EAI Endorsed Transactions
on Industrial Networks and Intelligent Systems Research Article **EAI.EU**

Single-level Discrete Two Dimensional Wavelet Transform Based Multiscale Deep Learning Framework for Two-Wheeler Helmet Classification

Amrutha Annadurai[1], Manas Ranjan Prusty[2], Trilok Nath Pandey[1, *], Subhra Rani Patra[3]

[1]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India
[2]Centre for Cyber Physical Systems, Vellore Institute of Technology, Chennai, India
[3]Information Systems and Operations Management, University of Texas at Arlington, Texas, 76019, United States

Abstract

INTRODUCTION: A robust method is proposed in this paper to detect helmet usage in two-wheeler riders to enhance road safety.
OBJECTIVES: This involves a custom made dataset that contains 1000 images captured under diverse real-world scenarios, including variations in helmet size, colour, and lighting conditions. This dataset has two classes namely with helmet and without helmet.
METHODS: The proposed helmet classification approach utilizes the Multi-Scale Deep Convolutional Neural Network (CNN) framework cascaded with Long Short-Term Memory (LSTM) network. Initially the Multi-Scale Deep CNN extracts modes by applying Single-level Discrete 2D Wavelet Transform (dwt2) to decompose the original images. In particular, four different modes are used for segmenting a single image namely approximation, horizontal detail, vertical detail and diagonal detail. After feeding the segmented images into a Multi-Scale Deep CNN model, it is cascaded with an LSTM network.
RESULTS: The proposed model achieved accuracies of 99.20% and 95.99% using both 5-Fold Cross-Validation (CV) and Hold-out CV methods, respectively.
CONCLUSION: This result was better than the CNN-LSTM, dwt2-LSTM and a tailor made CNN model.

Keywords: Multi Scale CNN; Long Short-Term Memory; Discrete Wavelet Transform; Helmet Detection

Received on 19 October 2024, accepted on 04 March 2025, published on 11 March 2025

doi: 10.4108/eetinis.v12i2.7612

*Corresponding author. Email: triloknath.pandey@vit.ac.in,

1. Introduction

In recent times, there has been a dramatic shift in the modes of transportation, with a noticeable rise in the use of motorbikes. This trend is the result of both the growing middle class's need for affordable transportation options and the growing urbanization of the population, which has increased demand for mobility. To meet these changing needs for urban mobility, there has consequently been a noticeable increase in demand for individualized private transportation services [1]. When road safety officials want to compel motorcyclists to wear helmets, they usually set up checkpoints where they visually inspect riders to ensure compliance. According to the law, there are fines for not wearing a helmet. This strategy is vulnerable to evasion techniques, too, including using detours to get around
checkpoints. As a result, to address the issue of low helmet wear rates and reduce the number of fatalities and injuries in traffic incidents involving motorcycles helmet detection system is needed.

Deep learning, a kind of artificial intelligence, allows machines to recognize patterns and features straight from data, it has revolutionized a number of fields. Convolutional Neural Network (CNN), which is trained on enormous datasets of annotated photos to acquire intricate representations of helmets, has demonstrated outstanding abilities in the recognition and localization of objects in images when it comes to helmet detection. Deep learning is enhanced by computer vision algorithms, which analyse and prepare images before feeding them into neural networks. Image enhancement and feature extraction are two methods that improve the quality of the data and help with helmet detection in a variety of situations, such as changing illumination and rider postures [2].

Automated helmet recognition systems quickly and accurately evaluate input photos in real-time by combining deep learning and computer vision. This allows the systems to detect helmets reliably and trigger the necessary actions to maintain road safety. We should expect even more accurate and effective helmet identification systems in the future as these technologies develop further, greatly improving rider safety. Over approximately 1.35 million individuals globally face fatalities as a result of road accidents each year, with children and young adults being the most affected by this alarming figure. Approximately 28% of these fatalities include people who are operating two- or three-wheeled vehicles. Roughly half of all motorcycle fatalities are caused by head and neck trauma, this stands as a prominent cause of mortality and serious injury. Effective helmet use has been found to be a critical intervention, lowering the likelihood of fatal injuries decreases by 42%, and there is a 69% reduction in the occurrence of brain injuries in traffic accidents.

Given the rising death toll from motorcycle accidents, an automated computer vision system that can identify two-wheeler riders and assess whether they are wearing helmets is desperately needed. A system like this would reduce the amount of work that traffic cops had to do, which would lower the death rate from motorcycle accidents. The principal objective is to reduce these types of traffic accidents by deploying an automated system that uses computer vision to detect when two-wheeler riders are wearing helmets. But despite its demonstrated efficacy, there is still a worrying trend: more than 50% of motorcycle riders in low- to middle-income countries (LMICs) prefer not to wear helmets [3]. There are several reasons for this hesitation, including the weight of the helmet, pain from the heat, sensory issues, and cultural beliefs. In order to improve road safety and reduce motorcycle-related deaths and injuries globally, it is imperative that these obstacles be removed.

Automation can reduce the number of human resources needed for operation while improving the robustness and reliability of such systems. It's crucial to remember, nevertheless, that placing security cameras everywhere might not be a practical or financially advantageous solution—especially in light of the fact that many streets lack monitoring cameras. The implementation of security cameras needs to be properly thought out and balanced with practicality and cost-effectiveness concerns, even if they can be useful tools for monitoring and improving safety. When choosing where to place surveillance systems, it's important to consider aspects like power source accessibility, network connectivity, upkeep needs, and privacy issues. Ultimately, even while automation and surveillance technologies can greatly improve security and safety, their implementation requires careful planning and awareness of real-world limits in order to be both successful and long-lasting. Organizations and authorities can reduce costs and logistical issues while maximizing the effectiveness of surveillance initiatives by effectively leveraging available resources and technologies.

The objective is to create an automated and dependable system for detecting helmets worn by two-wheeler riders in order to impose helmet usage laws and improve road safety. This involves developing and putting into practice two different approaches—multiscale deep CNN cascaded Long Short-Term Memory (LSTM) and customized CNN—and assessing how well they work to identify helmet use in a variety of real-world settings. The goal also includes to integrate the automated detection system into motorcycles in order to provide real-time traffic enforcement.

The subsequent divisions of this paper are structured as follows. In Section 2, the approaches currently in use helmet detection are discussed. Section 3 offers a synopsis of the objectives and contributions of this paper. The suggested technique is elucidated in Section 4, while Section 5 presents the results along with their respective explanations. The paper is finally concluded in Section 6.

2. Related Works

Using a variety of approaches, numerous studies have been carried out to address the problem of automatically detecting helmet usage among riders of two-wheelers. A technique for helmet violation identification using Faster RCNN on surveillance films taken by cameras positioned along the side of the road was presented by Waris et al. [4]. Their method's remarkable 97.6% accuracy allowed for real-time helmet regulation enforcement and monitoring. Nonetheless, restrictions were mentioned, including data collecting and possible data exploitation. From the perspective of cameras positioned on vehicles, Mercado Reyna et al. [5] used Inception v3 CNN to identify riders who were wearing helmets and those who weren't. With a high accuracy of 97.24%, their system was able to detect things in real time. However, one possible disadvantage was the requirement for interaction with databases of motorbike registrations. With a single CNN network, Narong et al. [6] were able to recognize motorcycles and riders with an accuracy of 85.19%. Even while it made processing easier, the low accuracy required work. Dasgupta et al. [7] achieved a 96.23% accuracy rate in helmet recognition using CNN and YOLOv3. Although it fared better than other CNN architectures, one drawback was that it required two stages of processing for detection.

For helmet detection, Singh et al. [8] investigated a number of machine learning techniques, such as CNNs and k-nearest neighbour. Their method not only detected helmets but also included number plate detection for legal cases, with an astounding accuracy of 99.2%. But the use of surveillance to ensure helmet compliance has drawn criticism. In order to identify helmet usage, Shine et al. [9] used a modified CNN to analyse traffic footage. Limitations were observed in the continuous monitoring throughout the driver's trip, even with a high accuracy of 96.98%. With an accuracy of 80.6%, Lin et al. [10] presented a CNN-based multi-task learning technique for recognizing and tracking individual motorcycles. On the other hand, it was determined that the current methodology's practicality was only partially fulfilled. Faster R-CNN was used by Afzal et al. [1] to detect helmets with a 97.26% accuracy rate. Predictive analytics and real-time data insights were provided; nonetheless, issues with data collecting and possible misuse were brought up. With pre-trained object detection algorithms, Rohith et al. [11] achieved 94.70% accuracy for real-time monitoring. Nonetheless, the accuracy attained was constrained by available resources.

With an accuracy of 80.9%, Cheng et al. [12] presented SAS-YOLOv3-tiny for helmet detection. The system was found to be less accurate than heavyweight models, although having a higher processing speed. YOLOv5 was used by Jia et al. to identify motorcycles and helmets, with recall and precision rates of 97.2% and 98.0%, respectively. However, it was noted that the multi-stage operation presented issues in obtaining real-time speed. YOLOv4-Darknet and YOLOv5s were used by Kanakaraj et al. [13] to detect license plates and helmets from traffic surveillance cameras. Although mAP of 67.67% and precision of 51.06% were attained, one constraint identified was the use of distinct procedures for license plate and helmet detection.

Yogameena et al. achieved a moderate accuracy of 79% by using GMM for ground labelling and Faster-RCNN for detection. It was understood that obtaining high accuracy presents challenges. Achieving an accuracy of 94.23%, Shuai et al. [14] used UAV aerial photography with LMNet and RT3DsAM. Although it is inexpensive and provides mobility, questions have been raised about the resilience and accuracy of helmet detection. With a deep learning model based on ResNet50, papers [14-15] successfully detected helmets. But the poor precision highlighted the areas that needed work.

The aforementioned results highlight the continuous endeavours to improve automated helmet identification systems and tackle obstacles related to practical implementation.

3. Motivation and Contributions

The paper provides a novel helmet identification method based on the use of a specially curated dataset that covers a wide range of real-world scenarios. Our methodology places a strong emphasis on creating a comprehensive dataset specifically designed for helmet identification tasks, in contrast to traditional approaches that rely on general datasets. This bespoke dataset of 1000 images covers a broad range of circumstances, such as differences in the size, colour, illumination, and ambient settings of helmets that are used on highways. We hope to increase detection performance, strengthen model generalization, and guarantee the practicality of our helmet detection system by utilizing this unique dataset. The development of this dataset also benefits the larger research community by offering an invaluable tool for evaluating, contrasting, and improving helmet detection techniques and algorithms.

The paper provides a novel multiscale feature extraction strategy that uses the Single-level Discrete 2D Wavelet Transform (dwt2) to break down images into various analytical scales, thus offering a comprehensive way to analyze images. By applying the dwt2, images can be broken down into a hierarchy of spatial frequencies, which allows for the capture of both fine and coarse features found in the original image. Our model can identify and evaluate patterns at various granularities thanks to this multiscale decomposition, which efficiently captures intricate visual information that might not be seen at a single scale. In particular, four different modes exist for segmenting a single image: approximation, horizontal detail, vertical detail and diagonal detail after feeding the segmented images into a multiscale deep CNN model, it is cascaded with an LSTM network for further processing in the image segmentation process. Our approach is well-suited for a variety of computer vision applications because it achieves higher performance in tasks like object identification, classification, and segmentation by integrating multiscale representation into our model.

Our customized CNN architecture is designed with helmet identification in mind, adding domain-specific improvements and heuristics to standard CNN architectures. We enhance the model's capability to precisely identify helmets in a variety of settings and circumstances by tailoring the architecture to the particular difficulties related to helmet recognition in practical contexts. With the help of this customized strategy, the model's discriminative ability is improved, allowing it to reliably and precisely discern helmets from other objects and background features.

Furthermore, the architecture is designed to manage the intricacies involved in helmet identification duties, including changes in the appearance, position, and occlusion of the helmet. We improve the model's resilience and performance in demanding real-world scenarios by customizing the CNN architecture to meet these unique needs. This eventually increases the model's efficacy in guaranteeing rider safety through precise helmet recognition.

In the field of camera-captured picture helmet detection, this work presents two unique methods. Important findings from this study include:

The creation of a novel cascaded CNN-LSTM model with mode extraction that makes use of dwt2 and is intended to identify helmets in a variety of situations. Extracting channels or sub-bands from camera-captured pictures using a fixed boundary-based Single-level dwt2 filter bank to enable effective processing. By utilizing dwt2's multiscale decomposition capabilities, our approach successfully manages partial occlusion. The method allows for the capture of fine-grained features by dividing images into approximation, horizontal, vertical, and diagonal detail sub-bands. This makes it possible for our system to identify helmets even when certain elements of them are obscured by the surrounding. On the other hand, earlier techniques usually depend on whole-image processing, which often overlooks critical features under occlusion.

Helmets come in a wide range of sizes, colours, shapes, and designs, further complicated by varying lighting conditions and rider postures. The multiscale analysis provided by dwt2 enhances the model's ability to capture subtle variations in texture and structure, making it more robust across diverse real-world scenarios. This attention to fine-scale variations helps the model better adapt to the complexity of different helmet types and conditions that other methods might miss.

Implementation of a custom CNN architecture that has been created to accommodate the complexities of helmet recognition in camera-captured images. By pre-processing images through hierarchical spatial frequencies, dwt2 enriches the feature extraction process of the CNN, providing more detailed and robust input. When coupled with the temporal context from LSTM, the model effectively captures both spatial and temporal variations, improving performance for dynamic and static helmet-use detection scenarios. Previous methods may have combined CNN and LSTM architectures, but without the multiscale decomposition offered by dwt2, they are less equipped to address the intricate variability seen in real-world datasets. Using CNN and LSTM architectures at different scales in concert to recognize helmets in a variety of camera-captured settings.

4. Dataset

A thorough data gathering and pre-processing approach was used for the helmet detection model. The self-curated dataset featured two main categories: pictures of people with helmets and pictures of people without them. Using cameras to take pictures and record videos, the data was collected. The videos were then divided into individual frames to create the dataset. In the instance of the customized CNN model, a total of 1000 images were used for training and assessment. To guarantee thorough coverage, a range of scenarios and conditions were included in these photographs. Similarly, a dataset of 1000 images was used for the cascaded deep-scale CNN-LSTM network. The dataset was carefully pre-processed to improve its quality and guarantee consistency between samples.

Our methodology places a strong emphasis on creating a comprehensive dataset specifically designed for helmet identification tasks, in contrast to traditional approaches that rely on general datasets. This bespoke dataset of 1000 images for Tailor-made CNN and 1000 images for Multiscale Deep CNN cascaded LSTM covers a broad range of circumstances, such as differences in the size, colour, illumination, and ambient settings of helmets that are used on highways. We have used this dataset to increase detection performance, strengthen model generalization, and guarantee the practicality of our helmet detection system by utilizing this unique dataset. The development of this dataset also benefits the larger research community by offering an invaluable tool for evaluating, contrasting, and improving helmet detection techniques and algorithms. This stable dataset makes sure that the various classes are distributed uniformly, which is easier to train and test the developed model for helmet identification. The sample images from the dataset is showed in Fig. 1.

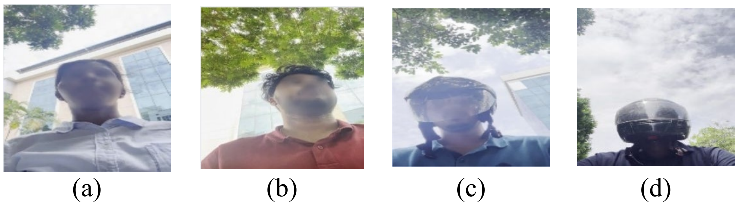


Figure 1. Images from the curated dataset containing people with or without helmet: (a) Without Helmet, (b) Without Helmet, (c) With Helmet, (d) With Helmet

5. Proposed Approach

The proposed methodology's experimentation process is described in Fig. 2 was applied to a dataset 1000 images that were taken in order to identify helmet wear among two-wheeler riders. The main goal of the implementation of this strategy was to increase road safety by enforcing laws pertaining to helmet wear. This dataset ensures thorough coverage of potential situations by incorporating a variety of real-world scenarios, such as differences in helmet size, colour, and lighting conditions. Two different approaches were put out to help with precise detection: one made use of a Multiscale Deep CNN cascaded LSTM framework, while the other made use of a custom CNN model. Using dwt2, the Multiscale Deep CNN architecture first extracts modes from the images, breaking down the original images into four categories to decompose the original images: approximation, horizontal detail, vertical detail, and diagonal detail images. In particular, these four different modes exist for segmenting a single image after feeding the segmented images into a multiscale deep CNN model, it is cascaded with an LSTM network for further processing in the image segmentation process using 5-Fold Cross-Validation (CV).

Overall, the customized CNN model and the cascaded deep-scale CNN-LSTM network were trained and evaluated using the helmet dataset that the author had carefully selected. The dataset made it possible to construct precise and dependable models for helmet detection in two-wheeler riders by combining a wide variety of images and scenarios, which improved road safety measures.

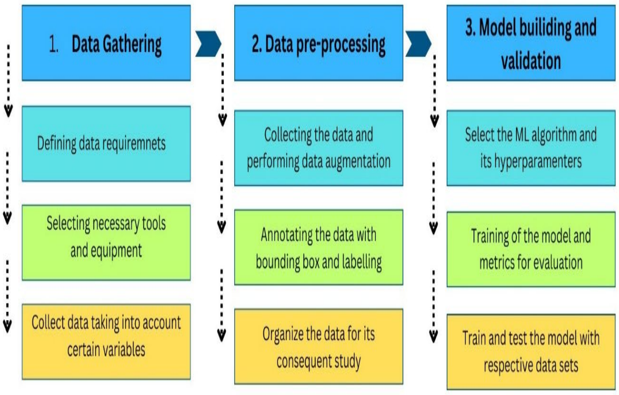

1. Data Gathering
2. Data pre-processing
3. Model builiding and validation
Defining data requiremnets
Collecting the data and performing data augmentation
Select the ML algorithm and its hyperparamenters
Selecting necessary tools and equipment
Annotating the data with bounding box and labelling
Training of the model and metrics for evaluation
Collect data taking into account certain variables
Organize the data for its consequent study
Train and test the model with respective data sets

Figure 2. Block diagram of the experimentation process

5.1 Utilizing Discrete Wavelet Transform for Image Decomposition and Mode Generation

The main elements and processes of the suggested helmet detection model is shown in Fig. 3. Using dwt2, the image is first broken down into four modes: approximation, diagonal, horizontal, and vertical. These modes are then carefully selected to serve as inputs for the multiscale deep CNN model, which is then combined with an LSTM network. This combination makes it easier to analyse and categorize helmet presence in a variety of real-world situations.

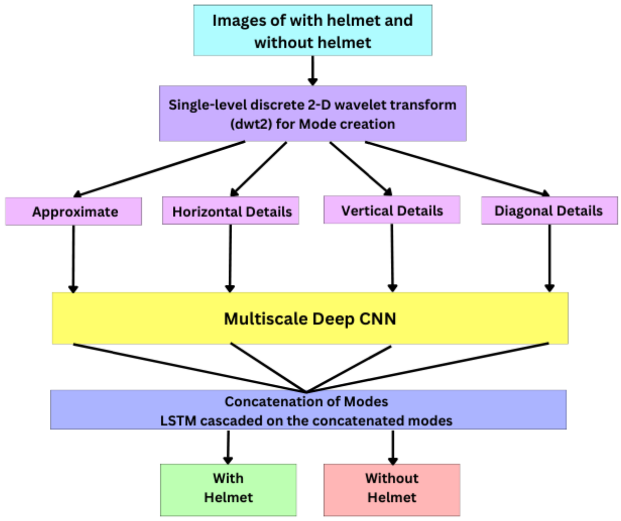

Images of with helmet and without helmet
Single-level discrete 2-D wavelet transform (dwt2) for Mode creation
Approximate
Horizontal Details
Vertical Details
Diagonal Details
Multiscale Deep CNN
Concatenation of Modes
LSTM cascaded on the concatenated modes
With Helmet
Without Helmet

Figure 3. Flow-diagram illustrating the proposed method for helmet detection.

A flexible technique for image processing, the dwt2 can be used for feature extraction, denoising, and picture compression. With this transformation, an image is divided into various frequency components, each represented by a collection of sub-bands. The dwt2 function is used to do the single-level 2-D wavelet decomposition is depicted in Fig. 4. Specifically designed wavelets or filters are used in this decomposition. The approximation coefficients matrix $[cA]$ and the matrices for the horizontal $[cH]$, vertical $[cV]$, and diagonal $[cD]$ features are obtained by applying wavelet decomposition to the input matrix X. The two-dimensional wavelet decomposition is carried out using the function $[cA, cH, cV, cD] = dwt2(X, LoD, HiD)$, which yields the previously indicated matrices. The low-pass filter (LPF) and high-pass filter (HPF) are used in this procedure, and their lengths must be comparable for optimal performance. During the decomposition process, these filters are essential in distinguishing the frequency components. We initiate the process with an input image of dimensions $A \times A$ and execute wavelet decomposition by applying both a low-pass filter (LPF) and a high-pass filter (HPF) to the rows of the image. These filters serve to divide the input data is into two components: one component contains low-frequency information while the other encapsulates high-frequency information. Subsequently, we repeat the application of the LPF and HPF to the columns of the image, generating two additional columns. The resulting sub-bands are approximation, horizontal detail, vertical detail, and diagonal detail shown in Fig. 5. This decomposition effectively dissects the image into its constituent frequency components.

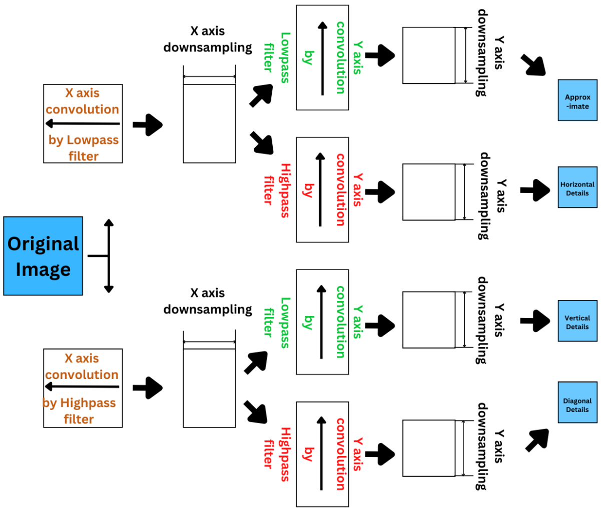

Original Image
X axis convolution by Lowpass filter
X axis downsampling
Lowpass filter by Y axis convolution
Y axis downsampling
Approx -imate
Highpass filter by Y axis convolution
Y axis downsampling
Horizontal Details
X axis convolution by Highpass filter
X axis downsampling
Lowpass filter by Y axis convolution
Y axis downsampling
Vertical Details
Highpass filter by Y axis convolution
Y axis downsampling
Diagonal Details

Figure 4. Representation of the Single-Level 2-D Wavelet Decomposition

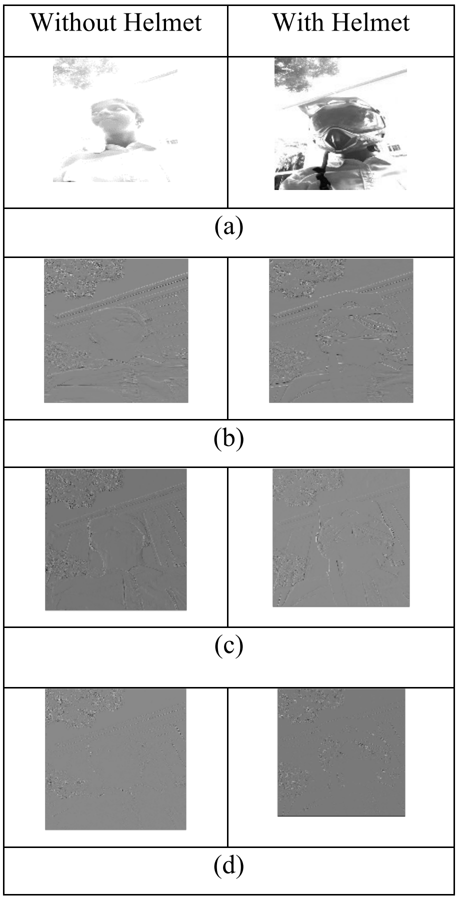


Figure 5. Sample representation of various modes of the images produced by dwt2 include: (a) Approximation, (b) Horizontal Detail, (c) Vertical Detail and (d) Diagonal Detail.

5.2 Multi-scale deep CNN cascaded LSTM

In the field of helmet identification systems, several approaches have been investigated to reliably identify whether or not helmets are present in pictures. These methods frequently rely on advanced deep learning models designed specifically to handle and evaluate visual data. Here, a Multi-Scale Deep CNN cascaded with an LSTM network shown in Fig. 6 is used as the deep model for helmet identification. LSTM networks are known for their ability to extract complex characteristics from images and capture temporal relationships.

The Convolutional layer, which is central to the CNN design, handles the majority of the processing required for the analysis of visual data. Through the utilization of this base layer, the model is able to recognize patterns and characteristics that indicate the presence or absence of a helmet. A unique classification system is developed utilizing a dataset of 1000 photos, each of which shows two-wheeler riders in a variety of real-world circumstances, in order to establish a strong helmet identification model. The collection of helmet sizes, colors, and lighting conditions that make up this dataset have been carefully chosen to reflect a wide range of possible roadside circumstances. The dataset is split using both 5-Fold Cross-Validation (CV) and Hold-out Validation, ensuring robust evaluation. During training, the model processes input images through 12 convolutional layers, each applying ReLU activation and batch normalization to stabilize learning. Max-pooling layers reduce spatial dimensions while preserving essential features. Extracted features are then passed to LSTM layers to capture temporal dependencies. The fully connected layers refine the features before classification is performed using a Softmax output layer. The model is trained for 25 epochs using the Adam optimizer with a learning rate of 0.001, a batch size of 64, and Sparse Categorical Cross-Entropy as the loss function. Regularization techniques such as dropout (set at 0.2) help prevent overfitting. The optimization process involves computing classification errors, backpropagating gradients, and updating weights iteratively. Model performance is assessed using accuracy, precision, recall, F1-score, and a confusion matrix, while accuracy and loss curves are monitored throughout training to detect overfitting or underfitting.

Every image in the suggested CNN-LSTM model is subjected to multiscale processing, with various convolution layers tasked with examining various facets of the input data. In particular, the CNN architecture consists of 12 layers, each of which is capable of handling the subtleties of the input images on its own. The model architecture is divided into several blocks, each of which consists of the following layers: dropout, max-pooling, batch normalization, input layer, and convolution layer. By applying these blocks to the input data in a sequential manner, it is possible to extract pertinent features at different processing stages. In order to capture temporal dependencies in the data and enable the model to evaluate image sequences and find patterns across time, the LSTM network is integrated into the CNN architecture. The model's capacity to precisely identify helmet usage in two-wheeler riders across a variety of real-world settings is improved by this cascading architecture. The computation of the feature map for the ith convolutional layer is performed as follows.

$$Z_{x,y}^{i} = f\left[\sum_{a=0}^{p-1}\sum_{b=0}^{q-1} K_{a,b}^{i} Z_{(x+a,y+b)}^{i-1} + b^{i}\right] \qquad (1)$$

The 2D kernel $K_{a,b}^{i}$ is convolved with the feature map $Z_{(x,y)}^{i-1}$ from the previous layer to produce the output feature map $Z_{x,y}^{i}$ of the ith convolutional layer. Batch normalization and ReLU activation are then applied to normalize inputs and speed up training. The output of Eq. (1) is transformed into the desired format by batch normalization (BN) calculation, guaranteeing standardized inputs and promoting effective training.

$$Q_{x,y} = \frac{z_{x,y}^{i} - \mu_Z}{\sigma_Z} \qquad (2)$$

Here, $Q_{x,y}$ is the updated value of each individual component; μ_Z is the average value across batch elements; and σ_Z is the standard deviation of the batch, which reflects its variability. The dropout layer in the proposed multiscale deep CNN comes after batch normalization, with the goal of preserving behaviour variety and avoiding overfitting by decorrelation of weights and preventing neuron convergence. The pooling layer evaluates the feature map following the dropout layer.

$$Z_{x,y}^{i} = \max[F_{(x+a,y+b)}^{i-1}] \qquad (3)$$

The cyclic connections between units of the recurrent neural network (RNN) enable it to capture temporal dependencies well. A variation that improves memory retention is the LSTM, which has input, output, and forget gates. After extraction, features are processed using 16 units of LSTM, then reshaped and dropped out by 0.2. Reshaped data then passes through two thick layers activated by ReLU, an LSTM, and a Softmax output layer. With this architecture, helmets can be detected with great accuracy by efficiently analyzing temporal fluctuations.

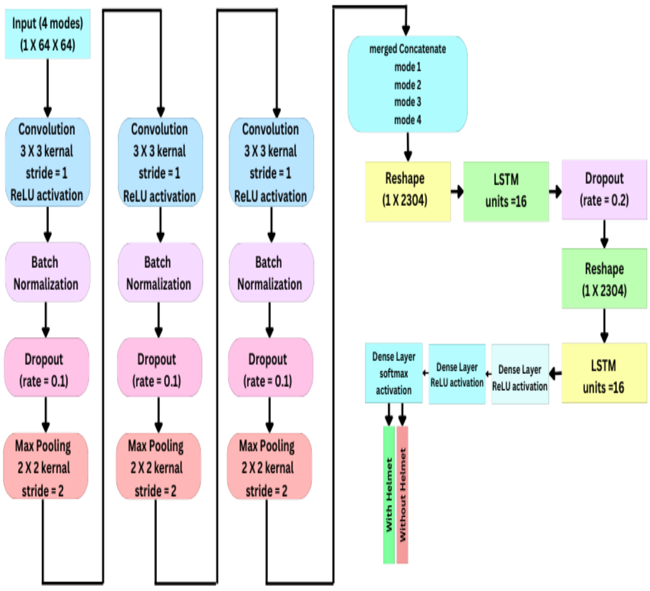


Figure 6. Proposed Multi-Scale Deep CNN cascaded LSTM architecture developed for the detection of helmet.

To maximize efficacy and assess robustness, the suggested model is trained over a span of 25 epochs using 5-Fold CV and Hold-out CV. Table 1 lists all of the deep CNN model's hyper-parameters. Evaluation includes confusion matrix-derived metrics such as accuracy, precision, recall, and F1-score to ensure a comprehensive comparison of model performances over 25 epochs using both validation procedures. This thorough analysis ensures a thorough evaluation of the model's effectiveness.

In this work, we focused on helmet detection by performing binary classification tasks on an image dataset. Depending on the particular classes, this classification method has different benefits and drawbacks. Positive cases for helmet detection are those that are accurately classified as wearing a helmet, and negative cases are those that don't. True Positive (TP) and True Negative (TN) denote occasions where a helmet was correctly identified, and where one was not, respectively. On the other hand, False Positive (FP) and False Negative (FN) indicate cases where the helmet was incorrectly identified and those where it was not. Additionally, we assessed the binary classification into two groups: those wearing helmets and those not. In this instance, cases that are affirmative have helmets on, whereas examples that are negative don't. TP and TN denote precisely recognized instances of helmets and absence of helmets, respectively, whereas FP and FN indicate misidentified instances.

Table 1. The hyper-parameters utilized in the proposed architecture of the Multi-Scale Deep CNN cascaded with LSTM.

Hyperparameters	Values for dataset
With Helmet	500
Without Helmet	500
Learning rate	0.001
Batch size	64
Epochs	25
Optimizer	Adam
Loss function	Sparse categorical cross-entropy

Tailor-made CNN Implementation

The goal of implementing helmet detection with a customized CNN model is to create a CNN architecture that can reliably determine whether or not helmets are present in Fig. 7. The CNN model extracts hierarchical information from the input images by first using many convolutional layers, which are then followed by max-pooling layers. Fully connected layers are then used to classify data according to the features that were extracted.

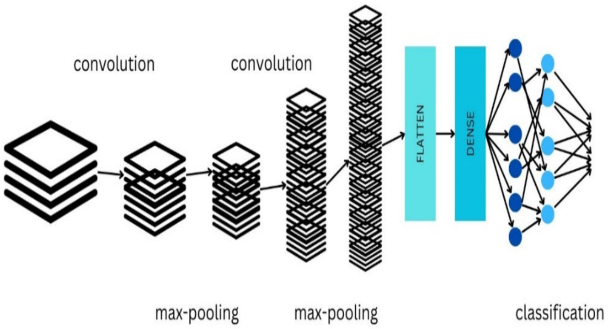


Figure 7. Proposed Tailor-made CNN architecture developed for the detection of helmet

One way to sum up the general architecture is as follows:

Input Layer: The input shape (640, 640, 3) is an RGB picture with 640 × 640 pixel size.

Convolution Layer: To provide non-linearity, three convolutional layers are used, each followed by an activation function that uses ReLUs. In order to capture more complicated characteristics, the number of filters in the convolutional layers continuously rises.

Sizes of filters: The symbol (3, 3) denotes a 3 by 3 filter window.

Maximum Pooling Layers: A max-pooling layer is employed to reduce the dimensionality of the feature maps through down sampling and lower dimensionality after each convolutional layer. The window size used for the max-pooling procedure is (2, 2).

Layer Flattening: To feed into the fully connected layers, the output of the last convolutional layer is flattened.

Completely Networked Layers: There are two dense (completely linked) layers used; the first layer has 512 units and is activated by the ReLU function. A single neuron employing a sigmoid activation function within the ultimate output layer generates a binary signal that indicates whether or not a helmet is present.

The convolution technique is essential to CNNs in order to extract features from the input data. In order to create feature maps, filters—also referred to as kernels—are applied to the input data. The input volume, filters, bias terms, and output feature maps are the essential elements of the convolution process.

Input Volume: Let's write $A^{[l-1]}$ for the input volume, where $l-1$ denotes the preceding layer. Activation maps, also known as feature maps, from the preceding layer make up this input volume. Each map represents a distinct feature or component of the input data.

Filters: Feature maps are generated by convolving a set of learnable parameters with the input volume, denoted as $W^{[l]}$. Different patterns or features found in the input data are captured by each filter. The height, width, and depth of filters are defined by these dimensions, which correspond to the depth of the input volume.

Bias Terms: Each convolutional layer normally contains bias terms, shown by $b^{[l]}$, in addition to filters. In order to help the model, learn the right output, these biases are scalar values applied to each element of the feature maps.

Convolution Equation: For a given layer l, the convolution operation computes the output feature map $Z^{[l]}$. This has the following mathematical representation:

$$Z^{[l]}_{i,j,k} = b^{[l]}_k + \left[\sum_{c=0}^{C^{[l-1]}-1} \sum_{r=0}^{F_h-1} \sum_{s=0}^{F_w-1} A^{[l-1]}_{i+r,j+s,c} \times W^{[l]}_{r,s,c,k} \right] \tag{4}$$

Where,

$Z^{[l]}_{i,j,k}$ is the activation at position (i,j) in the kth feature map of layer l.

$b^{[l]}_k$ is the bias term corresponding to the kth filter in layer l.

$C^{[l-1]}$ is the number of channels (depth) in the input volume.

F_h and F_w are the height and width of the filters, respectively.

$A^{[l-1]}_{i+r,j+s,c}$ is the activation at position (i+r, j+s) in the cth channel of the input volume.

$W^{[l]}_{r,s,c,k}$ is the weight parameter connecting the cth channel of the input volume to the kth filter at position (r,s).

Table 2. The hyper-parameters utilized in the proposed Tailor-made CNN model

Hyper-parameters	Values for dataset
With helmet instances	500
Without helmet instances	500
Learning rate	0.001
Batch size	32
Epochs	5
Optimizer	Adam
Loss function	Binary Cross-Entropy

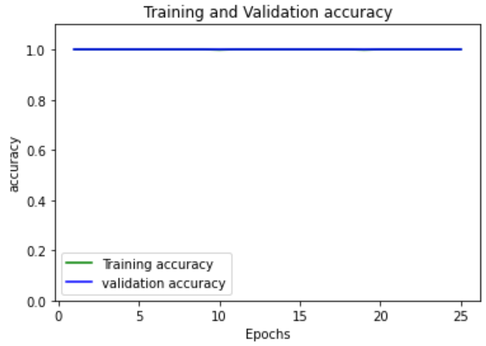
This CNN model is made to efficiently extract information from helmet photos and forecast whether or not a helmet will be present. The model strives for stable performance and high accuracy in real-world helmet recognition settings by fine-tuning its parameters and refining its architecture. Table 2 depicts the hyper parameters used for the proposed Tailor made CNN model. In our model, we have added a Multi-Scale Deep CNN cascaded with LSTM, which brings about significant computational demands, especially for training and real-time inference. Due to the deep CNN architecture that extracts multi-scale features, along with the sequential processing capability of LSTM, the model requires substantial GPU resources for training. This typically includes GPUs with at least 12GB of VRAM to accommodate the large number of parameters and intermediate activations. Training times can range from hours to days depending on the dataset size and the number of epochs. Additionally, the memory requirements for storing both CNN and LSTM components can be quite high. While TensorFlow Lite or model quantization can help optimize the model for real-time deployment on edge devices, achieving real-time performance without compromising accuracy remains challenging due to the complexity of the architecture.

In contrast, the simpler Tailor-made CNN model demands far fewer resources. Since it lacks the LSTM component and has a less complex architecture, the model requires less memory and can be trained more quickly, typically within a few hours on a standard GPU or even a powerful CPU. The real-time inference for this model is much more feasible, even on devices with limited computational capacity such as mobile phones or embedded systems. However, the trade-off comes in the form of reduced accuracy, as the model is less capable of handling complex patterns and environmental variations compared to the Multi-Scale Deep CNN + LSTM model. Thus, while the Tailor-made CNN model excels in speed and efficiency, it falls short in robustness and accuracy when compared to more sophisticated approaches like the one with LSTM.

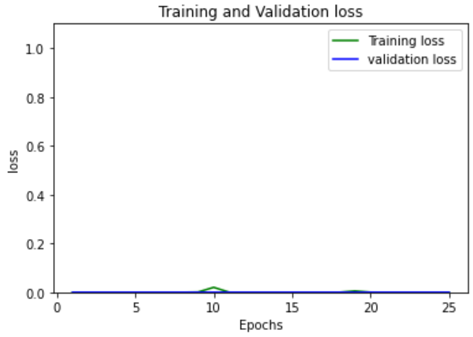
6. Result and Discussion

The outcomes of our suggested Multi-Scale Deep CNN model with LSTM which incorporates modes from the dwt2 image decomposition as well as the Tailor-made CNN model are shown in this section. The inclusion of dwt2 decomposition appears to have contributed to better extraction of meaningful features from the images, improving classification performance. The Multi-Scale Deep CNN, integrated with LSTM for temporal sequence learning, offers a significant advantage in learning complex spatiotemporal features, crucial for accurate helmet detection in real-world scenarios. Plots of accuracy and loss against epochs for the Multi-Scale Deep CNN cascaded with LSTM model and the Tailor-made CNN model, respectively, are shown in Figures 8 and 9. Remarkably, by utilizing several mode combinations, the Multi-Scale Deep CNN model in conjunction with LSTM was able to attain remarkable training accuracies of 95.99% and 99.20%, respectively, while utilizing 5-Fold CV and Hold-out CV approaches. This demonstrates how reliable and efficient our technology is at correctly identifying helmet wear in a variety of settings. By comparison, the accuracy of the Tailor-made CNN approach, which used our curated collection of photos, was significantly lower. In particular, it obtained accuracy of 54.86% and 55.26% with 5-Fold CV and Hold-out CV techniques, in that order.

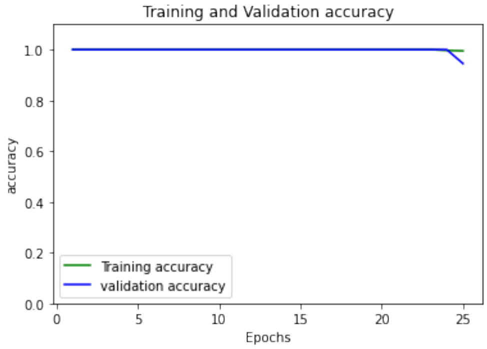
Although this method served as the basis for our investigation, the notable difference in performance emphasizes the superiority of the Multi-Scale Deep CNN cascaded with LSTM model, especially with the addition of dwt2 image decomposition. The use of dwt2 decomposition helps break down images into multiple frequency components, which is essential for capturing both low and high-frequency features. This is particularly useful for identifying helmets in diverse environmental conditions, where different frequency patterns (such as lighting changes or occlusions) might be present. The remarkable accuracy improvements suggest that the DWT2 method offers a way to extract relevant patterns that simpler CNN models fail to capture. This strengthens the Multi-Scale Deep CNN model's ability to generalize well across various image conditions. These findings highlight how well our suggested method works to improve the accuracy of helmet identification, which advances the development of road safety protocols for riders of two-wheelers.



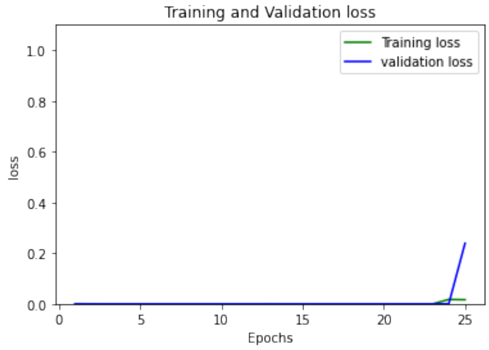
(a)



(b)



(c)



(d)

Figure 8. (a) Accuracy versus epochs graph using the Deep Multi-Scale CNN cascaded LSTM network for training and validation data of the results obtained using 5-Fold CV. (b) Loss versus epochs graph using the Deep Multi-Scale CNN cascaded LSTM network for training and validation data of the results obtained using 5-Fold CV. (c) Accuracy versus epochs graph using the Deep Multi-Scale CNN cascaded LSTM network for training and validation data of the results obtained using Hold-out CV. (d) Accuracy versus epochs graph using the Deep Multi-Scale CNN cascaded LSTM network for training and validation data of the results obtained using Hold-out CV.

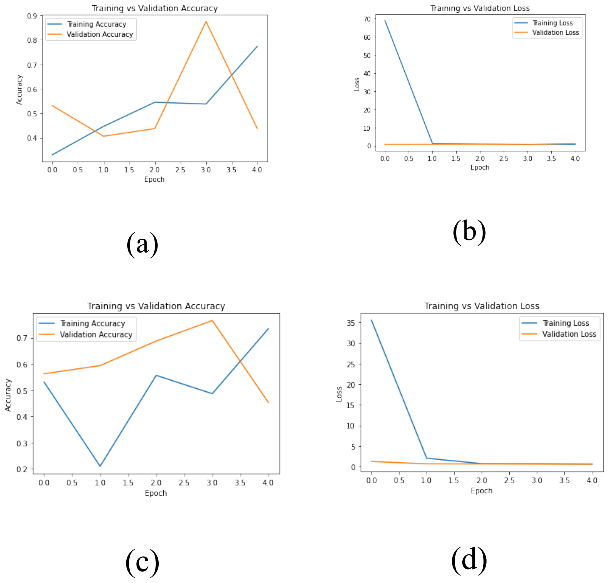
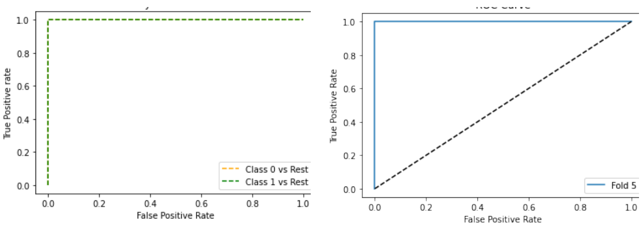


Figure 9. (a) Accuracy versus epochs graphs using the Tailor-made CNN model for training and validation data of the results obtained using 5-Fold CV. (b) Loss versus epoch graphs using the Tailor-made CNN model for training and validation data of the results obtained using 5-Fold CV. (c) Accuracy versus epochs graphs using the Tailor-made CNN model for training and validation data of the results obtained using Hold-out CV. (d) Loss versus epoch graphs using the Tailor-made CNN model for

training and validation data of the results obtained using Hold-out CV.

The True Positive Rate and False Positive Rate pertaining to the suggested Multi-Scale Deep CNN cascaded model with LSTM and Tailor-made CNN model are shown on the Receiver Operating Characteristic (ROC) curve in Figure 10. Alternatively, Figure 11 illustrates the confusion matrix plots, offering a visual depiction of the evaluation of the model's performance and guarantee originality devoid of plagiarism.

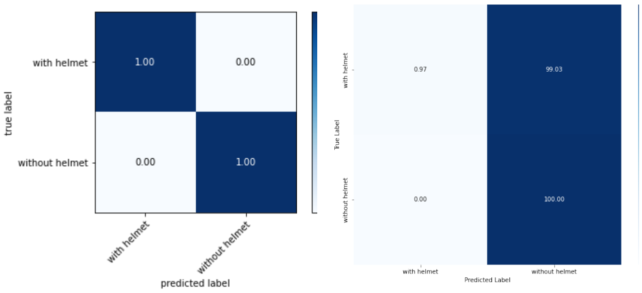
The above accuracy rates demonstrate how well the model was able to classify the two classes, guaranteeing uniqueness. With a 5-Fold CV accuracy of 0.992 for Multi-Scale Deep CNN cascaded LSTM and a custom CNN model of 0.5486. Performance metrics that offer a thorough understanding of the model's performance are displayed in Table 3 and include testing accuracy (ACC), precision (PRE), recall or sensitivity (REC), and the F1-score. Figure 12 shows the final results from this architecture.



(a)

(b)

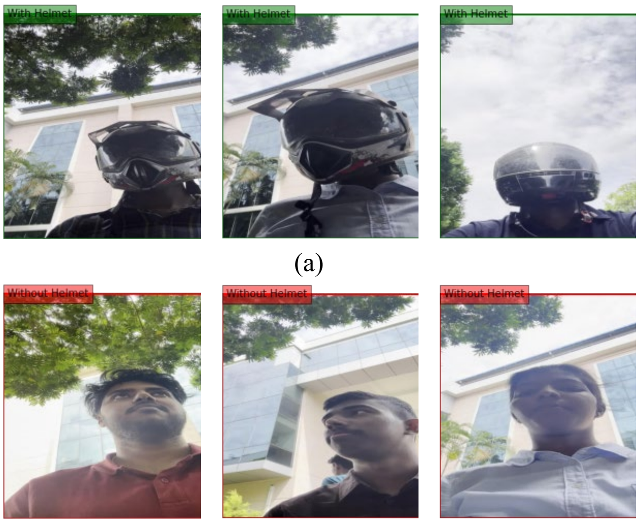
Figure 10. ROC curve for the proposed model for training and validation data of the results obtained: (a) deep multi-scale CNN cascaded LSTM network (b) the Tailor-made CNN model



(a)

(b)

Figure 11. Confusion Matrix: (a) deep multi-scale CNN cascaded LSTM network (b) the Tailor-made CNN model



(a)

(b)

Figure 12. Final output of the proposed model: (a) With Helmet (b) Without Helmet

As shown in Table 3 the performance of the Multi-Scale Deep CNN cascaded LSTM exceeds Tailor-made CNN model. Our goal is to improve road safety by creating a reliable model for identifying helmet use among two-wheeler riders. As explained, unique datasets were gathered designed especially for helmet detection. Because of this, the datasets used for the comparison analysis are specific to the suggested models and not found in other methods. A comparison summary of the suggested approach and earlier methods for helmet detection in two-wheeler riders may be seen in Table 4.

The results clearly emphasize the Multi-Scale Deep CNN cascaded with LSTM model as the superior approach for helmet detection, with its higher accuracy, robustness to environmental changes, and generalization capabilities. The inclusion of DWT2 decomposition and multi-scale learning ensures the model can better understand complex patterns, which is essential for real-world applications. This model significantly contributes to improving road safety for two-wheeler riders, demonstrating its potential in practical implementations for helmet detection. State-of-the-art helmet detection systems, such as those by [3] and [4], typically use CNN-based approaches, which provide high accuracy due to their ability to capture complex visual patterns. However, CNNs can be computationally intensive, making real-time detection challenging unless optimized, for instance, by using TensorFlow Lite or model quantization. [9] show that while hand-crafted features can offer faster processing speeds, CNNs generally outperform them in accuracy and robustness to environmental changes, as they are more adaptable to varied lighting and occlusions. Meanwhile, [12] leverage YOLOv3-Tiny, which balances accuracy and speed, making it more suitable for real-time applications, although it might struggle in extreme conditions without further adaptations. On the other hand,

[14] integrate residual transformers and spatial attention mechanisms, enhancing both accuracy and the system's ability to handle challenging environments. However, these methods are typically more computationally expensive, impacting processing speed unless optimized. [1] and [5] adopt deep learning methods with CNNs, achieving strong accuracy but requiring optimizations for speed and robustness. In summary, while CNNs and advanced architectures like YOLO and transformers offer the best accuracy and robustness, trade-offs with processing speed exist. Models optimized for real-time performance, such as SAS-YOLOv3-Tiny, excel in processing speed but might need enhancements to handle extreme environmental factors effectively. The table highlights important findings and highlights the unique benefits and contributions provided by the model, as well as its originality about previous approaches.

Table 3. Classification performance of proposed multiscale deep CNN cascaded LSTM along with other built models.

Proposed Model	***5-fold CV***				***Hold-out CV***			
	ACC	***PRE***	***REC***	***F1***	***ACC***	***PRE***	***REC***	***F1***
Multi-scale deep CNN-LSTM	0.992	0.926	0.920	0.962	1.00	1.00	1.00	1.00
Tailor-made CNN	0.548	0.548	1.00	0.708	0.553	0.551	1.00	0.710
CNN-LSTM	0.898	0.995	0.995	0.995	0.990	0.980	0.990	0.990
dwt2-CNN	0.904	1.00	1.00	1.00	0.519	0.270	0.520	0.356

Table 4. Comparative analysis of current approaches for the automated detection of helmets

Ref. no.	**Method**	**Accuracy**
Proposed	**Multiscale deep CNN-LSTM**	**99.20%**
[4]	Faster RCNN	97.6%
[5]	Inception v3 CNN	97.24%
	CNN	96.98%
[10]	CNN	80.6%
[12]	SAS YOLOv3	80.9%
[13]	YOLOv4 Darknet	67.67%
[15]	ResNet50	72.8%
[1]	Faster R-CNN	97.26%
[6]	Simple CNN	85.19%

The Multi-Scale Deep CNN-LSTM model, due to its more complex architecture with multi-scale feature extraction and sequential learning via LSTM, requires significant computational resources for both training and inference. Training this model would demand powerful GPUs with ample memory (e.g., 12GB VRAM) and considerable time, often spanning hours to days depending on dataset size and hardware. The inclusion of LSTM adds complexity, increasing memory usage due to maintaining sequence states, which makes real-time inference challenging unless the model is optimized through techniques like model quantization or using frameworks like TensorFlow Lite for edge devices. Despite these optimizations, achieving real-time performance could still be difficult without trading off some accuracy or processing speed.

On the other hand, the Tailor-made CNN model is much simpler and thus less computationally demanding. It requires fewer parameters and lacks the sequential learning of LSTM, making it quicker to train and more suitable for real-time inference, even on less powerful devices. With reduced memory requirements, this model can run efficiently on systems with lower computational capabilities. However, the trade-off for faster processing is lower accuracy, as it does not capture the complex spatial and temporal features that the multi-scale, LSTM-enhanced CNN model can extract. Therefore, while the Tailor-made CNN can deliver faster results, it may not perform as robustly across varied and complex real-world scenarios.

7. Conclusion

In conclusion, the goal of our study was to create a reliable model for helmet identification in order to improve road safety, especially for riders of two wheels. We have successfully developed a deep learning-based model that can reliably recognize helmet usage in a variety of real-world settings, after conducting a thorough investigation and experimentation. The development of an extensive dataset especially designed for helmet detection was a significant success. This dataset allowed for efficient training and assessment of our model's performance in various environmental circumstances because it included a wide range of helmet sizes, colors, and lighting conditions. We achieved promising results with our technique, which combines a Multi-Scale Deep CNN architecture with an LSTM network. We obtained 99.20% accuracy using 5-Fold Cross-Validation (CV) and 95.99% accuracy using Hold-out CV. Furthermore, our first method with a customized CNN model achieved 54.86% and 55.26% accuracy with 5-Fold CV and Hold-out CV, respectively. We contrasted our suggested model with other methods to confirm its superiority. In particular, utilizing 5-Fold CV and Hold-out CV, the CNN-LSTM model obtained 89.80% and 99.00% accuracy, respectively, while the dwt2-LSTM model obtained 90.40% and 51.90% accuracy, respectively. These findings demonstrate how well our approach works to increase the accuracy of helmet identification, which benefits motorcyclists' safety on the road.

Furthermore, our research provided new perspectives on the difficulties and nuances involved in actual helmet detection situations. We also identified other possible uses for our approach than standalone detection, such incorporating it into electric cars to enforce helmet use laws for the safety of riders. There are numerous opportunities for more work and advancement in the future. First off, improving the model's functionality in difficult lighting and weather scenarios could make it more useful in real-world scenarios. Furthermore, a viable avenue for additional study and development is the integration of the helmet recognition system into electric vehicles to enforce automatic start-stop functionality depending on helmet presence.

Acknowledgement

The authors would like to thank Vellore Institute of Technology (VIT), Chennai for providing the environment for this research. The authors also would like to express their gratitude to School of Computer Science and Engineering, VIT Chennai for their continuous support and encouragement.

Conflict of Competing Interests

The authors have no conflicts of interest to declare that are relevant to the content of this article.

Data availability and access

The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

References

[1] A. Afzal, H. Umer, Z. Khan, and M. U. Khan, “Automatic Helmet Violation Detection of Motorcyclists from Surveillance Videos using Deep Learning Approaches of Computer Vision,” Apr. 2021, pp. 252–257. doi: 10.1109/ICAI52203.2021.9445206.

[2] D. M. Trupthi, “Helmet Detection Based on Convolutional Neural Networks,” vol. 09, no. 06, 2022.

[3] A. Soni and A. Singh, “Automatic Motorcyclist Helmet Rule Violation Detection using Tensorflow & Keras in OpenCV,” Feb. 2020, pp. 1–5. doi: 10.1109/SCEECS48394.2020.55.

[4] T. Waris et al., “CNN-Based Automatic Helmet Violation Detection of Motorcyclists for an Intelligent Transportation System,” Math. Probl. Eng., vol. 2022, pp. 1–11, Oct. 2022, doi: 10.1155/2022/8246776.

[5] J. Mercado Reyna et al., “Detection of Helmet Use in Motorcycle Drivers Using Convolutional Neural Network,” Appl. Sci., vol. 13, no. 10, Art. no. 10, Jan. 2023, doi: 10.3390/app13105882.

[6] N. Boonsirisumpun, W. Puarungroj, and P. Wairotchanaphuttha, “Automatic Detector for Bikers with no Helmet using Deep Learning,” in 2018 22nd International Computer Science and Engineering Conference (ICSEC), Nov. 2018, pp. 1–4. doi: 10.1109/ICSEC.2018.8712778.

[7] M. Dasgupta, O. Bandyopadhyay, and S. Chatterji, “Automated Helmet Detection for Multiple Motorcycle Riders using CNN,” in 2019 IEEE Conference on Information and Communication Technology, Dec. 2019, pp. 1–4. doi: 10.1109/CICT48419.2019.9066191.

[8] A. Singh, D. Singh, J. Singh, P. Singh, and D. A. Kaur, “Helmet & Number Plate Detection Using Deep Learning and Its Comparative Analysis.” Rochester, NY, Jun. 29, 2022. doi: 10.2139/ssrn.4149145.

[9] L. Shine and J. C. V., “Automated detection of helmet on motorcyclists from traffic surveillance videos: a comparative analysis using hand-crafted features and CNN,” Multimed. Tools Appl., vol. 79, no. 19–20, pp. 14179–14199, May 2020, doi: 10.1007/s11042-020-08627-w.

[10] H. Lin, J. D. Deng, D. Albers, and F. W. Siebert, “Helmet use detection of tracked motorcycles using CNN-based multi-task learning.,” IEEE Access, vol. 8, Sep. 2020, doi: 10.1109/access.2020.3021357.

[11] **C**. A. Rohith, S. A. Nair, P. S. Nair, S. Alphonsa, and N. P. John, “An Efficient Helmet Detection for MVD using Deep learning,” in 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Apr. 2019, pp. 282–286. doi: 10.1109/ICOEI.2019.8862543.

[12] R. Cheng, X. He, Z. Zheng, and Z. Wang, “Multi-Scale Safety Helmet Detection Based on SAS-YOLOv3-Tiny,” Appl. Sci., vol. 11, no. 8, Art. no. 8, Jan. 2021, doi: 10.3390/app11083652.

[13] S. Kanakaraj, “Real-time Motorcyclists Helmet Detection and Vehicle License Plate Extraction using Deep Learning Techniques”.

[14] S. Chen, J. Lan, H. Liu, C. Chen, and X. Wang, “Helmet Wearing Detection of Motorcycle Drivers Using Deep Learning Network with Residual Transformer-Spatial Attention,” Drones, vol. 6, no. 12, Art. no. 12, Dec. 2022, doi: 10.3390/drones6120415.

[15] F. W. Siebert and H. Lin, “Detecting motorcycle helmet use with deep learning,” Accid. Anal. Prev., vol. 134, p. 105319, Jan. 2020, doi: 10.1016/j.aap.2019.105319.