INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Road Traffic Prediction and Optimal Alternate Path Selection Using HBI-LSTM and HV-ABC

Pankaj Kumar[1]*, Manohar Manur[2], Alok Kumar Pani[3]

1 Assistant Professor, Department of Computer Science and Engineering, Motihari college of Engineering, Bihar, India
2 Associate Professor, Department of Computer Science and Engineering, CHRIST (Deemed to be University), Bangalore
3 Assistant Professor, Department of Computer Science and Engineering, CHRIST (Deemed to be University), Bangalore

OPEN ACCESS

Received: 16-12-2021
Accepted: 09-03-2022
Published: 26.04.2022

Citation: Kumar P, Manur M, Pani AK (2022) Road Traffic Prediction and Optimal Alternate Path Selection Using HBI-LSTM and HV-ABC. Indian Journal of Science and Technology 15(15): 689-699. https://doi.org/10.17485/IJST/v15i15.2360

* **Corresponding author**.

paku9781@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: The objective of this work is to monitor and manage the traffic flow, so an Intelligent Transportation System (ITS) is developed that comprises the fundamental information of the real-time traffic flow. **Methods:** For reducing road Traffic Congestion (TC), this paper proffers an efficient traffic prediction framework and the optimal alternate route selection. The conversion of videos (from surveillance camera) into frames is done, and then pre-processing occurs. Then, for recognizing the traffic on the roadways, the background elimination utilizing Gaussian Mixture Model (GMM) is performed. Next, for identifying the vehicle motion, Motion Estimation (ME) utilizing the Virtual loop-based Lucas-Kanade (VLK) technique is performed. Utilizing the You Only Look Once (YOLO) technique, the frames are segmented centered on the estimated motion for identifying the type of objects on the road. Then, for classifying the traffic centered on the number of objects in the segmented frames, the H-detach optimized Bidirectional Long Short Term Memory (HBI-LSTM) is utilized. The traffic is classified by the classifier as heavy traffic, medium traffic, and low traffic. **Findings:** Utilizing the Horizontal Vertical cross-search appended Artificial Bee Colony (HV-ABC) optimization algorithm, the optimal alternate paths are chosen from different other routes if the traffic is heavy or medium. **Novelty:** The experimental outcomes demonstrate that the other top-notch models are outperformed by the proposed framework.

Keywords: Traffic Congestion; Surveillance Videos; Noise Removal; Motion Estimation; Path Selection; Artificial Bee Colony (ABC) Optimization

1 Introduction

Transportation issues are becoming increasingly difficult with the growth of urbanization and the popularity of automobiles[(1)]. TC is caused by the augment in the number of cars, which is both an economic along with environmental issue[(2)]. Across the world, a common issue is TC which has higher concentrations around cities that affect the

well-being of people and climate[3]. During the recent ten years, TC has become severe, particularly in huge urban areas along with highways connecting disparate cities[4]. Owing to several causes, namely weather conditions, road's poor condition, obstacles in the road that leads to blockage, etc TC occurs[5]. Detailed along with timely traffic flow information is needed by modern urban transport solutions for alleviating congestion[6]. For observing and handling traffic streams in an ITS, Traffic Flow Estimation (TFE) is utilized[7]. A process of examining the traffic conditions on urban roads, and predicting the trends of traffic on roads is called TFE[8]. Therefore, with the quick growth of ITS, it is vital to assist travellers in choosing reasonable travel time along with path[9]. For improving road control, Vehicular traffic surveillance is a vital civilian application[10].

For implementing the advanced technologies of traffic management, one of the most successful technologies is Video-based vehicle detection[11]. Currently, several surveillance cameras are possessed by urban roads and highways that are initially installed for different security reasons[12]. A huge amount of information is generated by these cameras which acts as a basis for the automated traffic surveillance system[13]. Non-intrusiveness, easier maintenance, and a broad variety of data gathering capabilities are the several advantages of utilizing cameras as sensors[14]. A huge, fully-functioning, real-time, precise, and effective transportation management framework is constituted by these systems[15]. For vehicle path planning, a vital reference is the predicted traffic flow since it could assist travellers in making better route choices[16]. But, inappropriate traffic diversion plans will cause problems in the alternate routes[17]. Furthermore, the robustness along with the accuracy analysis of traffic flow will be severely reduced by several variations in the environment[18]. In addition, diverse vehicle appearances, as well as stances, make it difficult to arrange a unified detection model[19].

Xiaohuan Liu et al.[20] developed a hybrid algorithm to assist intelligent driving vehicles in choosing the finest path in the traffic network in emergencies encompassing restricted height, width, accident, weight, along with traffic jams. It is confirmed via the simulation experiments along with practical examination that the HARL (Hybrid A* Algorithm and Reinforcement Learning) algorithm has benefits in path quality, path length, and algorithm efficacy. The outcomes of turns appear to be inadequate because of the environmental restrictions. Himanshu Gupta and Om Prakash Verma[21] introduced aerial image traffic monitoring along with surveillance algorithms centered on deep learning object detection models. The detection results discovered that all the other models were outperformed by the developed traffic recognition model via a minimum margin of 88%, 13%, and 25% for mean average precision, recall, along with F1 score, correspondingly. But, the implementation was limited by a very small object appearance with a cluttered background, especially at higher altitudes. Mohammed A. A. Al-qaness et.al[22] proffered a real-time road traffic management approach utilizing an improved YOLOv3. The evaluation results revealed that suitable performance was attained by the developed system when contrasted to the conventional method of observing vehicle traffic. On the contrary, traffics on cameras was not detected by this model with wide and panoramic coverage along with higher altitude, which covered a considerably larger section of the road. Balaji Ganesh Rajagopal[23] introduced a computer-vision-centered Intelligent Traffic Surveillance System for Indian Road conditions for enhancing the security and surveillance of vehicles in changing weather conditions. The outcomes exhibited that better results were attained by this technique when analogized to the existent traffic prediction models. But, it was perceived that over-segmentation took place for very huge trucks with cabins along with their trailers being identified as separate vehicles regardless of the success of the algorithm.

Shraddha Chaudhary et.al[24] detected a camera-centered traffic monitoring along with a prediction system without detecting or tracking vehicles. 89% accuracy was attained by experimental results on real-time videos. Better accuracy was obtained by the results. The algorithm accuracy reduced gradually when the network's complexity was augmented. Lingyi Han et.al[25] determined a structure that comprised a Deep Clustering (DC) technique for Short-term Traffic Prediction (STTP) in huge-scale road networks. Comparable prediction accuracy was attained by GBM in opposition to IBM with lesser prediction models. Nevertheless, for anticipating traffic in huge-scale data, the clustering technique was not efficient. For overcoming the above limitations, an efficient traffic prediction framework and the optimal alternate route selection are proposed in this paper for reducing road TC. This paper is systematized as: Section 2 explains the proposed traffic prediction framework. Section 3 represents the results and discussions. Lastly, section 4 presents the conclusion along with the future works.

2 Methodology

With the growth of population and economic level, the number of vehicles has been augmenting. In every developed urban area, TC has been increasing. Therefore, this problem could be managed by predicting the dynamic traffic flow. Utilizing HBI-LSTM and HV-ABC algorithms, an efficient road TC prediction, and optimal alternate path selection framework are proffered in this paper. Pre-processing, background subtraction, 'ME', segmentation, classification, and alternate path selection are encompassed by this framework. Figure 1exhibits the proposed TC prediction model's block diagram.

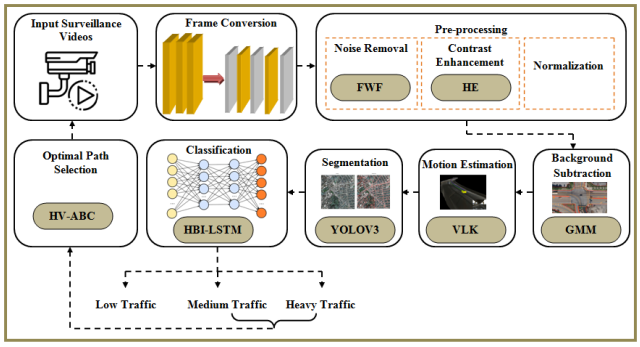

Fig 1. TC prediction framework.

2.1 Pre-processing

From the surveillance cameras, the input videos are initially gathered centered on the vehicle's location. Next, the conversion of videos into frames is done. After frame conversion, preprocessing is performed.

2.1.1 Noise Removal with FWF

FWF is utilized to eliminate noise within the input surveillance videos. For removing the noise present in the image, WF is utilized. It improves the image by inverting the blurring in the image. By preserving the edges along with high-frequency regions within the image, the WF works efficiently. But, the thin lines in the image vanish, and also the image becomes blurred during noise reduction. Centered on fuzzy membership function and the set of fuzzy rules, a new type of WF is presented for overcoming this issue. It avoids the fine lines in the image from disappearing. Next, for reducing the filter's MSE, a set of fuzzy rules is utilized. The steps are explained as.

Step 1: Let the input surveillance video - number of frames. It is signified by, (1)

$$f_i = \{f_1, f_2, \cdots\cdots, f_N\} \tag{1}$$

In equation (1),

Step 2: Both the noise-free and noisy components are encompassed by the image f_i . It is mathematically expressed as,

$$f_i = f_{n,i} + n_i \tag{2}$$

Where, $f_{n,i}$ indicates the noise-free image and n_i signifies the noise component present in the image.

Step 3: For noise removal, the FWF transfer function is presented by,

$$\tilde{f}_i = \left(\frac{T^*(n_i) \cdot S(f_{n,i})}{|T(n_i)|^2 S(f_{n,i}) + S(n_i)} \right) * m(f_i) \tag{3}$$

The above equation, $\tilde{f}_i$ implies the estimated image, $T(n_i)$denotes the degradation function, exhibits the complex conjugate of degradation function, $S(f_{n,i})$ signifies the Power Spectral Density (PSD) of the noise-free image, $S(n_i)$ indicates the PSD of noise component and $m(f_i)$ signifies the membership function. Centered on the input signal's characteristics, the membership function could be estimated.

Step 4: The fuzzy rule is adopted because the noise's amplitude is greater when contrasted to the small changes in the signal. The error function (E) is prferred by,

$$E = f_i - \widetilde{f_i} \tag{4}$$

Next, the fuzzy rule is produced as follows: If E is large, then impulsive noise is encompassed by the image; it could well be removed. If E=0, then there exists no impulsive noise within the image. Let L indicates the average difference betwixt the '2' input images that are chosen arbitrarily. If L is large, then there is impulsive noise in the image. There prevails no noise in the image if it is 0. If E is small and L is small, then the membership function $m(f_i)$is large $(m(f_i) \approx 1)$. Likewise, if E is small along with L is large, then $m(f_i)$is small $(m(f_i) < 1)$. If E is large and L is small, then $(m(f_i) < 1)$ and if both E and L are large, then $(m(f_i) \approx 1)$.

2.1.2 Contrast Enhancement using HE

The steps incorporated in the HE technique which improves image contrast are explained below.

Step 1: Here, the input image f_i'' is deemed with m number of pixels.The probability density function $p(g_l)$ presented by,

$$p(g_l) = \frac{m_l}{m} \tag{5}$$

Where, $l = 0, 1, \cdots\cdots, k-1$ indicates the input image gray level range, g_l signifies the $l-th$ gray level and $m_l$implies the number of pixels in the image possessing gray level g_l.

Step 2: Utilizing equation (6), the cumulative distribution function is also estimated.

$$c(g_l) = \sum_{j=0}^{l} p(g_j) \tag{6}$$

Step 3: The suitable gray level $G_l$is mapped to the gray level g_l of the input image by the HE. Thus,

$$G_l = (k-1) \cdot c(g_l) \tag{7}$$

Step 4: The change in gray level (ΔG_l) could be assessed as,

$$\Delta G_l = (k-1) \cdot p(g_l) \tag{8}$$

Thus, the image's histogram is devastated by HE and it leads to a considerable change in the image's brightness. Next, centered upon the pixel values, the image's normalization occurs.

2.1.3 Normalization

For removing any dissimilarity in the image by modifying the image pixel's intensity values, normalization is utilized. By subtracting the mean as of every pixel, normalization is performed. The image will be normalized in the range [0, 1] or [0, 255]. Therefore, the pre-processed image (F_i) could be modelled after pre-processing,

$$F_i = \{F_1, F_2, \cdots\cdots F_N\} \tag{9}$$

2.2 Background Subtraction

For removing the areas namely lanes, subways, etc that are not needed to predict TC, the background subtraction is performed. Utilizing the GMM, the background elimination is performed in the proposed TC prediction system. The probability of identifying the current pixel value is formulated as,

$$p(i^t) = \sum_{r=1}^{R} W^{rt} * \delta(i^t, \mu r^{rt}, \sum rt) \tag{10}$$

Equation (10), $r = 1, 2, \cdots\cdots R$specifies the number of Gaussian distribution, W^{rt} represents the weight of the r^{th}Gaussian mixture at the time t, μ^{rt} exhibits the mean value, the covariance matrix is denoted as $\sum rt$is δ called the Gaussian probability density function. It is specified in equation

$$\delta(i^t, \mu^{rt}, \sum rt) = \frac{1}{(2\pi)^{N/2} |\sum rt|^{1/2}} \exp\left(-\frac{1}{2}(i^t - \mu)^T \sum^{-1} (i^t - \mu)\right) \tag{11}$$

Where, N signifies the color space dimension. The weights and all the Gaussian parameters are updated if a new pixel matches with any one of the GD functions. The priority for every component is assessed and grouped in descending order after the parameter and weight updation. Next, the GD is categorized as background if it is greater than the designated threshold. If the GD is lesser than the threshold, then it is designated as foreground and is removed.

2.3 Motion Estimation based on VLK

Next, for identifying the vehicle's movements on the roads, the ME is performed. Utilizing the VLK optical flow method, the ME is executed. The motion betwixt '2' successive image frames at a time T and $\partial T$is calculated by the Lucas Kanade

(LK) optical flow model. Since the optical flow computed is unsure of the contour position of motion targets, the identified direction of motion is confusing at the edges of moving objects. This could be overcome by setting an observation window named virtual loop. By setting a threshold value, a few points that are not needed for calculation are neglected by the virtual loop. This modification in the general LK technique is the so-called VLK algorithm. The steps incorporated in VLK for motion detection are explained below.

Step 1: Presume that the image varies over time and could be modelled as the function of time T in the optical flow method. The pixel of the image (i, j) corresponding grayscale could be written as g(i, j, T) at time T. The image pixel shifts to $(i+di, j+dj)$ at the time T+d T. The equation is presented as.

$$g(i, j, T) = g(i+di, j+dj, T+dT) \tag{12}$$

Step 2: Here, using 1st order Taylor expansion, equation (12) could be approximated as,

$$g(i, j, T) = g(i, j, T) + \frac{\partial g}{\partial i}\Delta i + \frac{\partial g}{\partial j}\Delta j + \frac{\partial g}{\partial T}\Delta T \tag{13}$$

The above equation could be simplified as,

$$\frac{\partial g}{\partial i}V_i + \frac{\partial g}{\partial j}V_j + \frac{\partial g}{\partial T} = 0 \tag{14}$$

Where, V_i, V_j denotes the velocities at which the pixel moves correspondingly.

Step 3: Then, for solving the linear equation, the least square method is utilized. An observation window W comprising pixels $(\rho_1, \rho_2, \cdots\cdots, \rho_M)$ is deemed whose grey value is greater when contrasted to a certain threshold (ψ) which is mathematically denoted as,

$$W(i, j) = \begin{cases} 1 & \zeta > \psi \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

Here, ζ implies the grey value of the image pixels. Next, the linear equation is signified as,

$$P = \begin{bmatrix} \frac{\partial g}{\partial i}\rho_1 & \frac{\partial g}{\partial j}\rho_1 \\ \frac{\partial g}{\partial i}\rho_2 & \frac{\partial g}{\partial j}\rho_2 \\ \frac{\partial g}{\partial i}\rho_M & \frac{\partial g}{\partial j}\rho_M \end{bmatrix} \tag{16}$$

$$Q = -\begin{bmatrix} \frac{\partial g}{\partial T}\rho_1 \\ \frac{\partial g}{\partial T}\rho_2 \\ \frac{\partial g}{\partial T}\rho_M \end{bmatrix} \tag{17}$$

Step 4: Next, the velocities V_i, V_jcould be assessed as,

$$\begin{pmatrix} V_i \\ V_j \end{pmatrix} = \left[P^T P\right]^{-1} P^T Q \tag{18}$$

The speed of the pixel movements betwixt the frames is produced by the above equation (18). The block of pixels that appeared in numerous images could be computed when T is discrete. Utilizing the YOLO technique, the objects in the frames are segmented after the motions betwixt frames are assessed.

2.4 Segmentation

The biggest advantage of using YOLO is its superb speed – it's incredibly fast and can process 45 frames per second. YOLO also understands generalized object representation. Three times faster than previous methods like CNN, R-CNN, Fast R-CNN,

Faster R-CNN, Mask R-CNN. This method had the advantage of better detection of smaller objects in the images and detects outputs at three different stages rather than at the final output stage. Several objects in images are detected by YOLO in real-time with high speed, accuracy and it possesses better learning abilities. The input video frame is divided into several Bounding Boxes (BB) from which convolutional features are extracted. The input image F_i is separated into an equal number of $H\ x\ H$ non-overlapping grid cells in YOLO. For identifying the objects belonging to the cell, each grid cell is responsible. The '2' BBs with co-ordinates *(X, Y)* height, width, and also the probability of occurrence of every object are produced. For forecasting the confidence score with its BB for every grid cell, the probability of occurrence is calculated. The confidence score (*cs*) is assessed in equation

$$cs = p(O) + \eta \tag{19}$$

Wherein, $p(O)$ denotes the probability of occurrence of an object, and η signifies the intersection over the union betwixt the ground truth and predicted boxes. The confidence score becomes zero if there is no object in the BB. The intersection over union is offered for the predicted along with ground truth boxes if an object is present in the BB. At last, the type of objects (Humans, Cars, Bikes, etc.) in the roadways is identified by YOLO. It also detects the number of such objects for better classification results.

2.5 Classification using HBI-LSTM

For predicting the TC results, the segmentation results are offered to the HBI-LSTM classifier. Owing to an imbalance in gradient component magnitudes, the gradient component through the cell state paths gets suppressed contrasted to the gradient components through the other paths while the BI-LSTM weights are huge, which is the drawback in BI-LSTM. For overcoming this issue, a positive number less than 1 is multiplied in the cell state for reducing its magnitude. Thus, the suppression of gradient components in the cell state paths is neglected and the gradient is prevented from flowing into the hidden state. Therefore, efficient classification results are attained. This alteration in conventional BI-LSTM is called HBI-LSTM. The HBI-LSTM workflows could be illustrated below,

$$I^t = \chi\left(w^I * \left[h^{t-1}, X^t\right] + b^I\right) \tag{20}$$

$$F^t = \chi\left(w^F * \left[h^{t-1}, X^t\right] + b^F\right) \tag{21}$$

$$O^t = \chi\left(w^O \cdot \left[h^{t-1}, X^t\right] + b^o\right) \tag{23}$$

Wherein, $P \in [0, 1]$ implies a tuneable hyperparameter (positive number). The hidden state output is proffered by,

$$h^t = O^t * \tanh\left(C^t\right) \tag{24}$$

In the above equations, I^t, F^t, O^t, C^t implies the output of input gate, forget gate, output gate, along with cell at time t in the order, X^t denotes the input of BI-LSTM, h^i illustrates the hidden layer at a time *t,* χ signifies the sigmoid Activation Function (AF), *w, b* indicates the network's weight along with bias values respectively, and tanh imples the AF. The sigmoid AF χ is presented by,

$$\chi(X) = \frac{1}{1 + e^{-X}} \tag{25}$$

Likewise, the tanh AF is specified as,

$$\tanh(X) = \frac{e^X - e^{-X}}{e^X + e^{-X}} \tag{26}$$

Thus, the output is generated by this classifier as low traffic, medium traffic, and heavy traffic centered on the number of vehicle counts attained throughout segmentation.

2.6 Optimal Path Selection

After classification, the optimal alternate paths are chosen for decreasing the TCs utilizing the HV-ABC algorithm if heavy or medium traffic is identified. A population-centered algorithm is the Artificial Bee Colony (ABC) optimization which is enthused by the searching behavior of bees. But, the possibility of falling into local optimum solutions is possessed by the ABC

algorithm's convergence rate and the algorithm is not ideal. Therefore, the horizontal along with vertical cross-over search is applied in the EB search phase for overcoming these drawbacks. This alteration in the conventional ABC is called HV-ABC. The HV-ABC steps are elucidated as,

Step 1: Initially, the initial populace of bees (alternate paths) $x_j$is arbitrarily distributed as,

$$x_j = \{x_1, x_2, \cdots\cdots, x_n\} \tag{27}$$

Where, n denotes the number of bees and each bee belong to the d- dimensional space which is proffered by,

$$x_{jk} = \left(x_{j1}, x_{j2}, \cdots\cdots x_{jd}\right) \tag{28}$$

Step2: Utilizing the below equation (29), the FS (problem-solution) for the EB is generated after population initialization.

$$x_{jk} = x'_k + r\left(x''_k - x'_k\right) \tag{29}$$

Here, $x'_k$and x''_k exhibits the lower and also upper bounds of the i^{th} bee in the j^{th} dimension, respectively, and r implies an arbitrary number within [0, 1]. Next, utilizing a horizontal along with vertical cross-over search method, each EB looks for the FS for determining the better solution.

Step 3: By interchanging information, the horizontal crossover search is executed betwixt '2' bees (x_a and x_b) in the populace $\left(x_{jk}\right)$ The horizontal crossover search of bees is specified further,

$$x'_{ak} = r_2 \cdot x_{ak} + (1 - r_2) \cdot x_{bk} + \tau_1 (x_{ak} - x_{bk}) \tag{30}$$

$$x'_{bk} = r_3 \cdot x_{bk} + (1 - r_3) \cdot x_{ak} + \tau_2 (x_{bk} - x_{ak}) \tag{31}$$

In equation (30) & (31), r_2, r_2 denotes the random number within (0, 1), τ_1, τ_2 implies the random number betwixt (-1, 1) and x'_{ak} and x'_{bk}signifies the offspring of x_a and $x_b$in the same order.

Step 4: Next, a vertical crossover search is executed betwixt '2' disparate positions of each bee in the populace. This could be mathematically represented below,

In this equation,

$$x_{jC} = r \cdot x_{jC} + (1 - r) \cdot x_{jE} \tag{32}$$

In this equation, r symbolizes the random number betwixt (0, 1), and x_{jC} and x_{jE}illustrates the position vectors created by the vertical crossover search of bees x_{jk}. Therefore, the chosen FS is denoted as $\vec{x}_{jk}$ after a crossover search. Next, by updating the existing solution, a new candidate solution $\left(u_{jk}\right)$ is produced. It is indicated by,

$$u_{jk} = x_{jk} + r_1\left(x_{jk} - \vec{x}_{jk}\right) \tag{33}$$

In equation (33), x_{jk}signifies the current FS, and r_1 implies the random number in [-1, 1].

Step 5: Then, the fitness is computed. For selecting the best solution, a greedy selection method is employed grounded upon the fitness function. It is signified as,

$$y_{jk} = \begin{cases} u_{jk} & \text{if } f\left(u_{jk}\right) > f\left(x_{jk}\right) \\ x_{jk} & \text{otherwise} \end{cases} \tag{34}$$

Where, $f\left(x_{jk}\right), f\left(u_{jk}\right)$ signifies the fitness of the candidate solutions and y_{jk} denotes the best solution.

Step 6: The FS information is shared by EB with the OB in the form of dance. Centered upon the probability p_k of FSs. It is estimated as,

$$p_k = \frac{f\left(y_{jk}\right)}{\sum_{j=1}^{n} f\left(y_{jk}\right)} \tag{35}$$

Here, $f\left(y_{jk}\right)$ denotes the fitness value of the best solution attained so far. Next, until the stopping criterion is fulfilled, the food searching process continues. Therefore, for traffic prediction, the best solutions (optimal paths) attained are further utilized by gathering the surveillance videos as of that particular optimal path. Until the low traffic optimal path is identified, the process continues.

3 Result and Discussion

By analogizing with the existent techniques, the superiority of the proposed TC prediction and optimal path selection framework is examined. Centered on the accuracy, precision, recall, f-measure, sensitivity, and specificity for HBI-LSTM, fitness value for HV-ABC, Peak Signal to Noise Ratio (PSNR), MSE, and Spearman Rank Correlation (SRC) for FWF, and Motion Estimation Time (MET), Mean Square Error (MSE) for VLK techniques, the performance is examined. From the highway traffic videos dataset, the input data are taken. This is a database of videos of traffic on the highway used in [26] and [27]. From a stationary camera overlooking I-5 in Seattle, WA, the video was taken over two days. The training and test sets used in [26] and [27] are also provided. The proposed work is applied in MATLAB. The performance assessments are elucidated further down.

3.1 Superiority Measure of Proposed FWF

Centered on the above-mentioned performance metrics, the performance of the proposed FWF is analogized with the prevalent models, like WF, Median Filter (MF), and Gaussian filter (GF). Concerning PSNR, MSE, and SRC, the performance analysis is graphically denoted as.

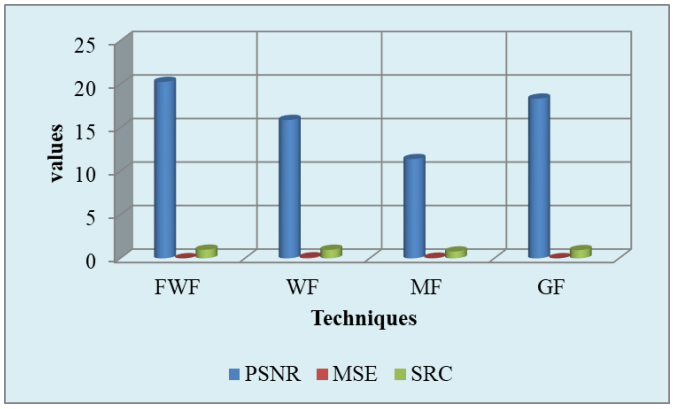

Fig 2. Performance comparison of proposed FWF with respect to PSNR, MSE, and SRC.

Centered on the different evaluation metrics, the proposed FWF with other existent filters' performance evaluation is exhibited in Figure 2. From Figure 2 , the PSNR value of 20.275 is obtained by the proposed FWF, which is higher when analogized to the prevailing WF (15.903), MF (11.406), and GF (18.354). Likewise, the MSE is exhibited in figure 2. 0.01043 is attained by the proposed work, which is very less when analogized to the existent methods. The measure of strength and association betwixt enhanced and original images is called SRC. 0.992 is attained by the proposed FWF while 0.982 (WF), 0.788 (MF), and 0.943 (GF) is attained by the prevalent techniques. Thus, high PSNR, low MSE, and high SRC are accomplished by the proposed system. It renders that the proposed work is better when weighted against the prevailing noise removal methods.

3.2 Performance Assessment of Proposed HBI-LSTM and VLK

By contrasting the outcomes with the prevalent Long Short Term Memory (LSTM), BI-LSTM, Artificial Neural Network (ANN), and CNN), the proposed HBI-LSTM's supremacy is examined. Similarly, utilizing the VLK technique, the performance is assessed by contrasting its MET and MSE with other techniques, namely Lucas Kanade Algorithm (LKA), Horn Schunck Algorithm (HSA), Pel-Recursive Algorithm (PRA) along with Block Matching Algorithm (BMA). Figure 3 renders the performance analysis centered on accuracy, precision, sensitivity, specificity, recall along with F-measure while Figure 4 gives performance analysis of MET and MSE. These metrics are evaluated centered on parameters like 1) true positive ($t^{(p)}$), 2) true negative ($t^{(n)}$), 3) false positive ($f^{(p)}$), together with 4) false negative ($f^{(n)}$). For a better system $f^{(p)}$ should be minimum and $t^{(p)}$ should be maximum.

$$\text{Sensitivity} = \frac{t^{(p)}}{t^{(p)} + f^{(n)}} \tag{36}$$

$$\text{Specificity} = \frac{t^{(n)}}{f^{(p)} + t^{(n)}} \tag{37}$$

$$\text{precision} = \frac{t^{(p)}}{t^{(p)} + f^{(p)}} \tag{38}$$

$$\text{recall} = \frac{t^{(p)}}{t^{(p)} + f^{(n)}} \tag{39}$$

$$\text{accuracy} = \frac{t^{(p)} + t^{(n)}}{t^{(s)}} \tag{40}$$

$$F - \text{score} = 2 \times \frac{\text{precision} * \text{recall}}{\text{precision} * \text{recall}} \tag{41}$$

$$MET = \frac{MET_p - MET_o}{MET_o} \tag{42}$$

$$MSE = \frac{\text{Observed values-Predictedvalues}}{\text{Number of data po int } s} \tag{43}$$

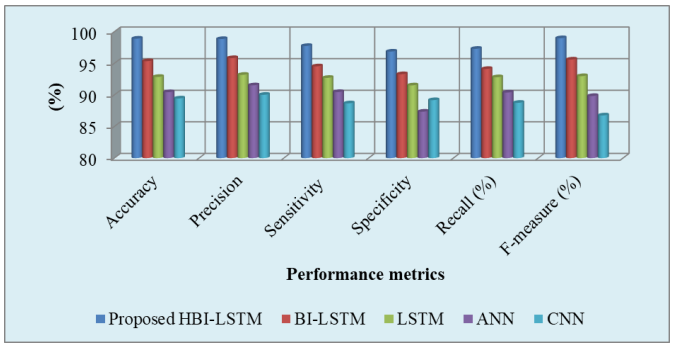

Fig 3. Performance evaluation of proposed HBI-LSTM with existing techniques.

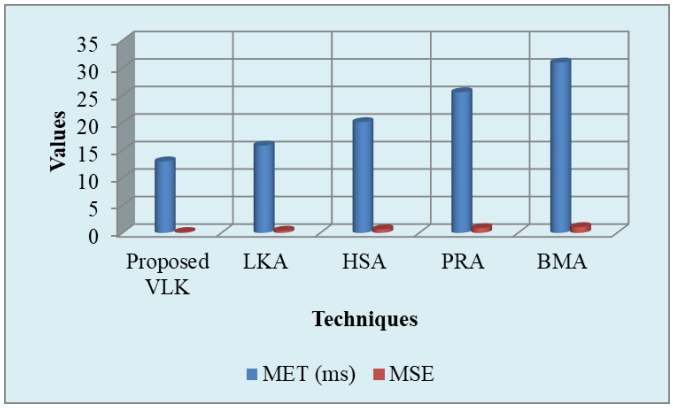

Fig 4. Performance evaluation of proposed VLK with existing techniques.

From Figure 3, it is known that the proposed method attained the values of 98.92%, 98.86%, 97.76%, 96.87%, 97.37%, and 98.99% for metrics like accuracy, precision, sensitivity, specificity, recall, and F-measure which is superior to the other existing methods. Similarly, from figure 4, the method VLK attained the values of 12.987ms and 0.1568 for MET and MSE respectively. This analysis shows the higher performance levels of the proposed work when compared with prevailing works.

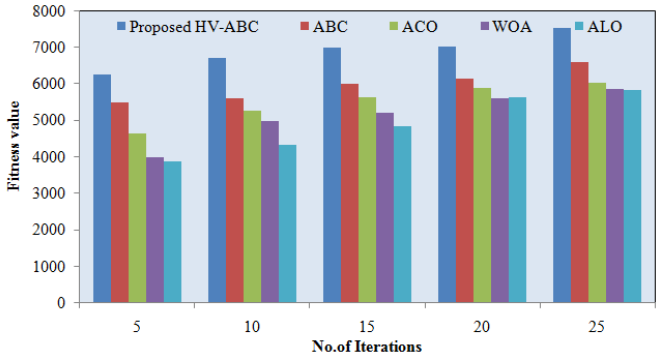

Fig 5. Performance investigation of proposed HV-ABC with other techniques based on fitness value.

3.3 Performance Comparison of Proposed HV-ABC

For verifying the proposed system's superiority, the outcomes of the technique are analogized with the prevailing optimization techniques concerning fitness value. The ABC algorithm, Ant Colony Optimization (ACO) technique, Whale Optimization Algorithm (WOA), and Ant Lion Optimization (ALO) technique are the existent techniques utilized for comparison. The performance comparison is graphically exhibited as.

Figure 5 the best fitness value obtained is 7525 for HV-ABC (iteration 25) and the lowest fitness attained is 6354 for HV-ABC (iteration 5). Meanwhile, lower values are attained by the existent techniques when analogized with the proposed one. Thus, it is apparent that the traffic is well predicted by the proposed TC framework and it chooses the best optimal paths when weighed against the other prevailing traffic prediction systems.

4 Conclusion

Utilizing HBI-LSTM and HV-ABC techniques, an efficient road TC prediction framework, and also the optimal alternate path selection is proffered in this paper. The outcomes illustrate that traffic with higher accuracy (98.92%) and precision (98.86%) for the HBI-LSTM classifier, and low MSE (0.1568) and MET (12.987ms) for VLK ME is predicted by the proposed work. High PSNR (20.275), SRC (0.992), and low MSE (0.01043) are also acquired by the proposed work. It is deduced that other top-notch methods are outperformed by the proposed TC prediction from surveillance videos and it stays more consistent. Furthermore, by deeming the traffic in the lane areas and more complicated road networks, this work will be enhanced in the upcoming future.

5 Author's contributions

All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed byPankaj Kumar, Manohar Manur, Alok Kumar Pani. The first draft of the manuscript was written by Pankaj Kumar and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

6 Acknowledgement

We thank the anonymous referees for their useful suggestions

References

1) Boukerche A, Wang J. Machine Learning-based traffic prediction models for Intelligent Transportation Systems. *Computer Networks*. 2020;181:107530–107530. Available from: https://dx.doi.org/10.1016/j.comnet.2020.107530.
2) Sciberras R, Inguanez F. Road traffic flow estimation via public IP cameras. In: 2018 IEEE 8th International Conference on Consumer Electronics - Berlin (ICCE-Berlin). IEEE. 2018. Available from: http://doi.org/10.1109/ICCE-Berlin.2018.8576229.
3) Sciberras R, Inguanez F. Automated traffic congestion estimation via public video feeds. In: 2019 IEEE 9th International Conference on Consumer Electronics (ICCE-Berlin). IEEE. 2019;p. 8–11. Available from: http://doi.org/10.1109/ICCE-Berlin47944.2019.8966134.
4) Sayani S, Nanvani P. Traffic analysis and estimation using deep learning techniques. *International Journal of Engineering Research & Technology (IJERT)*. 2019;8(9):803–807.

5) Mutharpavalar AA, Ir PS. Measuring of real-time traffic flow using video from multiple ip-based cameras. In: IEEE International Conference on Signal and Image Processing Applications. 2019. Available from: http://doi.org/10.1109/ICSIPA45851.2019.8977742.
6) Hong J, Gao Z, Mei T, Li Y, Zhao C. UAV-based Traffic Flow Estimation and Analysis. In: 2019 IEEE 15th International Conference on Control and Automation (ICCA). IEEE. 2019. Available from: http://doi.org/10.1109/ICCA.2019.8899679.
7) Chen PY, Hsieh JW, Gochoo M, Wang CY, Liao HYM. Smaller Object Detection for Real-Time Embedded Traffic Flow Estimation Using Fish-Eye Cameras. In: 2019 IEEE International Conference on Image Processing (ICIP). IEEE. 2019. Available from: http://doi.org/10.1109/ICIP.2019.8803719.
8) Zhao L, Song Y, Zhang C, Liu Y, Wang P, Lin T, et al. T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction. *IEEE Transactions on Intelligent Transportation Systems*. 2020;21(9):3848–3858. Available from: https://dx.doi.org/10.1109/tits.2019.2935152.
9) Pun L, Zhao P, Liu X. A Multiple Regression Approach for Traffic Flow Estimation. *IEEE Access*. 2019;7:35998–36009. Available from: https://dx.doi.org/10.1109/access.2019.2904645.
10) Meshram SA, Lande RS. Traffic surveillance by using image processing. In: International Conference on Research in Intelligent and Computing in Engineering (RICE). 2018;p. 22–24. Available from: http://doi.org/10.1109/RICE.2018.8627906.
11) Abbas M, Mehboob F, Khan SA, Rauf A, Jiang R. Real Time Fuzzy Based Traffic Flow Estimation and Analysis. In: Advances in Intelligent Systems and Computing. Springer International Publishing. 2019;p. 472–482.
12) Impedovo D, Balducci F, Dentamaro V, Pirlo G. Vehicular Traffic Congestion Classification by Visual Features and Deep Learning Approaches: A Comparison. *Sensors*. 2019;19(23):5213–5213. Available from: https://dx.doi.org/10.3390/s19235213.
13) Fedorov A, Nikolskaia K, Ivanov S, Shepelev V, Minbaleev A. Traffic flow estimation with data from a video surveillance camera. *Journal of Big Data*. 2019;6(1):1–15. Available from: https://dx.doi.org/10.1186/s40537-019-0234-z.
14) Cheng HY. Highway Traffic Flow Estimation for Surveillance Scenes Damaged by Rain. *IEEE Intelligent Systems*. 2018;33(1):64–77. Available from: https://dx.doi.org/10.1109/mis.2018.111144331.
15) Nagy AM, Simon V. Survey on traffic prediction in smart cities. *Pervasive and Mobile Computing*. 2018;50:148–163. Available from: https://dx.doi.org/10.1016/j.pmcj.2018.07.004.
16) Wu Y, Tan H, Qin L, Ran B, Jiang Z. A hybrid deep learning based traffic flow prediction method and its understanding. *Transportation Research Part C: Emerging Technologies*. 2018;90:166–180. Available from: https://dx.doi.org/10.1016/j.trc.2018.03.001.
17) Memarian A, Rosenberger JM, Mattingly SP, Williams JC, Hashemi H. An optimization-based traffic diversion model during construction closures. *Computer-Aided Civil and Infrastructure Engineering*. 2019;34(12):1087–1099. Available from: https://dx.doi.org/10.1111/mice.12491.
18) Mehboob F, Abbas M, Almotaeryi R, Jiang R, Al-Maadeed S, Bouridane A. Traffic Flow Estimation from Road Surveillance. In: 2015 IEEE International Symposium on Multimedia (ISM). IEEE. 2015. Available from: http://doi.org/10.1109/ISM.2015.14.
19) Appathurai A, Sundarasekar R, Raja C, Alex EJ, Palagan CA, Nithya A. An Efficient Optimal Neural Network-Based Moving Vehicle Detection in Traffic Video Surveillance System. *Circuits, Systems, and Signal Processing*. 2020;39(2):734–756. Available from: https://dx.doi.org/10.1007/s00034-019-01224-9.
20) Liu X, Zhang D, Zhang T, Cui Y, Chen L, Liu S. Novel best path selection approach based on hybrid improved A* algorithm and reinforcement learning. *Applied Intelligence*. 2021;51(12):9015–9029. Available from: https://dx.doi.org/10.1007/s10489-021-02303-8.
21) Gupta H, Verma OP. Monitoring and surveillance of urban road traffic using low altitude drone images: a deep learning approach. *Multimedia Tools and Applications*. 2021. Available from: https://dx.doi.org/10.1007/s11042-021-11146-x.
22) Al-qaness MAA, Abbasi AA, Fan H, Ibrahim RA, Alsamhi SH, Hawbani A. An improved YOLO-based road traffic monitoring system. *Computing*. 2021;103(2):211–230. Available from: https://dx.doi.org/10.1007/s00607-020-00869-8.
23) Rajagopal BG. Intelligent traffic analysis system for Indian road conditions. *International Journal of Information Technology*. 2020. Available from: https://dx.doi.org/10.1007/s41870-020-00447-3.
24) Chaudhary S, Indu S, Chaudhury S. Video-based road traffic monitoring and prediction using dynamic Bayesian networks. *IET Intelligent Transport Systems*. 2018;12(3):169–176. Available from: https://dx.doi.org/10.1049/iet-its.2016.0336.
25) Han L, Zheng K, Zhao L. Xianbin Wang and Xuemin (Sherman) Shen. Short-term traffic prediction based on deepcluster in large-scale road networks. *IEEE Transactions on Vehicular Technology*. 2019;68(12):12301–12313. Available from: http://doi.org/10.1109/TVT.2019.2947080.
26) Chan AB, Vasconcelos N. Probabilistic Kernels for the Classification of Auto-Regressive Visual Processes. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. 2005. Available from: http://doi.org/10.1109/CVPR.2005.279.
27) Chan AB, Vasconcelos N. Classification and retrieval of traffic video using auto-regressive stochastic processes. *IEEE Proceedings Intelligent Vehicles Symposium*. 2005. Available from: http://doi.org/10.1109/IVS.2005.1505198.