

Easeye: A Multimodal Real-Time Worker Stress Monitoring System Using Facial and Speech Emotion Recognition

Saransh Garg¹, Yash Chauhan², Siddhant Chaudhary³, Shashank Gautam⁴

1,2,3,4 Student, Department of Computer Science and Engineering, Raj Kumar Goel Institute of Technology and Management, Ghaziabad, India

Abstract - Workplace stress is a growing concern that negatively affects employee performance and well-being. This paper introduces Easeye, a real-time worker stress monitoring system based on multimodal emotion recognition. The system combines facial expression analysis and speech emotion detection to improve accuracy and reliability. OpenCV is used for real-time face detection, while a TensorFlow-based deep learning model classifies facial emotions. Simultaneously, audio signals are processed using Librosa to extract features such as MFCC, pitch, and energy for speech emotion recognition. A fusion mechanism integrates both outputs to estimate overall stress levels. The system is deployed through a Flask-based web interface, enabling continuous monitoring and visualization. Experimental results indicate that the multimodal approach performs better than single-modality systems. Easeye offers a practical and efficient solution for early stress detection, helping organizations enhance productivity and support employee mental health management in workplaces.

Key Words: Stress Monitoring, Emotion Recognition, Multimodal Systems, OpenCV, TensorFlow, Librosa, Machine Learning, Real-Time Systems

1. INTRODUCTION

In modern work environments, stress has become a critical issue affecting both individuals and organizations.

Limitations identified:

- Single modality reduces reliability
- Lack of real-time monitoring
- Limited integration with user interfaces

3. PROPOSED SYSTEM

3.1 System Overview

The Easeye system consists of four main modules:

High stress levels can lead to reduced productivity, increased errors, and long-term health issues. Existing stress detection methods, such as surveys and manual observation, are often delayed and subjective.

To address these limitations, automated systems using artificial intelligence have gained attention. However, most existing systems rely on a single input modality, such as facial expressions or speech signals, which limits their accuracy.

The proposed system, Easeye, introduces a multimodal approach that combines facial and speech emotion recognition to provide a more accurate and real-time assessment of worker stress.

2. LITERATURE REVIEW

Several studies have explored emotion detection using machine learning:

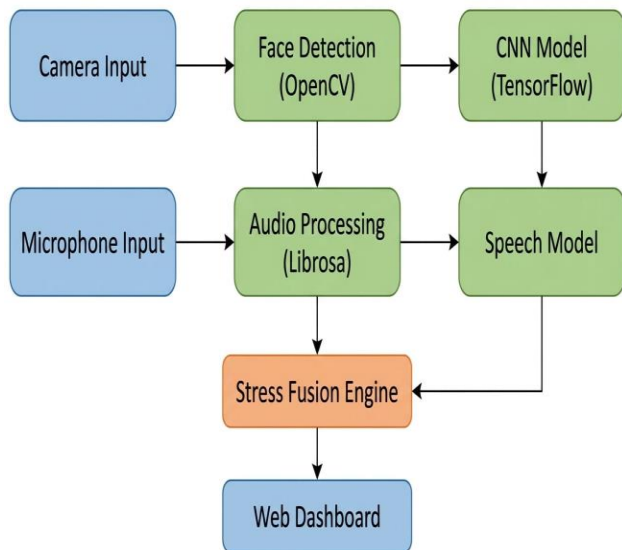
- Facial emotion recognition systems use CNN models trained on datasets like FER2013.
- Speech emotion recognition systems analyze vocal features such as pitch and MFCC.
- Some hybrid systems exist but lack real-time implementation or scalability.

1. Image Processing Module
2. Audio Processing Module
3. Stress Analysis Engine
4. Web Interface
5. Easeye addresses these gaps by integrating multiple modalities in a real-time framework.

3.2 System Architecture

The system architecture of Easeye is designed to perform real-time stress monitoring using both visual and audio inputs. The system takes input from a camera and microphone, which are processed through separate pipelines.

The camera input is used for face detection using OpenCV, and the detected face is passed to a CNN model implemented in TensorFlow for facial emotion classification. The microphone input is processed using Librosa to extract features such as MFCC and spectral characteristics, which are used by a speech emotion model to identify emotional states. As shown in the architecture, the face detection module is also connected to the audio processing stage to maintain synchronization between visual and audio data. The outputs from both the CNN model and speech model are then sent to the stress fusion engine, which combines the results to compute an overall stress level. Finally, the computed stress level is displayed on a web dashboard using a Flask-based interface, enabling real-time monitoring and visualization.



3.3 Facial Emotion Recognition

The facial emotion recognition module uses OpenCV for real-time face detection. Techniques such as Haar Cascade or deep learning-based detectors are used to locate faces in video frames. The detected face is preprocessed and fed into a convolutional neural network implemented in TensorFlow.

The model classifies emotions into categories such as happy, sad, angry, and neutral. Data preprocessing techniques such as normalization and resizing are applied to improve model performance. This module provides continuous emotion detection from live video input.

3.4 Speech Emotion Recognition

The speech emotion recognition module captures audio input in real time through a microphone. The audio signal is processed using the Librosa library to extract important features such as MFCC, spectral features, and zero-crossing rate.

These features are used as input to a trained model that predicts the emotional state of the speaker. The system handles short audio segments to ensure real-time processing and quick response.

3.5 Multimodal Fusion

The outputs from facial and speech modules are combined using a weighted fusion approach. This method assigns weights to each modality based on their reliability and computes a final stress score.

The stress score is calculated using a simple weighted formula:

$$\text{Stress Score} = (W1 \times \text{Facial Emotion}) + (W2 \times \text{Speech Emotion})$$

Based on the computed score, stress levels are classified into categories such as low, medium, and high. This approach improves accuracy by leveraging complementary information from both modalities.

3.6 Web-Based Monitoring System

The system is deployed using Flask, which provides a web-based interface for real-time monitoring. The dashboard displays stress levels dynamically and allows users to track changes over time. Additional features include historical data storage, visualization through graphs, and alerts when stress levels exceed a threshold.

4. IMPLEMENTATION

The Easeye system is implemented using Python and integrates multiple libraries and frameworks to enable real-time stress monitoring through multimodal emotion recognition. The implementation is divided into different stages, including data acquisition, preprocessing, model training, real-time processing, and web deployment.

4.1 Technologies Used

The system utilizes OpenCV for real-time face detection and image processing tasks. TensorFlow is used to design and deploy the convolutional neural network for facial emotion recognition. Librosa is employed for audio signal processing and feature extraction, particularly for extracting Mel Frequency Cepstral Coefficients (MFCC), spectral features, and zero-crossing rate. Flask is used to develop a web-based interface for real-time monitoring. Additional libraries such as NumPy, Pandas, and Scikit-learn are used for data handling, preprocessing, and model evaluation.

4.2 Data Acquisition and Preprocessing

The system captures real-time video input through a webcam and audio input through a microphone. For facial data, each video frame is converted into grayscale and resized to a fixed dimension before being passed to the model. Noise reduction and normalization techniques are applied to improve image quality.

For audio data, signals are segmented into short frames to enable continuous processing. Librosa is used to extract relevant features such as MFCC, pitch, and energy. Preprocessing steps such as normalization and filtering are applied to reduce noise and enhance feature quality.

4.3 Model Development and Training

The facial emotion recognition module uses a convolutional neural network trained on labeled datasets such as FER2013. The model learns to classify emotions into predefined categories including happy, sad, angry, and neutral. Techniques such as data augmentation, dropout, and batch normalization are used to improve generalization and prevent overfitting.

The speech emotion recognition module uses machine learning or deep learning models trained on datasets such as RAVDESS. Extracted audio features are used as input for classification. The model is trained to recognize emotional states based on speech patterns.

4.4 Real Time Processing

The system operates in real time by processing video and audio streams simultaneously. The facial emotion model predicts emotions frame by frame, while the speech model

processes audio segments continuously. Both outputs are generated with minimal latency to ensure smooth real-time performance.

4.5 Multimodal Integration

The outputs from facial and speech modules are combined using a weighted fusion technique. Each modality contributes to the final stress score based on predefined weights. This approach improves reliability and reduces the impact of errors from individual models.

4.6 Web Based Deployment

The system is deployed using a Flask framework that provides a web-based dashboard. The dashboard displays real-time stress levels, graphical representations, and alerts. Users can monitor stress trends over time through an interactive interface. The deployment ensures accessibility and scalability of the system in practical environments.

4.7 System Performance Optimization

To ensure efficient real-time performance, several optimization techniques are applied in the system. The facial emotion recognition model processes resized grayscale images to reduce computational load while maintaining accuracy. Similarly, audio signals are processed in short frames to minimize latency during speech analysis.

Parallel processing is used to handle both video and audio streams simultaneously, improving system responsiveness. Lightweight models and optimized libraries are selected to ensure smooth execution even on moderate hardware configurations. These optimizations enable the system to deliver real-time results without significant delays.

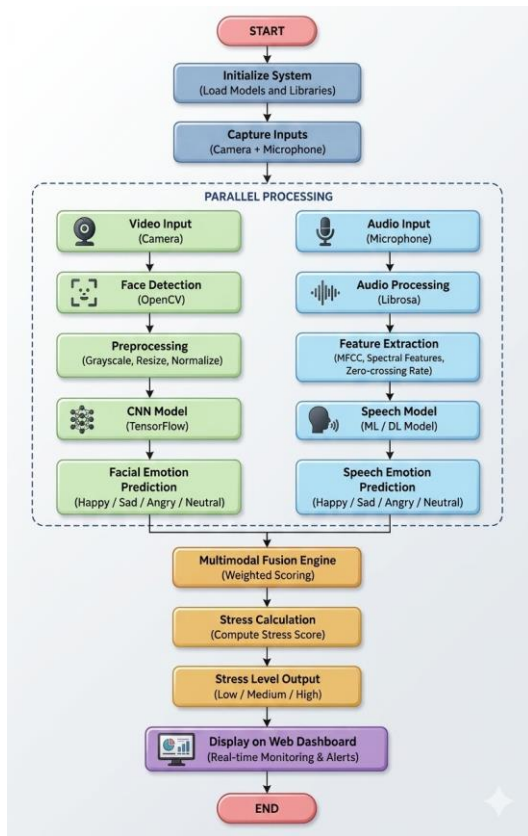


Fig -2: Flowchart

5. RESULTS AND DISCUSSION

5.1 Performance Metrics

The performance of the proposed system is evaluated using standard metrics such as accuracy, precision, recall, and F1-score. These metrics help in understanding the effectiveness of both individual modules and the combined system.

5.2 Experimental Results

The experimental results demonstrate that the proposed multimodal system outperforms individual facial and speech emotion models. The graph shows that the combined system achieves higher accuracy and better overall performance. This improvement is due to the integration of multiple data sources, which enhances reliability and reduces prediction errors.

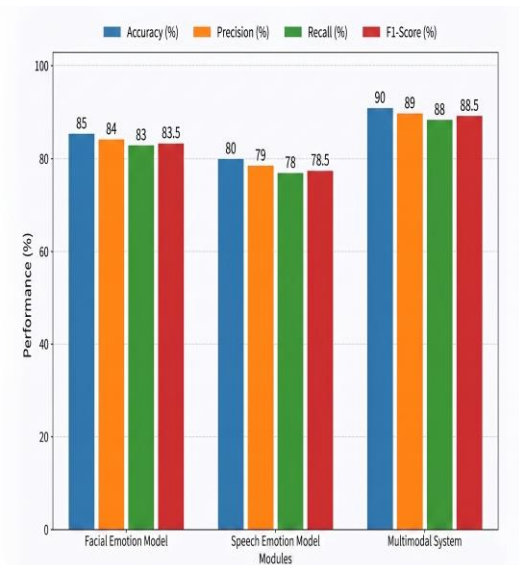


Fig -3: Performance Comparison of The Easeye System

5.3 Analysis

The experimental results demonstrate a consistent improvement in performance when transitioning from unimodal to multimodal emotion recognition. The facial emotion recognition module achieves strong accuracy under controlled conditions due to the discriminative power of convolutional features; however, its performance is susceptible to variations in illumination, pose, and partial occlusions. In contrast, the speech emotion recognition module captures prosodic and spectral cues that are complementary to visual features, but its reliability decreases in the presence of environmental noise and speaker variability.

The proposed multimodal framework leverages late fusion to integrate outputs from both modalities, thereby exploiting their complementary characteristics. This integration reduces the variance associated with individual models and improves generalization across diverse real-world conditions. The observed increase in overall accuracy, precision, recall, and F1-score indicates that the fusion strategy effectively mitigates modality-specific weaknesses while reinforcing consistent predictions. Moreover, the system demonstrates stable real-time performance with low latency, which is critical for continuous monitoring applications. The results suggest that multimodal learning provides a more robust and scalable solution for stress detection compared to traditional single-modality approaches. Consequently, the

proposed system achieves improved reliability and practical applicability in dynamic workplace environments.

6. ADVANTAGES

The proposed Easeye system offers several advantages over traditional stress detection methods. It provides real-time monitoring, enabling continuous assessment of stress levels without manual intervention. The use of a multimodal approach enhances accuracy by combining complementary information from facial and speech data. The system is non-invasive, as it relies only on camera and microphone inputs without requiring physical sensors. Additionally, the web-based architecture ensures accessibility and scalability, allowing deployment across different environments. The modular design also enables easy integration of additional features or models in the future.

7. LIMITATIONS

Despite its effectiveness, the system has certain limitations. The performance of the facial emotion recognition module is dependent on lighting conditions, camera quality, and facial visibility. Similarly, the speech emotion recognition module may be affected by background noise and variations in speech patterns. Real-time processing requires sufficient computational resources, which may limit performance on low-end devices. Additionally, the system currently relies on predefined emotion categories, which may not fully capture complex human emotional states. These factors can impact overall accuracy in uncontrolled environments.

8. FUTURE WORK

Future improvements can enhance the performance and applicability of the system. The integration of additional modalities, such as physiological signals (e.g., heart rate or EEG), can further improve stress detection accuracy. Advanced deep learning architectures, including transformer-based models, can be explored for better feature extraction and classification. The system can also be extended to a cloud-based platform for large-scale deployment and multi-user support. Furthermore, the development of a mobile application can increase accessibility and usability. Incorporating adaptive learning techniques can allow the system to personalize stress detection for individual users.

9. CONCLUSION

This paper presented Easeye, a real-time worker stress monitoring system based on multimodal emotion recognition. By combining facial expression analysis and speech emotion recognition, the system achieves improved accuracy and reliability compared to single-modality approaches. The integration of computer vision and audio processing techniques enables continuous and objective stress assessment. Experimental results demonstrate that the multimodal approach enhances performance and robustness in real-world conditions. The proposed system provides a practical solution for workplace stress monitoring and has the potential to improve employee well-being and productivity. With further enhancements, the system can be extended to broader applications and large-scale environments.

10. REFERENCES

- [1] P. Ekman, "Facial Expression and Emotion," American Psychologist, 1992. M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.
- [2] G. McKeown et al., "The SEMAINE Database: Annotated Multimodal Records of Emotion," IEEE Trans. Affective Computing, 2012.
- [3] S. R. Livingstone and F. A. Russo, "The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)," PLOS ONE, 2018.
- [4] A. Krizhevsky et al., "ImageNet Classification with Deep Convolutional Neural Networks," NIPS, 2012.
- [5] M. Pantic and L. J. M. Rothkrantz, "Automatic Analysis of Facial Expressions: The State of the Art," IEEE Trans. Pattern Analysis and Machine Intelligence, 2000.
- [6] B. Schuller et al., "Speech Emotion Recognition: Two Decades in a Nutshell," IEEE Signal Processing Magazine, 2018.
- [7] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," IJCV, 2004.
- [8] H. Gunes and M. Pantic, "Automatic, Dimensional and Continuous Emotion Recognition," International Journal of Synthetic Emotions, 2010.c
- [9] OpenCV Documentation, <https://opencv.org/>
- [10] TensorFlow Documentation, <https://www.tensorflow.org/>
- [11] Librosa Documentation, <https://librosa.org/>