for the intended application.

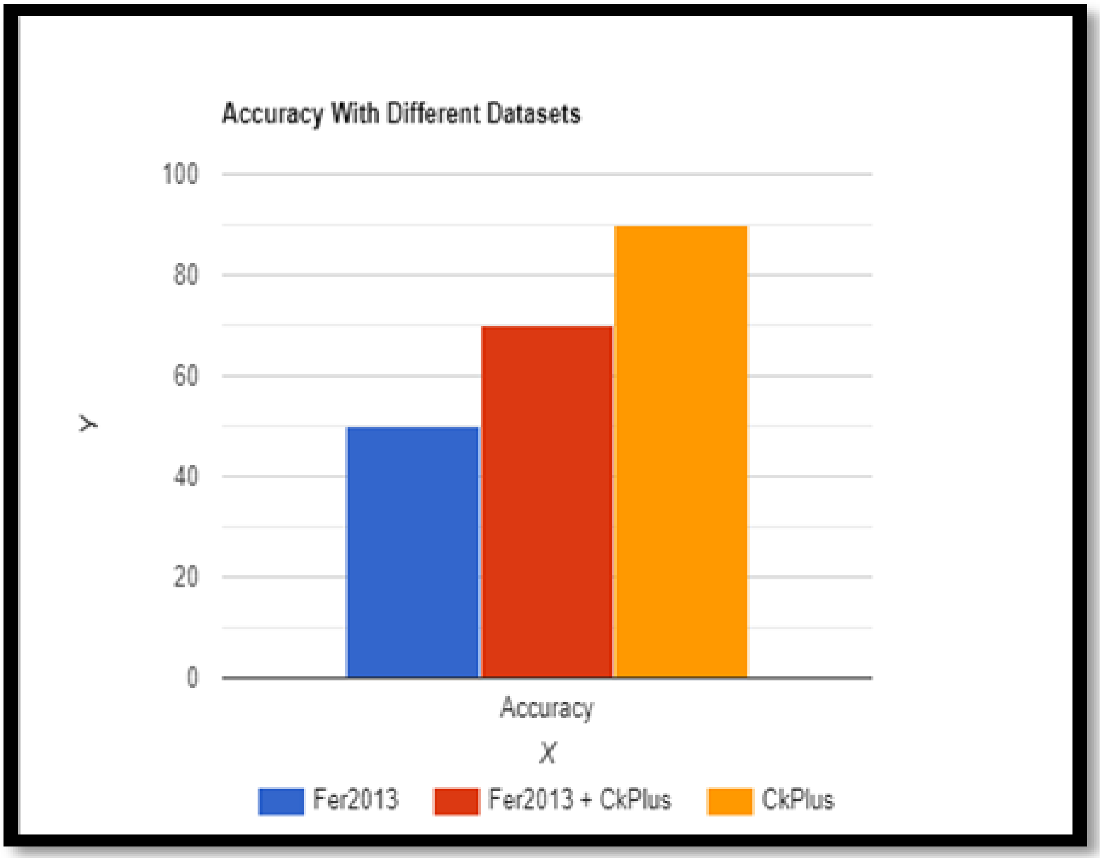


Figure 11: Accuracy Graph using Different Datasets

CONCLUSION

Purpose of this research is to get the new standard efficiency of facial expression recognition system by using SVM. For the accomplishment of our goal, we used ck+ dataset for SVM model. Much research has been done in this domain, but we try to get new performance. Firstly, we faced many hurdles in the form of environmental variables, training of model and using right dataset to satisfy our need. Training of model is the key factor in the entire process. This system was developed to process facial expressions and identify expressions in the term of six basic emotions. The efficiency of the facial expression recognition is dependent on the training of the model. The model will provide maximum efficiency with best training. Usually there is tradeoff between the accuracy and processing speed of the model. One has to be compromised. In future, we will try to work on such algorithm, where the accuracy of the model is not compromised with its processing speed.

REFERENCES

Abdulazeez, Adnan Mohsin, and Zainab Salih Ageed. "Face Emotion Recognition Based on Machine Learning: A Review." International Journal of Informatics, Information System and Computer Engineering (INJIISCOM) 5.1 (2024): 21-55.

Xia, Ziqing, et al. "An interpretable English reading proficiency detection model in an online learning environment: A study based on eye movement." Learning and Individual Differences 109 (2024): 102407.

Bellaj, Mohamed, et al. "Educational Data Mining: Employing Machine Learning Techniques and Hyperparameter Optimization to Improve Students' Academic Performance." International Journal of Online & Biomedical Engineering 20.3 (2024).

Pramanik, Bablu, and Sneha Singh. "Enhancing Online Education Experience Using Learners' Comments: A Novel Approach To Feedback Analysis." Educational Administration: Theory and Practice 30.4 (2024): 9633-9638.

Priya, M., et al. "Automatic Emotion Detection using SVM-based Optimal Kernel Function." 2024 11th International Conference on Computing for Sustainable Global Development (INDIACom). IEEE, 2024.

Tadese Henok Seifu, 2022, Automated Facial Expression Recognition using SVM and CNN, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 11, Issue 03 (March 2022)

Zhigang Yu, Yunyun Dong, Jihong Cheng, Miaomiao Sun, Feng Su, "Research on Face Recognition Classification Based on Improved GoogleNet", Security and Communication Networks, vol. 2022, Article ID 7192306, 6 pages, 2022. https://doi.org/10.1155/2022/7192306

I.Michael Revina, W.R. Sam Emmanuel, A Survey on Human Face Expression Recognition Techniques, Journal of King Saud University - Computer and Information Sciences, Volume 33, Issue 6, 2021,

Cossetin, M.J., Nievola, J.C., Koerich, A.L., 2016. Facial expression recognition using a pairwise feature selection and classification approach. IEEE Int. Jt. Conf. Neural Networks, pp. 5149–5155.

Salmam, F.Z., Madani, A., Kissi, M., 2016. Facial Expression Recognition using Decision Trees. IEEE 13th Int. Conf. Comput. Graph. Imaging Vis., 125–130. doi: 10.1109/CGiV.2016.33.

T.A. Rashid Convolutional neural networks based method for improving facial expression recognition Intell. Syst. Technol. Appl., 73–84 (2016), 10.1007/978-3-319-47952-1

P.C., Vasanth & K.R., Nataraj. (2015). Facial Expression Recognition Using SVM Classifier. Indonesian Journal of Electrical Engineering and Informatics (IJEEI). 3. 10.11591/ijeei.v3i1.126.

Poursaberi, H.A. Noubari, M. Gavrilova, S.N. Yanushkevi ch Gauss – Laguerre wavelet textural feature fusion with geometrical information for facial expression identification EURASIP J. Image Video Process. (2012), pp. 1-13

Song, D. Tao, Z. Liu, X. Li, M. Zhou Image ratio features for facial expression recognition application IEEE Trans. Syst. Man Cybern. Part B Cybern., 40 (2010), pp. 779-788.

Wang, H., Hu, Y., Anderson, M., Rollins, P., 2010. Emotion Detection via Discriminative Kernel Method. Proc. 3rd Int. Conf. PErvasive Technol. Relat. to Assist. Environ.

G. Zhao, M. Pietikäinen Boosted multi-resolution spatiotemporal descriptors for facial expression recognition Pattern Recognit. Lett., 30 (2009), pp. 1117-1127,

S. Bashyal, G.K.V. Venayagamoorthy Recognition of facial expressions using Gabor wavelets and learning vector quantization Eng. Appl. Artif. Intell., 21 (2008), pp. 1056-1064,

Y. Gao, M.K.H. Leung, S.C. Hui, M.W. Tananda Facial expression recognition from line-based caricatures IEEE Trans. Syst. Man Cybern. A Syst. Hum., 33 (2003), pp. 407-412.

Muhammed, A., Mol, R., Vijay, L. R., Muhammed, S. A., & Mol, A. S. (2020). Facial expression recognition using support vector machine (SVM) and convolutional neural network (CNN). *International Journal of Research in Engineering, Science and Management*, *3*(8), 574-577.

Feng, Luoyin, and Xin Chen. "Image recognition and encryption algorithm based on artificial neural network and multidimensional chaotic sequence." Computational Intelligence and Neuroscience 2022 (2022).

Cheng, Yongjian, et al. "Emotion-Recognition Algorithm Based on Weight-Adaptive Thought of Audio and Video." Electronics 12.11 (2023): 2548.

Majeed, Fahad, et al. "Investigating the efficiency of deep learning based security system in a real-time environment using YOLOv5." Sustainable Energy Technologies and Assessments 53 (2022): 102603.

Kavita, and Rajender Singh Chhillar. "Machine Learning Techniques for Real-Time Human Face Recognition." International Conference on Information Science and Applications. Singapore: Springer Nature Singapore, 2023.

Chang, Chih-Chung, and Chih-Jen Lin. "LIBSVM: a library for support vector machines." ACM transactions on intelligent systems and technology (TIST) 2.3 (2011): 1-27.

Abdiansah, Abdiansah, and Retantyo Wardoyo. "Time complexity analysis of support vector machines (SVM) in LibSVM." Int. J. Comput. Appl 128.3 (2015): 28-34.

El Mamoun, Mamouni. "An Effective Combination of Convolutional Neural Network and Support Vector Machine Classifier for Arabic Handwritten Recognition." Automatic Control and Computer Sciences 57.3 (2023): 267-275.

Kashef, Amirarash, and Junfeng Ma. "Real-time Facial Emotion Recognition Using FER2013 Image Dataset." IIE Annual Conference. Proceedings. Institute of Industrial and Systems Engineers (IISE), 2023.

Meena, Gaurav, et al. "Identifying emotions from facial expressions using a deep convolutional neural network-based approach." Multimedia Tools and Applications (2023): 1-22.

Han, Bin, et al. "Masked FER-2013: Augmented Dataset for Facial Expression Recognition." 2023 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW). IEEE, 2023.

REDUZZI, Carolina, et al. "Abstract P2-26-06: Association between CK+/CD45+ circulating tumor cells (CTCs) and circulating tumor DNA (ctDNA) alterations in advanced breast cancer patients." Cancer Research 83.5_Supplement (2023): P2-26.

Saha, Sanjib. "An Empirical Comparison of Linear and Non-linear Classification Using Support Vector Machines." (2023).

Takci, Hidayet, and Nusrat Fatema. "Highly Accurate Spam Detection with the Help of Feature Selection and Data Transformation." Int. Arab J. Inf. Technol. 20.1 (2023): 29-37.

Prentice, Justin SC. "Support Vector Classifiers in scikit-learn: Mathematical Detail, Part III." (2023).

Sahili, Mahabba El, and Faisal N. Abu-Khzam. "A Linear Kernel for Planar Vector Domination." arXiv preprint arXiv:2312.09374 (2023).

Cai, Difeng, Edmond Chow, and Yuanzhe Xi. "Data-driven linear complexity low-rank approximation of general kernel matrices: A geometric approach." Numerical Linear Algebra with Applications 30.6 (2023): e2519.