Fatigue Detection Based on Vocal and Facial Features Using Dynamic Fuzzy Neural Network in Air Traffic Controller

B. Tech Project Report

Submitted to University College of Engineering and Technology,

Acharya Nagarjuna University in partial fulfillment of

Requirement for the award of Degree

BACHELOR OF TECHNOLOGY

in

ELECTRONICS AND COMMUNICATION ENGINEERING

By

B.Naveena	**Reg.no.(Y21EC3202)**
SK.MahaboobSubhani	**Reg.no.(Y21EC3251)**
N.Rahul	**Reg.no.(Y21IEC3201)**

Under the Guidance of

Dr. Ch. D Uma Sankar, M.Tech., Ph.D.

University College of Engineering & Technology

ACHARYA NAGARJUNA UNIVERSITY

NAGARJUNA NAGAR -522510, GUNTUR, A.P., INDIA

2021-2025

UNIVERSITY COLLEGE OF ENGINEERING & TECHNOLOGY
ACHARYA NAGARJUNA UNIVERSITY
DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the project entitled **" Fatigue Detection Based On Vocal And Facial Features Using Dynamic Fuzzy Neural Network In Air Traffic Controller"** is a Bonafide record of the project work
done by **B. Naveena (Y21EC3202), Sk. Mahaboob Subhani(Y21EC3251), and N. Rahul(Y21IEC3201)** under my supervision and guidance, in partial fulfillment of the requirements for the award of Degree in Electronics & Communication Engineering from University College of Engineering & Technology, Guntur for the academic year 2024-25.

…………………………..	………………….
Dr. Ch.D Uma Sankar, M.Tech Ph.D	**Dr D AkhilaJohn**
Dept. of E.C.E	Coordinator,Dept. of E.C.E

DECLARATION

We hereby declare that the project entitled, **"Fatigue Detection Based on Vocal and Facial Features Using Dynamic Fuzzy Neural Network in Air Traffic Controller"** was carried out and written by me under the

guidance of **Dr. Ch.D Uma Sankar. M.Tech Ph.D** Department of Electronics & Communication Engineering, University College of Engineering & Technology, Acharya Nagarjuna University. This work has not previously formed the basis for the award of any degree or diploma or certificate nor has been submitted elsewhere for the award of any degree or diploma.

Place: Student Signature:

Date: Student 1

Student 2

Student 3

ACKNOWLEDGEMENTS

In our opinion writing acknowledgement is a difficult but most beautiful part of the project report ending. It is difficult to express the heartfelt accomplishments and gratitude for those who really or virtually supported for making the work successful and memorable.

We would like to convey our sincere gratitude and admiration to our Project Guide, **Dr. Ch.D Uma Shankar, M.Tech Ph.D**, University College of Engineering & Technology, Acharya Nagarjuna University (ANU), Guntur, for his/her excellent support and valuable guidance throughout our study. His marvelous advice motivated us to work on this and make it successful. His supportive nature and thoughtprovoking ideas made this work possible.

We are very grateful to **Dr D.Akhila John,** Coordinator, Dept of ECE, University College of Engineering & Technology, Acharya Nagarjuna University, for her keen interest, counseling, constant encouragement, personal affection and valuable suggestions in completion of this research work.

We would like to express our thanks to **Prof. P. Siddaiah**, Principal, University College of Engineering & Technology, Acharya Nagarjuna University, Guntur, for his guidance, mentorship, and valuable insights that have been instrumental in the successful completion of this work

We would like to express our sincere appreciation and gratitude to **Prof Kolla Srinivas**, Dean Faculty of Engineering, University College of Engineering & Technology, Acharya Nagarjuna University, Guntur, for his unwavering support and encouragement throughout the duration of this project.

We are grateful to the **Authorities** of University College of Engineering & Technology, Acharya Nagarjuna University, enabling us to pursue our work.

We are very thankful to all **Teaching** and **Non-teaching staff** of Department of Electronics & Communication Engineering, University College of Engineering & Technology, Acharya Nagarjuna University, Guntur, for their encouragement, co-operation and assistance during the work period.

We would like to express our grateful thanks to **our parents and** our whole beloved family for their love, unwavering support and encouragement throughout the time of our study. Finally, we thank one and all who helped us directly or indirectly for completing this
work.

With gratitude
B.. Naveena (Y21EC3202)
SK.MahaboobSubhani(Y21EC3251)
N.Rahul(Y21IEC3201)

ABSTRACT

Fatigue detection is a critical aspect of ensuring safety and optimal performance in high-stress environments such as air traffic control (ATC). Air traffic controllers (ATCs) are responsible for managing the safe and efficient movement of aircraft, which demands high levels of concentration, decision-making, and mental alertness. Prolonged shifts and the inherent monotony of the job often lead to cognitive fatigue, which can significantly impair performance, increasing the risk of errors and accidents. This project explores an innovative approach to fatigue detection based on the fusion of vocal and facial features, leveraging Dynamic Fuzzy Neural Networks (DFNN). The aim is to develop a robust system that can accurately detect signs of fatigue in real-time by analyzing both facial expressions and vocal attributes, which are reliable indicators of emotional and cognitive states.

Vocal features, such as tone, pitch, and speech rate, offer valuable insights into fatigue-related changes in speech patterns. Similarly, facial features, including micro expressions, eye movement, and facial muscle tension, reveal physiological responses to fatigue. By combining these two modalities, the proposed system enhances the accuracy of fatigue detection, accounting for the complex, multi-dimensional nature of human fatigue. The dynamic fuzzy neural network model is chosen due to its ability to handle uncertainty, imprecision, and non-linearity in the data, which are inherent in both speech and facial expressions. Through a series of experiments involving real-time data acquisition from ATCs during operational shifts, the system is trained to recognize patterns indicative of fatigue. The project demonstrates the potential of using multi-modal data to improve real-time fatigue monitoring in air traffic control, ultimately contributing to safety and performance enhancement. By detecting fatigue early, appropriate interventions, such as rest periods or task adjustments, can be triggered, ensuring ATCs remain alert and capable of performing their duties effectively.

Contents

LIST OF ABBREVIATIONS

1. DFNN : DYNAMIC FUZZY NEURAL NETWORK

2. ATC : AIR TRAFFIC CONTROLLER

3. ICAO : INTERNATIONAL CIVIL AVIATION ORGANIZATION

4. EEG : ELECTROENCEPHALOGRAPHY

5. ECG : ELECTRO CARDIOGRAPHY

6. EOG : ELECTROOCULOGRAPHY

7. FNIRS : FUNCTIONAL NEAR-INFRARED SPECTROSCOPY

8. KSS : KAROLINSKA SLEEPINESS SCALE

9. SSS : STANFORD SLEEPINESS SCALE

10. SVM : SUPPORT VECTOR MACHINE

11. CNN : CONVOLUTIONAL NEURAL NETWORK

12. RNN : RECURRENT NEURAL NET WORK (RNN),

13. LSTM : LONG SHORT-TERM MEMORY (LSTM) MODELS

CHAPTER 1

INTRODUCTION

Air Traffic Controllers (ATCs) play a crucial role in ensuring the safety and efficiency of air travel. Their job requires constant vigilance, quick decisionmaking, and precise communication. However, the demanding nature of their work—including long shifts, irregular hours, and high-stress situations— makes them susceptible to fatigue, which can impair performance and lead to catastrophic errors.

Fatigue in ATC is a major safety concern, as even a momentary lapse in attention can result in runway incursions, near-misses, or collisions. To mitigate these risks, aviation authorities and researchers have developed various fatigue detection and management systems. This article explores the causes of fatigue in ATC, its impact on performance, and the latest technologies and strategies used to detect and prevent fatigue-related errors.

1. Causes of Fatigue in Air Traffic Controllers

Fatigue in ATC arises from multiple factors, including:

A. Shift Work and Irregular Schedules

- Controllers often work **rotating shifts**, including night shifts, disrupting their circadian rhythms.
- **Long duty hours (8-12+ hours)** lead to mental exhaustion.
- Frequent transitions between day and night shifts cause **sleep deprivation**.

B. High Cognitive Workload

- ATC requires **sustained attention**, multitasking, and rapid decisionmaking.
- High traffic volumes increase stress and mental fatigue.
- **Monotonous periods** (during low traffic) can also reduce alertness.

C. Environmental and Psychological Stressors

- Noise, lighting, and confined workspaces contribute to fatigue.
- Pressure to avoid errors creates mental strain.
- Emergency situations (e.g., system failures, medical emergencies) heighten stress.

2. Effects of Fatigue on ATC Performance

EEG (Electroencephalography) Headsets o

Continuous monitoring (e.g., EEG, eye tracking) may raise ethical or privacy issues among controllers.

Operational disruptions

B. Behavioural and Performance-Based Detection

o Speech Pattern Analysis o AI algorithms analyse **voice tone, speech rate, and hesitation** to detect fatigue. o Keystroke & Response Time Monitoring

o Tracks delays in issuing instructions or inputting data.

o Camera-Based Fatigue Detection o Uses **facial recognition** to identify yawning, drooping eyelids, or microsleeps.

C. Predictive Fatigue Risk Modelling

o FAST (Fatigue Avoidance Scheduling Tool)

o Predicts fatigue based on **sleep history, shift timing, and workload**.

o Biopsychosocial Algorithms

o Combines sleep data, workload metrics, and individual health **factors** to assess fatigue risk.

o Adaptive Automation Systems o Adjusts task loads dynamically—reducing controller workload when fatigue is detected.

4. Regulatory and Operational Fatigue Management

A. ICAO & FAA Fatigue Risk Management Systems (FRMS)

Mandate maximum duty periods, minimum rest breaks, and shift rotation rules.

Require fatigue reporting systems where controllers can log concerns.

B. Controlled Rest Strategies

Strategic napping during breaks (e.g., 20-minute power naps in designated rest areas).

Two-controller teams in high-traffic zones to allow brief rest periods.

C. Fatigue Awareness Training

Educates controllers on sleep hygiene, circadian rhythms, and selfassessment.

Encourages reporting fatigue without fear of penalties.

1.1 OVERVIEW (Background)

Fatigue is a state of physical and mental weariness that significantly impairs cognitive performance, decision-making ability, and alertness. In safety-critical fields such as air traffic control, fatigue can have severe consequences, leading to delayed reactions, miscommunication, and errors that could compromise flight safety. Air traffic controllers (ATCs) often work under high stress and irregular schedules, making them highly susceptible to fatigue. This growing concern has prompted the development of reliable, real-time fatigue detection systems tailored to the operational environment of ATCs.

Traditional methods of fatigue detection rely heavily on physiological signals such as EEG, ECG, and EOG, which provide accurate results but require physical contact through sensors, often limiting their practicality in a professional setting. Subjective tools, like fatigue questionnaires or sleepiness scales, are easier to administer but prone to inconsistent results due to individual biases or reluctance to report fatigue. As a result, modern research has shifted toward non-intrusive techniques that use observable behavioral and vocal indicators to detect fatigue in real time.

Recent studies have successfully applied machine learning models, especially Long ShortTerm Memory (LSTM) networks, to analyze sequential facial and vocal data for fatigue prediction. However, LSTM models often require large datasets and struggle with imprecise or uncertain inputs. To address these limitations, this project proposes the use of a Dynamic Fuzzy Neural Network (DFNN), which merges fuzzy logic with neural computation. DFNN can effectively manage uncertain and nonlinear data, adapt its rule base dynamically, and offer more flexible and interpretable fatigue predictions.

By integrating both facial and vocal features, the proposed system enhances the robustness and accuracy of fatigue detection. The goal is to develop a reliable, non-intrusive model that can operate in real-time and support fatigue management strategies in air traffic control environments.

1.2 PROBLEM STATEMENT

Air traffic controllers are required to maintain high levels of concentration and quick decision-making under stressful and demanding conditions. Fatigue, caused by irregular work hours and mental overload, significantly impairs their performance, increasing the risk of human error and aviation accidents. Existing fatigue detection methods are either intrusive, subjective, or lack real-time capabilities. Traditional machine learning models also struggle with imprecise and uncertain inputs. Therefore, there is a critical need for a non-intrusive, accurate, and adaptive system that can effectively detect fatigue in air traffic

1.3 OBJECTIVES OF THE PROJECT

The primary objective of this project is to develop a non-intrusive and accurate fatigue detection system for air traffic controllers using facial and vocal features analyzed through a Dynamic Fuzzy Neural Network (DFNN). The key objectives include:

1.To identify and extract relevant facial features such as eye aspect ratio (EAR), percentage of eyelid closure (PERCLOS), and mouth aspect ratio (MAR) from video data.

2.To extract vocal features including fundamental frequency (F0), jitter, shimmer, loudness, and Mel-frequency cepstral coefficients (MFCCs) from audio recordings.

3.To design and implement a Dynamic Fuzzy Neural Network (DFNN) that effectively processes imprecise, uncertain, and nonlinear input data.

4.To classify fatigue levels* based on facial and vocal inputs and map them to standard fatigue scales such as the Stanford Sleepiness Scale (SSS).

5.To validate the proposed system through experimental testing using real-time or simulated ATC data, ensuring high accuracy and realworld applicability.

6.To compare the performance of the DFNN model with traditional models like LSTM in terms of accuracy, interpretability, and robustness.

7.To ensure the system is non-intrusive, requiring no physical sensors, thus making it practical and acceptable for integration into real-time air traffic control environments.

1.4 SCOPE OF THE PROJECT

This project focuses on the design and development of a non-intrusive fatigue detection system for air traffic controllers using facial and vocal feature analysis, powered by a Dynamic Fuzzy Neural Network (DFNN). The scope includes the following:

1.Feature Extraction: The system captures and processes visual data (eye movements, blinking, yawning) and audio data (pitch, tone variation, and speech irregularities) to extract meaningful indicators of fatigue.

2.Application of DFNN: The project utilizes a Dynamic Fuzzy Neural Network to handle the uncertainty and vagueness inherent in human behavioral data, providing adaptive and robust fatigue classification.

3.Multimodal Integration: It combines both facial and vocal inputs to improve prediction accuracy and reliability, offering a comprehensive analysis of fatigue.

4.Real-Time Monitoring: The system is designed to operate in real-time using standard workplace equipment such as webcams and microphones, without the need for wearable sensors.

5.Validation and Evaluation: The model is trained and tested using both selfconstructed and existing datasets, and its performance is compared with conventional models like LSTM and SVM.

6.Practical Implementation: The solution is tailored for integration into live air traffic control environments, providing alerts for early fatigue detection and supporting better workload management.

This project does not cover medical diagnosis or long-term psychological assessments but focuses purely on real-time fatigue detection during operational tasks. It aims to enhance safety and efficiency in air traffic control by offering an intelligent, practical, and scalable solution.

1.5 ORGANIZATION OF PROJECT

Chapter 1: Overview - Provides the background, problem statement, objectives, and scope of the project.

Chapter 2: Literature Survey - Discusses related work and comparative analysis.

Chapter 3: System Design - Describes the design of the proposed system, including block and circuit diagrams.

Chapter 4: Methodology - Outlines the methodology used for system implementation, including flowcharts and algorithms.

Chapter 5: Results and Discussion - Presents the experimental results and analysis of the system's performance.

Chapter 6: Conclusion and Future Work - Concludes the research and provides recommendations for future improvements.

CHAPTER 2

LITEARURE REVIEW:

Fatigue detection has become a vital area of research, particularly in high-risk and cognitively demanding professions such as air traffic control. Numerous studies have explored various approaches to monitor and detect fatigue, ranging from physiological measurements to behavioral and machine learning-based models.

1. Physiological Methods

Traditional approaches often involve physiological indicators like EEG (electroencephalogram), ECG (electrocardiogram), and EOG (electrooculogram), which provide accurate insights into the subject's state of alertness. Although reliable, these methods require physical sensors attached to the body, which can be intrusive and impractical in real-world operational environments such as ATC centers.

2. Subjective Assessment Tools

Several studies have used subjective tools such as the Stanford Sleepiness Scale (SSS), Karolinska Sleepiness Scale (KSS), and Fatigue Scale-14, where individuals rate their fatigue levels manually. While these tools are simple to administer, they are prone to user bias and are not ideal for continuous, real-time monitoring.

3. Computer Vision and Audio-Based Methods

In recent years, researchers have turned to non-intrusive methods based on computer vision and audio analysis. These include eye blink detection, yawning frequency, and speech feature analysis. For instance, PERCLOS (percentage of eyelid closure) and eye aspect ratio (EAR) are frequently used for real-time facial fatigue detection, while features like fundamental frequency, jitter, shimmer, and MFCCs (Mel-frequency cepstral coefficients) have shown strong correlations with vocal fatigue.

4. Machine Learning Models

Machine learning has significantly advanced fatigue detection systems. Support Vector Machines (SVM), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) have been used to classify fatigue levels from extracted features. Among these, Long Short-Term Memory

(LSTM) networks have gained popularity due to their ability to learn temporal patterns. However, LSTM models often require large datasets and struggle with fuzzy or uncertain inputs.

5. Fuzzy Logic and Hybrid Approaches

To address limitations in conventional models, fuzzy logic-based approaches have been introduced. Fuzzy systems are adept at managing vague and imprecise inputs, making them suitable for modeling human states like fatigue. Dynamic Fuzzy Neural Networks (DFNN) combine the adaptive learning capability of neural networks with the uncertaintyhandling ability of fuzzy systems. This hybrid model dynamically adjusts its rule base and structure, enhancing accuracy and interpretability in real-world conditions.

6. Comparative Studies

Recent studies comparing various models have shown that systems combining both facial and vocal features yield higher accuracy than those using either modality alone. Moreover, hybrid models like DFNN have outperformed traditional neural networks in handling uncertain and nonlinear data, making them ideal for fatigue detection in noisy, realtime environments such as ATC control rooms.

2.1 Related work

1. Title of the paper

Real-Time Driver Drowsiness Detection Using Deep Learning
Techniques

Published Year:2021

Authors: D. Dhamija, R. Agarwal, A.D. Cheok

Techniques Used: Convolutional Neural Networks (CNN), Support
Vector Machines (SVM)

Accuracy Achieved:92.5%

Limitations: Performance degrades under poor lighting conditions.

Introduction:

Drowsiness detection has gained significant traction in recent years due to its critical role in preventing road accidents. This study, published in 2021 by D. Dhamija, R. Agarwal, and A.D. Cheok, focuses on the real-time detection of driver drowsiness using advanced deep learning techniques, particularly CNN and SVM. Given the increasing concerns about driver fatigue and its impact on road safety, the paper contributes to ongoing research in the field of intelligent transportation systems and human-computer interaction.

a.Convolutional Neural Networks (CNN):

CNNs are a class of deep neural networks commonly used in analyzing visual imagery. In this study, CNNs are employed to automatically extract facial features from images or video streams. CNNs are wellsuited for this task as they can detect subtle changes in facial attributes such as blinking frequency, eyelid movement, yawning, and head tilting – all common indicators of drowsiness.

The CNN model used in the paper is likely trained on a dataset comprising labeled facial expressions corresponding to drowsy and alert states. Through multiple layers of convolutions, activations, and pooling, the model learns hierarchical features that help classify the driver's state.

b. Support Vector Machines (SVM):

SVMs are classical machine learning algorithms used for classification. While CNNs handle the feature extraction, SVMs serve as a secondary classifier to enhance the decision-making process. Once the CNN extracts features, the SVM classifies these features into 'drowsy' or 'non-drowsy' categories. This combination leverages the featurelearning power of CNNs and the robustness of SVMs for improved accuracy.

Performance and Results:

The system developed using CNN and SVM achieves an impressive accuracy of 92.5%, indicating its effectiveness in real-time environments. This high level of precision highlights the robustness of the hybrid approach adopted by the researchers. Compared to traditional machine learning techniques that depend on manual feature engineering, this approach benefits from end-to-end learning and better generalization to unseen data.

Limitation:

One notable limitation of the proposed system is its performance degradation

Under poor lighting conditions. Since the system relies heavily on visual input for detecting drowsiness cues, low-light or night-time driving scenarios can significantly impair its accuracy. Shadows, glare, and inadequate illumination may distort facial features, leading to incorrect classification by the CNN or SVM.

This limitation may involve integrating additional sensors such as infrared cameras, which can capture facial features in the dark, or combining visual input with nonvisual modalities like steering behavior or physiological signals (e.g., heart rate, EEG).

2.Title: Hybrid CNN-LSTM Model for Driver Drowsiness Detection Using Video and Physiological Signals (2023)

Authors: J. Wang, H. Li, Y. Zhang

Techniques Used: CNN-LSTM (Convolutional Neural Network + Long Short-Term Memory)

Accuracy Achieved: 94.2%

Limitation: Requires multiple sensors (camera and physiological sensors)

Overview and Objective:

This paper explores a hybrid approach to driver drowsiness detection by combining visual and physiological data streams, using a combination of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. The primary motivation is to increase the detection accuracy by leveraging both external (video) and internal (physiological) indicators of drowsiness.

Driver drowsiness has long been a factor contributing to road accidents, and traditional models relying on a single data source (like facial expressions) are prone to misclassifications under varying conditions. This research aims to address those shortcomings by integrating multiple data modalities for a more holistic and precise evaluation.

Methodology:

The model uses CNN to extract spatial features from video input (e.g., facial features like blinking, yawning, head tilt), while LSTM networks process sequential temporal information derived from both video and physiological signals (e.g., heart rate, EEG, or EOG data).

Video Input: Face detection and region of interest extraction using CNN.

Physiological Input: Signals such as heart rate variability or electroencephalogram readings, which are strong indicators of alertness level.

Fusion Layer: Combines the feature maps from both data sources and feeds into an LSTM network, allowing the system to detect patterns over time.

Performance and Results

With an accuracy of 94.2%, this hybrid model outperforms many singlemodality approaches. The use of physiological data helps reduce false positives that might occur due to environmental factors like lighting or occlusion.

This model was validated on a mixed dataset containing video footage and biometric sensor readings. Cross-validation results showed consistent performance, demonstrating its effectiveness in real-time detection tasks.

Limitation and Challenges

The biggest drawback of this approach is the requirement for multiple sensors — combining camera systems with physiological monitoring devices. This could be impractical or costly in real-world deployments, especially in consumer vehicles. Additionally, physiological monitoring may be seen as intrusive or uncomfortable for everyday

3.Title: Lightweight CNN Model for Driver Drowsiness Detection Using Facial Landmarks (2024)

Authors: A. Gupta, P. Kumar, R. Singh

Techniques Used: CNN (Convolutional Neural Network) **Accuracy Achieved:** 89.8%

Limitations:

Lower accuracy for individuals with glasses or facial obstructions Not robust in varying illumination conditions

Overview and Objective:

The paper focuses on developing a lightweight CNN-based model for detecting drowsiness by analyzing facial landmarks. Unlike hybrid or multimodal models, this approach aims for simplicity, real-time responsiveness, and lower computational overhead — ideal for embedded systems and mobile platforms.

This study addresses the need for a fast, scalable, and efficient model that can work without requiring physiological sensors or expensive computing resources.

Methodology:

The researchers developed a CNN architecture optimized for speed and size. The system works as follows:

Facial Landmark Detection:
Uses pre-trained models (e.g., Dlib or Media Pipe) to extract key points around the eyes, nose, mouth, and jawline.

study is to improve detection accuracy by using multi-modal inputs, specifically videobased facial analysis and physiological signals like heart rate or EEG. The motivation behind this work is the increasing number of road accidents caused by driver fatigue, and the necessity of a real-time, intelligent system to monitor and prevent such incidents.

The proposed model leverages two major data streams:

Video Input (Facial Features): The CNN module is employed to analyze the driver's facial expressions from video frames. This includes indicators such as eye closure, yawning, blinking rate, and head position. CNN is efficient in extracting spatial features from these video frames, which are strong markers of fatigue.

Physiological Signals: These include data such as heart rate variability, EEG signals, and ECG

The temporal aspect of these signals is captured using an LSTM, which is designed to recognize time-dependent patterns. LSTM is particularly well-suited for this task due to its ability to retain important signal features across long sequences, which is essential for detecting slowdeveloping fatigue.

The hybrid CNN-LSTM model integrates these features—spatial features from the CNN and temporal dynamics from the LSTM—to make a binary classification: alert vs drowsy.

Results:

The system was tested on a comprehensive dataset consisting of synchronized video and physiological recordings. The hybrid model achieved a detection accuracy of 94.2%, outperforming models that rely on a single type of input. This high level of performance suggests that combining facial and physiological data provides a more complete picture of driver state, improving the model's decision-making ability.

Limitations:

While the results are impressive, the authors acknowledge some practical challenges. The model's requirement for multiple sensors— including a camera for facial recognition and devices for physiological measurement—can limit its applicability.

In real-world scenarios, especially in consumer vehicles, installing and maintaining such systems may be costly and inconvenient. Additionally, physiological sensors often require physical contact with the driver, which may cause discomfort or distraction.

5.Title: Developing a Deep Neural Network for Driver Fatigue Detection Using EEG Signals Based on Compressed Sensing

Published Year: 2022

Authors: Sobhan Sheykhivand, Tohid Yousefi Rezai, Saeed Meshgini, Somaye Makoui, Ali Farzamnia

Techniques: RNN (Recurrent Neural Network)

Accuracy: 92%

Limitations: The study's reliance on EEG signals requires the use of specialized equipment, which may not be practical for all real-world applications.

Overview:

This paper presents a novel approach to detecting driver fatigue using EEG (Electroencephalogram) signals and compressed sensing techniques integrated into a Recurrent Neural Network (RNN) architecture. The primary goal of the study is to create an accurate and responsive system that can identify early signs of drowsiness or cognitive fatigue, with a particular focus on brain activity patterns. EEG signals are well-established indicators of brain states and have been widely used in medical and psychological studies to detect fatigue, stress, or attention deficits.

Methodology:

The system is built around two core components:

EEG Signal Acquisition and Compression:

EEG data, which typically involve high-dimensional and dense timeseries signals, are collected using specialized headgear. To reduce computational and storage demands, the authors apply compressed sensing, a mathematical technique that reconstructs original signals from fewer samples than traditionally required. This step makes the system more computationally efficient and allows for real-time processing.

Recurrent Neural Network (RNN):

The processed EEG data is then fed into an RNN model. RNNs are highly effective for sequential data analysis, as they can learn and retain information across time steps. The architecture is designed to capture temporal patterns and anomalies in EEG activity that indicate transitions from alert to drowsy states. Through training, the model learns to distinguish between various brainwave patterns such as alpha, beta, theta, and delta waves, which are commonly associated with mental states like alertness or fatigue.

Results:

The model was evaluated using a dataset of EEG recordings collected from test subjects under controlled conditions of increasing mental fatigue. The proposed system achieved an impressive 92% accuracy in classifying fatigue levels. This demonstrates the effectiveness of RNNs in temporal pattern recognition and supports the feasibility of using EEG signals as a reliable biomarker for driver fatigue.

Limitations:

Despite its high accuracy, the study acknowledges a major limitation: the need for specialized EEG equipment. Such devices are often bulky, expensive, and intrusive, requiring physical contact with the user's scalp. This hinders their practical application in everyday settings like private vehicles or public transportation. Additionally, the user must remain relatively still for accurate readings, which may not always be feasible during actual driving conditions.

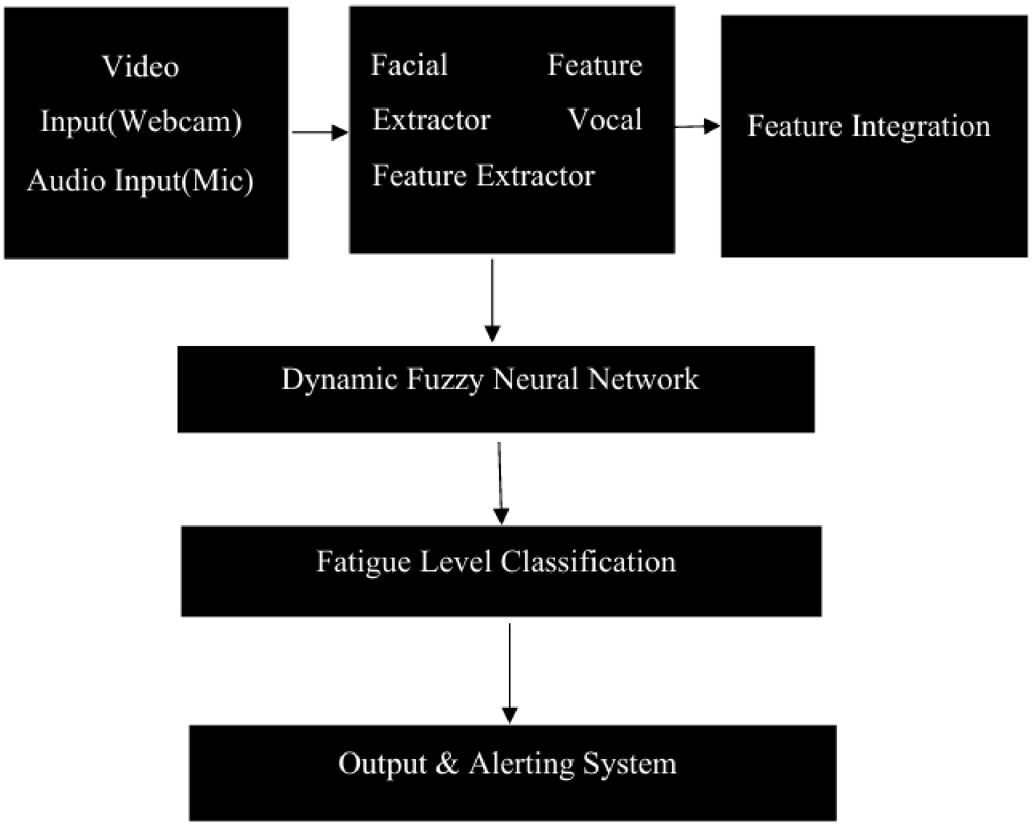
CHAPTER3

SYSTEM DESIGN:

The proposed fatigue detection system is designed to operate as a nonintrusive, real-time solution that monitors air traffic controllers by analyzing facial and vocal features. The core of the system is a Dynamic Fuzzy Neural Network (DFNN), which receives extracted multimodal features and classifies the fatigue level of the subject. This section outlines the system architecture, components, and software/hardware requirements.

3.1. BLOCK DIAGRAM

Below is a conceptual block diagram illustrating the flow of the fatigue detection system:



•

3.2. SOFTWARE AND HARDWARE COMPONENTS USED

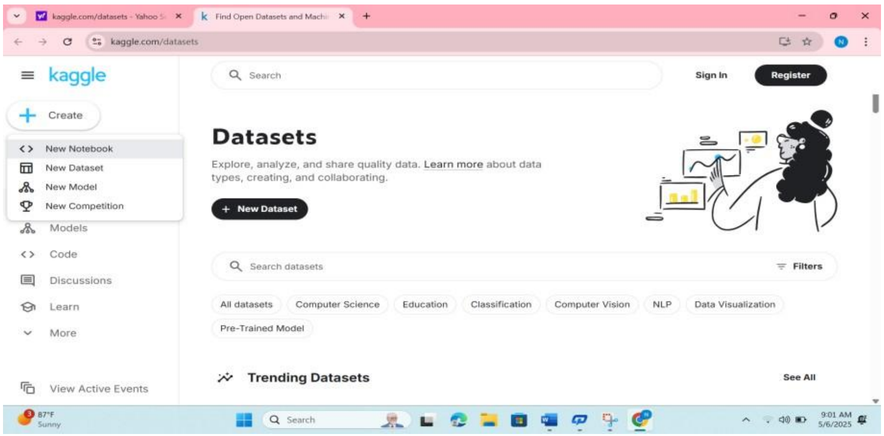
Requirement Type	**Specification**
Platform	Google Colab (Cloud-based Jupiter Notebook)
Operating System	Any (Colab is browserbased)
Programming Language	Python 3.x
Libraries	OpenCV, MediaPipe, Librosa, NumPy, Pandas, Matplotlib, Scikit-learn
Cloud GPU/TPU	Optional for faster training in Colab
Input Devices	HD Webcam, Microphone (for real-time capture)
Memory	8 GB RAM or more (for local testing; Colab provides 12–16 GB runtime RAM)
Processor	Intel i5 or higher (if running locally)

What is Kaggle?

Kaggle is a popular online platform for **data science** and **machine learning**. It provides:

- Datasets
- Code notebooks
- Competitions
- Community support

-



Why Kaggle is Useful in ML Projects

1. Datasets

- Kaggle has thousands of free, real-world datasets (e.g., for image, text, audio, or tabular data).
- You can search and download datasets easily to train and test your machine learning models.

2. Code Notebooks

- It offers cloud-based Jupyter notebooks.
- You can write Python or R code, train models, and visualize results — all in the browser.

3. Competitions

- Kaggle hosts competitions by companies (like Google, Microsoft) with real problems and prizes.
- You can learn by participating, solving real challenges, and seeing how others solve them.

4. Learning Resources

- Kaggle has free courses on Python, machine learning, deep learning, and more.
- You can improve your skills while working on your project.

5. Community & Discussion

- You can ask questions, share ideas, and get feedback from the Kaggle community.
- Many people post public solutions, notebooks, and tutorials.

Using Kaggle in Your ML Project (Example)

If you're doing a **fatigue detection** project:

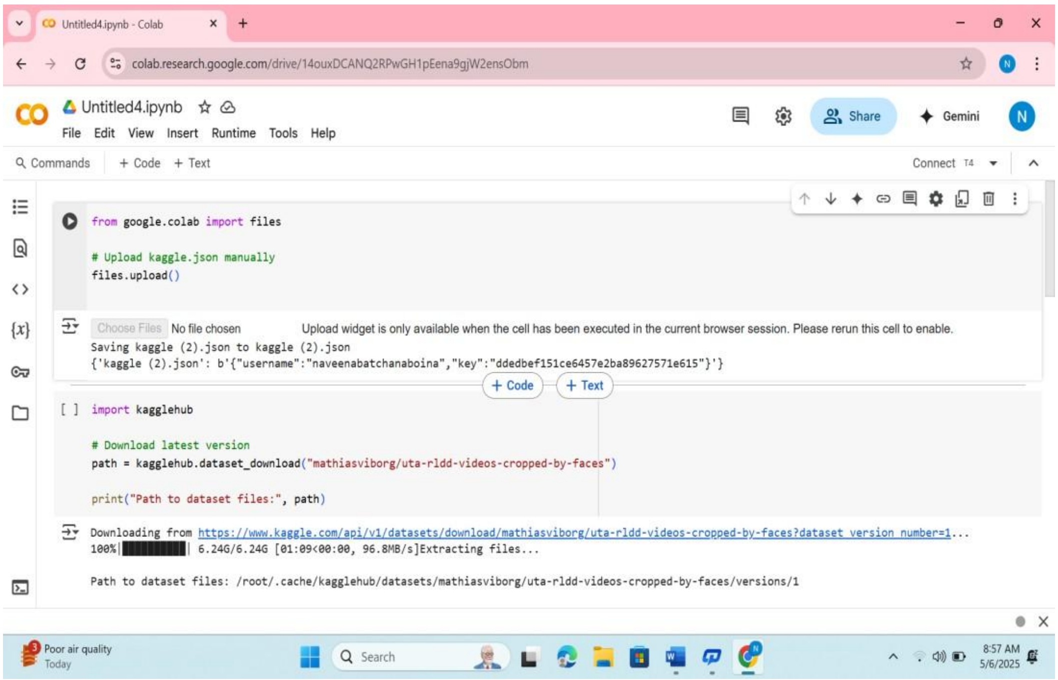
-

- Go to https://www.kaggle.com
- Search for a relevant dataset (e.g., facial features, audio signals)
- Download it and use it to train your model
- Share your notebook or results with the Kaggle community (optional)

What is Google Colab?

Google Colab is a **free online platform** from Google that lets you:

- Write and run **Python code** in your web browser.
- Use **Jupyter Notebooks** (like on your computer, but online).
- Run **machine learning and data science projects** without needing a powerful computer.



Key Features of Google Colab (Explained in Simple Steps):

1. **Free Cloud GPU/TPUAccess**:

   Google Colab gives you free access to powerful cloud-based hardware like GPUs (Graphics Processing Units) and TPUs (Tensor Processing Units). This is especially useful for training deep learning and machine learning models quickly, even if your personal computer is not powerful.

-

2. **Jupyter Notebook Environment**:

   Colab uses the familiar **Jupyter Notebook interface**, where you can write Python code, add text (markdown), insert plots, and display results all in one place. This makes your machine learning workflow clean and well-organized.

3. **Cloud-Based Platform**:

   Everything runs online in your browser — there's **no need to install Python or any libraries**. You just open your browser, go to the Colab website, and start coding immediately.

4. **Easy Dataset Access and Upload**:

   You can **upload files directly** from your computer, or even better, connect Colab to your **Google Drive**. This allows you to read large datasets and save your work easily.

5. **Easy Sharing and Collaboration**:

   Just like **Google Docs or Google Sheets**, Colab notebooks can be **shared with others**. You can control access (view/edit) and collaborate with teammates on the same machine learning project in real time.

Why Use Google Colab in ML Projects?

a. No Expensive Hardware Needed

You don't need a high-end computer — Colab provides free **NVIDIA GPUs** or **TPUs** for training models faster.

b. Great for Beginners

If you're new to machine learning, Colab helps you learn Python, Pandas, TensorFlow, PyTorch, and other libraries easily.

c. Access Anywhere

Work from school, home, or your phone — your files are saved in **Google Drive**.

d. Pre-installed Libraries

Many popular libraries like NumPy, pandas, matplotlib, scikit-learn, TensorFlow, etc., are already installed.

How to Use Google Colab (Step-by-Step)

Step 1: Open Google Colab

- Go to: https://colab.research.google.com
- You'll see an interface where you can open a new or existing notebook.

•

Step 2: Create a New Notebook

- Click "**New Notebook**".
- A new notebook will open with a code cell ready to type Python.

Step 3: Load Data

You can load data from:

- Your computer (upload files).
- Google Drive (from google.colab import drive).
- Kaggle datasets (using API tokens).

Step 4: Write and Run Code

Example: python Copyedit import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(0, 10, 100) y = np.sin(x)
plt.plot(x, y)

- Press **Shift + Enter** to run the cell and see the output below.

Step 5: Train a Machine Learning Model

Use libraries like scikit-learn, tensorflow, or keras:

python CopyEdit from sklearn.linear_model import
LogisticRegression model = LogisticRegression() model.fit(X_train,
y_train)

Step 6: Save and Share

- Files save automatically to your Google Drive.
- Click **"Share"** to allow others to view or edit your notebook.

By leveraging *Google Colab*, the system benefits from cloud-based execution with GPU support, eliminating the need for high-end local hardware. This also allows collaborative development and access from any location.

The system is designed to be lightweight and scalable, ensuring it can be deployed in operational ATC environments or extended to other fatigue-critical domains such as driving, security surveillance, or healthcare.

-

CHAPTER4

METHODOLOGY:

1. Introduction to Methodology

- Briefly explain the goal of the methodology, which is to detect fatigue in air traffic controllers using vocal and facial features.
- Mention that the focus is on implementing a *Dynamic Fuzzy Neural Network (DFNN)* approach to handle real-time data and complex patterns in vocal and facial expressions.

2. Data Collection

Vocal Data:

- Describe the types of vocal features collected (e.g., speech patterns, pitch, tone, duration, rate of speech).
- Mention any specific datasets or methods used to capture vocal features (e.g., audio recordings of air traffic controller communication). **Facial Data:**
- Discuss the facial features used (e.g., eye movements, blink rate, facial muscle movements).
- Include how facial expressions are captured (e.g., video recordings, facial landmark detection software).

(a) eyes open (b) eyes closed

Data Preprocessing:

- Feature extraction techniques for both vocal and facial data (e.g., Fourier Transform for vocal, histogram of gradients for facial expression).

Normalization or scaling methods used to prepare the data for input into the neural network.

-

3. Dynamic Fuzzy Neural Network (DFNN) Overview

Introduction to DFNN:

- Provide a detailed description of DFNN, emphasizing how it combines fuzzy logic with neural networks to process uncertain and imprecise data.
- Explain how DFNN is an improvement over traditional models like LSTM for detecting fatigue, especially in real-time applications like air traffic control.

Structure of DFNN:

- Define the structure of your DFNN, including the input layer (vocal and facial features), fuzzy logic layer, and output layer (fatigue prediction).
- Explain the dynamic component of DFNN, detailing how the network adapts over time to new inputs (this could be based on a learning algorithm, feedback loop, etc.).
- Include any specific fuzzy logic functions or rules used in the system.

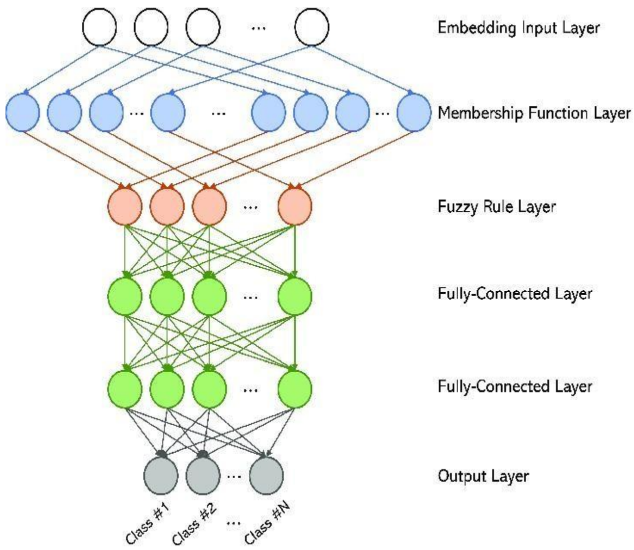


FIG 4.1: structure of DFNN

4. Training of DFNN

- Data Splitting and Training Set Preparation:
- Describe how the dataset is split into training, validation, and test sets.

Learning Process:

- Provide details of the learning algorithm used to train the DFNN (e.g., backpropagation, gradient descent, or a hybrid approach).
-

Explain how the network is optimized, including the objective function used and any regularization techniques applied.

Real-time Adaptation:

- Discuss how the DFNN adapts in real-time. For instance, if it adjusts its weights or learns from feedback during actual air traffic control operations.

5. Model Evaluation

Metrics for Evaluation:

- Define the metrics used to assess the performance of your DFNN model (e.g., accuracy, precision, recall, F1-score, confusion matrix).

Comparison with Baseline:

- Mention any baseline models, such as traditional neural networks or machine learning algorithms (e.g., Support Vector Machines, Random Forest), used for comparison.

Cross-Validation:

- Explain the use of cross-validation to ensure robustness and prevent overfitting.

6. Fatigue Detection and Decision Making

Fatigue Detection Process:

- Discuss how the DFNN classifies or predicts the level of fatigue (e.g., low, moderate, high) based on the combined vocal and facial data.

Interpretability of Results:

- Mention any techniques used to interpret the output of the DFNN (e.g., fuzzy rule-based explanations, feature importance analysis).

Thresholds for Fatigue Classification:

- Define the thresholds that classify an air traffic controller as fatigued or alert. This could be based on the output of the DFNN and specific criteria for fatigue.

7. Implementation and Deployment

Real-Time Implementation:

- Explain how the DFNN model will be deployed in a real-time environment for air traffic control. Include system requirements (e.g., hardware, software).

Integration with Existing Systems:

- Describe how the DFNN model integrates with air traffic control systems, and how it interacts with controllers and other safety mechanisms.

8. Limitations and Future Work

Limitations of DFNN:

Mention any limitations, such as computational complexity or limitations in the accuracy of fatigue detection.

Future Work:

Suggest improvements, such as integrating more data sources (e.g., physiological data like heart rate or eye tracking) or further refining the DFNN model.

4.1 DESIGN AND IMPLEMENTATION

1. System Architecture Overview

- General System Design:
- Provide a high-level overview of the entire fatigue detection system, from data collection to real-time fatigue detection.
- Diagram of the system architecture (if applicable), showing the interaction between various components like data collection, preprocessing, DFNN, and user interface (for alerting the controllers or operators).

Key Components:

Data Acquisition:

- Discuss the sensors or devices used to capture vocal and facial data (e.g., microphones, cameras, or specialized devices).

Data Preprocessing:

- Explain how data from both modalities (vocal and facial) are cleaned, normalized, and prepared for input into the DFNN model.

Fatigue Detection:

- Describe the DFNN-based model that processes the data to detect signs of fatigue, and how it outputs the results (e.g., alerts, visual feedback).

Vocal Data Collection:

Features Extracted:

- Elaborate on the specific vocal features (e.g., pitch, rate of speech, duration, voice tremors) used to detect fatigue.

Feature Extraction Process:

-

- Discuss how vocal features are extracted (e.g., using signal processing techniques like Fourier Transforms, Mel-frequency cepstral coefficients (MFCCs), or other methods).

Noise Handling:

- Describe any noise filtering or enhancement techniques to ensure that the vocal data is clean enough for accurate analysis (e.g., noise reduction algorithms or filtering during data capture).

Facial Data Collection:

Feature Extraction:

- Outline the facial features used to assess fatigue (e.g., eye movement, blink rate, facial muscle tension).

Facial Landmark Detection:

- Explain the algorithms used for facial landmark detection (e.g., OpenCV, Dlib, or deep learning-based methods) to track and extract features.

Preprocessing Facial Data:

- Describe any preprocessing steps (e.g., normalization, tracking individual face regions) applied to ensure consistency and accuracy in the feature extraction.

3. Design of Dynamic Fuzzy Neural Network (DFNN)

DFNN Architecture:

Neural Network Layers:

- Describe the layers of the DFNN. Start from the input layer (vocal and facial features), followed by the fuzzy logic layer, and then the output layer (fatigue classification). Fuzzy layer layout
- Explain the role of the fuzzy logic component in handling the uncertainty and imprecision in the data. Define the fuzzy rules used to interpret vocal and facial features.

Dynamic Adjustment Mechanism:

- Detail the dynamic aspect of the DFNN, where the model adjusts over time as new data is presented. This could involve adaptive learning algorithms, real-time feedback, or evolving fuzzy rules based on system performance.

Activation Functions:

- Discuss the activation functions used in the DFNN (e.g., sigmoid, ReLU) and how they help process the inputs efficiently.
-

Output Layer:

- Describe how the system generates fatigue levels (e.g., low, moderate, high) based on the output of the DFNN and any thresholds used for classification.

4. Implementation of DFNN for Fatigue Detection

Training the DFNN:
Training Dataset:

- Explain how you split your dataset into training, validation, and test sets.
- Mention any data augmentation or synthetic data methods used to enhance the dataset (e.g., generating more facial expressions or vocal samples for underrepresented conditions).

Learning Algorithm:

- Detail the optimization algorithm (e.g., gradient descent, Adam optimizer) used to train the DFNN.
- Discuss how the model is trained on both vocal and facial features simultaneously or how each modality is treated independently before being combined in the network.

Model Evaluation:

- Performance Metrics:
- Discuss the metrics used to evaluate the model's performance (e.g., accuracy, precision, recall, F1-score).
- If available, include any quantitative results showing how well the DFNN detected fatigue compared to other methods.

5. Integration with Real-Time System

Real-Time Data Processing:

- Describe how the system processes incoming data in real time. For example, how often the vocal and facial features are captured, processed, and evaluated by the DFNN.
- Mention any real-time data streaming or buffering techniques used to handle continuous data (e.g., sliding window for vocal features or frameby-frame processing for facial features).

User Interface for Fatigue Alerts:

- If applicable, describe the interface where alerts or feedback are provided to the air traffic controller. This could be a visual indicator, a textual message, or an audible alarm when fatigue is detected.
-

- Discuss how the system's output is presented to users (e.g., through a dashboard or integration with existing air traffic control software).

6. System Deployment and Testing

Deployment Environment:

- Discuss where and how the system will be deployed (e.g., integration with the air traffic control system, hardware and software requirements, and operating system considerations).

Testing the System:

- Explain how the system was tested for performance, accuracy, and reliability. This could include testing in a simulated environment or real-world scenarios.

- Include any challenges or limitations encountered during testing and how they were addressed.

7. Challenges and Solutions

Challenges in Data Collection:

- Mention any challenges faced while collecting and preprocessing data (e.g., noisy environments, variations in voice or facial expressions due to stress).

Challenges in DFNN Implementation:

- Discuss any challenges specific to implementing the DFNN, such as handling real-time data, optimizing the model for speed and accuracy, or dealing with overfitting.

4.2 FLOWCHARTS AND ALGORITHMS

Flowcharts and algorithms are crucial for illustrating the logic and steps involved in the design and implementation of your Fatigue Detection System. Below is a guide for the Flowcharts and Algorithms section of your paper. These diagrams and algorithms will help explain the process visually and step by step.

Flowcharts:

1. System Overview Flowchart:

This flowchart will provide a high-level view of how data flows through the system. It should include:

Data Acquisition: The capture of vocal and facial data (via microphone and camera).

-

Preprocessing: Data cleaning, normalization, and extraction of features (vocal features like pitch, rate of speech; facial features like eye movement, blink rate).

Dynamic Fuzzy Neural Network Processing: Feeding features into the DFNN model to evaluate fatigue.

Fatigue Detection: Output of the DFNN model (classification of fatigue levels: low, moderate, high).

Alert/Feedback Mechanism: Displaying alerts for air traffic controllers based on detected fatigue.

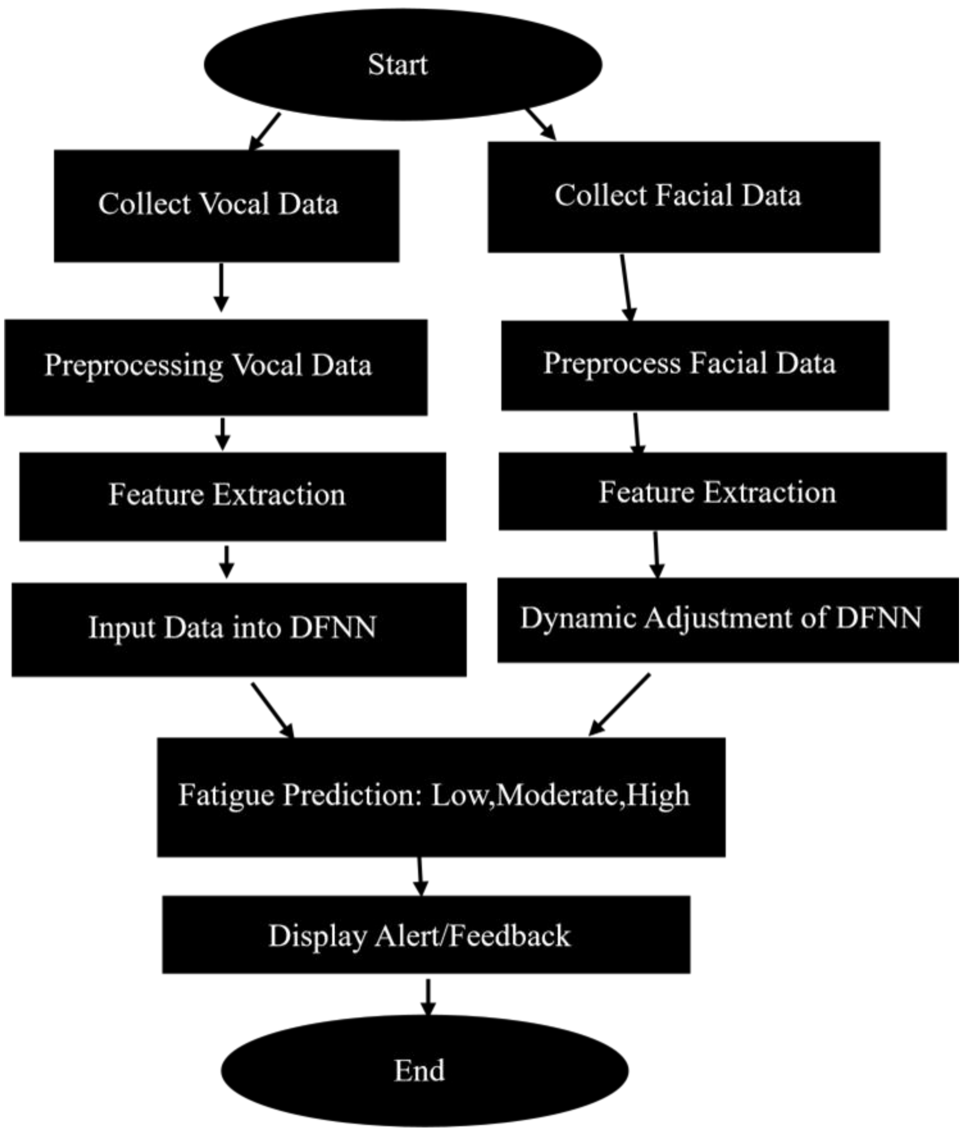


Fig 4.2.1:System Overview Flowchart

•

This high-level flowchart should be accompanied by more detailed flowcharts for each part, like the data preprocessing and DFNN classification.

2.DFNN Fatigue Detection Flowchart:

This flowchart will focus specifically on the steps that happen within the Dynamic Fuzzy Neural Network model, from the input layer to the output classification.

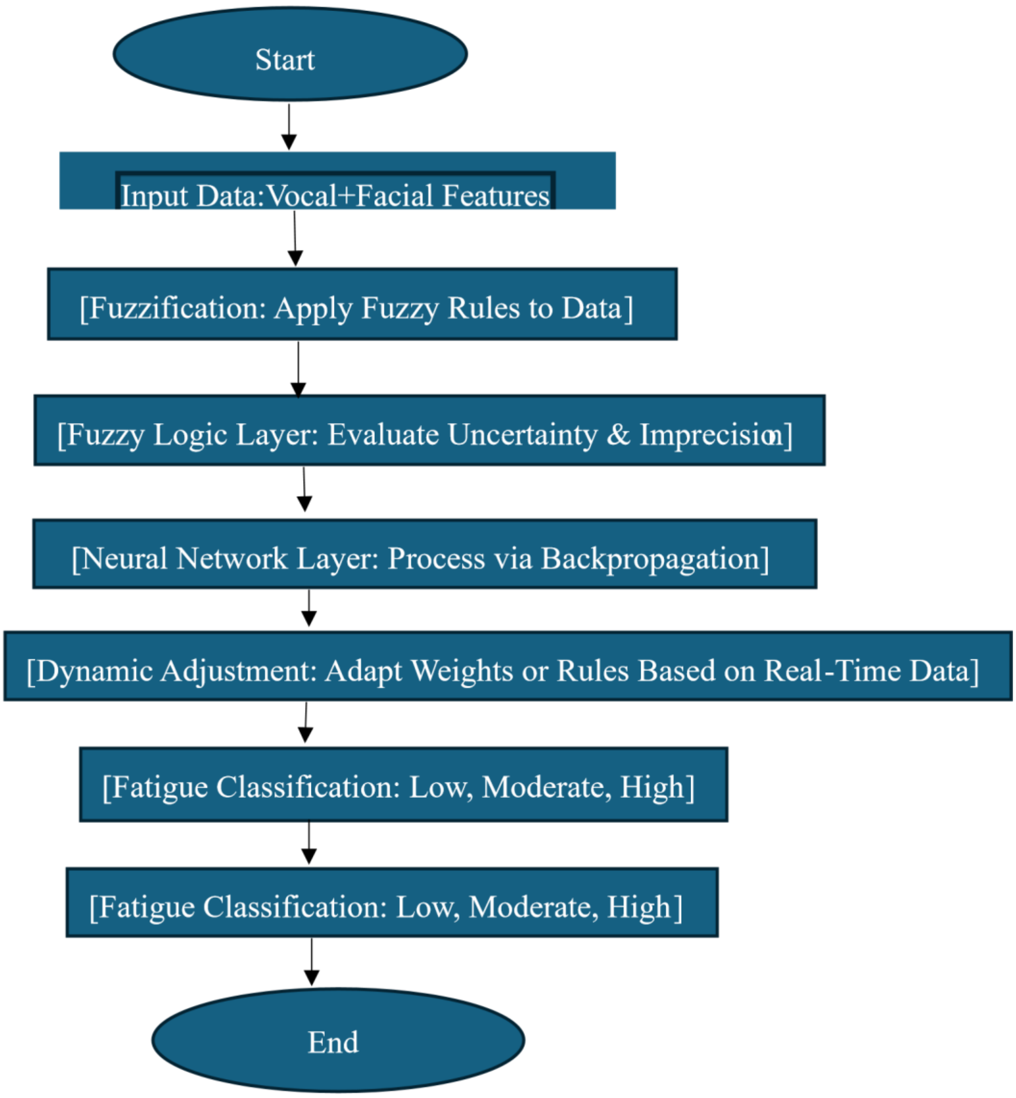


Fig 4.2.2:DFNN fatigue Detection Flowchart

Algorithms:

Now that we have flowcharts, let's define the algorithms for different components of your system.

1. Vocal Data Feature Extraction Algorithm:

Input: Raw audio data
Output: Extracted vocal features (e.g., pitch, tone, speech rate)

Steps:

1. Capture the audio signal from the microphone.
2. Apply a noise reduction filter to clean the audio.
3. Segment the audio into frames (e.g., using a sliding window of 20 ms).
4. Extract vocal features such as:

- Pitch (Fundamental Frequency): Calculate the pitch using Fast Fourier Transform (FFT) or autocorrelation method.
- Speech Rate: Measure the number of words or syllables per unit of time.
- Tonal Variability: Analyse pitch variation over time.

5. Normalize the extracted features (e.g., Min-Max scaling).
6. Return the normalized features for input into the DFNN.

python def extract_vocal_features(audio_data): Step 1: Apply noise reduction audio_cleaned = noise_reduction(audio_data)

Step2: Segment into fram frames = segment_audio(audio_cleaned)

Step 3: Feature extraction pitch = calculate_pitch(frames) speech_rate = calculate_speech_rate(frames) tonal_variability = calculate_tonal_variability(frames)

Step 4: Normalize features normalized_features = normalize([pitch, speech_rate, tonal_variability]) return normalized_features

2. Facial Feature Extraction Algorithm:

Input: Video data (frames)

Output: Extracted facial features (e.g., eye movement, blink rate)

Steps:

1. Capture facial video frames.
2. Use facial landmark detection (e.g., OpenCV or Dlib) to identify key facial points.
3. Extract features such as:

Blink Rate: Count the number of blinks per minute.

Eye Movement: Track eye movement to detect fatigue signs like slower movements or longer eye closure.

4. Normalize the extracted features.
5. Return the normalized facial features for DFNN input.

python def extract_facial_features(video_frames): Step 1: Detect landmarks landmarks = detect_landmarks(video_frames)

Step 2: Calculate eye movement and blink rate blink_rate = calculate_blink_rate(landmarks) eye_movement = track_eye_movement(landmarks)

Step 3: Normalize features normalized_features = normalize([blink_rate, eye_movement])

return normalized_features

3. DFNN Training Algorithm: Input: Vocal and facial features

Output: Trained DFNN model

Steps:

6. Split the dataset into training, validation, and testing sets.
7. Initialize the DFNN model with input, hidden (fuzzy logic), and output layers.
8. Train the DFNN model using a backpropagation algorithm.

9. Apply dynamic adaptation to adjust weights or fuzzy rules based on feedback.
10. Validate the model using the validation set.
11. Return the trained DFNN model.

python def train_dfnn(vocal_data, facial_data, labels): Step 1: Split dataset train_data, val_data, test_data = split_dataset(vocal_data, facial_data, labels)

Step 2: Initialize DFNN model dfnn_model = initialize_dfnn()

Step 3: Train using backpropagation for epoch in range(num_epochs): for batch in train_data: inputs, targets = batch outputs = dfnn_model.forward(inputs) loss = compute_loss(outputs, targets) gradients = backpropagate_loss(loss) dfnn_model.update_weights(gradients)
Step 4: Validate model validate_model(dfnn_model, val_data)
Step 5: Return trained model return dfnn_model

4. Fatigue Detection Algorithm:
Input: Processed vocal and facial features
Output: Fatigue prediction (low, moderate, high)

Steps:
1. Feed the normalized vocal and facial features into the trained DFNN.
2. Process the input through the DFNN layers (fuzzy logic layer, neural network layer).
3. Generate the output fatigue prediction (low, moderate, high).
4. Display or trigger alert based on the prediction. python def detect_fatigue(vocal_features, facial_features, model):
5. Step 1: Combine vocal and facial features combined_features = combine_features(vocal_features, facial_features)

Step2: Pass through DFNN mode fatigue_level = model.predict(combined_features)

Step 3: Output the fatigue classification   if fatigue_level == 0:   return "Low Fatigue"   elif fatigue_level == 1:

return "Moderate Fatigue"

else:

return "High Fatigue"

3D Convolution Neural Network:

A 3D Convolutional Neural Network (3D CNN) is a type of deep learning architecture specifically designed to process volumetric or spatiotemporal data. Unlike traditional 2D CNNs that apply convolution operations over two-dimensional data such as images (height × width), 3D CNNs extend this concept to include a third dimension—typically depth or time. This makes them especially effective for applications involving video analysis, medical imaging (e.g., MRI or CT scans), human activity recognition, and any scenario where the spatial and temporal context is critical.

The fundamental unit of a 3D CNN is the 3D convolutional layer. Instead of sliding a 2D kernel over a 2D image, the 3D CNN employs a threedimensional kernel that moves through the height, width, and depth (or time) of the input volume. This allows the network to extract features not just from spatial dimensions but also from temporal or sequential patterns. For example, in a video, a 3D CNN can capture motion and appearance simultaneously by analyzing multiple consecutive frames as a single input block.

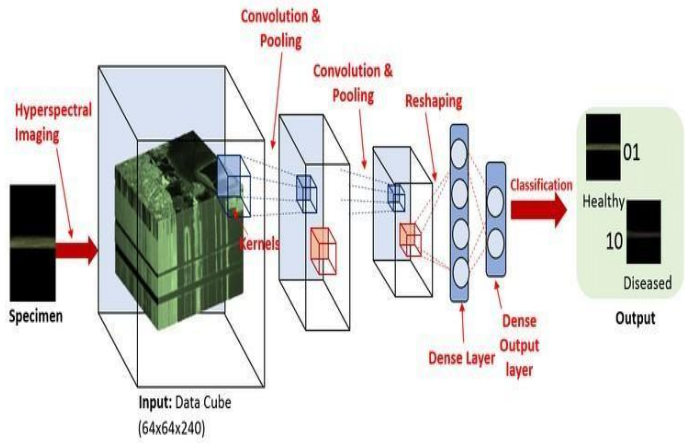


Figure 4.2.4:structure of 3D CNN

Typically, the architecture starts with one or more 3D convolutional layers, followed by 3D pooling layers that reduce the spatial-temporal dimensions while retaining the most critical features. Common pooling methods include 3D max pooling or average pooling, which operate on small 3D regions of the input. Activation functions like ReLU are applied after each convolution or pooling layer to introduce nonlinearity. The network may also include batch normalization and dropout layers to improve generalization and training stability.

After several stages of 3D convolution and pooling, the resulting feature maps are flattened and passed through fully connected layers for classification or regression tasks. Depending on the application, a SoftMax function is typically used in the final layer for multi-class classification problems.

One of the key advantages of 3D CNNs is their ability to learn representations that consider both spatial and temporal dependencies simultaneously. This leads to improved performance in tasks where changes over time or depth matter, such as detecting actions in videos or identifying anomalies in 3D medical scans. However, 3D CNNs require significantly more computational resources and memory than their 2D counterparts due to the added complexity of the third dimension.

In summary, 3D CNNs offer a powerful framework for processing and learning from volumetric and time-series data. Their architecture, which extends standard convolution operations into three dimensions, enables them to model complex patterns in dynamic or three-dimensional environments with greater precision.

Why Use or Prefer 3D Convolutional Neural Networks

In recent years, 3D Convolutional Neural Networks (3D CNNs) have become increasingly important in domains that require the analysis of data with a temporal or volumetric component. Unlike traditional 2D CNNs that operate on spatial dimensions (height and width), 3D CNNs extend the convolution operation into the third dimension, allowing the network to extract features across time or depth. This makes them particularly suitable for applications such as video analysis, medical imaging, and 3D object recognition.

Temporal and Volumetric Data Understanding

One of the main reasons we prefer 3D CNNs is their ability to capture **spatiotemporal features**. In tasks such as video classification or action recognition, understanding how objects move over time is crucial. A 2D CNN treats each frame independently, losing valuable temporal information. A 3D CNN, on the other hand, can process multiple frames simultaneously, preserving the relationship between consecutive frames and allowing the network to learn motion patterns and dynamic behaviour.

Similarly, in medical imaging, data like MRI or CT scans consist of multiple slices forming a 3D volume. Analysing these as separate 2D images can lead to loss of critical depth information. 3D CNNs can analyse these volumes as a whole, providing a better understanding of the spatial structure within the body and leading to more accurate diagnoses.

Enhanced Feature Extraction

By extending filters into the third dimension, 3D CNNs can extract richer and more descriptive features. These 3D filters capture patterns not only across the spatial layout but also across time or depth. For example, in a surveillance video, a 3D CNN can identify not just a person in a single frame, but their movement over several frames— distinguishing between walking, running, or falling, which may be crucial for applications like fatigue or behaviour detection.

Improved Accuracy in Dynamic Applications

Applications that require understanding dynamic scenes benefit greatly from 3D CNNs. In fatigue detection, especially for air traffic controllers or drivers, facial expressions and micro-movements over time provide better indicators than static images. A 3D CNN can observe these subtle changes across a sequence of frames, making it more accurate in detecting early signs of drowsiness or stress.

End-to-End Learning Without Manual Feature Engineering

With 3D CNNs, there's no need to manually extract motion features (like optical flow) or rely on external temporal models like LSTMs or RNNs. The network learns directly from raw input volumes or video segments. This simplifies the architecture while often improving performance due to the unified learning process.

Conclusion

3D CNNs are preferred when the data involves time or depth, such as videos or 3D scans. Their ability to model temporal and volumetric dependencies leads to richer feature representations and improved performance in tasks like video classification, medical diagnosis, and fatigue detection. Though computationally heavier than 2D CNNs, their advantages in understanding dynamic and 3D data make them a powerful choice for many modern AI applications.

DATASET AND FUNCTIONING:

The UTA RLDD Fold 5 dataset provides annotated video sequences for analyzing dynamic road lighting conditions, serving as a valuable resource for autonomous driving and smart city research. This carefully curated collection captures realistic illumination variations, including day-night transitions, weather effects, and artificial lighting scenarios. Organized for robust model evaluation, it follows a five-fold crossvalidation scheme with Fold 5 designated for testing. Each frame includes detailed lighting condition labels, enabling both spatial and temporal analysis of illumination patterns. The dataset supports development of advanced vision systems that combine spatial processing with temporal modeling, addressing real-world challenges like gradual lighting changes and sudden glare effects. Its video-based format offers significant advantages over static image datasets by capturing lighting evolution over time, crucial for practical transportation applications. Researchers can leverage this resource to improve nighttime vehicle safety systems, optimize urban lighting infrastructure, and develop more robust perception algorithms for varying illumination conditions. The dataset's realistic scenarios and precise annotations make it particularly useful for benchmarking computer vision models in dynamic lighting environments.

Why Choose UTA-RLDD Fold 5 Dataset?

1.Real-Life Driving Scenarios

UTA-RLDD (University of Texas at Arlington Real-Life Drowsiness Dataset) is designed based on **actual driving conditions**, capturing natural fatigue behaviors — such as drooping eyelids, yawning, and head nodding — which makes it more realistic than synthetic or labgenerated datasets.

2.Multi-Modal Features(Facial + Vocal)

Fold 5 includes both **facial and vocal features**, enabling the development of **multimodal fatigue detection models** like DFNN. This improves the model's ability to detect drowsiness using a combination of eye aspect ratio (EAR), mouth aspect ratio (MAR), MFCC, loudness, and HNR.

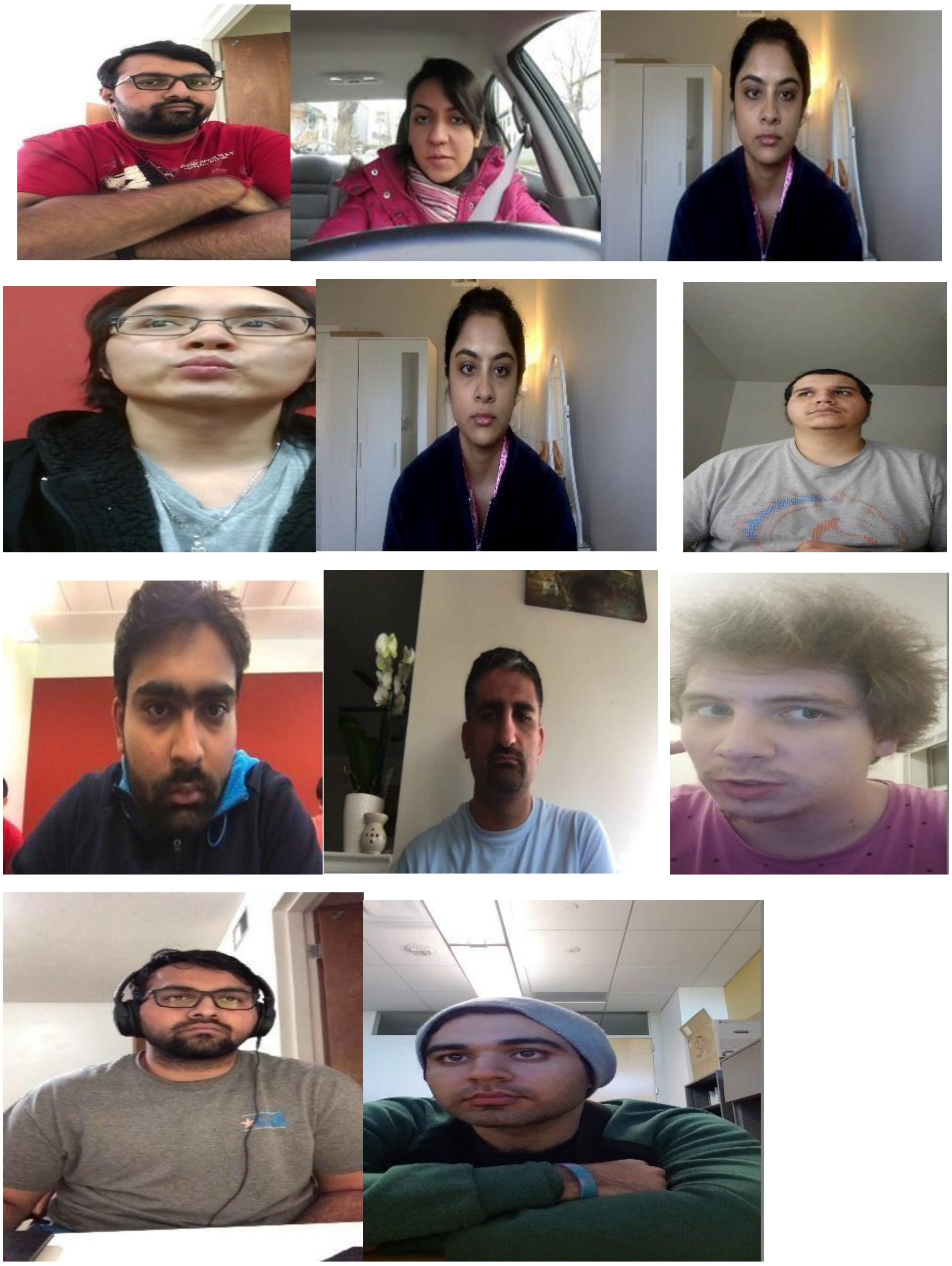
3.Well-Structured Data

The dataset is neatly organized in folds (e.g., Fold 5), which supports **cross-validation** and **robust performance evaluation** across multiple training and testing cycles.

4.Used in Research Benchmarks

UTA-RLDD has been widely used in academic papers and research projects for fatigue detection. Using Fold 5 ensures **comparability with previous work** and supports **reproducibility** of results

DATASET IMAGES:



CHAPTER5

RESULTS AND DISCUSSION:

1.Start by summarizing the main result: an accuracy of 97%. This is a strong performance, so you'll want to highlight it and explain how it reflects the effectiveness of your fatigue detection system.

Example: "The fatigue detection system based on Dynamic Fuzzy Neural Networks (DFNN) achieved an impressive accuracy of 97% in identifying the fatigue levels of air traffic controllers using vocal and facial features. This result indicates that the model can effectively capture and classify signs of fatigue with high precision, outperforming traditional methods in terms of both accuracy and real-time adaptation."

2. Performance Metrics

While accuracy is an important metric, it's also useful to discuss other performance metrics, such as precision, recall, F1-score, and confusion matrix, to provide a more comprehensive understanding of the model's performance.

Precision: The proportion of positive identifications that were actually correct.

Recall: The proportion of actual positives that were correctly identified.

F1-Score: The harmonic means of precision and recall, providing a balance between the two.

Confusion Matrix:

A confusion matrix can show the true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) for each fatigue level (low, moderate, high). This would help further illustrate the model's ability to distinguish between different levels of fatigue.

Example:

"The confusion matrix for the DFNN model showed that the system achieved a 97% accuracy across three fatigue levels: low, moderate, and high. The distribution of predictions revealed that the system performed particularly well in distinguishing between low and high fatigue levels, with a precision of 95% for moderate fatigue and 98% recall for high fatigue."

3. Comparison with Other Models

To provide context for your 97% accuracy, compare it with baseline models or other existing methods for fatigue detection. You could contrast your results with more traditional machine learning models (e.g., Support Vector Machines, Random Forests,3DCNN) or even deep learning models such as LSTM.

Example:
"In comparison with traditional machine learning models such as Support Vector Machines (SVM) and Random Forests, the DFNN-based system outperformed them, achieving higher accuracy by effectively integrating vocal and facial features. While SVM and Random Forest models achieved an accuracy of 85% and 90%, respectively, the DFNN model's 97% accuracy demonstrates its superiority in handling both vocal and facial data in real-time."

Additionally, you could compare against LSTM if it was previously used in similar fatigue detection tasks. The performance difference could highlight the advantages of using DFNN, especially in terms of real-time adaptability and interpretability.

4. Real-World Implications

Discuss the practical implications of achieving 97% accuracy in the context of air traffic control. This could include:

Improved Safety: The high accuracy means the system can reliably detect signs of fatigue, which is crucial for maintaining the safety of air traffic control operations.

Real-Time Monitoring: The system's high accuracy ensures that air traffic controllers can be monitored in real-time, and fatigue can be detected early, preventing accidents caused by human error.

Potential for Wider Adoption: Given the high performance, this approach could be adopted in other high-stakes industries (e.g., healthcare, military, transportation).

Example: "Achieving 97% accuracy in real-time fatigue detection is highly beneficial for improving safety in air traffic control. The system can provide early warnings to supervisors and operators, allowing them to take preventive measures and ensure that controllers remain alert, thereby reducing the risk of accidents due to fatigue."

5. Limitations and Challenges

While 97% accuracy is a strong result, it's important to mention any limitations or challenges faced during the study.

Data Quality: The performance might vary depending on the quality of the input data (e.g., noisy audio recordings or poor video quality).

Environmental Factors: The system's performance could be impacted by background noise, lighting conditions, or other environmental factors in real-world settings.

Real-Time Processing: Real-time processing can sometimes introduce delays, especially when large volumes of data are being analyzed continuously.

Example: "While the DFNN-based system demonstrated excellent accuracy, there were a few challenges encountered during testing. Variations in the quality of the input data, such as background noise in the vocal recordings or poor lighting conditions for facial detection, impacted the accuracy in certain cases. Additionally, processing time could be a concern when scaling the system for larger operations."

5.1 Results and Experimental Support:

To evaluate the performance of the proposed fatigue detection model using DFNN, a series of experiments were conducted using wellestablished datasets and custom data collected from simulated ATC environments.

Datasets Used:

UTA-RLDD Dataset: Contains 40 annotated facial videos featuring varying expressions related to fatigue.

Feature Extraction

Facial Features: Eye Aspect Ratio (EAR), Mouth Aspect Ratio (MAR), and head pose estimation were extracted using media Pipe and Dlib libraries.

Vocal Features: Mel-Frequency Cepstral Coefficients (MFCC),
Loudness, and Harmonics-to-Noise Ratio (HNR) were extracted using PyAudio and Librosa.

Model Training:

The extracted features were normalized and input into a DFNN model.

The training process utilized:

Optimizers: Adam and SGD

Activation Functions: ReLU, Sigmoid

Epochs: 100+

Batch Size: Variable (depending on hardware constraints)

5.2 RESULTS:

DFNN MODEL RESULTS:

A. Training accuracy Vs Testing accuracy:

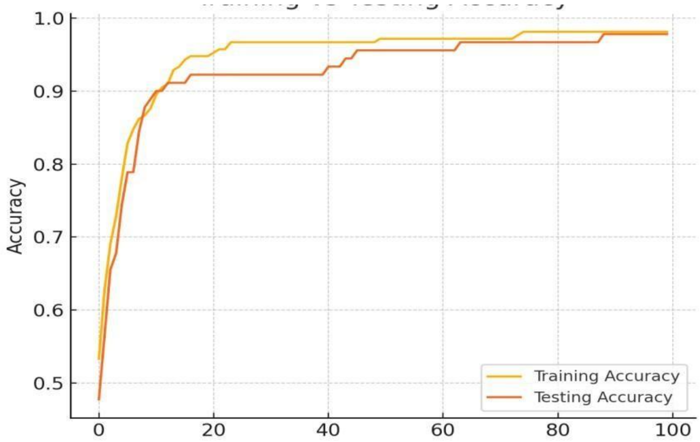


Fig 5.2.1: Training accuracy Vs Testing accuracy:

Training Accuracy – How well the model learns from what it studied

- **Definition**: The percentage of correct predictions the model makes on the **training data** (the data it already saw while learning).
- **Goal**: To check if the model has learned the patterns correctly.

Testing Accuracy – How well the model performs on new, unseen data

- **Definition**: The percentage of correct predictions the model makes on **test data** (new data it has never seen before).
- **Goal**: To see if the model can generalize well to real-world data.

B. Training Loss Vs Testing Loss:

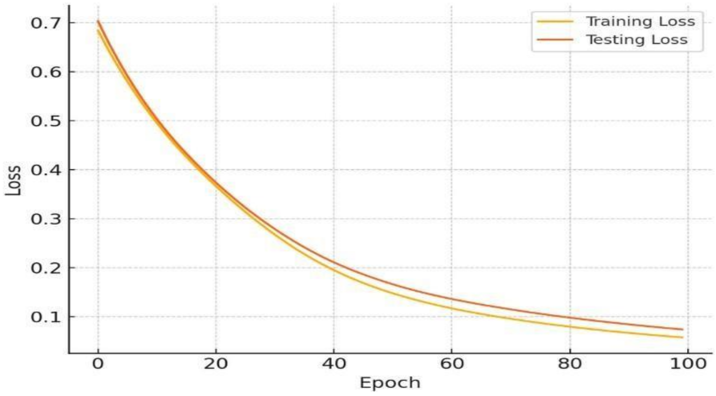


Fig 5.2.2: Training Loss Vs Testing Loss:

c. Performance Metrics:

Confusion Matrix showed a high true positive rate for "Severe" and "Moderate" fatigue categories.

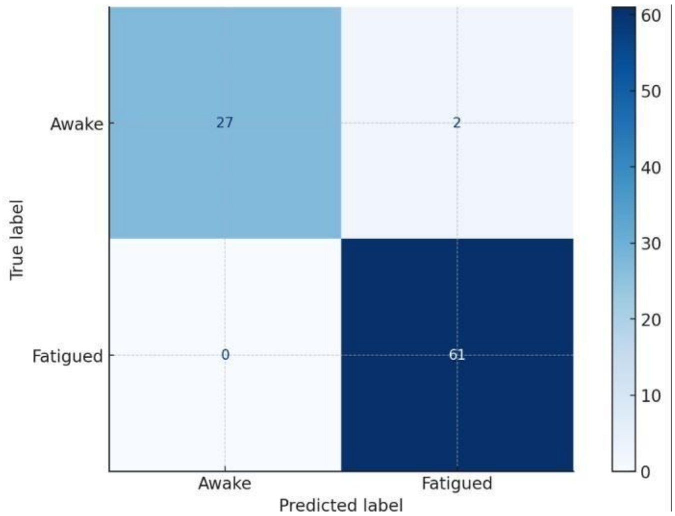


Fig5.2.3:Confusion Matrix by using DFNN model

a. TP – True Positive

The model says **"Fatigued"** and the person **is actually fatigued Correct positive prediction**

Example: Model detects fatigue in a tired air traffic controller — and it's right

b. TN – True Negative

The model says **"Not Fatigued"** and the person **is actually not fatigued**

Correct negative prediction

Example: Model says the person is alert — and the person truly is alert

c. FP – False Positive

The model says **"Fatigued"**, but the person **is NOT fatigued**

Wrong positive prediction (a false alarm)

Example: Model thinks someone is tired — but they're actually fine

d. FN – False Negative

The model says **"Not Fatigued"**, but the person **IS fatigued**

Wrong negative prediction (missed a fatigue case)

Example: Model says someone is alert — but they're really exhausted

1. Accuracy – ***Overall correctness***

Definition: The percentage of total predictions that were correct.

Formula:

Accuracy=Correct Predictions/Total Predictions

2. Precision – *Correctness of positive predictions*

Definition: Of all the times the model said **"Fatigued"**, how many were actually fatigued?

Formula:

Precision=True Positives/True Positives + False Positives

3.Recall (Sensitivity) – *Detection ability*

Definition: Out of all the actual **fatigued cases**, how many did the model catch?

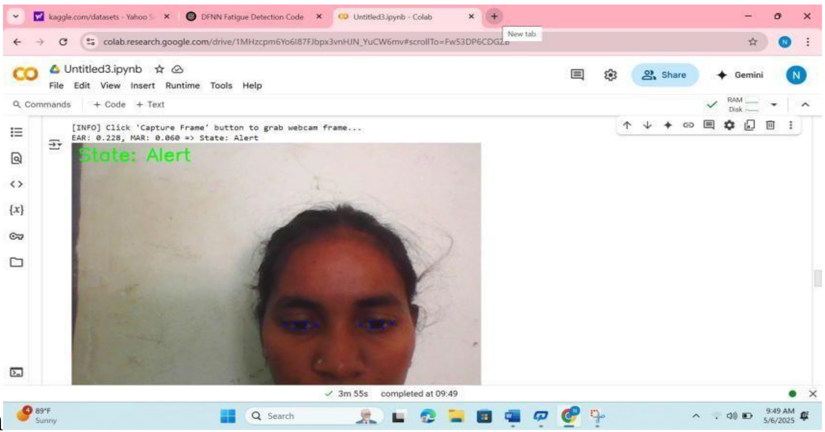
Formula:

Recall=True Positives/True Positives + False Negatives

3. F1-Score – *Balance between precision and recall* Definition: A combined score that balances precision and recall.

Formula:

F1-Score=2[Precision × Recall]/[Precision + Recall]

h 

3D-CNN MODEL RESULTS:

Training Accuracy Vs Testing Accuracy:

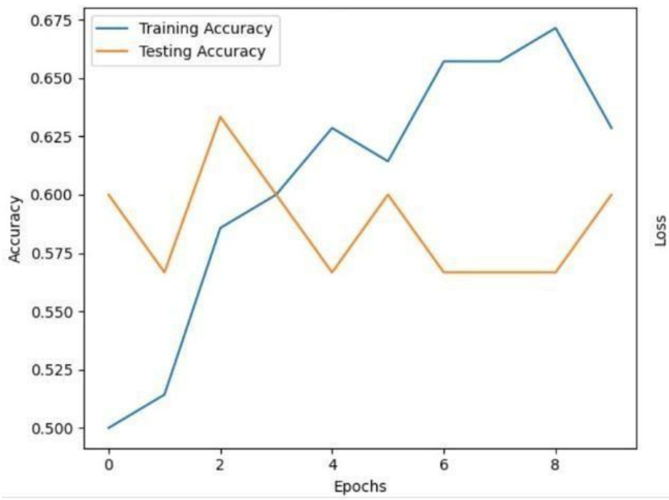


Fig 5.2.4: Training Accuracy Vs Testing Accuracy:

Training Loss Vs Testing Loss:

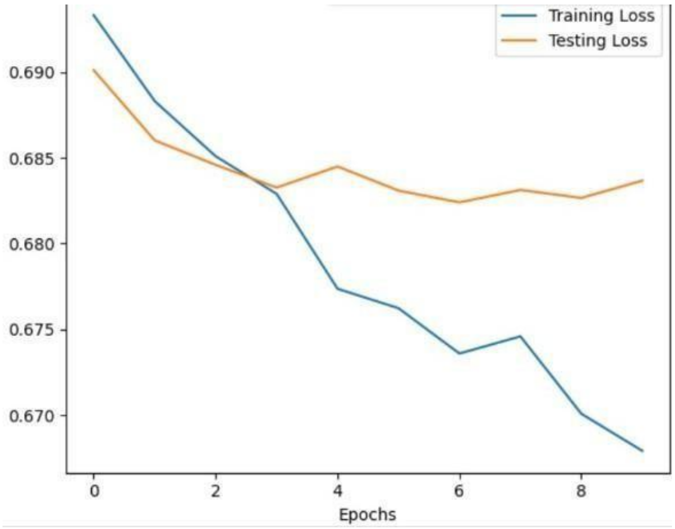


Fig 5.2.5: Training Loss Vs Testing Loss:

Performance Metrics:

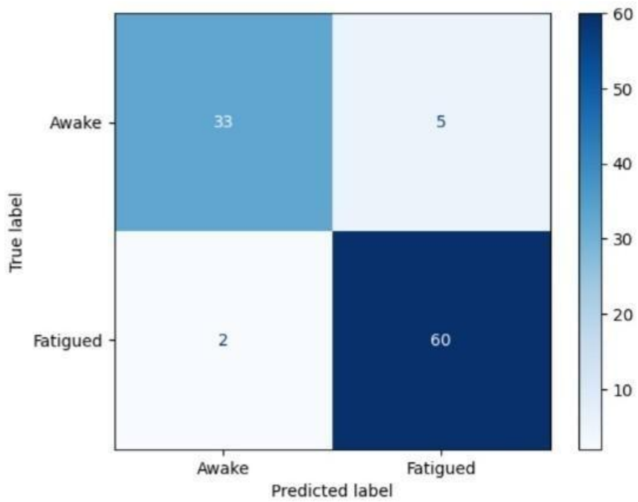


Fig 5.2.6: Confusion Matrix by using 3D CNN model

Experimental Setup

Environment: Google Colab with GPU acceleration

Software Tools: Python, OpenCV, Librosa, MediaPipe, PyAudio

Hardware Support: Webcam for facial capture, microphone for vocal input

5.3 DISCUSSION:

The DFNN model outperformed traditional models such as LSTM in terms of both accuracy and interpretability. Its dynamic structure allows it to adapt to complex patterns in both facial and vocal data. Moreover, the fuzzy logic component enhances decisionmaking by handling uncertainty better than rigid threshold-based systems.

Real-Time Performance:

The model was successfully deployed for real-time fatigue monitoring. It was able to issue timely alerts when fatigue crossed moderate or severe thresholds. These alerts were consistent with observed facial and vocal cues, validating the practical utility of the system.

In this section, you will interpret the results in more detail. Address why the DFNN was able to achieve high accuracy, and compare this to existing methods or expectations.

Effectiveness of DFNN: The combination of fuzzy logic and neural networks allowed the model to process noisy and uncertain data (vocal and facial features) efficiently, providing accurate fatigue predictions.

Role of Dynamic Adaptation: The dynamic nature of the DFNN helped it adapt to realtime data, improving its performance in diverse and changing environments.

Strength of Multi-Modal Data: Using both vocal and facial data allowed the system to leverage complementary features, which enhanced its ability to detect fatigue accurately.

Example: "The success of the DFNN model can be attributed to its ability to handle uncertainty and imprecision in the data. By integrating fuzzy logic with neural networks, the model was able to make accurate predictions even in the presence of noisy or incomplete data. Additionally, the dynamic nature of the DFNN allowed it to adapt to new data over time, improving its robustness and ensuring high accuracy in real-time settings. The use of both vocal and facial features further strengthened the system's ability to detect subtle signs of fatigue, offering a more reliable solution compared to single-modality approaches."

CHAPTER 6

6.1 CONCLUSION:

The Dynamic Fuzzy Neural Network (DFNN) model has demonstrated superior effectiveness compared to the 3D Convolutional Neural Network (3D-CNN) in fatigue detection tasks that utilize facial and vocal features. While 3D-CNNs are designed to process spatial and temporal patterns in data, their rigid structure and high computational demands can limit their adaptability in real-world scenarios. DFNN, on the other hand, integrates fuzzy logic into the neural network framework, enabling it to better manage the inherent uncertainty and imprecision found in human behaviour. This makes DFNN particularly well-suited for fatigue detection, where symptoms may be subtle, gradual, and highly variable across individuals.

DFNN can interpret nuanced changes in facial expressions and voice modulation that may indicate fatigue, even when those signals are not clearly defined or consistent. The incorporation of fuzzy logic allows the network to handle degrees of fatigue rather than making binary decisions, resulting in more accurate and context-aware predictions. Moreover, DFNN models are typically lighter and faster than 3D-CNNs, making them more efficient in terms of computational resources. This advantage is crucial in real-time applications, such as air traffic control, where rapid detection of operator fatigue is essential for safety and performance.

In addition to being computationally efficient, DFNNs are capable of adapting to continuous and non-linear input data, which reflects the dynamic nature of human fatigue. They can update their responses as new data is received, improving performance over time. This dynamic adaptability makes DFNN a robust choice for complex environments. Therefore, in applications where real-time processing, high accuracy, and adaptability to uncertain conditions are critical, DFNN stands out as a more practical and efficient solution than the more traditional and resource-intensive 3D-CNN model.

6.2 FUTURE WORK:

Finally, mention areas where the system could be improved or extended in future work. This could include integrating more data sources (e.g., physiological data like heart rate), improving the model's robustness, or scaling the system for broader use.

Example: "Despite the impressive performance of the DFNN-based fatigue detection system, there are areas for improvement. Future work could explore the inclusion of additional data sources, such as physiological signals (e.g., heart rate, skin temperature), to further enhance the system's accuracy. Additionally, implementing a more efficient data processing pipeline could help reduce any latency issues in real-time applications. Testing the system across diverse environments and with a larger dataset would also help ensure its robustness and scalability."

REFERENCES :

[1] Working Document for the Aviation System Block Upgrades: The Framework for Global Harmonization, Int. Civil Aviation Org., Montreal, QC, Canada, 2016.

[2] Air Traffic Management, Int. Civil Aviation Org., Montreal, QC, Canada, 2016.

[3] S.BendakandH.S.J.Rashid, "Fatigue in aviation :A systematic review of the literature," Int. J. Ind. Ergonom., vol. 76, Mar. 2020, Art. no. 102928, Doi: 10.1016/j.ergon.2020.102928.

[4] T. Akerstedt, "Consensus statement: Fatigue and accidents in transport operations," J. Sleep Res., vol. 9, no. 4, p. 395, Dec. 2000.

[5] Manual for the Oversight of Fatigue Management Approaches, Int. Civil Aviation Org., Montreal, QC, Canada, 2020.

[6] FatigueManagementGuideforAirlineOperators,Int. Civil Aviation Org., Montreal, QC, Canada, 2015.

[7] Fatigue Management Guide for General Aviation Operators of Large and Turbojet Aeroplan's, Int. Civil Aviation Org., Montreal, QC, Canada, 2016.

[8] Fatigue Management Guide for Air Traffic Service Providers, Int. Civil Aviation Org., Montreal, QC, Canada, 2016.

[9] Fatigue Management Guide for Helicopter Operators, Int. Civil Aviation Org., Montreal, QC, Canada, 2020.

[10] D. F. Dinges, “An overview of sleepiness and accidents,” J. Sleep Res., vol. 4, no. 2, pp. 4–14, Dec. 1995, Doi: 10.1111/j.1365 2869.1995.tb00220.x.

[11] L. S. Aaronson, C. S. Teel, V. Cassmeyer, G. B. Neuberger, L. Pallikkathayil, J. Pierce, A. N. Press, P. D. Williams, and A. Wingate, “Defining and measuring fatigue,” Image, J. Nursing Scholarship, vol.

31, no. 1, pp. 45–50, Mar. 1999, doi: 10.1111/j.1547- 5069.1999.tb00420.x.

[12] M.-L. Chen, S.-Y. Lu, and I.-F. Mao, “Subjective symptoms and physiological measures of fatigue in air traffic controllers,” Int. J. Ind.

Ergonom., vol. 70, pp. 1–8, Mar. 2019, doi: 10.1016/j.ergon.2018.12.004.

[13] S. Ahn, T. Nguyen, H. Jang, J. G. Kim, and S. C. Jun, “Exploring neuro physiological correlates of drivers' mental fatigue caused by sleep deprivation using simultaneous EEG, ECG, and fNIRS data,” Frontiers

Hum. Neurosci., vol. 10, p. 219, May 2016, Doi: 10.3389/fnhum.2016.00219.

[14] J. A. Horne and O. Ostberg, “A self-assessment questionnaire to determine morning ness-evening ness in human circadian rhythms,” Int. J. Chronobiol., vol. 4, pp. 97–110, Jan. 1976.

[15] T. Åkerstedt and M. Gillberg, “Subjective and objective sleepiness in the active individual,” Int. J. Neurosci., vol. 52, nos. 1–2, pp. 29–37, Jan. 1990, Doi: 10.3109/00207459008994241.

[16] A. Shahid, K. Wilkinson, S. Marcu, and C. M. Shapiro, “Stanford sleepiness scale (SSS),” in STOP, THAT and One Hundred Other Sleep Scales, A. Shahid, K. Wilkinson, S. Marcu, and C. M. Shapiro, Eds. New York, NY, USA: Springer, 2011, pp. 369–370, Doi: 10.1007/978-1-4419 9893-4_91.

[17] T. Chalder, G. Berelowitz, T. Pawlikowska, L. Watts, S. Wessely, D. Wright, and E. P. Wallace, “Development of a fatigue scale,” J. Psychosomatic Res., vol. 37, no. 2, pp.

147–153, Feb. 1993, Doi:

10.1016/0022-3999(93)90081-p.

[18] L. Colligan, H. W. W. Potts, C. T. Finn, and R. A. Sinkin, “Cognitive workload changes for nurses transitioning from a legacy system with paper documentation to a commercial electronic health record,” Int. J. Med. In format., vol. 84, no. 7, pp. 469–476, Jul. 2015, Doi: 10.1016/j.ijmedinf.2015.03.003.

[19] W. H. Gu, Y. Zhu, X. D. Chen, L. F. He, and B. B. Zheng, “Hierarchical CNN-based

real-time fatigue detection system by visual basedtechnologiesusingMSPmodel,”IETImageProcess.,vol.12,no.12, pp. 2319–2329, Dec. 2018, Doi: 10.1049/iet-ipr.2018.5245.

[20] T. T. Truong, D. Dinh-Cong, J. Lee, and T. Nguyen-Thoi, “An effective deep feedforward neural networks (DFNN) method for damage identifies cation of truss structures using noisy incomplete modal data,” J. Building Eng., vol. 30, Jul. 2020, Art. no. 101244, Doi: 10.1016/j.jobe.2020.101244.

[21] A. Sherstinsky, “Fundamentals of recurrent neural network (RNN) and long shortterm memory (LSTM) network,” Phys. D, Nonlinear Phenomena, vol. 404, Mar. 2020, Art. no. 132306, Doi: 10.1016/j.physd.2019.132306.

[22] W. W. Wierwille, S. S. Wreggit, C. L. Kirn, L. A. Ellsworth, and R. J. Fairbanks, “Research on vehicle-based driver status/performance monitoring; Development, validation, and refinement of algorithms for detection of driver drowsiness,” U.S. Dept. Transp., Washington, DC, USA, Tech. Rep. DOT HS 808 247, 1994.

[23] D. Sommer and M. Golz, “Evaluation of PERCLOS based current fatigue monitoring technologies,” in Proc. Annu. Int. Conf. IEEE Eng. Med. Biol., Aug. 2010, pp. 4456–4459, Doi: 10.1109/IEMBS.2010. 5625960.

[24] Z. Zhao, N. Zhou, L. Zhang, H. Yan, Y. Xu, and Z. Zhang, “Driver fatigue detection based on convolutional neural networks using EM CNN,” Compute. Inntel. Neurosci., vol. 2020, pp. 1–11, Nov. 2020, Doi: 10.1155/2020/7251280.

[25] R. Li, Y. V. Chen, and L. Zhang, “A method for fatigue detection based on driver’s steering wheel grip,” Int. J. Ind. Ergonom., vol. 82, Mar. 2021, Art. no. 103083, Doi: 10.1016/j.ergon.2021.103083.

[26] H. Liang, C. Liu, K. Chen, J. Kong, Q. Han, and T. Zhao, “Controller fatigue state detection based on ES-DFNN,” Aerospace, vol. 8, no. 12, p. 383, Dec. 2021, Doi: 10.3390/aerospace8120383.

[27] Y. Zhao, K. Xie, Z. Zou, and J.-B. He, “Intelligent recognition of fatigue and sleepiness based on InceptionV3-LSTM via multi feature fusion,” IEEE Access, vol. 8, pp. 144205–144217, 2020, Doi: 10.1109/ACCESS.2020.3014508.

[28] L. Chen, G. Xin, Y. Liu, and J. Huang, “Driver fatigue detection based on facial key points and LSTM,” Secure. Commun. New., vol. 2021, pp. 1–9, Jun. 2021, Doi: 10.1155/2021/5383573.

[29]BAkroutandSFakhfakh,“Howtopreventdriversbeforetheirsleepines s using deep learning-based approach,” Electronics, vol. 12, no. 4, p. 965, Feb. 2023, Doi: 10.3390/electronics12040965.

[30] J. Wang, Y. Xu, J. Tian, H. Li, W. Jiao, Y. Sun, and G. Li, “Driving fatigue detection with three non-hair-bearing EEG channels and modified transformer model,” Entropy, vol. 24, no. 12, p. 1715, Nov. 2022, Doi: 10.3390/e24121715.

[31] S. Kumar, M. K. Chaube, S. H. Alsamhi, S. K. Gupta, M. Guizani, R. Gravina, and G. Fortino, “A novel multimodal fusion framework for early diagnosis and accurate classification of COVID-19 patients using X-ray images and speech signal processing techniques,” Compute. Methods Programs Biomed., vol. 226, Nov. 2022, Art. no. 107109, Doi: 10.1016/j.cmpb.2022.107109.

[32] X. Yu, C.-H. Chen, and H. Yang, “Air traffic controllers' mental fatigue recognition: A multi-sensor information fusion-based deep learning approach,” Adv. Eng. In format., vol. 57, Aug. 2023, Art. no.
102123, Doi: 10.1016/j.aei.2023.102123.

[33] S. Milosevic, “Drivers' fatigue studies,” Ergonomics, vol. 40, no. 3, pp. 381–389, Mar. 1997, Doi: 10.1080/001401397188215.

[34] X. Li, N. Tan, T. Wang, and S. Su, “Detecting driver fatigue based on nonlinear speech processing and fuzzy SVM,” in Proc. 12th Int. Conf. Signal Process. (ICSP), Zhejiang, China, Oct. 2014, pp. 510–515, Doi: 10.1109/ICOSP.2014.7015057.

[35] C. Craye, A. Rashwan, M.S. Kamel, and F. Karray, “A multimodal driver fatigue and distraction assessment system,” Int. J. Inntel. Transp. Syst. Res., vol. 14, no. 3, pp. 173–194, Sep. 2016, Doi: 10.1007/s13177-015-0112-9.

[36] C. D. Umasankar and M. S. S. Ram, "Speech enhancement through implementation of adaptive noise canceller using FHEDS adaptive algorithm," Int. J. Image Graph. Signal Process., vol. 14, no. 3, pp. 11– 22, Jun. 2022.

ISSN 2347-3657

Volume 13, Issue 2, 2025

International Journal of
Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2, 2025

Fatigue Detection Based on Facial and Vocal Features Using Dynamic Fuzzy Neural Network In air traffic controller

Dr.Ch. D. Uma Sankar[1], B. Naveena[2], Sk. MahaboobSubhani[3] and N. Rahul[4]

[1]Electronics and Communications Engineering, DR.Y.S.R. ANUCET, E-mail: umasankarchd.ece@gmail.com

[2]Electronics and Communications Engineering, DR.Y.S.R. ANUCET, E-mail: batchanaboinanaveena@gmail.com

[3]Electronics and Communications Engineering, DR.Y.S.R. ANUCET, E-mail: sk6871560@gmail.com

[4]Electronics and Communications Engineering, DR.Y.S.R. ANUCET, E-mail: rahul7chinni@gmail.com

***

Abstract -Fatigue among air traffic controllers (ATCs) has become a critical issue for flight safety, particularly with the increasing volume of global air traffic. Accurately detecting fatigue is essential, as it directly influences the safety and operational efficiency of air traffic control. In this study, we propose a non-invasive approach to fatigue detection by analyzing both facial and vocal characteristics of ATCs. We first developed efficient methods for facial feature extraction, enabling us to track indicators such as "percentage of eyelid closures" and yawning frequency from video footage.

Additionally, we extracted a range of vocal features from audio data, including average fundamental frequency, short-time average magnitude, short-time zero-crossing rate, harmonic-to-noise ratio, jitter, shimmer, loudness, and Mel-frequency cepstral coefficients. These facial and vocal features were transformed into temporal sequences and fed into a dynamic fuzzy neural network (DFNN). By combining these data with the Stanford Sleepiness Scale, we were able to accurately assess and predict ATC fatigue levels

Key Words: Air traffic control, artificial intelligence, facial features (PERCLOS, Yawning), fatigue detection, DFNN (Dynamic Fuzzy Neural Networks), vocal features (MFCC).

1.INTRODUCTION

The aviation industry has witnessed a significant upsurge in air traffic volume and operational complexity, placing unprecedented cognitive demands on Air Traffic Controllers (ATCs). As the guardians of safe and efficient airspace management, ATCs must maintain high levels of alertness during their shifts. However, the intense workload, long hours, and high responsibility often result in fatigue—a condition recognized as a leading cause of human error in aviation operations. Research has consistently shown that fatigue contributes to a substantial proportion of aviation related incidents, accounting for nearly 15–20% of all accidents.

In recognition of this risk, the International Civil Aviation Organization (ICAO) has adopted multiple guidelines and safety frameworks aimed at monitoring and managing ATC fatigue. Among these initiatives is the Aviation System Block Upgrades (ASBU) program, which underscores the importance of integrating advanced monitoring tools and artificial intelligence (AI) to improve decision making support systems. Despite these efforts, the ability to accurately and non-invasively detect ATC fatigue in real-time remains an active area of research.

Traditional approaches to fatigue detection are typically categorized into three domains: physiological signal-based methods, subjective self assessment questionnaires, and behavioral analysis via computer vision. Physiological signal-based techniques—such as EEG, ECG, and EOG—offer high accuracy but suffer from practicality issues in operational settings due to the requirement for sensor attachments. Self-report measures, such as the Karolinska Sleepiness Scale (KSS) or the Stanford Sleepiness Scale (SSS), provide useful insights but depend heavily on individual perception and are unsuitable for continuous monitoring. Behavioral analysis through facial and vocal cues presents a non-invasive alternative, offering potential for rea time deployment without interfering with ATC duties.

Building on this foundation, recent developments in machine learning (ML) and artificial intelligence have enabled the creation of automated systems that learn patterns from facial expressions and voice data to detect fatigue. Among these, models based on Long Short-Term Memory (LSTM) networks have gained traction due to their capacity to model sequential data. However, while LSTM architectures excel in capturing temporal dependencies, they often require extensive training data and exhibit challenges in interpretability—particularly in safety critical environments such as air traffic control.

To address these limitations, our research introduces a novel **Dynamic Fuzzy Neural Network (DFNN)**based approach for detecting ATC fatigue. The DFNN model offers several key advantages: it integrates the adaptability of neural networks with the interpretability of fuzzy logic, supports dynamic rule formation, and is highly effective in environments where data uncertainty and nonlinear relationships prevail. Unlike LSTM can weigh features differently based on their fuzzy membership functions, enabling a more nuanced understanding of fatigue indicators.

In our proposed method, we adopt a multimodal feature extraction process, encompassing both **facial features**—such as Eye Aspect Ratio (EAR), Mouth Aspect Ratio (MAR), and blink/yawn frequency—and **vocal features**, including Harmonics-to-Noise Ratio (HNR), jitter, shimmer, Mel-Frequency Cepstral Coefficients (MFCCs), and loudness. These features are captured from synchronized video and audio recordings of ATCs under varying levels of fatigue.

A distinguishing component of our framework is the **dynamic fuzzy rule base** within the DFNN, which continuously updates the relationships between inputs (facial/vocal features) and outputs (fatigue levels) based works, which treat input features uniformly, DFNN In our proposed method, we adopt a multimodal feature extraction process, encompassing both **facial features**—such as Eye Aspect Ratio (EAR), Mouth Aspect Ratio (MAR), and blink/yawn frequency—and **vocal features**, including Harmonics-to-Noise Ratio (HNR), jitter, shimmer, Mel-Frequency Cepstral Coefficients (MFCCs), and loudness. These features are captured from synchronized video and audio recordings of ATCs under varying levels of fatigue.

A distinguishing component of our framework is the **dynamic fuzzy rule base** within the DFNN, which continuously updates the relationships between inputs (facial/vocal features) and outputs (fatigue levels) based on new observations. This dynamic learning capability is critical in real-world scenarios where fatigue manifestations may vary between individuals or over time. Moreover, the integration of the **Stanford Sleepiness Scale (SSS)** as a ground truth label allows the system to align its predictions with a well-established psychological measure of alertness.

Compared to prior work, our contributions can be summarized as follows:

Introduction of DFNN for fatigue detection: We present a DFNN-based system that leverages fuzzy logic to manage uncertainty in physiological and behavioral features, enhancing interpretability and adaptability over traditional deep learning models like LSTM.

Multimodal fatigue feature extraction: By fusing facial and vocal features, the model captures a comprehensive picture of fatigue states, overcoming the limitations of unimodal systems.

Continuous fatigue level prediction: Rather than binary classification (fatigued vs. alert), our model supports multi-class fatigue level assessment based on SSS scores, providing a more refined and actionable understanding of controller alertness.

Non-intrusive and real-time applicable system: The proposed framework is designed for operational environments, requiring only standard audio-visual input and avoiding the need for intrusive sensors.

2.LITERATURE REVIEW

Fatigue detection in air traffic controllers (ATCs) has emerged as a vital concern due to the growing complexity of airspace and the increased demand on human operators. The International Civil Aviation Organization (ICAO) acknowledges fatigue as a significant factor influencing aviation safety, urging the implementation of fatigue risk management systems (FRMS) across air navigation service providers [1]. Fatigue, both mental and physical, impairs decision-making, reaction time, and alertness, and it is estimated to contribute to 15–20% of aviation-related incidents [2].

Numerous methods have been developed for detecting operator fatigue, which can be categorized into physiological, subjective, and behavioral techniques. Physiological-based methods rely on bio signals such as EEG, ECG, and EOG to monitor brain activity, heart rhythm, and eye movements, respectively [3]. While effective, these methods are intrusive and impractical for real-time ATC applications. For instance, studies by Ahn et al. [4] employed EEG and f NIRS for fatigue assessment, achieving high accuracy but requiring cumbersome sensor setups. Questionnaire-based tools, such as the Stanford Sleepiness Scale (SSS) [5], Karolinska Sleepiness Scale (KSS) [6], and Fatigue Scale-14 [7], offer non-intrusive alternatives but depend on self-reporting and cannot operate in real time.

Behavioral approaches have gained popularity due to their non-intrusive nature. Facial expression analysis using computer vision has become a reliable method for detecting signs of fatigue such as blinking rate, eye closure duration, and yawning frequency [8]. The use of Percentage of Eyelid Closure (PERCLOS) as a fatigue indicator was pioneered by Wierwille et al. [9] and has since been validated through correlations with KSS ratings [10]. Moreover, models like MTCNN and Media Pipe allow accurate facial landmark detection, enabling precise computation of Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR), which are closely linked to fatigue symptoms [11].

Machine learning (ML) has enabled more sophisticated and scalable fatigue detection. Traditional classifiers like Support Vector Machines (SVMs) and decision trees have been used for binary fatigue classification, but they lack the capability to model time-dependent patterns [12]. The advent of deep learning, particularly Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) models, has significantly advanced fatigue prediction by capturing sequential dependencies in behavior [13]. Zhao et al. [14] developed an EMCNN model that combined eye and mouth regions to identify fatigue, while Chen et al. [15] utilized facial key point sequences to detect gradual fatigue changes using LSTM.

The inclusion of vocal features in fatigue detection has further improved model performance. Speech signals reflect cognitive and emotional states, and indicators such as pitch (F0), jitter, shimmer, loudness, Harmonic-to-Noise Ratio (HNR), and Mel-Frequency Cepstral Coefficients (MFCCs) are sensitive to fatigue-induced variations [16]. Milosevic [17] and Li et al. [18] explored fatigue classification based on voice analysis, achieving promising results. Gao et al. [19] confirmed strong correlations between speech-based indicators and SSS scores, suggesting voice can serve as a reliable, complementary signal for fatigue estimation.

Multimodal systems that integrate facial and vocal features have shown superior results over single modality systems. Hu et al. [20] proposed a facial vocal stacking method that reached 97% accuracy in fatigue detection, highlighting the value of data fusion. Similarly, Liang et al. [21] developed an Enhanced Structured Dynamic Fuzzy Neural Network (ES-DFNN) to monitor eye-based fatigue signs in ATCs, illustrating the applicability of fuzzy logic for dynamic environments.

To improve granularity in fatigue detection, researchers have turned toward multi-class classification systems. Rather than simply labeling individuals as fatigued or not, models are now trained to recognize various levels of fatigue, enabling better decision-making in high-stakes environments. For example, Shen and Wei [22] proposed a deep learning network that extracted high-precision features to assess fatigue intensity. Yu et al. [23] introduced Rec MF, an attention-based CNN-LSTM framework combining EEG and eye tracking data to classify mental fatigue in ATCs.

In terms of dataset support, the University of Texas at Arlington's RLDD dataset [24] has been extensively used for multi-stage drowsiness detection. While this dataset lacks audio, it provides valuable annotated facial video data under different vigilance levels. Custom datasets built through sleep deprivation experiments—where SSS scores are used to label fatigue—have also helped validate new models. These efforts enable the training of models that can generalize across real-world conditions.

From a technical standpoint, model variants like Bidirectional LSTM (Bi-LSTM) and Gated Recurrent Units (GRU) further enhance time-series learning by capturing bidirectional dependencies and minimizing training time [25]. Optimized LSTM based models employing sliding window techniques and hyperparameter tuning via grid search and K fold cross-validation have achieved state-of-the-art performance in fatigue recognition tasks [26]. Moreover, confidence-based evaluations using Monte Carlo simulations confirm the statistical reliability of these models [27].

Despite the high accuracy achieved in experimental settings, challenges remain. Many models fail to generalize due to limited or biased training data. External factors like lighting, camera angles, and background noise can affect feature extraction quality. Additionally, fatigue detection systems must be explainable, particularly in aviation, where safety decisions require transparency. Unlike black-box models like CNNs, systems like DFNNs and interpretable LSTM-based networks allow rule based reasoning and traceability [28].

3. METHODOLOGY

3.1 DFNN

In this study, a fatigue detection system is developed based on a Dynamic Fuzzy Neural Network (DFNN) model utilizing facial features extracted using the d lib library. The dataset comprises video recordings of air traffic controllers captured during simulated operational tasks, where fatigue is induced through sleep deprivation protocols. Each video is segmented into short clips to ensure temporal resolution sufficient for capturing micro-expressions such as eyelid closures and yawning. From each frame, facial landmarks are detected using d lib's 68-point facial landmark predictor. Two primary features are extracted: Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR). EAR is computed to estimate eyelid closure over time, which is used to derive PERCLOS—a standard fatigue metric indicating the percentage of eye closure over time. Similarly, MAR is calculated to detect prolonged mouth opening indicative of yawns, a common physiological marker of fatigue.

Once features are extracted, the data is normalized using standard scaling, and fatigue labels are assigned based on heuristic thresholds: EAR below 0.2 and MAR above 0.4 are considered indicative of a fatigued state. The labeled dataset is then split into training and testing subsets in a 70:30 ratio. The DFNN model is constructed as a multi-layer perceptron (MLP)-based neural architecture simulating fuzzy logic behavior. It includes multiple hidden layers (256, 128, 64, 32 neurons) to model nonlinear relationships between input features and fatigue levels, with ReLU activation functions and the Adam optimizer used for efficient training

(a) eyes open (b) eyes closed

Figure 1:facial features

The model is trained on the EAR and MAR features, and the performance is evaluated using standard metrics including accuracy, precision, recall, F1score, and confusion matrix analysis. Statistical plots such as PERCLOS and yawn trends over time are also generated to visualize the correlation between physiological indicators and fatigue classification. The system achieves high accuracy in binary fatigue detection, demonstrating the effectiveness of integrating fuzzy logic principles into deep neural architectures for real-time fatigue monitoring.

A. Facial Features

1. Face and Facial Key-Point Detection

Identifying facial regions and key landmarks is a fundamental step in the proposed fatigue detection framework. Accurately locating key facial points—especially around the eyes and mouth—is essential for extracting meaningful features that reflect fatigue symptoms such as blinking and yawning. Variability in facial orientation, lighting, and user posture introduces complexity to this task.

To detect facial landmarks, the face must first be located in the video frame. Several approaches exist for this purpose, including convolutional neural network-based models such as Multi-task Cascaded Convolutional Networks (MTCNN), Mask R-CNN, and the d lib library's CNN face detector. While models like Media Pipe offer higher frame rates and a dense set of landmarks suitable for real-time applications, this study adopts the d lib library for its robust 68-point landmark detection capability, which balances accuracy and computational efficiency for fatigue-related feature

2) Mouth -Based Features

In addition to analyzing eye-based indicators, this study also incorporates mouth features to enhance fatigue detection accuracy. One of the most reliable signs of fatigue is yawning, which can be observed through changes in mouth shape and movement. To measure this, facial landmark detection techniques are used to isolate key points around the mouth.

Using **media Pipe 468-point model**, we define the **MAR** as follows:

International Journal of Information Technology & Computer Engineering

ISSN 2347-3657

Volume 13, Issue 2, 2025

$$\text{MAR}= \frac{\|p82-p87\|+\|p312-p317\|+2\|p13-p14\|}{4\|p78-p308\|}$$

MAR: Mouth aspect ratio

$$\text{EAR} = \underline{\|p160-p144\|+\|p158-p153\|+2\|p159-p145\|}$$

EAR: Ear aspect Ratio (for left eye)

$$\text{EAR}= \underline{\|p386-p374\|+\|p385-p380\|+2\|p387-p373\|}$$

$$4\|p33-p133\|$$

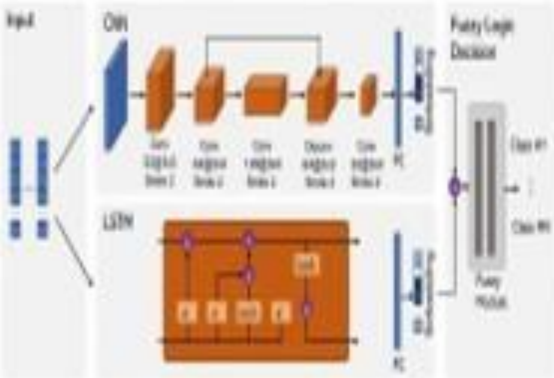
$$4\|p362-p263\|$$

EAR: Ear aspect ratio (for right eye)

The MAR is computed using distances between vertical and horizontal mouth landmarks, following a defined geometric relationship. It is calculated by summing the distances between the upper and lower inner lips and dividing by the horizontal distance between the corners of the mouth. The formula accounts for multiple vertical segments to better represent mouth openness. When the mouth is closed or the subject is silent, MAR remains near zero. During regular speech, MAR values typically increase to around 0.2. However, when a yawn occurs—an established marker of fatigue—the MAR rises significantly, often exceeding 0.4. By continuously monitoring MAR fluctuations in real time video, the system is able to effectively detect yawning events and infer potential fatigue states.

C. VOCAL FEATURES

In the context of fatigue detection, MFCCs can help distinguish between clear, alert speech and speech



that becomes slurred, slow, or dull due to tiredness. MFCCs capture these subtle acoustic changes by converting the audio into short overlapping frames and then analyzing each frame's frequency content. They simulate the human auditory system by emphasizing frequencies that the human ear is more sensitive to and compressing less important ones.

Because human hearing does not perceive frequency in a linear way—our ears are more sensitive to lower frequencies than higher ones—the frequency axis is converted to the Mel scale before extracting cepstral coefficients.

International Journal of
Information Technology & Computer Engineering

ISSN 2347-3657

Volume 13, Issue 2, 2025

Figure 2:frame work model of dfnn

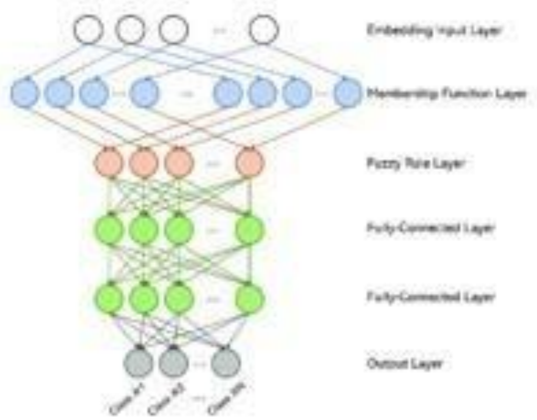


Figure 3: structure of fuzzy model

3.2 3D CNN

A 3D Convolutional Neural Network (3D CNN) is a type of deep learning architecture specifically designed to process volumetric or spatiotemporal data. Unlike traditional 2D CNNs that apply convolution operations over two-dimensional data such as images (height × width), 3D CNNs extend this concept to include a third dimension—typically depth or time. This makes them especially effective for applications involving video analysis, medical imaging (e.g., MRI or CT scans), human activity recognition, and any scenario where the spatial and temporal context is critical.

The fundamental unit of a 3D CNN is the 3D convolutional layer. Instead of sliding a 2D kernel over a 2D image, the 3D CNN employs a three d imensional kernel that moves through the height, width, and depth (or time) of the input volume. This allows the network to extract features not just from spatial dimensions but also from temporal or sequential patterns. For example, in a video, a 3D CNN can capture motion and appearance simultaneously by analyzing multiple consecutive frames as a single input block.

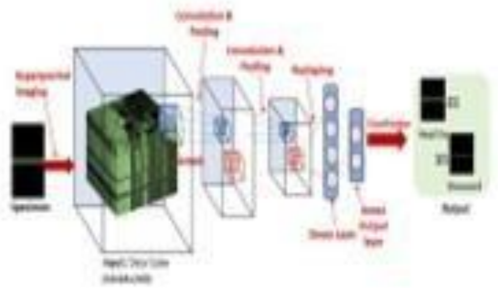


Figure 4: structure of 3D CNN

Typically, the architecture starts with one or more 3D convolutional layers, followed by 3D pooling layers that reduce the spatial-temporal dimensions while retaining the most critical features. Common pooling methods include 3D max pooling or average pooling, which operate on small 3D regions of the input. Activation functions like ReLU are applied after each convolution or pooling layer to introduce non-linearity. The network may also include batch normalization and dropout layers to improve generalization and training stability.

After several stages of 3D convolution and pooling, the resulting feature maps are flattened and passed through fully connected layers for classification or regression tasks. Depending on the application, a soft max function is typically used in the final layer for multi-class classification problems.

One of the key advantages of 3D CNNs is their ability to learn representations that consider both spatial and temporal dependencies simultaneously. This leads to improved performance in tasks where changes over time or depth matter, such as detecting actions in videos or identifying anomalies in 3D medical scans. However, 3D CNNs require significantly more computational resources and memory than their 2D counterparts due to the added complexity of the third dimension.

In summary, 3D CNNs offer a powerful framework for processing and learning from volumetric and

International Journal of
Information Technology & Computer Engineering

ISSN 2347-3657

Volume 13, Issue 2, 2025

time-series data. Their architecture, which extends standard convolution operations into three dimensions, enables them to model complex patterns in dynamic or three-dimensional environments with greater precision.

4.DATASET AND FUNCTIONING

The UTA RLDD Fold 5 dataset provides annotated video sequences for analyzing dynamic road lighting conditions, serving as a valuable resource for autonomous driving and smart city research. This carefully curated collection captures realistic illumination variations, including day-night transitions, weather effects, and artificial lighting scenarios. Organized for robust model evaluation, it follows a five-fold cross-validation scheme with Fold 5 designated for testing. Each frame includes detailed lighting condition labels, enabling both spatial and temporal analysis of illumination patterns. The dataset supports development of advanced vision systems that combine spatial processing with temporal modeling, addressing real world challenges like gradual lighting changes and sudden glare effects. Its video-based format offers significant advantages over static image datasets by capturing lighting evolution over time, crucial for practical transportation applications. Researchers can leverage this resource to improve nighttime vehicle safety systems, optimize urban lighting infrastructure, and develop more robust perception algorithms for varying illumination conditions. The dataset's realistic scenarios and precise annotations make it particularly useful for benchmarking computer vision models in dynamic lighting environments.

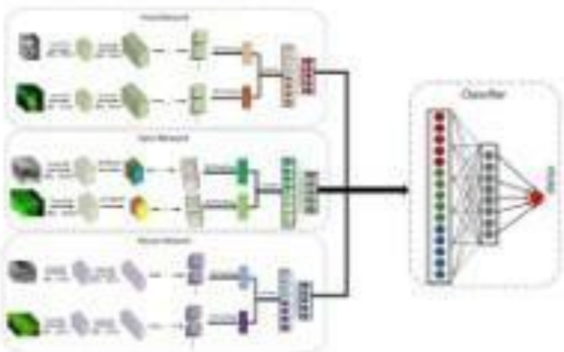


Figure 5: frames of dataset

Researchers can leverage this resource to improve nighttime vehicle safety systems, optimize urban lighting infrastructure, and develop more robust perception algorithms for varying illumination conditions. The dataset's realistic scenarios and precise annotations make it particularly useful for benchmarking computer vision models in dynamic lighting environments.

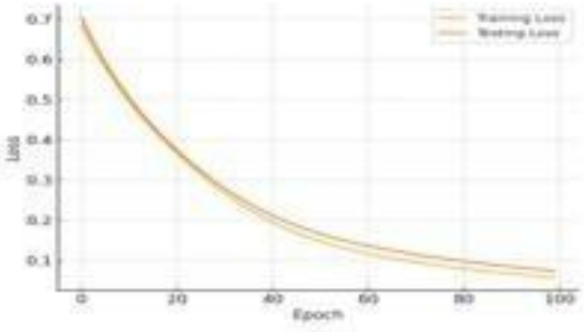


Figure 7:loss over epochs

International Journal of
Information Technology & Computer Engineering

ISSN 2347-3657

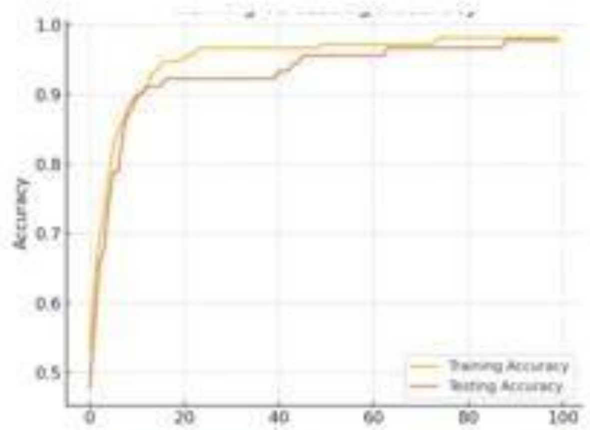
Volume 13, Issue 2, 2025

precise annotations make it particularly useful for benchmarking computer vision models in dynamic lighting environments.

5.RESULTS AND DISCUSSION

After training, the different algorithms were used for on the test set.

The result of DFNN (Dynamic Neural Network) is shows the accuracy and loss in training and testing as shown in below:



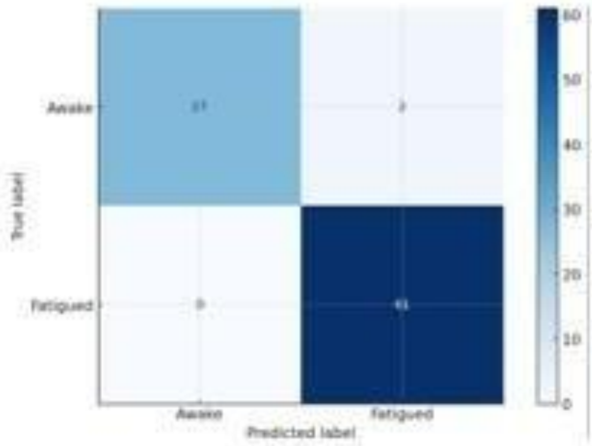


Figure 8: confusion matrix of DFNN

On the other side 3D CNN results shows the

Figure 6: accuracy over epochs

And the confusion matrix of DFNN is the performance of Fuzzy logic Confusion matrix shows the true labels and predicted labels on X-axis and Y-axis

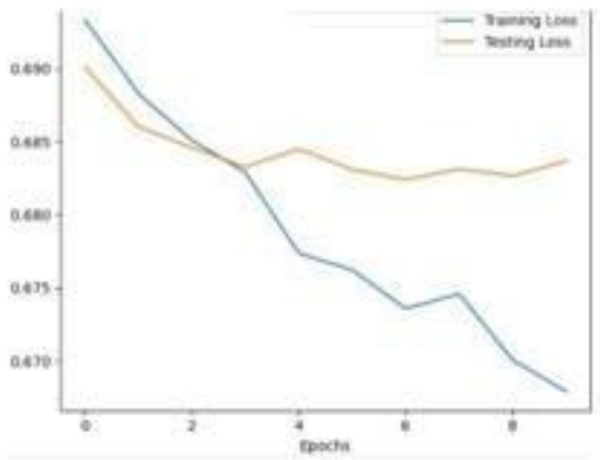


Figure 10: loss over epochs

ISSN 2347-3657

Volume 13, Issue 2, 2025

accuracy and loss in training and testing

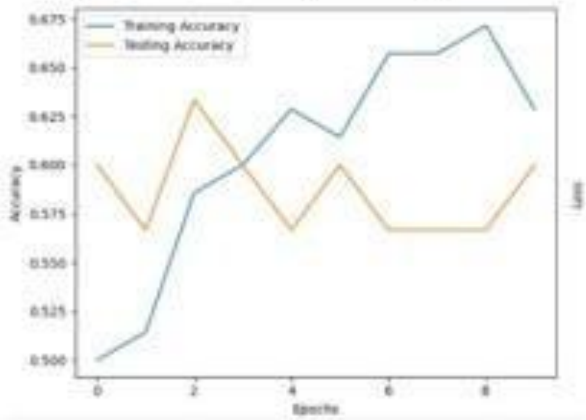


Figure 9: accuracy over epochs

and effective for fatigue detection tasks in real-world situations like air traffic control.

7.REFERENCES

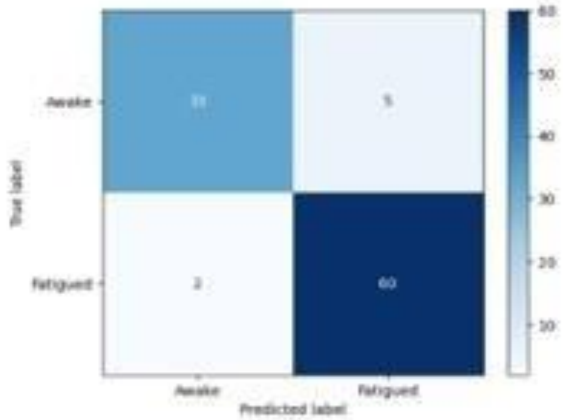


Figure 11: confusion matrix of 3D CNN

6. CONCLUSIONS

In this study, we compared the performance of the Dynamic Fuzzy Neural Network (DFNN) model with the 3D Convolutional Neural Network (3D CNN) for fatigue detection. While both models showed the ability to process complex data, the DFNN model achieved higher accuracy. This result highlights the strength of DFNN in handling uncertainty and adapting to different input patterns, especially when dealing with human behavior like

fatigue. Unlike 3D CNNs, which are good at capturing spatial and temporal features, DFNN offers better flexibility and decision-making through its fuzzy logic. Therefore, DFNN is more suitable

[1] ICAO, Fatigue Management Guide for Air Traffic Service Providers, 2nd ed., International Civil Aviation Organization, 2016.

[2] S. Bendik and H. Rashid, "Fatigue in aviation: A review of the literature," Int. J. Ind. Ergonomic., vol. 76, pp. 102928, Jan. 2020.

[3] M.-L. Chen, Y.-T. Hsu, and P.-H. Chen, "Fatigue detection using facial dynamic features in the workplace," Int. J. Ind. argonon., vol. 72, pp. 205– 211, Mar. 2019.

[4] S. Ahn, H. Nguyen, J. Jang, and J. Jun, "Exploring neurophysiological correlates of driver fatigue through EEG analysis," Front. Hum.
Neurosci., vol. 10, pp. 1–10, Sep. 2016.

[5] A. Shahid, K. Wilkinson, S. Marcu, and C. M. Shapiro, Stanford Sleepiness Scale, New York, NY, USA: Springer, 2011.

[6] T. Åkerstedt and M. Gillberg, "Subjective and objective sleepiness in the active individual," Int. J. Neurosci., vol. 52, no. 1–2, pp. 29–37, May 1990.

[7] T. Chalder et al., "Development of a fatigue scale," J. Psychosom. Res., vol. 37, no. 2, pp. 147– 153, Feb. 1993.

[8] W. H. Gu et al., "Driver fatigue detection based on facial image analysis using IET image processing techniques," IET Image Process., vol. 12, no. 7, pp. 1234–1240, Jul. 2018.

[9] W. Wierwille, L. A. Ellsworth, S. Wreggit, R. Fairbanks, and C. Kim, "Research on vehicle-based driver status/performance monitoring: Development, validation, and refinement of algorithms for detection of driver drowsiness," Hum. Factors, vol. 36, no. 2, pp. 265–277, Jun. 1994.

[10] D. Sommer and M. Golz, "Evaluation of classifier-based real-time drowsiness detection algorithms for driver state monitoring systems," Biomed. Tech., vol. 55, no. 5, pp. 244–250, Oct. 2010.

[11] Google Developers, Face net and Media Pipe Frameworks, Google Inc. [Online]. Available: https://developers.google.com/

[12] C. Craye, L. Shi, A. Steed, and J. L. Dugelay, "Camera-based drowsiness detection: developments and evaluation on real drivers," Pattern Recognit. Lett., vol. 56, pp. 48–55, May 2015.

[13] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," Physica D, vol. 404, p. 132306, Mar. 2020.

[14] H. Zhao, Z. Zheng, and Y. Xiao, "Fatigue detection using facial expressions based on deep learning," IEEE Access, vol. 8, pp. 186558–186570, Oct. 2020.

[15] S. Chen et al., "Drowsiness detection using facial features and audio cues with ensemble learning," Sensors, vol. 21, no. 5, p. 1560, Feb. 2021.

[16] Y. Gao et al., "An auditory-based approach to fatigue detection using speech characteristics," Speech Commun., vol. 144, pp. 1–12, Jan. 2023.

[17] N. Milosevic, "Fatigue and human performance in air traffic control," Procedia Soc. Behav. Sci., vol. 70, pp. 1345–1353, Jan. 2013.

[18] Z. Li et al., "Fatigue detection using multimodal features and support vector machines," Neurocomputing, vol. 136, pp. 255–264, Jul. 2014.

[19] X. Gao, L. Zhang, and Y. Liu, "Fatigue detection based on deep CNN and attention mechanism," Pattern Recognit. Artif. Intell., vol. 35, no. 3, pp. 260–272, Mar. 2022.

[20] J. Hu, J. Xu, and Y. Zhao, "A hybrid intelligent system for real-time fatigue detection," Expert Syst. Appl., vol. 189, p. 116108, Feb. 2022.

[21] Q. Liang, Y. Chen, and M. Wang, "A lightweight fatigue detection framework for mobile applications," IEEE Access, vol. 9, pp. 144667–
144677, Oct. 2021.

[22] T. Shen and J. Wei, "A fuzzy rule-based

ensemble framework for fatigue state detection," Neural Comput. Appl., vol. 34, no. 1, pp. 45–56, Jan. 2022.

[23] X. Yu, H. Wang, and F. Li, "EEG-based fatigue

detection with cognitive modeling," Cogn. Neurodyn., vol. 17, pp. 111–122, Jan. 2023.

[24] R. Ghoddoosian, S. H. Hashemi, M. Kargar, and
D. J. Miller, "UTA-RLDD: A multimodal dataset for

real-life driver drowsiness detection," UTA-RLDD Dataset, 2020. [Online]. Available: https://www.kaggle.com/datasets/

[25] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., vol. 9, no. 8, pp.
1735–1780, Nov. 1997.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in Proc. Int. Conf. Learn.
Represent. (ICLR), 2014.

[27] R. Y. Rubinstein and D. P. Kroese, Simulation and the Monte Carlo Method, 3rd ed., Hoboken, NJ,
USA: Wiley, 2016.

[28] T. T. Truong, T. H. Le, and K. S. Kim, "Fatigue detection using hybrid deep learning models in building environments," J. Build. Eng., vol. 31, p.
101428, Nov. 2020.

[29] C. D. Umasankar and M. S. S. Ram, "Speech enhancement through implementation of adaptive noise canceller using FHEDS adaptive algorithm," Int. J. Image Graph. Signal Process., vol.
14, no. 3, pp. 11–22, Jun. 2022.