

Machine and Deep Learning Techniques for Daytime Fog Detection in Real Time with In-Vehicle Vision Systems Using the SHRP 2 Naturalistic Driving Study Data

Transportation Research Record

1–17

© National Academy of Sciences:

Transportation Research Board 2022

Article reuse guidelines:

sagepub.com/journals-permissions

DOI: 10.1177/03611981221103236

journals.sagepub.com/home/trr


Md Nasim Khan¹  and Mohamed M. Ahmed¹ 

Abstract

The main focus of this study is to develop a system that can accurately detect the presence of fog in real time at a trajectory level. This study leveraged video data from the SHRP 2 Naturalistic Driving Study (NDS). Extensive data reduction steps were taken to classify various levels of foggy weather conditions from the video data to form two unique image data sets. Afterward, features based on the gray level co-occurrence matrix (GLCM) were extracted from the images and used as classification parameters for training support vector machine (SVM) and K-nearest neighbor (K-NN) algorithms. In addition, a convolutional neural network (CNN) was also examined to improve the detection performance. Although the analysis was done initially on a data set consisting of two weather conditions, clear and fog, it has been extended to include different levels of fog, that is, near fog and distant fog. While the accuracy of the first analysis with two categories was approximately 92% and 91% for SVM and K-NN classifiers, respectively, the CNN produced much greater accuracy of 99%. As expected, the accuracy of the second analysis, with more refined weather categories, was relatively lower than the first analysis where CNN, SVM, and K-NN models produced an accuracy of about 98%, 89%, and 88%, respectively. With the rapid advances in connectivity and affordable cameras, the proposed detection models could be integrated into the smartphones of regular road users, creating an effective way to collect real-time road weather information that could be used to improve weather-based variable speed limit (VSL) systems.

Keywords

data and data science, artificial intelligence and advanced computing applications, artificial intelligence, deep learning, machine learning (artificial intelligence), machine vision, neural networks, support vector machines, pedestrians, bicycles, human factors, human factors of vehicles, driver behavior, naturalistic data studies

Driving in foggy weather is challenging given the reduced visibility, limited contrast, and distorted perception. From a visual perspective, fog can be described as a reduction in contrast in the visual field. In fog, drivers face difficulty in perceiving speed and headway as well as road signs and markings, all of which are crucial for safe driving. Fog-related crashes usually involve multiple vehicles and have more fatalities compared with crashes in clear weather conditions. Many pile-up crashes have occurred in recent years caused by the presence of fog which resulting in fatalities, injuries, and property damage. Crash statistics from 2007 to 2016 show that motor vehicle crashes in foggy weather conditions were responsible for 25,451 crashes resulting in a total of 8,902

injuries and 464 deaths each year in the United States (1). It is, therefore, important to carry out research on real-time fog detection and to provide appropriate strategies for meteorological management to improve traffic safety.

Weather detection has been the focus of extensive research in recent years given its numerous applications in the emerging field of connected and autonomous

¹Department of Civil & Architectural Engineering & Construction Management, University of Wyoming, Laramie, WY

Corresponding Author:

Md Nasim Khan, mkhan7@uwyo.edu

vehicles (CAV). Many studies have leveraged the data from Road Weather Information System (RWIS) or video feeds from stationary roadside cameras to develop weather detection systems. For instance, Jonsson used sensor information from RWISs coupled with camera images to develop a weather detection system capable of identifying six levels of weather conditions (2). The use of color and texture patterns for weather identification has been investigated by Lee et al. (3). By analyzing the color and texture of images captured using closed-circuit television (CCTV) cameras, their study developed a weather detection system with an impressive overall accuracy of 86%. In another study, He et al. proposed a simple but effective technique, called dark channel prior, to remove fog from images. Using this prior, they estimated the thickness of the fog and recovered a high-quality fog-free image (4). Asery et al. extracted features based on GLCM from images and used a support vector machine (SVM) classifier to detect fog. They used both synthetic and real images of foggy weather and reached about 97% and 85% accuracy, respectively (5). Anwar and Khosla (6) used average intensity and entropy as classification parameters and successfully classified fog into two groups: heterogeneous and homogeneous fog based on a synthetic image data set. The use of deep learning for developing trajectory-level fog detection systems was investigated by Khan and Ahmed. Their study experimented with various neural network-based deep learning models, including deep neural network (DNN), recurrent neural network (RNN), long short-term memory (LSTM), and CNN. They reported that CNN outperformed other deep learning models in detecting foggy weather conditions (7).

The current practice of collecting roadway visibility information is mainly based on weather stations, which is expensive, making their widespread implementation unfeasible. Moreover, the visibility distance from weather stations may not represent real-time weather conditions on the roadway for various reasons. For instance, the reporting frequency for most weather stations is relatively long. Furthermore, weather stations are location-specific and mostly mounted at higher elevations that may not necessarily represent roadway surface visibility. Trajectory-level weather detection systems at road surface level could overcome these challenges. Keeping these research needs in mind, this study proposes some unique techniques to enhance the reliability of real-time detection of road surface visibility from trajectory-level video feeds extracted from the Second Strategic Highway Research Program (SHRP 2) Naturalistic Driving Study (NDS) data set. The study used various machine learning techniques, including CNN which is a promising deep learning technique. The major contributions of this paper can be listed as follows:

- This study conducted extensive data processing steps to create two unique image data sets consisting of a total of 36,200 images. To the extent of the authors' knowledge, this study is one of the first attempts to create such comprehensive annotated image data sets using trajectory-level video feeds from the SHRP 2 NDS.
- This study applied a robust method, named the synthetic minority oversampling technique (SMOTE), to balance the data in the minority group (i.e., near fog).
- This study applied traditional machine learning as well as more advanced CNN deep learning models. In addition, several pre-trained CNN models, including AlexNet, GoogLeNet, ResNet18, and VGG16, were also leveraged to develop the best detection model.
- Unlike previous studies, this study performed sensitivity analyses to determine the optimum input image size and the optimum number of convolutional layers for CNN-based fog detection models.

Data Source

To develop the fog detection system, trajectory-level video data from the SHRP 2 NDS were used. SHRP 2 is the largest naturalistic study in the United States. Vehicles participating in the SHRP 2 NDS were instrumented with cameras that captured and stored videos of the roadways in both front and rear directions. More than 3,100 participants were recruited to collect naturalistic data which, at the end of the project, resulted in more than 33 million travel miles from 3,800 vehicle years of driving. Note that this unique study was carried out in six geographically diverse states (Florida, Indiana, New York, North Carolina, Pennsylvania, and Washington) between 2010 and 2013 (8, 9).

Data Preparation

To effectively extract video data of trips occurring in foggy weather conditions from the huge SHRP 2 NDS data set, two unique methods were developed. In the first method, weather data from the National Climate Data Center (NCDC) were used. The weather stations were considered as points of interest with an influence zone of 5 nmi around them to identify the potential locations of trips that occurred in adverse weather. The second method used weather-related crash locations as points of interest to identify adverse weather trips. More details about this method can be found in (10, 11). By using these methods, video data were received from the Virginia Tech Transportation Institute (VTTI) for trips

that occurred in foggy weather and their corresponding matched trips in clear weather. Subsequently, all trips were manually verified to eliminate erroneous and unusable data which resulted in a total of 217 trips in foggy weather and 430 corresponding match trips in clear weather.

A new data set of images was created from the video data by extracting images from the videos at 12 frames-a-minute sampling rate. Subsequently, all the images were cropped at the bottom to remove the dashboard and to maintain consistency among the images. Based on visibility, the image data set was then annotated manually and grouped into two categories: clear and fog, as illustrated in Figure 1. A sample of 8,500 images from each category was selected for analysis, which resulted in 17,000 images. During the annotation process, the research team noticed that some images extracted from the clear video data have sun glare and were thus eliminated from the clear image group.

The analysis was then extended to add different levels of fog to investigate the performance of the models with a higher number of output categories. Images in the second data set were classified into three groups: clear, distant fog, and near fog. It is worth mentioning that the classification of fog is not consistent in the literature.

The National Oceanic and Atmospheric Administration (NOAA) classified fog into two categories (12): “near” if the visibility distance falls below 0.25 mi; and “light” if the visibility distance is between 0.3 mi and 6 mi. The South Carolina Department of Transportation (DOT) developed a low visibility warning system, defining fog as “dense” if the visibility falls below 300 ft and “light” if the visibility ranges between 300 ft and 900 ft (13). However, for this study, the fog was classified into two categories, near fog and distant fog, using qualitative-based measures extracted from the NDS videos. The fog was classified based on the visibility of road markings, readability of road signs, roadside surroundings (delineators, guardrails, jersey barriers, etc.), and the horizon. The fog was reported as near fog during manual image annotation if the following conditions were observed:

- Few road markings in front of the NDS vehicle could be observed.
- Information on the road signs could not be read.
- Roadside surroundings and traffic ahead could not be clearly recognized.
- The horizon were undefinable.

By contrast, the fog was classified as distant fog if:



Figure 1. Sample images of foggy and matched clear weather.



Figure 2. Sample image of near fog: (a) only one road marking is visible, the sign is unreadable, surroundings and the horizon cannot be seen properly; and (b) few road markings are visible, surroundings, traffic, and the horizon cannot be recognized properly.



Figure 3. Sample image of distant fog: (a) road markings are visible, signs are readable, surroundings and traffic can be seen to some extent, the horizon cannot be seen clearly; and (b) road markings are visible, the speed limit sign is readable, surroundings and traffic can be seen, the horizon cannot be recognized properly.

Table I. Summary Statistics of the Image Data Sets

	Weather	Number of images	Equivalent video duration (min)
Image data set 1	Clear	8,500	708.33
	Fog	8,500	708.33
	Total	17,700	1,416.67
Image data set 2	Clear	8,500	708.33
	Distant fog	8,500	708.33
	Near fog	1,500	125
	Total	18,500	1,541.67

- Road markings and information on road signs could be easily recognized.
- Roadside surroundings and traffic ahead were visible.
- The horizon were undefinable.

Sample images of near fog and distant fog are shown in Figures 2 and 3, respectively.

The summary of the data extracted in this study is shown in Table 1. For the second image data set, 8,500 images from clear weather, 8,500 from distant fog, and

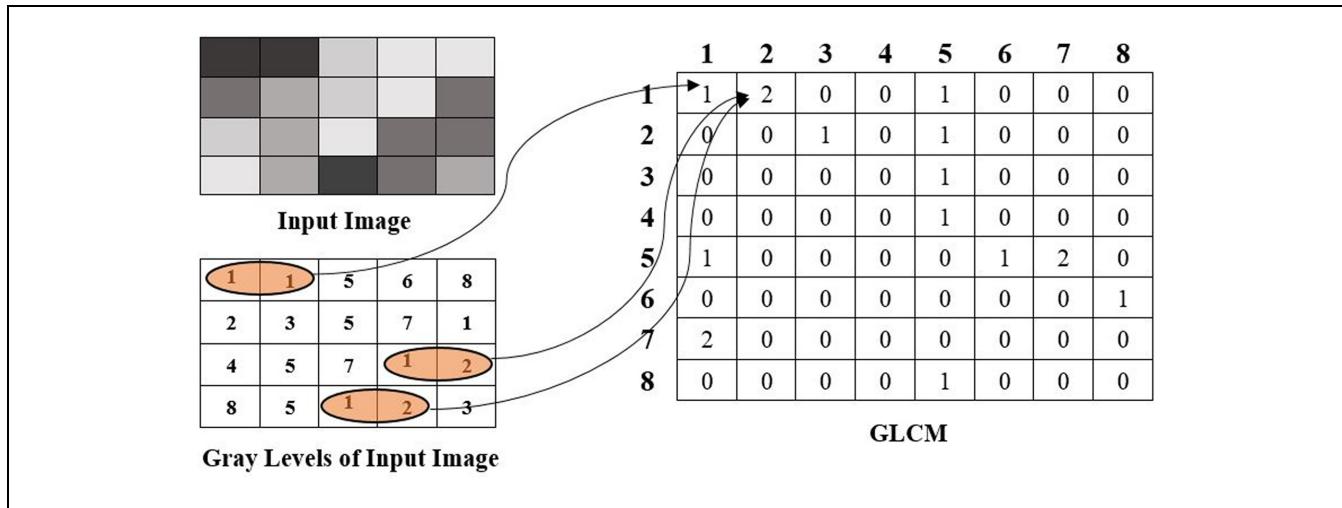


Figure 4. Example of the gray level co-occurrence matrix (GLCM).

1,500 images from near fog were extracted. As near fog is a relatively rare environment condition, there were fewer images of near fog compared with other categories, making the data set moderately imbalanced. In machine learning, a data set is defined as imbalanced if the distribution of the classification categories is not approximately uniform (14). Categories that make up a smaller proportion compared with other categories in the data set are called minority classes. Google developers devide the degree of data imbalance into three categories: mild with 20% to 40% minority class, moderate with 1% to 20% minority class, and extreme with less than 1% minority class (15). The problem of data imbalance was addressed in this study by leveraging SMOTE which is a robust method and one of the well-accepted data balancing techniques in the field of machine learning and data mining (16).

Methodology

To determine fog from the video data, two methodologies were adopted. The first methodology used the traditional machine learning technique, which includes the extraction of texture features from the image data sets, followed by the training of the extracted texture feature using different classifiers. Finally, the accuracy of the trained model was tested using a new test data set. For the second methodology, deep learning techniques were used for training the fog detection models and, similar to the previous method, the prediction accuracy of the trained models was tested using a test data set. For both methods, 70% of the data were used for training and validation and 30% of the data were used for testing the accuracy of the trained models. It is worth mentioning that the training and testing images were not correlated

since they are selected randomly. The following section describes the details of the methodologies used in this study.

Feature Extraction

This study used GLCM-based image features as learning parameters for developing the machine learning models. GLCM is one of the simplest approaches for extracting texture features from images. It is defined as a second-order histogram statistic from a gray image and was defined by Haralick et al. (17). It represents the frequency of occurrence of a pixel value $p(i,j)$ in an image with its surrounding pixels. Figure 4 illustrates the concept of GLCM. Element (1, 1) contains the value 1 because the input image has only one instance where two pixels with value 1 occurred horizontally. Similarly, the element (1, 2) holds the value 2 because there are two instances where horizontally adjacent pixels have values 1 and 2 (18).

Most of the studies based on in-vehicle cameras or sensors require the presence of a consistent object in front of the vehicle. For instance, a weather detection method proposed in (19) requires road making, shoulder boundaries, and tracks left by other vehicles to perform accurately. The fog detection system described in Gallen et al. (20) requires a distinct object in the image. Some studies also use road surface (21), the horizon (22, 23), or road edge lines (24) to develop a weather detection system. These detection methods might not be reliable in everyday scenarios because arbitrary objects can be obstructed by other vehicles, especially in congested traffic conditions. However, the proposed weather detection model in this study uses global features of the images instead of object detection so that no arbitrary object is required. It is true that in congested traffic conditions,

the contrast increases in the images of both clear and fog. However, the increase of contrast is not equal, since a major portion of the images consists of the sky and the roadside surroundings, which are not usually obstructed in the presence of congested traffic. This study used texture-based image features using GLCM, which represents the change in gray levels of an image. In clear weather, it is more likely to have a sudden change in gray levels, whereas, in foggy weather, more uniform gray levels are expected. This property has been captured using four GLCM features: contrast, correlation, homogeneity, and energy. However, the system is not foolproof. There are a few cases when the system might not work properly. For instance, if the windshield of the vehicle is not clear enough, it might provide a false representation of foggy weather.

Another important factor to be considered is the resolution of the video camera. The system is developed based on the SHRP 2 video data, which was captured using cameras with a resolution of 480×356 pixels and a maximum frame rate of only 14 frames per second. Considering the rapid advances in affordable cameras, most in-vehicle camera systems have a resolution greater than the one used in this study.

Machine Learning Algorithms

Machine learning is a field of engineering where computers can learn and improve from experience without depending on rules-based programming. In other words, machine learning is an array of techniques for revealing uncovered patterns in data that can be used to predict new instances (25). Note that many previous studies have used machine learning in the field of transportation engineering aiming to improve the safety of the roadways (26–30). El Naqa (31) defined machine learning as the science of getting computers to attain a particular task without rules-based programming to generate the desired outcome. Samuel (32) defined machine learning as a technique that can give computers the ability to learn without being programmed.

As mentioned earlier, this study applied GLCM to extract the features from the images and subsequently leveraged the extracted features to train two machine learning algorithms, including SVM and K-nearest neighbor (K-NN). In addition, deep learning techniques have been used to develop the fog detection system.

Support Vector Machine (SVM). An SVM is a classifier based on an optimal hyperplane in a high or infinite-dimensional space. SVM classifies data by finding the best hyperplane that separates data points into two or more classes. The hyperplane that has the largest margin between classes is considered the best hyperplane. The

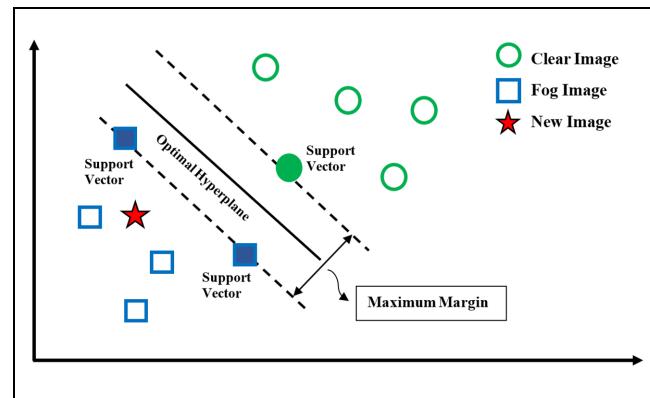


Figure 5. Support vector machine (SVM) classification.

margin represents the distance between the hyperplane and the nearest data point (support vector) from either side of the hyperplane (33). Figure 5 illustrates this definition where the green circles and the blue squares represent clear and fog images, respectively. The solid circles/squares represent support vectors, which define the width of the margin. The boundaries on the support vector (depicted as dotted lines) separate the data points into two distinct groups. As the new test image (red star) falls on the lower side of the optimum hyperplane, it will be assigned as a fog image.

K-Nearest Neighbor (K-NN). The K-NN classifier is a memory-based model that stores all available cases and classifies new cases based on their distance to points in a training data set. One of the most important factors that can influence the prediction accuracy of a K-NN classifier is k , which indicates the number of data points to be considered around the test sample. A small value of k will result in a large variance in predictions. Conversely, a large value of k may increase model bias. Therefore, k should be selected in such a way that the model achieves the right balance between the variance and bias of the model (34). The concept of the K-NN classifier is illustrated in Figure 6; the green circles represent clear images, and the blue squares represent fog images. The new test image (red star) needs to be classified either as a clear or fog image. If $k = 3$, it will be assigned as a fog image since the majority of the points inside this circle are fog images. Similarly, if $k = 7$, then it will be assigned as a clear image.

Deep Learning. Deep learning is a branch of supervised machine learning which is concerned with algorithms inspired by the structure of the human brain. Deep learning is based on a neural network architecture, where the term “deep” indicates many layers in the network. A deep network can represent a complex function by

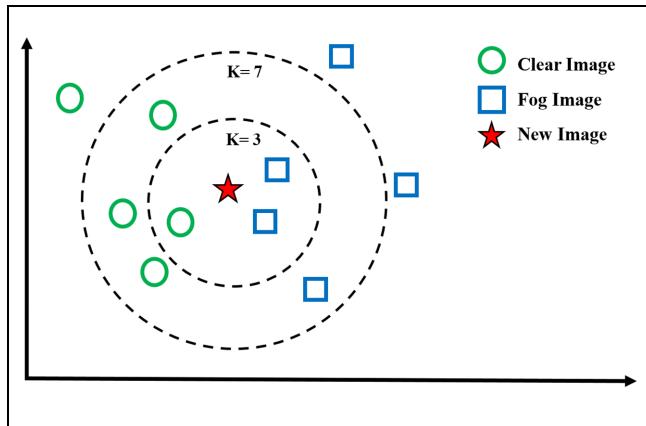


Figure 6. K-nearest neighbor (K-NN) classification.

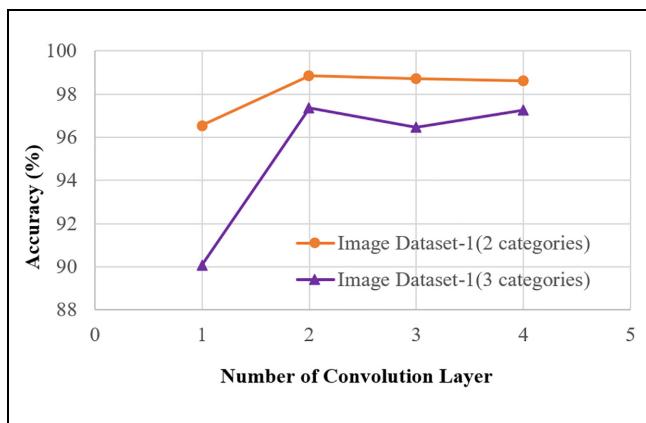


Figure 7. Selection of the number of convolution layers.

adding more layers and more units within a layer and can perform tasks related to mapping an input vector to an output vector with impressive accuracy, given a sufficiently large amount of annotated training data sets (35). In recent years, deep learning has emerged as a cutting-edge technique in the field of transportation engineering, especially in connected and autonomous vehicle-related research. Many researchers have been using this robust method to develop systems aiming to improve the safety and operation of roadways (36–39). To develop the fog detection system, CNN, which is one of the most popular algorithms for deep learning with images, was used.

The proposed CNN model consisted of an input layer, an output layer, and 10 layers in-between. The in-between layers can be categorized into two types of layer: feature detection layers and classification layers. The feature detection layer can perform three types of operation in the data, including convolution, pooling, or rectified linear unit (ReLU). Convolution operation can activate certain features from the images by passing them through a set of convolutional filters. To determine the optimum

number of convolution layers, the detection accuracy was investigated with the increase of convolution layers. No significant improvement in the accuracy was found after three convolution layers for both image data sets. The accuracy of the CNN model for image data set 1, consisting of two output categories, is saturated at around 98.5% after three convolution layers. The model accuracy for image data set 2, consisting of three categories, is also saturated after three convolutional layers, as can be seen in Figure 7.

The first convolutional layer took the images as input and applied 16 filters, each with a height and width of 3 pixels. The second and the third convolutional layer applied 32 and 64 filters of the same size, respectively. The number of filters of the convolutional layers was chosen as powers of 2 to maximize the usage of the graphics processing unit (GPU). In-between each convolution layer, two max-pooling layers were applied with pooling regions of 2 pixels \times 2 pixels. Pooling simplifies the output by performing nonlinear down-sampling, which reduces the number of parameters that the network needs to learn. Finally, a ReLU layer was used to perform a threshold operation on each element of the input. ReLU layer maps negative values to 0 to ensure faster and more accurate training.

After feature extraction, the architecture of the CNN shifted to classification. The next layer is a fully connected (FC) layer that outputs a vector of K dimensions, where K is the number of classes that the network will be able to predict. For the image data set 1 and image data set 2, two and three classes were used, respectively. Finally, the image data set was passed into a Softmax layer, which is the final layer of the CNN model. The architecture of the proposed CNN is illustrated in Figure 8 and the characteristics of different layers with their size parameters are listed in Table 2. Other researchers can use this information effectively via transfer learning and can apply the proposed CNN architecture to their data to develop weather detection models. Note that weights and biases of any CNN need to be optimized at each iteration during training and validation, and therefore these parameters are considered learnable parameters.

To investigate the effect of input image size on model performance and computational cost, a sensitivity analysis was conducted, in which the accuracy and training time of the CNN models were observed with a 25 pixels \times 25 pixels increment at each iteration. Considering image data set 1, the accuracy and the training time of the CNN model with an image size of 25 pixels \times 25 pixels were found to be 98.5% and 2.3 min, respectively, as illustrated in Figure 9a. The accuracy of the models was improved up to an image size of 75 pixels \times 75 pixels at the expense of more computational cost; after that, no

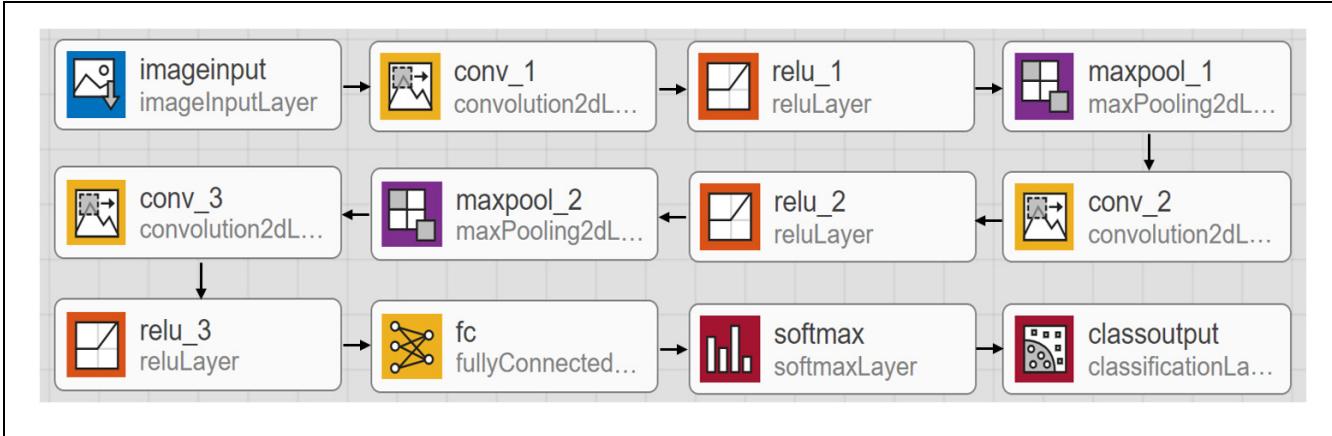


Figure 8. The architecture of the proposed CNN model.

Note: CNN = convolutional neural network; conv = convolutional layer; fc = fully connected layer.

Table 2. Layer Characteristics with Size Parameters of the Proposed CNN Models

Type	Data set	Description	Activations	Learnable parameters	Total learnable parameters
Image input	1 and 2	100×100×3 images with 'zero-centered' normalization	100×100×3	na	0
First convolution layer	1 and 2	16, 3×3 convolutions with stride [1 1] and padding [1 1 1]	100×100×16	W=3×3×3×16 B=1×1×16	448
First ReLU	1 and 2	ReLU	100×100×16	na	0
First max pooling	1 and 2	2×2 max pooling with stride [2 2] and padding [0 0 0 0]	50×50×16	na	0
Second convolution layer	1 and 2	32, 3×3 convolutions with stride [1 1] and padding [1 1 1]	50×50×32	W=3×3×16×32 B=1×1×32	4,640
Second ReLU	1 and 2	ReLU	50×50×32	na	0
Second max pooling	1 and 2	2×2 max pooling with stride [2 2] and padding [0 0 0 0]	25×25×32	na	0
Third convolution layer	1 and 2	64, 3×3 convolutions with stride [1 1] and padding [1 1 1]	25×25×64	W=3×3×32×64 B=1×1×64	18,496
Third ReLU	1 and 2	ReLU	25×25×64	na	0
Fully connected	1	2 fully connected layers	1×1×2	W=2×40,000 B=2×1	80,002
	2	3 fully connected layers	1×1×3	W=3×40,000 B=3×1	120,003
Softmax	1	Softmax layer	1×1×2	na	0
	2	Softmax layer	1×1×3	na	0
Output	1	Classification output	1×1×2	na	0
	2	Classification output	1×1×3	na	0

Note: CNN = convolutional neural network; W = weight; B = Bias; na = not applicable.

significant improvement in prediction performance was observed. The time required to train this optimum CNN model was found to be around 4 min. A similar trend was also observed for image data set 2 where an input

image size of 100 pixels × 100 pixels produced the best performance. The accuracy and training time for this CNN model were 98.2% and 7.4 min, respectively, as shown in Figure 9b.

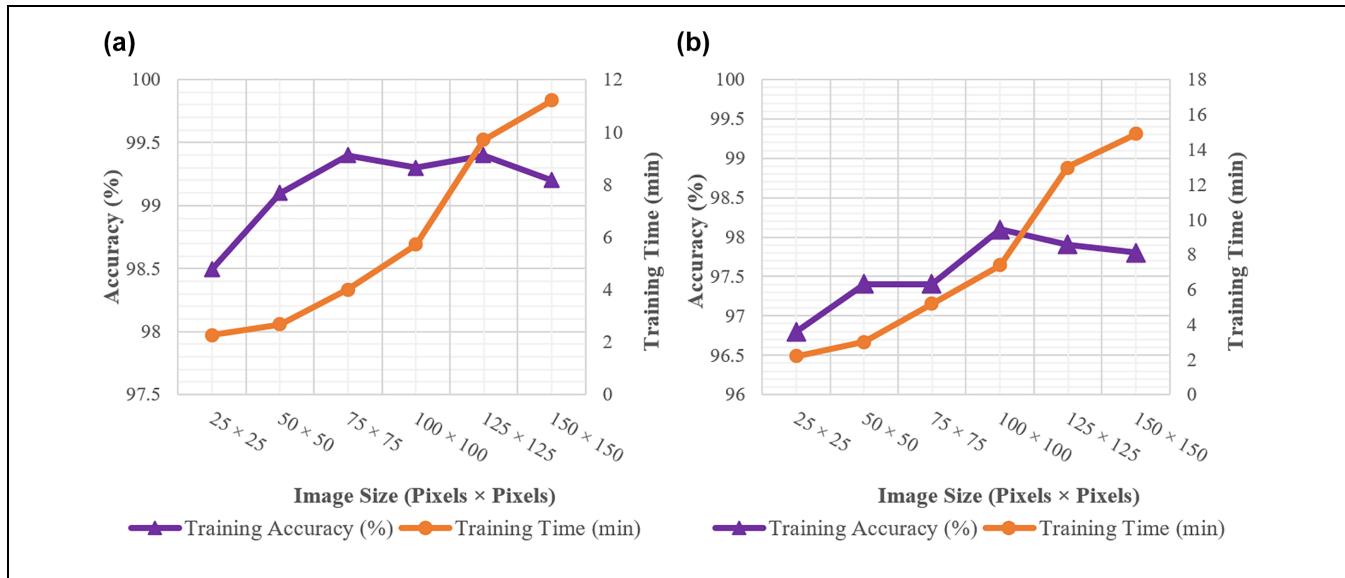


Figure 9. Effect of input image size on accuracy and training time: (a) image data set 1; and (b) image data set 2.

Table 3. Tuning of the Hyperparameters

Parameters/training options	Initial/default value	Final value	
		Image data set 1	Image data set 2
Optimizer	SGDM	SGDM	SGDM
Number of convolution layers	5	3	3
Max. epochs	30	6	10
Batch size	128	100	50
Initial learning rate	0.01	0.0001	0.0001
Learning rate drop period	10	8	8
Learning rate drop factor	0.01	0.1	0.1
Factor for L2 regularization	0.0001	0.004	0.004

Note: Max. = maximum; SGDM = stochastic gradient descent with momentum.

In an attempt to find the best fog detection model, the initial hyperparameters of the CNN models have been updated by carefully observing the training progress and validation results for different parameters. Table 3 lists the updated parameters for the developed CNN models.

The overall methodology used in this study to identify fog using machine and deep learning techniques is illustrated in Figure 10.

Results and Discussions

Preliminary Investigation of the Extracted Features

The GLCM features, including contrast, correlation, energy, and homogeneity, have been extracted from the images in a MATLAB® environment. Before the training of the models, boxplots were made to investigate the overall patterns of the GLCM features extracted from

the image data sets. The boxplots of GLCM features extracted from image data set 1 indicate that the overall patterns and shapes of each category were different, as can be seen in Figure 11. For instance, the range of contrast values of the clear image group varied between 0.02 and 0.55 with a mean value of 0.13. Conversely, the fog image group had a much narrower range between 0.04 and 0.26, with a mean value of 0.05. Considering correlation values, the clear image group had a wider range and lower mean compared with the fog image group. The boxplots of the other two GLCM texture features (e.g., homogeneity and energy) also exhibited similar kinds of variation in shapes and patterns.

The box plots of GLCM features extracted from image data set 2 also indicate that the level of visibility can be separated into different categories as the mean, range, and quartile values of each category are different

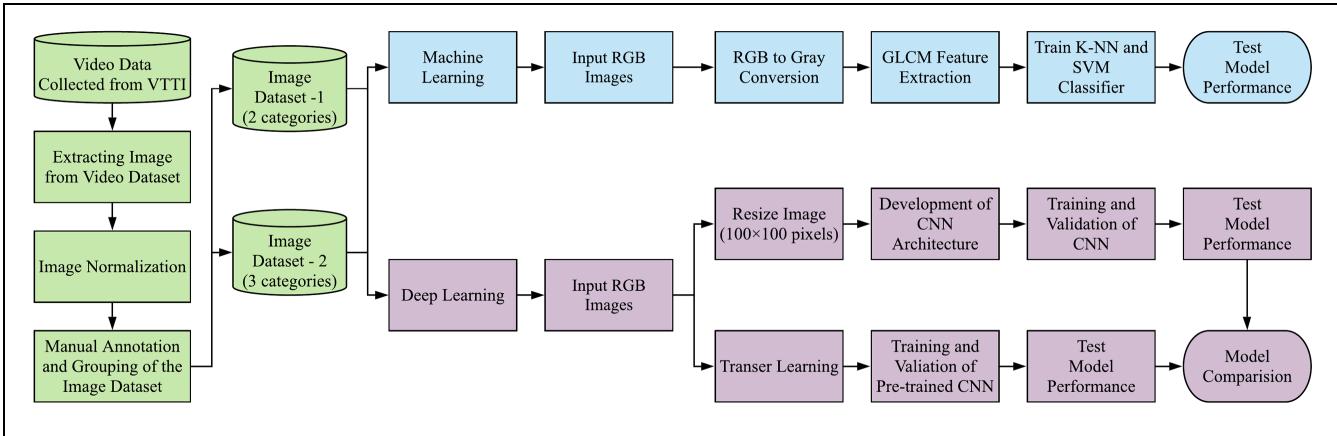


Figure 10. Flowchart summarizing machine and deep learning techniques for fog detection.

Note: VTTI = Virginia Tech Transportation Institute; RGB = red, green, and blue; GLCM = gray level co-occurrence matrix; K-NN = K-nearest neighbor; SVM = support vector machine; CNN = convolutional neural network.

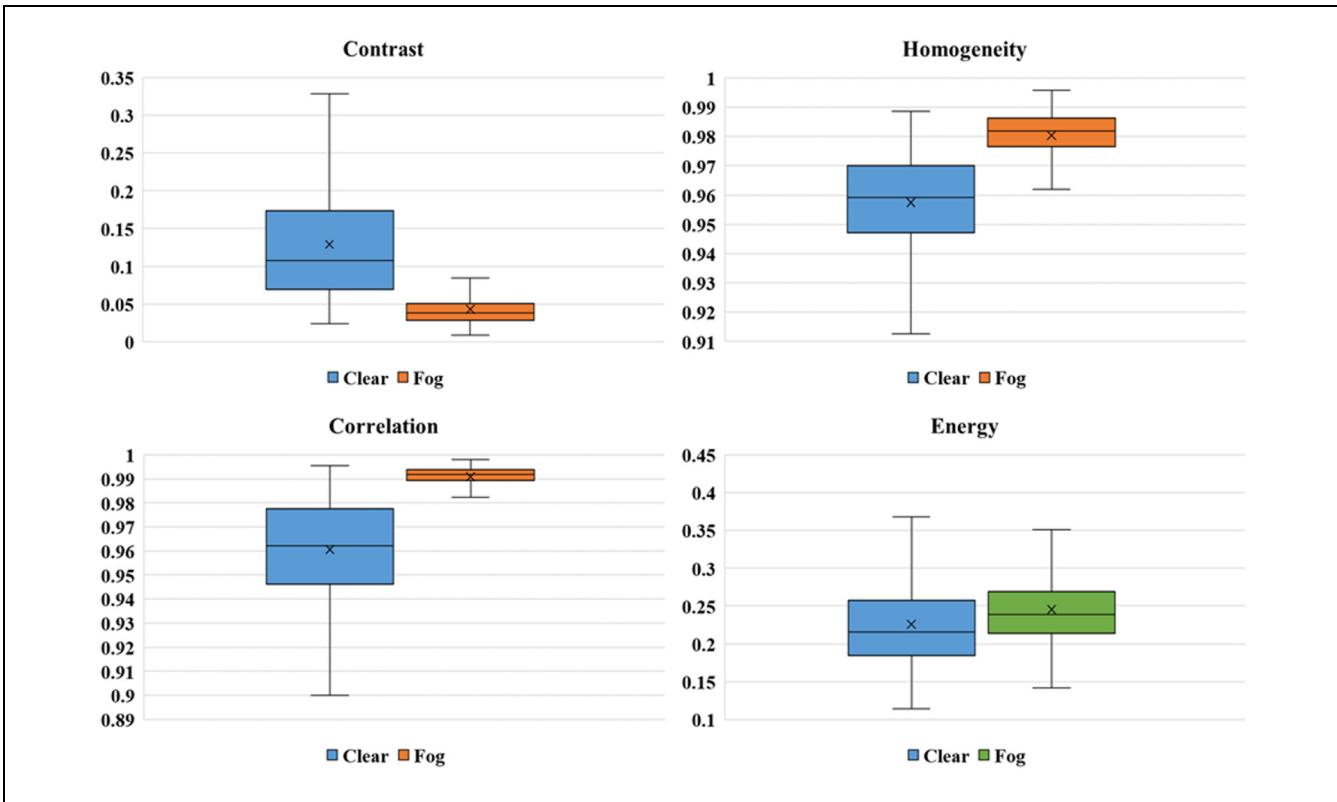


Figure 11. Boxplots of gray level co-occurrence matrix (GLCM) features of image data set 1.

from each other. The mean contrast value of the near fog image group was found to be 0.02, which was the lowest among the categories. The range of contrast values of the near fog image group was also narrower compared with the other image groups. Considering the correlation value, the near fog image group had a much higher mean and narrower range compared with the

other groups. Similar results were also found for homogeneity and energy values as shown in Figure 12.

Overall, from the boxplots of image data set 1 and image data set 2, it can be concluded that the GLCM features (i.e., contrast, correlation, energy, and homogeneity) can be used as significant classification parameters for training the SVM and K-NN classifiers.

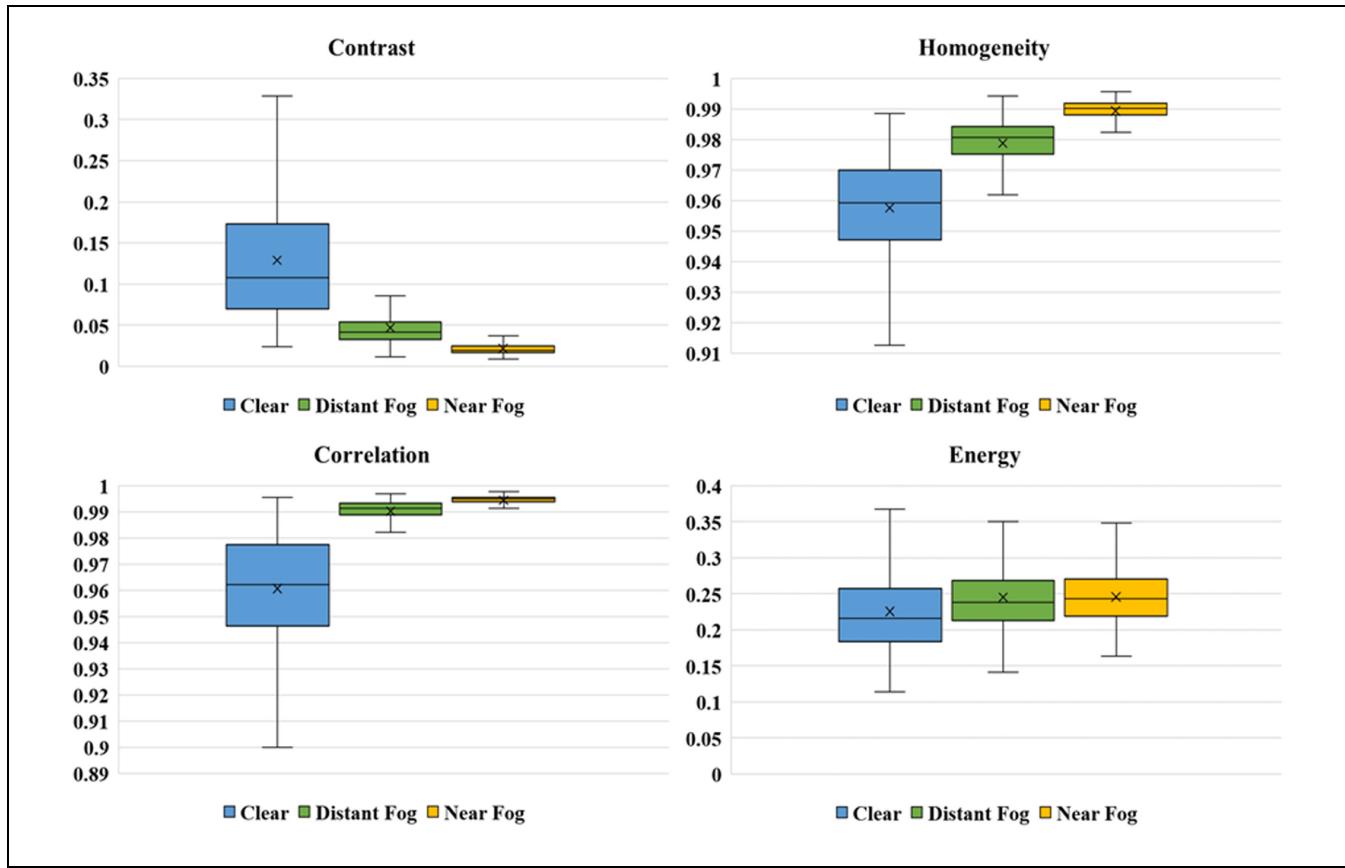


Figure 12. Boxplots of gray level co-occurrence matrix (GLCM) features of image data set 2.

True Class	Predicted Class	
	Clear	Fog
Clear	89%	11%
Fog	6%	94%

Figure 13. Confusion matrix of the trained support vector machine (SVM) model for image data set 1.

Performance of the Detection Models

The prediction capability of the trained SVM model for the image data set 1 is shown in Figure 13 using a confusion matrix. Although various types of kernel functions, including linear, quadratic, cubic, fine Gaussian, and coarse Gaussian, were used for the SVM-based classification, the fine Gaussian SVM produced the best result with an accuracy of 91.5%. It is worth noting that SVM performs poorly with data that are not linearly

separable. But, the kernel function can transform the data, which is not linearly separable, into higher-dimensional data where they become linearly separable (40).

The fog image group had the highest true-positive rate (94%) and the lowest false-negative rate (6%), which indicates that the trained SVM model misclassified only 6% of the fog images. On the other hand, the true-positive rate of the clear image group was about 89%, meaning that 89% of the clear images have been classified correctly. In addition, the false-positive rate for the clear and fog image group was found to be 6% and 10%, respectively, as can be seen in Figure 13.

Figure 14 shows the receiver operating characteristic curve (ROC) of different groups of the trained SVM model for image data set 1. The area under curve (AUC) of a ROC plot is a measure of the overall quality of a classifier. A perfect result with no misclassified points produces a right angle to the top left of the plot. On the other hand, a poor result that is no better than random produces a line at 45°. AUC values of 0.5 to 0.7 represent poor accuracy, values between 0.7 and 0.9 represent moderate accuracy, whereas values over 0.9 indicate high accuracy (41). All the categories of the trained SVM

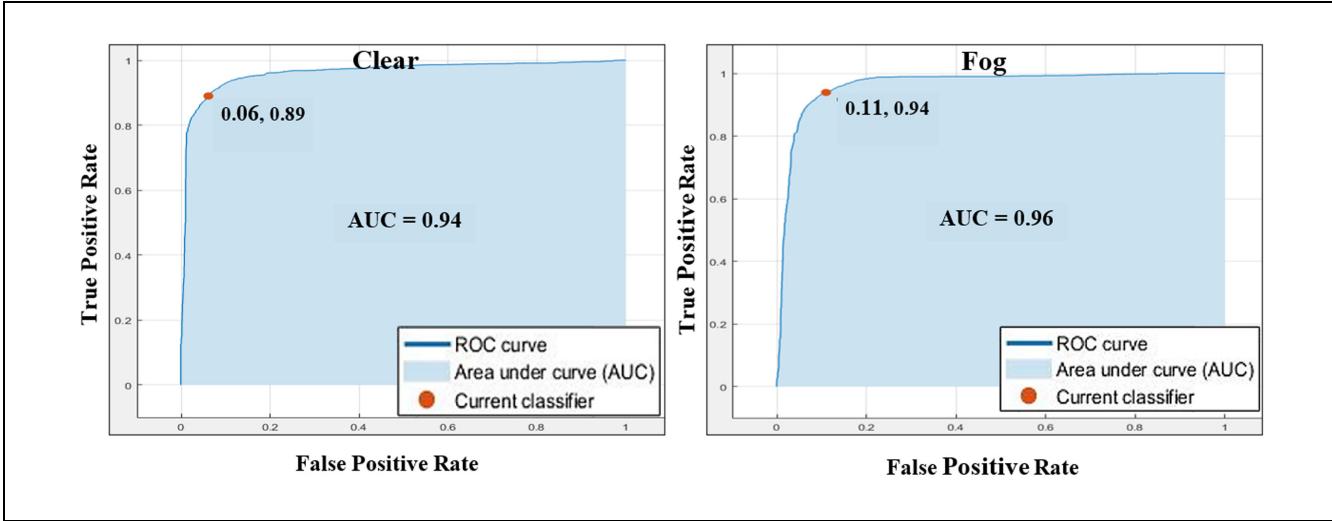


Figure 14. ROC curves of the trained SVM model for image data set 1.

Note: ROC = receiver operating characteristic curve; AUC = area under curve; SVM = support vector machine.

True Class			Predicted Class
	Clear	Fog	
Clear	89%	11%	
Fog	7%	93%	

Figure 15. Confusion matrix of the trained K-nearest neighbor (K-NN) model for image data set 1.

True Class				Predicted Class
	Clear	Distant Fog	Near Fog	
Clear	89%	10%	1%	
Distant Fog	6%	92%	2%	
Near Fog	1%	24%	75%	

Figure 16. Confusion matrix of the trained support vector machine (SVM) model for image data set 2.

models had AUC values greater than 0.9, which indicated high prediction accuracy.

The overall prediction accuracy of the K-NN model for image data set 1 was found to be about 91%. Similar

to the SVM model, the highest true-positive rate was found for the fog image group, where 93% of the images were correctly classified. On the contrary, the clear image group had the lowest true-positive rate (89%) and the highest false-negative rate (11%), which indicated that the trained K-NN model misclassified 11% of the clear images, as can be seen in Figure 15.

As mentioned earlier, a separate analysis was conducted to determine the performance of the SVM and K-NN model on image data set 2, which consists of three categories, including clear, distant fog, and near fog. As expected, the accuracy of the SVM and K-NN models was reduced to 88.6% and 88%, respectively. The prediction accuracy of the trained SVM model is shown in Figure 16, which indicates that the distant fog image group had the highest positive prediction rate, where 92% of the images were classified correctly. On the other hand, the near fog image group had the lowest prediction accuracy (75%), where about 24% and 1% of the images were misclassified as distant fog and clear images, respectively. In addition, the false-negative rate of the clear image group was found to be 89%, where 10% and 1% of the clear images were wrongly classified as distant fog and near fog, respectively.

Figure 17 shows the ROC curves of different image groups from the trained SVM model for image data set 2. The results indicate that the clear image group had the highest AUC (0.97), whereas the distant fog image group had the lowest AUC (0.94). However, the AUC of all the image groups was found to be greater than 0.9, which indicates good prediction accuracy of the trained models.

The confusion matrix of the trained K-NN model for image data set 2 is shown in Figure 18, which indicates that about 89%, 90%, and 74% of the clear, distant fog,

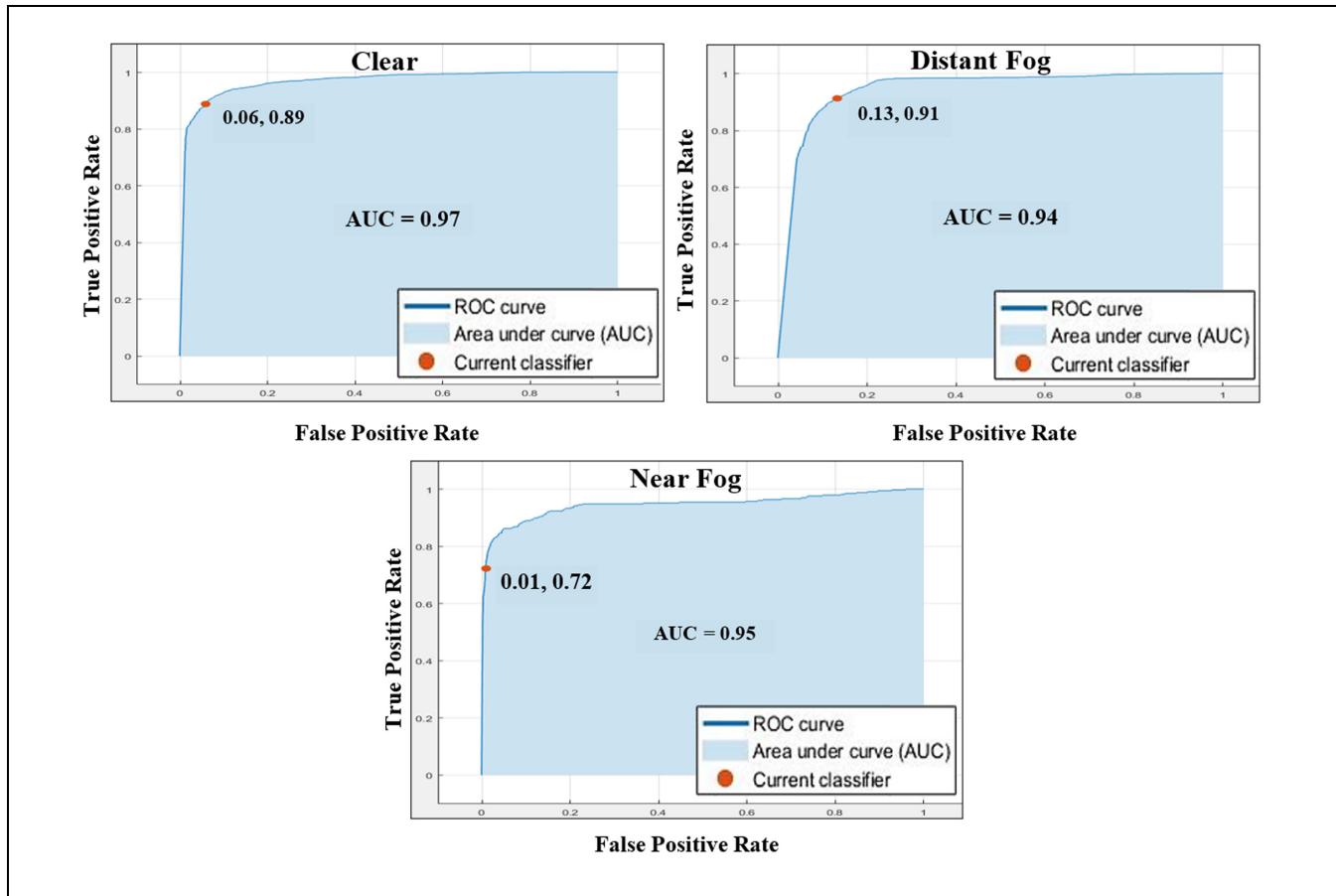


Figure 17. ROC curves of the trained SVM model for image data set 2.

Note: ROC = receiver operating characteristic curve; AUC = area under curve; SVM = support vector machine.

True Class	Predicted Class		
	Clear	Distant Fog	Near Fog
Clear	89%	10%	1%
Distant Fog	7%	90%	3%
Near Fog	2%	24%	74%

Figure 18. Confusion matrix of the trained K-nearest neighbor (K-NN) model for image data set 2.

and near fog images, respectively, have been correctly classified. The highest false-negative rate (26%) was found for the near fog image group, where 24% and 2% of near fog images have been wrongly identified as distant fog and clear images, respectively. On the other hand, the false-negative rate of the distant fog image group was found to be the lowest, where the trained

K-NN model misclassified 7% and 3% of the distant fog images as clear and near fog images, respectively. The highest false-positive rate was found for the near fog image group, where 21% of other images were classified as near fog. The false-positive rate for the clear and distant fog image group was found to be 7% and 16%, respectively.

As mentioned earlier, in addition to the machine learning techniques, the fog was also detected using the CNN deep learning technique. The CNN models significantly outperformed the SVM and the K-NN model. Considering image data set 1, which has two output categories, the overall accuracy of the CNN model was found to be 99.2%. The trained CNN model based on image data set 2 also provided a reliable prediction with an overall accuracy of 97.5%. It is worth noting that other neural network models, including RNN, were also investigated. However, the CNN performed far better than the other neural network models, and therefore only the CNN results have been reported in this study. Figure 19a shows the confusion matrices of the trained CNN model for image data set 1. The highest true-

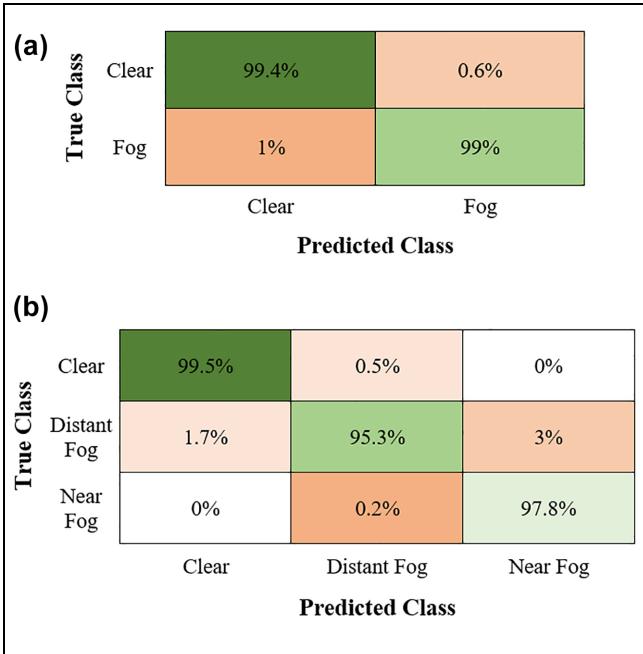


Figure 19. Confusion matrices of the trained CNN models:

(a) image data set 1; and (b) image data set 2.

Note: CNN = convolutional neural network.

positive rate (99.4%) was found for the clear image group, which indicates that about 99.4% of clear images were correctly classified. Conversely, the lowest true-positive rate was found for the fog image group, where 99% of the images were assigned correctly. Similarly, Figure 19b shows the confusion matrix of the trained CNN for image data set 2. While the clear image group had the highest (99.5%) true-positive rate, the near fog group had the lowest true-positive rate (97.8%).

Comparison With the Pre-Trained CNN Models

The performance of the proposed CNN models was compared with some pre-trained CNNs, including AlexNet, GoogLeNet, ResNet18, and VGG16. To use

the pre-trained models, transfer learning techniques were applied, where several layers of the pre-trained networks were updated and tuned to achieve the fog detection task. Most of the pre-trained networks have lots of layers with complex structures and require a relatively large input image size. The input image size of the pre-trained CNNs is predetermined. For instance, the input image size required to apply AlexNet is 227 pixels × 227 pixels. By contrast, the proposed CNN models required a smaller input image size that significantly reduced the training time without affecting the prediction performance. After a sensitivity analysis, this study suggested an input image size of 75 pixels × 75 pixels for image data set 1 and 100 pixels × 100 pixels for image data set 2. Table 4 shows the comparison between the pre-trained models with the proposed CNN models. Although the performance of the pre-trained models is marginally higher than the proposed CNN models, the training time of the proposed model is significantly lower than the pre-trained models. Note that the computer used for this study had an Intel Core i7-7500U 2.70Ghz processor, 12 GB of RAM, and an NVIDIA GeForce 940MX GPU. As training times can vary based on the processing power of the computer, a normalized index, named relative training time, was used to compare the results. Relative training time is determined by dividing the training time of the pre-trained models by the training time of the proposed CNN. The training times of AlexNet, GoogLeNet, ResNet18, and VGG16 were found to be about 3.7, 12.4, 10.9, and 22.1 times higher than the proposed CNN model with three categories (image data set 2). The overall accuracy for this model was found to be around 97.5% which was marginally lower compared with the pre-trained CNN models, as listed in Table 4.

As mentioned earlier, the main focus of these models was to provide motorists with a reliable prediction of the weather conditions through dynamic message signs (DMS) and variable speed limits (VSL). It is worth mentioning that the high false-positive rate of foggy weather will create frequent false alarms, which may lead to disrespect for these systems and may create compliance issues.

Table 4. Comparison of the Proposed Convolutional Neural Network (CNN) Models with Pre-Trained CNN Models

Name	Number of layers	Input image size (pixels × pixels)		Accuracy (%)		Training time (min)		Relative training time	
		Image data set 1	Image data set 2	Image data set 1	Image data set 2	Image data set 1	Image data set 2	Image data set 1	Image data set 2
Proposed CNN	4	75 × 75	100 × 100	99.2	97.5	4.0	7.4	1.0	1.0
AlexNet	8	227 × 227		99.4	98.2	27.0	27.4	6.8	3.7
GoogLeNet	22	224 × 224		99.5	97.9	90.8	91.7	22.7	12.4
ResNet18	18	224 × 224		99.8	98.6	81.8	80.7	20.5	10.9
VGG16	16	224 × 224		99.7	98.5	162.5	163.3	40.6	22.1

However, from a safety perspective, the false-negative rate is more hazardous because it will lead the drivers to foggy weather without providing them any prior warnings. As the calibrated CNN model has the lowest false-positive and false-negative rate with maximum prediction accuracy, integrating this model into the VSL algorithm will significantly improve its reliability.

Conclusions

The main focus of this study was to develop methodologies to identify fog from a single in-vehicle camera using the SHRP 2 NDS data. In this study, machine and deep learning techniques were used. First, data sets of images were created from the video data, which were then grouped into different categories based on weather and visibility conditions. Initially, the analysis was conducted on two groups: clear and fog. While the prediction accuracy of this analysis was 92% and 91% for SVM and K-NN classifiers, CNN deep learning techniques produced a superior result with an accuracy of about 99%. Subsequently, the same techniques were applied to a data set consisting of three groups: clear, near fog, and distant fog. As expected, the prediction accuracy of the second analysis, with more refined weather categories, was lower compared with the first analysis where CNN, SVM, and K-NN models produced an accuracy of about 98%, 89%, and 88% respectively.

The system developed in this study has the potential to provide cost-effective advanced driver assistance systems (ADAS) with multiple functions on a single hardware platform without including lots of expensive sensors. Several ADASs are currently being used to improve roadway safety. Although some of the well-known vehicle manufacturers are including these features in their newer models, aftermarket installation for older cars is expensive. Extensive deployment and widespread use of these technologies might not, therefore, be possible in the near future. However, the fog detection system proposed in this study is affordable and only requires a single video camera, giving it the potential to become a cost-effective solution for collecting trajectory-level weather information in real time.

Many Departments of Transportation (DOTs), including that in Wyoming, recently implemented an innovative road condition monitoring system using tablets mounted in maintenance vehicles. The monitoring system installed in maintenance vehicles requires drivers to report weather conditions manually by tapping nine codes on the tablet touchscreen while driving. The system uses automatic vehicle location (AVL) to link the specific weather code to the actual milepost. This may result in inaccurately reported weather conditions because of the variation of perception of drivers and the subjectivity in

reporting the different conditions. More importantly, this system may pose some risks to drivers, especially in the very challenging driving environment during adverse weather conditions. In addition, manual identification of weather by the drivers of maintenance vehicles and the processing of this information by Wyoming DOT Traffic Management Center (TMC) personnel to link it to the corresponding road networks are often subjected to human error and require a great deal of processing time. By contrast, the proposed study will automatically collect and extract images from regular road users via smartphone cameras and will detect real-time weather conditions by using machine learning techniques as well as link the weather conditions to the corresponding road networks automatically with no human involvement. The proposed study will, therefore, provide more accurate and consistent weather information in real time that can be made readily available to road users and other transportation practitioners aiming to improve the safety and operation of roadways, especially in adverse weather conditions.

Author Contributions

The authors confirm their contribution to the paper as follows: study conception and design: Md Nasim Khan, Mohamed M. Ahmed; data collection: Md Nasim Khan, Mohamed M. Ahmed; analysis and interpretation of results: Md Nasim Khan, Mohamed M. Ahmed; draft manuscript preparation: Md Nasim Khan, Mohamed M. Ahmed. All authors reviewed the results and approved the final version of the manuscript.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was sponsored by the Wyoming Department of Transportation (DOT). Grant Number: RS05220.

ORCID iDs

Md Nasim Khan  <https://orcid.org/0000-0001-5996-091X>
Mohamed M. Ahmed  <https://orcid.org/0000-0002-1921-0724>

References

1. Federal Highway Administration. How Do Weather Events Impact Roads? - FHWA Road Weather Management. https://ops.fhwa.dot.gov/weather/q1_roadimpact.htm
2. Jonsson, P. Road Condition Discrimination Using Weather Data and Camera Images. *Proc., IEEE Conference on Intelligent Transportation Systems, ITSC*, Washington, DC, USA, IEEE, New York, 2011, pp. 1616–1621.

3. Lee, J., B. Hong, Y. Shin, and Y. J. Jang. Extraction of Weather Information on Road Using CCTV Video. *Proc., International Conference on Big Data and Smart Computing, BigComp 2016*, Hong Kong, China, IEEE, New York, 2016, pp. 529–531.
4. He, K., J. Sun, X. Tang, K. He, J. Sun, and X. Tang. Single Image Haze Removal Using Dark Channel Prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 33, No. 12, 2011, pp. 2341–2353.
5. Asery, R., R. K. Sunkaria, L. D. Sharma, and A. Kumar. Fog Detection Using GLCM Based Features and SVM. *Proc., Conference on Advances in Signal Processing, CASP 2016*, Pune, India, IEEE, New York, 2016, pp. 72–76.
6. Anwar, M. I., and A. Khosla. Classification of Foggy Images for Vision Enhancement. *Proc., 2015 International Conference on Signal Processing and Communication, ICSC 2015*, Noida, India, IEEE, New York, 2015, pp. 233–237.
7. Khan, M. N., and M. M. Ahmed. Trajectory-Level Fog Detection Based on in-Vehicle Video Camera With TensorFlow Deep Learning Utilizing SHRP2 Naturalistic Driving Data. *Accident Analysis & Prevention*, Vol. 142, 2020, p. 105521. <https://doi.org/10.1016/j.aap.2020.105521>.
8. SHRP 2. Trip Files. Analyzing Driver Behavior Using Data From the SHRP 2 Naturalistic Driving Study. *SHRP 2 Safety Project Brief*. Transportation Research Board, Washington, D.C., 2013.
9. Hankey, J. M., M. A. Perez, and J. A. McClafferty. *Description of the SHRP2 Naturalistic Database and the Crash, Near-Crash, and Baseline Data Sets*. Task Report. Virginia Tech Transportation Institute, Blacksburg, 2016.
10. Ahmed, M. M., A. Ghasemzadeh, B. Hammit, M. N. Khan, A. Das, E. Ali, R. K. Young, and H. Eldeeb. *Driver Performance and Behavior in Adverse Weather Conditions: An Investigation Using the SHRP2 Naturalistic Driving Study Data-Phase 2*. Technical Report No. FHWA-WY-18/05F. U.S. Department of Transportation, Washington, D.C., 2018.
11. Ahmed, M. M., R. K. Young, A. Ghasemzadeh, B. Hammit, E. Ali, M. N. Khan, A. Das, and H. Eldeeb. Implementation of SHRP2 Results Within the Wyoming Connected Vehicle Variable Speed Limit System: Phase 2 Early Findings Report and Phase 3 Proposal. Technical Report, U.S. Department of Transportation, Washington, D.C., 2017.
12. Mueller, F. H. *History of Observational Instructions on Fog*. Key Meteorological Records Documentation No. 3.031, U.S. Government Printing Office, Washington, D.C., 1959, pp. 1–8.
13. Murphy, R., R. Swick, B. A. Hamilton, and G. Guevara. *Best Practices for Road Weather Management*. Version 3.0. Report No. FHWA-HOP-12-046. U.S. Department of Transportation, Washington, D.C., 2012.
14. Chawla, N. V. Data Mining for Imbalanced Datasets: An Overview. In *Data Mining and Knowledge Discovery Handbook* (O. Maimon and L. Rokach), Springer, Boston, MA, 2009, pp. 875–886.
15. Google. Imbalanced Data. <https://developers.google.com/machine-learning/data-prep/construct/sampling-splitting/imbalanced-data>. Accessed July 24, 2021.
16. Chawla, N. V., K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-Sampling Technique. *Journal of Artificial Intelligence Research*, Vol. 16, 2002, pp. 321–357. <https://doi.org/10.1613/JAIR.953>.
17. Haralick, R., K. Shanmugam, and I. Dinstein. Textural Features for Image Classification. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 6, 1973, pp. 610–621.
18. Khan, M. N., and M. M. Ahmed. Snow Detection Using In-Vehicle Video Camera With Texture-Based Image Features Utilizing K-Nearest Neighbor, Support Vector Machine, and Random Forest. *Transportation Research Record: Journal of the Transportation Research Board*, 2019. 2673: 221–232.
19. Pomerleau, D. Visibility Estimation From a Moving Vehicle Using the RALPH Vision System. *Proc., of IEEE Conference on Intelligent Transportation Systems*, Boston, MA, IEEE, New York, 1997, pp. 906–911.
20. Gallen, R., A. Cord, N. Hautière, and D. Aubert. Towards Night Fog Detection Through Use of In-Vehicle Multipurpose Cameras. *IEEE Intelligent Vehicles Symposium*, Baden-Baden, Germany, IEEE, New York, 2011, pp. 399–404. <https://doi.org/10.1109/IVS.2011.5940486>.
21. Busch, C., and E. Debes. Wavelet Transform for Analyzing Fog Visibility. *IEEE Intelligent Systems*, Vol. 13, No. 6, 1998, pp. 66–71.
22. Hautière, N., R. Labayrade, and D. Aubert. Real-Time Disparity Contrast Combination for Onboard Estimation of the Visibility Distance. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 7, No. 2, 2006, pp. 201–211.
23. Hautière, N., J. P. Tarel, J. Lavenant, and D. Aubert. Automatic Fog Detection and Estimation of Visibility Distance Through Use of an Onboard Camera. *Machine Vision and Applications*, Vol. 17, No. 1, 2006, pp. 8–20. <https://doi.org/10.1007/s00138-005-0011-1>.
24. Bronte, S., L. M. Bergasa, and P. F. Alcantarilla. Fog Detection System Based on Computer Vision Techniques. *Proc., 12th International IEEE Conference on Intelligent Transportation Systems*, St. Louis, MO, USA, 2009. <https://doi.org/10.1109/ITSC.2009.5309842>.
25. Murphy, K. P. *Machine Learning: A Probabilistic Perspective*. The MIT Press, Cambridge, MA and London, 2012.
26. Das, A., M. N. Khan, and M. M. Ahmed. Detecting Lane Change Maneuvers Using SHRP2 Naturalistic Driving Data: A Comparative Study Machine Learning Techniques. *Accident Analysis and Prevention*, Vol. 142, 2020, p. 105578. <https://doi.org/10.1016/j.aap.2020.105578>.
27. Das, A., and M. M. Ahmed. Machine Learning Approach for Predicting Lane Change Maneuvers Using the SHRP2 Naturalistic Driving Study Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2021. 2675: 574–594.
28. Khan, M. N., A. Das, and M. M. Ahmed. Non-Parametric Association Rules Mining and Parametric Ordinal Logistic Regression for an In-Depth Investigation of Driver Speed Selection Behavior in Adverse Weather Using SHRP2 Naturalistic Driving Study Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2020. 2674: 101–119.

29. Das, A., M. N. Khan, and M. M. Ahmed. Nonparametric Multivariate Adaptive Regression Splines Models for Investigating Lane-Changing Gap Acceptance Behavior Utilizing Strategic Highway Research Program 2 Naturalistic Driving Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2020. 2674: 223–238.
30. Das, A., M. M. Ahmed, and A. Ghasemzadeh. Using Trajectory-Level SHRP2 Naturalistic Driving Data for Investigating Driver Lane-Keeping Ability in Fog: An Association Rules Mining Approach. *Accident Analysis & Prevention*, Vol. 129, 2019, pp. 250–262. <https://doi.org/10.1016/j.aap.2019.05.024>.
31. El Naqa, I., R. Li, and M. J. Murphy. What is Machine Learning? In *Machine Learning in Radiation Oncology* (I. El Naqa, R. Li, and M. J. Murphy, eds.), Springer, New York, NY, pp. 3–11.
32. Samuel, A. Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, Vol. 3, No. 3, 1959, pp. 210–229.
33. Suthaharan, S. *Machine Learning Models and Algorithms for Big Data Classification*. Springer, Boston, MA, 2016.
34. Kramer, O. K-Nearest Neighbors. In *Dimensionality Reduction With Unsupervised Nearest Neighbors* (O Kramer, ed.), Springer, Berlin, Heidelberg, 2013, pp. 13–23.
35. Goodfellow, I., Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, Cambridge, MA, 2016.
36. Khan, M. N., A. Das, M. M. Ahmed, and S. S. Wulff. Multilevel Weather Detection Based on Images: A Machine Learning Approach With Histogram of Oriented Gradient and Local Binary Pattern Based Features. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, Vol. 25, No. 5, 2021, pp. 513–532. <https://doi.org/10.1080/15472450.2021.1944860>
37. Khan, M. N., and M. M. Ahmed. Development of a Novel Convolutional Neural Network Architecture Named RoadweatherNet for Trajectory-Level Weather Detection Using SHRP2 Naturalistic Driving Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2021. 2675: 1016–1030.
38. Gaweesh, S. M., M. N. Khan, and M. M. Ahmed. Development of a Novel Framework for Hazardous Materials Placard Recognition System to Conduct Commodity Flow Studies Using Artificial Intelligence AlexNet Convolutional Neural Network. *Transportation Research Record: Journal of the Transportation Research Board*, 2021. 2675: 1357–1371.
39. Khan, M. N., and M. M. Ahmed. Weather and Surface Condition Detection Based on Road-Side Webcams: Application of Pre-Trained Convolutional Neural Network. *International Journal of Transportation Science and Technology*, 2021. <https://doi.org/10.1016/j.ijtst.2021.06.003>.
40. Patle, A., and D. S. Chouhan. SVM Kernel Functions for Classification. Proc., *International Conference on Advances in Technology and Engineering (ICATE)*, IEEE, Mumbai, India, 2013, pp. 1–9.
41. McDowell, I. *Measuring Health: A Guide to Rating Scales and Questionnaires*. Oxford University Press, New York, NY, 2006.