Emotion Detection for Image-Based Classification

Montaño, Chleo Nicole C.
College of Computing and Information Technology
Manila, Philippines
motanocc@students.national-u.edu.ph

Paat, Margarete A.
College of Computing and Information Technology
Manila, Philippines
paatma@students.national-u.edu.ph

Pagdanganan, Xavier P.
College of Computing and Information Technology
Manila, Philippines
pagdangananxp@students.national-u.edu.ph

***Abstract*—This paper presents the development of an emotion classification system using facial expressions from images in the FER-2013 dataset. Two techniques were used: Principal Component Analysis (PCA) for feature reduction and a Support Vector Machine (SVM) for classification, with comparisons against Logistic Regression and k-Nearest Neighbors. Results show that SVM, using an RBF kernel, achieved 48.19% accuracy, outperforming the other baseline models. Findings highlight the potential of SVM for emotion classification but also suggest that further work is needed to improve accuracy, especially for underrepresented emotions.**

Keywords: emotion classification, machine learning, Support Vector Machine (SVM), Principal Component Analysis (PCA), facial expressions, FER-2013, Logistic Regression, k-Nearest Neighbors.

I. Introduction

In a progressively digitized world, the possibility of gathering human emotion data through visual means has wide applications for various fields such as psychology or marketing. With emotion detection technology for classification of images trained to seek human emotional expression, we can gain valuable insights into user behavior and preferences. As a recent study reported in Scientific Reports illustrates, deep learning techniques for image emotion classification extremely accurately, underlining how this technology might be used to improve people's experience in a variety of applications from social media interactions to customer service (Nature, 2024).

To tackle this problem, our solution utilizes powerful image processing algorithms and complex machine learning models to read facial expressions as well as interpret other visual cues that allow us to detect emotional states very accurately. This contribution holds special importance for the wider field of human computer interaction, as there is a current trend in making systems more responsive, empathic and personalized by considering emotions. As written in Widanagamaachchi (2009), "The ability to recognize and interpret emotions in images can transform how machines interact with humans, making them not just tools, but companions that understand our feelings."

Although current solutions show some promise, they need to be improved in terms of dataset diversity and real-time processing. Many existing models are trained on limited datasets that fail to capture the full spectrum of human emotions or cultural expressions, often resulting in biased outcomes that do not reflect the rich diversity of human experience(Widanagamaachchi 2009). One of the main bottlenecks for real time emotion detection is its computational demand which can lead it to be impractical and most importantly if a technology is not practical, then in no case it will prove useful.

It can be used by a social media platform that wants to suggest content better or help tailored mental health apps better support users with respect to their emotions. Even as we wallow in these difficulties, toiling under the weight of tools without empathy — and at times seemingly incapable even by design of meeting them – one can only dream about a future where technology could not just pick up on our most complex emotions but respond back.

Use cases range from enhancing virtual reality experiences, to designing more ergonomic user interfaces in software. Our mission is to work towards creating more natural interactions and empathetic experiences for end-users by overcoming the limitations of current methods, through a novel paradigm built on image-based emotion detection.

II. Review of Related Literature

A Support Vector Machine (SVM) is a powerful machine learning technique for classification tasks, as highlighted by Awad and Khanna (2015). In the book by Schölkopf, B., & Smola, A. J. (2002), it stated that SVM functions as a sparse kernel decision machine, utilizing a subset of training data known as support vectors to define hyperplane margins. This approach allows SVM to handle classification complexity more effectively, as the number of support vectors, rather than the input dimensionality, primarily influences performance.

Additionally, SVM employs the kernel trick to map data into higher-dimensional spaces, enabling the learning of nonlinear decision boundaries through convex optimization. By maximizing the margin between classes, SVM enhances generalization and mitigates overfitting.

III. Methodology

In this study, we used machine learning techniques to classify facial expressions from the FER-2013 dataset. The goal was to explore the efficacy of feature extraction using Principal Component Analysis (PCA) and classification using a Support Vector Machine (SVM). This section outlines the data collection, preprocessing, experimental setup, algorithms, training procedures, evaluation metrics, and baselines.

A. Data Collection

The FER-2013 dataset contains 35,887 grayscale images of 48x48 pixels, divided into 7 emotional classes: happy, sad, angry, surprise, disgust, fear, and neutral. It is split into a training set with 28,709 images and a test set with 7,178 images. The FER-2013 dataset is publicly available and frequently used in facial expression classification research. It was compiled and shared as part of a Kaggle competition. Each image in the dataset is labeled with one of seven emotion categories, providing a balanced and labeled dataset ideal for supervised learning.

B. Data Pre-Processing

To classify emotions based on the facial expressions present in the images, the dataset undergoes several pre-processing steps to achieve optimal results:
oi

- The images were rescaled by a factor of 1/255 to normalize pixel values between 0 and 1. Using ImageDataGenerator from Keras.
- Training Data: Loaded in batches from the training path (/kaggle/input/fer2013/train) with a target size of 48x48 pixels, grayscale color mode, and categorical class mode. The Validation Data is loaded from the test path (/kaggle/input/fer2013/test) under similar conditions.
- To better understand the dataset, a sample of images from each emotion class was plotted using plot_images function for the following emotions: Surprise, Disgust, Angry, Fear, Happy, Neutral, and Sad.
- The dataset is split into training and validation sets. The training data consists of 28,709 images, while the validation data contains 7,178.
- The distribution of images across each emotion category was analyzed and plotted for both training and testing datasets. This revealed imbalances, with emotions like happiness having more samples than others like disgust.
- PCA was applied to reduce dimensions while retaining 95% variance, creating a more compact feature set for efficient SVM training. This preprocessing step mitigated the curse of dimensionality and computational load.

C. Feature Generation, Transformation and Experimental Setup

To classify the Emotion, we have used different techniques and approaches for feature generation to extract the important parts of the dataset and import the necessary tools and libraries to attain the highest accuracy possible for this SVM Model. Here's the detailed breakdown of the usages:

1) Experimental Setup
   Application of Python with key libraries, including NumPy for numerical operations, Matplotlib for plotting, scikit-learn for PCA and SVM, and TensorFlow/Keras for data loading and augmentation.
   Includes:
   - TensorFlow/Keras
   - scikit-learn
   - Matplotlib
2) Feature Generation and Transformation
   The SVM classifier operates on flattened pixel values of the images as features. The images, originally 48×48 pixels, are transformed into 1D vectors for SVM input. This results in a feature vector of 2,304 dimensions (since 48×48=2304). Here's a detailed breakdown of the feature generation process:
   - **Flattening the Images**: Each image is converted from a 2D array into a 1D array. This transformation is necessary because SVMs require input features to be in a vector format.
   - **Principal Component Analysis (PCA)** was used to reduce the dimensionality to 95% variance retention, whitening the data to normalize variance across components. This produced a compact feature set representing each image.
   - **Data Split**: The dataset was pre-split into training (80%) and test sets (20%).
   - **SVM parameters**: radial basis function (RBF) kernel, C=10, gamma='scale'.

D. Algorithm

Support Vector Machine (SVM) with RBF kernel was chosen for classification. SVM is known for distinguishing between classes by finding the optimal hyperplane that maximizes the margin between the closest data points of opposite classes which is suitable for emotion recognition. The SVM model is also known for its robustness and is well-suited for high-dimensional data with smaller training datasets. Its feature for classifying the data by finding an optimal line or hyperplane that maximizes the distance between each class in an N-dimensional space was suited for this dataset.

SVMs, particularly with RBF kernels, are effective for nonlinear classification and handle high-dimensional spaces, which PCA transformed images inherently represent. The decision_function_shape was set to 'ovr' for multi-class support. Equation 1 shows the formula for SVM.

$$\text{minimize } \frac{1}{2}\ |w|^2, \qquad \text{subject to} \qquad y_i(w \cdot x_i + b) \geq 1,\ \forall i$$

Equation 1. Support Vector Machine

where

- A. *minimize* for minimizing a specific loss function
- B. w is the weight vector (the normal to the hyperplane)
- C. b is the bias term
- D. y_i is the label of the emotion category ($y_i \in \{0,1,2,3,4,5,6\}$)
- E. x_i is the feature vector of the sample ($x_i \in \mathbb{R}^{2304}$)
- F. i corresponds to a data point

E. Training procedure

The model was trained on PCA-transformed data. A single train-test split was used due to the structure of the FER-2013 dataset. Additionally, one-hot encoded labels were converted to single integer classes for SVM compatibility.

0-angry, 1-disgust, 2-fear, 3-happy, 4-neutral, 5-sad, 6-surprised. Total of 7 classes.

A Support Vector Machine (SVM) classifier with a radial basis function (RBF) kernel was trained on the PCA-reduced features. PCA reduction was applied to flatten and reduce the feature space of images, removing redundant information and aiding in faster training for the SVM model. The Hyperparameters also defined a C=10 and gamma='scale' to balance model flexibility and accuracy.

F. Evaluation Metrics

The model was tested against a baseline accuracy and hinge loss, demonstrating SVM's comparative performance. Accuracy is the standard for emotion recognition. Hinge loss specifically reflects SVM performance.

Metrics Used:

- **Accuracy:** Percentage of correctly classified emotions, serving as the primary metric**.** Model accuracy on the test set was computed using accuracy_score.
- **Hinge Loss:** Measures the average misclassification penalty in SVM. Hinge loss was calculated to assess the margin error.
- **Precision**: Measures the accuracy of the positive predictions of each emotion.
- **Recall**: Identifies all actual positives of each emotion.
- **F1-Score**: Precision and Recall is being balanced to make sure both false positives and false negatives are minimized.

Accuracy and hinge loss are standard metrics in SVM-based classification. They provide a balanced view of performance, focusing on both predictive accuracy and margin effectiveness.

G. Baseline and Comparison

Deep-learning models provide a more accurate model for a large dataset like FER2013.

The SVM classifier was implemented as a primary method, utilizing a high loss function for performance evaluation. In addition to SVM, we implemented baseline models for comparison, specifically Logistic Regression and k-Nearest Neighbors (k-NN).

1) Logistic Regression was used as a linear baseline model, with a maximum of 1000 iterations for convergence. This model provides a simple baseline for classification.

2) A k-NN classifier with k=5 was applied, serving as a non-linear baseline to capture local patterns in the data.

IV. Results and Discussion

The FER2013 dataset has noisy data and variation in facial expressions. The Support Vector Machine (SVM) model, using RBF kernel, achieved an accuracy of 48.19% that indicates a performance better than Logistic Regression (37.25%) and k-Nearest Neighbors (25.17%). The model demonstrates stronger recognition of emotions like 'happy' and 'sad' but shows challenges with disgust and fear due to dataset imbalances. The evaluation metrics highlight the different performances of models included in the study. The table below summarizes the accuracy for each model tested.

Model	Accuracy
SVM	48.19%
Logistic Regression	37.25%
k-Nearest Neighbors	25.17%
Deep-Learning Models	at least 65%

Table 1

Table 1 shows that the primary classifier is SVM. To compare it with the baseline models such as Logistic Regression and k-Nearest Neighbors, the SVM surpassed the two, having an accuracy of 48.19%. The deep-learning models, on the other hand, have at least 65% accuracy which has the highest accuracy rate among the four. In improving the test accuracy of our model, changes in hyperparameters and the use of feature engineering techniques are potential as we achieve an enhanced SVM performance.

Emotion	Training Samples	Validation Samples
Surprise	Moderate	Moderate
Fear	Moderate	Moderate
Angry	Moderate	Moderate
Neutral	Moderate	Moderate
Sad	Moderate	Moderate
Disgust	Few	Few
Happy	More	More

Table 2: Emotion Distribution in Dataset

Table 2 shows that 'happy' has the highest training and validation samples, 'surprise', 'fear', 'angry', 'neutral', and 'sad' have moderate amounts of training and validation samples, but 'disgust' has only few samples. The imbalances illustrate the challenges in accurately classifying emotions.

Emotion	Precision	Recall	F1-score	Quantity
surprise	0.77	0.60	0.68	831
happy	0.56	0.70	0.62	1774
disgust	1.00	0.37	0.54	111
neutral	0.45	0.45	0.45	1233
sad	0.37	0.36	0.37	1247
fear	0.38	0.34	0.36	1024
angry	0.35	0.32	0.33	958
macro average	0.55	0.45	0.48	7178

Specifically, the model shows that 'surprise' and 'happy' have the highest performance with a f1-score of 0.68 and 0.62 respectively. This outcome means that when the model predicts 'surprise' or 'happy', it is relatively accurate. With a f1-score of 0.54, 'disgust', having few samples, also demonstrates a fairly good performance. However, having its very high precision but low recall indicates overfitting. 'Neutral', 'sad' and 'fear' have f1-scores of 0.45, 0.37, and 0.36 respectively. The model struggles with these categories as stated in their low precision and recall. Lastly, 'angry' has the lowest score of 0.33.

The findings align with existing research on SVMs which showcases its strengths in emotion classification while revealing specific challenges with certain emotions. However, the expected performance levels for emotions like 'disgust' indicate the necessity for further data augmentation and refined feature extraction techniques.

- Advantages and Limitations

The model was effective in dimensionality reduction through PCA, which enhanced its performance in high-dimensional spaces. A moderate accuracy of 48.19% implies that the model may not fully influence the dataset's potential, suggesting a need for additional feature engineering or alternative modeling approaches.

- Model Errors in Training Data

There are misclassifications found when an emotion is misidentified as a different one. For instance, some images of 'sad' are classified as 'neutral' due to the similarities of their facial expressions. Fewer samples, such as the disgust samples, also need a wider range of training data. Due to insufficient samples being trained, the disgust samples were not learning as much as other emotions. Thus, to enhance the model accuracy, it is important to add enough data for training, as well as modifying feature extraction methods.

V. Conclusion

This study used the FER-2013 dataset to investigate emotion recognition through facial expression classification. Our findings show that the Support Vector Machine (SVM) with a radial basis function (RBF) kernel obtained 48.19% accuracy, beating baseline models such as Logistic Regression (37.25%) and k-Nearest Neighbors (25.17%). This demonstrates SVM's effectiveness in complicated categorization jobs.

Our study makes a great contribution to human-computer interaction. Emphasizing the importance of working systems and compassion. Recognizing and interpreting emotions can turn your device into a helpful friend. Improved user experience in a variety of applications

However, there are limitations such as the unevenness of the data set, especially emotions like disgust and fear. It emphasizes the need to increase data diversity and improvement strategies. The computational requirements of real-time emotion recognition also present issues that need to be addressed.

Future research should consider expanding the data set to include emotional expressions and broader cultural contexts and various modeling techniques are explored. To improve classification accuracy by modifying these limitations. We hope to promote a more natural interaction between technology and users. It opens the door for machines to understand human emotions. In summary, our study lays the foundation for further investigation into emotion recognition technology. Bridging the gap between human emotional complexity and machine understanding and demonstrates a future where technology responds to emotions in a meaningful way.

References

[1] Agung, E.S., Rifai, A.P. & Wijayanto, T.. (2024). Image-based facial emotion recognition using convolutional neural network on emognition dataset. Scientific Reports, Volume(Issue).

[2] Widanagamaachchi, V. (2009). Emotion recognition from facial expressions: A review. International Journal of Computer Applications, 1(1), 1-8.

[3] Awad, M., Khanna, R. (2015). Support Vector Machines for Classification. In: Efficient Learning Machines. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4302-5990-9_3 .

[4] Schölkopf, B., & Smola, A. J. (2002). Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond .

[5] Downs, Tom, Kevin E. Gates, and Annette Masters. "Exact Simplification of Support Vector Solutions." *Journal of Machine Learning Research* 2 (2002): 293–297.

[6] Fine, Shai, and Katya Scheinberg. "Efficient SVM Training Using Low-Rank Kernel Representations." *Journal of Machine Learning Research* 2 (2002): 243–264.

[7] Kong, Bo, and Hong-wei Wang. "Reduced Support Vector Machine Based on Margin Vectors." In *CiSE 2010 International Conference on Computational Intelligence and Software Engineering*," 1–4. Piscataway, NJ: Institute for Electrical and Electronic Engineers, 2010.

[8] Lee, Yuh-Jye, and O. L. Mangasarian. "SSVM: A Smooth Support Vector Machine for Classification." *Computational Optimization and Applications* 20 (2001a): 5–22.