DOI: 10.24996/ijs.2021.62.6.30

ISSN: 0067-2904

Vehicles Detection System at Different Weather Conditions

Mustafa Najm Abdullah, Yousra Hussein Ali
Department of Computer Science, University of Technology, Baghdad, Iraq.

Received: 2/6/2020 Accepted: 25/8/2020

Abstract

The importance of efficient vehicle detection (VD) is increased with the expansion of road networks and the number of vehicles in the Intelligent Transportation Systems (ITS). This paper proposes a system for detecting vehicles at different weather conditions such as sunny, rainy, cloudy and foggy days. The first step to the proposed system implementation is to determine whether the video's weather condition is normal or abnormal. The Random Forest (RF) weather condition classification was performed in the video while the features were extracted for the first two frames by using the Gray Level Co-occurrence Matrix (GLCM). In this system, the background subtraction was applied by the mixture of Gaussian 2 (MOG 2) then applying a number of pre-processing operations, such as Gaussian blur filter, dilation, erosion, and threshold. The main contribution of this paper is to propose a histogram equalization technique for complex weather conditions instead of a Gaussian blur filter for improving the video clarity, which leads to increase detection accuracy. Based on the previous steps, the system defines each region in the frame expected to contain vehicles. Finally, Support Vector Machine (SVM) classifies the defined regions into a vehicle or not. As compared to the previous methods, the proposed system showed high efficiency of about 96.4% and ability to detect vehicles at different weather conditions.

Keywords: Vehicle Detection, ITS, Gray Level Co-occurrences Matrix method (GLCM).

نظام الكشف عن المركبات لظروف الطقس المختلفة

مصطفى نجم عبدالله ، يسرا حسين علي
قسم علوم الحاسبات ، الجامعة التكنولوجية ، بغداد ، العراق *

الخلاصة

تزداد أهمية الكشف الفعال عن المركبات (VD) مع التوسع في عدد شبكات الطرق والمركبات في أنظمة النقل الذكية (ITS). تقترح هذه الدراسة نظامًا للكشف عن المركبات في ظروف الطقس المختلفة مثل الأيام المشمسة والأيام الممطرة والأيام الغائمة والأيام المضببة. تتمثل الخطوة الأولى لتنفيذ النظام المقترح في تحديد ما إذا كانت حالة الطقس للفيديو طبيعية ام لا. تقوم الغابة العشوائية (RF) بإجراء تصنيف لحالة الطقس في الفيديو، حيث يتم استخراج الميزات من اول اطارين باستخدام مصفوفة التواجد المشترك ذات المستوى الرمادي (GLCM). في هذا النظام، يتم تطبيق عملية الطرح في الخلفية بواسطة خليط من Gaussian (MOG 2) ثم تطبيق عدد من عمليات المعالجة المسبقة مثل (Gaussian blur filter ، و تمدد ، وتآكل ، وعتبة). تتمثل المساهمة الرئيسية لهذه الدراسة في اقتراح تقنية معادلة المدرج التكراري لظروف الطقس المعقدة بدلاً من

*Email: mostafaprince101@gmail.com

مرشح طمس غاوسي لتحسين وضوح الفيديو، مما يؤدي إلى زيادة دقة الكشف. بناءً على الخطوات السابقة، يحدد النظام كل منطقة في الإطار من المتوقع أن تحتوي على مركبة. أخيرًا، تقوم آلة ناقل الدعم (SVM) بتصنيف المناطق المحددة كونها مركبة أو ليست كذلك. تظهر النتائج التجريبية أن النظام المقترح كان دقيقا وقويا بما يكفي للعمل في ظروف الطقس المختلفة. بالمقارنة مع الطرق السابقة، أظهر النظام المقترح كفاءة عالية وقدرة على اكتشاف المركبات في ظروف الطقس المختلفة، حيث حقق النظام دقة بنسبة 96.4٪.

1. Introduction

With increasing the need for the VD processes in ITSs in the current days, the need for efficient and robust vehicle detection systems that are able to work in all conditions was increased. To accumulate the traffic information for better employment in the highway systems and increase the safety of transport, huge amounts of money were invested in the traffic monitoring system (TMS), which is considered as a fundamental part of the ITS [1].

In the vehicle detection fields, most systems face many problems and challenges which they are trying to overcome in order to reach a strong and effective system operating at different weathers. The main challenges are summarized in the following points [2]:

1) The systems attempt to classify the vehicles according to their types, such as Sedan, SUV, VAN Truck, and so on.

2) For detection, these systems should be accurate and effective at different weather and lighting conditions.

3) To work in real-time applications, the systems should reduce processing time as much as possible.

Due to these difficulties, researchers have employed diverse data acquisition and algorithms to solve vehicle detection problems. The traffic controller, which exists in each intersection area, controls the modern traffic systems, wherein intersections are managed based on the coordination between the controllers of the different roadways. Real-time conclusions cannot be made based on the current road condition [3]. Image processing is one of the data acquisition approaches where diverse approaches of image processing are utilized to detect road vehicles. The commonly used approaches such as background subtraction and edge detection assist in increasing the efficiency of vehicle detection.

2. Relevant Works

Many researches have been focusing on vehicle detection algorithms at different weather conditions, some of which can be summarized as follows**:**

Yi-Ming et al. [2] proposed a vehicle detection system which worked in various illumination and weather conditions, depending on computer vision tools. Four points were used for vehicle detection in this system. These points were underneath, vertical edge, symmetry, and rear light. Particle filter was utilized to multiple vehicle detection via a tracking approach called the Basic Sequential Algorithmic Scheme (BSAS). The adopted system achieved detection accuracy of 84.9% in sunny and darky conditions, noting that several cases exhibited unsatisfactory outcomes. Wu and Juang [4] suggested a vehicle detection system in traffic complex conditions. They used the histogram extension to adjust contrast, and then applied Gray-level Differential Value Method (GDVM) to detect moving objects from the video frames. The last step was tracking and error compensation in order to improve the objective tracking accuracy. Experimental results demonstrated that the suggested system was good in traffic jams and complex weather conditions but the accuracy at rainy and dark conditions was not good enough. Yang and Qu [5] proposed a system to detect and track vehicles in complex traffic, where the system was depending on a background subtraction approach using ‘low rank + sparse’ dismantling. Kalman filter algorithm was utilized to track multiple moving objects. The proposed approach yielded good experimental results in both qualitative and quantitative evaluations. Kumar *et al*. [6] suggested an approach for counting and classifying vehicles by regression, which started with preprocessing low quality videos by deleting noise utilizing Bi-lateral filter, then creating Background mask using the Multi-layer Background subtraction method. The construction of a fast performing kernel was used after that to extract the Foreground mask using a mixture of Gaussians. Both of the extracted contour and cascaded regressions were the results of the foreground moving objects in the Low-quality video. Ershadi *et al*. [7] proposed an approach for the detection and tracking of vehicles. The approach started by using a modified inverse perspective mapping to eliminate the perspective from the video. They applied Hough transform to detect road lines and moving objects based on `Gaussian Mixture Models (GMM), where a chromaticity-based strategy was applied. The results indicated that the suggested system was good, particularly in occlusions and weather conditions. However, this approach was not tested under camera vibrations and dusty weather. Ershadi and Menéndez [8] suggested a system to track and count vehicles in sandy weather with the shaking of the camera. The strategy was based on background subtraction combined with additional processing for segmenting vehicles. This process involved the analysis of the headlight size, location, and area. Then, the tracking was applied via a particle filter to detect the vehicle and compare the headlights using the connected component analysis. Therefore, the vehicle counting implemented in this work was depending on the pairing result. The obtained results showed that the strategy was suitable for difficult conditions, such as nights and snowy and sandy weather with a vibrating camera, but it was difficult to detect white vehicles under the snowy weather. Additionally, vibrations of the camera or background movement possibly influenced the background subtraction method. Hsu-Yung Cheng [9] proposed an estimation system for freeway traffic monitoring applications under bad weather conditions (rainy). Traffic flow estimating approach used diverse features of day and night scenes to estimate the flow. Then, the graph-based mapping model integrated the information of the vehicle count sequences and compensated for the mistakes of per-frame vehicle counting estimate. The experiment results demonstrated that the suggested system was able to analyze traffic flow under rainy conditions for freeway surveillance cameras. Padilla *et al.*[3] proposed a system for vehicles and weather detections via the optical flow and color histogram methods. They were able to use the histogram and optical flow analysis to analyze frames of the video. Raindrops were placed through spatial and temporal analysis, where the result was then deciding the weather condition in the video (raindrops or normal). Testing the proposed system through fair weather displayed small deviations among the real and theoretical values. The same was observed for rainy conditions

3. Feature Extraction

Frame features are the properties that indicate the frame completely. Feature extraction is the process of defining the behavior of each pixel in the frame, depending on geometrical techniques that allow distinguishing each pixel. The most effective features that are utilized for vehicle image classification are the Texture Features, which depend on the repetition regarding the patterns or patterns over the image region. The texture consists of essential information associated with the main structural arrangement regarding the image surfaces, which is important for the computer vision system to detect the edges of the objects [10] .

The GLCM method takes into concern the spatial pixel correlation, which specifies the frame texture by calculating the number of occurrences of the pixel with absolute values in a specific spatial correlation in the frame, which produces a Grey-Level Spatial Dependence Matrix, then extracts statistical measurements from the matrix [11].

The dimensions of the GLCM matrix are determined based on the number of gray levels (N) in the frame, which contains information about the frequency of occurrence of two neighboring pixel combinations in a gray image. Texture features are defined by six features that are produced by the GLCM, including contrast, correlation, homogeneity, angular second moment, energy, and dissimilarity. Below is a detailed explanation of the six extracted features that were adopted in the present study [12]:

a. Contrast: It calculates the amount of local variation that occurs in the image, and is expressed as follows:

$$\boldsymbol{Contrast} = \sum_{i,j=1}^{N} P(i,j)(i-j)^2 \qquad \dots (1)$$

where i and j indicate the matrix coordinates, N represents matrix dimension, and P (i, j) denotes the probability value of the matrix at (i, j) coordinates.

b. Correlation: It is the linear dependence of grey levels in GLCM. If the elements of the matrix are equivalent, then the correlation is high. The value of the correlation can be calculated as follows:

$$\boldsymbol{Correlation} = \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{[(i,j)P(i,j)\,] - \mu_i \cdot \mu_j}{\sigma_i \cdot \sigma_j} \qquad \dots (2)$$

c. Homogeneity: It calculates the local change of frame texture. It is an important value which shows that the frame's texture packs variations between the different areas. It can be measured by the following equation:

$$\boldsymbol{Homogeneity} = \sum_{i,j=1}^{N} \frac{P(i,j)}{1+(i-j)^2} \qquad \dots (3)$$

d. Angular Second Moment (ASM): It calculates the frame homogeneity, reflecting the grey equivalence and the texture roughness of the frame. ASM can be measured using the following equation:

$$\boldsymbol{ASM = \sum_{i,j=1}^{N}\{P(i,j)\}^2} \quad \textbf{... (4)}$$

e. Energy: It is the second angular moment of grey levels. For calculating the energy, the square root of the ASM(as in equation 5) or the square of all pixels in GLCM (see equation 6) can be used:

$$\boldsymbol{Energy = \sqrt{ASM}} \quad \textbf{... (5)}$$

$$\boldsymbol{Energy = \sum_{i,j} P(i,j)^2} \quad \textbf{... (6)}$$

f. Dissimilarity: It is similar to GLCM contrast (variation of grey levels), and it has a high value when the local area has a high contrast. Equation (7) is used to calculate the dissimilarity.

$$\boldsymbol{Dissimilarity = \sum_{i,j=1}^{N}|i-j|\ \mathrm{P}(i,j))} \quad \textbf{... (7)}$$

1. Background Subtraction (BS)

BS is one of the essential steps of many computer vision systems. It is used for segmenting the parts of a scene into the foreground (remained frames) and background (removed frames). This process depends on the estimate of variances between the foreground and the background to produce a new image, which depends on a given threshold to detect the moving objects in the scene from the static objects [13]. Depending on the hypothesis that background is more repeatedly noticeable than the foreground, and that background variance is little, a mixture of K Gaussian distribution models can be made for each pixel in the scene. The probability that a specific pixel has a value of X_t at time t is displayed in the equation bellow [14]:

$$\boldsymbol{P(X_t) = \sum_{i=1}^{K}\omega_{i,t}\cdot\eta(X_t,\mu_{i,t},\sum i,t)} \quad \textbf{... (8)}$$

$$\boldsymbol{\eta(X_t,\mu,\sum) = \frac{1}{(2\pi)^{n/2}|\sum|^{1/2}}exp^{-\frac{1}{2}(X_t-\mu_t)^T\sum_{i,t}^{-1}(X_t-\mu_t)}} \quad \textbf{... (9)}$$

Once the background is defined, the different parameters of the mixture of Gaussians must be initialized. The parameters of the MOG's are the number of Gaussians K, the weight $W_{i,\,t}$ associated with the ith Gaussian at time t, the mean $\boldsymbol{\mu}_{\mathrm{i,t}}$ and the covariance matrix $\Sigma_{\mathrm{i,t}}$. The mixture of Gaussian 2 aims to solve the problem of the fixed amount of used distributions. By using an adaptable amount of Gaussian distributions, mapped pixel by pixel, the MOG2 succeeds in improving the representation of the complexity of colors in each frame [14].

4. Random Forest Classifier (RFC)

RFC is one of the most powerful methods in pattern recognition and machine learning for high-dimensional classification. RFC contains a group of decision trees where each tree is produced by utilizing a random vector sampled independently from the input vector, and each tree performs an entity vote for the most general class to categorize an input vector. The RFC concept is to construct binary subtrees via the training bootstrap samples from the learning sample L, and choosing randomly, at each node, a subgroup of X [15]. The design of decision trees requires the selection of a characteristic selection measure. The random forest classifier uses the Gini Index as a characteristic selection measure, which measures the impurity of a characteristic that is related to the classes. For an assumed training set T, and selecting one pixel at random and saying that it belongs to class Ci, the Gini index equation is written as follows [16]:

$$\boldsymbol{\sum\sum_{j\neq i}(f(Ci,T)/|T|)(f(Cj,T)/|T|)} \quad \textbf{... (10)}$$

where f (Ci, T)/|T| is the probability of the chosen case to belong to Ci class. The RFC consists of N trees, as defined in a previous work [16].

5. Support Vector Machines Classifier (SVMsC)

SVMsC is dependent on the statistical learning principle, which has the objective of identifying the decision boundaries that provide the optimal isolation between the classes and leave the greatest margin between the two classes. The margin is the sum of the distances to the hyperplane from the nearest points of the classes. If the two classes are not linearly separable, the SVMsC attempts to discover the hyperplane that maximizes the margin, besides,minimizing a quantity proportional to the number of misclassifications. The trade-off between the margin and misclassification is controlled by a positive user-defined parameter C [16].

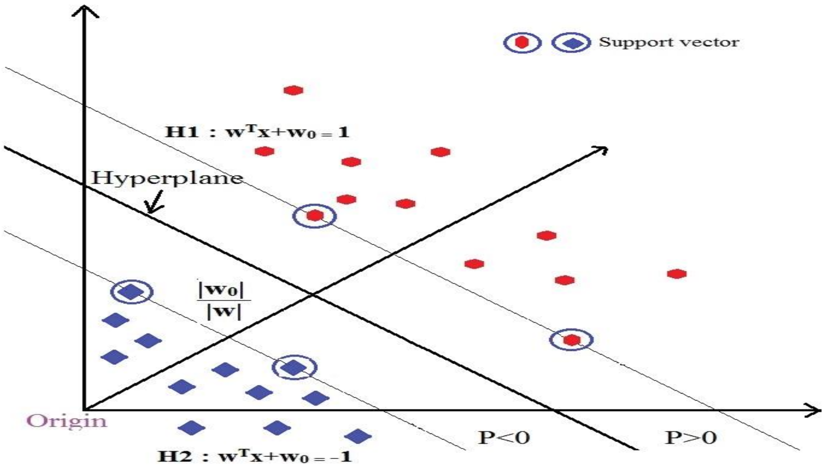


Figure 1-Two class hyperplane SVMsC.

Assume the training dataset to SVMsC of the formula {w_i, y_i}, i = 1, 2, 3…, and that y_i is either 1 or -1. As displayed in Figure- 1, the hyperplane isolates the red positive and the blue negative points. The points that are put on the hyperplane w satisfy the relation $w^t x + w_0 = 0$, where x is normal to the hyperplane, as in equation (11), $|W_0|/|W|$ is the perpendicular distance from the hyperplane to the origin, and |w| is the Euclidean normal of w, as in equation (12). The reply principle for the data classification for Figure-1 is set according to the hyperplane with margin, as shown in an earlier study [16]:

$$\boldsymbol{W^T + W_0 \geq +1 \quad for\ class - 1\ H1} \qquad \textbf{... (11)}$$

$$\boldsymbol{W^T + W_0 \leq -1 \quad for\ class - 2\ H2} \qquad \textbf{... (12)}$$

These can be joined into one set of inequalities:

$$\mathbf{yi}(\boldsymbol{W^T + W_0}) \mathbf{- 1 \geq 0} \quad \forall \boldsymbol{i} \qquad \textbf{... (13)}$$

6. Design Methodology

In this paper, a vehicle detection system is proposed to suit various normal and complex weather conditions, such as snow, fog, rain and cloud. The steps of the proposed system are illustrated in Figure-2. The proposed system begins by extracting gray features from the frame to determine the video weather condition; Random Forest Classifier (RFC) is adopted in the classification. The system changes the pre-processing operations and the value of their parameters according to the weather condition. After the weather condition is determined by the system, background subtraction process is performed using a Mixture of Gaussian 2 (MOG2) method to detect the moving objects.

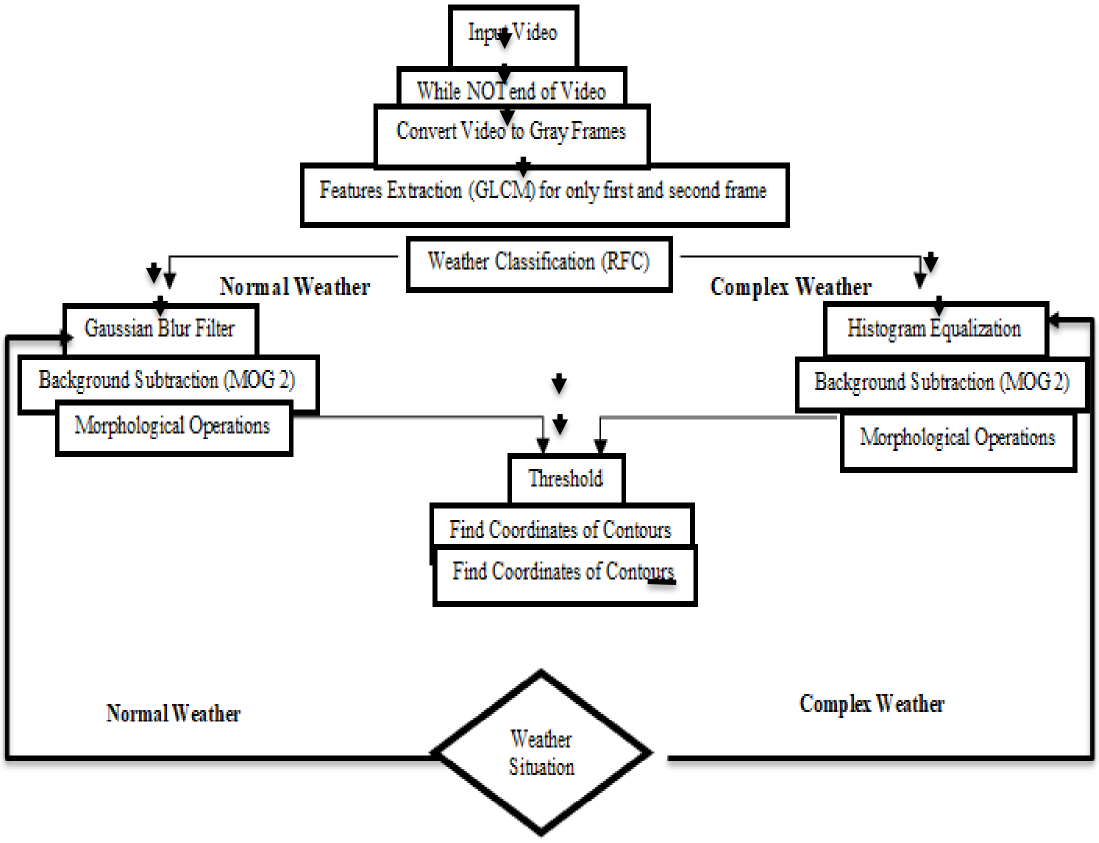

Input Video
While NOT end of Video
Convert Video to Gray Frames
Features Extraction (GLCM) for only first and second frame
Weather Classification (RFC)
Normal Weather
Complex Weather
Gaussian Blur Filter
Background Subtraction (MOG 2)
Morphological Operations
Histogram Equalization
Background Subtraction (MOG 2)
Morphological Operations
Threshold
Find Coordinates of Contours
Find Coordinates of Contours
Normal Weather
Weather Situation
Complex Weather

Figure 2-The steps of the proposed system.

As mentioned above, the pre-processing operations and values of their parameters depend on the state of the weather; the Gaussian blur filter, morphological, and threshold operations were applied on the normal weather. While, in the complex weather, the Gaussian blur filter, histogram equalization, morphological, and threshold operations were adopted. Videos at cloudy, snowy, or rainy weather conditions have poor visibility, causing a difficulty in detecting objects. In order to resolve this problem, the histogram equalization method was used to enhance the contrast of videos, while GLCM was applied for extracting features from the detected objects as well as SVMC for determining whether the detected object is a vehicle or not. All the above steps are explained in the algorithm (1).

Algorithm (1): Proposed System Steps.

Input: Input Video.
Output: Vehicle detection video.

```
Begin :
Step 1: Read Input Video
Step 2: While ( Not End of video) do                    // Split video into sequences of frames
        Read Next_ Frame
        Do For i=0
            Do For j=0
                $G(i,j)=0.299×FR(i,j)+0.587×FG(i,j)+0.114×FB(i,j)$   // Convert each frame into Gray level
            Loop until i= Colored Image width -1
        Loop until j= Colored Image high -1
Step 3: if frame_count == 2 then
              Let I = first frame                       // Feature Extraction by
```

```
GLCM
        Let J = second frame
        Compute the GLCM values for I,J
        Compute Contrast  Value using equation 1              // compute contrast b
        Compute Correlation Value using equation 2            // compute correlation
        Compute Homogeneity Value  using equation 3           // compute homogeneity
        Compute Energy Value using equation 5                 // compute energy
        Compute ASM Value using equation 4                    // compute ASM
        Compute Dissimilarity Value using equation 7          // compute dissimilarity
        Classification I , J Weather_State using equation 10  // Classification two frame by Random Forest Classifier
Step 4: remove noise by Gaussian Blur Filter                  // Apply Gaussian blur filter for each frame
Step 5: if Weather_State==1 then                              // in Complex Weather condition
          Apply Histogram Equalization                        //implement histogram equalization approach for each frame
          Remove background using equation 9                  // perform background subtraction for each frame
          Apply Morphological Operations                      //dilation and erosion
        Else if Weather_State==0 then
          Remove background  using equation 9                 // perform background subtraction for each frame
          Apply Morphological Operations                      // dilation and erosion
Step 6: If f (x, y) > T then f (x, y) = 0 else f (x, y) = 255 //segment an image using the threshold.
Step 7: contours, hierarchy = cv2.findContours(threshold,
                    cv2.RETR_EXTERNAL,
                    cv2.CHAIN_APPROX_SIMPLE)                  // find contour in threshold frames using find Contours function in cv2 in python
Step 8: Do for contour in contours                            // loop over each contour
          Compute Contrast  Value using equation 1            // compute contrast b
          Compute Correlation Value using equation 2          // compute correlation
          Compute Homogeneity Value  using equation 3         // compute homogeneity
          Compute Energy Value using equation 5               // compute energy
          Compute ASM Value using equation 4                  // compute ASM
          Compute Dissimilarity Value using equation 7        // compute dissimilarity
          Classification I , J Weather_State using equation 13 // Classification two frame by support vector machine classifier
Step : End While
End
```

The GLCM approach begins with generating the probability matrix for each frame in the scene and, depending on the generated matrix, the features are extracted. The main features extracted in the proposed system include contrast, correlation, homogeneity, dissimilarity, energy, and ASM for both weather and vehicles.

Figure-3 demonstrates the features extraction from the gray frame, where, in the weather classification process, the system takes two frames and converts them into the gray level to check if the weather conditions being normal weather or complex. In the object classification, each frame in the video is converted into the gray level to detect if the moving objects being a vehicle or not.

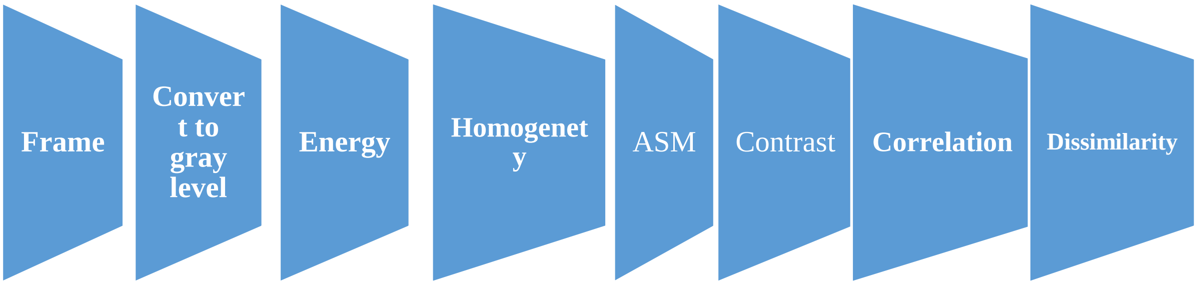


Figure 3- Frame features extraction.

In this paper, the proposed method used a background subtraction approach called MOG2 for the detection of moving objects in the scene. Figure-4 demonstrates the implementation of MOG 2 in a sequence of frames for four videos in normal and complex weather conditions.

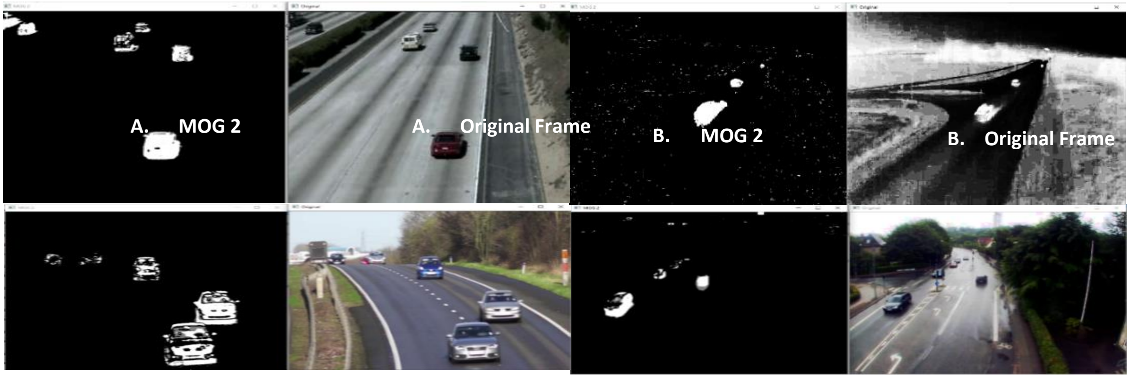


Figure 4-Implementation of a Mixture of Gaussian 2 in the sequential frames of videos.

7. Pre-processing Operations

In the preprocessing operations step, the proposed system begins with converting each frame into a gray level, where the value of each pixel represents only spectral intensity information and ranges between 0 and 255. The grey scale image (G) is produced by calculating the intensity of the illumination of the color image, using equation (14):

$$\mathbf{G = 0.299 \times \mathit{R} + 0.587 \times \mathit{G} + 0.114 \times \mathit{B}} \qquad \textbf{... (14)}$$

Then Gaussian blur filter is applied on the gray scale frames to remove the noise. The Gaussian function is expressed in one dimension and two dimensions, as in equations 15 and 16:

$$\mathbf{G(X)} = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \qquad \textbf{... (15)}$$

$$\mathbf{G(X, Y)} = \frac{1}{2\pi\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad \textbf{... (16)}$$

Histogram equalization technique is used when the weather condition is abnormal for image contrast enhancement. Figure-5 illustrates the outcome of using a histogram equalization technique in the video frames sequence.

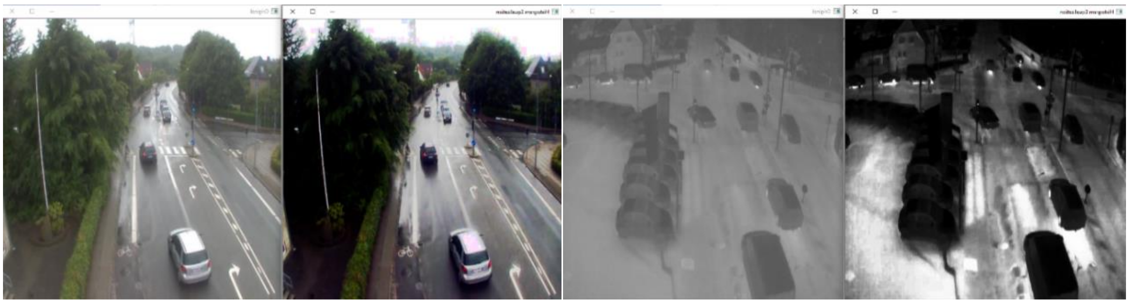


Figure 5- Application of of histogram equalization.

Suppose that F(i, j) is a given image represented as a matrix of integer pixel intensities with a range from 0 to L − 1. (L) represents the number of probable intensity values, which is mostly 256. Let P indicate the normalized histogram of F. As a result:

$$P_n = \frac{number\ of\ pixels\ with\ intensity\ \ n}{total\ number\ of\ pixels} \qquad n= 0, 1\dots, L\text{-}1. \qquad \textbf{... (17)}$$

The histogram equalized image g will be defined by:

$$g_{i,j} = floor\left((L-1)\sum_{n=0}^{f_{i,j}} p_n\right) \qquad \textbf{… (18)}$$

Where floor () are the rounds down (Approximate value) to the nearest integer.

Then morphological (dilation and erosion) operations are performed depending on the pixel values. When dilation is applied, objects in the image are expanded, while when erosion is applied, objects in the image become smaller. Dilation and erosion are shown in Figure-6.

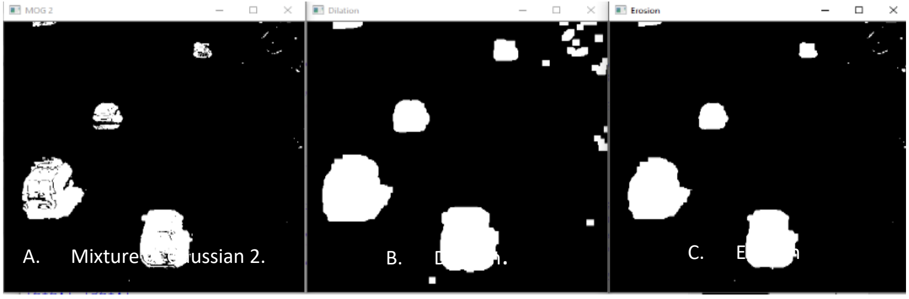


Figure 6- Morphological operations.

Thresholding is the final step within the pre-processing operations, which is implemented to convert the pixel that has intensity less than the threshold into black, whereas if an intensity of the pixel is larger than the threshold it is converted into white, as shown in Figure-7.

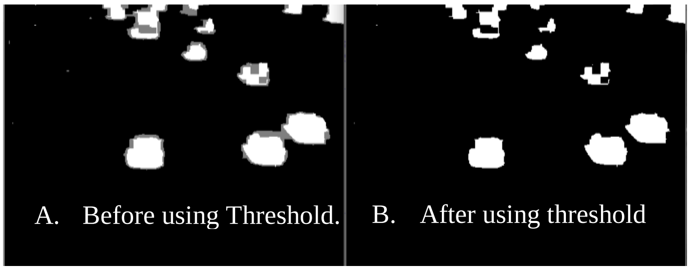


Figure 7-Results of the proposed threshold method.

8. Results and Discussion

In the proposed system, two different classification models were built, which needed two different datasets. In the weather classification, a random forest classifier model was trained on a dataset consisting of 2 classes: normal weather condition with 122 images and complex weather condition with 81 images, as shown in Figure-8.

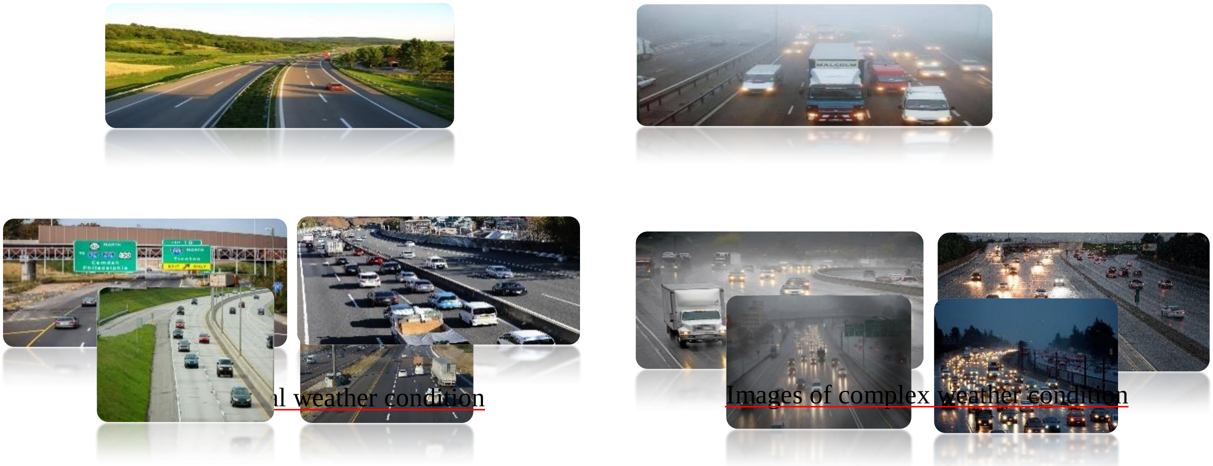


Figure 8-Dataset in different conditions of weather.

The training and testing phase in the suggested system for the extraction and classification of features is illustrated in Figure-9.

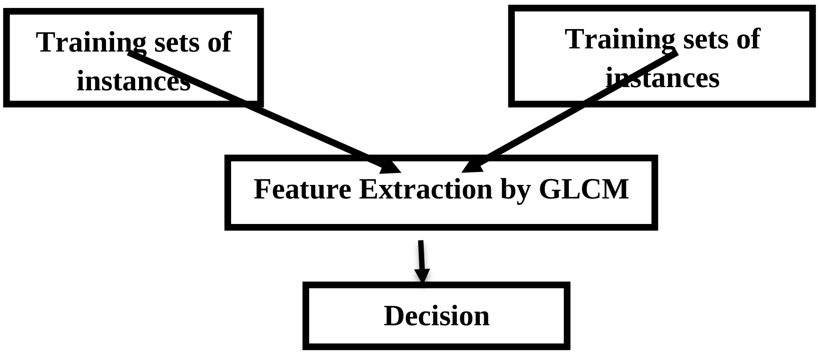
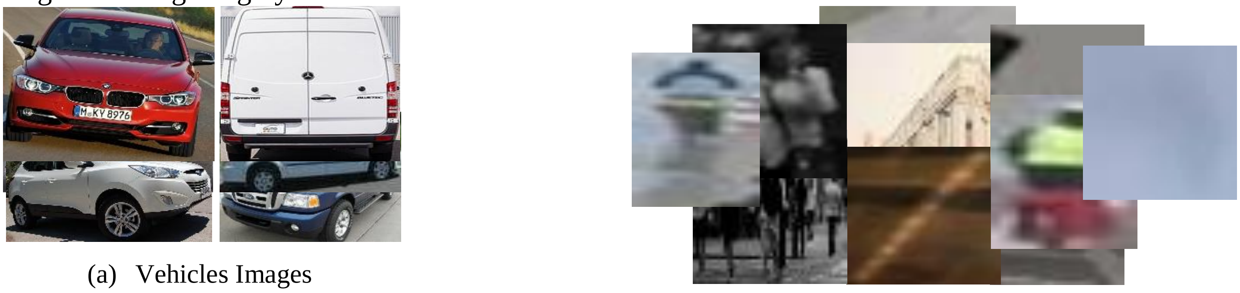

Training sets of instances
Training sets of instances
Feature Extraction by GLCM
Decision

Figure 9-The Classification System.

In the training phase for weather classification, the proposed system achieved an accuracy of around 99%. The system was tested on 42 videos for different weather conditions collected from different datasets, besides some videos captured in the real world. The proposed system accomplished an of accuracy about 95.23%, as calculated by the following accuracy equation:

$$Accuracy = \frac{total\ number\ of\ correctly\ classified\ videos}{total\ number\ of\ videos} \quad \ldots (19)$$

The dataset for training the vehicles classification model consisted of 2 classes ,“Vehicle” with 12111 RGB images and “Not vehicle” with 4111 RGB images, which were collected from Stanford Cars Dataset and GTI Dataset . Figure-10 demonstrates examples for Vehicles classification dataset. It should be noted that all the GLCM features are extracted from the weather dataset images after converting each image to gray level.



(a) Vehicles Images

(b) Not Vehicles Images

Figure 10-Vehicles and not vehicles dataset.

Samples of videos used to evaluate the proposed system are shown in Figure-11. In the training phase of vehicles classification, the proposed system achieved an accuracy of about 99%. The system tested 8 different videos of weather conditions collected from different datasets. The proposed system accomplished a high accuracy in normal and complex weather conditions, as demonstrat in Table 1.

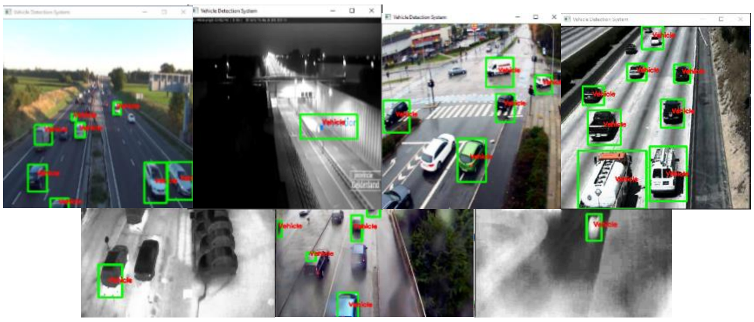


Figure 11-Video samples

Table 1- Videos Test Descriptions

Conditions	Total Vehicles No.	Correct Detection No.	Miss Detection No.	Not Vehicles Wrong Classified	Detection Accuracy Rate (%)
Cloudy Weather	**83**	**81**	**2**	**0**	**97.59**
Normal weather + Heavy Traffic	**485**	**480**	**5**	**6**	**98.96**
Normal Weather in Sunset.	**80**	**77**	**3**	**3**	**96.25**
Foggy Weather + Night	**25**	**25**	**0**	**3**	**100**
Low Rain Weather + Daytime	**237**	**229**	**8**	**25**	**96.62**
Snow Weather + Night	**57**	**50**	**7**	**4**	**87.7**
Snow Weather + Heavy Traffic + Night	**307**	**9**	**298**	**8**	**97.06**
Heavy Rain Weather + Daytime	**137**	**133**	**4**	**15**	**97.08**
Average Detection Rate (%)					**96.4**

9. Comparisons with Other Approaches

To demonstrate the robustness and accuracy of the proposed system, we compared its experimental results with those of other systems that were previously tested in different lighting and weather conditions, as given in Table-2:

Table 2- Comparison of the present results with other approaches

References Conditions	[10]	[4]	[2]	[9]	[7]	Our Proposed System
Normal Weather + Heavy Traffic	**98.9 %**	**97.5 %**	**93.34%**	**98.20 %**	**93.75 %**	**98.96 %**
Snow Weather	**83.4 %**	**N/A**	**N/A**	**98.46 %**	**N/A**	**97.06**
Cloudy Weather	**N/A**	**99.17 %**	**N/A**	**N/A**	**N/A**	**97.59**
Low + Heavy rain Weather	**N/A**	**93 %**	**81.32 %**	**92.12 %**	**N/A**	**L:96.62 % H:97.08%**
Normal Weather + Sunset	**N/A**	**N/A**	**96.48 %**	**N/A**	**N/A**	**96.25 %**
Foggy + Poor Light Condition (Night)	**N/A**	**93.7 %**	**77.63 %**	**96.34 %**	**94.74%**	**100 %**

10. Conclusions

This study suggested an adaptive vehicle detection system working in different weather conditions (rainy, cloudy, foggy, and dusty, at daytime and nighttime). Background subtraction step was applied based on the MOG 2 for the detection of moving objects in the sequence frames. The proposed system included the collection of pre-processing operations in order to enhance the accuracy of feature extraction and classification steps, such as Gaussian blur filter, histogram equalization, dilation, erosion, and threshold. The proposed system classified the kind of weather as well as vehicles from not vehicles, where the features should be extracted based on gray level co-occurrences matrix (GLCM). Two different classification models, namely Support Vector Machine Classifier and Random Forest Classifier) were utilized for the classification step.

The proposed system used GLCM as one of the important approaches in texture feature extraction. Due to the strength of the features obtained from this method, it was reliable to achieve the best results for determining the state of the weather utilizing RFC. It was also reliable to determine the detected object of a vehicle or not by SVMC.

The proposed system addressed the problem of difficult visibility in complex weather conditions by adapting the pre-processing operations to these conditions and adding a Histogram Equalization method to improve the contrast in the video. The proposed system changed adaptively the type and parameters of preprocessing operations, which depended on the classification results of the weather classifier model. In this paper, the importance of the proposed system was demonstrated, which was able to recognize the state of the weather to achieve the best results, when compared with similar previous systems.

The system was tested on eight manually collected realistic videos of different weather conditions. The experimental results presented accurate and robust performance in complex conditions, both with a shaking in the camera and in normal conditions. The strength of the suggested system was confirmed with a very challenging dataset, where it was appeared to be superior and highly adaptable, as characterized by the ability to achieve high accuracy with different weather conditions and lighting. Therefore, in future work, more methods such as vehicle tracking and counting can be implemented to obtain useful traffic information to make the system more generally applicable.

References

1. M. Won. **2020**.“Intelligent Traffic Monitoring Systems for Vehicle Classification: A Survey,” *IEEE Access*, **8**(9): 73340–73358. doi: 10.1109/ACCESS.2020.2987634.
2. Y. M. Chan, S. S. Huang, L. C. Fu, and P. Y. Hsiao. **2007**. “Vehicle detection under various lighting conditions by incorporating particle filter,” *IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC*, pp. 534–539, 2007, doi: 10.1109/ITSC.2007.4357745.
3. D. A. Padilla, J. F. Villaverde, J. J. T. Magdaraog, A. J. L. Oconer, and J. P. Ranjo. **2019**. “Vehicle and Weather Detection Using Real Time Image Processing Using Optical Flow and Color Histogram,” *2019 5th Int. Conf. Control. Autom. Robot. ICCAR 2019*, pp. 880–883, 2019, doi: 10.1109/ICCAR.2019.8813346.
4. B. F. Wu and J. H. Juang. **2012**.“Adaptive vehicle detector approach for complex environments,” *IEEE Trans. Intell. Transp. Syst.*, **13**(2): 817–827, 2012, doi: 10.1109/TITS.2011. 2181 366.
5. H. Yang and S. Qu. **2017**.“Real-time vehicle detection and counting in complex traffic scenes using background subtraction model with low-rank decomposition,” *IET Intell. Transp. Syst.*, **12**(1): 75–85, 2018, doi: 10.1049/iet-its..0047.
6. B. G. Rajagopal, N. Vishakraj, U. Kumar, and P. Jothivenkatesh. **2017**. “Vision-based system for counting of moving vehicles in different weather conditions,” *Proc. Int. Conf. Electron. Commun. Aerosp. Technol. ICECA 2017*, **2017-January**: 86–91, 2017, doi: 10.1109/ICECA.2017.8203649.
7. N. Y. Ershadi, J. M. Menéndez, and D. Jiménez,**2018**. “Robust vehicle detection in different weather conditions: Using MIPM,” *PLoS One*, **13**(3):1–30, doi: 10.1371/journal.pone.0191355.
8. N. Y. Ershadi, **2017**. “Improving vehicle tracking rate and speed estimation in dusty and snowy weather conditions with a vibrating camera,” *PLoS One*, **12**(12), doi: 10.1371/journal. pone. 0189145.
9. H. Y. Cheng, **2018**.“Highway Traffic Flow Estimation for Surveillance Scenes Damaged by Rain,” *IEEE Intell. Syst.*, **33**(1): 64–77, 2018, doi: 10.1109/MIS.2018.111144331.
10. T. Acharya and A. K. Ray, **2005**. *Image Processing: Principles and Applications*. 2005.

11. A. Suresh and K. L. Shunmuganathan, **2012**. “Image texture classification using gray level co-occurrence matrix based statistical features,” *Eur. J. Sci. Res.*, **75**(4): 591–597.

12. R. Asery, R. K. Sunkaria, L. D. Sharma, and A. Kumar,**2016**. “Fog detection using GLCM based features and SVM,” *Conf. Adv. Signal Process. CASP 2016*, pp. 72–76, 2016, doi: 10.1109/CASP.2016.7746140.

13. L. A. Marcomini and A. L. Cunha, **2018**. “A Comparison between Background Modelling Methods for Vehicle Segmentation in Highway Traffic Videos,” 2018, [Online]. Available: http://arxiv.org/abs/1810.02835.

14. C. Stauffer and W. E. L. Grimson,**1999**. “Adaptive background mixture models for real-time tracking,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, **2**: 246–252, 1999, doi: 10.1109/cvpr.1999.784637.

15. A. T. Azar, H. I. Elshazly, A. E. Hassanien, and A. M. Elkorany, **2014**.“A random forest classifier for lymph diseases,” *Comput. Methods Programs Biomed.*, **113**(2): 465–473, 2014, doi: 10.1016/j.cmpb.2013.11.004.

16. M. Pal, **2005**.“Random forest classifier for remote sensing classification,” *Int. J. Remote Sens.*, **26**(1): 217–222, 2005, doi: 10.1080/01431160412331269698.