Two-Level Hierarchical Adaptive Dynamic Fusion for CNN–LSTM Integration in Fatigue Level Prediction

Marlince NK Nababan[1]*, Poltak Sihombing[2], Erna Nababan[3], T Henny Febriana Harum[4]
Student of Doctoral Program in Computer Science[1]
Department of Computer Science-Faculty of Computer Science and Information Technology,
Universitas Sumatera Utara, Medan, Indonesia[1, 2, 3,4]

***Abstract*—Driver fatigue is a major contributor to traffic accidents, yet most existing detection systems rely on unimodal inputs or static fusion mechanisms that lack robustness under poor lighting, partially obscured faces, and missing sensor data. This study aims to overcome these limitations by proposing a Hierarchical Adaptive Dynamic Fusion (HADF) model. HADF integrates a two-level adaptive fusion mechanism combining a CNN (ResNet-18) for facial micro-expressions and an LSTM for physiological signals (heart rate, temperature, and accelerometer). The first stage computes adaptive intra-modality weights (α), while the second stage assigns inter-modality weights (γ), enabling context-aware and resilient multimodal integration even under missing-modality conditions. Experiments on a multimodal fatigue dataset show that HADF achieves a validation accuracy of 96.5%, a macro F1-score of 0.96, and ROC-AUC values of 1.00 (Normal), 0.99 (Eye-Closed), and 0.93 (Yawn). Compared with unimodal and static-fusion baselines, HADF improves accuracy by approximately 4.5% and macro F1-score by 6–9%, while maintaining stable performance under incomplete data. These results confirm the novelty of HADF as a two-stage adaptive fusion strategy that enhances accuracy and system robustness, making it suitable for real-time fatigue monitoring in transportation, occupational safety, and healthcare applications.**

***Keywords*—Multimodal fusion; adaptive dynamic fusion; CNN-LSTM; fatigue level prediction**

I. Introduction

Fatigue is a significant factor that reduces concentration, reflexes, and alertness, thus contributing significantly to traffic and occupational accidents. WHO and ILO data (2021) recorded more than 398,000 global deaths per year related to fatigue. In Indonesia, the KNKT (2022) report indicated that fatigued drivers were responsible for more than 20% of land transportation accidents. This underscores the importance of a thorough, adaptive, and real-time fatigue detection system.

In recent years, deep learning-based approaches such as CNNs (Convolutional Neural Networks) and RNNs (Recurrent Neural Networks) have been used to process facial images and physiological signals to detect fatigue. CNNs are effective at capturing spatial patterns (such as facial expressions), while RNNs—especially LSTMs—can analyze temporal patterns in sensor data (such as heart rate or body temperature). Therefore, early detection of fatigue levels has become a crucial focus in many recent studies. Single-feature-based methods (e.g., those focusing only on the eyes or only on the mouth) are prone to failure when the face is partially covered (by glasses/mask) or in poor lighting conditions. Various deep learning-based approaches have been developed, such as CNN for spatial features (actualization) and LSTM for temporal features (physiological frequencies). However, previous research still has limitations, namely that single-feature methods are prone to failure when the face is obscured or in poor lighting conditions. [1]. CNN-RNN has a high computational cost due to the complexity of its data and examples, which require substantial computing power. Furthermore, this example is often used for large datasets, such as video or other sequential data, which require significant computing power. As a result, training can be slow, especially with large and complex datasets [2].

Currently, many prediction methods are employed, such as deepening the model to become more complex and adding parameters, which can lead to problems, including a less stable model (less durable), inflexibility, and a one-dimensional approach (failing to capture enough features). The model is not robust to data variations or noise. Limited flexibility: difficult to apply to various types of materials or conditions. One-dimensional approaches are limited to a single category or feature [3]. Learning-based techniques have been widely used to process data from various sources, namely facial images, physiological signals (EEG, EOG), and time data, which have been analyzed through multiple learning-based methods. Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) models can extract spatial features from images and capture temporal relationships in time series data. Learning-based techniques have been widely used to process data from various sources, namely facial images, physiological signals (EEG, EOG), and time data, which have been analyzed through multiple learning-based methods. Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) models can extract spatial features from images, while also capturing temporal relationships in time series data [4].

The combination of hybrid CNN-LSTM models still has limitations in terms of flexibility and adaptability to complex data. Developing a system to assess a person's fatigue and physical load by combining various subjective and objective data. Measuring physical load and fatigue using traditional methods is ineffective and inaccurate. Expanding machine learning applications that can monitor and predict fatigue [5].

*Corresponding author.

The hybrid CNN–LSTM has been widely used, but its weakness lies in its non-adaptive, static fusion. In baseline trials, static fusion accuracy only reached 92% and dropped to 80% when one modality was missing [6].

This study demonstrates that a CNN-LSTM-based multimodal fusion approach can non-invasively enhance driver stress detection accuracy by up to 95.5% using eye data, vehicle dynamics, and environmental variables. However, this study is limited by the fact that stress labels are based on participants' subjective assessments of their own experiences. Furthermore, other studies have employed a single-feature fusion mechanism without hierarchical fusion and have not considered models with noise or missing modalities [7].

The non-invasive multi-index fusion and CNN–BiLSTM approaches can achieve 98.2% accuracy, but the fusion remains static and single-level. As a result, these methods are susceptible to noise and deficiencies in modality [8]. Another study [9][10][11] shows that the hybrid CNN-RNN model improved accuracy. However, limited generalization, high complexity, and the lack of an adaptive fusion mechanism remain issues.

In summary, existing multimodal fatigue detection approaches still rely on static or single-stage fusion, which limits their ability to adapt to variations in data quality, environmental conditions, and missing modalities. These constraints reduce the robustness and generalization of current CNN-LSTM-based frameworks. To address this gap, this study proposes the Two-Level Hierarchical Adaptive Dynamic Fusion (HADF) model, which integrates dynamic intra-modality weighting (α) and inter-modality weighting (γ) to enable context-aware, flexible, and resilient multimodal feature integration. The contributions of this work include: 1) introducing a hierarchical and adaptive fusion mechanism for non-invasive fatigue prediction, 2) designing a hybrid CNN–LSTM model capable of handling incomplete or degraded multimodal inputs, and 3) providing comprehensive experimental evaluations demonstrating improved accuracy, stability, and robustness compared to unimodal and static-fusion baselines. The subsequent sections detail the model architecture, experimental methodology, evaluation metrics, and performance analysis.

II. Materials and Methods

A. Fatigue Level

Fatigue is a condition that can affect performance and safety across many jobs. Predicting fatigue levels is crucial for preventing accidents and increasing productivity. Numerous studies have used multiple approaches to predict fatigue levels. The condition characterized by decreased cognitive and motor function due to sustained fatigue is called fatigue. In the areas of work, transportation, and health, it is crucial to recognize and address fatigue. In [12], the authors have shown that combining facial images with physiological signals such as heart rate, temperature, and rPPG can significantly improve fatigue prediction.

The aim is to compile and present the latest research on fatigue analysis, covering knowledge of fatigue analysis theory, fatigue life prediction methods, and design and application techniques. Another aim is to provide a comprehensive overview of the progress and challenges achieved in the field. Furthermore, the researchers propose a path forward to improve the reliability and effectiveness of fatigue life prediction methods in industry. Since the combination of existing approaches is insufficient to address the increasingly complex challenges, this study demonstrates the urgent need for the development of new technologies. Therefore, new solutions for fatigue analysis must be developed to meet current industry needs [13].

One of the Prediction Model Development is building a deep learning-based model, namely Deep Belief Neural Network-Back Propagation (DBN-BP), to predict the fatigue life of Ti-6Al-4V in the very high fatigue range (VHCF). Parameter influence analysis examines how process parameters, such as energy density, tensile strength, and fabrication method, impact fatigue life behavior. By using deep learning models, researchers can improve prediction accuracy, precision, and model stability compared to traditional methods and existing machine learning algorithms. However, if the model is too complex for a limited dataset, it can lead to overfitting. Deep learning models can also be very complex and risk overfitting when the number of parameters exceeds the available data.

B. Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM)

The combination of CNNs and LSTMs has been used to exploit the advantages of both spatial and temporal features simultaneously. The CNN serves as a feature extractor, and its output is then fed into the LSTM as sequential input. This hybrid model has shown promising results in the fatigue domain [14]. But it still uses static feature fusion, which limits the model's flexibility. The problem of multidimensional data remains a subject of ongoing research aimed at finding suitable models for various applications. Table I shows a critical literature analysis.

TABLE I. Critical Literature Analysis

Study /Approach	Methods	Limitation / Gap Identified
Remaining Useful Life Prediction [15]	CNN–LSTM	Prone to overfitting on small datasets, fusion remains static, and interpretation is challenging.
Industrial and Machined Surfaces	CNN–LSTM	Very sensitive to image noise, validation is limited to specific data types.
Renewable Energy and Weather [16]	CNN–LSTM	Accuracy drops sharply during extreme weather conditions, and the interpretation of spatial features is limited.
General Image Classification [17][18]	RNN–ResNet18	Small, unbalanced dataset; high potential for bias.

C. Hybrid Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM)

This approach is combined with the Hierarchical Adaptive Dynamic Fusion (HADF) method. This model uses non-invasive multimodal input from heart rate, facial video, and environmental factors to predict fatigue level (fatigue level). Making model research is shown in Fig. 1:

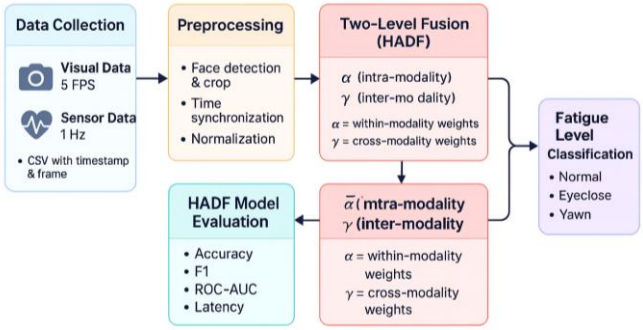


Fig. 1. Making model research.

The research began with the collection and creation of datasets from three primary data sources: visual data (facial images) used to detect expressions, closed eyes, or nervousness, and physiological sensor data used to detect changes in physical condition. A temperature sensor measures body temperature. An accelerometer measures movement or posture. The data collection method involves attaching the sensor to the subject during video recording. Sensor data is synchronized with the video recording time (timestamp). Data is stored in a CSV file. Data acquisition is shown in Fig. 2:

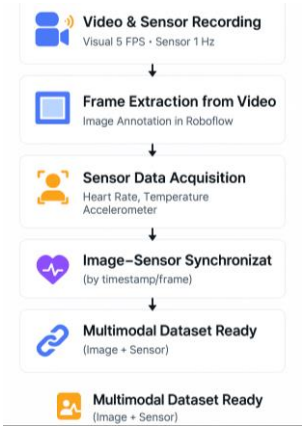


Fig. 2. Data acquisition.

Preprocessing is performed on two data sources—images (visual) and sensor signals—and then they are temporally aligned (synchronized) before being formed into sequential samples for a CNN–LSTM model with adaptive fusion (HADF). All steps are designed to minimize data leakage and maintain inter-subject consistency.

The Hierarchical Adaptive Dynamic Fusion (HADF) model was developed to address the limitations of static fusion models, which generally use simple concatenation weights (e.g., direct concatenation). The HADF model is designed to integrate visual (facial images) and physiological (e.g., heart rate, body temperature, acceleration) modalities gradually and adaptively.

D. Hierarchical Adaptive Dynamic Fusion (HADF) Architecture

The Hierarchical Adaptive Dynamic Fusion (HADF) architecture is proposed in the study. Preprocessing—collecting facial images (RGB), eye area detection (including eye landmarks), and physiological sensor data (heart rate, acceleration)—begins the process. To extract spatial features from facial images and eye landmarks, a CNN (ResNet18) is used, while an LSTM is used to extract temporal features from the sensor data. A two-stage fusion mechanism is then used to combine the feature extraction results. The first is intrafusion (α), which combines features within a single modality with dynamic relevance weights, and the second is interfusion (γ), which combines across modalities to form a more contextual and responsive multimodal representation. In the final stage, a classification layer is used to determine the fatigue level and divide it into three categories: Normal, Eyeclosed, and Yawn (see Fig. 3).

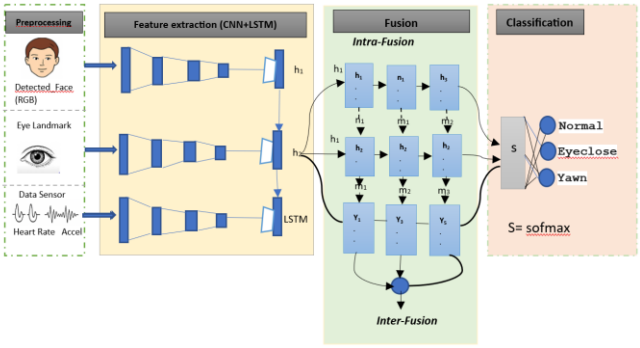


Fig. 3. HADF architecture visualization with two levels of fusion.

III. Results and Discussion

A. Model Static

Previous research proposed hierarchical fusion, which processes each modality independently and then performs level-wise fusion. Each fusion stage produces a combined representation that is fed back to the network to learn inter-modality interactions. This approach improves robustness but does not yet implement dynamic weight adaptation to changes in modality quality. Basic formula:

$$u_t = tanh(Wh_t + b) \quad (1)$$

where, the hidden state vector at time t, ht, is passed to a fully connected layer: multiplied by the weight matrix W, added to the bias b, and then activated using the tanh function to produce the hidden representation u_t.

$$\alpha_t = \frac{exp(u_t^T u)}{\Sigma_{t=1}^{n} exp(u_t^T u)} \quad (2)$$

The attention score for time t, α_t, is calculated by applying a softmax function to the alignment (dot product) between u_t and the trained context parameter vector, u. The resulting α_t is a non-negative weight that sums to 1 across all time steps (probabilities).

B. Hierarchical Adaptive Dynamic Fusion (HADF)

In this study, the proposed Hierarchical Adaptive Dynamic Fusion (HADF) model is a two-level fusion mechanism, namely intra-modality and inter-modality, with weights calculated based on temporal data and input feature characteristics. Basic formula:

- Fusi Intra-Modality: In a single diffusion modality, each feature is weighted by α, which represents its importance.

$$F_t^{(k)} = \sum_{i=1}^{N_k} \alpha_{t,i}^{(k)} \cdot f_{t,i}^{(k)} \ dengan \ \sum_{i=1}^{N_k} \alpha_{t,i}^{(k)} = 1 \quad (3)$$

The value is calculated through the softmax function of the relevance scores generated by the multilayer perceptron (MLP) module:

$$\alpha_{t,i}^{(k)} = \frac{\exp(e_{t,i}^{(k)})}{\sum_{j=1}^{N_k} exp(e_{t_j}^{(k)})} \ with \ e_{t,i}^{(k)} = MLP(f_{t,i}^{(k)}) \quad (4)$$

where,

$\boldsymbol{F_t^{(k)}}$ is the representation of the intra-modality fusion results for the kth modality at time t.

N_k is the Total number of features in the kth modality.

$f_{t,i}^{(k)}$ is the fitur-i, feature of the k-t modality at time-t

$\alpha_{t,i}^{(k)}$ Adaptive weight for the i feature in the k modality at time t.

$\sum_{i=1}^{N_k} \alpha_{t,i}^{(k)} = 1$ Normalization: all α weights in one modality sum to 1.

- Fusi Inter-Modality: The results from each modality are then combined using the weight γ in the formula.

$$F_{fused}(t) = \sum_{k=1}^{K} yk(t).F_t^{(k)} \ with \ \sum_{k=1}^{K} yk(t) = 1 \ (5)$$

The weight γ is also calculated using softmax over the relevance between modalities.

$$yk(t) = \frac{\exp(gk(t))}{\sum_{l=1}^{K} \exp(gl(t))} \ with \ gk(t) = MLP(F_t^{(k)}) \ (6)$$

$F_{fused}(t)$ is the final multimodal representation at time t, which is the combination of all features from all modalities after going through a two-level adaptive fusion process.

K is the total number of modalities used (e.g., facial image, heart rate, body temperature, accelerometer).

$\boldsymbol{N_k}$ is the Number of features in the k modality

$\boldsymbol{f_{t,i}^{(k)}}$ Intra-modality weights (level within modality) for the i feature of the k modality.

$\boldsymbol{yk(t)}$ Inter-modality weights for the k modality at time t.

C. Maths Pseudocode

Between mathematical conceptual thinking and effective programming implementation. Maths Pseudocode is a logical and methodical presentation of mathematical algorithms organized in half-code form.

- Simulation using HADF

Algorithm in Fig. 4 presents the simulation results, implemented in Python.

```
for each sequence:
  for t = 1..T:
    # Intra-modality
    for k in 1..K:
      e_k[i]   = MLP_alpha(f_k[i,t])
      alpha_k  = softmax(e_k)                    # size N_k
      F_k[t]   = Σ_i alpha_k[i] * f_k[i,t]

    # Inter-modality (mask-aware)
    for k in 1..K:
      s_k[t]   = (m_k[t]==1) ? MLP_gamma(F_k[t]) : -inf
    gamma[t]   = softmax(s_[1..K][t])            # over available modalities

    # Fusion + classify
    Z[t]       = Σ_k gamma[t][k] * (W_k * F_k[t])
    y_hat[t]   = softmax( g_clf(Z[t]) )
    L[t]       = CE(y[t], y_hat[t])
  Update params by ∇(mean_t L[t])
```

Fig. 4. Algorithm for the simulation results.

This simulation helps analyze the performance of systems with limited capacity, such as contact centers, banks, and similar systems. Fig. 5 presents the running system simulation for HADF.

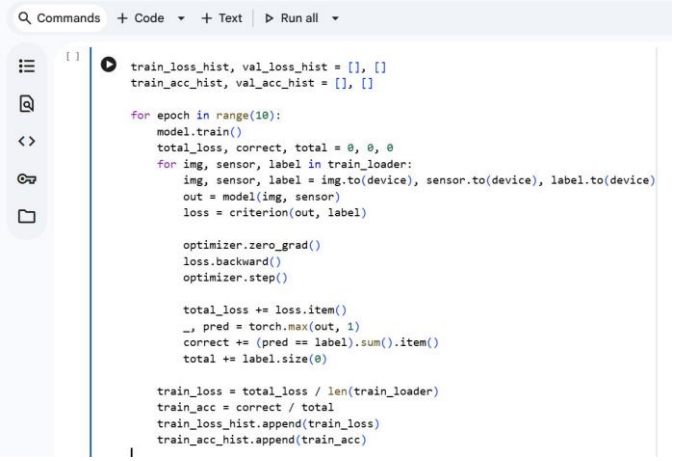


Fig. 5. Running system simulation for HADF.

The HADF model was trained using supervised learning with the cross-entropy loss function. Training was conducted over 10 epochs, comprising two main phases: training and validation. In the training phase, the model received input in the form of facial images and physiological sensor data, which were then processed through a CNN–LSTM architecture with a HADF fusion mechanism. The loss value was calculated for each batch, then the weights were updated using a backpropagation-based optimizer. Next, in the validation phase, the model was evaluated on test data without weight updates to measure its generalization ability (see Fig. 6).

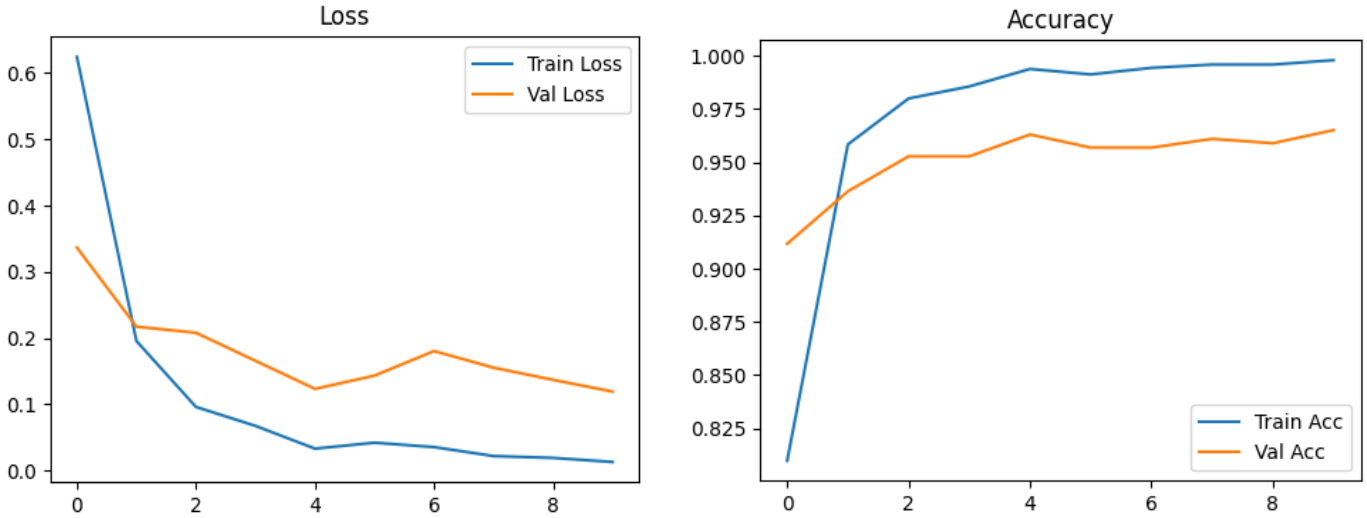
```
Epoch 1 | Train Loss: 0.6244 | Train Acc: 0.8098 | Val Loss: 0.3365 | Val Acc: 0.9117
Epoch 2 | Train Loss: 0.1953 | Train Acc: 0.9584 | Val Loss: 0.2170 | Val Acc: 0.9363
Epoch 3 | Train Loss: 0.0958 | Train Acc: 0.9799 | Val Loss: 0.2078 | Val Acc: 0.9528
Epoch 4 | Train Loss: 0.0671 | Train Acc: 0.9856 | Val Loss: 0.1654 | Val Acc: 0.9528
Epoch 5 | Train Loss: 0.0328 | Train Acc: 0.9938 | Val Loss: 0.1230 | Val Acc: 0.9630
Epoch 6 | Train Loss: 0.0417 | Train Acc: 0.9913 | Val Loss: 0.1430 | Val Acc: 0.9569
Epoch 7 | Train Loss: 0.0351 | Train Acc: 0.9943 | Val Loss: 0.1802 | Val Acc: 0.9569
Epoch 8 | Train Loss: 0.0216 | Train Acc: 0.9959 | Val Loss: 0.1550 | Val Acc: 0.9610
Epoch 9 | Train Loss: 0.0187 | Train Acc: 0.9959 | Val Loss: 0.1366 | Val Acc: 0.9589
Epoch 10 | Train Loss: 0.0126 | Train Acc: 0.9979 | Val Loss: 0.1189 | Val Acc: 0.9651
```

Fig. 6. Model training logs.

- Summary of Fig. 6:

Epoch 1: The model is still in the early stages of learning. The training loss is relatively high (0.6244), while the training accuracy is 80.9%. Fairly good accuracy (91.1%) in the validation data indicates that the model can instantly recognize basic patterns in multimodal data. Epochs 2-4: Training accuracy increases significantly (from 95.8% to 98.6%), and the value loss decreases from 0.3365 to 0.1654. Validation accuracy rises to 95.2%, indicating the model's increasing generalization ability. Epochs 5-7: The model achieves a training accuracy above 99%. Validation accuracy stabilizes at 95.6–96.3%, while the value loss remains low (0.1230–0.1802). This indicates that the model is approaching optimal conditions, though slight fluctuations in the loss value are due to the complexity of the data. Epochs 8-10: The model stabilizes, with near-perfect training accuracy (99.8%), and value accuracy peaking at 96.5% at the 10th epoch. Furthermore, the value drops to 0.1189, indicating that the model's generalization remains good without significant overfitting.

Overall, these training results demonstrate that the HADF model learns well from the dataset. The hierarchical adaptive dynamic fusion mechanism produces robust multimodal representations that balance the ability to learn from training data and generalize to new data. This demonstrates that training accuracy approaching 100% does not degrade validation performance, which remains stable between 95% and 96%. In Fig. 7(a), the simulation results obtained using Python are shown.



(a) (b)

Fig. 7. (a) Shows the simulation results (loss) obtained using Python. (b) Shows the simulation results (accuracy) obtained using Python.

Fig. 7(b) shows a gradual, consistent decrease in training and validation scores, indicating a stable learning process. Furthermore, the curves show that model accuracy increases significantly over time, with no significant differences between the training and validation phases. This indicates that the HADF model has good generalization capabilities; it recognizes patterns in the training data and can maintain them when exposed to new data. This stability suggests that the model does not overfit and can achieve ideal convergence in a short training time. On the left, the graph shows how the loss values evolve. The training loss (blue line) continues to decrease as the number of epochs increases, indicating that the model has successfully learned the patterns in the training data. The validation loss (orange line) also decreases, albeit more fluctuating, indicating that the fused representation continues to generalize well. The graph on the right side shows accuracy. The adaptive fusion result (f) obtained from the α, β, and γ mechanisms can produce an effective and robust multimodal representation that is robust to data variations while preventing overfitting. The improvement in instruction accuracy is nearly 100%, while validation accuracy remains stable at 95%-97%.

The confusion matrix indicates that the HADF model has very high classification performance across all classes, including minority classes such as 'eyeclose' and 'yawn'. The misclassification rate is very low, indicating that this model successfully overcomes the limitations of previous models, which tend to be biased towards the majority class. These results suggest that the two-level adaptive fusion strategy implemented by HADF has consistently improved the sensitivity and classification accuracy across all classes. The confusion matrix for the validation data shows the HADF model's performance in distinguishing between three classes: eyeclosed, normal, and yawn. From the prediction results, it can be seen that the model correctly recognized the eyeclose class for 147 samples and was misclassified only 5 times, as usual. In the regular class, the model correctly detected 281 samples, with relatively small errors: 7 samples were predicted as 'eye closed' and four samples as 'yawn'. Meanwhile, in the yawn class, 42 samples were correctly classified with only one error as usual. The dominant distribution of predictions lies along the main diagonal of the matrix, indicating that the model exhibits excellent classification performance, with a low error rate and high accuracy in distinguishing the three conditions, as shown in Fig. 8 (Confusion Matrix HADF).

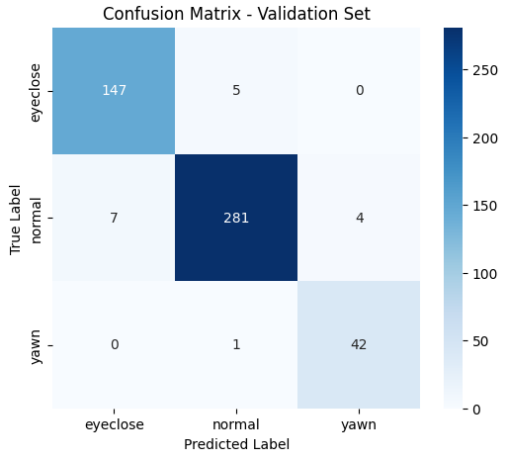


Fig. 8. Confusion matrix HADF.

- Simulation using Static Fusion: Between mathematical conceptual thinking and effective programming implementation. Maths Pseudocode is a logical and methodical presentation of mathematical algorithms organized in half-code form.

Fig. 9 presents the algorithm for the simulation results, implemented in Python.

```
class StaticFusion(nn.Module):
    def __init__(self, ...):
        super().__init__()
        self.fc = nn.Linear(total_feat_dim, num_classes)

    def forward(self, feat1, feat2, feat3):
        fused = torch.cat([feat1, feat2, feat3], dim=1)    # <--- Fusi
statis
        out = self.fc(fused)
        return out
```

Fig. 9. Pseudocode using fusion.

Confusion matrix showing the results of the fatigue detection model evaluation on the test data (test set) using the Static Fusion model. Confusion matrix showing the results of the fatigue detection model evaluation on the test data (test set). The following Fig. 10 presents the confusion matrix fusi static.

It shows the model's performance in distinguishing three fatigue condition classes: Normal, Eyeclose, and Yawn. In general, the model achieves an excellent classification rate in the Normal class, with almost perfect accuracy (69 correct instances and only one incorrect instance, which is Eyeclose). Performance in the Eyeclose class is also optimal, with all 45 cases correctly predicted, compared to other courses. However, a significant weakness is seen in the Yawn class. Of the 142 Yawn instances, 108 are predicted correctly, but 34 cases are incorrectly classified as eye closed. This indicates confusion between the yawn expression and the eye-closed condition, both of which exhibit similar visual patterns. This error can be influenced by data quantity imbalance and limitations in visual features. To overcome this, additional strategies are needed, such as data augmentation for the Yawn class, the use of weighted loss, or the utilization of physiological sensor modalities in the adaptive fusion mechanism to increase the model's robustness.

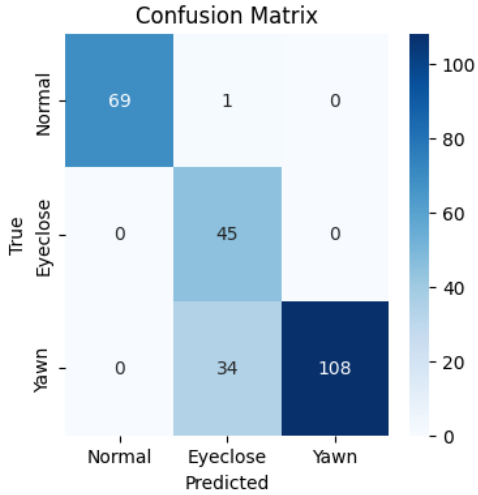


Fig. 10. Confusion matrix fusi static.

D. Comparison of Static Fusion Model Performance of HADF Model

After each model is evaluated separately, this subsection compares their performance. The purpose of this comparison is to determine whether the HADF model, which incorporates weights from the resulting classifications, can achieve significant performance improvements over the Static Fusion model approach. Table II presents a performance comparison between the HADF model and the Static Fusion model, including training and validation accuracies and loss values, to evaluate the stability and generalization capability of each model.

TABLE II. COMPARISON OF STATIC FUSION MODEL PERFORMANCE OF HADF MODEL

Aspect	HADF Model	Static Fusion Model	Analysis
Training Accuracy	Increased from 80.9% → 99.8% (epoch 10)	Increased from 80.7% → 99.6% (epoch 10)	Both models exhibit fast convergence, as they can learn the training data patterns very well.
Validation Accuracy	Stable, peak 96.5% (epoch 10)	Fluctuating, peak 95.0% (epochs 9–10)	HADF is more stable with better generalization; Static Fusion is slightly less stable.
Validation Loss	Consistently decreasing (0.3365 → 0.1189)	Initially decreased (0.3841 → 0.1869) then fluctuated (0.22–0.26)	HADF is more consistent; Static Fusion shows instability.

1) Table Analysis: The results highlight differences in model performance across two training trials — HADF and Static Fusion — focusing on four key aspects: training accuracy, validation accuracy, validation loss, and model stability.

- Training Accuracy: From around 80% in the initial epoch to nearly perfect (99.6–99.8%) in the 10th epoch, both HADF and Static Fusion show very rapid improvement.

This demonstrates the model's ability to learn patterns from the training data, and both methods show rapid convergence.

- Validation Accuracy: HADF consistently achieved a peak validation accuracy of 96.5%, whereas Static Fusion achieved only 95% with a more fluctuating trend. This indicates that HADF is better at maintaining generalization to the validation data, while Static Fusion is slightly less stable despite maintaining high accuracy.
- Loss: In Static Fusion, the value loss dropped drastically initially, but then fluctuated between 0.22 and 0.26, indicating mild overfitting. In contrast, in HADF, the value loss dropped consistently from 0.3365 to 0.1189.

E. Discussion

In summary, the differences between the previous Attention Fusion and HADF models can be summarized in the following points: 1) The Attention Fusion model computes weights for each hidden state of the LSTM in one modality. In contrast, HADF computes stepwise adaptive fusion weights, including intra-modality (spatial-temporal) and inter-modality (between modalities) weights. 2) Weight type, the previous model α_t = attention weight per time (time-step) for one modality, while HADF. α_t^m time in one modality (intra-fusion). β_t^m = Temporal attention weight per modality, y_m = weight between modalities (inter-fusion). 3) Fusion level, the previous model uses a single-level (directly from hidden states to aggregation), while HADF Multi-level: intra-modality-temporal pooling-inter-modality.

IV. CONCLUSION

The Two-Level Hierarchical Adaptive Dynamic Fusion (HADF) formulation represents a novel and scientifically grounded solution for fatigue prediction, addressing key limitations of prior multimodal fusion approaches such as static weighting, lack of adaptivity to modality quality, and absence of hierarchical processing. HADF integrates two adaptive weighting mechanisms: α (intra-modality), which determines the importance of features within each modality, and γ (inter-modality), which dynamically regulates the contribution of each modality based on input reliability at each timestep. This mechanism provides a scientific explanation for the model's stability—when a modality becomes degraded, noisy, or missing, the adaptive weights automatically downscale its influence while shifting emphasis to more reliable modalities. Experimental results demonstrate that HADF achieves high accuracy and maintains strong robustness under missing-modality and low-quality conditions, outperforming conventional static fusion methods that cannot adjust their weights dynamically. Consequently, the CNN–LSTM architecture augmented with HADF offers not only improved predictive accuracy but also substantial practical relevance for real-world fatigue-monitoring systems in transportation safety, industrial operator supervision, and healthcare applications, where detection failures may have critical consequences.

REFERENCES

[1] H. Jia, Z. Xiao, and P. Ji, "Fatigue Driving Detection Based on Deep Learning and Multi-Index Fusion," IEEE Access, vol. 9, pp. 147054–147062, 2021, doi: 10.1109/ACCESS.2021.3123388.

[2] S. Bhattacharya and P. Singh, "CNN–RNN Hybrid Deep Learning Model for Monthly Rainfall Prediction," Smart Innov. Syst. Technol., vol. 413 SIST, no. January, pp. 549–559, 2025, doi: 10.1007/978-981-97-7717-4_39.

[3] R. Pan, J. Gao, L. Meng, F. Heng, and H. Yang, "A new approach to multiaxial fatigue life prediction: A multi-dimensional multi-scale composite neural network with multi-depth," Eng. Fract. Mech., vol. 310, no. May, p. 110501, 2024, doi: 10.1016/j.engfracmech.2024.110501.

[4] R. S. El-Sayed, "A Hybrid CNN-LSTM Deep Learning Model for Classification of the Parkinson Disease," IAENG Int. J. Appl. Math., vol. 53, no. 4, 2023.

[5] Y. Gordienko, S. Stirenko, Y. Kochura, O. Alienin, M. Novotarskiy, and N. Gordienko, "Deep Learning for Fatigue Estimation based on Multimodal Human-Machine Interactions," 2017, [Online]. Available: http://arxiv.org/abs/1801.06048

[6] Y. Hu, Y. Chen, X. Li, and J. Feng, "Dynamic feature fusion for semantic edge detection," IJCAI Int. Jt. Conf. Artif. Intell., vol. 2019-Augus, pp. 782–788, 2019, doi: 10.24963/ijcai.2019/110.

[7] L. Mou et al., "Driver stress detection via multimodal fusion using attention-based CNN-LSTM," Expert Syst. Appl., vol. 173, no. November 2020, p. 114693, 2021, doi: 10.1016/j.eswa.2021.114693.

[8] L. Kong, K. Xie, K. Niu, J. He, and W. Zhang, "Remote Photoplethysmography and Motion Tracking Convolutional Neural Network with Bidirectional Long Short-Term Memory: Non-Invasive Fatigue Detection Method Based on Multi-Modal Fusion," Sensors, vol. 24, no. 2, 2024, doi: 10.3390/s24020455.

[9] S. Gheisari et al., "A combined convolutional and recurrent neural network for enhanced glaucoma detection," Sci. Rep., vol. 11, no. 1, pp. 1–11, 2021, doi: 10.1038/s41598-021-81554-4.

[10] M. King, S. I. Woo, and C. Y. Yune, "Utilizing a CNN-RNN machine learning approach for forecasting time-series outlet fluid temperature monitoring by long-term operation of BHEs system," Geothermics, vol. 122, no. March, p. 103082, 2024, doi: 10.1016/j.geothermics.2024.103082.

[11] F. Zhou, Y. Chen, and J. Liu, "Application of a New Hybrid Deep Learning Model That Considers Temporal and Feature Dependencies in Rainfall–Runoff Simulation," Remote Sens., vol. 15, no. 5, 2023, doi: 10.3390/rs15051395.

[12] Y. Hou et al., "Attention-based cross-modal fusion for audio-visual voice activity detection in musical video streams," Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH, vol. 6, pp. 4590–4594, 2021, doi: 10.21437/Interspeech.2021-37.

[13] M. A. Al Imran, F. Nasirzadeh, and C. Karmakar, "Designing a practical fatigue detection system: A review on recent developments and challenges," J. Safety Res., vol. 90, no. May, pp. 100–114, 2024, doi: 10.1016/j.jsr.2024.05.015.

[14] T. Wang, B. Huang, and H. Li, "Optimized third-generation prospect theory-based three-way decision approach for conflict analysis in multi-scale Z-number information systems," Inf. Sci. (Ny)., vol. 663, no. January, p. 120309, 2024, doi: 10.1016/j.ins.2024.120309.

[15] N. Cohen and I. Klein, "Adaptive Kalman-Informed ransformer," Eng. Appl. Artif. Intell., vol. 146, no. December 2024, p. 110221, 2025, doi: 10.1016/j.engappai.2025.110221.

[16] B. Ladjal et al., "Hybrid deep learning CNN-LSTM model for forecasting direct normal irradiance: a study on solar potential in Ghardaia, Algeria," Sci. Rep., vol. 15, no. 1, pp. 1–16, 2025, doi: 10.1038/s41598-025-94239-z.

[17] S. Benbakreti, S. Benbakreti, K. Benyahia, and M. Benouis, "Using Resnet18 in a Deep-Learning Framework and Assessing the Effects of Adaptive Learning Rates in the Identification of Malignant Breast Masses in Mammograms," Jordanian J. Comput. Inf. Technol., vol. 10, no. 1, pp. 93–107, 2024, doi: 10.5455/jjcit.71-1699818406.

[18] Z. Wu, R. Zhuo, X. Liu, B. Wu, and J. Wang, "Enhancing surgical decision-making in NEC with ResNet18: a deep learning approach to predict the need for surgery through x-ray image analysis," Front. Pediatr., vol. 12, no. June, pp. 1–10, 2024, doi: 10.3389/fped.2024.1405780.