Accelerated Fog Removal from Real Images for Car Detection

Rawan Ibrahim Khader Younis

Follow this and additional works at: https://scholarworks.uaeu.ac.ae/all_theses
Part of the Engineering Commons

Recommended Citation
https://scholarworks.uaeu.ac.ae/all_theses/725

This Thesis is brought to you for free and open access by the Electronic Theses and Dissertations at Scholarworks@UAEU. It has been accepted for inclusion in Theses by an authorized administrator of Scholarworks@UAEU. For more information, please contact fadl.musa@uaeu.ac.ae.
United Arab Emirates University

College of Engineering

Department of Electrical Engineering

ACCELERATED FOG REMOVAL FROM REAL IMAGES FOR CAR DETECTION

Rawan Ibrahim Khader Younis

This thesis is submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering

Under the Supervision of Dr. Nabil Bastaki

December 2015
Declaration of Original Work

I, Rawan Ibrahim Khader Younis, the undersigned, a graduate student at the United Arab Emirates University (UAEU), and the author of this thesis entitled “Accelerated Fog Removal from Real Images for Car Detection”, hereby, solemnly declare that this thesis is my own original research work that has been done and prepared by me under the supervision of Dr. Nabil Bastaki, in the College of Engineering at UAEU. This work has not previously been presented or published, or formed the basis for the award of any academic degree, diploma or a similar title at this or any other university. Any materials borrowed from other sources (whether published or unpublished) and relied upon or included in my thesis have been properly cited and acknowledged in accordance with appropriate academic conventions. I further declare that there is no potential conflict of interest with respect to the research, data collection, authorship, presentation and/or publication of this thesis.

Student’s Signature ___________________________ Date ________________
Advisory Committee

1) Advisor: Nabil Bastaki
Title: Assistant Professor
Department of Electrical Engineering
College of Engineering

2) Co-advisor: Rashad M. Ramzan
Title: Assistant Professor
Department of Electrical Engineering
College of Engineering
Approval of the Master Thesis

This Master Thesis is approved by the following Examining Committee Members:

1) Advisor (Committee Chair): Dr. Nabil Bastaki
   Title: Assistant Professor
   Department of Electrical Engineering
   College of Engineering

   Signature ___________________________    Date ____________

2) Member: Dr. Muhammad Rashad Ramzan
   Title: Assistant Professor
   Department of Electrical Engineering
   College of Engineering

   Signature ___________________________    Date ____________

3) Member: Dr. Qurban Ali Memon
   Title: Associate Professor
   Department of Electrical Engineering
   College of Engineering

   Signature ___________________________    Date ____________

4) Member (External Examiner):
   Title: 
   Department of …
   Institution:

   Signature ___________________________    Date ____________
This Master Thesis is accepted by:

Dean of the College of Engineering: Professor Mohsen Sherif

Signature ___________________________  Date ____________________

Dean of the College of the Graduate Studies: Professor Nagi T. Wakim

Signature ___________________________  Date ____________________

Copy ____ of ____
Abstract

Image de-hazing improves the visual quality of images in computer vision applications, such as object detection and object tracking. Fog removal from car photos taken by street cameras is considered essential to accurate car detection. An accelerated image enhancement technique is presented for car detection as part of an effort to count cars using existing street cameras for the purpose of traffic management. Two aspects of car detection are tackled: 1) An existing image fog removal technique is accelerated by replacing a time consuming image filter with a faster filter while maintaining negligible image degradation, 2) A quick and practical algorithm to detect a car in a fog-free image is proposed and applied to a database of about 100 car images. The main idea is to give an indication of the capacity of cars on a given road by counting them using street cameras in the presence of fog. Such car counting method can assist traffic centers to manage traffic flow and prevent traffic incidents. The devastating effect of fog-related accidents inspired this research to develop a fast execution-time algorithm to detect cars in the presence of heavy fog using existing road cameras. Acceleration is the main goal of this research, in addition to car detection accuracy. In order to achieve the required acceleration and accuracy, several image processing techniques are investigated. The techniques are proposed to accelerate fog removal from car images and accurately detect cars with an execution time, faster than any other existing fog removal and car detection technique. Therefore, the developed techniques provide a viable solution to a difficult problem in the area of intelligent transportation systems. The improved fog removal technique is performed by estimating the transmission map using the Proposed Adaptive Filter (PAF) to recover the scene depth of the foggy image. After filtering, a simple, yet exact and effective, car detection algorithm is executed to confirm the presence or absence of a car in the processed image. The system is fairly robust and although all images were obtained from existing sources, the proposed algorithm is expected to perform equally well with any side-view image of a car in the presence of heavy fog and under real conditions.

**Keywords:** Image processing, car detection, Sobel operator, fog removal, proposed adaptive filter, dark channel prior.
المستوى

تسريع إزالة الضباب من صور واقعية للكشف عن وجود سيارة

المستوى

تعتبر إزالة الضباب من صور السيارات بكاميرات الشوارع ضرورية لكشف السيارات بشكل دقيق. نقدم تقنية محسنة وسرعية تحدد وجود سيارة كجزء من عدة السيارات باستخدام كاميرات الشوارع الموجودة لغرض إدارة حركة المرور. وتتناول جانبين من جوانب الكشف عن السيارة: 1) هو تسريع في تقنية إزالة صورة الضباب الموجودة عن طريق تقنية صورة أسرع مع تدحر لا يكاد يذكر في الصورة، (2) خوارزمية سريعة وعملية للكشف عن سيارة في وجود ضباب وتطبيقها على قاعدة بيانات نحو 100 صورة لسيارات. والتقنية الرئيسية هي لتطبيق موشياً لقدرة السيارات على الطريق ومعرفة عددها باستخدام كاميرات الشارع في وجود الضباب. هذه الطريقة تساعد مراكز الحركة لإدارة حركة السير ومنع حوادث المرور. الأثر المدمر لحوادث السيارة ألهمت هذا البحث لتطوير خوارزمية التنفيذ في وقت سريع وقبح للكشف عن السيارات في وجود ضباب كثيف باستخدام كاميرات الطرق الثابتة. التسارع هو الهدف الرئيسي من هذا البحث، بالإضافة إلى دقة الكشف عن وجود السيارة. نركز على تقنية مطورة تعمل على تحسين وقت التنفيذ للمستويات الحالية يتم فيها تحقيق السرعة المطلوبة ودقة معالجة الصور الرقمية. وباختصار، نقترح في هذه الأطروحة معالجة للصور الرقمية مع إمكانية الكشف عن وجود سيارة على حد سواء باستخدام المحاكاة لتسريع إزالة الضباب ودقة الكشف عن السيارات مع وقت تنفيذ أسرع تحتاج التقنية الموجودة لإزالة الضباب و الكشف عن جود السيارة. و لذلك فإن التقنيات المعروضة توفر حلاً قوياً لتطبيق لمشكلة صعبة في مجال أنظمة التحكم الذكية. يتم تصميم نظام الكشف عن سيارة لقراءة الصور من قاعدة بيانات موجودة تتألف من صورة السيارة الجانبية. بدأ النظام بأكمله مع تدحر الصور عن طريق معاييرهم مع الآثار البيئية التي تظهر الضباب. بعد ذلك، يتم استخدام الصور التي تم تدحرها لإختبار الخوارزميات المقترحة. يتم تنفيذ تحسين تقنية إزالة الضباب عن طريق تدحر خريطة إنتقال باستخدام تقنيات التكيفية لاستمرار عمق المشهد من صورة ضبابية. بعد التشخيص، يتم تنفيذ خوارزمية الكشف عن سيارة بسبيطة، لكنها دقيقة وفعالة للتأكد من وجود أو عدم وجود سيارة في الصورة المحسنة. كون النظام قوياً
إلى حد ما، على الرغم من أنه تم الحصول على جميع الصور من المصادر الحالية، فإن من المتوقع أن الخوارزمية المقترحة تقدم أداء جيد على قدم المساواة مع أي صورة سيارة جانبية في وجود ضباب كثيف وتحت ظروف حقيقية.

مفهوم البحث الرئيسية: معالجة الصور، إزالة الضباب، الكشف عن سيارة جانبية، الكشف عن الحدود، تبسيط عامل الظلام السابق، ترشيح إزالة الضباب التكيفية.
Acknowledgements

Many thanks go to all faculty members of the Department of Electrical Engineering at the United Arab Emirates University for their continuous support and encouragement. I, especially, would like to express my sincere appreciation for my thesis supervisor Dr. Nabil Bastaki for his exceptional support, friendless and insightful guidance, providing me with valuable understanding and advice during the whole project stages.

My special thanks go to my parents, brothers, and sisters who helped me along the way. I am sure they suspected it was endless. My deepest thanks go to my small family, husband and daughters, who supported me with all facilities, gave strength and encouragement to complete this work.
Dedication

To my beloved parents and family
# Table of Contents

Title ........................................................................................................................................... i
Declaration of Original Work ....................................................................................................... ii
Copyright ....................................................................................................................................... iii
Advisory Committee ................................................................................................................... iv
Approval of the Master Thesis ................................................................................................. v
Abstract ........................................................................................................................................ vii
Title and Abstract (in Arabic) ............................................................................................... viii
Acknowledgements .................................................................................................................. x
Dedication .................................................................................................................................... xi
Table of Contents .................................................................................................................. xii
List of Tables ........................................................................................................................ xiv
List of Figures ........................................................................................................................ xv
List of Abbreviations ................................................................................................................ xvi

Chapter 1: Introduction .............................................................................................................. 1
  1.1 Thesis Overview ........................................................................................................... 4
  1.2 Problem Statement ................................................................................................................ 6
  1.3 Relevant Literature .............................................................................................................. 8
    1.3.1 Fog removal techniques ................................................................................................. 8
    1.3.2 Edge detection ................................................................................................................ 22
    1.3.3 Circle detection ............................................................................................................. 25
    1.3.4 Car detection ................................................................................................................ 26

Chapter 2: Methods .................................................................................................................. 30
  2.1 Research Design .................................................................................................................. 30
  2.2 Design Challenges ............................................................................................................. 30
  2.3 Expected Impact ................................................................................................................ 31
  2.4 The proposed method ........................................................................................................ 32
    2.4.1 Fog removal approach ................................................................................................. 33
    2.4.2 Edge detection ............................................................................................................ 37
    2.4.3 Circle detection ............................................................................................................ 38
    2.4.4 Car detection .............................................................................................................. 39
  2.5 Data collection ................................................................................................................... 41

Chapter 3: Results ..................................................................................................................... 42
  3.1 Profile and Statistics of Respondents ............................................................................... 42
  3.2 Reliability of Individual Influence Scales ........................................................................... 56
Chapter 4: Discussion ........................................................................................................ 69
  4.1 Fog removal ........................................................................................................ 69
  4.2 Car detection ...................................................................................................... 73
Chapter 5: Conclusion .................................................................................................... 77
Bibliography .................................................................................................................. 78
List of Tables

Table 1: Detection rate of different car detection algorithms .......................... 29
Table 2: Execution time (in seconds) comparison among different filters.......... 43
Table 3: Executed time comparison between existing and the proposed algorithms .. 46
Table 4: Comparison of detection rate for different applied car detection algorithms ................................................................................................................. 74
Table 5: Detection rate under different conditions .................................................. 76
Table 6: Car detection rate of images in the UIUC with various fog densities ....... 76
List of Figures

Figure 1: Gx and Gy are gradient operator ................................................................. 23
Figure 2: Image strip feature vs. car structure .......................................................... 27
Figure 3: System architecture .................................................................................... 32
Figure 4: The template of the side car (red part is to be detected) ............................. 39
Figure 5: Comparison of execution time of various applied filters for SDCP ............. 44
Figure 6: Example-1 applied Proposed Adaptive Filter results ................................. 47
Figure 7: Example-2 applied Proposed Adaptive Filter results ............................... 48
Figure 8: Falsely detected cars example .................................................................... 49
Figure 9: Normal image car detection .......................................................................... 50
Figure 10: Critical image car detection ....................................................................... 51
Figure 11: Impossible image – no car detection ......................................................... 52
Figure 12: Applied circle and car detection algorithms .............................................. 52
Figure 13: Detection of three cars .............................................................................. 54
Figure 14: Normal image car detection ...................................................................... 54
Figure 15: Car detection for critical image ................................................................. 55
Figure 16: Car detection for impossible image ............................................................ 55
Figure 17: Car detection for two types of car .............................................................. 56
Figure 18: Dark channel output .................................................................................. 57
Figure 19: Comparison between the outputs using DCP and SDCP .......................... 58
Figure 20: Transmission maps ................................................................................... 58
Figure 21: Outputs of Sobel Operator ........................................................................ 59
Figure 22: Example-1, applied proposed methods to real foggy images from UAE. 62
Figure 23: Example-2, applied proposed methods to real foggy images from UAE. 64
Figure 24: Example-3, applied proposed methods to real foggy image ...................... 66
Figure 25: Summary of the two stages in this system ................................................. 69
Figure 26: Fog removal process .................................................................................. 71
Figure 27: Flow of Simplified Dark Channel Prior. ..................................................... 71
## List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCP</td>
<td>Dark Channel Prior</td>
</tr>
<tr>
<td>SDCP</td>
<td>Simplified Dark Channel Prior</td>
</tr>
<tr>
<td>HT</td>
<td>Hough Transform</td>
</tr>
<tr>
<td>CHT</td>
<td>Circular Hough Transform</td>
</tr>
<tr>
<td>DIP</td>
<td>Digital Image Processing</td>
</tr>
<tr>
<td>MRF</td>
<td>Markov Random Field</td>
</tr>
<tr>
<td>RGB</td>
<td>Red, Green, Blue</td>
</tr>
<tr>
<td>BGIF</td>
<td>Basic Guided Image Filter</td>
</tr>
<tr>
<td>PAF</td>
<td>Proposed Adaptive Filter</td>
</tr>
<tr>
<td>OEA</td>
<td>Object Extraction Algorithm</td>
</tr>
<tr>
<td>LLF</td>
<td>Local Laplacian Filter</td>
</tr>
<tr>
<td>WGIF</td>
<td>Weighted Guided Image Filter</td>
</tr>
<tr>
<td>GD</td>
<td>Gaussian Distribution</td>
</tr>
<tr>
<td>CCD</td>
<td>Charge-Coupled Device</td>
</tr>
</tbody>
</table>
Chapter 1: Introduction

Image de-hazing improves the visual quality of images in computer vision applications, such as object detection and object tracking; however, haze removal is a challenging problem because of the significant difference between the haze and the unknown scene depth [3].

Haze (fog, mist, dust and other atmospheric phenomena) is a main recession of outdoor images, by affecting both color and contrast. The general definition of fog is a collection of suspended water droplets or ice crystals near the Earth's surface [5].

Fog reduces visibility to less than 1 km [6], and in some cases, to 50 meters or less. Moreover, fog also deforms visual perception, limits contrast and causes many car accidents each year. Since fog is made of very small water particles suspended in the air, it causes the incident light to scatter after hitting the water particles. When this happens, it leads to loss of contrast and the formation of dense white background [70]. Furthermore, as the water particles become smaller and fog becomes thicker, the fog creates a blanket that covers roadways. Such a blanket can be a cause of many traffic accidents each year.

Fog also affects images when they are captured in such foggy weather conditions with poor contrast. As a result, several attempts have been made to device various computer vision algorithms in order to remove fog from images. Under bad weather conditions, the light reaching a camera is severely scattered by the atmosphere. Moreover, the formation of fog is a function of depth; and hence, removal of fog requires several assumptions or prior knowledge about the captured scene. Fog removal algorithms estimate the depth information under various assumptions as will be discussed for the Dark Channel Prior (DCP) technique. The
importance of fog removal algorithm is due to its wide application in tracking and navigation, consumer electronics, and entertainment industries [8].

In this work, Dark Channel Prior (DCP) is used as a base line for the proposed fog removal algorithm. Initially, the transmission map is defined using the DCP technique; subsequently, the transmission map is refined with the aid of a Simplified Dark Channel Prior (SDCP) using a set of filters consisting of the Proposed Adaptive Filter and an edge-preserving filter. Next, the refined transmission map is used to modify the scene radiance as the fog is removed. Although, the Dark Channel Prior is an efficient method to remove fog and enhance image contrast, it suffers from lengthy execution time, computational complexity and large memory requirement. The fog removal method proposed here, is fast and has negligible image degradation when used as input for the subsequent stage of car detection. The quality of the fog-free image is observed using the Sobel operator to ensure that during the enhancement process, the edges and the main features of the car are clearly visible and are not adversely affected, in order to be able to properly detect the edges and circles.

Once the process of fog removal is complete, an edge detection algorithm is performed on the image, which is followed by a simplified car detection algorithm. The resultant image is clear with distinctively visible and detectable edges. The next step is to apply the proposed and simplified car detection algorithm. In this thesis, the main idea behind the car detection algorithm is to detect the wheels that are horizontally aligned within a few pixels. The algorithm is simple, exact and fast at detecting a car in an image when compared to the technique proposed by Tan, Li, Cai, and Zhang in [23]. In [23], a template is used that consists of mainly wheel positions, horizontal and oblique lines of a car. In [23], strict assumptions had to be
made to detect the wheels, e.g., they have to be of the same size, exist on the same horizontal line, and not at the top of the image with an approximate width and height of a car. The height was assumed to be about two-third and the width was assumed to be about three-fifth of the distance between the two wheels. It is important to state that this method cannot detect cars with oval shaped roof.

The work proposed in this thesis, is somewhat similar to the method used in [23], however, the proposed work is based on two constraints only, the location and size of the wheels. The proposed car detection algorithm either indicates the presence or absence of a car in the image. A pseudo code of the proposed car detection algorithm can be found in chapter 2.

The first chapter of this thesis introduces the main idea behind this research. An overview of the thesis is presented, followed by the problem statement, literature review and background information on various relevant image-processing algorithms. Chapter two mainly covers the proposed car detection methodology. Results produced on a sample database of images are displayed in Chapter 3, while the results are discussed in Chapter 4. The conclusion and possible future work are covered in Chapter 5.
1.1 Thesis Overview

Digital image processing (DIP) performs automatic processing, manipulation and interpretation of visual information, and plays an important role in many aspects of our daily life, as well as in a wide variety of disciplines and fields in science and technology, with applications such as television, photography, robotics, remote sensing, medical diagnosis, finger print identification and industrial inspection. To implement sophisticated DIP algorithms and to process large amounts of data captured from sources such as speed radars and traffic cameras, intelligent high-speed real-time systems are becoming imperative.

The purpose of this research is to detect cars from real images in the presence of heavy fog. As a matter of fact, on Thursday morning 8th, January, 2015, on Abu Dhabi – Dubai Road, multi-vehicle collisions were reported with 114 vehicles pile-up and 20 people injured due to a thick blanket of fog that had significantly reduced visibility and was the main cause of the accident [1]. The incident was one of many that occur across various parts of UAE, especially during fall, winter and spring. Many of UAE residents live in the northern cities and commute on a daily basis to either the capital city – Abu Dhabi – or Dubai. The heavy traffic – even without fog – is, by itself, a big factor in the numerous accidents that occur each year. The traffic situation becomes much worse in the presence of fog, especially highways, where the speed is relatively higher and roads are void of traffic lights. An English language multi-platform news organization, “The National” [2], has published a report of fog-related road accidents across the UAE. The report gives us an indication of how much trouble fog creates for the drivers each year. The devastating effect of fog-related accidents inspired this research to develop a fast algorithm to detect cars in
the presence of heavy fog. For practical reasons, a database of existing car photos was obtained from various online sources, such as the UIUC Image Database for Car Detection [17]. Only real car images with side-view were considered in the database. Fog was added to all images using Gaussian distribution with randomly selected mean ranging from 0.1 to 0.4 and randomly chosen variance ranging from 0.001 to 0.01 per image, in the presence of all original image components, such as the car itself, polls and trees. Of course, neither the cars nor the backgrounds are identical across the images in the database. The car detection system starts by reading the foggy image, and then removing the fog and finally, detecting or not detecting the presence of a car. The system is fairly robust and even though all images were obtained from existing sources, the system is expected to perform equally well with any side-view image of a car in the presence of fog and under real conditions.

The car detection system is designed to read images from an existing database consisting of side-view car images. The entire system starts with modifying the images by treating them with environmental effects that appear as fog. Next, the modified images are used to test the proposed algorithms. The improved fog removal technique is performed by estimating the transmission map using an Adaptive Filter to recover the scene depth of the foggy image. After filtering, a simple, yet exact and effective, car detection algorithm is executed to confirm the presence or absence of a car in the processed image.

In the following sections, the problem statement and the literature review will be presented including fog removal techniques, along with the adopted and modified circle and car detection methodology.
1.2 Problem Statement

In the United Arab Emirates, fog is almost an everyday problem for drivers in winter, especially on highways. As an example, last year, thick fog led to 57-car pile-up on both sides of the motorway between Abu-Dhabi and Al-Ain and 14 people were injured on January 16, 2014 [2]. The fog density or thickness significantly varies from time to time and from spot to spot, even on the same road. For drivers, it is crucial to know the visibility in the presence of fog, which is usually given in meters. A visibility of less than 50 meters is considered very extreme and in such conditions, it is highly recommended not to drive, or drive while being extremely cautious. Although, drivers are recommended not to drive under extremely foggy conditions, it is not practical for many drivers who have to commute on a daily basis.

Assuming the existence of traffic cameras at various points on highways and roads, the piling of cars or even an indication of the number of cars, can give a clue to the presence of a traffic accident or, in general, some type of an incident such as a stalled car. This thesis is mainly concerned with detecting cars from side-view images of cars on the road in the presence of fog, so that it can be used in counting the numbers of cars for purposes of traffic load or incident detection. Although, non-camera-based, i.e., counter based, methods are also used in some places, the focus of this thesis is on the use of existing traffic cameras.

As a result of this work, an indication of the capacity of cars on any camera monitored road can be obtained and thence, a decision can be made to take the necessary action to either prevent accidents or be quickly informed of existing accidents under poor visibility conditions. This can provide valuable information to traffic centers across the UAE and in other countries that suffer from similar
problems. By using existing traffic cameras, no additional spending in the road infrastructure is needed. It is the objective of this research to accelerate the car detection, in order, to make it suitable for real-time implementations. There are existing techniques that tackle similar problems and hence, the proposed methodology, will be an improvement over all existing ones. The methodology is based on image processing techniques that can be divided into two stages, namely fog removal and car detection.

The methodology is based on removal of existing fog in a short time as opposed to other fog removal techniques, that are relatively time consuming, when realistic fog models are used. Once fog is removed, circles present in the image are determined, and finally using the coordinates of the circles and the range of diameter of wheels, the presence or absence of a car is confirmed. The methodology is developed to accelerate existing image processing techniques in the area of intelligent transportation systems. Furthermore, the system is designed to read images from an existing database where fog has been superimposed using the best available mathematical fog model. During processing, images are retrieved from the database and are processed in the presence of environmental effects, and the presence or absence of a car is correctly indicated in a short period of time.

Typically, such problems fall under traffic management. The traffic detection system provides the traffic center with traffic data which is obtained from the car detection algorithm proposed in this research and that data leads to find the capacity of cars in exact time which is necessary for deciding of the presence or absence of incidents.
1.3 Relevant Literature

This section presents a few fog removal techniques that are currently used along with the description of the dark channel prior method. Furthermore, the filter needed to complete this work is highlighted. In addition, several edge and circle detection approaches will be reviewed in order to give a good background on the subject to be explored further in this research.

1.3.1 Fog removal techniques

Fog, haze and smoke are a major factor in road accidents. Fog affects the visual quality of the image by reducing the contrast level of the image. Comparing a foggy image with a clear day one, the later has more contrast. Therefore, a fog removal algorithm should improve the scene contrast, which will lead to better car detection [18].

In computer vision and image processing, physical models are used to forecast the pattern of image degradation information. Fog is a combination of two components air-light and direct attenuation. Light from the atmosphere and light reflected from an object are scattered by the water droplets, resulting in the degradation of image quality [20, 70].

In 1924, Koschmieder [9] proposed his theory on the apparent luminance of objects observed against sky background on the horizon. Accordingly, the formation of a haze image is written as follows:

\[ I(x) = J(x) t(x) + A(1 - t(x)) \]  \hspace{1cm} (1)

where I is the observed intensity, J is the scene radiation, A is the global atmospheric
light, and \( t \) is the medium transmission describing the portion of the light that is not scattered and reaches the camera. The goal of haze removal is to recover \( J \), \( A \), and \( t \) from \( I \). While the atmospheric light \( A \) can be estimated using the sky area in the original image, two unknowns remain [67]. For this reason, prior knowledge must be used in single image haze removal based on the atmospheric scattering model. Other fog removal approaches will be presented for comparison purposes.

Tan [24] assumes that the contrast of a haze-free image is much higher than that of a hazy image based on statistical information. In the beginning, Tan preprocesses the original image using white balance algorithm. Then, the contrast of the image is measured using the quantity of visible edges. Tan observed that the quantity of visible edges of the haze removed image initially increases and later it decreases. With the aim of obtaining the air-light \( A(x) \), Tan used Markov Random Field (MRF) under the assumption that sudden changes of depth rarely occur to the scene and the \( A(x) \) is changing smoothly in local areas of the image. Subsequently, to find the value of \( A(x) \), Tan maximized the probability distribution of \( A(x) \) by using graph-cut. This method performs well to enhance the contrast and recover the scene feature. In spite of the fact that the processed image is not physically valid but over-saturated, there are halo effects in the areas in which depth changes frequently.

Fattal [25] divides the unknown haze removed image \( J(x) \) into two components: the surface albedo coefficients \( R \), which is a three-channel RGB vector, and \( l \) is a scalar describing the light reflected from the surface. Mainly, this method is physically valid and the visual effects of the results is natural. However, this method is invalid when used with gray scale images due to the basis of this method depending on the statistical property of color images. Besides, the statistical property is invalid in areas with heavy haze and low signal to noise ratio.
He Sun and Tang [3] found a prior named Dark Channel Prior by studying the statistical characteristics of a large set of haze-free outdoor images. The dark channel prior is based on the fact that very often some pixels have very low light intensity, some even tend to be zero, in one or more color channels in most of the local regions which do not cover the sky. Additionally, they refine the transmission map by using a soft matting algorithm. Single image haze removal using dark channel prior is simple and has been proved to be physically valid. Moreover, guided filtering was used instead of soft matting to refine rough transmissions. This improvement makes the haze removal method much more efficient with almost the same image quality. Haze removal based on dark channel prior is simple and effective, but the contrast enhancement of the algorithm is not quite enough and the dark channel prior is invalid when the scene objects are inherently similar to the atmospheric light.

Fang et al. [29] improved the transmission refinement based on the differences between pixels in RGB space. They overcame the block effects using dark channel prior by a variant method using the rough transmission as the initial estimate.

Matlin and Milanfar [30] added a de-noising process in the haze removal using the BM3D algorithm proposed by Dabov et al. [31] as the preprocessing and, for robustness, an iterative kernel regression process was used as a post-processing step.

Ding and Tong [32] indicated that it is unnecessary to solve the soft matting problem at pixel level, but rather the smooth area of the image only needs to be process at low resolution. The transmission refinement uses the soft matting, where
rough transmission map would be segmented iteratively using an adaptively subdivided quad-tree based on the gradient within the corresponding domain and a user-given threshold.

Similarly, Zhu et al. [33] segmented the input image using the watershed image segmentation method before the soft matting process. The improvement made the haze removal method more flexible and reduced the complexity of the transmission refinement using soft matting, however, the precision of the watershed image segmentation method was reduced in the regions with dense haze.

Pei and Lee [34] proposed an improvement to increase the robustness of dark channel prior haze removal method to be used for nighttime haze removal using the color transfer algorithm proposed by Reinhard et al. [35] and the bilateral filter in local contrast correction algorithm proposed by Schettini et al. [36].

Yu and Liao [37] improved the estimation of the atmospheric veil $V(x)$ using bilateral filtering. The method was shown to be very efficient because the complexity of the method was linearly correlated with the number of the pixels in the image.

Xiao and Gan [38] proposed an improvement on rough transmission refinement using guided joint bilateral filtering, which was based on the bilateral filtering proposed by Tomasi and Manduchi [39] and the joint bilateral filtering proposed in [40,41]. The modifications improved the edge-preserving capability of the haze removal method.

In [19], Nayer and Narasimhan focused on building up the dichromatic atmospheric scattering model and then obtained a three-dimensional model of the scene object by analyzing the air-light in two different haze density images of the
same scene object. In [26], the atmospheric scattering model was extended to be chromatic. Narasimhan and Nayar’s method [19] required user interaction, who need to indicate the sky area or the densest haze area and point out the maximum and minimum depth area from the hazy image. The haze removal method with user interaction is effective in single image haze removal and leads to significant image enhancement.

Kopf et al. [27] obtained the depth information by analyzing the three-dimensional texture of outdoor scene objects. It was followed by further study of depth estimation using two weather conditions, scene structure from two weather conditions and contrast restoration using scene structure were made. These methods were able to obtain the depth information effectively and usually had excellent performance with haze removal. However, methods that need extra information or instruments and aim at particular situations, are cannot be widely used in various computer vision systems.

Guo et al. [69] proposed an image de-hazing method based on neighborhood similarity dark channel prior by He et al. [3]. After obtaining the transmission map, the difference between the dark channel and the dark value of nearest eight pixels were evaluated and the pixel of minimal difference was refined as used a the (new) dark channel.

1.3.1.1 Dark channel prior

He et al. [3] proposed an effective "dark channel prior" to remove haze from a single input image. The dark channel prior makes use of statistical information of outdoor haze-free images. Mainly, this technique is based on a key observation, that the most local patches in outdoor haze-free images contain a few pixels whose
intensity is very low in at least one color channel. Thus, haze thickness estimation using dark channel prior with haze imaging model lead to recovering high quality haze free images. Furthermore, a high quality depth map could also be obtained as a by-product of haze removal.

He et al. were able to find that air-light is the main contributor to the intensity of the dark pixels in one channel. According, they were able to estimate the transmission map of the haze. Combining a haze imaging model and a soft matting interpolation method, they were able to recover a high-quality haze-free image and produce a good depth map. Moreover, this lead to the need for finding a different filter that could give the transmission map faster as compared to the soft matting followed by producing the depth scene of the desired output.

The approach makes use of strong assumptions, and hence, the approach also suffers from certain limitations. It can be said that depending on the assumptions, the dark channel prior will sometimes be invalid, especially, when the scene object is naturally similar to the air-light over a large local region when no shadow is cast on the object. Although, the approach worked well, it may fail in some extreme cases.

Using the dark channel prior [3], and based on the observation on haze-free outdoor images, there will exist at least one color channel with very low intensity at some pixels. Formally, for an image $J$, they defined:

$$J_{\text{dark}}(x) = \min_{c \in \{r,g,b\}} \left( \min_{y \in \Omega(x)} (J^c(y)) \right)$$  \hspace{1cm} (2)

where $J^c$ is a color channel of $J$ and $\Omega(x)$ is a local patch centered at pixel $x$.

Their observation states that, except for the sky region, the intensity of $J_{\text{dark}}$ is low and tends to be zero, if $J$ is a haze-free outdoor image. They call $J_{\text{dark}}$ the dark
channel of J, and they call the above statistical observation or knowledge the dark channel prior.

Due to the additive nature of air-light, a hazy image is brighter than its haze-free version where the transmission t is low. Therefore, the dark channel of the hazy image will have higher intensity in regions with denser haze. Visually, the intensity of the dark channel is a rough approximation of the haze thickness. Moreover, they used this property to estimate the transmission and the atmospheric light.

1.3.1.2 Estimating the Transmission map

Firstly, assuming that the atmospheric light A is given. He et al. [3] also assumed that the transmission in a local patch Ω(x) is constant. They denoted the patch’s transmission as \( \tilde{t}(x) \). Taking the min operation in the local patch of the haze imaging to Equation (1), as follows:

\[
\min_{y \in \Omega(x)} (I^c(y)) = t^-(x) \min_{y \in \Omega(x)} (I^c(y)) + (1 - t^-(x))A^c
\]  

(3)

According to the dark channel prior, the dark channel \( J^{\text{dark}} \) of the haze-free radiance J should tend to be zero and as \( A^c \) is always positive, this leads to:

\[
\min_{c} \left( \min_{y \in \Omega(x)} \left( \frac{I^c(y)}{A^c} \right) \right) = 0
\]

(4)

By substituting and simplifying, Also knowing the atmospheric light, they can estimate the transmission \( \tilde{t}^\cdot \):

\[
t^-(x) = 1 - \min_{c} \left( \min_{y \in \Omega(x)} \left( \frac{I^c(y)}{A^c} \right) \right)
\]

(5)

In practice, even in clear days, the atmosphere is not absolutely free of
particles. So, the haze still exists when they look at distant objects with this assumption, they introduce a constant parameter $\omega$ ($0 < \omega \leq 1$) into Equation (5):

$$t^\sim(x) = 1 - w \min_c \left( \min_{y \in \Pi(x)} \frac{I^c(y)}{A^c} \right)$$

(6)

The nice property of this modification is that they adaptively kept more haze for the distant objects. The value of $\omega$ is application-based. They fixed it to 0.95 for all results.

Transmission map is used in many other algorithms in addition to dark channel prior. Depth map is also estimated with the help of transmission map function.

Various techniques were suggested to refine the transmission map; for example, soft matting and guided joint bilateral filter. These techniques were applied to the transmission maps of the original foggy images where – in most cases – several operations were needed to achieve a good result, however, with demanding computational processing power. For image haze removal, the time complexity is a critical problem that needs improvement, as the algorithm would be impracticable with high time complexity.

Moreover, in this research, several filters were used to estimate the new transmission map, in order, to reduce the execution time required for the soft matting while maintaining good quality fog-free output images.

In the following section, the different filters along with their properties will be explained.
1.3.1.3 Soft Matting

The image matting equation is formed below with similarity to the haze imaging equation (1):

\[ I = F \alpha + B (1 - \alpha) \]  \hspace{1cm} (7)

where F and B are foreground and background colors, respectively, and \( \alpha \) is the foreground opacity. A transmission map in the haze imaging equation is exactly an alpha map. The derivation of the matting Laplacian matrix in [28] was based on a color line assumption: The foreground/background colors in a small local patch lie on a single line in the RGB color space.

1.3.1.4 Restoration Model

Inverse of Kosmider’s law are also used for restoration process [42]. The transmission map \( \hat{t}(x) \) is then refined to \( t(x) \) using soft matting. With the refined transmission map \( t(x) \), the recovered image \( J(x) \) can then be estimated:

\[ J(x) = \frac{I(x) - A}{\max(t(x), t_0)} + A \]  \hspace{1cm} (8)

\( t_0 \) is a lower limit in case the transmission map \( t(x) \) is close to zero.

The key problem of single image based de-hazing methods is exploring a prior, with constraints on the scene for the purpose of scene structure estimation. However, these methods usually adopt some type of optimization or filtering method to obtain precise scene estimation. Moreover, optimization based de-hazing algorithms are usually time-consuming and unsuitable for real-time applications. In addition, Tarel’s methods tend to have halo effects due to the usage of median filters [43], [44]. To remove halo effects in Tarel’s methods, the bilateral filter was adopted in haze removal algorithms [45], even though the bilateral filter is computationally
expensive and suffers from memory inefficiency. Tarel and Hautiere [43] used the median filter instead of the minimum filter used in [46] under the assumption that the atmospheric light is smooth in local regions of the image. This algorithm significantly simplifies the processing and improves the efficiency while the haze removal quality is only negligibly decreased. However, median filter has no edge-preserving abilities. In addition, the parameters of the algorithm in [46], have to be adjusted manually based on the scene objects.

Moreover, He et al. [3] proposed a novel prior (DCP) haze removal, which is based on the statistics of outdoor haze-free images. Combining a haze imaging model and a soft matting interpolation method, they were able to recover a high-quality haze-free image. The success of these methods lies in using a stronger prior or assumption.

Xiao and Gan [50] obtained an initial atmospheric scattering light through median filtering, and then refined it using guided joint bilateral filtering to generate a new atmospheric veil which removes the abundant texture information and recovers the depth edge information.

Common drawbacks of all the methods above are their computational complexity and storage cost. To alleviate these drawbacks, the proposed technique in this thesis, can run faster than all of the above methods while preserving edges present in the processed image.

**Wiener filtering**

Wiener filtering [47] is based on dark channel prior which responds to color distortion while utilizing dark channel prior. The approximation of media function was rough which made halo effects in the final image. Thus, median filtering is utilized to approximate the media function to preserve edges. In order to have a
better median function, it is united with wiener filtering which transforms the image restoration problem into an optimization problem.

**Bilateral filtering**

The filtering technique in [48, 49, 68] performs image smoothing without affecting edges, by means of a non-linear grouping of nearby image values. The filtering technique replaces every pixel with a weighted average of its neighbor's pixel. This conserves sharp edges by methodically looping through each pixel and adjusting weights to the adjacent pixels accordingly. The bilateral filter (BF) is widely used due to its simplicity. However, the BF could suffer from “gradient reversal” artifacts [49], which refer to the artifacts of unwanted sharpening of edges despite its popularity. Furthermore, the results may exhibit undesired profiles around edges, usually observed in detail enhancement of conventional low dynamic range images or tone mapping of high dynamic range images.

**Guided Image Filter (GIF)**

Guided filter is derived from a local linear model. The filtering output is computed by the guided filter while considering the content of a guidance image. Moreover, the guided filter naturally is a fast and non-approximate linear time algorithm, regardless of the kernel size and the intensity range. In [56], a general linear translation-variant filtering process was defined, which involves a guidance image G, a filtering input image I, and an output image Q. Both G and I are given beforehand, and they can have the same scalar values. The filtering output at pixel i is expressed as a weighted average:

\[ Q = \sum_j W_{ij} (G)I_j \]  

(9)
Here i and j are pixel indexes. The filter weight $W_{ij}$ depends on the guidance image G and is explicitly expressed by

$$W_{ij} = \frac{1}{|w|^2} \sum_{k(i,j) \in w_k} \left( 1 + \frac{(G_i - \mu_k)(G_j - \mu_k)}{\sigma_k^2 + \varepsilon} \right)$$  \hspace{1cm} (10)

Here, $w_k$ is a square window centered at pixel k. $\mu_k$ and $\sigma_k^2$ are the mean and variance of G in $w_k$. $|w|$ is the number of pixels in $w_k$. $\varepsilon$ is a regularization parameter.

Moreover, He et al. [56] proposed this guided filter, in order, to reduce the execution time required for the soft matting while maintaining good quality fog-free output images. Li et al. [54, 64] introduced Weighted Guided Image Filter (WGIF) by adding an edge-aware weighting into an existing Guided Image Filter [56] to solve the halo artifacts that are unavoidable using GIF. In comparison, Li et al. [65, 66, 71] came out with a content adaptive GIF that is derived from normalized local variance of a guidance image into an existing guided filter.

In this thesis, a similar approach is followed to enhance the image quality and to restore it correctly, using an applicable filter. In addition, in this work, the target is to develop a new filter with advantages similar to the bilateral filter, e.g., edge-preserving filtering and non-iterative, while – at the same time – reducing the execution time and without any gradient distortion. The proposed filter will be referred to as the Proposed Adaptive Filter (PAF).
**Proposed Adaptive Filtering (PAF)**

In image processing, filters are mainly used to suppress either the high frequency components of the image, *i.e.*, smoothing the image, or the low frequency components, *i.e.*, enhancing or detecting edges in the image.

PAF is mainly a low pass filter because the additive fog in the input image can be considered as a high frequency signal that should be eliminated using a low pass filter mask.

Adaptive Filters are a class of filters which change their characteristics according to the values of greyscales under the mask and hence, the name “Adaptive Filter” is suitable for the proposed filtering technique. The Proposed Adaptive Filter makes use of local statistical properties of values under the mask which correspond to the mean and the variance measures. These measures are suitable parameters on which to base an adaptive filter because they are quantities closely related to the appearance of an image. The mean gives a measure of average gray level in the region over which the mean is computed, and the variance gives a measure of average contrast in the region.

Since fog is added to the image, the foggy image (X) can be written as:

\[
X = Y + G
\]

(11)

where Y is the clear (fog-free) image, and G is the fog; The fog is assumed to be normally distributed with mean 0. However, within the mask, the mean may not be zero. Given that \(m_f\) is the mean, \(\sigma_f^2\) is the variance, and \(\sigma_g^2\) is the fog variance over the entire image. In this case, the output value can be calculated as:
where \( g \) is the current value of the pixel in the foggy image. Note that if the local variance \( \sigma_f^2 \) is high, then the fraction will be close to 1, and the output will be close to the original image value \( g \). This is appropriate, because high variance implies high detail such as edges, which should be preserved.

Conversely, if the local variance is low, such as in a background area of the image, the fraction is close to zero, and the value returned is close to the mean value \( m_f \). In this case, the output can be defined by:

\[
g - \frac{\sigma_g^2}{\sigma_f^2} (g - m_f) \tag{13}
\]

Hence, the filter returns a value close to either \( g \) or \( m_f \) depending on whether the local variance is high or low.

Practically, \( m_f \) can be calculated by simply taking the mean of all grey values under the mask, and \( \sigma_f^2 \) by calculating the variance of all grey values under the mask. In fact, this filter attempts to minimize the square of the difference between the input and output images.

Moreover, the PAF mathematically is formulated using the following equations (14, 15 and 16). PAF estimates the local mean and variance around each pixel using equations (14 and 15 respectively):

\[
\mu = \frac{1}{NM} \sum_{n_1, n_2 \in \eta} x(n_1, n_2) \tag{14}
\]

and

\[
\sigma_f^2 = \frac{1}{NM} \sum_{n_1, n_2 \in \eta} [x(n_1, n_2) - \mu]^2 \tag{15}
\]
\[ \sigma^2 = \frac{1}{NM} \sum_{n_1, n_2 \in \eta} x^2(n_1, n_2) - \mu^2 \]  

(15)

where \( \eta \) is the \( N \)-by-\( M \) local neighborhood of each pixel in the foggy image. Since, a 7×7 window was used here, \( N = M = 7 \). The pixel-wise mask can, therefore, filter the image based on the mean and variance estimates as follows:

\[ y(n_1, n_2) = \mu + \frac{\sigma^2 - \nu^2}{\sigma^2} (x(n_1, n_2) - \mu) \]  

(16)

In spite of being a low pass filter, the Proposed Adaptive Filter does not completely eliminate edges, but to some extent, smoothens them while preserving enough features to be detectable by the second stage of the car detection.

1.3.2 Edge detection

1.3.2.1 Sobel operator

In edge detection algorithms, Sobel operator is used mainly to emphasize edges. Edge detection is the process of determining where the boundaries of objects fall within an image.

The Sobel edge detection method was introduced by Sobel in 1970. The Sobel technique performs a 2-D spatial gradient quantity on an image and, therefore, highlights regions of high spatial frequency that correspond to edges. Typically, Sobel (or any other edge detection) is used to find the estimated absolute gradient magnitude at each point in an input grayscale image [4].

The Sobel operator calculates the approximate image gradient of each pixel by convolving the image with a pair of 3×3 window. Sobel estimates the gradients in the horizontal (x) and vertical (y) directions which can then be combined to find the absolute magnitude of the gradient at each point and the orientation of the gradient as
seen in Figure 1.

![Gradient Operator](image)

Figure 1: Gx and Gy are gradient operator

The x-coordinate is defined here as increasing in the "right"-direction, and the y-coordinate is defined as increasing in the "down"-direction. At each point in the image, the resulting gradient approximations can be combined to give the gradient magnitude, using the following equation:

\[ |G| = \sqrt{G_x^2 + G_y^2} \]  

(17)

Natural edges in an image often lead to lines in the output image that are several pixels wide due to the smoothing effect of the Sobel operator.

The Local Laplacian Filters (LLF) [51] are based on standard image pyramids and the Gaussian pyramid is a set of images called levels, representing details at different spatial scales, representing progressively lower resolution versions of the image by down-sampling, in which high frequency details progressively disappear. The method produces consistently high-quality results, without degrading edges or introducing halos, but the iterative solvers are too slow to converge [52].
The Roberts Cross algorithm [53] implements a two dimensional spatial gradient convolution on the image. For the resulting edge detection, the main idea is to bring out the horizontal and vertical edges individually and then combine them. Roberts Cross is not quite as effective as the Sobel technique. It does bring out edges, but when compared to the same image using only the Sobel algorithm, the number of edges detected is poor. The Sobel approach seems to be superior.

In the Roberts Cross algorithm only two pixels are used, and hence, noise can be rather large. The Sobel algorithm deals with six pixels and in doing so, computes a better average of the neighboring pixels. This along with slight blurring can eliminate most noise found within normal images.

Prewitt edge detection [53] produces an image where higher grey-level values indicate the presence of an edge between two objects. The Prewitt edge detector mask is one of the oldest and best understood methods of detecting edges in images. Basically there are two masks, one for detecting image derivative in X and the other one for detecting image derivative in Y. In practice, one usually thresholds the result of Prewitt edge detection in order to produce a discrete set of edge.

In Canny edge detection, the image is initially run through a Gaussian blur to help get rid of the majority of noise. Next, an edge detection algorithm is applied such as the Roberts Cross or Sobel. Presently, the Canny edge detection algorithm is able to detect well-defined edges; however, for minor edges, the best results are obtained using the Sobel operator with the only drawback of being too slow [53].
1.3.3 Circle detection

1.3.3.1 Hough Transform for Circle Detection

The Circular Hough Transform (CHT) is a feature extraction technique for detecting circles. It is a special case of Hough Transform [11]. CHT finds circular formations of a given radius R within an image. The basic circle equation, based on this technique, in a two dimensional space, can be described by:

\[(x - a)^2 + (y - b)^2 = r^2\] (18)

using the parametric form of a circle: \(x = x_0 + R\cos \alpha\) and \(y = y_0 + R\sin \alpha\), where \((x_0, y_0)\) is point in the parameter space fall on a circle of radius R centered at \((x, y)\). The operation of CHT can be summarized in the following steps: Edges are detected in an image and then each edge point is taken as a center of a circle of radius R drawn onto an accumulator array which is raised with one. This circle is drawn in the parameter space, such that the x axis is the a-value and the y axis is the b-value while the z axis is the radius. When this step is complete, the accumulator will contain numbers corresponding to the number of circles passing through the individual coordinates. The highest numbers will correspond to the center of the circles in the image. The CHT can be formulated as a convolution applied to an edge magnitude image whose binary mask coefficients are set on the circle boundary and are zero elsewhere.

A 3-D accumulator array is employed and edge direction information is used in the Standard Hough Transform. One of the problems with the standard Hough Transform is the storage space required if the range of circle radii is large. Instead of using a three dimensional accumulator array for the Hough space, a single 2-D
accumulator and a 1-D histogram are adopted in the two-stage Circular Hough Transform [11], which are available as MATLAB built-in functions. This approach is implemented in this thesis to reduce the storage requirements when edge direction is available, in order, to decompose the circle finding problem into two stages.

1.3.4 Car detection

In general, there are different approaches to extract a vehicle from an image, and in this section a few of them will be highlighted. Firstly, the Object Extraction Algorithm (OEA) [12] using the background difference and time difference methods will be presented. In the background difference method an image free of any vehicle which is prepared beforehand is used to extract a vehicle. The comparison operation is performed on the input and background images to extract an object. On the other hand, in the time difference method a comparison operation is performed on two images differing in time interval to extract an object that has moved between the two. OEA uses a combination of background difference and time difference methods to make the best use of the features of each method best suited to the environmental change. OEA is able to extract a vehicle from an image stably.

Yong-Kul Ki [13] made efforts towards traffic accident recording and reporting model at intersections. This model first extracts the vehicles from the video images, tracks the moving vehicles and extracts features such as position, area and direction. Based on these features, the model can make decisions about the traffic accident. The work has a detection rate of 60%, with only 50% being correct.

Wei Zheng and Luhong Liang [14] proposed a novel set of image strip features to describe the appearances of common structural components such as
wheels, pillars, bumpers, etc. Image strip features are described using back-to-back regions considered as a template of a curve segment with a certain strip pattern. The algorithm is based on integral image method that builds a full set of image strip features with different curve segments, strip patterns, and positions. The new features represent various types of lines and arcs with edge-like and ridge-like strip patterns, which enrich simple features such as Haar-like features and edgelet features. For detection of cars, the authors use a sliding window strategy and the mean-shift clustering algorithm to merge the positive responses of the classifier and obtain the bounding boxes of the objects. The experimental results using UIUC and PASCAL 2006 [60] car datasets show that their approach outperforms the methods based on edgelet and haar-like features.

This algorithm can, mainly detect special cars only as it is limited by the shape of the strips. Moreover, if it is compared with complex local descriptors, the image strip features discard some statistical information, which weakens its discriminative capability [14]. See the following figure:

![Figure 2: Image strip feature vs. car structure](image)
Vinoharan, Ramanan and Kodituwakku [21] illustrated a novel side-view car detection technique which constructs an initial contour for the sake of wheel detection. The Snake algorithm proceeds with the initial contour to fit the boundary closer to the car. This system correctly detected and segmented 95 cars out of the 100 side-view cars that were tested.

Tan et al. [23] proposed a vehicle detection system for cars with different direction perspectives: front, back, side, and oblique. For the side-view cars, the knowledge of horizontal structures is used to generate the hypothesis area, and for oblique cars, a template matching technique is applied to generate the vehicle hypothesis. Using this hypothesis, line features are extracted from each sub-part of the designed template. The method used to detect side-view cars in [23] has been revisited in 2011 by the same authors [22]. They reported that side-view cars can be detected based on a template matching technique. The template is constructed using knowledge about the shape of a car which mainly consists of wheel positions and horizontal and oblique lines of a car. Hough transformation is used to detect circles of an image and then the wheels are detected based on assumptions that the two wheels are of the same size, exist in the same horizontal line, and are not present in the top part of the image. They estimate the approximate width and height of a car, where the height is about two-thirds and width is about three-fifth of the distance between the two wheels. Based on these assumptions, the initial bounding box is generated. They use the Hough transform method to detect lines that roughly picks out relatively long line segments part of the edges. The authors have tested their method on a set of images that has been collected on the World Wide Web. Their method shows an average success rate of 90.7% in detecting cars under the
assumptions that wheels exist on the same horizontal line with approximate width and height of a side-view car facing in the left direction.

The car detection algorithm proposed in this thesis is particle because it depends on a simple, yet exact and fast model of the side-view car facing any direction and present in any part of the image, i.e., top or bottom without assuming a particular car size.

The following Table presents the detection rate for the different algorithms that mostly used the same database as the one used by the proposed algorithm in this thesis.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Detection rate</th>
<th>Database</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zheng et al [14]</td>
<td>95 %</td>
<td>UIUC and PASCAL 2006 [60]</td>
</tr>
<tr>
<td>Vinoharan et al [21]</td>
<td>95 %</td>
<td>Google Images</td>
</tr>
<tr>
<td>Tan et al [22]</td>
<td>88 %</td>
<td>UIUC and World Wide Web</td>
</tr>
<tr>
<td>Tan et al [23]</td>
<td>90 %</td>
<td>UIUC</td>
</tr>
</tbody>
</table>

Table 1: Detection rate of different car detection algorithms
Chapter 2: Methods

2.1 Research Design

This chapter includes the proposed method followed in this research. The design challenges and expected impact will be explained first. Later, the system architecture will be illustrated with the simulation environment adopted and the data collected.

2.2 Design Challenges

Typically, the execution speed of algorithms is a concern when images are processed in real time. In this work, using a new filter for the transmission map of the fog removal algorithm and an accurate and quick car detection system will be presented.

The challenge is to be able to accurately detect a car in an image and as quickly as possible. Thus, high importance is given to the quality of the output of the Simplified Dark Channel Prior by using the Proposed Adaptive Filter and – at the same time – reducing the filtering execution time.

Needless to say, the system must also be capable of detecting cars even under changing conditions due to external environmental factors such as day and night, haze and sunlight.

Data collection was challenging due to lack of access to real car images – in the presence of fog – from street cameras. In order to overcome this difficulty, an existing database of car images was used which did not have fog in them, but fog was added to them. Different fog models exist,
linear, exponential and exponential square. In this work, a model based on Gaussian distribution – exponential square –, with mean ranging from 0.1 to 0.4 and variance ranging from 0.001 to 0.01 was used.

The proposed system is an integration of several techniques: fog removal, edge detection, circle detection and car detection algorithms.

2.3 Expected Impact

The car detection is the main concern of this research as it is intended to be used by traffic centers and to redirect traffic flow as needed in time; and thence prevents possible traffic incidents and delays that may occur due to that. Practically, the whole system should be executed in a few seconds to surpass other existing techniques with accurate results. The outcome of this research will, hopefully, assist traffic centers in UAE and around the world to better manage traffic under severe weather conditions. Moreover, a significant contribution in the fog removal technique has been proposed for fast transmission mapping based on a proposed adaptive filter without sacrificing quality of the fog-free image.
2.4 The proposed method

The system architecture is depicted in the figure below, illustrating the input, flow of key steps, stages and processes in the accelerated car detection system.

![System Architecture Diagram]

Figure 3: System architecture

The system is implemented in MATLAB on a laptop with an i5 quad-core 2.30 GHz CPU, 6G RAM, and 64-bit Windows 7. The input consists of a foggy image containing a car. After the fog is removed from the image, circles are detected and then as a final step, the presence or absence of a car is confirmed. The next section will present the fog removal approach, circle and edge detection techniques, and finally the car detection algorithm. The proposed algorithms are designed and implemented in MATLAB, v. R2014a, using image processing toolbox.
2.4.1 Fog removal approach

The focus in this section is on fast defogging methods, therefore we investigate a single image defogging method that aim to improve the contrast under a time budget constraint.

The estimation of the transmission map is critical for single image haze removal algorithms. There are two steps in most of the above algorithms that are presented in the literature section. In the first step, an initial value of the transmission map is given using a prior. In the second step, the transmission map is refined by using a local edge-preserving filter [54]. The proposed algorithm is based on the concepts of minimal color channel and simplified dark channel. The major function of the simplified dark channel is to reduce the variation of the direct attenuation. In addition, the simplified dark channel of the haze image can be decomposed into a base layer and a detail layer via an existing edge-preserving smoothing technique. The base layer is composed of the transmission map. Based on the observation, the Proposed Adaptive Filter is applied to decompose the simplified dark channel of the haze image. The dark channel image and atmospheric light are computed using the minimal color channel of the haze image. The estimated transmission map is finally used to recover the haze image. Experimental results show that the proposed SDCP algorithm with Proposed Adaptive Filter is applicable to haze images.

The filter is constructed by Adaptive filter with edge-stopping function and leads to an efficient method for edge-preserving smoothing, which can be used in the transmission estimation. Moreover, there is a comparison of different filters with their executed time processing to give an indication of the influence of filters.
The proposed method has following advantages. First, it can have halo-free image filtering without increasing any computational complexity and memory requirements. Second, it can achieve larger scale edge-aware image filtering than previous methods, which is quite useful for dehazing algorithms. For hazy image enhancement, we propose a more suitable edge-stopping function and the corresponding haze removal algorithm.

2.4.1.1 Koschmieder’s law

The well-known Koschmieder’s law presents the following haze image equation

\[
I(x) = J(x)t(x) + A(1 - t(x))
\]  

where \(I\) is the observed intensity, \(J\) is the scene radiance, \(A\) is the global atmospheric light and \(t\) is the transmission medium. The goal of haze removal is to recover \(J, A, \) and \(t\) from \(I\) [3].

2.4.1.2 Simplified Dark Channel Prior (SDCP)

The transmission \(t(x)\) is based on the Lambert–Beer law for transparent objects, which states that light traveling through a transparent material will be attenuated exponentially. When the atmosphere is homogenous, the transmission \(t\) can be expressed as:

\[
t(x) = e^{-\beta d(x)}
\]

where \(\beta\) is the scattering coefficient of the atmosphere [16].

Mainly, by understanding the previous equations, we can say that:

- The scene radiance is attenuated exponentially with the depth.
the observed image observed image $I(x)$, scene radiance $J(x)$, and airlight $a$, are all vectors with one intensity value per color channel.

- the attenuation coefficient due to scattering $\beta$ is not a function of the color channel, for a given pixel.

- The transmission is constant over all three RGB color channels.

Moreover, firstly, a good estimation of the airlight $A$ is needed, and then the two unknowns in (19) have to be solved: the transmission $t(x)$, which is related to the scene depth, and $J(x)$, the clear fog removed image.

The DCP was used as a base layer for the proposed fog removal method. The low intensities in the dark channel are mainly due to three factors: a) shadows. e.g., the shadows of cars, buildings and the inside of windows in cityscape images, or the shadows of leaves, trees and rocks in landscape images; b) colorful objects or surfaces. e.g., any object (for example, green grass/tree/plant, red or yellow flower/leaf, and blue water surface) lacking color in any color channel will result in low values in the dark channel; c) dark objects or surfaces. e.g., dark tree trunk and stone. As the natural outdoor images are usually full of shadows and colorful, the dark channels of these images are very dark [3]. Due to fog (airlight), a hazy image is brighter than its image without haze [55].

Due to the additive airlight, a haze image is brighter than its haze-free version in where the transmission $t$ is low. So the dark channel of the haze image will have higher intensity in regions with denser haze. Visually, the intensity of the dark channel is a rough approximation of the thickness of the haze. Moreover, they will use this property to estimate the transmission and the atmospheric light.
Actually, the main contribution in this stage is the filtering. As fact the DCP with soft matting give excellent results of the image enhancement and the contrast in the free-fog image, rather than its drawback is in the time and memory size. So, finding another filter that used to re-find the transmission map with consist of the scene radiance is must with fast response and good quality of the output image which will be more than enough required as an input for the next stage, the car detection method. As mentioned before, our new filter which is constructed by Adaptive filter with edge-stopping function and leads to an efficient method for edge-preserving smoothing, which can be used as new transmission estimation.

Algorithm-1

1. Inputs: >>>> I: foggy image
2. read the foggy RGB image
3. generate Dark Channel using eq.(2)
4. estimate atmospheric light
5. estimate transmission map using eq.(6)
6. recover the scene radiance eq.(8)
7. re-find transmission map using Adaptive filter eq.(16)
8. find edges information using edge-preserving gradient eq.(17)
9. re-find the new scene radiance >>>> Output: fog-free image

Given an input image, the algorithm generates Dark Channel which is determined by taking a 15×15 window. The entire set of pixels is set to the minimum value in all color channels within the window. This gray-scale image is then used to estimate the atmosphere lighting. The atmosphere lighting is defined by taking the
brightest pixels from the dark image, and using the brightest intensity of those pixels in the original image, as the Atmospheric Lighting.

The output image is then finally calculated by removing the atmospheric lighting from the input image and dividing by the transmission matrix, then re-adding the atmospheric light back to the image.

2.4.2 Edge detection

Filtering is normally used to improve the performance of an edge detector with respect to noise. However, more filtering to reduce noise results in a loss of edge strength. As the Sobel operator is using in this research as a kind of indication to examine the quality of the fog removal consisting with the car detection purpose.

Algorithm for edge detection using Sobel operator consist of three steps:

Algorithm -2

1. Filtering: gradient computation based on intensity values of only two points are critical to noise in discrete computations
2. Enhancement: determine changes in intensity in the neighborhood of a point as enhancement emphasizes pixels where there is a significant change in local intensity values
3. Detection: we only want points with strong edge content, so thresholding provides the criterion used for detection
2.4.3 Circle detection

In this thesis, the Circular Hough Transform algorithm is used to find the possible circular-like structures that are present in a given image. A simplified algorithm is proposed to identify the two wheels and reject the rest of the circles that have been detected in the previous step. As indicated earlier, the difference between the Standard Hough Transform which suffers from the large storage space required to detect circles with large radii. Moreover, instead of using a three dimensional accumulator array for the Hough space, a single 2-D accumulator and a 1-D histogram can be used for the two-stage Hough Transform. This approach is implemented here to reduce the large storage requirement when edge direction is available to decompose the circle finding problem into two stages. The storage space required for the method is quite small. The algorithm for Circular Hough Transform can be summarized as follows:

---

Algorithm -3

1. Specify radius range: a good rule of thumb is to choose radius range such that radius max < 3* radius min and (radius max – radius min) < 100
2. Find edges
3. Draw a circle with center at the edge point with radius r and increment all coordinates that the perimeter of the circle passes through in the accumulator
4. Find one or several maxima in the accumulator
5. Map the found parameters (radius and center) corresponding to the maxima back to the original image
2.4.4 Car detection

In the final step – after performing the Hough transformation to detect circles of an image – the wheels are detected using an algorithm based on a few assumptions. In [23], assumptions such as, the two wheels are to be of the same size, appear on the same axis, and are not at the top of the image with an approximate distance between the two wheels, were considered. In [23], the approximate width and height of a car were estimated, where the height was assumed to be about two-thirds and the width to be about three-fifth of the distance between the two wheels. The proposed car detecting algorithm is based on fewer assumption, i.e., the size of car wheels has to be similar and the wheels have to be on the same horizontal line with specified range of distance between the two wheels. The radius of the circles (to be detected as wheels) have to be within 2 pixels, while the difference between the y-coordinates of the two wheels has to be less than 3 pixels. Moreover, the distance between the two wheels should be in range 40 to 55 pixels. Based on the above assumption, the car is either detected or its absence is confirmed. See the design model, illustrated in Figure 4 below. The figure gives a template that depends on the physical configuration of the car model and it is a simple template that gives accurate results in short period of time. In this thesis, the assumptions are relaxed and used to detect cars without loss of detection accuracy.

![Figure 4: The template of the side car (red part is to be detected)](image-url)
In [23], the car detection method is highly dependent on the physical structure of the car, however this method will not work with other types of vehicles due to the variations in car body measurements and various shapes of sports cars, sedans, pickups, etc and also for the oval-roof shape. In order to alleviate the problem of relying on car shape, its presence at the top of the image, a simplified car detection algorithm is proposed below that simply relies on a range of wheel radii and a range of distances between wheels. The pseudo-code is provided below:

---

**Algorithm -4**

`% i and j : two different circles
% with radius ri, rj
% center (xi , yi)

For ( i= circle, i<= max circle-1, i+1)
For ( j= circle, j<= max circle-1, j+1)
{
    If abs (ri – rj) <= r_error % value of 2 was used for r_error
    If abs ( ci (y) – cj(y) )<= c_error % value of 3 was sued for c_error
    If (abs(ci(x)-cj(x))<= max_distance && (abs(ci(x)-cj(x)))>= min_distance
        Print: Car detected % 40 ≤ (ci(x) – (cj(x)) ≤ .55
    }
---
2.5 Data collection

The data base that was used in this research is the UIUC Image Database for Car Detection. It consists of 100 images of side views of cars for use in evaluating object detection algorithms [17].
Chapter 3: Results

In this chapter, the complete image filtering and car detection system is described as stages with supported techniques in MATLAB.

3.1 Profile and Statistics of Respondents

The entire system has been implemented in MATLAB. The system has been tested using side-view car images present in the UIUC database. The test set consisted of all the 100 different car photos included in the database. All images were modified by adding fog to them. The car detection system starts with the fog remover technique which uses the simplified dark channel prior with the combined filter. This combined filter consists of the Proposed Adaptive Filter (PAF) with edge-preserving operator to estimate the transmission map quickly while maintaining good image quality. Moreover, Sobel operator is applied to detect edges, in order to, ensure that the enhancement performed using the Proposed Adaptive Filter is acceptable for the next stage. In the next stage, the circle detection algorithm is applied which is based on the Circular Hough transform executed in two stages. Finally, the car detection algorithm is applied to decide on the presence or absence of a car in each image. This should aid in counting the number of cars on a road, which is very useful for traffic incident detection.

Having applied DCP, the main problem with it was the lengthy execution time as spends much time in filtering the image to remove any trace of fog. The soft matting filter, which is mainly based on Laplace function, consumes much time to complete. To resolve this, a different filter had to be used that consumed much less time. Alternative filters to accelerate the fog removal part of the system are
Laplacian, Gaussian, Laplace of Gaussian, Guided filter and Median filter. They were all examined and compared with the Proposed Adaptive Filter (Table-2.)

The Proposed Adaptive filter is a low pass filter that acts with statistical properties such as the mean and the variance of the proposed mask to obtain an estimated transmission map.

After performing simulation runs in MATLAB, it was determined that the execution times of the DCP and SDCP give approximately the same result with less time for the SDCP. However, the best result for edge-preserving issue was obtained using Sobel filter, which enhances the edges of an image fairly quickly. This is why Sobel filter was used to give an indication of the quality of the proposed fog removal algorithm.

The following tabulates the time required to run the different algorithms in seconds; all coded in MATLAB, using built-in functions where appropriate.

<table>
<thead>
<tr>
<th>Photo</th>
<th>DCP (s)</th>
<th>Soft Matting filter (s)</th>
<th>Laplacian filter (s)</th>
<th>Gaussian filter (s)</th>
<th>LOG filter (s)</th>
<th>Guided filter (s)</th>
<th>Median filter (s)</th>
<th>Proposed fog removal (s)</th>
<th>Proposed Adaptive filter (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>test 1</td>
<td>29.542</td>
<td>28.836</td>
<td>4.379</td>
<td>8.725</td>
<td>4.297</td>
<td>3.825</td>
<td>4.225</td>
<td>0.559</td>
<td>2.289</td>
</tr>
<tr>
<td>test 3</td>
<td>103.484</td>
<td>102.261</td>
<td>4.364</td>
<td>7.741</td>
<td>4.220</td>
<td>3.825</td>
<td>4.225</td>
<td>0.398</td>
<td>2.281</td>
</tr>
<tr>
<td>test 5</td>
<td>42.886</td>
<td>41.477</td>
<td>4.394</td>
<td>8.066</td>
<td>4.231</td>
<td>3.821</td>
<td>4.221</td>
<td>0.680</td>
<td>2.215</td>
</tr>
<tr>
<td>test 6</td>
<td>55.052</td>
<td>54.061</td>
<td>4.377</td>
<td>8.568</td>
<td>4.221</td>
<td>3.718</td>
<td>4.218</td>
<td>0.897</td>
<td>2.244</td>
</tr>
<tr>
<td>test 7</td>
<td>58.812</td>
<td>57.112</td>
<td>4.382</td>
<td>8.698</td>
<td>4.223</td>
<td>3.827</td>
<td>4.227</td>
<td>1.063</td>
<td>2.298</td>
</tr>
<tr>
<td>test 8</td>
<td>53.630</td>
<td>52.668</td>
<td>4.388</td>
<td>7.989</td>
<td>4.219</td>
<td>3.728</td>
<td>4.228</td>
<td>1.077</td>
<td>2.262</td>
</tr>
<tr>
<td>test 9</td>
<td>93.149</td>
<td>91.367</td>
<td>4.396</td>
<td>8.745</td>
<td>4.220</td>
<td>3.918</td>
<td>4.218</td>
<td>0.427</td>
<td>2.279</td>
</tr>
<tr>
<td>test 10</td>
<td>101.159</td>
<td>100.105</td>
<td>4.387</td>
<td>8.798</td>
<td>4.226</td>
<td>3.520</td>
<td>4.220</td>
<td>0.371</td>
<td>2.283</td>
</tr>
</tbody>
</table>

| Total time | 635.037 | 622.976 | 43.85 | 86.224 | 42.298 | 38.239 | 42.239 | 7.33 | 22.704 |
| Average    | 63.504 | 62.298 | 4.385 | 8.622 | 4.229 | 3.824 | 4.224 | 0.733 | 2.270 |

Table 2: Execution time (in seconds) comparison among different filters
All filters were applied for the SDCP algorithm to re-fine the transmission map based on mathematical formulations.

The Gaussian filter is implemented using the following formula [57]:

\[ h_g(n_1, n_2) = e^{-\frac{(n_1^2 + n_2^2)}{2\sigma^2}} \]  
(21)

\[ h(n_1, n_2) = \frac{h_g(n_1, n_2)}{\sum_{n_1} \sum_{n_2} h_g} \]  
(22)

And the Laplacian filter is implemented using the following formula [57]:

\[ \nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \]  
(23)

Figure 5: Comparison of ejection time of various applied filters for SDCP
The Laplacian of Gaussian (LoG) filters are implemented using the following formula [57]:

\[
\varphi^2 = \frac{4}{(\alpha + 1)} \begin{pmatrix}
\frac{\alpha}{4} & \frac{1 - \alpha}{4} & \frac{\alpha}{4} \\
\frac{1 - \alpha}{4} & -1 & \frac{1 - \alpha}{4} \\
\frac{\alpha}{4} & \frac{1 - \alpha}{4} & \frac{\alpha}{4}
\end{pmatrix}
\] (24)

Now, the following table mainly illustrate the difference in the executed time of the original DCP and the proposed one SDCP (here in this thesis) using appropriate filters. Moreover, the edge, circle and car detection execution time were included in the total execution time for the applied system as shown in Table-3.

The table clearly shows significant improvement in the total execution time using the proposed system. The speedup achieved using the proposed system as compared with the existing DCP ranges from 7 to 30, which is remarkable.

\[
h_g(n_1, n_2) = e^{-\frac{(n_1^2 + n_2^2)}{2\sigma^2}}
\] (25)

\[
h(n_1, n_2) = \frac{(n_1^2 + n_2^2 - 2\sigma^2)h_g(n_1, n_2)}{2\pi \sigma^6 \sum_{n_1} \sum_{n_2} h_g}
\] (26)
<table>
<thead>
<tr>
<th>Photo</th>
<th>DCP</th>
<th>Soft Matting filter</th>
<th>Proposed fog removal</th>
<th>Proposed Adaptive filter</th>
<th>Edge detection</th>
<th>Car detection</th>
<th>Total time (current)</th>
<th>Total time (proposed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>test 1</td>
<td>29.542</td>
<td>28.836</td>
<td>0.559</td>
<td>2.289</td>
<td>0.320</td>
<td>0.498</td>
<td>30.360</td>
<td>3.666</td>
</tr>
<tr>
<td>test 2</td>
<td>28.634</td>
<td>26.734</td>
<td>0.685</td>
<td>2.275</td>
<td>0.313</td>
<td>0.513</td>
<td>29.460</td>
<td>3.786</td>
</tr>
<tr>
<td>test 3</td>
<td>103.484</td>
<td>102.261</td>
<td>0.398</td>
<td>2.281</td>
<td>0.330</td>
<td>0.488</td>
<td>104.302</td>
<td>3.497</td>
</tr>
<tr>
<td>test 4</td>
<td>68.889</td>
<td>68.407</td>
<td>1.173</td>
<td>2.278</td>
<td>0.392</td>
<td>0.425</td>
<td>69.706</td>
<td>4.268</td>
</tr>
<tr>
<td>test 5</td>
<td>42.886</td>
<td>41.477</td>
<td>0.680</td>
<td>2.215</td>
<td>0.335</td>
<td>0.403</td>
<td>43.624</td>
<td>3.633</td>
</tr>
<tr>
<td>test 6</td>
<td>55.052</td>
<td>54.061</td>
<td>0.897</td>
<td>2.244</td>
<td>0.309</td>
<td>0.402</td>
<td>55.763</td>
<td>3.852</td>
</tr>
<tr>
<td>test 7</td>
<td>58.612</td>
<td>57.112</td>
<td>1.063</td>
<td>2.298</td>
<td>0.329</td>
<td>0.397</td>
<td>59.338</td>
<td>4.687</td>
</tr>
<tr>
<td>test 8</td>
<td>53.630</td>
<td>52.668</td>
<td>1.077</td>
<td>2.262</td>
<td>0.337</td>
<td>0.420</td>
<td>54.387</td>
<td>4.096</td>
</tr>
<tr>
<td>test 9</td>
<td>93.149</td>
<td>91.367</td>
<td>0.427</td>
<td>2.279</td>
<td>0.338</td>
<td>0.427</td>
<td>93.914</td>
<td>3.471</td>
</tr>
<tr>
<td>test 10</td>
<td>101.159</td>
<td>100.083</td>
<td>0.371</td>
<td>2.283</td>
<td>0.324</td>
<td>0.459</td>
<td>101.942</td>
<td>3.437</td>
</tr>
<tr>
<td>Total time</td>
<td>635.037</td>
<td>622.976</td>
<td>7.330</td>
<td>22.704</td>
<td>3.327</td>
<td>4.432</td>
<td>642.796</td>
<td>37.793</td>
</tr>
<tr>
<td>Maximum</td>
<td>103.484</td>
<td>102.261</td>
<td>1.173</td>
<td>2.298</td>
<td>0.352</td>
<td>0.513</td>
<td>104.302</td>
<td>4.268</td>
</tr>
<tr>
<td>Minimum</td>
<td>28.634</td>
<td>26.734</td>
<td>0.371</td>
<td>2.215</td>
<td>0.309</td>
<td>0.397</td>
<td>29.460</td>
<td>3.437</td>
</tr>
<tr>
<td>Average</td>
<td>63.504</td>
<td>62.298</td>
<td>0.733</td>
<td>2.270</td>
<td>0.333</td>
<td>0.443</td>
<td>64.279</td>
<td>3.779</td>
</tr>
</tbody>
</table>

Table 3: Executed time comparison between existing and the proposed algorithms
The image and its appearance after Proposed Adaptive filtering is shown in the following figure:

Figure 6: Example-1 applied Proposed Adaptive Filter results
Figure 7: Example-2 applied Proposed Adaptive Filter results
Needless to say, there were a few cases of falsely detected cars. An example is depicted in the following figure which is a real foggy image of cars in the city of Dubai [61].

Figure 8: Falsely detected cars example

After fog removal, the second step is performed to detect circles based on Circular Hough transform implemented in MATLAB, followed by the proposed car detection algorithm.

As a result, the set of images in the data-base, can be divided into different categories, which are: normal images, critical images and impossible images. The first group represents images that are clear and complete car features are present in them; such images are detected by the proposed car detection algorithm immediately. Some images, i.e., critical images were special as they required higher efficiency to
detect the circles due to having objects standing between the camera and the car to be detected. Obstructive objects can be light poles, pedestrians or trees. The last group consists of the impossible images in which circles can’t be detected as there is no complete wheel in the images.

The following images describe example of each category of images.

(a) original image  
(b) foggy image  
(c) fog-free image  
(d) edge detection  
(e) car detection

Figure 9: Normal image car detection
Figure 10: Critical image car detection
Figure 11: Impossible image – no car detection

Figure 12: Applied circle and car detection algorithms
Moreover, in figure-12, the output image when the Circular Hough Transform is applied shows many circles being detected. Note that, many are circles detected that are mostly not part of a car along with circles that represent the wheels of a car.

The database included additional 45 images with multiple cars (one had three cars and rest only two cars). The proposed car detection algorithm successfully detected the correct number of multiple cars in 41 images (including the image with three cars. The proposed algorithm achieved a detection rate of 91% for multiple cars. In order to properly detect multiple cars, the car detection algorithm must be slightly modified in such a way to avoid counting the same pair of wheels multiple times, i.e., as part of two different cars. This can be easily achieved by flagging the wheels of already detected cars and not considering them further while the car detection algorithm is running (see Algorithm-4 in section 2.4.4 for further details on the car detection algorithm).
The following images show an example of each category of images.

(a) original image  (b) foggy image

(c) fog-free and cars detection

Figure 13: Detection of three cars

(a) original image  (b) foggy image

(c) fog-free and cars detection

Figure 14: Normal image car detection
Figure 15: Car detection for critical image

Figure 16: Car detection for impossible image
3.2 Reliability of Individual Influence Scales

In this section, the assumptions considered in this thesis will be summarized. First, the assumptions of the fog removal technique using dark channel prior are relaxed. The shadows in the image are assumed to be the shadows of cars, buildings or leaves, trees and rocks. Dark objects or surfaces represent dark tree trunk and stones. Since natural outdoor images are usually full of shadows and color, the dark channels of these images are really dark. See Fig-18
Using a combined filter (proposed here), the fog-free image was obtained. While the picture was slightly degraded when compared to soft-matting (used in the original DCP), the edges remained intact, and the image’s contrast was satisfactorily enhanced, as was illustrated by the modified transmission map. The simulation results show that using the proposed combined filtering, 97% of the time cars were successfully detected which is only 1% less than the original soft matting filter. The benefit of the proposed filter is in its low execution time, i.e., speedup.

There is also a difference in the contrast enhancement of the output images from the two presented fog removal methods. By examining the refined transmission map (after performing fog removal), it can be seen that the difference – as shown in the following images – is mainly in the smoothing of edges representing the building and not the car. The applied modified (proposed) transmission map results in a large reduction in the execution time. Comparing the two techniques, SDCP (proposed) achieves a speedup of 7 to 30 over DCP presented by He Sun and Tang in [3].
(a) using DCP  
(b) using SDCP

Figure 19: Comparison between the outputs using DCP and SDCP

The transmission map is a very important and critical step of any fog removal technique because the recovered scene radiance depends on the transmission map. Figure 20 below, compares the original transmission map that is produced using the basic DCP, with two additional transmission maps that were refined after applying different fog removal algorithms (DCP with soft matting and the proposed SDCP with Proposed Adaptive Filter).

(a) original transmission map, b) refined transmission map using DCP with soft matting, c) refined transmission map using SDCP with PAF

Figure 20: Transmission maps

Although, the transmission map produced using the Proposed Adaptive Filter is able to recover the output scene radiance, the quality of the two output images as depicted in figure-19 suffers from a small degradation in the quality of the output image. The slight degradation in the output image quality is acceptable for the
purposes of this research because the objective is to detect the car(s) present in the image after removing the fog. This leads to a need for gauging the output quality to ensure that image quality is good enough for the second stage of the proposed system, i.e., car detection. One way to ensure image quality is by using the Sobel operator as an edge detector to check whether all car features (edges) are clearly visible and can be used to detect the car. The following figure gives the magnitude and direction gradients after applying the Sobel operator. It can be visually verified that the outputs have all car features preserved.

![Gradient Magnitude, Gmag (left), and Gradient Direction, Gdir (right), using Sobel method](image)

Figure 21: Outputs of Sobel Operator

In the car detection stage, the proposed algorithm needs to identify which circles belong to the car and represent the wheels. The identification is mainly based on the geometry of a car. Since the wheels are of the same size (given the side-view), both wheels exist on the same horizontal line, with a known approximate distance between the two wheels, circles that represent car wheels can be accurately detected.

So far, all images used for comparison with other techniques were artificially fogged images using the Gaussian distribution model. In the following, real foggy images [61–63] are used to examine the validity of the entire proposed method, i.e., the proposed fog removal and the car detection. In the absence of a database of side-view cars with real fog in the UAE taken from street cameras, a few images with real
fog were obtained through Google search from websites of local news agencies [61, 62]. The difficulty is mainly in locating real foggy images with side-view cars. The first two images are the original foggy images and the processed images using DCP with soft matting, and using SDCP with the Proposed Adaptive filter. The figures are show the edges detected followed by the detected wheels after applying the proposed fog removal and car detection techniques. In order to compensate for the distance between the camera that took the photo and the cars, the range of the distance and the range of the distance separating the wheels used in the car detection algorithm had to be modified to properly detect the cars.
(a) foggy image

(b) output of DCP with soft matting

(c) output of SDCP with the Proposed Adaptive Filter
Figure 22: Example-1, applied proposed methods to real foggy images from UAE
(a) foggy image

(b) output of DCP

(c) output of SDCP with the proposed adaptive filter
(d) edge detection

(e) car detection

Figure 23: Example-2, applied proposed methods to real foggy images from UAE
(a) foggy image

(b) output of DCP

(c) output of SDCP with the proposed adaptive filter
(d) edge detection

(e) car detection

Figure 24: Example-3, applied proposed methods to real foggy image

It is important to state that all images in the UIUC database and present in this research are mainly for cars in one lane streets (roads). Nonetheless, this proposed techniques can apply to multiple lanes as long as the side-view car images are provided, which may require separate cameras for each lane. For the case of two lanes (both single and two directions), each camera can be position on the shoulder side of each lane to obtain side-view images. In cases of more than two lanes, the proposed technique, will most certainly suffer from not being able to detect cars in the middle lane(s) as car wheels could be either partially or completely hidden from
the side cameras. In order to detect cars in cases of more than two lanes, other techniques are needed, perhaps based on the shapes of cars (almost rectangular).

Practically, the type of camera used in capturing the images can improve image quality and with certain types of cameras, fog can be filtered by the camera. Firstly, cameras based on Charge-Coupled Device (CCD) can produce images or recordings for surveillance purposes, and can be either video cameras, or digital stills cameras. Secondly, VITRONIC’s PoliScan traffic enforcement operate using VITRONIC’s innovative laser technology as established and used in the city of Dubai since 2014. A fan-shaped field of laser beams sweeps the tracking zone several times per second. This scanning light detection and ranging (LIDAR) records the speed and position of all vehicles within the enforcement area. Thirdly, Thermal imaging sensors are perfect for traffic monitoring. The thermal sensor can “see” vehicles in all conditions. Vehicles in traffic look the same to the sensor in broad day light or in the darkest of nights. They can operate in poor weather and through light fog and produce an image. Finally, The TrafiCam series of vehicle presence sensors combines a CMOS camera and video detector in one. The TrafiCam series allows you to exactly position and verify the vehicle presence detection zones. TrafiCam and TrafiCam x-stream detect vehicles, day and night. This allows vehicle presence detection over different lanes. The TrafiCam series has detection zones indicating presence of vehicles moving in a specific direction. Such cameras produce high quality thermal images on which the smallest of details can be seen [72 – 75].

The first three types of cameras need the fog removal algorithm as their outputs are affected by weather conditions, especially heavy fog. The fourth type which depends on CMOS technology would only need the car detection algorithm as
images from such cameras are clear enough and, to a large extend, are not affected by weather conditions.
Chapter 4: Discussion

In this chapter, the results obtained by executing – in MATLAB – the proposed methodology is discussed, using a database of 100 car images.

4.1 Fog removal

Fog is a serious problem in the UAE as it is one of the main factors of traffic accidents each year. Many accidents, including fatal ones involving multiple cars, occur during winter, especially when fog reduces visibility to near or less than 50 meters. The fog removal and car detection methodology developed in this thesis can assist traffic management systems to identify areas with incidents or heavy traffic in areas with existing cameras. In this thesis, the proposed system is divided into two stages: the fog removal stage and the car detection stage as depicted in Figure 25.

Figure 25: Summary of the two stages in this system
In the proposed system, the time required to remove fog from an image is approximately 30% of the entire time of executing the complete system (both stages combined). Fog removal is a crucial part of this work, as the failure rate to detect cars can reach up to about 30%, when circle detection is performed without fog removal.

The equation that describes fog formation in a foggy image is:

\[ I(x) = J(x)t(x) + A(1 - t(x)) \]  

(27)

\( I(x) \): clear mage as the required output image  
\( A(x) \): atmospheric light (air light)  
\( I(x) \): foggy image as input image  
\( t(x) \): transmission map

when the input (foggy image) enters the proposed system, the simplified dark channel prior technique is applied first to estimate the depth of the channel before performing any fog removal which occurs independently of the fog density and without any user intervention.

A new (proposed) and fast visibility restoration method is implemented that is performed on a single image, based on adaptive filtering and edge-preserving operator. Its main advantage is its speed compared to other methods, especially given that its complexity is only a linear function of the input image size. The two filtering operations are based on grayscale images, and hence, the method is applicable to real-time requirements. It performs better (faster) when compared with the state-of-the-art algorithms as was illustrated experimentally using real car images. The main reason for the speedup is because the method generates sufficient atmospheric light which recovers the depth edge information and smoothen useless texture information well.
From Table-2, it can be seen that the time of the proposed technique approximately reaches 5 seconds for all images as compared to other techniques which require a minimum of 30 seconds.

Indeed, there is a big difference in the execution time of the two techniques, which highlights the acceleration rate achieved.

The following graph illustrates a summary of the fog removal technique:

![Diagram](attachment:image)

**Figure 26: Fog removal process**

![Diagram](attachment:image)

**Figure 27: Flow of Simplified Dark Channel Prior**

One of the main objectives of this thesis is to accelerate the fog removal technique with the use of a proposed adaptive filter (see Figure-27). The simplified
dark channel prior is the proposed technique which is applied first, followed by an estimation of the atmospheric light. Next, the modified transmission map is estimated to describe the depth, which produces an enhanced image. Finally, the image is filtered to produce an output image without fog. Acceleration of the fog removal technique is achieved by proposing a fast Adaptive Filter (PAF). Table-1 compares the execution time of all studied techniques.

Due to the subjective nature of contrast enhancements, there has not been an established state-of-the-art method for single image defogging. The proposed work in this thesis is compared with the Dark Channel Prior (DCP) method proposed by He et al. [3] because it is the most common method for single image defogging. The DCP method, however, is complex and takes several seconds to minutes to process and is not considered a fast algorithm. Table-1 illustrates that the proposed SDCP Defogging method not only is extremely fast, but is subjectively comparable to the DCP method.

When circle detection technique was applied to the output images of the Simplified Dark Channel Prior (SDCP) without the use of filtering, only 10% of the cars in the entire database were detected correctly; and hence, the need for a good filter is essential.

The proposed Simplified Dark Channel Prior (SDCP) with filtering is performed in the first stage. The modified estimated transmission map (proposed in this research) produces as output fog-free images with good enough quality, but less than the original DCP output due to replacing the soft matting filter with the Proposed Adaptive Filter (PAF); however, PAF was able to preserve edges. Although differences in the enhancement output exits, the executing time of the proposed technique is much shorter when compared with that of the DCP technique.
The slight degradation in the filtered image had negligible effect on car detection (only 1% less).

4.2 Car detection

Car detection process includes circle detection, followed by counting the number of cars present in the image. Image processing techniques are implemented to count the number of cars in different streets under severe weather conditions. Streets are already provided with cameras for traffic monitoring / control purposes; hence, using existing street cameras, has the advantage of no additional infrastructure costs. Furthermore, street features such as the distance between the camera and cars is relatively fixed and the angle of view of existing camera can be easily modified if needed or deemed necessary.

The proposed car detection algorithm (not filtering) differs from previously published algorithms mainly in the assumptions. First, the size of the car wheel, limits the relevant radius of the circles in the image and hence simplifies the search process. Next, the distance between the two wheels is another factor. Eventually, it needs to be decided which circles in the image represent wheels of the car(s) to be detected. By restricting the search to the circles that are located on same horizontal line, the search process is greatly simplified for car detection.

After running the MATLAB code on 100 database images, images were classified as normal, critical and impossible groups. These groups depend on the car detection decision. As displayed in the results section (Figure 9 - 11) the various image groups are identified. It is imperative to mention that in a few cases, there were some difficulties in detecting the circles, but such difficulties were resolved by
adjusting the threshold values used in the Hough Transform from 0.75 in the normal cases to 0.95 in other cases.

The following comparison illustrates the efficacy of the proposed car detection algorithm using the same database as compared with other car detection algorithms.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Detection rate</th>
<th>Database</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zheng et al [14]</td>
<td>95 %</td>
<td>UIUC and PASCAL 2006 [60]</td>
</tr>
<tr>
<td>Vinoharan et al [21]</td>
<td>95 %</td>
<td>Google Images</td>
</tr>
<tr>
<td>Tan et al [22]</td>
<td>88 %</td>
<td>UIUC</td>
</tr>
<tr>
<td>Tan et al [23]</td>
<td>90 %</td>
<td>UIUC and World Wide Web</td>
</tr>
<tr>
<td>Proposed Algorithm</td>
<td>97 %</td>
<td>UIUC</td>
</tr>
</tbody>
</table>

Table 4: Comparison of detection rate for different applied car detection algorithms

The objective of this research was to achieve faster processing. It has been demonstrated that the proposed method achieves a speedup of up to 30 times faster than existing methods. This is mainly because of the simplicity of the proposed algorithm as it estimates local statistics using a linear filter. It has also been demonstrated that the proposed method is subjectively comparable to the DCP method by He et al. [3]. However instead of taking seconds to minutes to process the images, the proposed method takes a fraction of a second to complete.

Most practical computer vision systems require that the images are processed automatically and adaptively. An ideal haze removal method should be able to perform the haze removal automatically from images of different scenes under different weather conditions.
For the purpose of real-time car detection, it is imperative to reduce the complexity of the haze removal algorithm. Practical computer vision systems usually require the image processing method to be relatively fast applicable in real time. However, most of the existing haze removal methods suffer from high execution time and space complexities.

To enhance the visibility of foggy scenes, in this thesis, a fast fog removal algorithm, has been proposed. Although not tested in real-time, the algorithm is expected to perform well in real-time using outdoor (street) vision systems.

The proposed algorithm is independent of the fog density and does not require user intervention. It can handle both color as well as greyscale images.

To summarize, this research presents a fast method of detecting multi-view cars in real-world scenes. Cars are artificial objects with various appearance changes, but they all have relatively consistent characteristics in structure that consist of a few basic local elements such as wheels.

Table 5 below compares the car detection rate when used with various filtering and lack of filtering techniques. It shows how the detection rate of the proposed SDCP combined with the proposed Adaptive Filter achieved 97% detection rate. Furthermore, the database included 45 images with multiple cars (one had three cars and rest only two cars). The proposed car detection algorithm successfully detected the correct number of multiple cars in 41 images (including the image with three cars). And therefore, the proposed algorithm achieved a detection rate of 91% for multiple cars.
<table>
<thead>
<tr>
<th>Condition</th>
<th>Detection rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>with Fog</td>
<td>18%</td>
</tr>
<tr>
<td>with DCP / without filtering</td>
<td>30%</td>
</tr>
<tr>
<td>with DCP / with soft matting</td>
<td>98%</td>
</tr>
<tr>
<td>with DCP / with Laplace</td>
<td>90%</td>
</tr>
<tr>
<td>with DCP / with Gaussian</td>
<td>90%</td>
</tr>
<tr>
<td>with DCP / with Laplace of Gaussian</td>
<td>90%</td>
</tr>
<tr>
<td>with DCP / with Guided filter</td>
<td>95%</td>
</tr>
<tr>
<td>with DCP / with Median filter</td>
<td>92%</td>
</tr>
<tr>
<td>with SDCP and Proposed Adaptive Filter</td>
<td>97%</td>
</tr>
</tbody>
</table>

Table 5: Detection rate under different conditions

In order to gauge the effectiveness of the proposed algorithm when the fog density increase, additional experiments were performed using the same Gaussian fog model with the mean, ranging gradually from 0.1 to 0.9. The results are displayed in the table below.

**Car detection rate of images in the UIUC with various fog densities. Fog model used Gaussian Distribution (GD) with mean \( m \) ranging from 0.1 to 0.9**

<table>
<thead>
<tr>
<th>GD/( m=0.1 )</th>
<th>GD/( m=0.2 )</th>
<th>GD/( m=0.3 )</th>
<th>GD/( m=0.4 )</th>
<th>GD/( m=0.5 )</th>
<th>GD/( m=0.6 )</th>
<th>GD/( m=0.7 )</th>
<th>GD/( m=0.8 )</th>
<th>GD/( m=0.9 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>96%</td>
<td>96%</td>
<td>97%</td>
<td>95%</td>
<td>90%</td>
<td>67%</td>
<td>40%</td>
<td>11%</td>
<td>2%</td>
</tr>
</tbody>
</table>

Table 6: Car detection rate of images in the UIUC with various fog densities
Chapter 5: Conclusion

The fog was removed from the input images and the edges and circles were successfully detected in each side-view image. Finally, the majority (97%) of the cars were detected in a very short processing time using MATLAB for a database of 100 images.

The outcome of this thesis can help traffic centers to manage traffic flow, detect incidents and, in some cases, prevent accidents, using existing street cameras. Accelerating the fog removal and car detection was the main goal of this research which was achieved. A Proposed Adaptive Filter (PAF) and a simplified car detection algorithm were developed to improve the execution time of existing techniques. The proposed system was split into mainly two stages, fog removal and car detection.

A simple single-image haze removal algorithm has been proposed in this research by introducing an edge-preserving decomposition technique to estimate a new transmission map in a hazy image and recover the scene depth in a quick and efficient way.

The work in this thesis can be extended by adding more functionality. Below are some suggestions for further work:

- Use database images that represent photos of cars in the UAE, especially images of cars on the highways which often suffer from heavy fog.
- Use video rather than 2D image in order to determine the speed of passing cars.
- Implement in a hardware.
Bibliography

[1] Al Taher, Nada. "Thick Fog Leads to 114-vehicle Pile-up, 20 Hurt on Abu


[34] Pei, S.-C. and T.-Y. Lee (2012). Nighttime haze removal using color transfer pre-processing and dark channel prior. Image Processing (ICIP), 2012 19th IEEE International Conference on, IEEE.

graphics and applications(5): 34-41.


<http://www.ukspeedcameras.co.uk/guide.htm>.
<http://www.flir.com/traffic/content/?id=66601>.