Available online at http://scik.org
Commun. Math. Biol. Neurosci. 2026, 2026:4
https://doi.org/10.28919/cmbn/9648
ISSN: 2052-2541

HYBRID MOBILENETV3-LSTM MODEL FOR DETECTING PHASIC AND TONIC RECEPTOR RESPONSES IN FACIAL IMAGES

MUHAMMAD RESTU AGAM, ANINDYA APRILIYANTI PRAVITASARI*, TRIYANI HENDRAWATI, I GEDE NYOMAN MINDRA JAYA, YUSEP SUPARMAN

Department of Statistics, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Bandung 45363, Indonesia

Copyright © 2026 the author(s). This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract. Pain was a complex and subjective experience that involved sensory, emotional, and cognitive aspects simultaneously. The responses of phasic and tonic receptors to painful stimuli exhibited different patterns and could be observed through facial expressions as a form of nonverbal communication. This study aimed to implement a MobileNetV3-LSTM model to classify phasic, tonic, and normal receptor responses using human facial expression images. The objective was to obtain the most optimal model for classifying facial expressions exposed to pain stimuli targeting phasic or tonic receptors. The methods involved the development and evaluation of three models: MobileNetV3Large, MobileNetV3Small, and their respective hybrid versions combined with LSTM, to examine the effect of incorporating temporal information on classification performance. According to results 10, the hybrid MobileNetV3Large-LSTM model performed the best, with an F1-score of 94%, accuracy of 93%, precision of 96%, and recall of 93% on the 12 test data. Meanwhile, the MobileNetV3Small-LSTM model reached 74% accuracy, 80% precision, 74% recall, and a 74% F1-score. The standalone MobileNetV3Large model only obtained 68% accuracy and an F1-score of 0.59, while MobileNetV3Small without LSTM achieved 75% accuracy and an F1-score of 0.56. These results suggest that the inclusion of LSTM layers greatly enhanced the accuracy in the model. This research added to the development of facial expression classification methodologies to recognize pain and complemented the body of knowledge on hybrid model utilization in deep learning.

*Corresponding author
E-mail address: anindya.apriliyanti@unpad.ac.id
Received October 18, 2025

Keywords: hybrid mobilenetv3-lstm; facial expression classification; phasic receptor; tonic receptor; deep learning.

2020 AMS Subject Classification: 68T07, 68T45.

1. INTRODUCTION

Pain is a multi-dimensional subjective human phenomenon. Apart from containing sensory information, pain has emotional as well as cognitive content [1, 2]. These contents are in constant interaction with one another and determine the manner in which an individual perceives pain [3]. Because of this multi-dimensionality, pain is very difficult to quantify objectively. This remains the primary challenge in clinical practice as well as in research in medical science [4, 5, 6].

Phasic and tonic receptors have characteristic patterns of reaction to stimuli. Phasic receptors are rapid in their reaction but their activity promptly abolishes when the stimulus is withdrawn [7, 8]. Tonic receptors are slow in their reaction but their activity continues with time [7, 9]. Awareness of this characteristic is significant in establishing the type of pain that the patient is in [10, 11]. Such knowledge is especially useful in diagnosing diseases like neuropathy where receptor responses are strong clinical indicators [12, 13].

Facial expression is one of the most important forms of nonverbal communication. Small changes in facial muscles often reflect emotional or physical states [14, 15]. In the case of pain, facial expression provides highly relevant diagnostic information. Detecting pain through facial expression allows assessment even when patients struggle to express their pain verbally [19, 20]. This is important for children, elderly patients, and people with neurological or communication problems.

Facial expressions are often detected manually using techniques like the Facial Action Coding System (FACS). These methods have high accuracy but require long processing time. They also demand professional training, which limits their practical use in a clinical setting [31]. In a hospital where decisions must be made quickly, manual coding is not efficient [21]. Therefore, automated systems for facial expression recognition are urgently needed.

The rise of deep learning technology offers a solution to these limitations [16, 20, 21, 22]. Direct feature learning from unprocessed photos or videos is possible with deep learning models

[21, 22]. They remove the need for handcrafted features and improve generalization ability. In healthcare, these systems promise better accuracy and consistency compared to manual methods [25, 26, 27]. Real-time classification systems based on deep learning can support faster and more reliable decision-making [23, 24].

Several deep learning approaches have been developed for pain detection. When it comes to extracting spatial characteristics from photos, Convolutional Neural Networks (CNNs) excel [10]. Modeling temporal dynamics is a common use of recurrent neural networks (RNNs), particularly Long Short-Term Memory (LSTM) [11]. Hybrid architectures combining CNNs with LSTM provide a more complete understanding of visual information over time [35]. Such designs are suitable for video-based analysis of facial expression.

MobileNet is one of the most popular CNN families for lightweight image classification. MobileNetV2 introduces depthwise separable convolutions and inverted residuals, which make the model faster and more efficient [17]. MobileNetV2 combined with LSTM has been tested in several domains [18]. These models achieve strong accuracy while keeping computational demands low. However, the ability to balance lightweight design and deep feature representation is still under discussion.

MobileNetV3 is introduced as an improved version of MobileNetV2. It integrates squeeze-and-excitation modules and optimized nonlinear functions [28]. The architecture is more lightweight than V2 while maintaining strong representational capacity [28]. Because of this, it is better suited for use in clinical settings with limited resources and in real-time [28, 36]. However, limited studies explore its potential when combined with temporal models such as LSTM.

The main problem of conventional CNN models is their inability to capture both spatial and temporal information simultaneously [37, 39]. Pain-related facial expressions evolve over time and require temporal context to be classified correctly [29]. Models that ignore temporal dynamics often lose accuracy in distinguishing phasic from tonic responses [38]. This limitation creates significant barriers for clinical application. Therefore, hybrid models that merge CNN-based spatial analysis with LSTM-based temporal analysis are needed.

Recent literature shows increasing interest in CNN-LSTM hybrids for emotion recognition, action recognition, and medical diagnosis. In pain detection research, several models adopt

MobileNetV2 or ResNet as the spatial backbone [30, 32, 33]. While these models achieve promising results, they are not always optimal for real-time medical settings. The computational load remains high and inference time can be slow. This makes the search for more efficient architectures highly relevant.

Our study addresses this research gap by proposing a hybrid model using MobileNetV3 and LSTM. MobileNetV3 extracts compact but discriminative spatial features from facial images. LSTM captures the temporal evolution of these features across video frames. Together, the hybrid architecture can classify phasic pain, tonic pain, and non-pain states. This combination is designed to provide both efficiency and accuracy for clinical application.

This work's innovation is in employing MobileNetV3 as a backbone for hybrid pain classification. Unlike MobileNetV2, MobileNetV3 is more lightweight and has improved attention mechanisms [34]. This allows the model to maintain performance even with limited resources. By integrating this with LSTM, the system is able to encode both static and dynamic attributes in facial expression. We expect this strategy to improve phasic and tonic pain response classification.

This paper contributes three main points. Firstly, we offer a pain detection framework that is a combination between time analysis and lightweight spatial extraction. Secondly, we apply MobileNetV3 in the clinical setting, where computational efficiency is the main priority. Thirdly, we contribute to the building of more objective pain detection systems in poorly communicating patients. All these are towards the broader objective to advance automatic health diagnostics.

Pain remains a qualitative and complex issue in medical measurement. Recognition of facial expression is one promising way to objectively measure pain. Optimal and accurate methods to this task are provided by deep learning. However, most existing models are incapable of extracting time-aware as well as spatial features optimally. A MobileNetV3-LSTM cascade model to classify pain in real time is introduced in this work to fill that gap.

2. Materials and Methods

2.1. Data. This research utilized a supplementary dataset released by Fernandes-Magalhães et al. (2022) [40], containing facial expression images of the subjects who were provided stimulus and non-stimulus to their pain receptors. The dataset included a cumulative number of

2,424 facial images, which were marked as 379 phasic receptor responses, 725 tonic receptor responses, and 1,320 neutral expressions. The dataset was split into training and validation datasets in a proportion of 90:10.

2.2. Data Preprocessing. Prior to commencing the modeling, extensive preprocessing was done to the image datasets in order to improve the accuracy of models as well as gain optimal performance. These transformations were executed to make the pixel information suitable to match the model's architecture as well as improve all-around computational efficiency.

2.2.1. *Image Resizing.* The original resolution of each image in the dataset is 1920×1080 pixels. To enhance computational efficiency and minimize the risk of overfitting, each image is resized to 224×224 pixels. This resizing process uses bilinear interpolation, a widely adopted resampling method that estimates new pixel intensities by computing a weighted average of the four surrounding pixels.

The estimated pixel value at a new location is obtained by linearly interpolating along both the horizontal and vertical axes. The interpolation uses weights denoted as η, $1-\eta$, ξ, and $1-\xi$, which are determined by the fractional distance of the new pixel position relative to its nearest four neighbors. The bilinear interpolation is mathematically represented in Equation (1):

$$\begin{aligned}\hat{\psi}(u,v) = {} & (1-\xi)(1-\eta)\cdot\phi(m,n) + \xi(1-\eta)\cdot\phi(m+1,n) \\ & + \eta(1-\xi)\cdot\phi(m,n+1) + \xi\eta\cdot\phi(m+1,n+1)\end{aligned} \tag{1}$$

In this equation, $\hat{\psi}(u,v)$ represents the interpolated pixel intensity at the new spatial location (u,v) in the resized image. The function $\phi(m,n)$ refers to the original pixel value at location (m,n) in the input image. Likewise, $\phi(m+1,n)$, $\phi(m,n+1)$, and $\phi(m+1,n+1)$ denote the intensities of the three neighboring pixels required to estimate the target value. The parameters ξ and η indicate the horizontal and vertical fractional distances from the reference pixel (m,n), respectively.

2.2.2. *Image Pixel Normalization.* Each image consisted of color intensity values ranging from 0 to 255. To facilitate the model training process and enhance computational stability, the pixel values were normalized by rescaling the range to between -1 and 1 [41]. Mathematically,

this normalization process was expressed in Equation (2):

$$x_{\text{norm}} = \frac{x}{127.5} - 1 \tag{2}$$

Where x_{norm} represents the pixel value after normalization, and x denotes the original pixel value.

2.2.3. *Image Augmentation.* Image augmentation was a technique used to increase data diversity so that the model could learn a broader range of variations during the training process. In this study, the image augmentation methods applied included rotation, shifting, zooming, horizontal flipping, and shearing.

Rotation. The rotation operation transforms pixel locations based on a rotation matrix applied to each coordinate point. A random angular adjustment of 20° is introduced using Equations (3) and (4):

$$\lambda_u^* = (\lambda_u - \zeta_\alpha) \cdot \cos(\vartheta) - (\lambda_v - \zeta_\beta) \cdot \sin(\vartheta) + \zeta_\alpha \tag{3}$$

$$\lambda_v^* = (\lambda_u - \zeta_\alpha) \cdot \sin(\vartheta) + (\lambda_v - \zeta_\beta) \cdot \cos(\vartheta) + \zeta_\beta \tag{4}$$

Where ϑ represents the rotation angle in radians, while $(\zeta_\alpha, \zeta_\beta)$ refers to the coordinates of the image center. The values λ_u^* and λ_v^* are the transformed horizontal and vertical positions of a given pixel (λ_u, λ_v) after rotation.

Shift. Pixel shifting is performed to translate an image spatially, enhancing the model's sensitivity to positional variations. Horizontal and vertical displacements of 15% are applied as in Equations (5) and (6):

$$\chi_1^* = \chi_1 + \rho_h \cdot \delta \tag{5}$$

$$\chi_2^* = \chi_2 + \rho_v \cdot \varepsilon \tag{6}$$

Where (χ_1, χ_2) are the original coordinates, (χ_1^*, χ_2^*) are the new coordinates after shifting, ρ_h and ρ_v represent the horizontal and vertical shift proportions, and δ, ε denote the image width and height, respectively.

Zoom. Zooming involves scaling pixel positions relative to the image center, effectively enlarging or shrinking specific regions. A zoom ratio of 20% is applied using Equations (7) and (8):

$$\omega_u^* = (\omega_u - \gamma_u) \cdot (1 + \pi) + \gamma_u \tag{7}$$

$$\omega_v^* = (\omega_v - \gamma_v) \cdot (1 + \pi) + \gamma_v \tag{8}$$

Where ω_u and ω_v are the original coordinates, γ_u and γ_v denote the central reference point, and π is the zoom factor. The new coordinates after zooming are represented as (ω_u^*, ω_v^*).

Horizontal Flip. This augmentation technique mirrors the image along the vertical axis, flipping each pixel horizontally without affecting its intensity. The transformation is defined in Equations (9) and (10):

$$\varphi_x^* = (\Delta - 1) - \varphi_x \tag{9}$$

$$\varphi_y^* = \varphi_y \tag{10}$$

Where (φ_x, φ_y) are the original pixel coordinates, φ_x^* is the mirrored horizontal position, and Δ is the image width.

Shear. Shearing introduces angular distortion by offsetting pixels in horizontal and/or vertical directions. A 5% shear factor is applied using Equations (11) and (12):

$$\kappa_u^* = \kappa_u + \sigma_h \cdot \kappa_v \tag{11}$$

$$\kappa_v^* = \kappa_v + \sigma_v \cdot \kappa_u \tag{12}$$

Where (κ_u, κ_v) are the original pixel coordinates, (κ_u^*, κ_v^*) are the transformed coordinates, and σ_h, σ_v represent horizontal and vertical shear coefficients, respectively.

2.3. Data Modeling Using MobileNetV3-LSTM. The MobileNetV3-LSTM architecture consisted of several main layers as described below.

2.3.1. *Input Layer.* The input layer accepted image data as a matrix, according to the specified dimensions for subsequent processing.

2.3.2. *Convolutional Layer.* The convolutional layer extracted spatial features from the image using filters (kernels) that moved across the input. The architecture had one or more convolutional layers succeeded by fully linked layers, yielding a linear transformation of the input data contingent upon the spatial attributes of the information.

In the MobileNetV3-LSTM architecture, two main types of activation functions were used: the Rectified Linear Unit (ReLU) and the Hard-Swish (H-Swish). The ReLU function served to eliminate negative values in the image by replacing them with zero [42]. The ReLU function was defined in Equation (13):

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{13}$$

Where x represented a numerical value indicating the intensity or features extracted from the image, and $f(x)$ denoted the output of the ReLU activation function. The graphical representation of ReLU is illustrated in Figure 1.

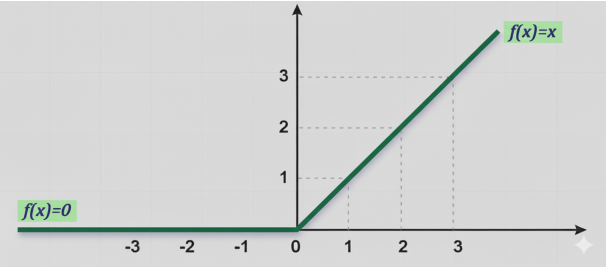


FIGURE 1. ReLU activation function graph.

Although ReLU offered advantages in computational efficiency and its ability to handle the vanishing gradient problem, it had a notable limitation known as the "Dying ReLU Problem," where neurons could become inactive if negative values occurred too frequently during training.

As an alternative to improve computational efficiency in the MobileNetV3 architecture, the Hard-Swish (H-Swish) activation function was implemented in several layers. This function was an approximated version of the more complex Swish activation function but was lighter to compute as it only involved linear and ReLU-like operations [43]. Its mathematical formulation was given in Equation (14):

$$f(x) = x \cdot \frac{\text{ReLU6}(x+3)}{6} \tag{14}$$

Where the ReLU6 function was defined as:

$$\text{ReLU6}(x) = \min\left(\max(0,x),6\right) \tag{15}$$

2.3.3. *Flatten Layer.* The flatten layer functioned to convert the multi-dimensional representation from the previous layer into a one-dimensional tensor [44]. The primary purpose of this layer was to flatten the spatial data so it could be processed by the subsequent fully connected layer, which required input in the form of a vector. An example of flattening is illustrated in Figure 2.

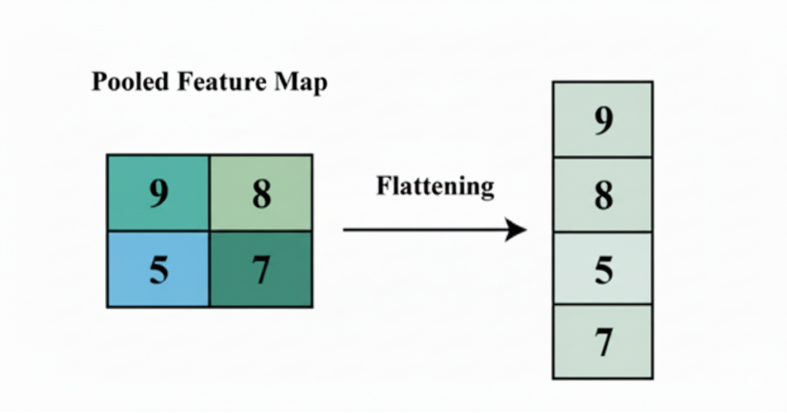


FIGURE 2. Flatten Layer.

2.3.4. *Pooling Layer.* Pooling was a non-linear down-sampling process used in Convolutional Neural Network (CNN) architectures to reduce the spatial dimensions of the feature maps produced by the convolutional layer. This study employed both max pooling and average pooling to obtain a more compact representation by emphasizing key features, reducing computational complexity, and providing invariance to positional variations of features [45]. An illustration of pooling mechanisms is shown in Figure 3.

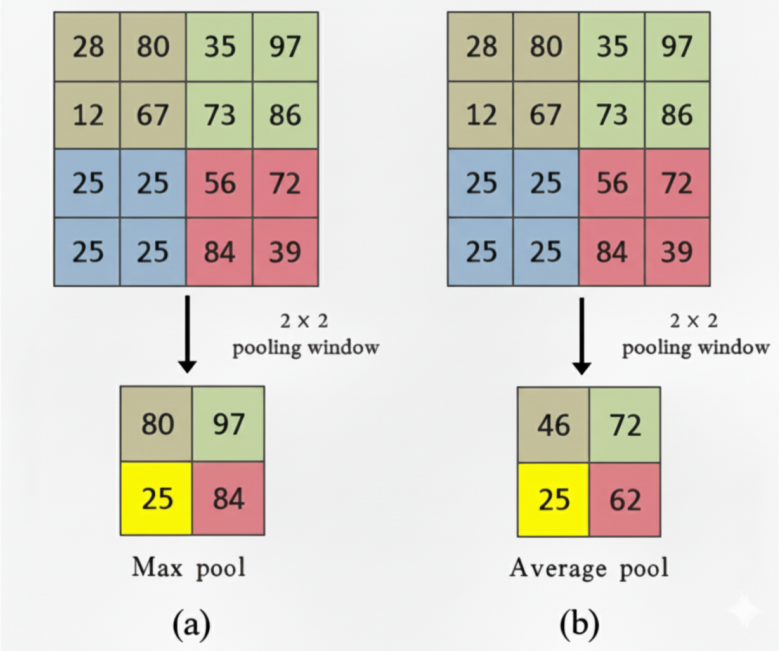


FIGURE 3. Example of Max Pooling and Average Pooling.

2.3.5. *LSTM Layer.* In the CNN-LSTM architecture, the CNN extracted essential features from an image, while the LSTM replaced the role of the fully connected layer by performing classification based on the extracted features. The LSTM structure is presented in Figure 4.

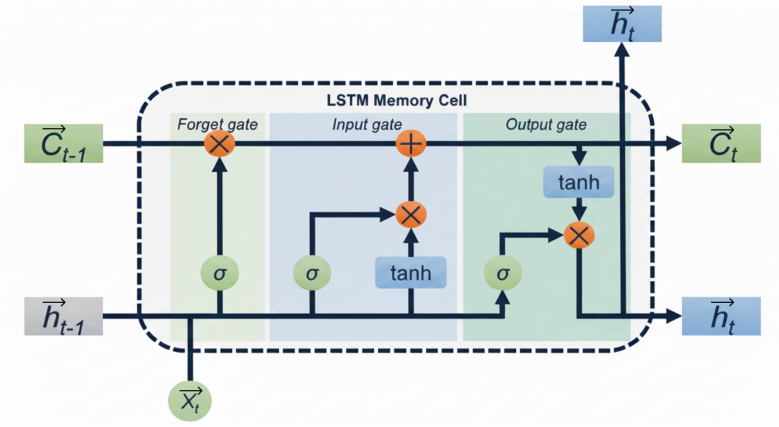


FIGURE 4. LSTM Structure.

Figure 4 shows the LSTM model structure in the MobileNetV3 architecture. This structure included the forget gate, input gate, and output gate. These gates functioned to either allow or restrict access to the LSTM memory. Each memory cell in the LSTM contained three sigmoid

layers and one tanh layer, which worked together to regulate the flow of information within the model [46].

2.3.6. *Dropout Layer.* The dropout layer was a regularization technique used to reduce overfitting in neural networks. This approach functioned by randomly deactivating certain neurons during training, hence inhibiting intricate co-adaptations to the training input [47].

2.3.7. *Fully Connected Layer.* The fully connected layer interlinks each neuron in one layer with every neuron in the subsequent layer. This layer resembled a multilayer perceptron, where the flattened matrix was passed through to perform classification on the image. Additionally, this layer merged all nodes into a single dimension and was commonly referred to as a dense layer [48].

2.3.8. *Output Layer.* This layer acted as the conclusive component within the network architecture, tasked with producing the final forecast of the model. The activation mechanism employed at this stage was the softmax function, which is particularly suitable for scenarios involving multiclass classification. The mathematical formulation of the softmax transformation is presented in Equation (16):

$$\varsigma(\eta_\tau) = \frac{e^{\eta_\tau}}{\sum_{\tau=1}^{\kappa} e^{\eta_\tau}} \tag{16}$$

Where η_τ denotes the unnormalized logit or pre-activation value associated with the τ-th class, and κ corresponds to the total number of distinct categories in the classification task. The function $\varsigma(\cdot)$ yields a normalized probability distribution over all possible classes, ensuring that the output satisfies $0 \leq \varsigma(\eta_\tau) \leq 1$ and $\sum_{\tau=1}^{\kappa} \varsigma(\eta_\tau) = 1$.

2.4. Transfer Learning. The transfer learning process was divided into two stages, namely freezing the hidden layers and unfreezing them. The purpose of this process was to train the output layer first using classification knowledge learned from previous datasets, such as ImageNet, so that it could classify new data—in this case, facial expression images.

2.5. Optimization Method. This study employed the Adaptive Moment Estimation (Adam) optimization approach for automatic parameter tweaking. Adam was a versatile algorithm that calculated distinct learning rates for each parameter, supplanting the conventional stochastic

gradient descent method. The optimizer updated the model parameters based on Equation (17):

$$\theta_{t+1} = \theta_t - \frac{lr}{\sqrt{\hat{v}_t} + \varepsilon}\hat{m}_t \tag{17}$$

Where θ_t denoted the parameter vector at iteration t, lr represented the learning rate, ε was a small constant added for numerical stability (usually set to 1×10^{-8}), $\hat{m}_t$ was the bias-corrected first moment (mean of gradients), and $\hat{v}_t$ was the bias-corrected second moment (uncentered variance of gradients).

The Adam optimizer maintained exponentially decaying averages of past gradients (first moment m_t) and squared gradients (second moment v_t), calculated as follows:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \tag{18}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \tag{19}$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \tag{20}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t} \tag{21}$$

Where β_1 and β_2 were decay rates and g_t was the gradient at the t-th iteration.

2.6. Loss Function. The loss function calculated the disparity between the algorithm's actual output and the anticipated output. One of the loss functions used for multi-class classification with imbalanced data was focal loss [49]. Focal loss naturally addressed class imbalance by down-weighting well-classified examples from the majority class and focusing more on hard-to-classify minority class samples. This made it effective for improving model performance under class imbalance conditions. The mathematical formulation of focal loss was given in Equation (22):

$$\text{Focal Loss} = -\alpha(1-p_t)^{\gamma}\log(p_t) \tag{22}$$

Where p_t represented the predicted probability for the true class, α was a weighting factor (ranging between 0 and 1), and γ was the focusing parameter optimized through cross-validation.

Besides focal loss, class weights were also applied in this study to further address the class imbalance.

2.7. Model Evaluation. In this study, the model was evaluated using several metrics: precision, recall, and F1-score.

Precision. Precision measured the model's ability to correctly identify positive predictions. It was calculated using Equation (23):

$$\text{Precision} = \frac{TP}{TP+FP} \tag{23}$$

Recall. Recall measured the model's ability to detect all actual positive instances. In this study, the positives referred to the phasic receptor response, tonic receptor response, and neutral state. Recall was computed using Equation (24):

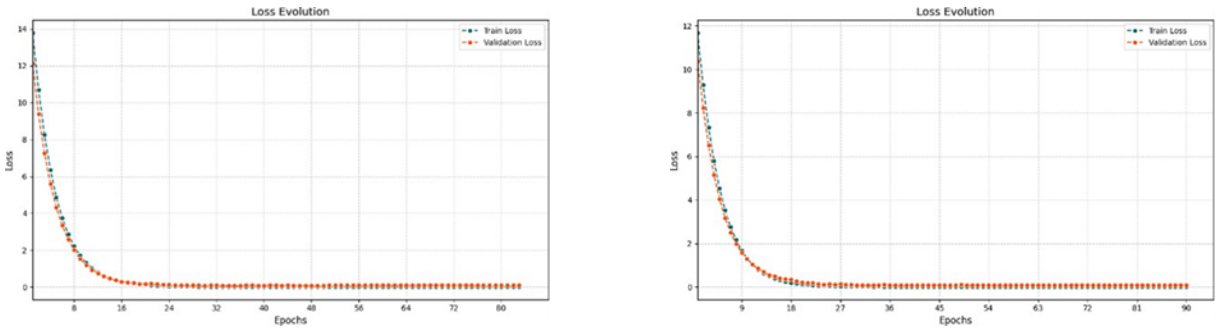
$$\text{Recall} = \frac{TP}{TP+FN} \tag{24}$$

F1-score. The F1-score provided a harmonic mean of precision and recall to evaluate the model's overall performance, especially in cases of class imbalance. It was calculated using Equation (25):

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{25}$$

3. Main Results

3.1. Modeling Data Using MobileNetV3-LSTM. In this study, two hybrid architectures were developed—MobileNetV3Large-LSTM and MobileNetV3Small-LSTM—to detect phasic and tonic receptor responses in facial expression images. These hybrid models sought to combine the feature extraction efficacy of MobileNetV3 with the temporal sequence learning proficiency of LSTM. The training process was visualized using key metrics, particularly the training loss and validation loss, as shown in Figure 5.



(A) MobileNetV3Large-LSTM (B) MobileNetV3Small-LSTM

FIGURE 5. Training and validation loss graphs using (a) MobileNetV3Large-LSTM and (b) MobileNetV3Small-LSTM.

As illustrated in Figure 5, both models experienced a smooth training process. The training and validation loss graphs showed a decreasing trend, which indicated that the models effectively learned from the input data over time. No sign of overfitting was detected since the validation loss mirrored the decline of the training loss throughout the training phase. Among the two, MobileNetV3Large-LSTM reached optimal performance earlier, specifically at the 83rd iteration, while MobileNetV3Small-LSTM achieved it by the 90th iteration, demonstrating a slightly slower convergence.

The performance of each model was further evaluated by predicting the validation dataset and comparing the predicted labels with the actual labels. The confusion matrices summarizing this classification performance are presented in Tables 1 and 2.

TABLE 1. Confusion Matrix – MobileNetV3Large-LSTM (Validation Data)

Actual Class	**Phasic**	**Tonic**	**Neutral**
Phasic	3	18	17
Tonic	0	82	0
Neutral	0	0	120

TABLE 2. Confusion Matrix – MobileNetV3Small-LSTM (Validation Data)

Actual Class	Phasic	Tonic	Neutral
Phasic	8	8	22
Tonic	1	78	3
Neutral	19	2	99

Based on Table 1, MobileNetV3Large-LSTM performed exceptionally well in classifying tonic and neutral expressions. The model correctly identified all 82 tonic samples and all 120 neutral samples without error. However, it struggled with the phasic class, correctly predicting only 3 samples while misclassifying the rest.

Meanwhile, Table 2 indicated that MobileNetV3Small-LSTM also faced challenges in identifying phasic expressions. Out of 38 phasic samples, only 8 were correctly predicted, while 8 were misclassified as tonic and 22 as neutral. This suggested that the model had difficulty distinguishing phasic expressions from neutral ones—likely due to the similarity in visual features between these classes.

3.2. Model Evaluation. To further evaluate the models, precision, recall, and F1-score metrics were calculated based on the confusion matrices. The results are summarized in Tables 3 and 4.

TABLE 3. Performance Metrics – MobileNetV3Large-LSTM (Validation Data)

Class	Precision	Recall	F1-score
Phasic	0.88	1.00	0.93
Tonic	1.00	0.08	0.15
Neutral	0.82	1.00	0.90
Overall Accuracy		0.85	

TABLE 4. Performance Metrics – MobileNetV3Small-LSTM (Validation Data)

Class	**Precision**	**Recall**	**F1-score**
Phasic	0.80	0.82	0.81
Tonic	0.29	0.21	0.24
Neutral	0.89	0.95	0.92
Overall Accuracy		0.77	

The results showed that MobileNetV3Large-LSTM achieved the best performance with an overall F1-score of 0.85 and high precision and recall for neutral and phasic classes. It achieved perfect precision on the tonic class and perfect recall for both phasic and neutral. On the other hand, MobileNetV3Small-LSTM achieved a lower F1-score of 0.77, with its best results appearing in the neutral class.

To deal with class imbalance, focal loss was employed using the parameters $\alpha = 0.25$ and $\gamma = 2$. This effectively improved the balance of evaluation scores across classes by emphasizing the learning from hard-to-classify samples.

Additionally, a performance comparison was conducted between the hybrid models (MobileNetV3-LSTM) and the original MobileNetV3 architecture without LSTM layers. The comparative results are shown in Table 5.

TABLE 5. Comparison of Facial Expression Recognition Methods

Method	**Accuracy**	**F1-score**
MobileNetV3Large-LSTM	0.85	0.66
MobileNetV3Large	0.68	0.59
MobileNetV3Small-LSTM	0.77	0.66
MobileNetV3Small	0.75	0.56

From Table 5, it was evident that MobileNetV3Large-LSTM outperformed other models in both accuracy and F1-score. The inclusion of the LSTM layer enhanced the model's capability to capture temporal dynamics in facial expressions, resulting in better classification performance.

3.3. Model Prediction Results. The trained models were also used to predict classes in the test dataset. Each video was processed frame by frame, and the model provided classification results for each frame. The confusion matrices for the test data predictions are presented in Tables 6 and 7.

TABLE 6. Confusion Matrix – MobileNetV3Large-LSTM (Test Data)

Actual Class	Phasic	Tonic	Neutral
Phasic	20	2	2
Tonic	1	23	0
Neutral	0	0	24

TABLE 7. Confusion Matrix – MobileNetV3Small-LSTM (Test Data)

Actual Class	Phasic	Tonic	Neutral
Phasic	22	2	0
Tonic	12	12	0
Neutral	5	0	19

From Table 6, MobileNetV3Large-LSTM demonstrated high accuracy in classifying neutral and tonic expressions, showing model consistency. However, a small number of phasic frames were misclassified. In contrast, Table 7 showed that MobileNetV3Small-LSTM made more classification errors, especially in the tonic class, where many frames were incorrectly predicted as phasic.

To evaluate the models further, the precision, recall, and F1-score were calculated for the test data, as shown in Tables 8 and 9.

TABLE 8. Performance Metrics – MobileNetV3Large-LSTM (Test Data)

Class	**Precision**	**Recall**	**F1-score**
Phasic	0.952	0.833	0.889
Tonic	0.920	0.958	0.939
Neutral	1.000	1.000	1.000
Overall Accuracy		0.931	

TABLE 9. Performance Metrics – MobileNetV3Small-LSTM (Test Data)

Class	**Precision**	**Recall**	**F1-score**
Phasic	0.564	0.917	0.698
Tonic	0.857	0.500	0.632
Neutral	1.000	0.792	0.884
Overall Accuracy		0.736	

Based on the results, MobileNetV3Large-LSTM achieved superior performance, with a test accuracy of 93.1% and strong F1-scores across all classes. In comparison, MobileNetV3Small-LSTM achieved a lower accuracy of 73.6%, with considerable errors in predicting tonic and phasic expressions. The low precision for the phasic class (0.564) and low recall for the tonic class (0.500) highlighted the limitations of the smaller architecture in learning complex spatiotemporal patterns.

In conclusion, the MobileNetV3Large-LSTM hybrid structure was found as the most optimal classification model for facial expression categories under phasic and tonic responses, as balanced performance in all applied metrics was provided.

4. DISCUSSION

MobileNetV3-LSTM hybrid model learning process exhibits very encouraging results in facial expression classification. Throughout the learning phase, loss curves for the training as well as validation subsets exhibit steadily descending trends indicative of good learning as well as

good generalization capacity for the model. There is no indication of overfitting in the model, indicating that it can learn well to the training dataset while maintaining good performance in new data.

Comparing the two versions tried, MobileNetV3Large-LSTM will always have superior performance to MobileNetV3Small-LSTM. MobileNetV3Large-LSTM is more accurate in classification and will converge more rapidly, which is consistent with previous findings that larger architectures capture more complex patterns in sequential facial data [50]. This is indicative that large architectures with increased representational capacity are in a position to capture more complex patterns in facial expression data when working with time features.

The confusion matrix output provides more detailed insight into the classification accuracy of the models. MobileNetV3Large-LSTM is optimal in classifying tonic as well as neutral faces. The model is completely precise in recognizing all instances of these classes without any misclassifications, showing consistency in recognizing steady as well as subtle emotional cues. However, this model is still weak in the phasic class, which is changing as well as possibly less stable in facial patterns. MobileNetV3Small-LSTM, on the other hand, has frequent misclassifications, particularly for phasic faces. Such misclassifications can arise from the high similarity between the visual cues in phasic as well as neutral faces, which creates confusion as well as classification ambiguity.

Employment of accuracy, recall, and F1-score in performance assessment again highlights differences between the two models, in line with standard evaluation practices in facial expression recognition [51, 52]. MobileNetV3Large-LSTM has the highest F1-score in all classes in general, justifying balanced performance in all classes. Its accuracy is very high in the tonic class to indicate low false-positive instances, while maximum recall is observed in the phasic as well as neutral classes, to show that most instances are selected by the model. MobileNetV3Small-LSTM, in comparison, has relatively low performance, however, significant power is noted in the detection of neutral expressions. This outcome again highlights the significance of model complexity in handling fine differences in facial expression classes.

A comparison to the baseline reference MobileNetV3 architecture, not employing the LSTM layer, shows their substantial performance gain due to the incorporation of temporal analysis.

Employing the LSTM module enables the network to identify temporal dependencies as well as intrinsic patterns in sequences that are characteristic for facial expressions, most notably those that are characterized by dynamic transitions like phasic responses. Overall accuracy as well as F1-score are the highest for the MobileNetV3Large-LSTM hybrid, thereby proving that employing the use of temporal modeling is significant in order to improve CNN performance in this application.

The performance in predicting the test dataset also offers validation to the superior performance in the MobileNetV3Large-LSTM architecture. Its accuracy in classification is always good, especially for the neutral class as well as the tonic class. Despite still misclassifying from time to time when identifying phasic expression, accuracy remains still significantly higher than in the small variant. MobileNetV3Small-LSTM, in comparison, has more frequent misclassifications, especially in the tonic class, which points to insufficiency in identifying subtle class-specific features.

Quantitative evaluation with the standard metrics of classification on the testset provides further corroboration. MobileNetV3Large-LSTM's very good accuracy is 93.1%. All the relevant metrics—precision, recall, F1-score—are equally high, as a consequence indicating that not only are predictions extremely precise, classification is also very reliable and consistent overall. MobileNetV3Small-LSTM has poorer overall accuracy as well as marked imbalance in class-wise performance. Recall on the model's part is extremely poor in the case of the tonic class, and this is aggravated by low precision while recognizing phasic expressions. These are points of weakness that are characteristic of low capacity to recognize fine differences between rather similar facial expressions.

The best-performing model for facial expression classification based on tonic, phasic, and neutral responses is the MobileNetV3Large-LSTM combination architecture. Its ability to utilize both time and spatial information makes this architecture most appropriate for this task. Its time modeling capacity in the LSTM section is most helpful in enabling the network to better capture the order of facial muscle movement, which is significant in discriminating dynamic movements such as phasic responses.

The benefit in using a hybrid deep learning architecture is all the more significant when comparing models that incorporate and models that don't incorporate temporal modeling. While CNNs like MobileNetV3 are master at pulling spatial information from images, they can't learn to recognize how facial features change over time. By implementing an LSTM layer, the model is given this memory mechanism that can accept sequences, which makes the model much more efficient in real-world use cases in which expressions change over time rather than in the form of disjointed snapshots.

Computationally, the MobileNetV3Large-LSTM is an effective solution for facial expression recognition systems in real-time. Its effectiveness in doing well on both the training and the test sets thus has great potential for use in emotion-aware human-computer interaction, tracking of mental health, as well as affective computing systems. There are numerous directions for future research that can potentially involve expanding the dataset to enhance its recognition ability for phasic expressions, possibly through the implementation of complex temporal attention mechanisms.

Systematically, this study also re-asserts that time-awareness as well as model complexity are significant factors in accounting for improved facial expression recognition performance. MobileNetV3Large-LSTM is found to be the most robust as well as most accurate solution among all the experimented configurations, offering a balanced yet high-performance organization for recognizing diverse facial expression patterns. The unifying framework that incorporates CNN as well as LSTM represents a sound platform for future study as well as practical application in emotional computing.

5. CONCLUSIONS

This study presents the classification performance of the hybrid MobileNetV3-LSTM model in facial expression classification across phasic as well as tonic receptor responses. The model can effectively extract both spatial as well as time-based features from facial images by integrating a Long Short-Term Memory (LSTM) layer with MobileNetV3. The LSTM component is essential in dealing with the dynamic patterns of expression in time, thereby realizing superior classification performance in comparison to the control MobileNetV3 model that does not use LSTM.

The performance obtained from experiments indicates that the MobileNetV3Large-LSTM is the best-performing among all models, with accuracy, precision, recall, and F1-score values of 93%, 96%, 93%, and 94% respectively. Comparatively, MobileNetV3Small-LSTM has low performance, with accuracy and F1-score values of 74% respectively. This indicates that increased complexity in the model is desirable for classification performance.

These findings contribute to theory for hybrid deep learning models for emotion recognition detection as well as receptor responses. The ability of the model to classify phasic versus tonic responses based on facial images alone offers hope that the model is adaptable to clinically practical use in affective computing, clinical diagnosis, and adaptive human-computer interaction.

In conclusion, the proposed hybrid MobileNetV3-LSTM model presents a robust and promising solution for analyzing facial expressions with temporal dependencies, making it a valuable tool for future advancements in emotion-aware technologies.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests.

REFERENCES

[1] M. Karst, Addressing the Emotional Body in Patients with Chronic Pain, JAMA Netw. Open 7 (2024), e2417340. https://doi.org/10.1001/jamanetworkopen.2024.17340.

[2] L. Goudman, N. Vets, J. Jansen, A. De Smedt, M. Moens, The Association Between Bodily Functions and Cognitive/Emotional Factors in Patients with Chronic Pain Treated with Neuromodulation: A Systematic Review and Meta-Analyses, Neuromodulation: Technol. Neural Interface 26 (2023), 3–24. https://doi.org/10.1016/j.neurom.2021.11.001.

[3] C.N. Li, K.A. Keay, L.A. Henderson, R. Mychasiuk, Re-Examining the Mysterious Role of the Cerebellum in Pain, J. Neurosci. 44 (2024), e1538232024. https://doi.org/10.1523/jneurosci.1538-23.2024.

[4] S.Y. Lee, J.B. Kim, J.W. Lee, A.M. Woo, C.J. Kim, et al., A Quantitative Measure of Pain with Current Perception Threshold, Pain Equivalent Current, and Quantified Pain Degree: A Retrospective Study, J. Clin. Med. 12 (2023), 5476. https://doi.org/10.3390/jcm12175476.

[5] A.K. Pace, M. Bruceta, J. Donovan, S.J. Vaida, J.M. Eckert, An Objective Pain Score for Chronic Pain Clinic Patients, Pain Res. Manag. 2021 (2021), 6695741. https://doi.org/10.1155/2021/6695741.

[6] H.F. Posada–Quintero, Y. Kong, K.H. Chon, Objective Pain Stimulation Intensity and Pain Sensation Assessment Using Machine Learning Classification and Regression Based on Electrodermal Activity, Am. J. Physiol. Integr. Comp. Physiol. 321 (2021), R186–R196. https://doi.org/10.1152/ajpregu.00094.2021.

[7] H. Cheng, D. Chen, X. Li, U. Al-Sheikh, D. Duan, et al., Phasic/tonic Glial GABA Differentially Transduce for Olfactory Adaptation and Neuronal Aging, Neuron 112 (2024), 1473–1486.e6. https://doi.org/10.1016/j.neuron.2024.02.006.

[8] L. Skora, A. Marzecová, G. Jocham, Tonic and Phasic Transcutaneous Auricular Vagus Nerve Stimulation (TaVNS) Both Evoke Rapid and Transient Pupil Dilation, Brain Stimul. 17 (2024), 233–244. https://doi.org/10.1016/j.brs.2024.02.013.

[9] J. Eggert, B.B. Au-Yeung, Functional Heterogeneity and Adaptation of Naive T Cells in Response to Tonic TCR Signals, Curr. Opin. Immunol. 73 (2021), 43–49. https://doi.org/10.1016/j.coi.2021.09.007.

[10] S. Mei, J. Ji, J. Hou, X. Li, Q. Du, Learning Sensor-Specific Spatial-Spectral Features of Hyperspectral Images via Convolutional Neural Networks, IEEE Trans. Geosci. Remote. Sens. 55 (2017), 4520–4533. https://doi.org/10.1109/tgrs.2017.2693346.

[11] A. Macias-Hernandez, D.F. Orozco-Granados, I. Chairez, Adaptive Modeling of Systems with Uncertain Dynamics via Continuous Long-Short Term Memories, Neurocomputing 599 (2024), 127955. https://doi.org/10.1016/j.neucom.2024.127955.

[12] J.M. Navia-Pelaez, J. Borges Paes Lemes, L. Gonzalez, et al., AIBP Regulates TRPV1 Activation in Chemotherapy-Induced Peripheral Neuropathy by Controlling Lipid Raft Dynamics and Proximity to TLR4 in Dorsal Root Ganglion Neurons, Pain 164 (2022), e274–e285. https://doi.org/10.1097/j.pain.0000000000002834.

[13] R. Gálvez, V. Mayoral, J. Cebrecos, F.J. Medel, A. Morte, et al., E-52862—A Selective Sigma-1 Receptor Antagonist, in Peripheral Neuropathic Pain: Two Randomized, Double-blind, Phase 2 Studies in Patients with Chronic Postsurgical Pain and Painful Diabetic Neuropathy, Eur. J. Pain 29 (2024), e4755. https://doi.org/10.1002/ejp.4755.

[14] T.N. Efthimiou, J. Baker, A. Elsenaar, M. Mehu, S. Korb, Smiling and Frowning Induced by Facial Neuromuscular Electrical Stimulation (FNMES) Modulate Felt Emotion and Physiology, Emotion 25 (2025), 79–92. https://doi.org/10.1037/emo0001408.

[15] H. Kim, D. Zhang, L. Kim, C. Im, Classification of Individual's Discrete Emotions Reflected in Facial Microexpressions Using Electroencephalogram and Facial Electromyogram, Expert Syst. Appl. 188 (2022), 116101. https://doi.org/10.1016/j.eswa.2021.116101.

[16] I. Park, J.H. Park, J. Yoon, H. Na, A. Oh, et al., Machine Learning Model of Facial Expression Outperforms Models Using Analgesia Nociception Index and Vital Signs to Predict Postoperative Pain Intensity: A Pilot Study, Korean J. Anesth. 77 (2024), 195–204. https://doi.org/10.4097/kja.23583.

[17] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L. Chen, MobileNetV2: Inverted Residuals and Linear Bottlenecks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, 2018, pp. 4510-4520. https://doi.org/10.1109/cvpr.2018.00474.

[18] P.N. Srinivasu, J.G. SivaSai, M.F. Ijaz, A.K. Bhoi, W. Kim, et al., Classification of Skin Disease Using Deep Learning Neural Networks with MobileNet V2 and LSTM, Sensors 21 (2021), 2852. https://doi.org/10.3390/s21082852.

[19] J. Huo, Y. Yu, W. Lin, A. Hu, C. Wu, Application of AI in Multilevel Pain Assessment Using Facial Images: Systematic Review and Meta-Analysis, J. Med. Internet Res. 26 (2024), e51250. https://doi.org/10.2196/51250.

[20] N. Ben Aoun, A Review of Automatic Pain Assessment from Facial Information Using Machine Learning, Technologies 12 (2024), 92. https://doi.org/10.3390/technologies12060092.

[21] H. Ge, Z. Zhu, Y. Dai, B. Wang, X. Wu, Facial Expression Recognition Based on Deep Learning, Comput. Methods Programs Biomed. 215 (2022), 106621. https://doi.org/10.1016/j.cmpb.2022.106621.

[22] R. Gutierrez, J. Garcia-Ortiz, W. Villegas-Ch, Multimodal AI Techniques for Pain Detection: Integrating Facial Gesture and Paralanguage Analysis, Front. Comput. Sci. 6 (2024), 1424935. https://doi.org/10.3389/fcomp.2024.1424935.

[23] J.O. Pinzon-Arenas, Y. Kong, K.H. Chon, H.F. Posada-Quintero, Design and Evaluation of Deep Learning Models for Continuous Acute Pain Detection Based on Phasic Electrodermal Activity, IEEE J. Biomed. Health Inform. 27 (2023), 4250–4260. https://doi.org/10.1109/jbhi.2023.3291955.

[24] S. Gkikas, M. Tsiknakis, Automatic Assessment of Pain Based on Deep Learning Methods: A Systematic Review, Comput. Methods Programs Biomed. 231 (2023), 107365. https://doi.org/10.1016/j.cmpb.2023.107365.

[25] M. Fahad, N.E. Mobeen, A.S. Imran, S.M. Daudpota, Z. Kastrati, et al., Deep Insights into Gastrointestinal Health: A Comprehensive Analysis of GastroVision Dataset Using Convolutional Neural Networks and Explainable AI, Biomed. Signal Process. Control. 102 (2025), 107260. https://doi.org/10.1016/j.bspc.2024.107260.

[26] B. Schott, D. Pinchuk, V. Santoro-Fernandes, Ž. Klaneček, L. Rivetti, et al., Uncertainty Quantification via Localized Gradients for Deep Learning-Based Medical Image Assessments, Phys. Med. Biol. 69 (2024), 155015. https://doi.org/10.1088/1361-6560/ad611d.

[27] Y. Chen, Y. Li, S. Li, S. Lv, F. Lin, Dualcascadetsf-Mobilenetv2: A Lightweight Violence Behavior Recognition Model, Appl. Sci. 15 (2025), 3862. https://doi.org/10.3390/app15073862.

[28] A. Howard, M. Sandler, B. Chen, W. Wang, L. Chen, et al., Searching for MobileNetV3, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, 2019, pp. 1314-1324. https://doi.org/10.1109/iccv.2019.00140.

[29] C.W. Tan, T. Du, J.C. Teo, D.X.H. Chan, W.M. Kong, et al., Automated Pain Detection Using Facial Expression in Adult Patients with a Customized Spatial Temporal Attention Long Short-Term Memory (sta-Lstm) Network, Sci. Rep. 15 (2025), 13429. https://doi.org/10.1038/s41598-025-97885-5.

[30] R. Islamadina, K. Saddami, M. Oktiana, T.F. Abidin, R. Muharar, et al., Performance of Deep Learning Benchmark Models on Thermal Imagery of Pain Through Facial Expressions, in: 2022 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT), IEEE, 2022, pp. 374-379. https://doi.org/10.1109/COMNETSAT56033.2022.9994546.

[31] P. Ekman, W.V. Friesen, Facial Action Coding System, American Psychological Association (APA), 1978. https://doi.org/10.1037/t27734-000.

[32] Y. Shuang, G. Liangbo, Z. Huiwen, L. Jing, C. Xiaoying, et al., Classification of Pain Expression Images in Elderly with Hip Fractures Based on Improved ResNet50 Network, Front. Med. 11 (2024), 1421800. https://doi.org/10.3389/fmed.2024.1421800.

[33] M. Dragomir, C. Florea, V. Pupezescu, Automatic Subject Independent Pain Intensity Estimation Using a Deep Learning Approach, in: 2020 International Conference on e-Health and Bioengineering (EHB), IEEE, 2020, pp. 1-4. https://doi.org/10.1109/EHB50910.2020.9280190.

[34] X. Liang, J. Liang, T. Yin, X. Tang, A Lightweight Method for Face Expression Recognition Based on Improved MobileNetV3, IET Image Process. 17 (2023), 2375–2384. https://doi.org/10.1049/ipr2.12798.

[35] R. Archana, P.S.E. Jeevaraj, Deep Learning Models for Digital Image Processing: A Review, Artif. Intell. Rev. 57 (2024), 11. https://doi.org/10.1007/s10462-023-10631-z.

[36] A. Khanapure, H. Kashyap, A. Bidargaddi, S. Habib, A. Anand, et al., Bone Fracture Detection with X-Ray Images Using MobileNet V3 Architecture, in: 2024 IEEE 9th International Conference for Convergence in Technology (I2CT), IEEE, 2024, pp. 1-8. https://doi.org/10.1109/I2CT61223.2024.10544356.

[37] J. Mao, L. Yu, Convolutional Neural Network Based Bi-Prediction Utilizing Spatial and Temporal Information in Video Coding, IEEE Trans. Circuits Syst. Video Technol. 30 (2020), 1856–1870. https://doi.org/10.1109/TCSVT.2019.2954853.

[38] S. Liu, X. Zhou, H. Chen, Multiscale Temporal Dynamic Learning for Time Series Classification, IEEE Trans. Knowl. Data Eng. 37 (2025), 3543–3555. https://doi.org/10.1109/tkde.2025.3542799.

[39] J. Li, X. Liu, W. Zhang, M. Zhang, J. Song, et al., Spatio-Temporal Attention Networks for Action Recognition and Detection, IEEE Trans. Multimed. 22 (2020), 2990–3001. https://doi.org/10.1109/tmm.2020.2965434.

[40] R. Fernandes-Magalhaes, A. Carpio, D. Ferrera, D. Van Ryckeghem, I. Peláez, et al., Pain E-Motion Faces Database (PEMF): Pain-Related Micro-Clips for Emotion Research, Behav. Res. Methods 55 (2022), 3831–3844. https://doi.org/10.3758/s13428-022-01992-4.

[41] A. Bansal, Image Processing Results of Chlamydia Pneumonia X-Ray, TechRxiv:21778694 (2023). https://doi.org/10.36227/techrxiv.21778694.

[42] A.F. Agarap, Deep Learning using Rectified Linear Units (ReLU), arXiv:1803.08375, (2018). https://doi.org/10.48550/arXiv.1803.08375.

[43] A. Howard, M. Sandler, B. Chen, W. Wang, L. Chen, et al., Searching for MobileNetV3, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, 2019, pp. 1314-1324. https://doi.org/10.1109/iccv.2019.00140.

[44] F. Chollet, Deep Learning with Python, Simon and Schuster, (2021).

[45] L. Alzubaidi, J. Zhang, A.J. Humaidi, A. Al-Dujaili, Y. Duan, et al., Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions, J. Big Data 8 (2021), 53. https://doi.org/10.1186/s40537-021-00444-8.

[46] J. Qiu, B. Wang, C. Zhou, Forecasting Stock Prices with Long-Short Term Memory Neural Network Based on Attention Mechanism, PLOS ONE 15 (2020), e0227222. https://doi.org/10.1371/journal.pone.0227222.

[47] J. Zirk-Sadowski, D. Szucs, J. Holmes, Content-Specificity in Verbal Recall: A Randomized Controlled Study, PLoS ONE 8 (2013), e79528. https://doi.org/10.1371/journal.pone.0079528.

[48] S. Albelwi, A. Mahmood, A Framework for Designing the Architectures of Deep Convolutional Neural Networks, Entropy 19 (2017), 242. https://doi.org/10.3390/e19060242.

[49] Q. Lin, S. Zhao, D. Gao, Y. Lou, S. Yang, et al., A Conceptual Model for the Coronavirus Disease 2019 (COVID-19) Outbreak in Wuhan, China with Individual Reaction and Governmental Action, Int. J. Infect. Dis. 93 (2020), 211–216. https://doi.org/10.1016/j.ijid.2020.02.058.

[50] X. Zhao, X. Liang, L. Liu, T. Li, Y. Han, N. Vasconcelos, S. Yan, Peak-Piloted Deep Network for Facial Expression Recognition, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, Springer, Cham, 2016: pp. 425–442. https://doi.org/10.1007/978-3-319-46475-6_27.

[51] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, Cambridge, (2016).

[52] S. Li, W. Deng, Deep Facial Expression Recognition: A Survey, IEEE Trans. Affect. Comput. 13 (2022), 1195–1215. https://doi.org/10.1109/taffc.2020.2981446.