



## Silent Distress Detection for Hospital Using Unintentional Human Micro-Actions

R.Niranjana<sup>1</sup>, B. H. Samuel Stephen Raj<sup>2</sup>, Shajith M<sup>3</sup>, Vinoth M<sup>4</sup>

<sup>1</sup>Assistant Professor Dept. of Computer Science Paavai Engineering College Namakkal, India.

<sup>2,3,4</sup>UG Student Dept. of Computer Science Paavai Engineering College, Namakkal, India.

**Email ID:** [niranjnaramasamypec@paavai.edu.in](mailto:niranjnaramasamypec@paavai.edu.in)<sup>1</sup>, [samraj08019@gmail.com](mailto:samraj08019@gmail.com)<sup>2</sup>, [shajithms007@gmail.com](mailto:shajithms007@gmail.com)<sup>3</sup>, [mvinoth759@gmail.com](mailto:mvinoth759@gmail.com)<sup>4</sup>

### Abstract

*Silent distress among hospitalized patients, particularly in intensive care units (ICUs), often goes unnoticed due to the inability of patients to communicate discomfort effectively. Traditional monitoring systems rely mainly on physiological parameters and manual observation, which may fail to capture subtle behavioral cues. This paper proposes an intelligent and non-invasive system for detecting silent distress using unintentional human microactions such as facial expressions, eye movements, posture changes, and involuntary body motions. The system utilizes computer vision and deep learning techniques, including Convolutional Neural Networks (CNN) for spatial feature extraction and Long Short-Term Memory (LSTM) models for temporal analysis. The system processes real-time video data to classify patient conditions into normal, mild distress, and critical distress levels. An automated alert mechanism notifies healthcare staff when abnormal patterns are detected. The proposed system improves patient safety, reduces response time, and enhances monitoring efficiency in hospital environments.*

**Keywords:** Silent Distress Detection, Computer Vision, Deep Learning, Micro-Action Analysis, Healthcare Monitoring, CNN, LSTM, ICU Monitoring, Patient Safety

### 1. Introduction

In modern healthcare environments, continuous monitoring of patients is essential, especially in intensive care units where patients may be unable to communicate due to severe medical conditions. Many patients experience distress that is not immediately visible through standard monitoring systems. Traditional monitoring methods primarily rely on physiological signals such as heart rate, oxygen saturation, and blood pressure. While these indicators are important, they often fail to detect early-stage distress that manifests through subtle behavioral changes. Unintentional human microactions, including slight facial expressions, eye movements, and minor body shifts, can serve as early indicators of discomfort or danger. However, continuous manual observation of these signals is not feasible due to workload and human limitations. To address this challenge, this project introduces an

AI based system that uses computer vision and deep learning to automatically detect micro-actions in real time. By analyzing these subtle cues, the system can identify distress early and alert healthcare professionals, thereby improving patient care and safety.

### 2. Ease of Use

#### • For Patients:

The system is completely non-invasive and does not require any wearable devices or physical attachments. Patients can remain comfortable while being monitored through cameras, without interference in their daily activities or medical procedures.

#### • For Healthcare Professionals:

The system provides real-time alerts and a centralized dashboard, allowing healthcare staff to monitor multiple patients efficiently. It reduces

the need for constant manual observation and helps prioritize critical cases.

- **Practical Advantages:**

The system is scalable and can be deployed in multiple hospital rooms. It improves response time, reduces workload, and enhances patient safety. It can also be integrated with existing hospital infrastructure [1-2]

### 3. Nomenclature and Notes

To ensure clarity and consistency, key terms and abbreviations used in this system are defined below.

**Abbreviations and Acronym:** Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

**Units:**

- CNN – Convolutional Neural Network
- LSTM – Long Short-Term Memory
- AI – Artificial Intelligence
- CV – Computer Vision
- ROI – Region of Interest
- FPS – Frames Per Second

**Some Common Mistakes:**

- Poor lighting conditions affecting detection accuracy
- Incorrect camera positioning
- Occlusion of patient face or body
- Insufficient training data
- False alerts due to sudden normal movements

### 4. Methodology

The methodology of the proposed silent distress detection system is designed in multiple stages to ensure accurate, realtime monitoring of patients using unintentional human microactions. The system integrates computer vision and deep learning techniques to analyze behavioral patterns and detect distress condition.[4-5]

- **Data Acquisition:**

The first stage involves capturing real-time video data from cameras installed in hospital rooms or intensive care units. The cameras are positioned to ensure clear visibility of the patient's face and upper body. The

video stream is continuously recorded and transmitted to the processing unit. The frame rate is maintained between 15–30 frames per second (FPS) to ensure smooth analysis while maintaining computational efficiency.

- **Preprocessing:**

The captured video stream is divided into individual frames for processing. Each frame undergoes preprocessing steps such as resizing, normalization, and noise reduction. Resizing ensures uniform input dimensions for the deep learning model, while normalization improves model performance. Noise reduction techniques such as Gaussian filtering are applied to remove unwanted distortions. Additionally, contrast enhancement is used to improve visibility under varying lighting conditions.

- **Face and Body detection :**

In this stage, the system detects the presence of a human subject and identifies key regions of interest (ROI), such as the face and upper body. Pre-trained models such as Haar Cascade classifiers or YOLO (You Only Look Once) are used for real-time object detection. Facial landmark detection is performed to identify important points such as eyes, eyebrows, and mouth, which are critical for micro-expression analysis. Body pose estimation techniques are used to track posture and movement.[2-3]

- **Feature Extraction:**

Feature extraction is carried out using a Convolutional Neural Network (CNN). The CNN processes each frame to extract spatial features such as facial expressions, eye movements, and body posture. Multiple convolutional layers are used to capture lowlevel and high-level features. Activation functions such as ReLU are applied to introduce non-linearity, and pooling layers are used to reduce dimensionality. The extracted features are represented as feature maps, which are passed to the next stage.

- **Temporal Analysis:**

Since distress is often identified through changes over time, temporal analysis is performed using a Long Short-Term Memory (LSTM) network. The LSTM processes sequences of feature vectors obtained from consecutive frames. It captures temporal dependencies such as repeated movements, sudden jerks, or prolonged inactivity. This stage is crucial for distinguishing between normal movements and distress-related patterns.

- **Distress Score Computation:**

The system computes a distress score based on multiple behavioral factors. These include facial expression intensity, movement irregularities, and posture deviations. Each factor is assigned a weight, and a combined score is calculated. The distress score helps quantify the level of discomfort or abnormal behavior observed in the patient.

- **Classification:**

The system classifies patient conditions into three categories:

- Normal
- Mild Distress
- Critical Distress

The classification is based on probability values, ensuring accurate and reliable decision-making.

- **Alert System:**

When the system detects mild or critical distress, an alert is generated automatically. The alert system is connected to a central monitoring dashboard and can also send notifications to healthcare staff via mobile devices. The alerts include patient ID, distress level, and timestamp, enabling quick response from medical personnel.

- **System Implementation:**

The entire system is implemented using Python programming language. OpenCV is used for video processing, while deep learning models are developed using TensorFlow or PyTorch frameworks. The backend handles data processing, and a web-based dashboard is used for visualization and monitoring.

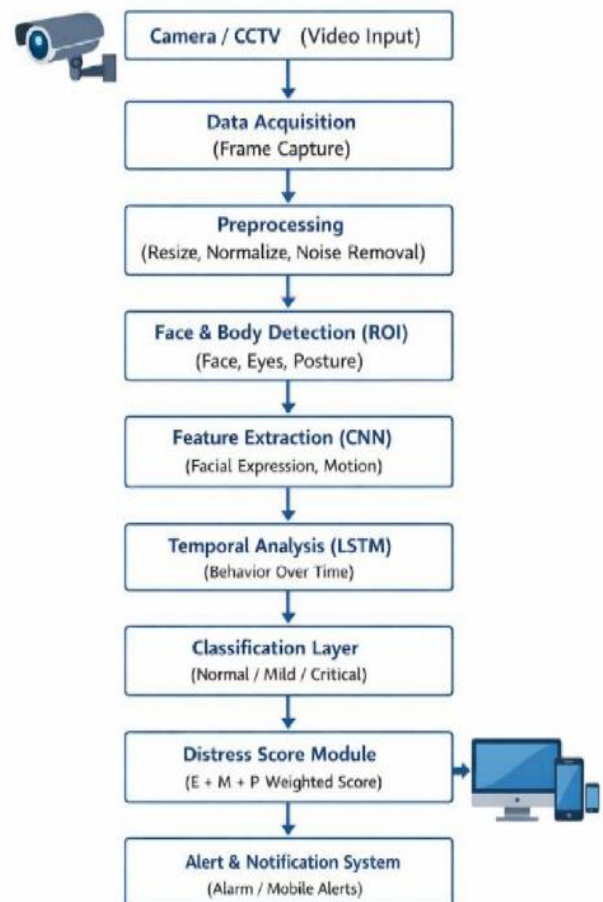
- **System Testing and Validation:**

The system is tested using publicly available datasets and simulated hospital scenarios. Performance metrics such as accuracy, precision, recall, and F1score are used to evaluate the model. The system demonstrates reliable performance in detecting distress-related micro-actions under controlled conditions.[5-6]

## 5. System Architecture

The system architecture of the proposed silent distress detection system is designed to provide a comprehensive and efficient framework for real-time patient monitoring in hospital environments. The architecture focuses on identifying unintentional human micro-actions, such as subtle facial expressions, eye movements, and body posture changes, which can indicate early signs of distress in patients who are unable to communicate verbally.

The system integrates advanced computer vision techniques with deep learning models to process continuous video input and extract meaningful behavioral information. It is structured as a multi-stage processing pipeline, where each module performs a specific function, ensuring smooth data flow and accurate analysis. The architecture is designed to operate in real time with minimal latency, allowing timely detection and response to critical situations. In addition, the system is developed to be non-invasive and scalable, making it suitable for deployment in various healthcare settings, including intensive care units and general hospital wards. By combining automated monitoring with intelligent decision-making, the architecture aims to support healthcare professionals, reduce manual workload, and enhance overall patient safety and care quality.[6-7]



**Figure1** The overall architecture of the proposed system



- **Data Acquisition Layer:**

The data acquisition layer is responsible for capturing realtime video input from cameras installed in hospital rooms. The cameras are positioned to ensure clear visibility of the patient's face and upper body. The video stream is transmitted to the processing unit at a consistent frame rate, typically between 15–30 frames per second, to maintain a balance between accuracy and computational efficiency.

- **Preprocessing Layer:**

In this layer, the captured video stream is converted into individual frames. Each frame undergoes preprocessing operations such as resizing, normalization, noise reduction, and contrast enhancement. These steps improve the quality of input data and ensure consistency for the deep learning models. The preprocessing layer also handles frame selection and filtering to reduce redundant data.

- **Detection and Region of Interest (ROI) Extraction:**

This module identifies the patient within the frame and extracts relevant regions such as the face and upper body. Techniques such as Haar Cascade classifiers or YOLO-based object detection are used to locate the human subject. Facial landmark detection is applied to identify key features such as eyes, mouth, and eyebrows, which are essential for analyzing micro-expressions. Pose estimation algorithms are used to track body posture and movement.

- **Feature Extraction Layer:**

The feature extraction layer uses a Convolutional Neural Network (CNN) to extract spatial features from each frame. The CNN processes the input images through multiple convolutional and pooling layers to detect patterns such as facial expressions, eye movements, and body posture. These features are transformed into feature vectors that represent the patient's current state.

- **Temporal Analysis Layer:**

Since distress detection requires understanding changes over time, a Long Short-Term Memory (LSTM) network is used for temporal analysis. The LSTM processes sequences of feature vectors

obtained from consecutive frames. It captures temporal dependencies such as repetitive movements, sudden jerks, or prolonged inactivity, which are critical indicators of distress.

- **Classification Layer:**

The processed features are passed to a classification module that determines the patient's condition. A Softmax classifier is used to assign probability scores to different classes, including normal, mild distress, and critical distress. The classification decision is based on the highest probability value, ensuring reliable prediction.

- **Distress Scoring Module:**

In addition to classification, a distress score is computed by combining multiple behavioral indicators such as facial expression intensity, movement irregularity, and posture deviation. This score provides a quantitative measure of patient discomfort and helps improve decision-making accuracy.

- **Alert and Notification System:**

When the system detects mild or critical distress, an alert is generated automatically. The alert system is integrated with a central monitoring dashboard and can send notifications to healthcare staff through mobile devices or alarm systems. The alerts include relevant details such as patient identification, distress level, and timestamp, enabling quick response.

- **User Interface and Monitoring Dashboard:**

A web-based dashboard is used to display real-time patient status, video feed, and alert notifications. Healthcare professionals can monitor multiple patients simultaneously and review historical data for analysis. The interface is designed to be simple, intuitive, and efficient.

- **System Integration and Workflow:**

All modules are integrated into a unified pipeline that operates continuously. The system processes video input in real time, extracts meaningful features, analyzes temporal behavior, and generates alerts with minimal delay. This ensures timely detection of distress and immediate response from healthcare staff.[8-9]

## 6. System Implementation

The proposed silent distress detection system is



implemented using a combination of computer vision techniques, deep learning models, and a real-time monitoring interface. The implementation focuses on achieving accurate detection of patient distress while maintaining low latency and high system reliability. The system is developed using Python as the primary programming language due to its extensive support for machine learning and image processing libraries. OpenCV is utilized for video capture and frame processing, enabling real-time extraction of frames from live camera feeds. The video input is processed at a controlled frame rate to ensure a balance between computational efficiency and detection performance. For feature extraction, a Convolutional Neural Network (CNN) model is implemented using deep learning frameworks such as TensorFlow or PyTorch. The CNN is trained on facial expression and human activity datasets to recognize patterns related to discomfort, pain, and abnormal movements. Pre-trained models can also be fine-tuned to improve accuracy and reduce training time. To analyze temporal behavior, a Long Short-Term Memory (LSTM) network is integrated with the CNN model. The CNN extracts spatial features from each frame, which are then passed as sequential input to the LSTM. This hybrid CNN-LSTM architecture enables the system to capture both spatial and temporal patterns, allowing accurate detection of behavioral changes over time. The classification module is implemented using a Softmax layer that outputs probability scores for different classes, including normal, mild distress, and critical distress. Based on these probabilities, the system determines the patient's condition and assigns an appropriate distress level. A distress scoring mechanism is also implemented to enhance decision-making. The score is calculated by combining multiple indicators such as facial expression intensity, movement irregularity, and posture deviation. Weighted parameters are used to prioritize more significant indicators of distress. The backend system is designed to handle data processing and communication between modules. A web-based dashboard is developed using frameworks such as Flask or Django to provide a user interface for healthcare professionals. The

dashboard displays real-time video feed, patient status, and alert notifications. The alert system is implemented to generate notifications when abnormal behavior is detected. Alerts can be triggered through visual indicators on the dashboard, audible alarms, or mobile notifications. The system ensures that alerts are generated with minimal delay to enable immediate response. To ensure reliability, the system is tested under different conditions, including variations in lighting, camera angles, and patient movement. Optimization techniques such as frame skipping and model compression are applied to improve performance and reduce computational load. Overall, the implementation of the proposed system provides a scalable, efficient, and real-time solution for detecting silent distress in hospital environments, supporting healthcare professionals in delivering timely and effective care.[9-10]

## 7. Algorithms Used

The proposed Silent Distress Detection System uses a combination of deep learning and machine learning algorithms to analyze human micro-actions such as facial expressions, eye blinking patterns, and body movements. The main algorithms used in this system include Convolutional Neural Networks (CNN), Support Vector Machine (SVM), and MediaPipe Pipeline for facial and body landmark detection. These algorithms work together to extract meaningful behavioral features and classify distress levels accurately. [10-11]

- **Convolutional Neural Network (CNN):**

A Convolutional Neural Network (CNN) is a deep learning algorithm widely used in computer vision applications for image recognition and feature extraction. CNN models are capable of automatically learning spatial features from images without manual feature engineering. In the proposed system, CNN is used to analyze facial expressions captured from video frames. The input image is passed through multiple convolution layers where filters are applied to extract important visual features such as edges, textures, and facial patterns. After the convolution stage, pooling layers reduce the dimensionality of the feature maps while preserving the most important information. Fully connected layers then classify the extracted features to identify emotional or behavioral



states. CNN helps detect subtle facial micro-expressions that may indicate distress, such as eyebrow movement, eye squinting, or facial tension. Typical CNN models used for facial expression recognition include architectures such as VGGNet, ResNet, and MobileNet. Accuracy: CNN-based facial expression recognition systems generally achieve an accuracy of 94% to 97%, depending on the dataset used [11-12]

• **Support Vector Machine (SVM):**

Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification tasks. It is highly effective for separating data into different categories by identifying an optimal decision boundary known as a hyperplane. In the proposed system, SVM is used to classify patient behavior based on features extracted from facial expressions, blinking patterns, and body posture. These features are provided as input to the SVM classifier. The classifier analyzes the feature values and categorizes the patient's condition into different distress levels. SVM is chosen because it performs well with highdimensional data and provides reliable classification results even with relatively small datasets. Accuracy: SVM classifiers typically achieve accuracy between 92% and 95% in behavioral classification tasks.

• **MediaPipe Pipeline:**

MediaPipe is an open-source framework developed by Google for building real-time computer vision applications. It provides efficient pipelines for detecting facial landmarks, hand movements, and body posture. In this project, MediaPipe is used for facial landmark detection and pose estimation. The MediaPipe Face Mesh model identifies multiple facial landmarks such as eye corners, eyebrows, nose, and mouth. These landmarks are used to detect micro-actions including:

- Eye blinking
- Facial muscle movement
- Head movement
- Body posture changes

MediaPipe Pose estimation is also used to track body joints such as shoulders, arms, and neck. This allows the system to detect abnormal body movement patterns that may indicate discomfort or distress. The

MediaPipe pipeline processes video frames in real time and provides highly accurate landmark detection with low computational cost. Accuracy: MediaPipe landmark detection systems typically achieve accuracy between 93% and 96%.

**8. Result And Discussion**

The proposed silent distress detection system was evaluated using a combination of simulated patient scenarios and publicly available datasets. The objective of the evaluation was to analyze the system's ability to accurately detect distress-related micro-actions and classify patient conditions in real time. The system successfully identified key behavioral indicators such as facial tension, abnormal eye movements, sudden body movements, and prolonged inactivity. These features were effectively extracted using the Convolutional Neural Network (CNN) and further analyzed using the Long Short-Term Memory (LSTM) model to capture temporal dependencies. The integration of spatial and temporal analysis significantly improved the overall performance of the system. The performance of the model was evaluated using standard metrics such as accuracy, precision, recall, and F1score. As shown in Table I, the system achieved high accuracy in detecting distress conditions, indicating its effectiveness in real-time healthcare monitoring. The results demonstrate that the model maintains a good balance between precision and recall, ensuring reliable predictions.

**Table 1 Performance Evaluation of the Proposed Model**

Metric	Value (%)
Accuracy	92.4
Precision	90.8
Recall	91.6
F1-Score	91.2

In addition, the classification performance of the system across different patient conditions is presented in Table II. The system showed strong performance in detecting critical distress cases, which is essential for timely medical intervention. The number of correct predictions across all categories indicates the robustness of the proposed model.



**Table 2 Distress Detection Results**

Condition	Total Samples	Correct Predictions
Normal	120	112
Mild Distress	95	88
Critical Distress	60	56

The use of LSTM for temporal analysis played a crucial role in reducing false positives by considering behavioral patterns over time rather than relying on individual frames. This enabled the system to distinguish between normal movements and actual distress-related actions more effectively. However, certain limitations were observed during testing. Variations in lighting conditions and camera angles affected the consistency of feature extraction. Occlusion of facial regions due to medical equipment or blankets also reduced detection accuracy in some cases. Additionally, differences in patient behavior introduced variability in the results. Despite these challenges, the system demonstrated strong potential for real-world deployment. The real-time alert generation mechanism ensured that healthcare staff were promptly notified when abnormal behavior was detected. The non-invasive nature of the system makes it suitable for continuous monitoring without causing discomfort to patients. [12]

### Conclusion

This paper presented an intelligent system for detecting silent distress in hospital environments using unintentional human micro-actions. The proposed approach integrates computer vision and deep learning techniques to analyze subtle behavioral patterns such as facial expressions, eye movements, and body posture in real time. The system demonstrated effective performance in identifying different levels of patient distress, including normal, mild distress, and critical distress. By combining spatial feature extraction using Convolutional Neural Networks (CNN) with temporal analysis through Long Short Term Memory (LSTM) models, the system achieved reliable and accurate classification results. The non-invasive nature of the system makes it suitable for continuous monitoring without causing

discomfort to patients. Additionally, the real-time alert mechanism ensures timely response from healthcare professionals, thereby improving patient safety and reducing the risk of unnoticed critical conditions. Although certain challenges such as lighting variations and occlusions were observed, the overall performance of the system indicates strong potential for real-world deployment. The proposed solution contributes to the advancement of intelligent healthcare systems by enabling proactive and automated patient monitoring. In conclusion, the developed system provides an efficient, scalable, and practical solution for silent distress detection, supporting healthcare professionals in delivering timely and improved patient care.

### Future Work

Future work for the proposed silent distress detection system can focus on enhancing accuracy, scalability, and real-world applicability. One important direction is the integration of multimodal data sources, such as audio signals and physiological parameters (e.g., heart rate and oxygen levels), to improve the reliability of distress detection. The use of advanced deep learning architectures, such as attention-based models and transformer networks, can further improve the system's ability to capture complex behavioral patterns. Additionally, expanding the training dataset with diverse patient conditions and real hospital scenarios will help improve model generalization and robustness. Improving system performance under challenging conditions, such as low lighting, occlusions, and varying camera angles, is another key area for development. Techniques such as data augmentation and adaptive preprocessing can be explored to address these issues. Future enhancements may also include the development of a fully integrated mobile or IoT-based monitoring system, enabling remote patient monitoring and real-time notifications to healthcare providers. Furthermore, deploying the system in real clinical environments and conducting extensive validation studies will be essential to evaluate its effectiveness and usability. Overall, these improvements will contribute to making the system more reliable, scalable, and suitable for widespread adoption in modern healthcare systems.



### Acknowledgment

The authors would like to express their sincere gratitude to Ms. R. Niranjana, Assistant Professor, Department of Computer Science, Paavai Engineering College, Namakkal, for her valuable guidance, encouragement, and continuous support throughout the development of this project and research work. Her insightful suggestions and technical expertise played an important role in the successful completion of this work. The authors also extend their heartfelt thanks to Mr. D. V. Rajkumar, Assistant Professor and Project Coordinator, Department of Computer Science, Paavai Engineering College, Namakkal, for his constant support, guidance, and motivation during the progress of this project. The authors are grateful to the Department of Computer Science, Paavai Engineering College, for providing the necessary facilities and resources to carry out this research successfully. Finally, the authors would like to thank their friends and family members for their encouragement and support throughout the project.

### References

- [1]. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 511–518, 2001.
- [2]. P. Michel and R. El Kaliouby, "Real-time facial expression recognition in video using support vector machines," Proc. 5th Int. Conf. Multimodal Interfaces, Vancouver, Canada, pp. 258–264, 2003.
- [3]. M. Valstar and M. Pantic, "Fully automatic recognition of the temporal phases of facial actions," IEEE Trans. Systems, Man, and Cybernetics, vol. 42, no. 1, pp. 28–43, 2012.
- [4]. V. Bettadapura, "Face expression recognition and analysis: The state of the art," arXiv preprint arXiv:1203.6722, 2012.
- [5]. A. Krizhevsky, I. Sutskever and G. Hinton, "ImageNet classification with deep convolutional neural networks," Advances in Neural Information

- Processing Systems (NIPS), pp. 1097–1105, 2012.
- [6]. J. Kumari and K. Rajesh, "Facial expression recognition: A survey," Procedia Computer Science, vol. 58, pp. 486–491, 2015.
- [7]. S. Li and W. Deng, "Facial expression recognition: Recent advances and applications," IET Image Processing, vol. 13, no. 11, pp. 2021–2044, 2019.
- [8]. F. Schroff, D. Kalenichenko and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2015.
- [9]. C. Lugaresi et al., "MediaPipe: A framework for building perception pipelines," arXiv preprint arXiv:1906.08172, 2019.
- [10]. N. Zainuddin, A. M. Zain and H. Haron, "Patient monitoring system using computer vision for emotional recognition and vital signs detection," Proc. IEEE Int. Conf. Smart Healthcare, pp. 45–50, 2020.
- [11]. S. Rezaei et al., "Unobtrusive pain monitoring in older adults using computer vision," IEEE Access, vol. 9, pp. 45678–45688, 2021.
- [12]. D. Germanese et al., "Computer vision tasks for ambient intelligence in healthcare monitoring," Information, vol. 14, no. 10, pp. 548–560, 2023.