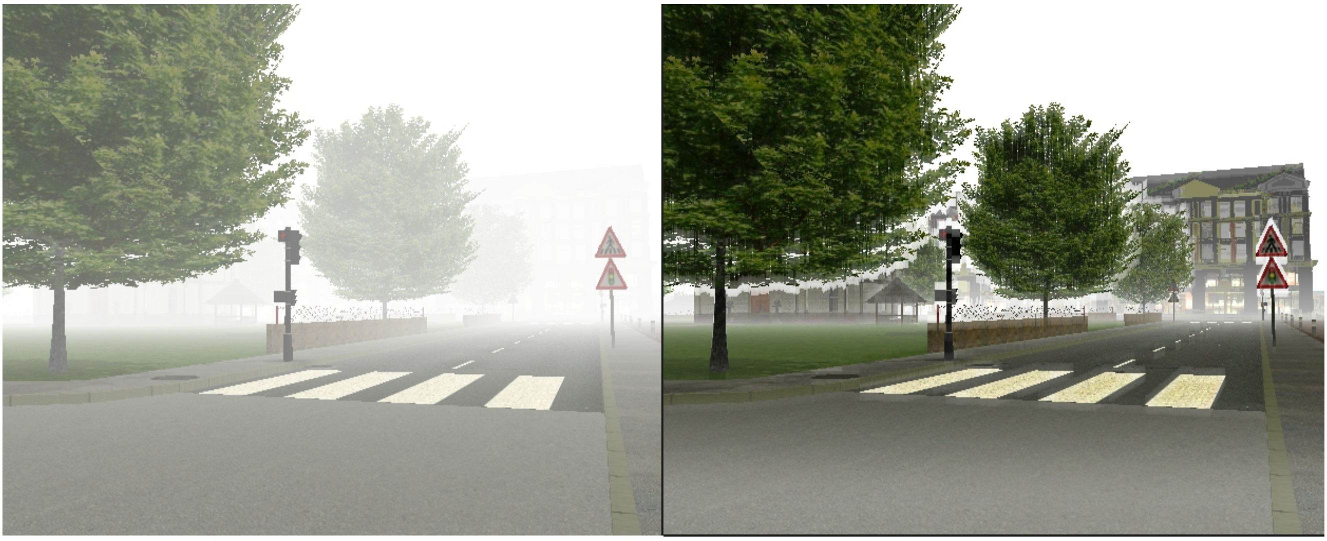
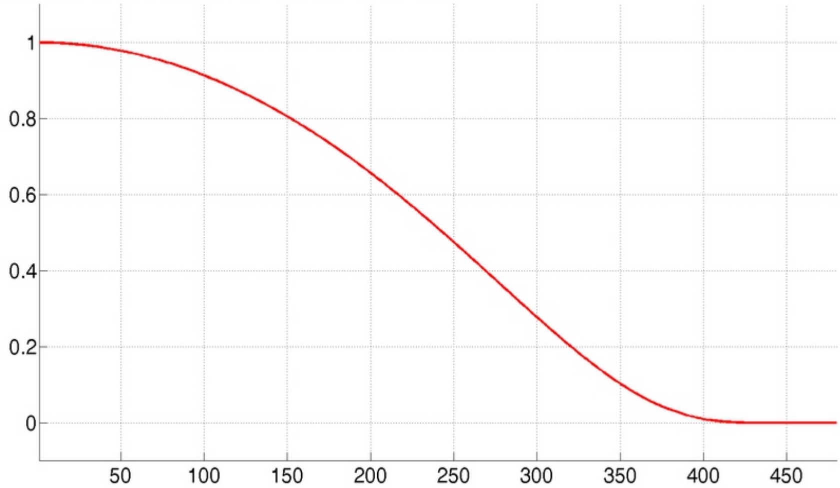
Mihai NEGRU

AUTONOMOUS DRIVING IN ADVERSE WEATHER CONDITIONS
Case Study - FOG





UTPRESS
Cluj-Napoca, 2019
ISBN 978-606-737-358-5

Mihai NEGRU

Autonomous Driving in Adverse Weather Conditions
Case Study – FOG

UTPRESS
Cluj-Napoca, 2019
ISBN 978-606-737-358-5

Editura U.T.PRESS
Str. Observatorului nr. 34
C.P. 42, O.P. 2, 400775 Cluj-Napoca
Tel.:0264-401.999
e-mail: utpress@biblio.utcluj.ro
http://biblioteca.utcluj.ro/editura

Director: Ing. Călin D. Câmpean

Recenzia: Prof.dr.ing. Radu Dănescu
Conf.dr.ing. Florin Oniga

ISBN 978-606-737-358-5

Contents

Preface **5**

1 Introduction **7**

2 Fog Meteorological Phenomenon **13**

2.1 Fog Meteorological Phenomenon 13

2.2 The Effects of Fog on Vision 20

2.2.1 Koschmieder's Law 22

2.2.2 Camera Model in the Vehicle Environment . . . 24

2.3 Conclusions 29

3 Fog Detection in Traffic Scenarios **31**

3.1 Fog Detection in Traffic Scenarios 32

3.2 Visibility Distance Estimation 48

3.3 Our Method for Fog Detection in Road Scene Images . 52

3.3.1 Horizon Line Estimation 52

3.3.2 Region growing 55

3.3.3 Fog Detection and Classification 58

3.3.4 Speed Warning Recommendation 60

3.3.5 Fog Detection Results 63

3.4 Driving Assistance in Fog Conditions 66

3.5 Conclusions 70

4 Contrast Restoration of Foggy Images 75

4.1 Image Enhancement 76

4.2 Non-Model Based Image Enhancement Methods 77

4.3 Model based Image Enhancement Methods 80

4.4 Contrast Restoration based on Koschmieder's Law . . . 96

4.4.1 Inferring the Atmospheric Veil 98

4.4.2 Median Filters for Image Defogging 100

4.4.3 Exponential Inference of the Atmospheric Veil . 109

4.4.4 Contrast Restoration Results 115

4.4.5 Optimizing Contrast Restoration on Embedded Hardware 125

4.5 Conclusions 134

5 Conclusions 137

Preface

This book is intended to highlight the fact that the path towards fully autonomous driving is becoming a huge struggle in the research communities, especially due to the limitation of sensors in perceiving the outside environment of the vehicle, but also due to the challenging and constant changing driving, environmental and weather conditions. The book is based on the research I have started during my PhD thesis [1] and continued throughout the years.

For being able to write this book I would like to offer special thanks, all my gratitude and love to my wife and daughter for their constant patience, trust and support. I would like to express my gratitude to my PhD adviser, Professor Dr. Eng. Sergiu Nedevschi. His guidance and constant push towards achieving the best possible results during these past years were very inspiring. His patience and objective opinions had an important role in fulfilling and improving the scientific contributions of this book. Special thanks go to my colleagues from the Image Processing and Pattern Recognition Group for their support and good advices. Being part of such a renowned research group is an important opportunity in one's academic carrier. Last but not least, my thanks go towards Professor Dr. Math. Ioan Radu Peter for his contributions to some of the mathematical aspects related to my PhD thesis and also this book.

The book is structured in five chapters. The first chapter consists in a brief introduction of the autonomous driving aspects and challenges. The second part introduces the fog meteorological phenomenon and presents why this phenomenon is dangerous for driving scenarios together with the effects of fog on vision systems. Image based fog detection techniques are described in chapter 3 with an overview of the fog categories present in traffic scenes. The next chapter deals with contrast restoration of fog degraded images or image defogging. The state of the art methods are presented together with our assessment and contributions to foggy image enhancement. Furthermore, we present an embedded solution for image defogging, suitable for advanced driving assistance and autonomous driving. The last chapter presents the conclusions and further discussions for achieving autonomous vehicle navigation in fog conditions.

I wish you a pleasant reading!

Chapter 1

Introduction

The future trend in the vehicle manufacturing industry is to build fully autonomous vehicles, i.e. vehicles capable of driving by themselves on the road, with no human intervention. This trend has started more than a decade ago with the introduction, inside the vehicle, of several sensors, systems and different functions not only for increasing road safety and for helping with the driving process, but also to introduce comfort inside the vehicle. These functions are grouped in several categories: passive safety sensors and systems (seat belts, airbags and airbag control units, pedestrian protection), active safety systems (anti-lock braking system – ABS, traction control system – TCS, electronic stability program – ESP$^{®}$) and driver assistance systems.

The driving assistance functions that are deployed today in modern vehicles, are designed based on the information provided by different sensors, such as: radars, ultrasound sensors, monocular and stereo cameras, laser scanners, etc. The driving safety systems can be further grouped in two main classes: systems that monitor the inside environment of a vehicle with a focus on the driver and his attention to the road and traffic conditions (driver behavior monitoring, driver fatigue detection systems, etc.) and systems that monitor the outside traffic environment and try to provide a degree of comfort and safety for the driving process (automatic cruise control – ACC, intelligent headlight control, parking aid and park assist, blind spot assist, rear

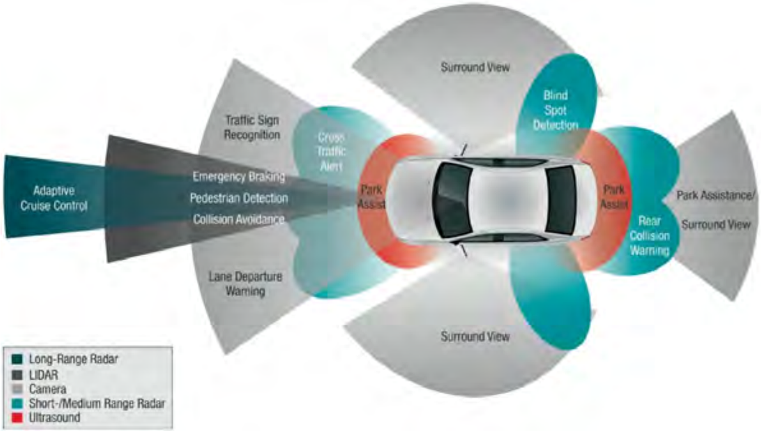


Figure 1.1: Sensors and the functions they will be used for in future autonomous vehicles [2]

view systems, lane assist and lane departure warning, collision avoidance, predictive pedestrian protection, automatic emergency braking, traffic jam assistance, etc.). Since in the future there will be no driver inside the vehicle, the industry focuses heavily on understanding and monitoring the whole 360°area surrounding the vehicle. The future placement of the sensors for achieving this task is depicted in figure 1.1.

According to the World Health Organization [3] road injuries have become the 9th cause of death in the world in 2009. It was estimated that 1 million people died in road accidents in 2000, while in 2012 this figure rose to 1.3 million people, that is a 30% growth in twelve years. This number has not changed in the recent years. Moreover, up to 50 million people are injured every year due to numerous accidents. Every day nearly 3500 people die in car accidents, with 600 people more than in 2000. If this trend continues, it is highly likely that road accidents will become the 7th cause of death in the world by 2030.

A detailed classification of the causes of road accidents is presented in [4]. Poor or bad driving behavior is the cause of about 90% of road accidents (distracted driving, speeding, drunk driving, reckless driving, running red lights, running stop signs, teenage drivers, night driving, unsafe lane changes, wrong-way driving, improper turns, tailgating, driving under the influence of drugs, road rage, drowsy driving, deadly curves, street racing). Other causes of traffic accidents are related to weather conditions (rain, ice, snow and fog), defects of the road surface or vehicles (potholes, tire blowouts, design defects) or stray animals (animals crossing the road). This great variety of causes for traffic accident, together with the wide range and complexity of traffic scenarios rises the need of new and improved automotive safety systems and advanced driving assistance applications that are capable to address all these difficult situations.

The aim of the automotive industry today is to prevent traffic accidents, to diminish the costs of such accidents and in the end to develop fully autonomous vehicles that are capable of driving in any scenarios. Another important aspect to consider is the comfort provided by an autonomous vehicle, i.e. the traffic participants can relax, read, or work during their journey with no stress induced by the traffic conditions.

The five levels of Autonomous Driving.

- Level 0 – No Automation. The driver is in full control of the vehicle, even if there exist several warning or intervention systems.

- Level 1 – Driver Assistance. There exist one or more driving assistance functions or modes that can provide assistance (in certain scenarios) for steering or acceleration / deceleration based on information about the driving environment. The driver must perform all other aspects of the driving task.

- Level 2 – Partial Automation. Both steering and acceleration / deceleration can be performed by a driving assistance system in certain scenarios. The driver must perform all other aspects of the driving task.

- Level 3 – Conditional Automation. The driving task is performed by an Autonomous Driving System. The human driver is not

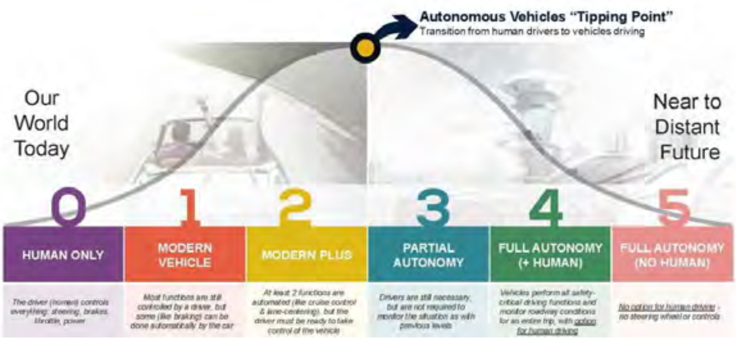

Autonomous Vehicles "Tipping Point"
Transition from human drivers to vehicles driving
Our World Today
Near to Distant Future
0
HUMAN ONLY
The driver (human) controls everything: steering, brakes, throttle, power
1
MODERN VEHICLE
Most functions are still controlled by a driver, but some (like braking) can be done automatically by the car
2
MODERN PLUS
At least 2 functions are automated (like cruise control & lane-centering), but the driver must be ready to take control of the vehicle
3
PARTIAL AUTONOMY
Drivers are still necessary, but are not required to monitor the situation as with previous levels
4
FULL AUTONOMY (+ HUMAN)
Vehicles perform all safety-critical driving functions and monitor roadway conditions for an entire trip, with option for human driving
5
FULL AUTONOMY (NO HUMAN)
No option for human driving - no steering wheel or controls

Figure 1.2: The 5 stages of vehicle's autonomy [5]

required to monitor the driving environment, but is expected to respond promptly at a request to intervene.

- Level 4 – High Automation. The driving task is performed by an Autonomous Driving System in all aspects, even if the human driver does not rapidly respond at a request.

- Level 5 – Full Automation. The full-time driving task is performed by an Automated Driving System under all roadway, environment and weather conditions, that can be managed by a human driver.

In the reminder of this book we present some aspects and systems based on monocular and stereo cameras used in driving assistance applications in adverse weather conditions. An important part of computer vision is the environment perception and scene understanding. This perception can suffer in the case of different or challenging illumination scenarios and can affect the quality of the observed scene in images. Traditional camera functions like automatic exposure and gain control are slow and do not cope well with the variety of lighting conditions changes when the camera (vehicle) is moving on the road, or in adverse weather conditions. In order for computer vision algorithms to be able to perform at high standards, the captured images must provide a high quality of the observed scene; the objects and artifacts must be clearly visible. Since the quality of the acquired images depends on many external factors, such as weather (sun, clouds, rain, snow, fog, mist) and lighting conditions changes (entering and going out of tunnels, shadows from trees or buildings, cars headlights) a methodology to obtain good quality images must be defined. The high dynamic range of the imaging sensor and the provided image resolution are also important factors to consider since the performance capability of a computer inside the vehicle or of a vision embedded system are limited. If the acquired images are poor all the other image processing functions will not perform well (3D reconstruction, lane detection and identification, obstacle detection and classification, semantic segmentation, etc.).

Weather conditions that also affect the image acquisition pipeline, such as fog, haze or mist, are very dangerous for driving. In these situations not only the visibility distance of the driver is greatly diminished, but also the image presents artifacts that induce noise in the scene reconstruction and object recognition processes. Fog is one of the most deadly weather phenomena for driving; a lot of fatalities happen in chain reaction accidents, because drivers tend to overestimate the visibility distance while traveling in fog conditions and drive with excessive speeds [6]. Fog produces accidents because a driver cannot see as far ahead. The presence of fog affects our perceptual judgment of speed and distance [7]. Humans tend to observe objects and perceive the distance to objects in the scene not by their absolute brightness or darkness, but based on their difference between the object's brightness and the background. This distinction between foreground objects and background is hard in fog conditions, because objects become fainter as fog density increases. Furthermore, the scattering effect of the fog makes people perceive objects which are low contrasted and indistinct as being farther away; humans judge the motion of objects to be slower than their real velocity. In extreme fog conditions humans are incapable to distinguish between motion and motion-less objects. So an important factor to be considered by advanced driving assistance systems and autonomous vehicles is this dangerous weather phenomenon.

The images captured in fog conditions have degraded contrast, that makes current image processing applications sensitive and error prone [1]. First one has to identify the fog situation, i.e. to detect the fog in the acquired images, then to estimate the visibility distance. For performing some reasoning or for understanding the scene in which the car is driving a contrast restoration procedure is suitable to be applied on the foggy image in order to obtain a fog free representation of the driving environment and to ensure that the image based perception functions work well in all situations. In driving assistance applications, the driver has to be informed about the fog scenario and advised about the maximum recommended speed in the given situation. In both driving assistance and autonomous driving, the effort of providing high quality reasoning in fog conditions is present throughout the whole image processing pipeline, from image acquisition to object detection and classification, lane identification, semantic scene understanding, and so on.

Chapter 2

Fog Meteorological Phenomenon

In this chapter we introduce the reader with the causes that lead to the formation of the fog meteorological phenomena: how is it formed and in what conditions. Then we briefly present why fog is one of the most dangerous weather phenomena for driving. Since the accidents that happen in fog conditions leave a lot of casualties and material damages, the research community in intelligent transportation systems has given a lot of attention for assisting the drivers in fog conditions. Additionally, we also introduce in this chapter the effects of this meteorological phenomenon in images, how it can be modeled mathematically and what are the particularities that can be exploited, by knowing the position and orientation of the camera in the vehicle environment.

2.1 Fog Meteorological Phenomenon

In some cases, when the environment's temperature is below the dew point, it is relatively cold and humid outside, the meteorological phenomenon that forms is called fog. The dew point is the temperature at which the air becomes saturated, it condenses into water droplets, which humans perceive as fog. Fog is actually a cloud formed or lying

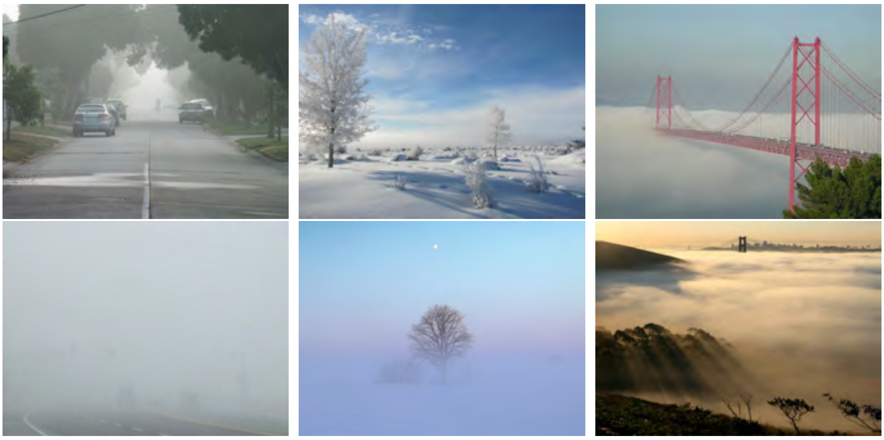


Figure 2.1: Example of images taken in different fog conditions

on the ground surface and is associated with a visibility distance less than 1000 meters. According to [8], in order for fog to appear in the atmosphere, there are two atmospheric criteria that must be met. The first one is humidity, i.e. the amount of water vapors present in the atmosphere must be high. If the air is very dry it is highly unlikely that fog is present. This is one of the reasons why fog forms after a rain or during rainy periods of time. The second one is the fact that the air mass must be almost saturated with respect to its humidity and temperature [8]. Only when saturated, water vapors change their form into liquid water droplets that comprise the fog as seen by the human eye. These droplets are very small in diameter; their size is from 1 to 20 μm. This is why fog and clouds are suspended in the air. But when there exist a slight warming of the air or the air is mixed with dryer air, the liquid water droplets will return very fast to vapor state and fog will quickly dissipate. This is why fog disappears quickly in the mornings when the sun raises on the sky or when there is a change in atmospheric pressure and the airs start to move (forming mild winds).

The human's visibility distance is the factor that distinguishes fog from other similar meteorological phenomena, such as haze or mist. Fog reduces visibility to less than 1000 meters, whereas the visibility in case of mist is between 1000 to 5000 meters. Some examples of images captured in different fog situations are presented in figure 2.1. The first two images present fog that appears in driving scenarios (urban and highway scenario, respectively). The next two images present foggy images captured during the winter season (ice fog and freezing fog).

Fog can be classified according to the mechanisms that lead to its formation [9], but several condensation mechanisms may simultaneously occur in the fog formation process. In the following we present the various types of fog and where and when are they formed.

Radiation Fog

Radiation fog is a type of fog that forms mainly during the nights with calm winds and clear skies. The heat that is absorbed by the surface of the earth during daytime, is radiated into space. The radiation fog may vary in depth from 1 meter to 300 meters. This type of fog is is stationary and is always formed at ground level. It can reduce a driver's visibility to near 0 meters as the density of water droplets present in the air increases, thus making it very difficult for driving. Radiation fog is mostly formed during autumn and early winter months, in cities and in rural and suburban zones. In cities is rather shallow, because a city produces and retains heat much better than more open spaces or vegetated suburbs, and thus it dissipates faster. But it can get very dense in surrounding rural and suburban zones, making driving and autonomous driving very hazardous.

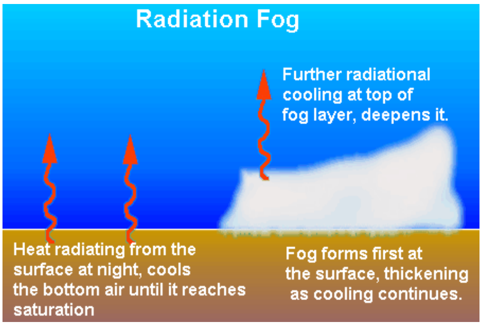

Radiation Fog
Further radiational cooling at top of fog layer, deepens it.
Heat radiating from the surface at night, cools the bottom air until it reaches saturation
Fog forms first at the surface, thickening as cooling continues.

Figure 2.2: Radiation Fog [10]

Advection Fog

Advection fog forms due to the horizontal movement of air that is warmer and more moist over a colder ground surface. The difference from radiation fog is that advection fog can form even when there are winds and cloudy skies. Fogs formed at see level are an example of advection fogs. Advection fog may also form when warm maritime air drifts over a colder mainland area. The warmer air will be cooled due to the influence of the colder ground surface. When this air cools the temperature drops towards the dew point. If the relative humidity of air reaches 100%, the condensation of water droplets will produce fog. After forming, this kind of fog can be very persistent because the weather situation that forms it can last for a longer periods of time, even up to one day or more.

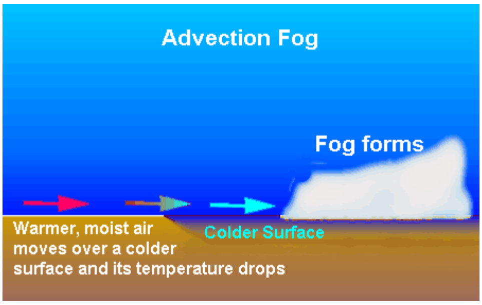

Advection Fog
Fog forms
Warmer, moist air moves over a colder surface and its temperature drops
Colder Surface

Figure 2.3: Advection Fog [10]

Precipitation Fog

Precipitation fog forms during rains through colder air and light winds, when warm raindrops reach the ground and then evaporate into a cooler drier layer of air near the ground. When enough rain drops have evaporated into the layer of cool surface, the relative humidity of this air reaches 100% and fog forms.

Up-slope or Hill Fog

Up-slope fog forms when the air moving in hilly terrain cools to condensation. A light wind can push warm air up a hill to a level where

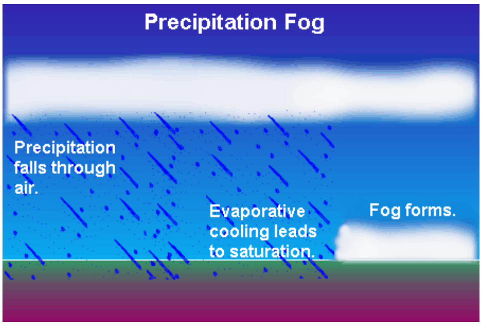

Precipitation Fog
Precipitation falls through air.
Evaporative cooling leads to saturation.
Fog forms.

Figure 2.4: Precipitation Fog [10]

the air becomes saturated and condenses. This type of fog usually covers large areas.

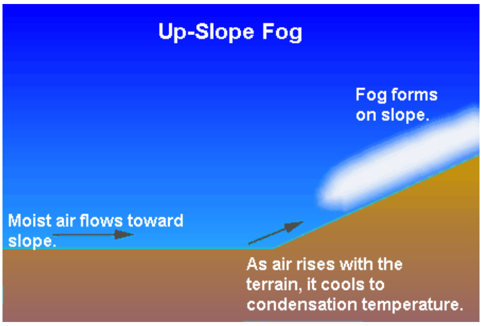

Up-Slope Fog
Fog forms on slope.
Moist air flows toward slope.
As air rises with the terrain, it cools to condensation temperature.

Figure 2.5: Up-slope Fog [10]

Valley Fog

Valley fog forms especially during the winter in mountain valleys. The formed fog is similar to the radiation fog, but limited by local topography. It can last for several days if the atmospheric conditions do not change. When the air at higher elevations cools down, this cold air drains down into the valley and thus deep fog forms inside the valley. This form of fog is also called Tule fog (in California is the leading cause of accidents caused by weather phenomenons).

Evaporation Fog or Steam Fog

Evaporation Fog or Steam Fog forms when a large quantity of water

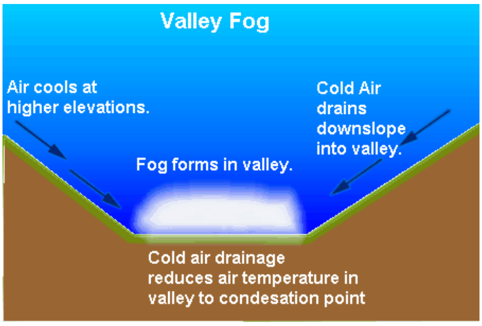

Valley Fog
Air cools at higher elevations.
Cold Air drains downslope into valley.
Fog forms in valley.
Cold air drainage reduces air temperature in valley to condesation point

Figure 2.6: Valley Fog [10]

vapor evaporates and mixes with cooler and drier air. There are two types of evaporation fog: steam fog and frontal fog. Steam fog forms when cold air moves over warm water, usually over lakes in the autumn season. When the cool air mixes with the warm moist air over the water, the moist air cools until its relative humidity reaches 100% and fog forms. This type of fog takes on the appearance of wisps of smoke rising off the surface of the water. Steam fog does not become very dense. Frontal fog forms when warm raindrops evaporate into a cooler drier layer of air near the ground. Once enough rain has evaporated into the layer of cool surface, the humidity of this air reaches saturation and fog forms. An example is the steam evaporating from the top of a forest after a heavy rain fall.

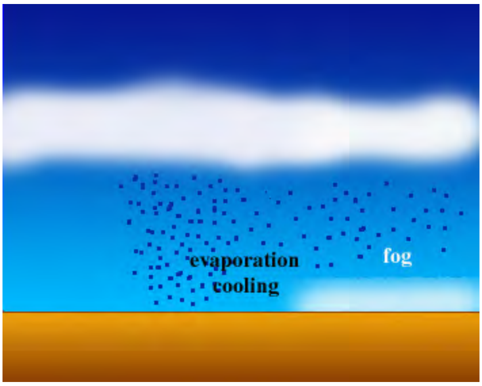


Figure 2.7: Evaporation Fog [10]

Ice Fog

Ice fog forms in cold areas (polar and arctic) when the temperature is well below freezing temperature. Generally this kind of fog requires temperatures between $-10^\circ\ C$ to $-35^\circ\ C$. It consists entirely of very small ice crystals that are suspended in the air.

Freezing Fog

Freezing fog usually occurs at mountain tops that are exposed to low clouds, when the water droplets are supercooled. These droplets remain in the liquid state until they come into contact with an object's surface upon which they can freeze. As a result, any object the freezing fog comes into contact with will become coated with ice. The same thing happens with freezing rain or drizzle.

The most important type of fog present in driving situations is the radiation fog. As a precaution drivers should always slow down and pay more attention to the road when traveling in fog. Even autonomous vehicles should take different measures or behave differently (on an algorithmic level) when driving in fog conditions. Fog affects a drivers perception in four ways [4]:

1. Fog distorts a driver's perception of speed. Due to the lowered contrast present in the atmosphere, humans might think that an object is traveling slower than its real velocity. This also applies to their own vehicle's speed, as one is not able to correctly judge his own speed based on the surroundings. Sometimes one cannot even see the surroundings! This is also true for image processing task. In heavy fog conditions the speed of detected objects is estimated wrong.

2. People have a hard time distinguishing between a stationary object and a moving one. More specifically, people have difficulties determining between parked cars and driving cars (both up coming and on going traffic). This easily results in a traffic accident.

3. People misjudge distance when driving in fog conditions. Humans instinctively judge that blurry objects are further away than clear ones. With fog reducing contrast and clouding the viewpoint, humans are more likely to misjudge just how far away another car or stationary object is, making car accidents far more likely. This is also true for a stereo vision system. In fog conditions the distance to an object is erroneously reported as being bigger than in normal conditions.

4. Faulty logic in fog conditions can increase the driving hazard. When driving at night in fog conditions some people use their high beams. Instead of increasing visibility, like one would expect, using the high beams will only make the situation more difficult, because more light will be scattered back to the driver. This will make driving conditions even worse.

When driving in fog conditions drivers must always use the low beams and the fog lamps (if their vehicle is equipped with fog lamps). The vehicle must proceed at a lower speed, even slower than the speed limit. By slowing down the vehicle, one allows an increase in the distance between the ego vehicle and the vehicles in front that could not be seen due to the fog's thickness. The high beams are never to be used in fog situations, because a large amount of light from the headlights will be back scattered by the fog droplets and back into the driver's eyes or into the cameras. This will reduce visibility even more and make it much more difficult to see or detect the objects on the road in front of the vehicle and along the side of the road.

2.2 The Effects of Fog on Vision

In day time fog scenarios the visible light, having a wavelength between 390 and 700 nanometers [11], must travel through the fog cloud in order to reach a human observer or a camera system. During its trajectory the light particles are attenuated by the large number of water droplets that form the fog cloud. This results in an absorption and diffusion phenomena that is characteristic for the fog meteorological phenomenon. In reality, the absorption phenomenon is negligible in this type of aerosol [12], so only the diffusion phenomenon is predominant in fog situations. The diffusion or multiple scattering of the light is a physical process where light deviates from its original straight trajectory by one or more paths due to some localized non-uniformities in the medium through which the light particles travel. This deviation also includes the deviation of the reflected light from the angle predicted by the laws of reflection. In our situation the scattering effect is caused by the large amount of water droplets present in the fog cloud.

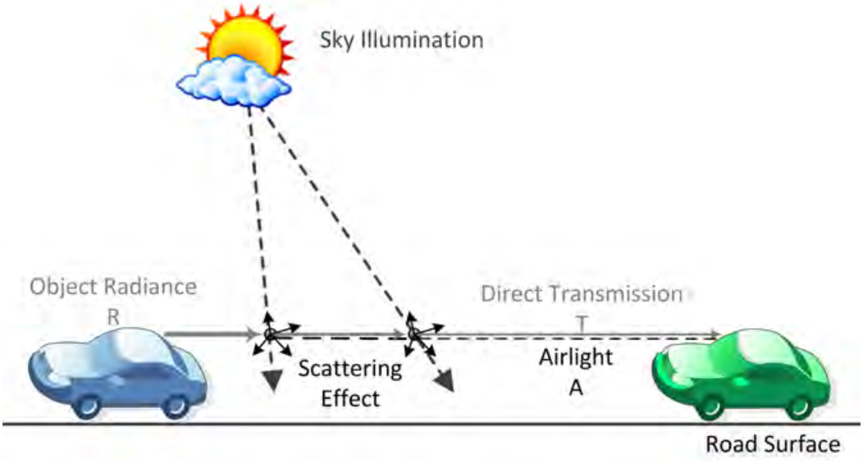


Figure 2.8: The effects of the daylight scattering on fog or haze illumination. The light coming directly from the sun and scattered by atmospheric particles (water droplets) toward the vehicle's camera is the air light (A). The light coming from the object (R) is attenuated by the same particles along the line of sight and is perceived as direct transmission T.

Figure 2.8 illustrates the scattering effect in fog or haze conditions. When observing an object (blue car) through fog the light that comes from that object, i.e. the object's radiance (R), is attenuated by the the water droplets that form the fog cloud along the line of sight and is perceived as direct transmission (T). The light coming from the sun is scattered, by the same water droplets when entering the fog cloud, and is perceived as air light (A) at the observing vehicle. The direct transmission decreases with the distance, while the air light increases with the distance between the object and the observer.

2.2.1 Koschmieder's Law

In 1924, Koschmieder [13] studied the attenuation of luminance through the atmosphere and proposed a relationship between the attenuation of an object's luminance L at distance d and the luminance L_0 close to the object:

$$L = L_0 \cdot e^{-\beta d} + L_\infty \cdot (1 - e^{-\beta d}) \tag{2.1}$$

L_∞ is the atmospheric luminance and β is the extinction coefficient (the sum of the absorption and diffusion coefficients). This equation states that the luminance of an object seen through fog is attenuated with an exponential factor $e^{-\beta d}$; the atmospheric veil, or fog addendum, obtained from daylight scattered by fog between the object and the observer is expressed by $L_\infty(1 - e^{-\beta d})$.

When dealing with images, the response function of a digital camera can be applied to the Koschmieder's equation in order to model the mapping from scene luminance to image intensity. Thus, the intensity perceived in the image is the result of a function (f) applied to equation (2.1).

$$I = f(L) = f(T + A) \tag{2.2}$$

Since equation (2.2) represents a linear mapping and assuming that:

$$\begin{aligned} f(T) &= f(L_0 e^{-\beta d}) = f(L_0)e^{-\beta d} \\ f(A) &= f(L_\infty(1 - e^{-\beta d})) = f(L_\infty)(1 - e^{-\beta d}) \end{aligned} \tag{2.3}$$

We obtain the following:

$$I = f(T) + f(A) = Re^{-\beta d} + A_\infty(1 - e^{-\beta d}) \tag{2.4}$$

where R represents the intrinsic pixel intensity of the image without fog (object radiance in figure 2.8) and A_∞ is the intensity of the sky in fog conditions (the air light from figure 2.8). Thus, the pixel intensity of an object seen through fog is attenuated with the distance of the object to the camera ($e^{-\beta d}$). The atmospheric light (A_∞) also contributes to the perceived intensity in the image. The distance d from equation (2.4) is a function of image coordinates and the term $e^{-\beta d}$ can be regarded as a transmission map of the medium, i.e. the proportion of light that travels towards the camera without any scattering.

Another property of the Koschmieder's law was derived by Duntley [13]. By rearranging the terms of equation (2.1) we obtain the following equation:

$$L - L_\infty = (L_0 - L_\infty) \cdot (e^{-\beta d}) \tag{2.5}$$

By dividing equation (2.5) with L_∞ one obtaines Duntley's atenuation law [13] which states that an object having the contrast C_0 with the background is perceived at distance d with contrast C:

$$\frac{L - L_\infty}{L_\infty} = \frac{L_0 - L_\infty}{L_\infty} \cdot (e^{-\beta d}) \tag{2.6}$$

$$C = C_0 \cdot (e^{-\beta d}) \tag{2.7}$$

This law can be applied only in day light uniform illumination conditions. From this expression the meteorological visibility distance is defined as: "the greatest distance at which a black object, having contrast C_0 ($C_0 = 1$), of a suitable dimension can be seen in the sky on the horizon [14]". In order for an object to be barely visible, the International Commission on Illumination [14] has adapted a threshold for the contrast, i.e. 5%. This value was chosen because 5% is considered as the minimum visual contrast required to recognize an object against its background. Having this constant in mind, i.e. $C = 5\%$ and knowing that $C_0 = 1$, one is able to solve equation (2.7).

$$e^{-\beta d} = 0.05 \tag{2.8}$$

By solving equation (2.8) the International Commission on Illumination has defined the "meteorological visibility distance" (d_{vis}) as:

$$d_{vis} = \frac{-1}{\beta} \cdot log(0.05) \simeq \frac{3}{\beta} \tag{2.9}$$

Hence, the meteorological visibility is in fact a more intuitive expression of the extinction coefficient β. If we are able to estimate the extinction coefficient in fog conditions than we can determine an approximation of the visibility distance for those driving conditions.

2.2.2 Camera Model in the Vehicle Environment

Usually the sensors used for image based advanced driving assistance systems and the front cameras for autonomous vehicles are mounted inside the vehicle behind the windshield, near (or embedded in) the rear view mirror. The main objectives of image processing algorithms are not only to detect and classify objects and other relevant traffic information from the observed scene but also to reliably estimate the distance to them. In the case of a stereo vision camera sensor it is possible to compute directly the distance to the camera of each point in the scene. In the case of a single camera system or monocular vision it is only possible to directly approximate the distance to a line in the image if the world in front of the camera is considered to be flat. This assumption is called the flat world hypothesis and is valid only for road scenes, where a large part of an image is formed by the road surface, which can be assumed to be planar [15]. However it is possible to infer the optical flow (the apparent movement) in the scene and derive the distance from the optical flow field. The approximated distance in this later scenario is more erroneous than in the stereo vision case.

In order to understand the next sections we briefly revise the main parameters of the pin hole camera model and its coordinate system. Figure 2.9 presents the camera coordinate system used for the camera calibration procedure. The camera parameters are:

- The position of a pixel in the image plane is given by its (u, v)

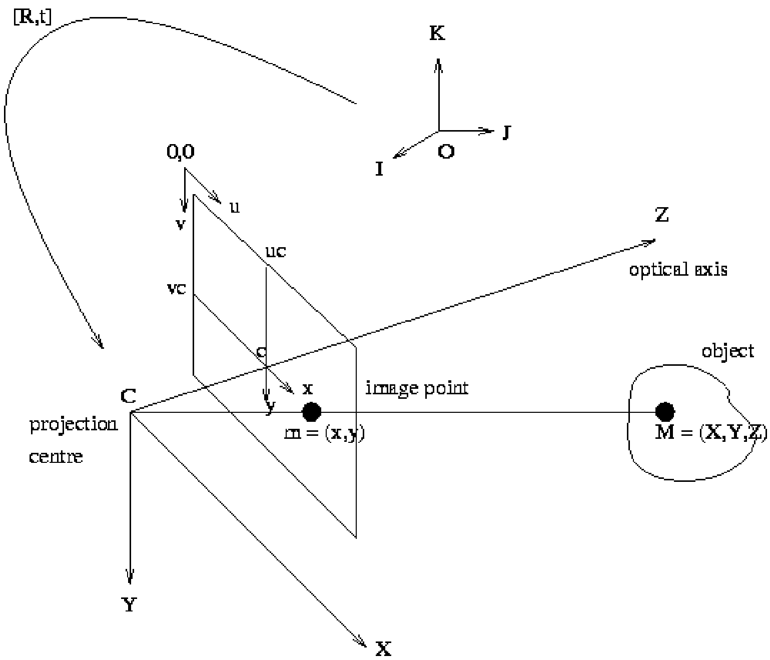


Figure 2.9: Camera Coordinate System [16].

coordinates

- The position of the optical center C is given by the (u_0, v_0) coordinates
- f represents the focal length of the camera
- the horizontal pixel size t_{pu}
- the vertical pixel size t_{pv}

Nowadays, cameras usually have square pixels, so the horizontal pixel size t_{pu} is equal to the vertical pixel size t_{pv}, $(t_{pu} = t_{pv} = t_p)$, and thus the focal lengths in the x and y direction f_x and f_y are also equal. A point $M = [X_C, Y_C, Z_C]^T$ in the coordinate system associated to the camera system will be projected in point $m = [x, y, -f]^T$ in the camera coordinate system, respectively in the point $[u, v]^T]$ in the image coordinate system. Writing the projection equations of the pinhole camera model, one obtains:

$$\begin{bmatrix} x \\ y \end{bmatrix} = f \cdot \begin{bmatrix} X_C/Z_C \\ Y_C/Z_C \end{bmatrix} = f \cdot \begin{bmatrix} x_N \\ y_N \end{bmatrix} \tag{2.10}$$

Where $[x_N, y_N]^T$ represent the normalized coordinates of point M with $1/Z_C$. The projection equation of the point M in the image plane is written as:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A \cdot \begin{bmatrix} x_N \\ y_N \\ 1 \end{bmatrix} \tag{2.11}$$

Where A is the internal camera matrix:

$$A = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.12}$$

When analyzing figure 2.10 and by using the perspective camera model [17] we can deduce the following:

$$\frac{u - u_0}{f_x} = \frac{x}{z} \tag{2.13}$$

$$\frac{v - v_0}{f_y} = \frac{y}{z} \tag{2.14}$$

Since $f_x = f_y = f$, let us introduce introduce a new constant denoted by $\alpha = \frac{f}{t_p}$ in order to express the value of the focal length in pixels. From the perspective camera model [17] and equations 2.13 and 2.14 we can derive that:

$$\begin{cases} u = u_0 + \alpha \frac{x}{z} \\ \\ v = v_0 + \alpha \frac{y}{z} \end{cases} \tag{2.15}$$

Figure 2.11 presents a typical camera system mounted inside a

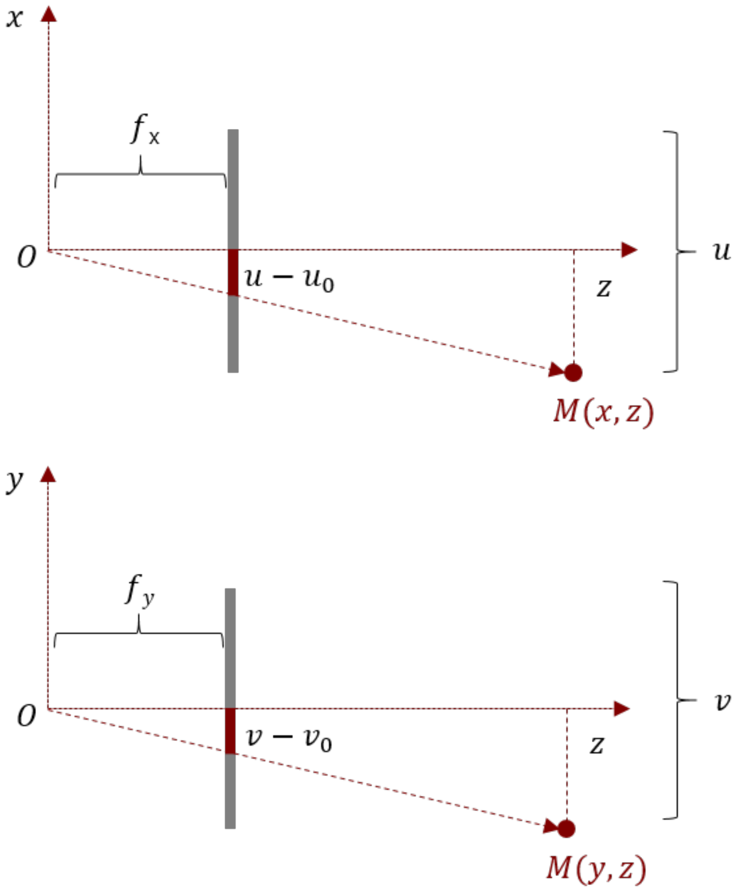


Figure 2.10: Point projection on the image.

vehicle. Usualy, the camera is mounted at height H relative to the $S(X, Y, Z)$ coordinate system and θ represents the pitch angle, i.e. the angle between the optical axis of the camera and the horizontal. In figure 2.11 we can observe that the horizontal line (v_h) passing through the optical center makes an angle θ with the Z axis [18]. Therefore it can be expressed as:

$$v_h = v_0 - \alpha \tan \theta \tag{2.16}$$

By using equation 2.16 and the expression of v from equation (2.15)

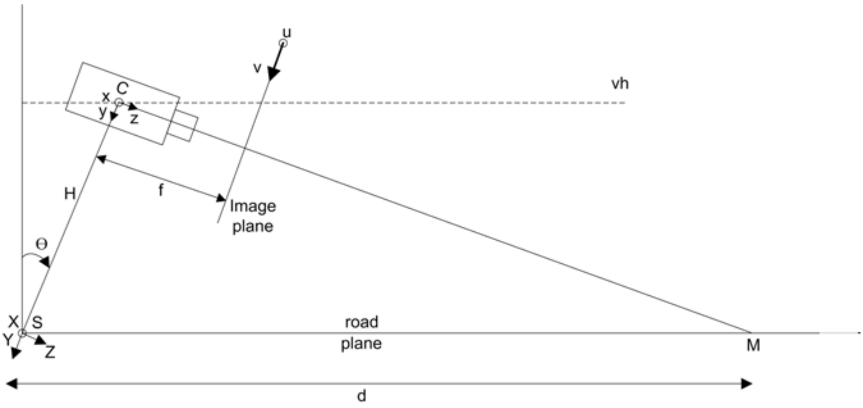


Figure 2.11: Camera system in the vehicle environment.

we obtain the following:

$$\frac{v - v_h}{\alpha} = \frac{y}{z} + \tan\theta \tag{2.17}$$

Considering the $S(X, Y, Z)$ coordinate system relative to the scene, equation (2.17) becomes:

$$\frac{v - v_h}{\alpha} = \frac{Y + H}{Z} + \tan\theta \tag{2.18}$$

Taking into account that a point M, belonging to the road at distance d from the origin S is given by:

$$M \cdot \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = M \cdot \begin{pmatrix} X \\ -d\sin\theta \\ d\cos\theta \end{pmatrix} \tag{2.19}$$

one can deduce that:

$$\frac{v - v_h}{\alpha} = \frac{H}{d\cos\theta} \tag{2.20}$$

Finally, the distance d of an image line v, can be expressed as follows:

$$d = \begin{cases} \frac{\lambda}{v - v_h} & \text{if } v < v_h \\ \infty & \text{otherwise} \end{cases} \tag{2.21}$$

where $\lambda = \frac{\alpha H}{\cos\theta}$ is a constant that depends only on the parameters of the used camera and v_h is the position of the horizon line in the image. The value of d expressed by equation (2.21) can be used in image processing applications in order to estimate the visibility distance in fog conditions.

2.3 Conclusions

This chapter has introduced the fog meteorological phenomenon, how and when does fog appear and why it is dangerous for driving. The most important type of fog present in driving scenarios is the radiation fog. This is the only type of fog that touches the road surface and thus reducing the visibility distance of the driver. Nevertheless other types of fog can also affect the driver's or a camera's perception in fog conditions, for example when driving over bridges, that cross over water. When driving in fog conditions people misjudge distance. Even in images, acquired with a camera inside the vehicle, it is hard to distinguish between stationary and moving objects and obstacles. The objects that are located further away are not correctly perceived, and the direct effect is driving with excessive speeds. All these facts constitute the reason why advanced driving assistance applications and autonomous vehicles must take into account the weather information, especially fog. The fact that the drivers are warned about the fog's density, the visibility distance and a safe driving speed can drastically reduce the accidents that happen in fog conditions.

The last parts of these chapter introduced the mathematical background that is required for the reader in order to understand the remaining of this book. Section 2.2 presented the effects of the light scattering produced in fog condition over the images. A brief introduction in Koschmieder's law and the background mathematical equations of the perspective camera model in the vehicle environment are also included in this section. In order to be able to compute the meteorological visibility distance, one has to know how to compute the distance to a point located on the ground when using a single camera system. For autonomous vehicles other sensors can also be used to asses this distance, but at this point in time these sensors are rather expensive, and their deployment in future vehicles is not certain.

Chapter 3

Fog Detection in Traffic Scenarios

As we have presented in the previous chapter, fog can have negative effects on driving, not only because the number of high accidents that happen in fog situations, but also because it reduces the visibility distance of a human driver, it hardens the distinction between stationary and moving objects or obstacles. In the end it induces stress on the driver because of the diminish perception capability. This also happens in the case of image processing functions in fog situations. So advanced driving assistance and autonomous vehicles need to learn how to cope when driving in fog conditions.

This chapter introduces some algorithmic approaches for detecting fog in traffic scenarios using the sensors equipped in today's vehicles. We first briefly study the state of the art in this field in section 3.1 and then present our theoretical and applicative contributions for fog detection in day time traffic scenarios. We analyze both urban and rural scenarios and we describe an original and robust solution for fog detection. Our algorithm works on both gray scale and color images, it is scalable and is able to first asses the presence of fog in the scene, then provide the driver with information about the density of the fog, i.e. the fog category, estimate the visibility distance and compute a safe speed for traveling on the given road segment. Section 3.2 presents the mathematical methodology behind the estimation of the visibility distance. The next section focuses on our algorithm for fog detection. Section 3.4 presents the overall fog detection framework and finally the last section draws the conclusions of this chapter.

3.1 Fog Detection in Traffic Scenarios

Accidents that happen in fog conditions represent an important concern for the road authorities around the world, because this accidents leave a lot of casualties and material damages. In the United States, The California Department of Transportation (Caltrans) [19] has developed a fog detection and warning system in California Central Valley area on Highway 99. This region is well known for its dense kind of fog, called “Thule fog”. This fog usually forms during the winter season and it reduces visibility to less than 200 m, and in some cases to nearly zero. The fog detection and warning system is based on an array of PWD10 forward scatter sensors, to detect fog. These sensors are installed every half mile in both directions of the highway, see figure 3.1. In addition specialized radar spot speed sensors were also deployed for measuring traffic volume, classification, speed and lane occupancy. The data from these sensors is used to asses the visibility conditions, infer the fog density and inform the drivers about the maximum speed that they should travel. All the necessary information for the drivers is displayed on Changeable Message Signs. All the sensors are connected in a wireless sensor network such that the fog information is collected and transmitted to a regional server. Traffic Message Channel (TMC) technology is also implemented by means of a Verizon wireless modem, so other drivers can be immediately notified about the weather conditions when they are approaching the target foggy area. Although it is a very expensive system the fact that the drivers are informed of the weather conditions and speed limits reduced drastically the number of accidents on this highway. So a system that alerts the driver about the visibility conditions and maximum safe driving speed would be a very good solution for advanced driving assistance systems. In addition such a system deployed on today's vehicles would be a more convenient and cost effective solution than deploying such a big array of sensors on every highway and national road.

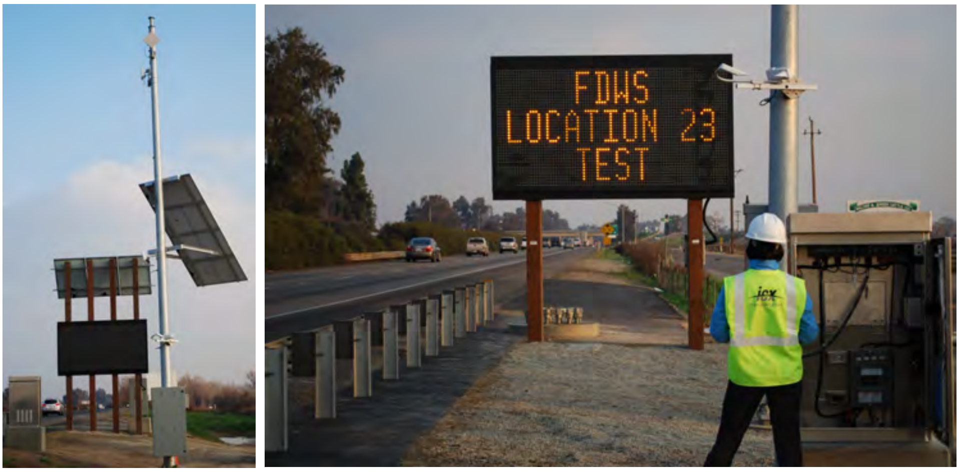


Figure 3.1: CalTrans Fog Detection and Warning System [19].

Cameras and radar systems, that are already deployed in today vehicles for various driving assistance functions can provide more cost convenient solutions that can also be used for assessing the visibility conditions in traffic scenarios. One of the first systems able to estimate the visibility distance was the RALPH system [20]. RALPH (Rapidly Adapting Lateral Position Handler) was an image processing based driving assistance system able to determine the road curvature, the lateral offset of the vehicle relative to the lane center and provide a steering command to the vehicle. In [21] Dean Pomerleau presents a solution for visibility estimation by using the RALPH system. In this work the visibility is computed by measuring the attenuation of contrast between consistent road features (road markings) at various distances in front of the vehicle. The assumption is that the visibility distance is considered to be the furthest distance at which a target can be reliably detected. The RALPH system utilizes a variety of road features that are visible on the road surface (lane markings, road boundaries, tracks left by other vehicles, oil stripes, etc.). But the absence of such features or if the field of view of the camera is blocked by a vehicle in front of the ego vehicle, makes the visibility distance estimation error prone.

There exist three main approaches for detecting fog in day time images: **stereo vision based methods, monocular vision based methods and the fusion between monocular vision and radar**.

Fog detection systems based on stereo vision are presented in [22], [23] and [24]. The authors try to build a depth map of the vehicle environment in fog conditions, using edge based stereo reconstruction. This depth map may contain numerous false matches, so by using the "V-disparity" method some of the false matches can be removed. In addition the authors perform an evaluation of the visibility distance and introduce the concept of mobilized visibility distance (v_{mob}), as being the distance to the most distant object existing on the road surface having enough contrast with respect to the background. Figure 3.2 presents a simplified road environment with road markings. The most distant visible object is the extremity of the last road markings. By comparison the meteorological visibility distance (v_{max}) is the greatest distance at which an image element from the road surface is visible. The authors propose to measure the mobilized visibility distance, by estimating the distance to the most distant road feature that has a contrast above 5%. The contrast measurement is based on Kohler's thresholding technique [25]. This scheme is able to find the threshold that maximizes the mean contrast between two parts of the image. The mobilized visibility distance is considered to be the distance associated with the disparity of the pixel that has a contrast above 5%. The problem with these methods is given by the errors induced by the fog in the $3D$ reconstruction process. Some of these reconstruction errors are proven in the next chapters. Since these errors cannot be quantified, the obtained mobilized visibility distance is error prone and the drivers can over speed in fog conditions. Furthermore, the mobilized visibility distance is always smaller than the meteorological visibility distance ($v_{mob} \leq v_{max} \simeq d_{vis}$). When no road features are visible in the foggy image the system is not able to estimate the mobilized visibility distance.

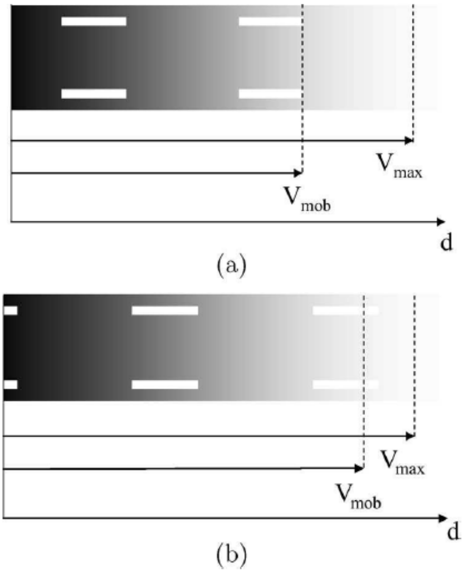


Figure 3.2: Examples of mobilized and meteorological visibility distances.

The authors in [26] and [27] fuse the information from an in-vehicle camera with a millimeter wave radar ($mm - W$) in order to classify the fog density and estimate the visibility range. The advantage of a millimeter wave radar is that it is not influenced by weather conditions, unlike the laser scanners or supersonic wave radar that are disturbed by the particle scattering, especially in fog conditions. Their method is based on the assumption made by Cavallo et. al. [28]: "under foggy conditions the distance between a preceding vehicle's tail lamp is perceived as being 60% further away than under fair driving conditions".

The first method [26] determines the fog's density by extracting a visibility feature from an image of a preceding vehicle and then classifies the fog's density into three levels based on the distance provided by the millimeter wave radar. First the preceding vehicle is detected by referring to the distance obtained by the millimeter wave radar. The position and size of the vehicle are detected by template matching in the candidate area. The authors claim that since contrast in images captured under fog conditions becomes low, the amount of high frequency energy also decreases in the frequency representation of the foggy image. So the image of the preceding vehicle is re-sized to a lower resolution of 32 x 32 pixels by linear interpolation and then it is converted into frequency domain by applying a discrete cosine transform (DCT) procedure. An indicator value was computed for the visibility of the preceding vehicle. The authors prove that as the preceding vehicle becomes indistinguishable the indicator decreases. The algorithm steps are depicted in figure 3.3. The problem with this method is the fact that the authors use only one preceding vehicle in the dictionary image, so the method is error prone in real traffic situations. Another concern is, that the authors are not able to categorize the fog conditions in case there is no preceding vehicle available. The fog classification is not done based on the visibility distance, but rather on the value of the visibility indicator.

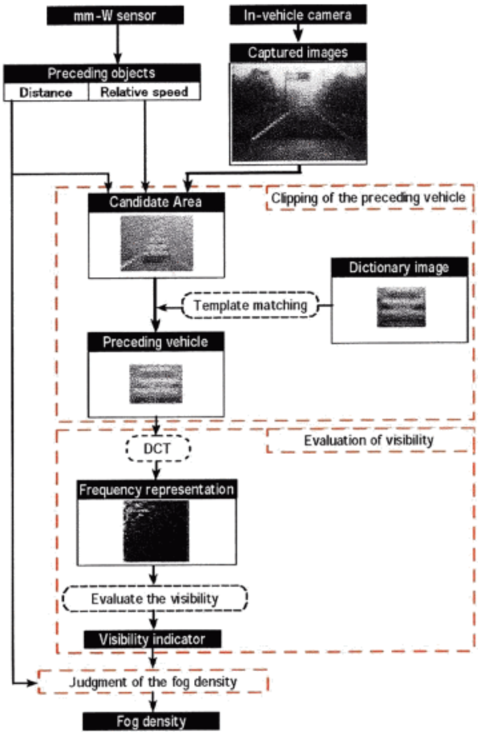


Figure 3.3: Fog density estimation method from [26].

The second method [27] tries to improve the fog classification by computing the value of the extinction coefficient from both visibility and distance information of the preceding vehicle and then use the computed extinction coefficient for fog density classification. In order to evaluate the visibility of the preceding vehicle, the authors use the variance of brightness as a metric. The image area of the detected preceding vehicle is compared with a fog free reference image (of the same vehicle). Then the distance obtained between the original image and the fog free image is used together with the distance measured by the millimeter wave radar in order to compute Koschmieder's atmospheric extinction coefficient. The main drawback of this method is the fact that the authors are not able to estimate the visibility distance of the driver; they provide only three classes of fog: light, moderate and high. In addition the method is not robust if the preceding vehicle's brake lamps are lit. Another drawback of that the vehicle used as a template image is also the vehicle used in the testing phase, which is not good for any template matching algorithm; not only that the training set and the test set should be different, but also the training set should be significantly large enough to cover all types and colors of vehicles. The accuracy of the method strongly depends on the computed distance between the original and the fog free reference image. Finally the fog conditions can not be inferred, when there is no vehicle in front of the ego vehicle.

A more reliable fusion approach for an in-vehicle camera and a 77 GHz radar is presented in [29]. The authors try to distinguish between two scenarios in order to determine the visibility range of the driver: an oncoming vehicle and a preceding vehicle. Figure 3.4 presents these two scenarios. Approaching vehicles allow for instant determination of the visibility range, while in the preceding vehicle case the camera measurements need to be constantly updated.

Their system works on-top of a multi-sensor fusion system described in [30]. A monocular camera with a vehicle detector [31] is fused with a radar sensor for distances input. The data from these two sensors is fused by a joint probabilistic data association (JIPDA) filter, that not only estimates the position of the vehicles, but also their existence

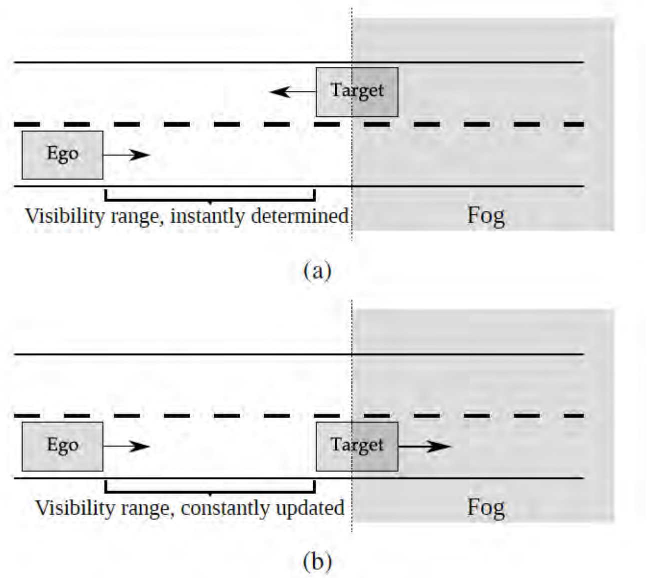

Target
Ego
Visibility range, instantly determined
Fog
(a)
Ego
Target
Visibility range, constantly updated
Fog
(b)

Figure 3.4: The two scenarios presented in [29]. To correctly determine the visibility range the authors try to distinguish between an approaching vehicle and a preceding one.

probability. Once a camera region of interest (ROI) is associated to a tracked vehicle, the visibility range can be estimated to be equal to the distance of the detected vehicle. As mentioned before, the system makes a distinction between on-coming and preceding vehicles. In the case of an approaching vehicle the visibility distance is considered to be the distance to the vehicle when it was first "seen" by the vehicle detector, while in the preceding vehicle case the detected vehicle needs to be monitored and the visibility distance is considered to be distance just before the tracked vehicle does not receive any measurement updates from the camera's vehicle detector. In real driving scenarios there are more situations than approaching and preceding vehicles. What happens if the ego vehicle and the preceding vehicle travel at the same speed over a highway (where there is no approaching vehicle)? Then the visibility range will be falsely estimated. Another weakness of this method is the fluctuation in the vehicle detector system. Darker approaching vehicles will be detected earlier, at greater distances, in fog situations than vehicles with brighter colors, thus the visibility range estimation will also be erroneous.

The last approach for detecting the presence of fog in the scene is by using monocular vision, monochrome or color images. Some approaches try to estimate the visibility distance of the driver by using only single image vision algorithms together with an a priori knowledge of the camera parameters in the vehicle's environment (pixel size, focal length, camera optical center, camera height, etc.). Other approaches are suited mainly for traffic surveillance systems [32]. The work presented in [15] tries to compute the visibility distance by estimating the position of the inflection point of the fog in images. Thus the visibility distance will be computed based on equation (2.21). The position of the horizon line in the image is considered to be known from the camera calibration parameters. The position of the inflection point in the image, i.e. the line in the image where fog ends, can be computed from the Koschmieder's law. By rewriting (2.1) and using the value of d from equation (2.21) one obtains:

$$L = L_0 - (L_0 - L_\infty)(1 - e^{-\beta \frac{\lambda}{v - v_h}}) \tag{3.1}$$

By taking the derivative of equation (3.1) with respect to the distance d the following is obtained:

$$\frac{\partial L}{\partial d} = \frac{\beta\lambda(L_0 - L_\infty)}{(v - v_h)^2} e^{-\beta \frac{\lambda}{v - v_h}} \tag{3.2}$$

As fog density increases, the objects appear less visible in the scene, they are dimmed by the fog; moreover the maximum of the derivative decreases more significantly and deviates from the horizon line. So, the second order derivative of equation (3.1) is computed as:

$$\frac{\partial^2 L}{\partial d^2} = \frac{\beta\lambda(L_0 - L_\infty)}{(v - v_h)^3} e^{-\beta \frac{\lambda}{v - v_h}} \left(\frac{\beta\lambda}{v - v_h} - 2 \right) \tag{3.3}$$

The equation $\frac{\partial^2 L}{\partial d^2} = 0$ has two possible solutions. The first one would be $\beta = 0$ which means there is no fog in the image and hence is of no interest for fog related scenarios. So the only valuable solution of equation (3.2) would be:

$$\frac{\beta\lambda}{v - v_h} - 2 = 0 \tag{3.4}$$

which leads to

$$\beta = \frac{2(v_i - v_h)}{\lambda} = \frac{2}{d_i} \tag{3.5}$$

where v_i represents the position of the inflection point in the image and d_i represents the distance of the inflection point to the camera.

An interesting observation is the following:

$$\lim_{v_i \to v_h} \beta = 0 \tag{3.6}$$

This can be used to detect fog in the image. If v_i is greater than v_h than fog will be detected in the image; otherwise no fog is present in the scene. Considering equations (3.5) and (2.9) the authors introduce the meteorological visibility distance d_{vis} as:

$$d_{vis} = \frac{3\lambda}{2(v_i - v_h)} \tag{3.7}$$

If one considers that v_v is the line in the image that represents the visibility distance, we can write from equation (2.9) that:

$$d_{vis} = \frac{\lambda}{v_v - v_h} \tag{3.8}$$

so finally, it can be deduced that the position in the image that designates the line of the visibility distance is given by:

$$v_v = \frac{2v_i + v_h}{3} \tag{3.9}$$

Their conclusion is that by finding the inflection point in the image, by using the second order derivative of the Koschmieder's law, the authors are able to determine the value of the extinction coefficient β that characterizes the density of the fog.

The determination of the position of the inflection point in the image becomes a problem of knowing which is a suitable object where the luminance variation can be measured. In the specific case of driving scenarios, the most specific object is the road surface, since it represents a dark object which is always present in the scene and is an interface between the road and the sky. The authors propose to seek a surface area in the image that presents small line to line variations in luminance. Since the road blends into the fog, the surface considered includes parts of the road and of the sky. So a bottom-up region growing algorithm is employed on the image. Inside this region a measurement bandwidth is considered for computing the inflection point. If the region does not cross the image from top to bottom the algorithm is not capable to measure the luminance variation, otherwise a vertical bandwidth is chosen in this region that crosses the image from top to bottom. Once the limits of this vertical band are found the median values of the pixels that belong to a line in the band is computed. This yields a function whose local maximum positions represent the possible positions of the inflection point. The position of the global inflection point is the one that minimizes the squared error between the issued model and the measured curve. Figure 3.5 presents some results of using this method. The first row in figure 3.5 represent the edge detection output of some foggy scenes. The second row is the output of the region growing algorithm while the third row presents vertical band where the position of the inflection point is computed; the white line is the final result, i.e. the inflection point line that is used to asses the visibility distance in the scene. Figure 3.6 shows the evolution of the visibility distance computation on three separate image sequences, each having 150 frames. It can be seen that the visibility distance varies across a sequence, since there are situations when the visibility distance cannot be computed: the field of view of the camera is blocked by preceding vehicles, the existence of road signs, etc.

One other benefit of such an algorithm is the possibility to infer the free space in front of the ego vehicle, directly from the region growing process. Such an approach is presented in [33]. Figure 3.7 presents the

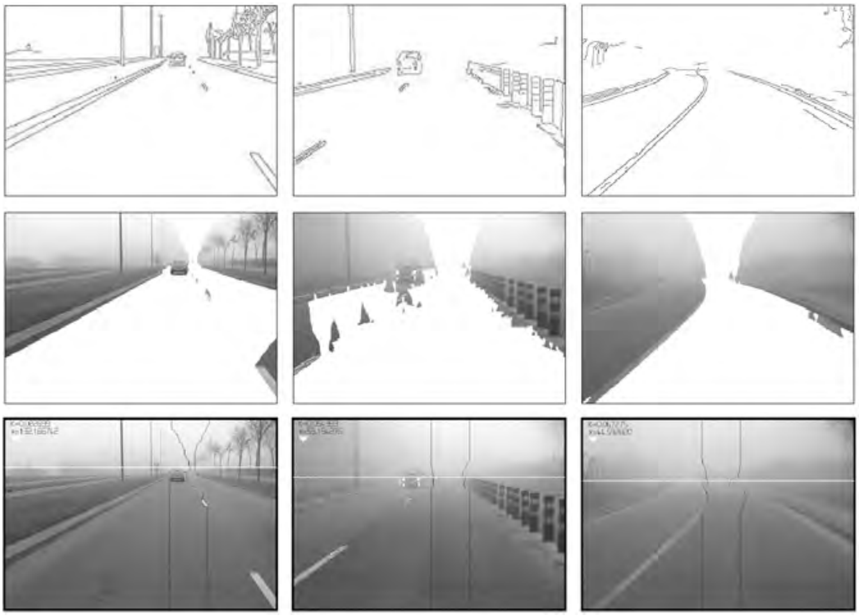


Figure 3.5: Visibility distance estimation [15]. First row represents edge images, second row region growing, third row the white line represents an estimation of the visibility distance.

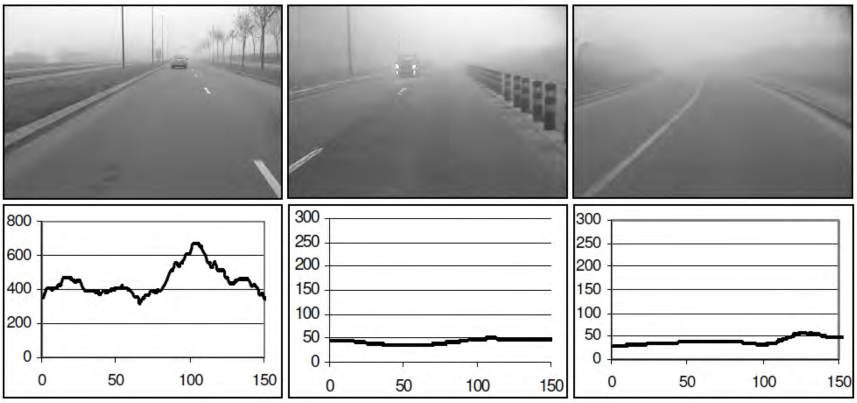


Figure 3.6: Visibility distance measurement on three sequences[15]

results of inferring the vertical objects present in the foggy scene. A drawback of this method is the fact that it can only be used to detect the drivable road surface and the vertical obstacles in the scene. A more reliable solution should be design in order to detect more artifacts in the scene, such as: curb stones, sidewalks, pedestrians, traffic signs, poles, vehicles, etc.

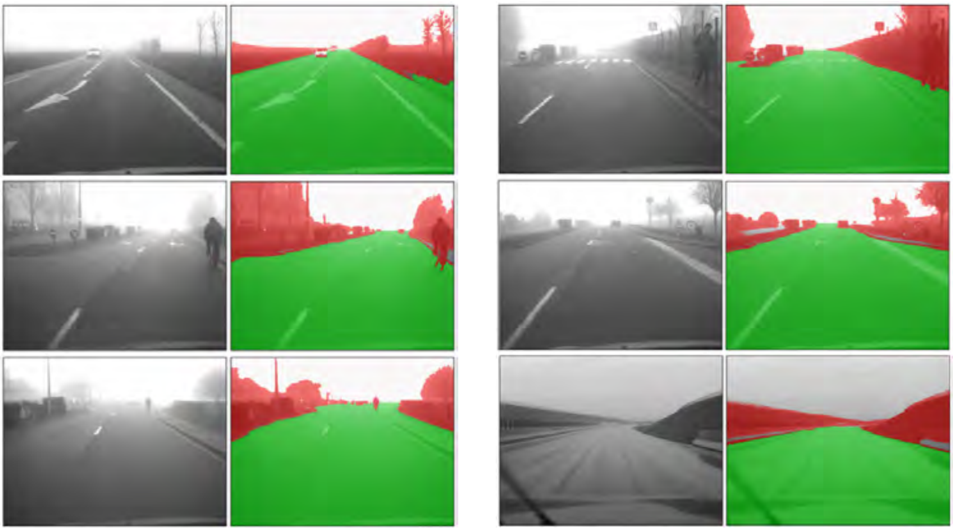


Figure 3.7: Free-space detection of foggy road scenes [33].

Another approach for detecting the presence of fog in images is presented in [34]. The authors rely on the fact that foggy images are blurrier and have lower contrast than images captured in normal weather conditions. Based on this assumption the authors scale the input image to a lower resolution of 320 x 240 pixels and perform a shallow image classification based on the Sobel edge detector [35]. The outcome is a binary classifier: sunny or foggy/cloudy scene. The steps of the algorithm are presented in figure 3.8. In case the image is categorized as being foggy the vanishing point is computed in the scene in order to compute the position of the horizon line.

In this work the road lines are filtered and only two relevant lines are used for the computation of the vanishing point. After the vanishing point is found a sky-road segmentation process in performed. When the image is foggy a portion of the road will be segmented as sky, so the difference between the location of the vanishing point and the location of the sky-road limit will be used to characterize the fog conditions, see figure 3.9. The visibility distance is then estimated by using the

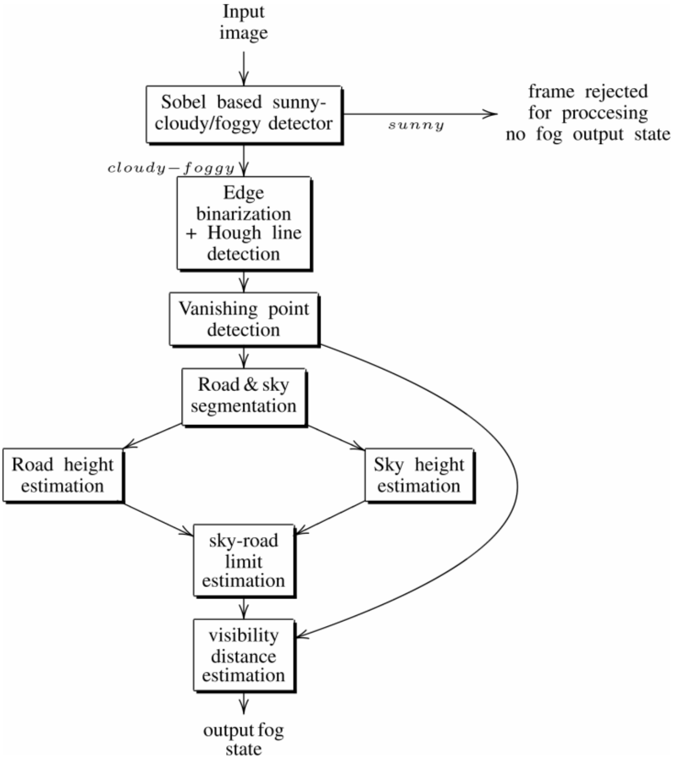

Input image
Sobel based sunny-cloudy/foggy detector
sunny
frame rejected for proccesing no fog output state
cloudy−foggy
Edge binarization + Hough line detection
Vanishing point detection
Road & sky segmentation
Road height estimation
Sky height estimation
sky-road limit estimation
visibility distance estimation
output fog state

Figure 3.8: The fog detection steps presented in [34].

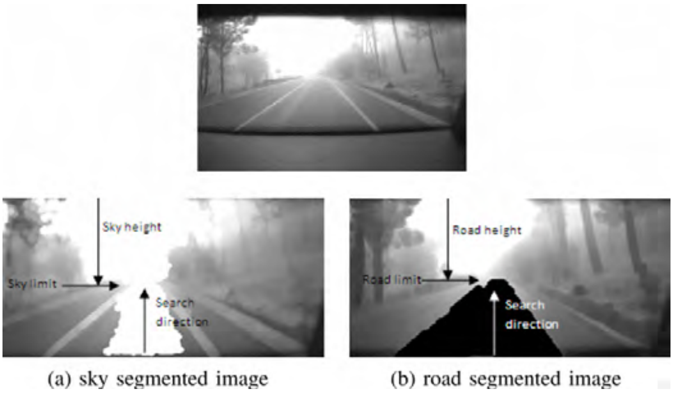


Figure 3.9: Segmented foggy images to find the sky / road boundary [34].

perspective camera's projection equations. The authors use only three fog categories: when the visibility distance is over 300 meters, low fog or sunny; moderate fog when the visibility distance is between 100 and 300 meters and high fog when the visibility distance is below 100 meters. However, this approach presents some problems: it is not able to differentiate between very sunny weather and fog conditions because of the road-sky segmentation process used or when more vehicles are present on the road and the algorithm is not able to perform the road-sky segmentation process; the edge based classification done in the first step of the algorithm is erroneous, since, in real scenarios, one can not estimate the amount of edges present in the top half of the image in order to classify the image as foggy or sunny / not foggy. In some scenarios where there are trees or buildings on the side of the road more edges could appear in the top of the image, so this assumption is erroneous.

Fog detection in night time driving environments was studied in [36] and [37]. The visual effects of night fog are analyzed and the authors propose two methods to detect the presence of fog by detecting the back scattered veil induced by the ego vehicle's own lights and by detecting the halos around different light sources present in the considered scene. The first approach is suitable when the vehicle is alone on the street and no public lighting is available, while the second method is suitable in the cases where there exist road traffic or public lighting on the road. In the case of night fog, if a vehicle is alone on the road the fog can only be noticed by the back scattered veil from the ego vehicle's light. If the vehicle meets other vehicles on the road or when there exists public lighting, the back scattered veil is no longer observable, but the fog situation can still be identified due to the presence of halos around these light sources. Images in bad weather conditions that contain light sources were analyzed in [38], [39] and [40]. But this methods use only static cameras, they were not designed for advanced driving assistance systems. Other methods rely on the back scattered light of the ego vehicle's headlights in order detect the presence of fog [41]. The drawback of these methods is the fact that they cannot cope with situations where there are more light sources in the scene. In order to compute the back scattered veil from the vehicle's own lights the authors compare the obtained image in fog conditions with reference images by means of an image correlation technique: SAD (Sum of Absolute Differences), ZSAD (Zero mean Sum of Absolute Differences), SSD (Sum of Squared Differences), ZNSSD (Zero mean Normalized Sum of Squared Differences), etc. The presence of fog in images is assessed by using a similarity threshold on the correlation score. A high correlation score implies that fog is present in the image, while a low correlation score corresponds to a fog free image. Several reference images are used and the highest correlation score obtained is used for fog detection. When other light sources are present in the scene the authors analyze the halo effect around these light sources. First the light sources need to be detected in the image. This is done by an image thresholding technique at high intensity levels (close to the maximum pixel value). The geometric properties of the resulting components are computed in order to exclude road markings and other bright features in the image that do not correspond to light sources. The remaining components in the image are then selected for further analysis and a tree of these components is built knowing the geometrical properties of each potential node. A segmented component must contain a light source and a large part of the light source's halo. The intensity of the halo must decrease smoothly when fog is present as compared to a fog free scene. This characteristic is used in order to detect the presence of fog. The main drawback of this method is that it cannot be used to asses the visibility distance of the driver, only a coarse classification can be made into fog or fog free scene.

A more different approach for fog detection and classification is used in [42] and [43]. Their algorithm is based on computing global image descriptors (Gabor filters at different frequency, scales and orientations) and a classification procedure in order to distinguish between fog and fog free scenarios. The images captured in fog conditions provide a power spectrum concentrated around the zero frequency, while image captured in normal (fog-free) weather conditions have much more high frequency components. Figure 3.10 present a result of this analysis. Similar observation can also be made in night-time foggy scenes. Figure 3.11 present such analysis in night time scenes. Based on this

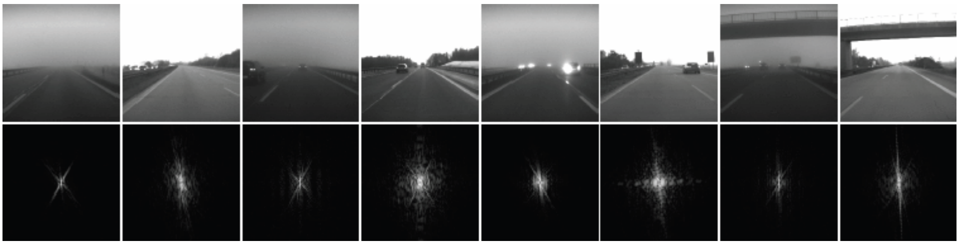


Figure 3.10: Example images of fog and fog-free scenes with their corresponding power spectra. The power spectra are displayed in logarithmic unit, with zero frequency in the middle of the image and higher frequencies at the border. The above images present clear differences in the spectra between similar scenes. While in fog scenes the frequency components are concentrated at the origin, in fog-free scenes they are broadly spread. [43]

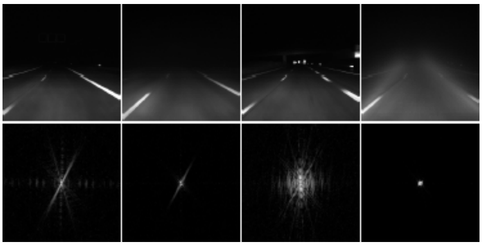


Figure 3.11: Examples of clear and fog images during night time. The first row presents camera images during night, the second row their corresponding power spectra. From left to right: No fog scene, fog scene, no fog scene with high beam and fog scene with high beam [43].

assumption the authors compute the power spectrum of each image, after a normalization step has been carried out, and perform a feature extraction on this power spectrum. Afterward, a two-stage feature reduction procedure is employed, which consists in sampling the spectrum in the frequency domain and subsequent feature selection based on Principal Components Analysis (PCA). A Gabor feature vector is obtained which is used in a classification procedure: Support Vector Machine (SVM) in [42] and a linear classifier based on Fisher's Linear Discriminant Analysis (LDA). The output of this learning process is depicted in figure 3.12. The algorithm uses images captured in highway driving scenarios. A training set was manually labeled in the following

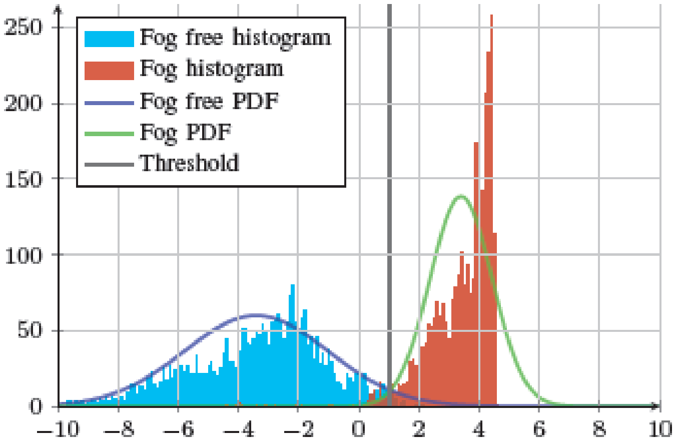


Figure 3.12: Threshold result after the training phase based on the Fisher LDA, from [43]. The plot presents the histogram of fog-free scenes in cyan and the histogram of fog scenes in red. The corresponding normal distributions are drawn in blue and green. The threshold corresponds with the intersection of the two normal distributions.

categories: Excluded, No Fog, Low Fog, Fog and Dense Fog. The algorithm assumes a planar road and the categorization was done based on the computed perspective projection from the known camera setup. A strength of this method is the fact that it is able to work in both day time and night time driving situations. On the other hand, only a coarse classification can be provided to the driver. The algorithm is not able to compute the visibility distance, nor it is able to work when there is an incoming or a preceding vehicle in the image.

3.2 Visibility Distance Estimation

In order to detect the presence of fog in a driving scenario one must estimate the visibility distance in the scene. Based on the computed visibility distance the algorithm should be able to classify the fog, present in the surrounding environment, into five different categories presented in table 3.1.

Taking into account Koschmieder's law presented in chapter 2, section 2.2.1 and applying this law in the image domain one obtains the

Visibility Distance		**Fog Category**
Min	**Max**	
1000 m	∞ m	No Fog
300 m	1000 m	Low Fog
100 m	300 m	Moderate Fog
50 m	100 m	Dense Fog
0 m	50 m	Very Dense Fog

Table 3.1: Fog Categories [6]

following mathematical representation of the image (I) captured in fog conditions:

$$I = Re^{-\beta d} + A_{\infty}(1 - e^{-\beta d}) \tag{3.10}$$

where β represents the extinction coefficient of the medium in front of the ego vehicle, R represents the fog free image and A_{∞} represent the sky intensity. This equation is the foundation of any fog detection algorithm and it states that the image captured in fog conditions is altered by an exponential factor $e^{-\beta d}$ according to the distance. The second term of this equation (3.10), $A_{\infty}(1 - e^{-\beta d})$, represents the fog addendum in the image or, in other words, the atmospheric veil obtained from the light scattered by fog between the scene and the observer (in our case the camera mounted inside the vehicle).

For detecting the presence of fog in the scene we have to detect the value of the extinction coefficient β. If $\beta = 0$ it means that there is no fog in the scene, while if a value larger than zero can be found $\beta > 0$ it means that fog is present in the outside environment.

Starting from equation (3.10) and replacing the expression of the distance to an image line from equation (2.21) we obtain the following equation:

$$I = Re^{-\frac{\beta\lambda}{v - v_h}} + A_{\infty}(1 - e^{-\frac{\beta\lambda}{v - v_h}}) \tag{3.11}$$

where $\lambda = \frac{\alpha H}{\cos\theta}$ is a constant that depends only on the camera parameters (H is the height at which the camera is mounted inside the vehicle, α is the focal length expressed in pixels and θ is the pitch angle of the camera), v denotes an image line and v_h is the position of the horizon line in the image, see figure 2.11 for an explanation of all these parameters. A prior calibration of the camera system is required in order to compute the value of the λ parameter. In real scenarios, when the camera is mounted in a production vehicle, an online calibration procedure should be implemented, in order to estimate the pitch angle θ and the camera height on a frame by frame basis. The value for the horizon line v_h can also be computed at calibration time, but a more robust approach would be to estimate the horizon line each frame and track / filter the position of the horizon line over time. An important observation is that fog is not uniform in all the image, it starts from the top of the image and at one point in the bottom of the image fog is no longer visible(the fog is very shallow or not existent in the close vicinity of the camera). This point is called the inflection point of the fog in the image. If we are able to detect such an inflection point in the image then we can asses the value of the extinction coefficient β and infer the fog conditions in the scene. This inflection point is found in the place where the second order derivative of the function I presented in equation (3.11) is equal to zero [15]. By taking the derivative of equation (3.11) with respect to v, we obtain the following:

$$\frac{\partial I}{\partial v} = \frac{\beta\lambda}{(v-v_h)^2}(R - A_\infty)e^{-\frac{\beta\lambda}{v-v_h}} \tag{3.12}$$

From a qualitative point of view, when fog density increases, the objects appear less visible in the scene, they get obscured more quickly by the luminance emanating from the sky, they tend to disappear more quickly inside the fog. Moreover, the maximum derivative decreases significantly and deviates more substantially from the horizon line [15]. By computing again the derivative of I with respect to v, we obtain the following equation:

$$\frac{\partial^2 I}{\partial v^2} = \frac{\beta\lambda}{(v-v_h)^3}(R - A_\infty)e^{-\frac{\beta\lambda}{v-v_h}}\left[\frac{\beta\lambda}{v-v_h} - 2\right] \tag{3.13}$$

The inflection point in the scene can be found where equation (3.13) equals zero, i.e. $\frac{\partial^2 I}{\partial v^2} = 0$. In this point we can compute the value of the β parameter. There are two possible solutions to this equation. The first on $\beta = 0$ is of no interest to us, since this implies that there is no fog in the scene. So the solution can be found in the second part of the equation:

$$\frac{\beta\lambda}{v - v_h} - 2 = 0 \tag{3.14}$$

This implies that

$$\beta = \frac{2(v_i - v_h)}{\lambda} = \frac{2}{d_i} \tag{3.15}$$

In equation (3.15) v_i represents the position of the inflection point in the image and d_i represents its distance to the camera, according to equation (2.21). By examining equation (3.15) we can observe that $\lim_{v_i \to v_h} \beta = 0$, so we can infer the fog conditions. Indeed, once the positions in the image of the inflection point line v_i and of the horizontal line v_h are found, the computation of the extinction coefficient β of the Koschmieder's law is straightforward. If v_i is greater than v_h fog will be detected in the image, otherwise we conclude that there is no fog in the scene. Considering equations (2.9) and (3.15) we are able to estimate the visibility distance in the image:

$$d_{vis} = \frac{3d_i}{2} = \frac{3\lambda}{2(v_i - v_h)} \tag{3.16}$$

Based on the obtained visibility distance we can classify the fog conditions according to intervals presented in table 3.1. For estimating the rest of the Koschmieder's law parameters we use I_i and $\frac{\partial I}{\partial v}|_{v=v_i}$, which represent the values of the function I and its derivative in $v = v_i$ [44].

$$R_{road} = I_i - (1 - e^{-\beta d_i}) \frac{v - v_h}{2e^{-\beta d_i}} \frac{\partial I}{\partial v}\bigg|_{v=v_i} \tag{3.17}$$

$$A_\infty = I_i + \tfrac{v_i - v_h}{2} \tfrac{\partial I}{\partial v}\bigg|_{v=v_i} \tag{3.18}$$

A_∞ is the sky's intensity and R_{road} represents the intensity of the road surface. Only the intensity of the road surface can be correctly estimated by using this method, due to the flat world hypothesis.

3.3 Our Method for Fog Detection in Road Scene Images

The overall architecture of our fog detection [6] framework is presented in Figure 3.13. The main contributions of our work are in the blue highlighted areas. Our method can use both gray scale or color images as input and is able to provide information about the presence of fog in the image and to estimate the maximum speed on the given road segment. As presented in the previous section, in order to detect if fog is present in the scene we must first detect the vertical position of the horizon line and inflection point line in the image. The existence of an inflection point in the image will provide the basis of our fog detection solution.

The first step of our algorithm is to apply a Canny-Deriche [45] edge detector on the input image. Then we estimate the horizon line and the inflection point in order to assess whether fog is present in the image. If fog is present in the image, i.e. $(v_i > v_h)$ then we perform the estimation of the visibility distance and based on this visibility distance we are able to classify the fog and to perform a recommendation about the safe speed on the given road segment. The methodology for computing the horizon line, inflection point line and the visibility distance are detailed in the following sections.

3.3.1 Horizon Line Estimation

Several methods can be employed for computing the horizon line in the image. The first one relies on using a simple calibration procedure

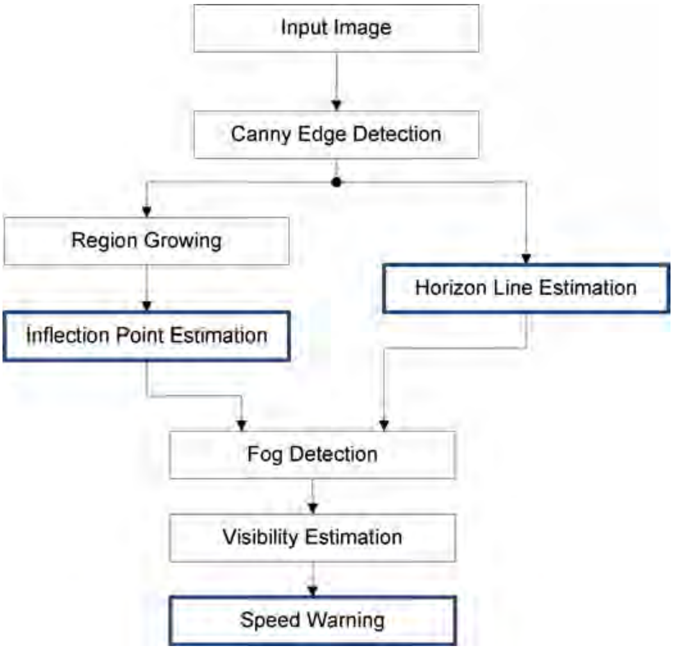


Figure 3.13: Fog detection System Architecture [6]

to compute the pitch angle of the camera [15]. An alternative, our choice, is to estimate the horizon line based on the image features. This will ensure a better result for the horizon line estimation in different traffic situations, but it does require the vehicle to be driven, at least for a small distance on a road with visible road margins and/or lane markings.

The horizon line in the image will be detected by finding the vanishing point of the painted quasi-linear and parallel (in 3D) road features such as lane markings. In [34] only the two longest lines are considered for finding the vanishing point. We employ a more statistical approach that uses more lines and was previously used to find the vanishing point of the 3D parallel lines from pedestrian crossings [46], with the goal of detecting the crossing. The following three main steps are applied for the detection of the vanishing point:

1. The first step involves selecting a set of relevant lines in the half lower part of the image (mostly road). The Hough accumulator

was built from the edge points present in the interest area. The highest m peaks were selected from the accumulator, and those that were having at least n votes were validated as the relevant lines.

2. A RANdom SAmple Consensus (RANSAC) [47] approach is applied to find the largest subset of relevant lines that pass through the same image point. A number of K (=48 for a success probability p=0.99 and percentage of good lines w=0.3) random samples of two relevant lines are selected. For each sample the intersection P of the two lines is computed and consensus set is determined as the subset of relevant lines that pass through P (within a small circle around P). The sample having the largest consensus set is selected.

3. The intersection points of each distinct pair of lines from the largest consensus set are computed. Finally, the vanishing point is computed as the center of mass of the intersection points.

Figure 3.14 presents the results of applying the horizon line detection process. The bottom image presents only the detected Hough lines that form the consensus set and their intersection points in green color.

Using this RANSAC approach for computing the vanishing point provides an additional benefit. The consensus score of the best pair of lines can be used to measure how good the current frame's vanishing point is in comparison to previous frames. A temporal filtering approach is then implemented in order to deal with scenes that lack painted lane markings: the vanishing point with the highest consensus score is selected from the last N frames. N can be chosen large enough to ensure the car has traveled along multiple road segments (hundreds or thousands of frames). This scheme would make the system cope with offline movements (even online after at least N frames pass) of the camera system, due to camera mounting/un-mounting, static uneven loads of the car, or uneven tire pressure. In addition this temporal integration makes the horizon line detection algorithm more stable.

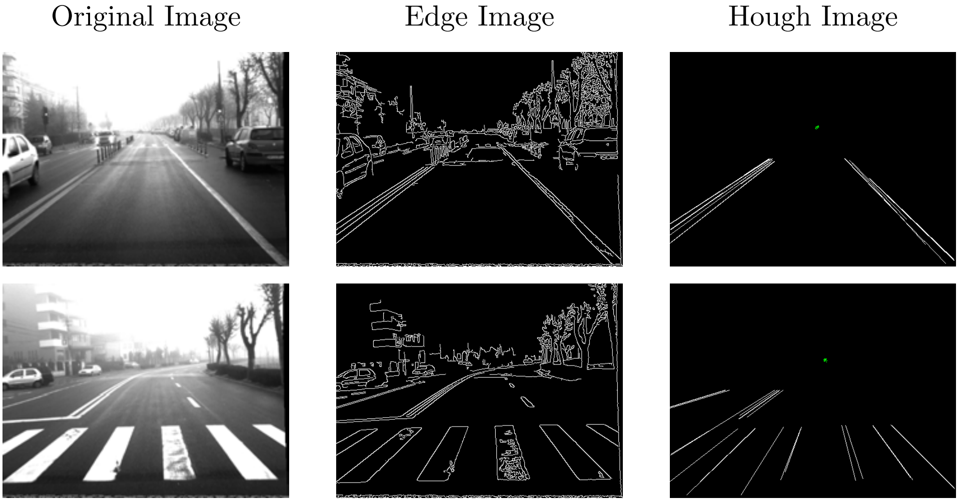


Figure 3.14: Horizon Line estimation algorithm based on Random Sample Consensus. The Hough Image also displays the intersection points of the consensus lines.

3.3.2 Region growing

The region growing procedure is required in order to estimate the position of the inflection point in the image. In order to achieve this we must first find an object whose luminance variation must be measured. In our scenarios this object is the road surface on which the vehicle travels. The reasons why the road is chosen is because it is always present in the scene and it is somehow homogenous in color, being a dark gray surface (almost black). In order to assure consistency with Koschmieder's law that assumes a dark object seen through fog, we assume that the road is homogenous in color and its luminance is only affected by the fog atmospheric phenomenon present in the scene.

The region growing employed follows the guidelines presented in [15]. The objective is to find a homogenous surface within the image, between the sky and the road surface, that displays minimal row to row gradient variation. The algorithm starts from a row in the bottom of the image whose pixels belong to the road surface. By tacking into account the geometry and position of the camera, the majority of pixels in this line belong to the road surface layer. A pixel from this row is considered to be a seeded pixel only if its intensity is close to the median of the gray levels on this row. The majority of pixels on this line are from the road surface, so they should be similar in gray levels. Only road pixels are taken into account, avoiding the region's increase in certain seeds on road markings and other objects from the road surface. Starting from a seed point only the three pixels above the current pixel are added to the region R. By using this approach one is able to "circumvent" the objects that must not be part of the region, such as cars, pedestrian, bicyclists and other vertical objects in the scene. The vertical objects usually present a lot of edge points, so this means that if a pixels belongs to the set of edge points E it must not be added to the region.

A pixel $P(i,j)$ is aggregated to the region R if it satisfies the following constraints:

- The pixel does not belong to the region:

  $$P(i,j) \notin R \tag{3.19}$$

  Only new pixels are incorporated into the region.

- The pixel is not an edge point:

  $$P(i,j) \notin E \tag{3.20}$$

  Usually edge points belong to vertical objects on the road surface, lane markings or other painted road objects, that need to be disregarded.

- The pixel is similar to the one located just below. This similarity is based on a Nagao type filter [48]. For this reason the following notation is used [49], G^j that corresponds to the distances between the current pixel $P(i,j)$ and the pixels just above the current pixel, where $j \in \{-1,0,1\}$. These distances must satisfy the following constraints:

$$G^{-1} = G^{1} \leq G^{0} \tag{3.21}$$

By imposing such a constraint one is able to sustain a minimum line to line gradient in order to satisfy the Koschmieder's law. Different thresholds are applied in the angle directions in order to allow the vertical movement for the region expansion algorithm. The typical values for this thresholds lie in the $[4, 8]$ interval. The values used for the G^{j} thresholds in the algorithm are $\{5, 8, 5\}$. Figure 3.15 illustrate the aggregation of a an image pixel to the region for this constraint.

- The pixel presents a certain similarity with the seed. if the seed is denoted by P_g the similarity is evaluated by computing the difference in gray levels between the investigated pixel and the seed.

$$P(i, j) - P_g \leq \rho n_r \min_{j \in -1,0,1} G^{j} \tag{3.22}$$

where $0 < \rho < 1$ and n_r is the number of lines between the seed P_g and the current pixel $P(i, j)$. In our work we have used the following values: $\rho = 0.8$, and $\min_{j\in-1,0,1} G^{j} = 5$. This constraint ensures that pixels with very different intensities from the seed are not aggregated to the target region. If we do not impose this constraint the black and white pixels could be added to the region in fewer than 32 iterations.

The algorithm ends when the top of the image is reached. The region detected by this algorithm contains both portions of the road and the sky, thus it suitable for the detection of the inflection point in the image. Figure 3.16 presents the results of our region growing procedure. The first column represents the original foggy image obtained in fog conditions, the second column represents the edge image, while the third column depicts the results of region growing. One can notice

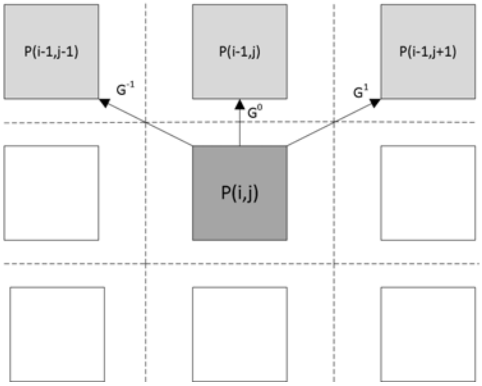


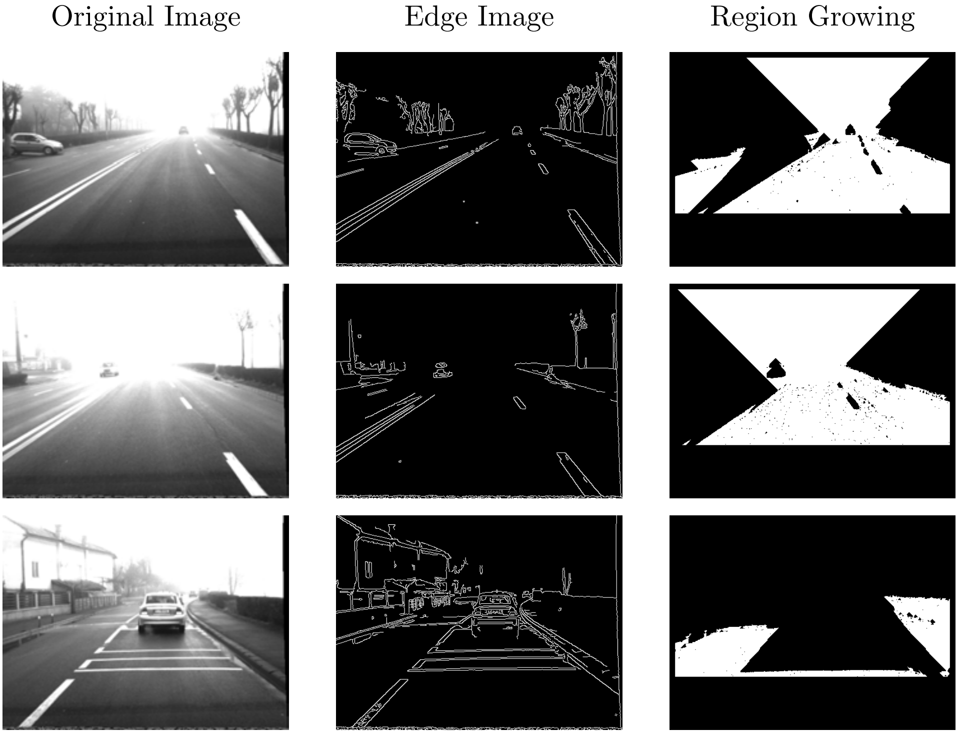
Figure 3.15: The forth condition for aggregation to target region R for a pixel $P(i, j)$

that in the cases when the field of view of the camera is not blocked by other vehicles or when there are fewer road marking, the algorithm is able to provide a region for inflection point detection, otherwise we are not able to obtain such a region, which means that we are not always able to infer the fog conditions.

3.3.3 Fog Detection and Classification

In this section we are dealing with the detection and classification of the fog from images. The idea is to analyze the previously obtained region and try to find the place where the fog and the road meet, i.e. the line in the image (on the road surface) where there is a sudden change in the fog's intensity. If we are able to find such a line then we can infer the fog conditions in the scene, otherwise we cannot estimate the visibility distance of the driver and the fog's density.

In order to compute the inflection point v_i we must first find the maximum band that crosses the region from top to bottom [15]. If we cannot find such a band, then we can conclude that the region growing procedure is not successful and we can assume that there is no fog in the image or that the algorithm s not able to make a prediction for the current frame. If the region growing algorithm was successful, then we compute the median value for each line of this band and we smooth



Original Image Edge Image Region Growing

Figure 3.16: The results of the region growing algorithm on traffic images

these values such that the obtained function is strictly decreasing. Next we extract the local maxima of the derivative of this function and compute the values for β, R and A_∞ for these maxima. The point that minimizes the square error between the model and the measured curve is considered to be the global inflection point v_i of the image. Another approach of finding the inflection point is to find the function that best fits our inflection point curve and deduce the inflection point of this function.

Once the horizon line v_h and inflection point v_i are computed, we can detect the presence of fog in the image and we can estimate the visibility distance d_{vis} from equation (3.16). If the position of the inflection point line is greater than the vertical position of the horizon line we are able to detect fog in the image. Based on the visibility distance estimation we are able to classify fog into the five different

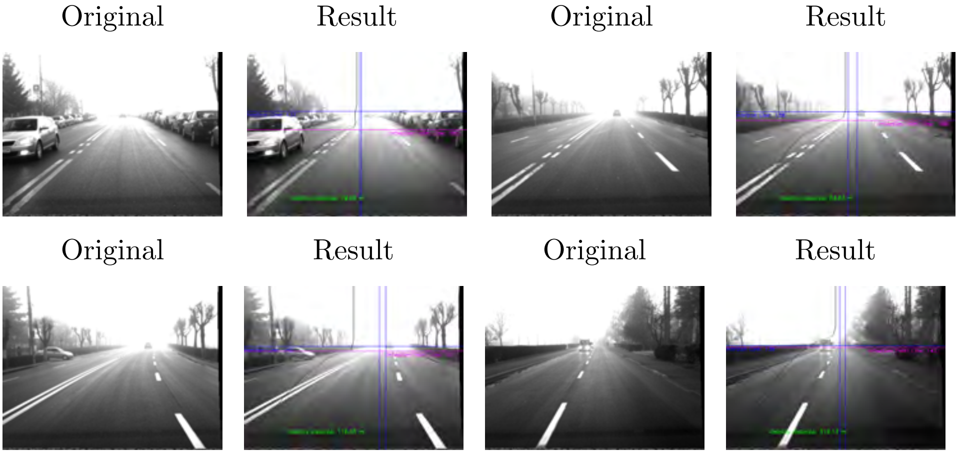


Figure 3.17: Fog detection results on traffic images [6]. The blue horizontal line represents the horizon line, the pink line represents the inflection point line and the two vertical blue lines delimit the vertical band. The black curve represents the smoothed median values from the vertical band. The visibility distance is displayed in green on the resulting image.

categories presented in table 3.1. Figure 3.17 present the results of the fog detection algorithm. The two vertical blue lines delimit the vertical band in the image, where the inflection point is computed. The black curve represents the value of the median intensities on each line from the vertical band (smoothed values). The position of the horizon line and the inflection point line are highlighted with blue and pink colors, respectively. The visibility distance is displayed with green color on the resulting image. We present four scenarios, from the city of Cluj-Napoca, corresponding to the four fog categories identified in table 3.1.

3.3.4 Speed Warning Recommendation

Many accidents that happen during fog conditions are caused by excessive driving. For this reason a lot of efforts have been made so that advanced driving assistance systems will be able to provide good maximum speeds for driving. A method for determining variable speed limits taking into account the geometry of the road, sight distance, tire-road friction and vehicle characteristics is presented in [50]. They construct an Intelligent Speed Adaptation (ISA) system based on a very detailed digital map. Although a lot of information is implemented in the ISA, the system is expensive and hard to retrofit on older vehicles. In order to avoid any accidents in fog conditions we consider that a "zero risk" approach would be more cautious [51]. Thus we consider the total stopping distance to be equal to the distance traveled during the reaction time and the breaking distance. So, for providing the driver with a good recommendation of safe driving speed we consider that the visibility distance d_{vis} computed by our method is given by the following equation:

$$d_{vis} = R_t v_r + \frac{v_r^2}{2gf} \tag{3.23}$$

The first term of equation (3.13) represents the distance traveled during the safety time margin (including the reaction time of the driver), and the second term is the braking distance. This is a generic case formula and does not take into account the mass of the vehicle and the performance of the vehicle's breaking and tire system. The terms are of the equation are described below:

- R_t is a time interval that includes the reaction time of the driver and several seconds before a possible accident may occur. NHTSA considers that the majority of accidents due to distracted drivers happen because of 3 seconds of driver inattention. Because we are aiming to obtain a cost effective solution for warning drivers about the speed that they should travel during fog conditions and because we do not take into account the geometry of the road since we do not use an augmented digital map, we have considered this interval to be equal to 5 seconds. This covers the interval of distracted drivers' inattention for most of the dangerous events that might occur during fog conditions.

- g is the gravitational acceleration constant, $9.8m/s^2$

- f is the friction coefficient. We know that during foggy weather the asphalt is not dry. For this reason we consider a reasonable value for f to be between $[0.3, 0.4]$

- v_r denotes the recommended driving speed.

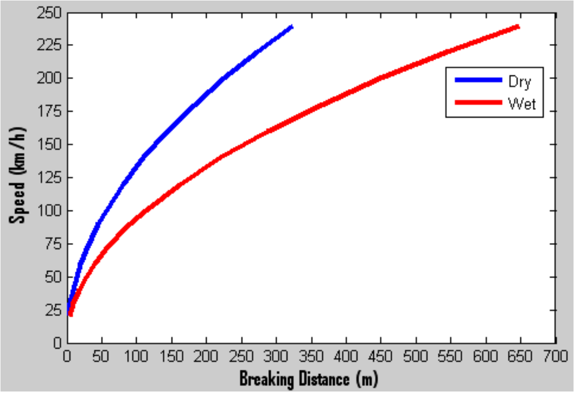


Figure 3.18: Braking distance on dry and wet asphalt. These values were computed with the following friction coefficients: 0.7 for dry asphalt and 0.35 for wet asphalt.

By solving equation (3.23) we obtain the following positive solution for v_r:

$$v_r = -gfR_t + \sqrt{g^2 f^2 R_t^2 + 2gfd_{vis}} \tag{3.24}$$

Figure 3.18 represents a plot of the braking distances on wet and dry asphalt according to the speed of traveling. The friction coefficient for dry asphalt was set to 0.7 and for wet asphalt to 0.35 [52].

Table 3.2 presents some maximum recommended speeds according to the fog density and the visibility distance measured by our algorithm. In addition we show the braking distances on wet asphalt. According to our model of computing the recommended speed, the driver has enough time in order to react and to break the vehicle in case of an emergency or hazardous event.

Visibility Distance	**Maximum Recommended Speed**		**Braking Distance**
	m/s	**km/h**	
20 m	3.61 m/s	13 km/h	1.90 m
50 m	8.09 m/s	29 km/h	9.54 m
100 m	14.15 m/s	51 km/h	29.21 m
150 m	19.22 m/s	69 km/h	53.87 m
200 m	23.66 m/s	85 km/h	81.65 m
300 m	31.34 m/s	113 km/h	143.25 m

Table 3.2: Speed Recommendations under Fog Conditions [53]

3.3.5 Fog Detection Results

In order to assess our method we have synthetically generated road traffic images using the GLSCENEINT framework [54]. Then we were able to add fog into these images using the Koschmieder's equation, by considering $A_\infty = 255$ and by varying β from 0.01 to 0.15. Figure 3.19 presents three scenarios for synthetic images. The first scenarios includes only the road, the second one includes the road and road side trees, while the third scenarios also includes vehicles in front of the ego vehicle. For $\beta = 0.03$ (moderate fog in the image) we can observe that in the first two scenarios the results are similar: visibility distance is of 202 meters and the maximum recommended speed is about 85 km/h. However, for the third scenario we are not able to compute the inflection point for $\beta = 0.03$, due to the presence of the vehicles on the road (we are not able to find a vertical band after the region growing process). In the dense fog scenario ($\beta = 0.06$) we are able to estimate the inflection point in all three scenarios (we present the result only for the first and third scenario). The visibility distance is approximately 82 meters and the maximum recommended speed is below 45 km/h (43.76 km/h). For $\beta = 0.09$ (dense fog situation), we show the results for the second and third scenario. The visibility distance is approximately 55 meters and the recommended speed is 31.81 km/h. For larger values of k the images look very similar, almost no difference exists between the three scenarios. In order to limit the oscillation due to driving scenarios

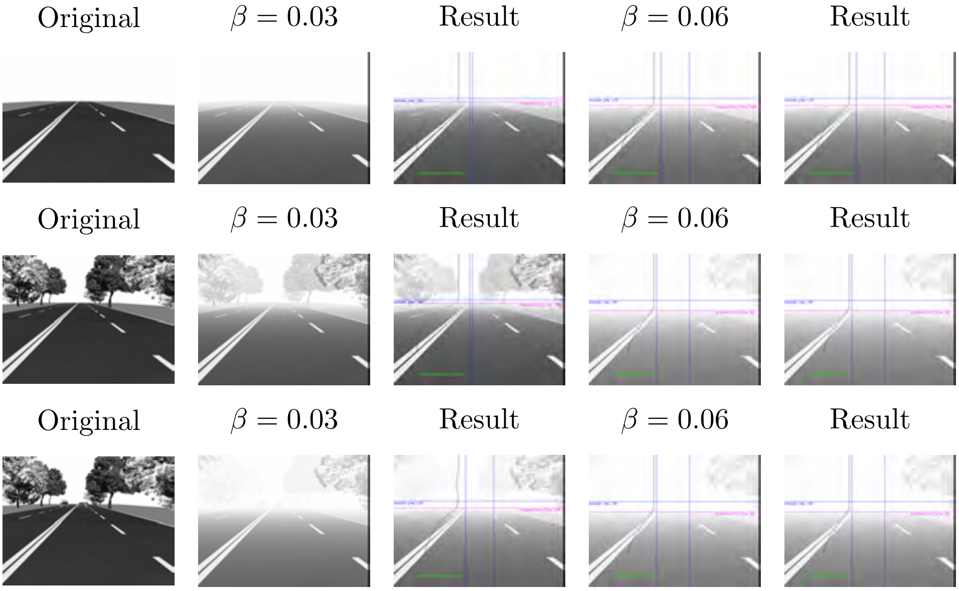
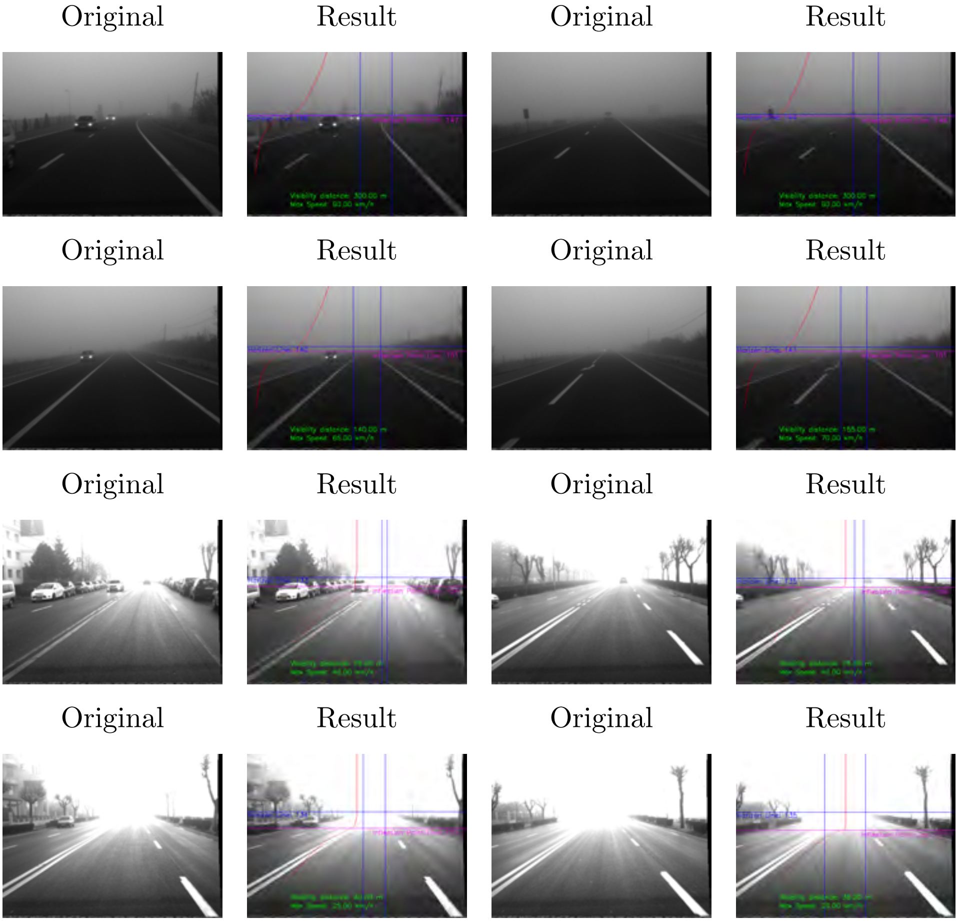


Figure 3.19: Results obtained on synthetic images. The blue horizontal line represents the horizon line, the pink line represents the inflection point line and the two vertical blue lines delimit the vertical band. The black curve represents the smoothed median values from the vertical band. The visibility distance is displayed in green on the resulting image.

we have limited the maximum recommended speed in fog condition to 90 km/h, and in order to provide a more driver friendly recommended speed we provide this speed in multiples of 5 km/h. The same applies to the visibility distance. Figure 3.20 present the results of our fog detection framework on real traffic images. These images were acquired with a vehicle equipped with JAI-A10-CL cameras during different fog conditions, in the city of Cluj-Napoca. From top to bottom we present different fog situations in accordance to the fog categories presented in Table 3.1. The first row presents two low fog scenarios: the visibility distance is of 300 meters and the maximum recommended speed is approximately 90 km/h. The second row shows moderate fog scenarios. The visibility distance is 140 and 150 meters, respectively and the maximum recommended speed is 85 km/h and 75 km/h. The third scenario represents a dense fog situation the visibility distance is below



Original Result Original Result

Original Result Original Result

Original Result Original Result

Original Result Original Result

Figure 3.20: Results obtained on real traffic images. The visibility distance and the maximum recommended speed are written on the resulting images.

100 meters (75 m) and the recommended speed is 40 km/h. And finally, the forth row presents two images captured in very dense fog; the estimated visibility distance is under 50 meters (40 and 35 meters respectively) and the maximum speed is of 25 and 20 km/h.

Table 3.3 presents the braking distances and the necessary time for braking in the above four scenarios. Form this table we can observe that the recommended speed is accurate enough in order for the driver to reduce the speed of the vehicle or even come to a complete stop so as to avoid any collisions with other vehicle or other traffic participants.

Fog Scenario	**Maximum Speed**		**Braking Distance**	**Braking Time**
	km/h	**m/s**	**m**	**s**
1. Low Fog	90.00	25.00	91.10	3.64
2. Moderate Fog	70.00	19.44	55.11	2.83
3. Dense Fog	40.00	11.11	17.99	1.62
4. Very Dense Fog	25.00	6.94	7.02	1.01

Table 3.3: Braking distance in the presented scenarios

Our fog detection algorithm was implemented in C++ and was tested on an i7 based PC running Windows operating system. The synthetic images have a resolution of 688 x 515 pixels and the average processing time is of 21 ms. The real traffic images were obtained with a JAI CV-A10CL camera. Their resolution is 512 x 383 pixels and the average processing time on one image is 18 ms.

3.4 Driving Assistance in Fog Conditions

In this section we describe the architecture of our fog assistance solution ported onto Android based smart mobile platforms [55]. Nowadays smart mobile devices, such as smart phones or tablets have become a trend even for driving assistance applications. Recently there is a growing interest in developing and deploying economically efficient safety functions for all types of vehicles; these functions are able to be retrofitted to older models as well, whereas the most advanced driving assistance systems that are today in production are dependent on expensive hardware, which is usually difficult, if not impossible, to deploy on older vehicles. Since the capabilities and processing power of smart mobile devices is continuously increasing, they can be easily integrated in today's vehicles. Due to the various driving scenarios and weather conditions, advanced driving assistance systems must cope with this diversity and must be able to perform well in any traffic condition. Their main focus is to detect and prevent any hazardous situation and to inform the driver about its occurrence. Examples of such driving assistance functions ported to smart mobile devices are: lane departure warning system implemented on a smart mobile device [56], [57]; monocular vision based obstacle detection [58]; stereo vision based obstacle detection [59], [60], [61]; pedestrian detection on android based smart mobile devices [62].

The functional architecture of our fog assistance solution ported onto android based smart mobile devices [55] is presented in figure 3.21.

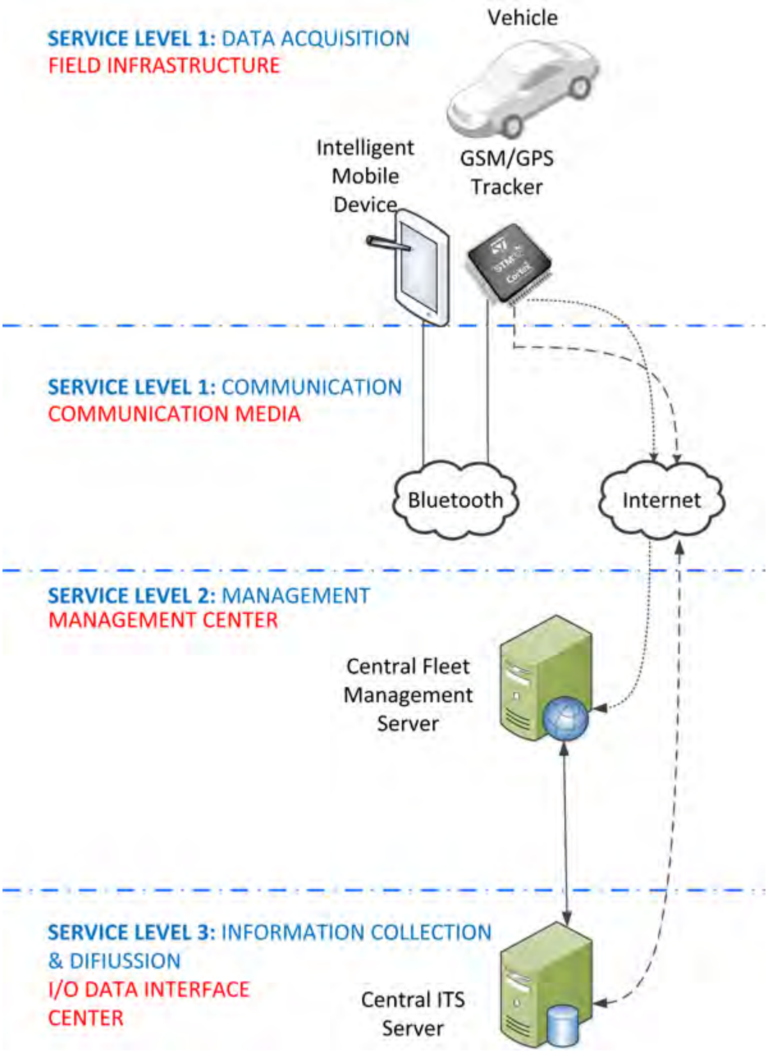

SERVICE LEVEL 1: DATA ACQUISITION
FIELD INFRASTRUCTURE
Vehicle
Intelligent Mobile Device
GSM/GPS Tracker
SERVICE LEVEL 1: COMMUNICATION
COMMUNICATION MEDIA
Bluetooth
Internet
SERVICE LEVEL 2: MANAGEMENT
MANAGEMENT CENTER
Central Fleet Management Server
SERVICE LEVEL 3: INFORMATION COLLECTION & DIFIUSSION
I/O DATA INTERFACE CENTER
Central ITS Server

Figure 3.21: Functional Architecture for Fog Assistance [55]

The hierarchical architecture illustrates the possible data flow in the system: from local level (the vehicle) to regional level (the central ITS servers) and back. From the point of view of the fog assistance solution the aim is to have the fog information (resulted from image processing algorithms) transmitted from vehicle (local) level to central (regional) level, where this information can be collected, and then disseminate the information to other interested traffic participants. The functional architecture consists of the following service levels: data acquisition, communication, management and information collection and diffusion.

The hardware infrastructure corresponding to the data acquisition layer consists in the stations placed on-board the vehicle, which are: a smart mobile device and a GSM/GPS tracker, which is an embedded device used here for communicating the fog information from the smart mobile device to the central ITS servers. The software solutions responsible for image acquisition and image processing for fog detection and fog removal are implemented on the smart mobile device.

The communication layer is responsible for assuring the inter-station communication. The communication between the on-board stations is achieved via Bluetooth, and the communication between the on-board stations and the central ITS servers is done via 3G/GPRS.

The central level is composed by two layers: (i) the management layer, a central fleet management server, responsible for collecting, storing and analyzing the data received from fleet and regular vehicles. It hosts an extended digital map that is capable of storing the fog information, (ii) the information collection and diffusion layer, (a central ITS server), which is responsible for collecting weather data from the central fleet management server, such as fog information, for performing various statistics. This information can then be disseminated to interested parties.

Figure 3.22 illustrates the in-vehicle on-board unit diagram and its communication means with the central ITS servers. The on-board unit is composed of the Android based smart mobile device endowed with a camera, and the GSM/GPS tracker, which for the fog application is used only as a communication mean, with the central ITS servers. As

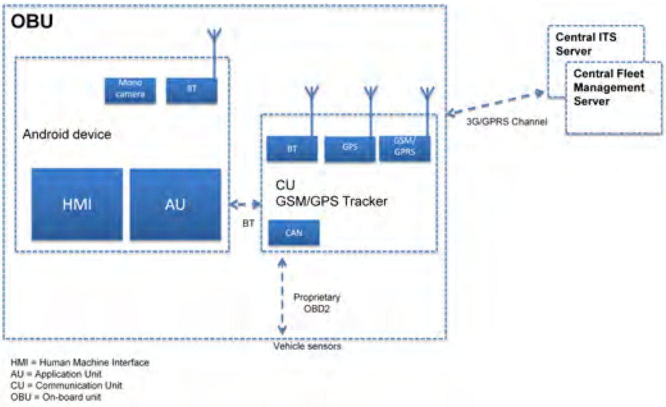


Figure 3.22: On-board unit diagram and communication protocol for fog assistance

illustrated in the system architecture, the fog assistance solution offers both local assistance, as well as remote assistance. The local assistance consists in two main functions: the fog detection solution that has been previously described in this chapter and a fog removal (image enhancement or contrast restoration) algorithm which will be detailed in the next chapter. The system alerts the driver about fog situations and displays the fog category and visibility distance, as well as the recommended speed on the given road segment in fog situations. The remote assistance consists in the dissemination of the fog information to a regional traffic information system server. This fog information can then be further transmitted to other drivers in order to warn them about the hazardous situation on the road.

Our android based fog assistance solutions has been implemented and tested on an Asus Transformer Pad with the following characteristics: Quad Core 1.2 GHz Cortex-A9 CPU, 1GB RAM, Android 4.2 OS, NVIDIA Tegra 3 chipset, 8MP auto-focus camera. The image processing algorithms have been implemented in native code making use of Android NDK and JNI; OpenCV library for Android has been used only for basic image processing functions. The execution time of the fog detection and fog removal algorithms are 80 and 60 ms, respectively, so the fog assistance solution is appropriate to run on a mobile device since the fog detection algorithm does not need to run continuously, but rather every 5 – 10 seconds, while the image enhancement solution can run on a frame-by-frame basis.

3.5 Conclusions

This chapter presented the most relevant approaches for detecting fog in day time and night time driving scenarios. The state of the art fog detection methods and approaches were presents in section 3.1. After investigating all these algorithms and implemented an tested some of them we can conclude that:

- Fog detection systems based on stereo-vision are not reliable due to the stereo reconstruction errors that occur because of the presence of fog. A more robust solution would be to remove the effects of fog from images and then perform stereo reconstruction.

- The fusion between camera and radar is not always a good solution for detecting the visibility distance because if there is no vehicle in front of the ego vehicle it means that the algorithm will not work. In addition, if the ego vehicle and the vehicle in front travel at the same speed, it does not mean that the perceived visibility distance by the radar sensor is the actual visibility distance of the driver in the current foggy scene.

- A fog detection algorithm must be able to provide the driver with some information about the density of the fog, to estimate the visibility distance of the driver and in addition to compute a safe driving speed for the given road segment under fog conditions.

- The most promising results for fog detection are based on monocular vision. There are two main approaches: one that works in the frequency domain and one that tries to estimate some visibility features in the image. The problem with the first approach is that although these methods are able to detect and classify fog they are not able to compute the visibility distance of the driver. The second approach seems more robust and mathematically sound: the fog detection is based on the computation of the inflection point of the fog in the image and when such inflection point is found the computation of the meteorological visibility distance is possible by taking into the account the flat world assumption.

- The road sky segmentation approach is not so robust and it can give false results in very sunny weather or when more vehicles are present on the road in front of the ego vehicle.

Based on the above conclusions we have described our improvements to the algorithms form the state of the art and we have a designed a complete solution for day time fog detection, visibility distance estimation and driving assistance in fog conditions. A new framework for fog detection from images acquired from a moving vehicle was designed and implemented. The goal of this framework is to provide the driver with the necessary information about the fog's density and the maximum visibility distance on the given road segment. Furthermore the framework assists the driver with a safe driving speed recommendation in order to avoid any collisions with other traffic participants.

In order to detect the presence of fog we must first estimate the horizon line and the inflection point in the image. Based on these two parameters we are able to infer whether the images are “plagued” with fog, to compute the extinction coefficient (β), to estimate the visibility distance and the density of the fog based on our fog categories presented in table 3.1.

The results are very good on roads that are not very crowded or when the field of view of the ego vehicle's camera is not occluded by other vehicles. One of our main contributions is the continuous estimation of the horizon line by using the RANSAC method and the temporal integration based on the consensus score. This approach proves to be very stable and provides accurate results when comparing to the estimation of the horizon line by using only the camera parameters (obtained during the offline calibration). By using the temporal filtering based on the consensus score, we are able to detect the horizon line even in tough scenarios where the lane markings are not visible, during curves, when the vehicle passes over speed bumps or even in situation where the road is not flat. Another important contribution is the temporal integration of the inflection point line. This provides us with the mean of estimating the density of the fog even in the situations where we are not able to detect the vertical band after the region growing process. Thus we are still able to provide the driver with some information about the fog density and the maximum safe speed on the given road segment. The algorithm used to estimate the maximum speed in fog situations proves to be accurate enough in order for the driver to reduce the driving speed as to avoid any collision with other traffic participants.

Since our method performs in real time, the fog assistance solution was also ported to an android based smart mobile device. Our system is able to detect fog by using the smart mobile device's camera, to estimate the visibility distance and to provide the driver with additional information like the fog's category and the recommended safe driving speed. In addition the fog information is sent to a regional traffic information system that can aggregate this and inform other traffic participants about the hazardous location on the road, where fog is present. This solution is a very convenient one that can be retrofitted on older vehicles in order to provide driving assistance functions in fog conditions.

In the future, we will focus on the robustness of this solution and the optimization of the fog detection algorithm, one important problem being the fix and rigid mounting of the smart mobile device inside the vehicle. Another method that we will investigate, considering that our target platform is a smart phone, is the use of the smart phone's accelerometer to compute the pitch angle. This alternative will require no offline calibration or additional time spent for horizon line detection. In order to improve the speed estimation during fog conditions we consider using the speed limits imposed by the national authorities in order not to recommend speeds that are higher than the imposed speed limits. This can happen in low fog situations when the vehicle is driven in a city and the speed recommendation exceeds the speed limit. Other future works include the integration of the fog detection system in a fixed traffic surveillance system in order to inform the drivers about the density of the fog and the safe speed on the given road.

For vehicles that are equipped with level 2 and 3 automation level, the ability to warn the driver in case of fog conditions, when the system cannot cope with the understanding of the vehicle surroundings at the required performance level, is crucial. In such situations the driver must be able to take over the driving process. For vehicles that are equipped with level 4 and 5 automation, performing only fog detection is not enough since the driver is no longer needed in level 5. For such autonomous systems it is crucial that the vehicle can travel even in the presence of fog, so other algorithms are also necessary in order for the vehicle to be able to perceive the surroundings even in tough fog situations. Such algorithms are described in the next chapter.

Chapter 4

Contrast Restoration of Foggy Images

Different natural phenomena can reduce the quality of images and diminish the visibility distance. Such natural phenomena are rain, haze, mist, fog and others. In these situations the driver's or the camera's visibility distance is decreased because of the absorption and scattering of light by the atmospheric particles, i.e. the light that is reflected from objects in the observed environment is attenuated by scattering along the line of sight of the camera.

In this chapter we present the state of art methods for restoring the contrast and enhancing the quality of the images captured in fog conditions. A complete driving assistance function in fog conditions or a highly autonomous vehicle, must also be able to understand the surrounding environment in adverse weather conditions, and this can be achieved by performing an image defogging technique, thus enabling other image processing applications and functions to perform well. In this book, the terms contrast restoration, image enhancement and image defogging are used interchangeably to refer to the same algorithmic procedure, that is to remove the effects of the fog condition form an input image and obtain a fog free representation of the observed scene.

Contrast restoration techniques can be used in wide range of applications starting from the computational photography field. If the restoration algorithm performs in real time, then image enhancement can also be applied as a pre-processing step for real time computer vision applications such as: basic image defogging, traffic surveillance, driving assistance (stereo reconstruction, object detection and recognition, etc.) and for autonomous vehicles.

4.1 Image Enhancement

Images taken in day time fog conditions usually present a white, light blur over the objects that are present in the scene. This blur varies with the thickness of the fog, i.e. the denser the fog is the bigger is the blur. Thus the quality of these images is drastically degraded. This weather phenomenon is especially dangerous in driving situations, because drivers tend to overestimate the visibility distance while traveling in fog conditions and drive with excessive speeds [28], [6]. Due to the presence of fog, the visibility distance decreases exponentially according to Koschmieder's law presented in the previous chapter, thus making fog one of the most dangerous weather condition for driving.

Some of the negative effects of fog on the quality of the image are the loss of contrast and the alteration of the natural colors from the image. In addition the scattering effect of the transmitted light causes additional lightness in some parts of the image [63]. These effect is called air-light or atmospheric veil. In order to overcome these impediments we must either try to change the operating parameters of the camera (exposure time, amplification gain, focus, etc.) or try to detect the presence of fog and remove its effects from the images.

There are several methodologies for performing image enhancement. The first one is to obtain two or more images of the same scene through polarizing filters with different degrees of polarization [64], [65]. The problem with these methods is the fact that they cannot be applied in a dynamic environment in which the changes in the scene are much faster than the (manual) rotation of the polarizing filters. A second class of methods uses images of the same scene taken in different weather and lighting conditions [66], [67], [68]. These methods exploit the differences between the obtained images of the scene, but even if their performance can better restore the contrast of the original fog free images, they cannot be applied until the user is able to capture images in these different weather environments. The third approach is to use only a single image of the environment and try to approximate or infer the fog addendum in the image. The term single image enhancement or defogging describes any method that removes the fog's effects by using only a single image, i.e. no other information is provided for the algorithm. Several single image based methods were proposed in literature for restoring the contrast of foggy images. These methods can be categorized in two groups: model and non-model based enhancement techniques. The next two sections focus on the single image enhancement algorithms and compare their utility and performance for driving assistance applications and autonomous driving.

4.2 Non-Model Based Image Enhancement Methods

Non-model based methods perform image enhancement relying only on the information obtained from the image; such as histogram equalization or adaptive histogram equalization [69], approaches based on Retinex theory [70]. Unfortunately, these methods do not maintain color fidelity and are not suitable for real time computer vision.

The authors in [69] are the ones to introduce the term of Contrast Limited Adaptive Histogram Equalization (CLAHE). This method can locally enhance the contrast of an image. CLAHE divides the images into several tiles (usually 8 x 8 regions of the image) and enhances the contrast of each tile such that the output of the histogram of the tile approximately matches a flat histogram. The method also requires the use of a bi-linear interpolation procedure to combine the neighboring tiles in order to eliminate artificially introduce boundaries. The main problem of such methods is the introduction of noise and texture in images. A similar method is presented in [71]. Their idea is to use a cumulative function to generate a gray level mapping of the local histograms. By modeling different cumulative functions the authors can achieve better results. The authors propose two types of cumulative functions: signed power-law accumulation and local mean replacement. First an estimate of the histogram is found by spatial smoothing and then the cumulative function converts this histogram into the mapping. However the results are not satisfactory on images captured in fog conditions.

Histogram equalization techniques can also be applied to color images. In such a case the adaptive histogram equalization procedure can be applied separately for each R, G and B color channel. This can lead to changes in the hue of the image. Better results are obtained in [72] where the image is first converted to Hue, Saturation, Intensity color space and the histogram equalization procedure is applied only for the Intensity component. However, even if the noise is reduced in the final image the color fidelity will not be maintained.

Retinex theory is investigated in [73], [70]. The multi-scale retinex (MSR) represents a non linear enhancement algorithm used to brighten up areas that have a poor contrast/brightness. The multi-scale retinex output is the sum of the outputs of several single scale retinex (SSR) by using different scales. Each color component or chanel of an image is separately processed in the case of single scale retinex. The basic form of a (SSR) for an image I is given by:

$$R_k(x,y)) = \log I(x,y) - \log[F_k(x,y) * I(x,y)] \tag{4.1}$$

where $R_k(x,y)$ is the single scale retinex output, $F_k(x,y)$ represents the k^{th} surround function, and $*$ represents the convolution operator. The surround functions is represented by a normalized Gaussian function:

$$F_k(x,y) = \kappa_k e^{-(x^2+y^2)/\sigma_k^2} \tag{4.2}$$

where σ_k is the Gaussian surround space constant and κ_k is selected such that

$$\iint F_k(x,y)dxdy = 1 \tag{4.3}$$

The final multi-scale retinex output is given by the weighted sum of the outputs of different single scale retinex functions. This is mathematically described by the following equation:

$$R(x,y) = \sum_{k=1}^{K} W_k R_k(x,y) \tag{4.4}$$

where W_k is the weight associated to $F_k(x,y)$ and K is the number of scales. The authors have chosen three scales for representing narrow, medium and wide surround functions that are sufficient enough in order to provide dynamic range compression. The values for these parameters are: $K = 3$, $\sigma_1 = 15$, $\sigma_2 = 80$, $\sigma_3 = 250$ and equal weights for the surround functions i.e. $W_k = 1/3$ for $k = 1, 2, 3$. Although the results are good for the enhancement of natural scenes acquired in poor lighting conditions, in the case of foggy road images the results tend to be very dark. Furthermore, the method can only remove fog of constant thickness from an image, which is not the case of road images and there is a slight dilution of color consistency.

A Wavelet based method is presented in [74]. The authors try to combine multiple images of the same scene in order to remove the blurring effects of the fog from images. Because wavelets capture edge information at multiple resolutions in an image the authors consider that by maximizing this edge information at each level of resolution a sharpened version of the initial image can be obtained. One problem of such methods is the enhancement of the noise present in images since maximizing edge information also maximizes the noise. This can be tackled by a soft thresholding procedure over the image with the effect of first removing the noise. The results are not so promising since a very dark representation of the environment is obtained so this type of methods are not suitable for driving assistance applications. Furthermore the resulting enhanced images exhibit a lot of noise and the color fidelity is not maintained.

Due to their nature, all these non-model based contrast restoration techniques are not suitable for the enhancement of images acquired in daytime fog in driving scenarios because on the one hand they introduce noise in images and on the other hand they are not able to maintain the color fidelity between images of the same scene with and without fog. Furthermore, they represent methods that are applied uniformly over the whole image and are not trying to infer the shape and thickness of the fog present in the acquired images.

4.3 Model based Image Enhancement Methods

Model based contrast restoration techniques can be divided in two categories: with given depth and unknown depth. When the depth is supposed to be known, this information can be used to restore the original contrast of the image. The authors in [75], [76] and [77] studied different haze removal approaches based on given depth information. The depth is inferred by using the altitude, tilt and position of the camera [75], through the manual approximation of the sky area and vanishing point in the captured image [76] or by approximating the geometrical model of the analyzed image scene [77]. Because the depth information is provided by the user in all these above mentioned approaches and because the obtained depth information is erroneous and unreliable, these methods are not feasible for real world applications.

Methods for restoring contrast without depth information are presented in [63], [78], [79] and [80]. They are based on a single image for performing image enhancement and a mathematical model that describes the fog in the image.

Oakley and Bu [63] assume that the distance between the camera and the points in the scene is approximately constant, such that the air-light on the whole image is constant and uniform. They model the obtained image in fog conditions as:

$$x' = x + c \tag{4.5}$$

where x is the original pixel value, x' is the modified pixel value and

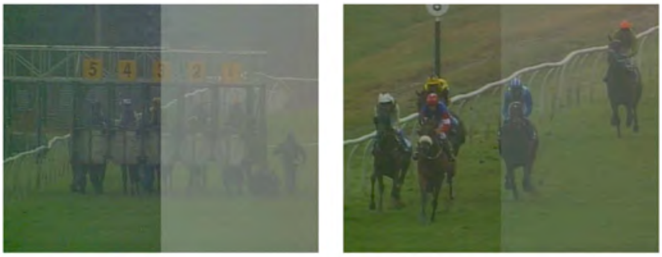


Figure 4.1: Results of the Oakley-Bu algorithm [63]. The right side of the images represents the original foggy scene and the left part is the obtained images after image enhancement.

c is the air-light. They propose to correct the simple contrast loss in foggy images by using the following equation:

$$y = m(x' - \lambda) \tag{4.6}$$

at each pixel of the image, where x' denotes the distorted pixel value, λ is an estimate of the air-light in the image, m represents a scaling parameter and y is the modified pixel value. The authors propose to take into account each color channel separately and thus to compute different values for the λ parameter for each red, green and blue color channel. For estimating the value of the air-light the methods attempts to minimize a global cost function on the whole image. This cost function is a scaled version of the standard deviation of the normalized brightness in the image. No image segmentation procedure is required in order to obtain the value of this air-light. This approach is only suitable for simple contrast loss correction of broadcast images, when the distance of objects present in the scene and the camera is rather constant, and fails in scenes where the distance to the scene points is not constant, such as driving scenarios. The results of applying this method to broadcast television images are presented in figure 4.1

The method proposed by Tan in [78] restores the contrast of the original image (only for color images) by using a cost function in Markov Random Fields setting for estimating the air-light. The authors start from the Lambert-Beer law for transparent objects that states that the light traveling through a material is attenuated exponentially:

$$I(x) = L_\infty \rho(x) e^{-\beta d(x)} + L_\infty(1 - e^{-\beta d(x)}) \tag{4.7}$$

The first term of equation (4.7) represents the direct attenuation and the second term represents the air-light.

- L_∞ represents the atmospheric light
- d represents the distance between an object from the scene and the observer (camera)
- β is the atmospheric attenuation coefficient
- ρ represents the reflectance of an object in the image

The authors introduce the notation of image chromaticity σ_c, where c represents the color channel (r, g, b) :

$$\sigma_c = \frac{I_c}{I_r + I_g + I_b} \tag{4.8}$$

When assuming that an object is situated at an infinite distance from the camera ($d = \infty$) the image chromaticity will depend only on the atmospheric light (L_∞). In this situation $e^{-\beta d} = 0$ and the air-light chromaticity is defined as:

$$\alpha_c = \frac{L_{\infty c}}{L_{\infty r} + L_{\infty g} + L_{\infty b}} \tag{4.9}$$

On the other hand, if there is no effect of scattering in the medium, i.e. there is no haze or fog in the atmosphere ($e^{-\beta d} = 1$) it results that the light chromaticity will depend only on the direct attenuation. Thus, the authors introduce the notion of object chromaticity:

$$\gamma_c = \frac{L_{\infty c}\rho_c}{L_{\infty r}\rho_r + L_{\infty g}\rho_g + L_{\infty b}\rho_b} \tag{4.10}$$

and equation (4.7) becomes:

$$I(x) = D(x)e^{-\beta d(x)}\gamma(x) + A(x)\alpha \tag{4.11}$$

where:

$$D(x) = L_{\infty r}\rho_r(x) + L_{\infty g}\rho_g(x) + L_{\infty b}\rho_b(x) \tag{4.12}$$

$$A(x) = (L_{\infty r} + L_{\infty g} + L_{\infty b})(1 - e^{-\beta d(x)}) \tag{4.13}$$

γ and α represent normalized color vectors while D and A represent scalar values. Furthermore, from the definitions of the chromaticity functions we know that:

$$\begin{aligned} \sum \sigma_c &= \sigma_r + \sigma_g + \sigma_b = 1 \\ \sum \alpha_c &= \alpha_r + \alpha_g + \alpha_b = 1 \\ \sum \gamma_c &= \gamma_r + \gamma_g + \gamma_b = 1 \end{aligned} \tag{4.14}$$

Usually the global value of L_∞ in the image can be obtained from the pixels that have the highest intensity in the input image. If one assumes that the sky can be seen in the image and that the image is not saturated, than the value of L_∞ can be obtained and thus the value of the light chromaticity α for each color channel can be immediately computed. A normalization procedure must be further applied in order to transform the color of the atmospheric light into pure white color, by dividing each color channel of the image intensity I_c by the obtained α_c and obtain the following equations:

$$I'_c(x) = \frac{I_c}{\alpha_c} \tag{4.15}$$

$$I'_c(x) = D(x)e^{-\beta d(x)}\frac{\gamma_c(x)}{\alpha_c(x)} + A(x) = D(x)e^{-\beta d(x)}\gamma'_c + A(x) \tag{4.16}$$

where I'_c represents the normalized input image and γ'_c is the normalized object chromaticity.

The vector form of equation (4.16) thus becomes:

$$I'(x) = D(x)\gamma'(x)e^{-\beta d(x)} + A(x) \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{4.17}$$

Tan [75] then considers the vector representation from equation (4.17) and by assuming that the value of L_∞ is known, it results that image enhancement becomes the estimation of $D(x)\gamma'(x)$ for the whole image in order to obtain an image that is not affected by the scattering effect induced by the fog. The problem of this estimation is straight forward. Since a single image is used to perform image enhancement and the depth is unknown, the authors infer the value of $e^{-\beta d(x)}$ from equation (4.13) as:

$$e^{-\beta d(x)} = \frac{\sum_c L_{\infty c} - A(x)}{\sum_c L_{\infty c}} \tag{4.18}$$

$$D(x)\gamma'(x) = (I'(x) - A(x) \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix})e^{\beta d(x)} \tag{4.19}$$

Thus, the problem becomes one of estimating the value of A at each pixel in the input image, which is easier to compute because it is independent of the reflectance of the objects present in the scene. An observation here is that the obtained image must have a better contrast than the original image captured in haze conditions and the variation of A must be smooth. By taking into account the smoothness constraint the authors model the air-light using Markov Random Fields. Unfortunately, this is an iterative process resulting in huge computation time (for an image of 600 x 400 pixels the computation time is between five to seven minutes).

Figures 4.2 and 4.3 present the results of this algorithm on an image obtained in fog conditions. The proposed method can produce halos at depth discontinuities due to the employed smoothness constrained. In addition the method can introduce several artifacts in the input image

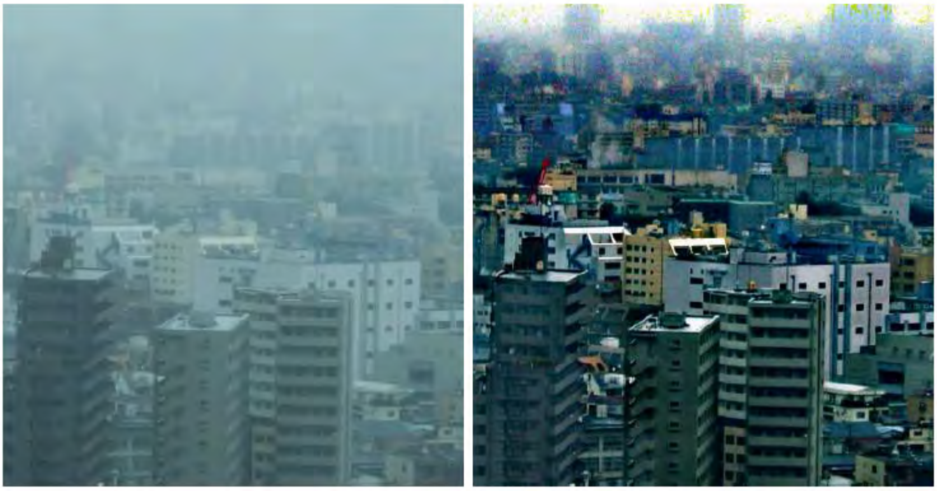


Figure 4.2: A foggy image is displayed in the left and the result of image enhancement with Tan's method [78] is displayed on the right.

and the processing time is not real time. The output images tend to have larger saturation values because the authors are focused on the visibility enhancement and they do not try to recover the original radiance in the scene. Although these observations prevent this method to be applied for real time advanced driving assistance functions or autonomous driving tasks, this method presents several key mathematical formulas that are used by all later image defogging methods.

In [81] and [79] the authors introduce the dark channel prior (DCP) terminology, which states that in most of non-sky scenes at least one color channel has very low intensity at some pixels. The authors assume that in a hazy or foggy image the intensity of the dark pixels in the dark channel is mainly contributed by the air-light. By combining this assumption with a mathematical model (Lambertian) that defines the haze or fog in the image and a soft matting interpolation method the authors are able to obtain a haze free image and produce an approximation of the depth map in the image. The authors start from the following mathematical Lambertian model for describing an image

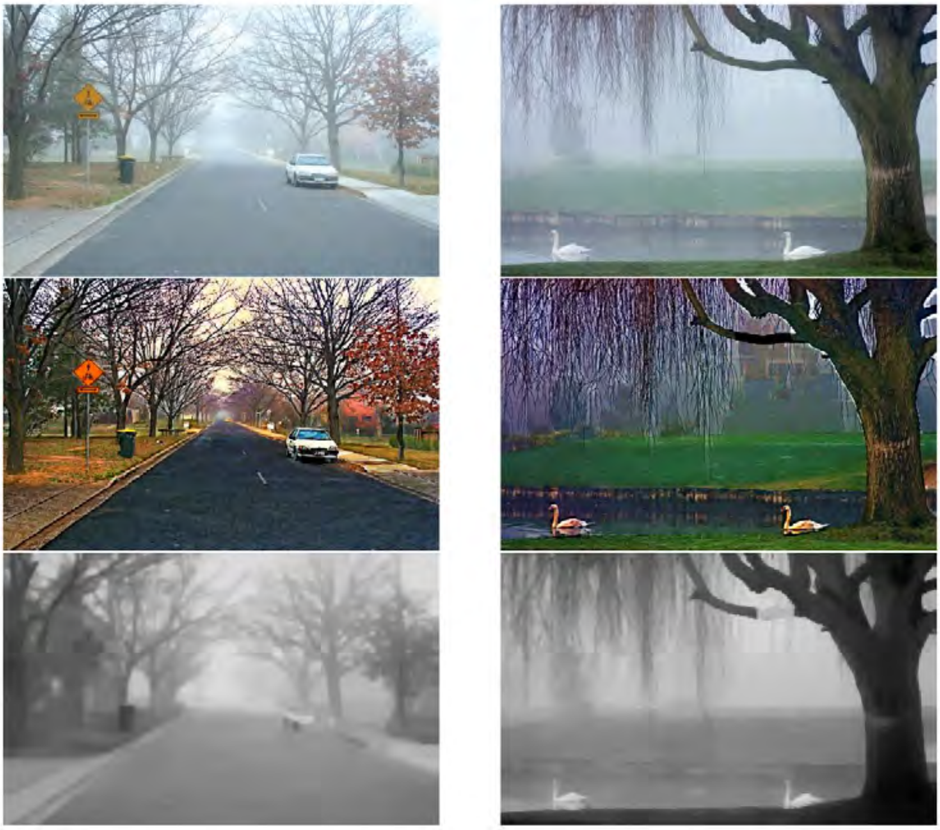


Figure 4.3: Top: input image, middle: the result obtained by Tan's method [75], bottom: the air light. Images taken from [75]

captured in haze or fog conditions:

$$I(x) = J(x)t(x) + A(1 - t(x)) \tag{4.20}$$

where:

- $I(x)$ represents the observed image
- $J(x)$ is the scene radiance
- $t(x)$ is the medium transmission
- A is the atmospheric light

Similar with the previous presented paper [78] the first term of equation 4.20 ($J(x)t(x)$) constitutes the direct attenuation of the air-light and the second term ($A(1 - t(x)$) represents the air-light component. The direct attenuation in the scene describes the scene radiance, while the air-light results from the scattering of light effect present in haze or fog conditions. In homogenous atmospheric conditions ($t(x) = e^{-\beta d(x)}$, where β is the scattering coefficient of the atmosphere), the dark channel of an image is simply defined as:

$$J^{dark}(x) = \min_{c \in \{r,g,b\}} (\min_{y \in \Omega(x)} J^c(y)) \tag{4.21}$$

where $\Omega(x)$ is a local patch in the image centered at x, c represents the color channel ($c \in \{r, g, b\}$). If J is a haze free image than J^{dark} has very low intensities and tends to be equal to zero. These low intensities in J^{dark} belong to shadows of cars, buildings or the inside of windows, dark objects or surfaces, or colorful objects in the scene. The dark channel prior is supported by a very wide statistical analysis made by the authors, which resulted that more than 90% of the intensities in a dark channel of a haze free image are below 25 (in image intensity values for an 8-bit image). On the other hand, when assessing this dark channel for an image acquired in haze or fog conditions (where the value of $t(x)$ is low) one can observe that the intensities in this dark channel are higher in the regions with denser haze or fog.

Taking into account the dark channel of an image captured in fog conditions, the authors estimate the transmission map ($t(x)$) in hazy images, considering that the transmission in a local patch is constant and that the transmission and surface shading are locally uncorrelated. For estimating the transmission map ($t(x)$) in a patch of the input image the minimum of equation (4.20) is considered:

$$\min_{y \in \Omega(x)} I^c(y) = \widetilde{t}(x) \min_{y \in \Omega(x)} J^c(y) + (1 - \widetilde{t}(x))A^c \tag{4.22}$$

and by dividing with the value of A^c and taking the minimum operation along each color channel the following equation is obtained:

$$\min_c(\min_{y\in\Omega(x)} \frac{I^c(y)}{A^c}) = \widetilde{t}(x) \min_c(\min_{y\in\Omega(x)} \frac{J^c(y)}{A^c}) + (1 - \widetilde{t}(x)) \tag{4.23}$$

Considering that the dark channel prior (DCP) of a fog free image $J(x)$ is zero and that the air-light A^c is always positive:

$$J^{dark}(x) = \min_c(\min_{y\in\Omega(x)} J^c(y)) = 0 \tag{4.24}$$

The transmission in a patch is given by the following equation:

$$\widetilde{t}(x) = 1 - \min_c(\min_{y\in\Omega(x)} \frac{I^c(y)}{A^c}) \tag{4.25}$$

This equation can handle the patches across the hole image, even in the sky regions. In order not to distort the haze free image the authors introduce a constant parameter w ($0 < w < 1$) into equation (4.25). This value is fixed at 0.95. The equation for computing the

$$\widetilde{t}(x) = 1 - w \min_c(\min_{y\in\Omega(x)} \frac{I^c(y)}{A^c}) \tag{4.26}$$

The value of the atmospheric light A^c can be computed as the maximum intensity in the image. The authors use the dark channel prior to improve this estimation of the atmospheric light. The top 0.1% of the brightest pixel in the dark channel are chosen and the pixel with the highest intensity in the input image is selected as the atmospheric light. In order to smooth the transmission map across the whole foggy image the authors use a soft matting procedure considering a Matting Laplacian matrix and a regularization parameter. The refined transmission map ($t(x)$) is obtained by minimizing the following cost function:

$$E(t) = t^T Lt + \lambda(t - \widetilde{t})^T(t - \widetilde{t}) \tag{4.27}$$

where L is the Matting Laplacian matrix and λ is the regularization

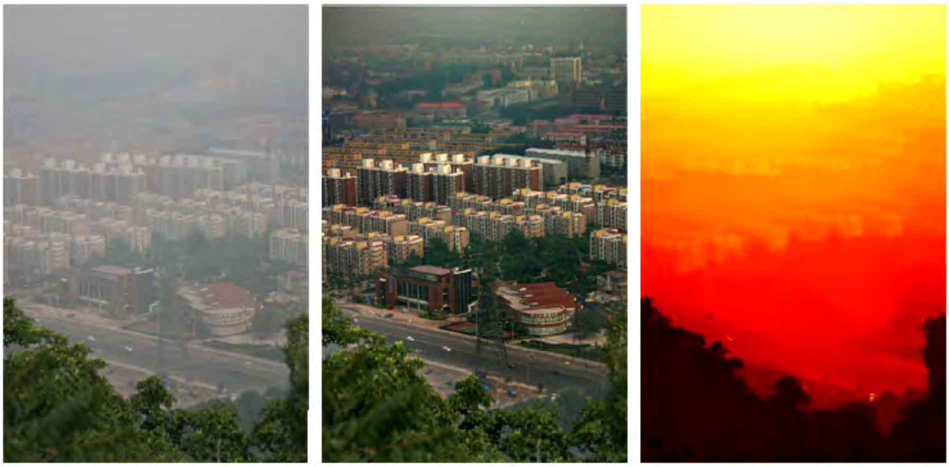


Figure 4.4: The results of haze removal using the method of He et. al. [79]. From left to right we present the input hazy image, the restoration result and the approximation of the recovered depth map.

parameter.

The recovering of scene radiance J^c is straightforward from equation (4.20). However when the transmission is close to zero the direct attenuation term is also close to zero. Therefore the transmission map is limited to a lower bound t_0 such that a small amount of haze is still present in the recovered scene:

$$J(x) = \frac{I(x) - A}{\max(t(x) - t_0)} + A \tag{4.28}$$

Figure 4.4 presents the results of using this method together with an approximation of the depth map in the scene. One drawback of this method is the fact that it cannot deal with gray scale images. Although the results look good in the lower part of the image the top part is not correctly restored. Figure 4.5 presents the results of He's algorithm with and without the soft matting refinement of the transmission map. It can be clearly seen that the soft matting refinement leads to the removal of several artifacts present in the reconstructed images. This approach is very costly from the processing time point of view, because of the used soft matting procedure.

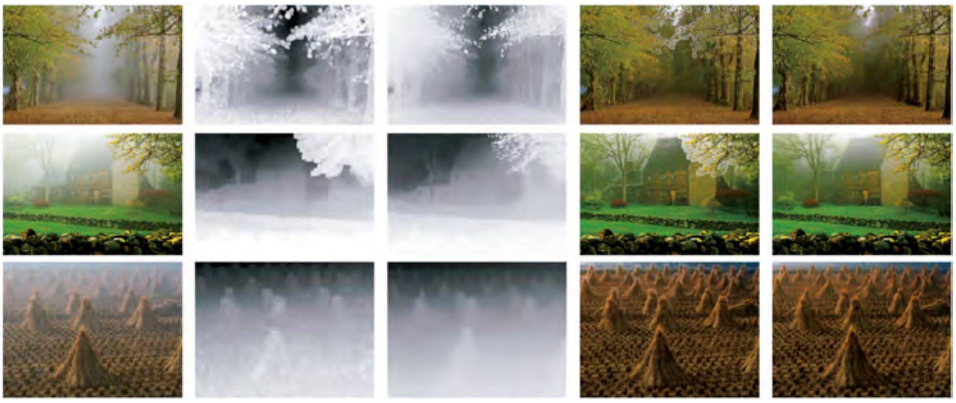


Figure 4.5: The results of haze removal using the method of He et. al. [79]. From left to right we present the input hazy image, estimated transmission map before soft matting, estimated transmission map after soft matting, restoration result without soft matting and restoration result with soft matting.

The main disadvantage of the two methods presented in [78] and [79] is the fact that these methods are not performing in real time (5 to 7 minutes and 10 to 20 seconds respectively for an image with a resolution of 600 x 400 pixels). This problem can be tackled by using more fast approaches that we will present in the next paragraphs and sections.

The authors in [24] start from the classical Koschmieder law [13] for restoring the original contrast. The basic idea is to recover the parameters of the Koschmieder's law and then use these parameters in order to compute the fog free representation of the input image.

$$I = Re^{-\beta d} + A_{\infty}(1 - e^{-\beta d}) \tag{4.29}$$

This law was described in chapter 2. The restored image R or the fog free image is obtained in the following manner:

$$R = Ie^{\beta d} + A_{\infty}(1 - e^{\beta d}) \tag{4.30}$$

An approximation of the βd parameter is used, i.e. $\beta d = s\frac{v_i - v_h}{v - v_h}$. A_{∞} is considered to be the highest intensity in the image.

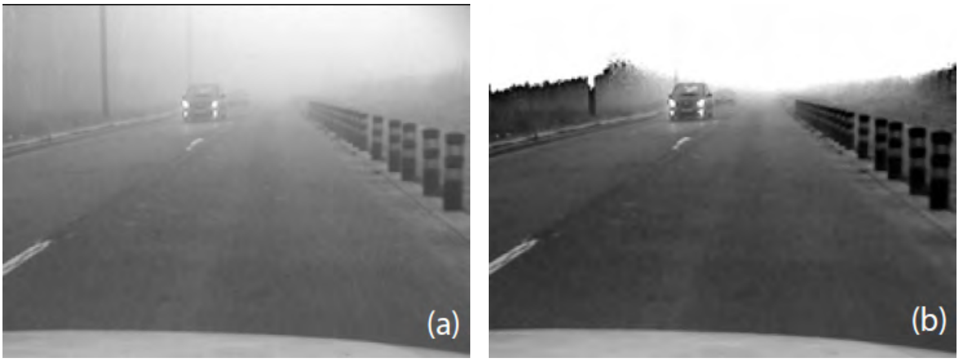


Figure 4.6: The results of contrast restoration by using the method proposed in [24]

The results of using this method are presented in figure 4.6. It can be seen that by using this method only the contrast of the road can be correctly restored, since the vertical objects present in the scene tend to be very dark (black objects). A somehow different methodology for modeling the scene depth is presented in [82], but there is no obvious improvement in the contrast restoration results. The vertical objects are still very dark, due to the fact that estimation of depth (transmission map) of the scene is erroneous. However, the authors try to turn this flaw into an advantage. In [83] the flat world assumption is used and only the contrast of the road can be reconstructed. Since the contrast of the vertical objects is falsely restored, being black and thus having an intensity close to zero, the authors use this in order to compute the free space in front of the vehicle by searching for the largest connected components in the scene. This process is called free space segmentation and it's results are presented in figure 4.7. Several applications of this contrast restoration algorithm, for assisting drivers in fog conditions are presented in [44]. These include the improvements to the extraction of road marking features, the detection of vertical obstacles and road traffic signs.

In [80] the authors try to infer the atmospheric veil (i.e. the fog addendum in the image) present in foggy images and use this atmospheric veil to obtain the fog free representation of the original scene.

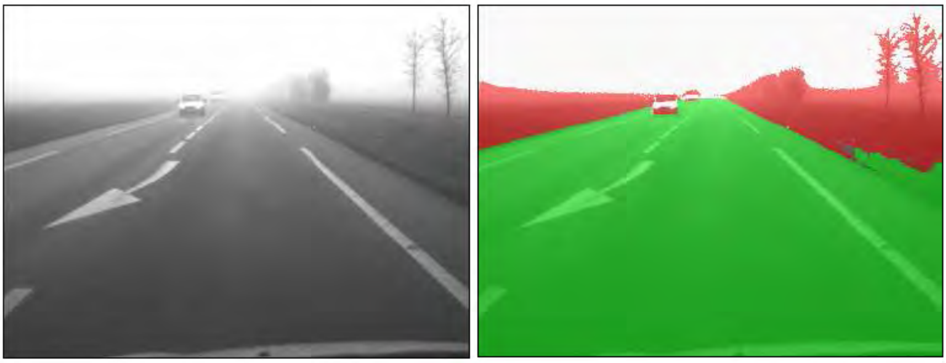


Figure 4.7: The results of using free space segmentation [83] after contrast restoration.

The atmospheric veil is described as:

$$V = A_{\infty}(1 - e^{-\beta d}) \tag{4.31}$$

Using the above notation for the atmospheric veil (V) the following representation of the fog free image can be obtained ((x, y) denote the pixel position in the image):

$$R(x,y) = \frac{I(x,y) - V(x,y)}{1 - \frac{V(x,y)}{A_{\infty}}} \tag{4.32}$$

Certain constraints are imposed on this atmospheric veil:

- The atmospheric veil should be greater or equal to zero: $0 \leq V(x,y)$
- The atmospheric veil cannot be higher than the minimum of each color channel of the input image $V(x,y) \leq W(x,y)$, where $W(x,y) = min(I(x,y))$
- The atmospheric veil must be a smooth function, except in the places were there exist depth discontinuities.

The authors propose to infer the atmospheric veil as a percentage (p) of the difference between the local average of $W(x,y)$ and of the local standard deviation of $W(x,y)$. Since the average filter does not

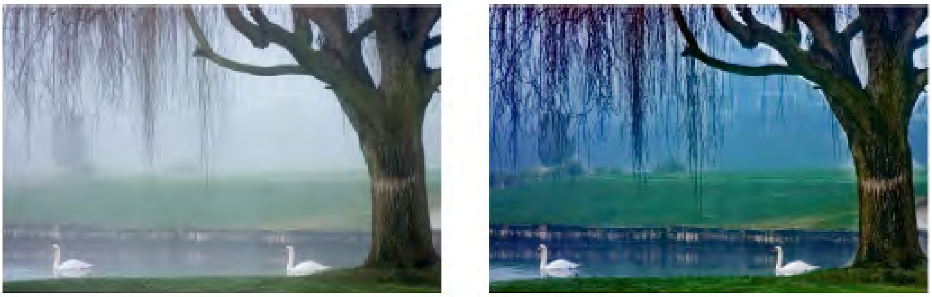


Figure 4.8: The results of contrast restoration with the method described in [80] on natural scenes.

preserve the edges in the image the authors use a median of median along the lines of the image. This median filter is than applied on the difference between W and the obtained median filtered image. Finally the authors introduce the factor p to control the strength of the image restoration process. The final version of the veil is presented in the following equations:

$$A(x,y) = median_{sv}(W(x,y)) \tag{4.33}$$

$$B(x,y) = A(x,y) - median_{sv}(|W - A|(x,y)) \tag{4.34}$$

$$V(x,y) = \max(\min(pB(x,y), W(x,y)), 0) \tag{4.35}$$

This method contains only two parameters, i.e. the size of the median filter sv and the percentage p of image enhancement process. Figure 4.8 presents the results of applying this method on natural images while figure 4.9 presents the results with different parameters on traffic images (gray scale). The result seem good, the road is darkened and depending on the used parameter a better representation is obtained. The higher the sv parameter the better the quality but a worse processing time is achieved. This method is also called in literature the no-black pixel constraint (NBPC) image enhancement.

The method is compared to other relevant contrast restoration

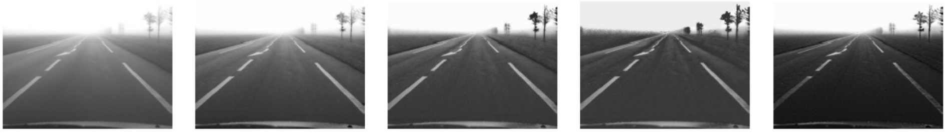


Figure 4.9: The results of contrast restoration with the method described in [80] on traffic scenes with different parameters. From left to right the original foggy image and the results with: $p = 0.7$ and $sv = 61$, $p = 0.9$ and $sv = 61$, $p = 0.98$ and $sv = 61$, $p = 0.90$ and $sv = 21$

methods and obtains similar or better results. The used metrics are the rate of new visible edges, the percentage of pixels that become completely black or completely white after restoration and the ratio of gradients at visible edges. These will be used in the next sections for comparing the contrast restoration results of different methods.

A new constraint over the atmospheric veil, called the planar assumption constraint is introduced in [84] and [85]. This constraint prevents the over estimation in the bottom of the image. The name of the method is no-black pixel constraint with planar assumption (NBPC + PA). The constraint equation is presented below:

$$V(u,v) \leq A_{\infty}(1 - e^{\frac{ln(0.05)\lambda}{d_{min}(v-v_h)}}) \tag{4.36}$$

where A_{∞} is the intensity of the sky in fog conditions, λ is a coefficient that depends only on the camera parameters, v is an image line and v_h represent the position of the horizon line in the image. The planar road constraint assumes that the road is a plane until a certain distance and the authors set this distance $d_{min} = 50m$. Having this constraint in mind, if the meteorological visibility distance is lower than $50m$, the image enhancement will be partial at short distances, even if the percentage of restoration is $p = 100\%$. The restoration results with this new constraint added are superior than the ones previously presented, in the case of road images, see figure 4.10 for more details.

Another problem with this constraint is that the computation of the atmospheric veil does not yield a smooth function over the entire

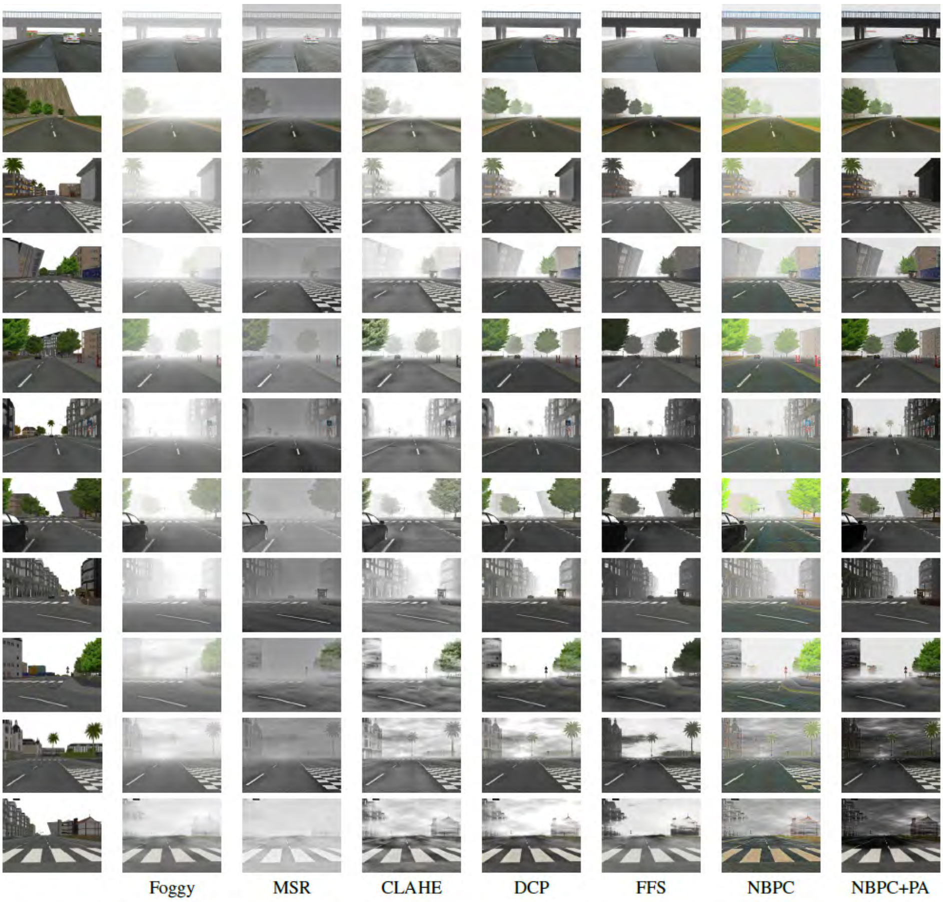


Figure 4.10: Contrast restoration with the method described in [85]. From left to right the original synthetic image without fog, the image with fog, the images enhanced using multi-scale retinex, adaptive histogram equalization, dark channel prior, free-space segmentation, no-black-pixel constraint and no-black-pixel constraint with planar scene assumption. The images are taken from the FRIDA data-set [86].

input foggy image. The function depicted in equation 4.36 present a cutoff around the position of the horizon line (v_h).

The authors introduce a new metric for image enhancement: the average absolute difference between the synthetic image without fog and the image obtained after restoration. In order to accomplish this they introduce the Foggy Road Image DAtabase (FRIDA) [86], which is a data set of synthetic images of traffic scenes. The FRIDA data set contains eighteen synthetic images that try to simulate real traffic scenes; The FRIDA2 data set contains 66 images of traffic scenes. The resolution of these images is 640×480 pixels. Four different types of fog were added to these images:

- homogenous (uniform) fog: Koschmieder's law applied directly on the input image with a meteorological visibility distance of 80 meters.

- heterogeneous β fog: Koschmieder's law applied with different values of β depending on the pixel's position, A_∞ is kept constant. These values are were obtained by using a Perlin's noise [87], [88] between 0 and 1. The average meteorological visibility distance was set around 80 meters.

- cloudy homogeneous fog: Koschmieder's law applied with β kept constant for a meteorological visibility distance of 80 meters and different values for A_∞. The Perlin's noise in this case is used for the sky's intensity thus producing a cloudy sky.

- cloudy heterogeneous fog: Koschmieder's law applied with different values for both β and A_∞. In this case two independent Perlin's noises are used and the average meteorological visibility distance is also 80 meters. For more information on this data set, please refer to [85] and [86].

The restoration assessment is carried out over these types of fog and on the entire data set.

4.4 Contrast Restoration based on Koschmieder's Law

Our method for contrast restoration starts from the well known Koschmieder's law [13]. In order to model the fog in an image Koschmieder equation is applied directly to the response function of a camera, in order to model a mapping from the scene luminance to image intensity (see chapter 2, section 2.2.1 for more details). So, the following representation of the input image (I) is obtained:

$$I = Re^{-\beta d} + A_\infty(1 - e^{-\beta d}) \tag{4.37}$$

where:

- R represents the pixel intensity of the image without fog, i.e. the fog free representation of the analyzed scene
- A_∞ is the intensity of the sky in fog conditions
- β is the extinction coefficient of the medium
- d is the distance to the camera (for each pixel)

The first term of the equation states that an image captured in fog conditions is altered by an exponential factor $e^{-\beta d}$ according to the distance. The second term of the equation, $A_\infty(1 - e^{-\beta d})$, represents the atmospheric veil obtained from the light scattered by fog between the scene and the camera sensor, or in other words the fog addendum in the image.

By using a single image we are not able to compute the depth at every pixel in the scene, only an estimation to an image line that belongs to the road can be made (planar road assumption). Other methods for computing the depth from a single image, like structure from motion based on optical flow, also fail in case of foggy scene. Hence, we introduce the notion of atmospheric veil [80] (V) as (similarly to [80]:

$$V = A_\infty(1 - e^{-\beta d}). \tag{4.38}$$

This veil represents the fog addendum in an image. Our approach is to estimate the atmospheric veil and then to use it in order to compute the original fog free image.

The veil equation (4.38) implies that $e^{-\beta d} = 1 - \frac{V}{A_\infty}$, so by substituting in equation (4.37) one obtains the following representation for the input image:

$$I = R(1 - \frac{V}{A_\infty}) + V. \tag{4.39}$$

At this moment we are able to compute the original fog free representation of the input image as:

$$R = \frac{I - V}{1 - \frac{V}{A_\infty}}. \tag{4.40}$$

The above equations demonstrate that one approach to obtain the fog free image, without knowing the depth of the scene, is to estimate the atmospheric veil. The intensity of the sky (A_∞) is considered to be equal to 255 (for an 8-bit input image), or it can be inferred as the maximum intensity in the image. Another approach [79] is to use a rectangle in the top of the image and to compute the average intensity inside this rectangle as the sky's intensity.

4.4.1 Inferring the Atmospheric Veil

Usually, a visibility enhancement method must be able to work with both gray scale and color images. In order to compute the atmospheric veil for color images, the used input of our algorithm is a gray level image W that consists in the minimum of each color channel (R, G, B). This image is called the dark channel prior (DCP) of a color image and is computed similarly to method presented in [79]. The obtained veil V provides the amount of white that must be subtracted from each color channel. In case of gray scale images, the W image is equal to the input foggy image.

$$W = \begin{cases} min(R, G, B) & \text{, for color images} \\ I & \text{, for monochrome images} \end{cases} \tag{4.41}$$

In order to infer the atmospheric veil we must first examine some properties that it must have. First, the photometric constraint must be introduced [78], V must be higher or equal to zero and V must be lower then W:

$$0 \leq V \leq W \tag{4.42}$$

Another property of the atmospheric veil is that V must be a smooth function (in most of the cases). In [84] a new constraint is considered, the no black pixel constraint. This constraint states that the local standard deviation of the enhanced pixels around a given pixel position must be lower than it's local average. If we consider that in the vicinity of a pixel the scene depth is constant and the fog is homogenous than it results that the local averages of the original and enhanced images are related. The same applies for the standard deviation of these images. Thus we can infer that the veil is smaller or equal to the difference between the average and the standard deviation of the input image W:

$$V \leq Average(W) - std(W) \tag{4.43}$$

The problem concerning the local average filter is that it does not preserve edges. A better approach is to use a median filter, not only because it preserves edges, but it also removes noise from the input image. Thus a median filter with a variable size (k) will be applied on the W image instead of using the classical average.

$$M = median_k(W) \tag{4.44}$$

For computing the standard deviation on the obtained W image we use the classical standard deviation computed in each pixel of the image, rather than computing the median on the absolute difference between the W image and the previously computed average, as in [80].

$$Std = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(W_i - \mu)^2} \tag{4.45}$$

By tacking all these constraints into account, we consider that only a percentage p will be used to calculate the value for the atmospheric veil in each pixel. This percentage is used to control the strength of the restoration process. The usual values for p are set from 85% to 99%. A lower value for p is used for gray scale images, while a higher value is used for color images. Thus, in order to satisfy the constraint from equation (4.42) the equation for computing the atmospheric veil becomes:

$$V = max(min(p|M - Std|, W), 0) \tag{4.46}$$

Once the atmospheric veil is computed we can derive the fog free representation of the input image simply by applying equation (4.40) for every pixel in the image. Figure 4.11 presents the general architecture of our image enhancement system. The presented algorithm is able to enhance both gray scale and color images.

The last step in the enhancement process is to perform a normalization on each color channel of the restored image, for obtaining an image having natural colors.

4.4.2 Median Filters for Image Defogging

In order to find the best solution for single image defogging, a short analysis of three median filters was carried out, and after this analysis we have deduced which is the most suited filter that can be applied for contrast restoration of foggy images [89]. In figure 4.12 we present the atmospheric veil obtained using three median filters: the classic square median filter (M), a median filter applied only along the lines of the image (ML) and a median filter applied only along the columns of the image (MC). All filters have the same kernel size $k = 39$, a square kernel for the classic median filter ($[k, k]$) and two vectors for the median filter along lines ($[k, 1]$) and median filter along columns ($[1, k]$). As one can observe the most consistent veil image with a fog representation is the median filter along columns of the image. One would expect that fog is more present in the top part of the image and tends to disappear in the bottom part of the image which is closer to

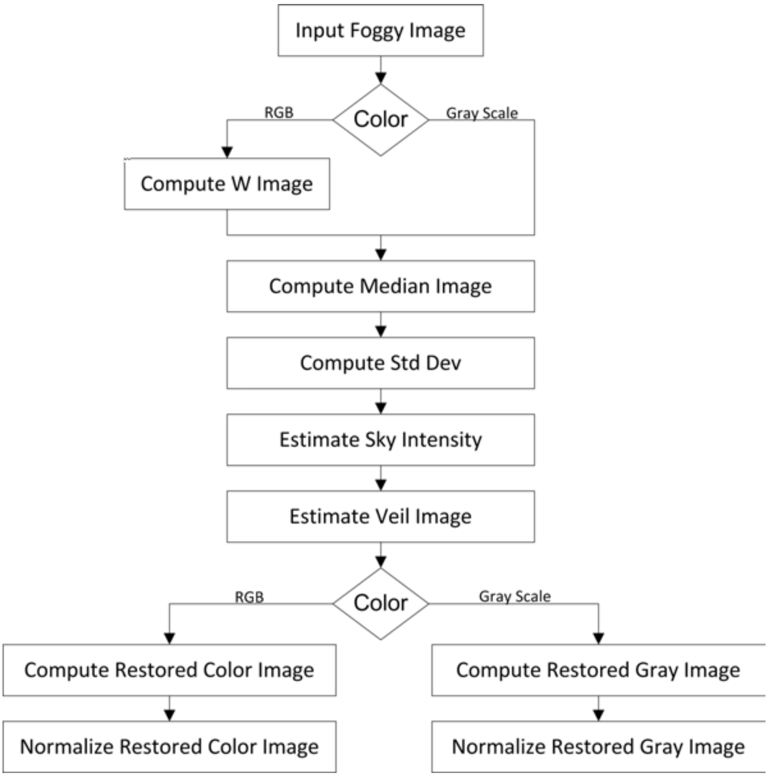


Figure 4.11: The General Architecture of the Image Enhancement Algorithm

the camera. This is true especially for road images, when the camera mounted inside a vehicle captures images from the small vicinity in front of the vehicle up to infinity. For this reason it is natural to introduce another constraint for obtaining the atmospheric veil which should prevent overestimation of the veil at small distances in front of the camera. In addition the median filter applied on the columns of the image makes the atmospheric veil adaptable according to the fog's density.

In order to evaluate our image enhancement algorithm we require images of the same scene with and without fog. Such images are very hard to acquire in practice: first, the camera has to be mounted in exactly the same position; second, the camera must have exactly the same configuration and orientation in both scenarios and third, the illu-

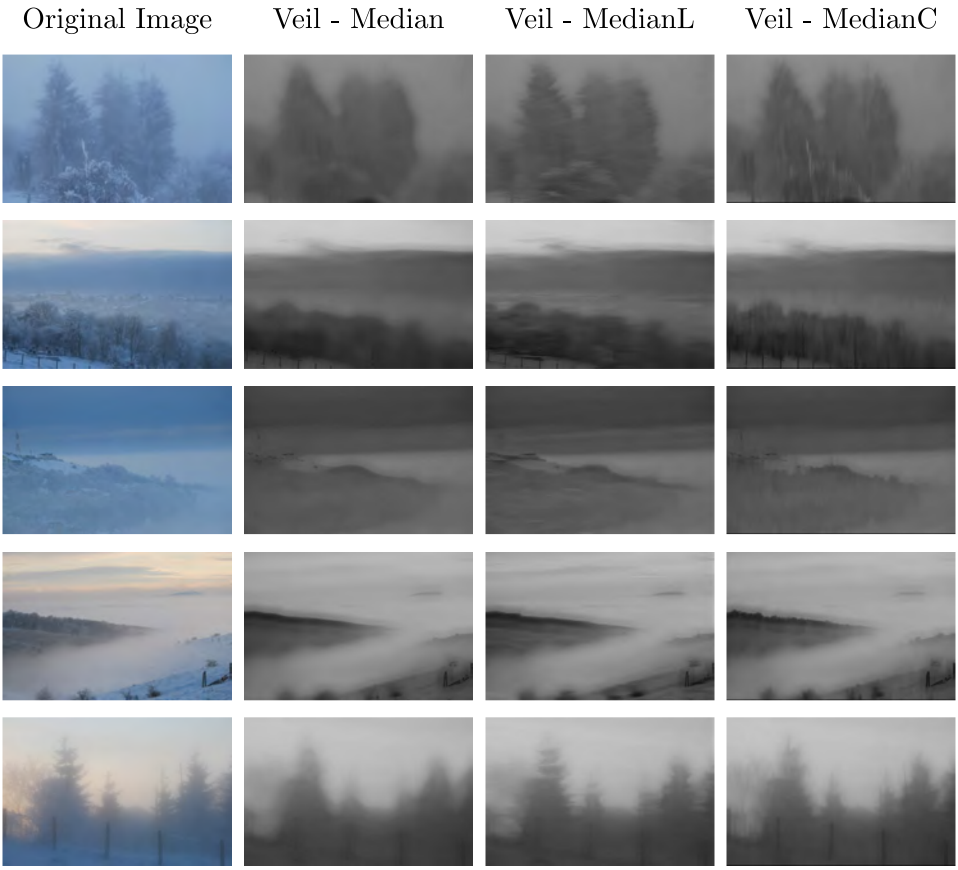


Figure 4.12: From left to right, the original image, atmospheric veil obtained using the classic median filter, median filter along lines and median filter along columns (the parameters used are $k = 39$, $p = 95\%$)

mination and environment conditions must be the same in both scenes (with and without fog). Taking into account all these reasons, we will first present the results on synthetic images from the FRIDA (Foggy Road Image DAtabase) data set [85], and then we will investigate the restoration results on natural scenes and real traffic images.

Figure 4.13 presents eight images from this data set that were enhanced with our algorithm, using the three methods for computing the atmospheric veil. The columns represent the original images without fog, the image with added fog and the three enhanced images with the proposed algorithm by using the classical median filter (M), the me-

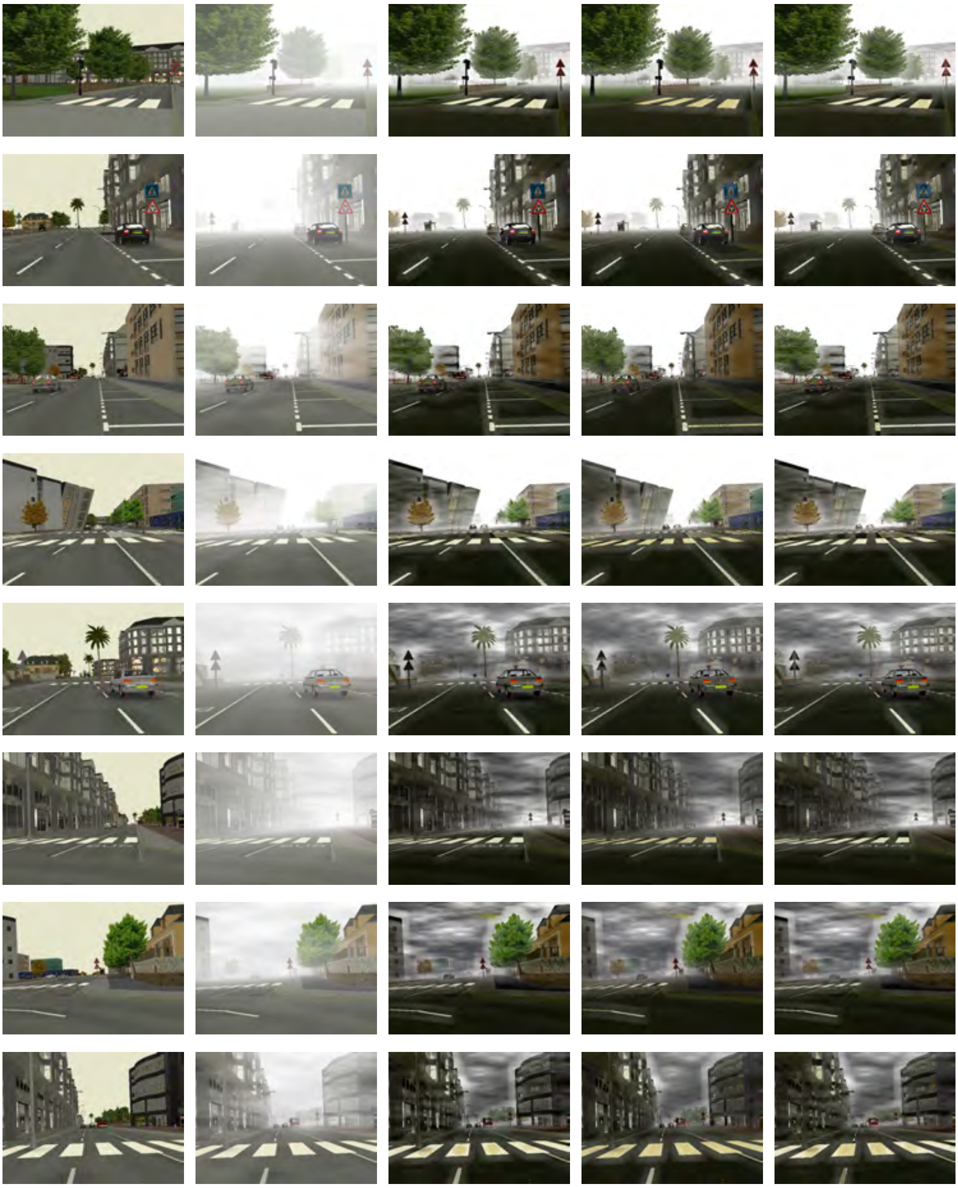


Figure 4.13: Image Enhancement Results on Synthetic Images. Form left to right, the original image without fog, the original image with synthetically added fog and the three images enhanced with the proposed algorithm by using the squared median filter, the median filter along lines and the median filter along columns. The used parameters are $k = 51$ and $p = 0.95\%$

Algorithm	**Hom. Fog**	**Het. Fog**	**Cloudy Hom. Fog**	**Cloudy Het. Fog**	**All Types of Fog**
No	74.8 ± 11.2	53.2 ± 10.8	60.8 ± 10.4	42.1 ± 10.2	57.73 ± 10.6
M	48.3 ± 8.4	40.9 ± 8.5	56.3 ± 11.8	52.3 ± 9.1	49.45 ± 9.4
ML	48.5 ± 8.3	41.2 ± 8.5	56.3 ± 11.4	52.4 ± 8.7	49.60 ± 9.2
MC	47.6 ± 8.3	40.3 ± 8.2	53.6 ± 11.5	51.7 ± 8.8	48.30 ± 9.2

Table 4.1: Average absolute difference between the original synthetic images without fog and the enhanced images for the three compared algorithms, for the four types of synthetic fog (Homogeneous, Heterogeneous, Cloudy Homogeneous and Cloudy Heterogeneous Fog) and for the whole data set.

dian filter along lines (ML) and the median filter along columns (MC), respectively. On the lines from figure 4.13 we have chosen two images for each kind of fog (homogenous fog, heterogeneous β fog, cloudy homogenous fog and cloudy heterogeneous fog). Some object that could not be seen in the original foggy images can be clearly identified in the enhanced images. By carrying out a first visual analysis, it can be seen that the results using the median filter applied on columns have a significantly improved contrast than the ones using the classical median filter and the median filter on lines, especially in the regions where we have details (for example on traffic signs, on cars and on buildings). A first quantitative comparison is presented in table 4.1 and consists in computing the average absolute difference between the original images without fog and the enhanced images using the three above mentioned methods. In order to asses the enhancement improvements we also present the average absolute difference between the initial images and the foggy images and compare this to the obtained results. The results in table 4.1 show that the enhancement results by using the median filter along columns are better than the ones obtained by using the normal median filter and the median filter along lines. In the cloudy homogenous and heterogeneous cases the average enhancement results are larger because we have taken into account the whole restored image that also includes the cloudy sky region, which is not present in the original fog free scenes, and thus cannot be restored.

In the next paragraphs we perform an evaluation on both natural-

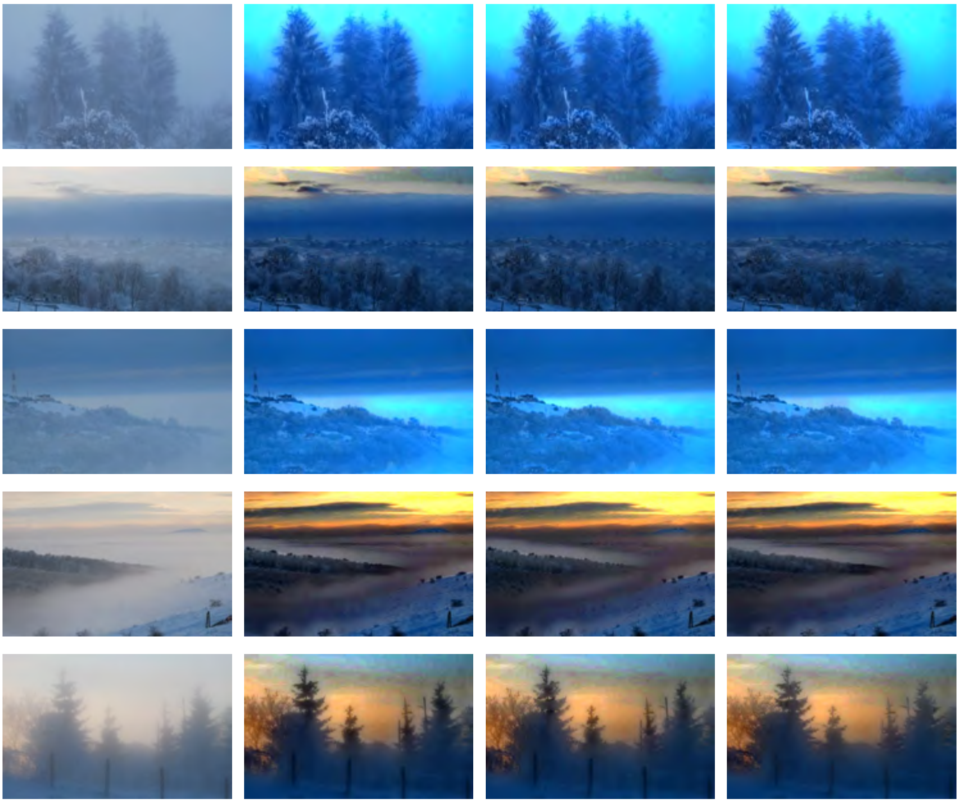


Figure 4.14: Image Enhancement Results on Real World Images. From Left to right, the original image captured in fog conditions, and the three images enhanced with the proposed method by using the squared median filter, the median filter along lines and the median filter along columns.

istic scenes and images obtained from an in-vehicle camera.

- Natural images: Natural images (see figure 4.14) were obtained using a Nikon D80 DSLR camera with 12-24 lens during different fog conditions. We have chosen to enhance the natural images in order to verify the algorithm on foggy scenes that were acquired with a high quality camera and lens system and with adequate exposure time, ISO and gain settings. A visual evaluation of the results in figure 4.14 show that our algorithm is also suitable to be applied in the computational photography domain.

- Traffic images: The images presented in figure 4.15 were obtained in driving scenarios in the neighborhood of Cluj-Napoca by using a CV-A10-CL monochrome camera. Although the original images are in gray scale the results obtained demonstrate that the enhanced images provide a more detailed and consistent in depth scene. Thus, the image enhancement algorithm is also suitable to be applied for driving assistance systems.

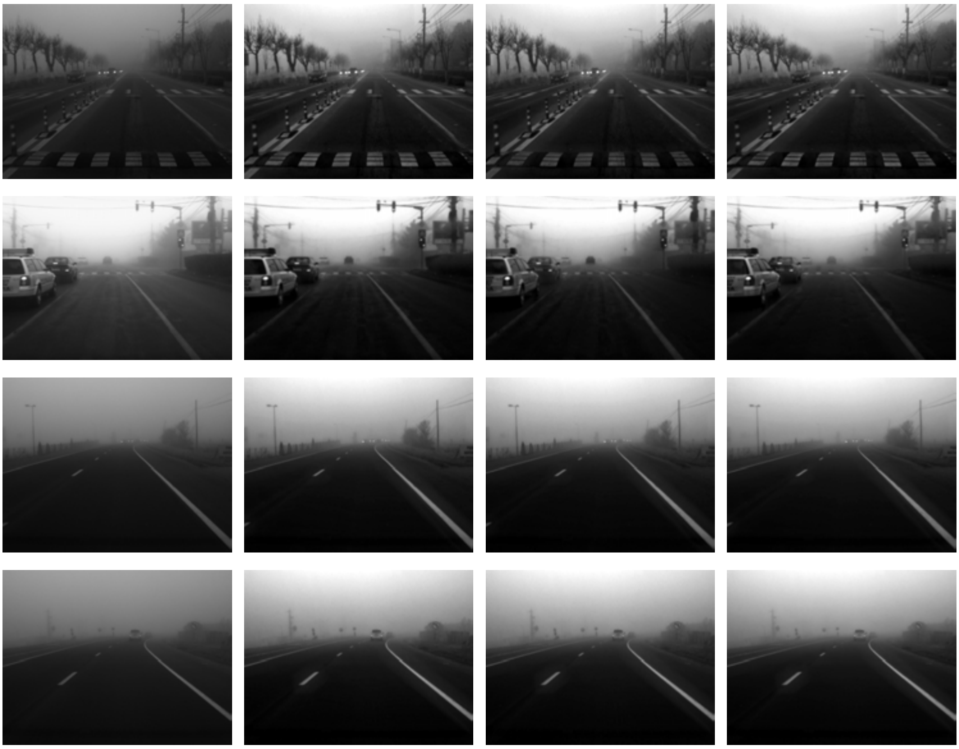


Figure 4.15: Image Enhancement Results on Traffic Images. From Left to right, the original image captured in fog conditions, and the three images enhanced with the proposed method by using the squared median filter, the median filter along lines and the median filter along columns.

Table 4.2 presents the average absolute difference between the acquired foggy images and the enhanced images, in natural scenes and traffic scenarios. The enhanced images that have the greatest absolute difference represent the ones where the captured scene is more clear. It is obvious the this average differences are much larger for the color images (natural scenes) than for the gray scale images (traffic scenarios); and that the image enhancement using the median filter on columns provides the best results.

Restoration Algorithm	**Natural Images**	**Traffic Images**
Median	59.5 ± 17.2	19.03 ± 12.6
Median Lines	59.4 ± 16.6	19.08 ± 12.6
Median Columns	60.5 ± 16.2	19.29 ± 12.6

Table 4.2: Average absolute difference between the enhanced images and the foggy images for our restoration algorithm, by using the three types of median filters, for the natural images and for the real traffic images

In order to assess the quality of restoration, Hautiere et al. propose in [90] to compute the rate of new visible edges in the enhanced image e:

$$e = \frac{n_r - n_o}{n_o} \tag{4.47}$$

Since computing the enhanced images implies subtracting the atmospheric veil from the original foggy image another interesting metric for image enhancement is σ, the percentage of pixels that become completely black or completely white after restoration [90]:

$$\sigma = \frac{n_s}{N} * 100 \tag{4.48}$$

where n_s represents the total number of pixels that become completely black or completely white after image enhancement and N represent the total number of pixels in the image.

Table 4.3 presents the average enhancement results for this two metrics on the images from the FRIDA data-set, the natural scenes and traffic images. It can be observed (Table 4.3.a) that the algorithm using the median filter applied on the columns of the image produces more edge points than the other two algorithms, which means that the enhanced image will present more fine details than the other two enhanced images (The same Canny edge detector [17] was applied to all outputs and the number of edge points was counted for each image). For the synthetic images we have presented the average of these metrics for all four types of fog. The number of edge points in the natural images are much higher than in the other cases because these images are not saturated (they were acquired with correct exposure time, ISO and gain settings) nor underexposed, so they can be reconstructed such that the results resemble a naturalistic scene. In Table 4.3.b it can be seen that our algorithm produces a very small number of underexposed or saturated pixels. Indeed the algorithm using the median along columns produces the smallest amount of bad pixels.

a: Mean values for e

e	M	ML	MC
Synthetic Images	0.42	0.40	0.43
Natural Images	1.21	1.22	1.27
Traffic Images	0.11	0.14	0.22

b: Mean values for σ

σ	M	ML	MC
Synthetic Images	0.0022	0.0019	0.0018
Natural Images	0	0	0
Traffic Images	0.0541	0.0545	0.0505

Table 4.3: Mean values for the e and σ indicators, produced by the three compared methods on the three tested scenarios: synthetic images from the FRIDA data set, natural scenes and traffic images.

Up to this point we have not considered the shape of the fog present in images especially in traffic scenarios. The algorithm assumes that the veil is constant on all image lines. In the next section we are going to model the exponential decay of the fog by a mathematical model that exploits the variation of the fog's density with the distance to the camera.

4.4.3 Exponential Inference of the Atmospheric Veil

The atmospheric veil is not uniform on the whole image, especially in real traffic images where there exists a great variation in the fog density with the distance. Since Koschmieder's law was originally intended to measure the visibility distance of a black object seen in fog having a contrast of only 5%, the contrast restoration based on this law is very good to be applied on scenes that are captured from a certain distance, i.e. where all the objects appear far from the camera. The atmospheric veil computed with this method is over compensated in the bottom part of the image (figure 4.15), thus resulting in very dark restored images (figures 4.13 and 4.15). So this method is not well suited for traffic scenes. Therefore, there is a need for a smooth exponential function that can model the atmospheric veil in traffic scenarios according to the shape of the fog in scene. For this reason we will model a smooth exponential filter on the atmospheric veil such that this exponential function decreases inversely with the distance. In order to apply such an exponential function over the atmospheric veil, we must first investigate some properties of real world images in fog conditions [91]:

- Fog has a very high density in the sky regions. Restoration must be maximum (100%).
- Fog has a very low density, close to zero in the near vicinity of the camera. Restoration must be minimum (0%).
- Fog has an exponential decay between the sky region and until the vicinity of the camera. There must exist an exponential restoration mechanism (percentages between 0% and 100% according to the distance.

Our exponential filter is inspired by the functions which appear in partition of unity in differential geometry field [92], [93]. The reason for modeling this filter with such kind of functions is their monotonicity and smooth shape in relationship with the consistence of fog. The decreasing of these functions on the interval of interest is monotonous, not abrupt as the one presented in [85]. In the previous mentioned paper an exponential decay is introduced in the veil's computation that introduces another constraint for computing the atmospheric veil: the planar assumption (NBPC + PA). Our approach is somehow different in the sense that we treat the whole atmospheric veil as an exponential function. For this reason we construct an exponential filter which will be applied on the whole atmospheric veil, rather than the approach presented in [85]. The final formula for computing the atmospheric veil is:

$$V_{final} = V \cdot G \tag{4.49}$$

where G is an exponential function with values between 0 and 1. In the reminder of this section we will focus on modeling the function G according to the above mentioned properties.

We start from two exponential functions from the partition of unity; we will call them squared and modulus partition of unity functions (figure 4.16).

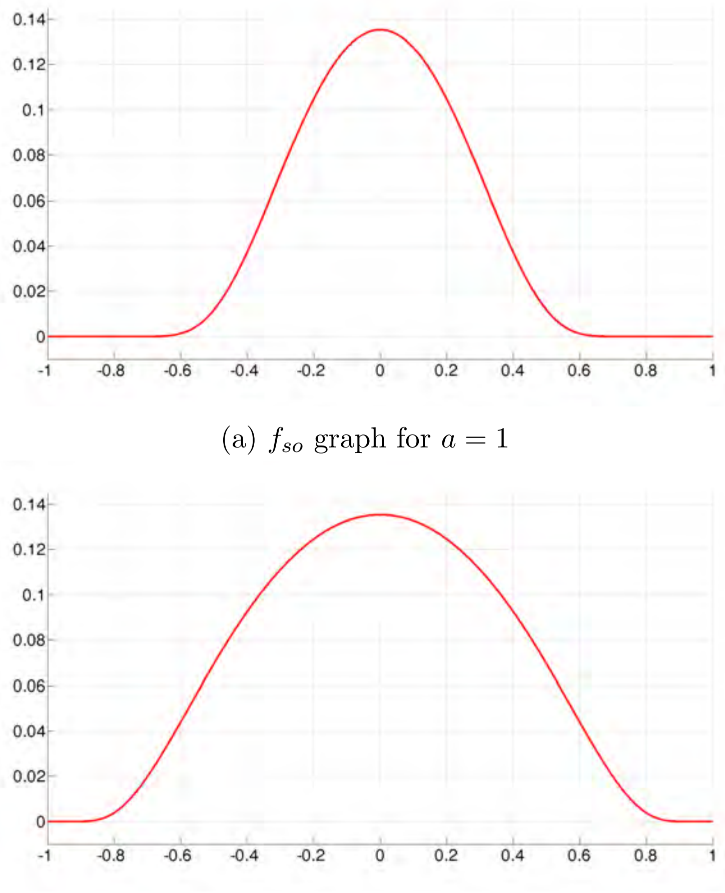
Let $f_{so} : [-a, a] \to [0, 1]$ (squared) and $f_{mo} : [-a, a] \to [0, 1]$ (modulus) having the following form:

$$f_{so}(x) = e^{-\frac{1}{(x-a)^2} - \frac{1}{(x+a)^2}} \tag{4.50}$$

$$f_{mo}(x) = e^{-\frac{1}{|x-a|} - \frac{1}{|x+a|}} \tag{4.51}$$

For using the functions f_{so} and f_{mo} in image processing, we must modify them such that they are defined in the image domain and take values in the $[0, 1]$ interval. For this reason the variable x will denote the image lines.

Hence, we obtain two new exponential functions $f_s : [0, H - 1] \to$



(a) f_{so} graph for $a = 1$

(b) f_{mo} graph for $a = 1$

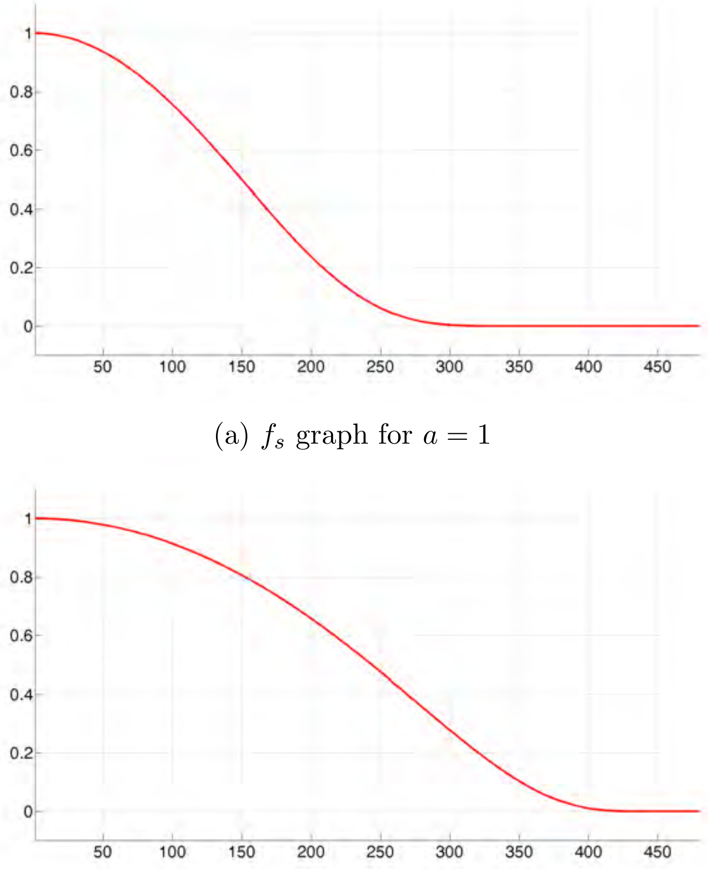
Figure 4.16: The two original partition of unity functions

$[0, 1]$ and $f_m : [0, H-1] \to [0, 1]$, (figure 4.17) with the following form:

$$f_s(x) = \frac{e^{-\frac{1}{(\frac{ax}{H}-a)^2}-\frac{1}{(\frac{ax}{H}+a)^2}}}{f_{so}(0)} \tag{4.52}$$

$$f_m(x) = \frac{e^{-\frac{1}{|\frac{ax}{H}-a|}-\frac{1}{|\frac{ax}{H}+a|}}}{f_{mo}(0)} \tag{4.53}$$

where H is the height of the image (the number of lines in the image). The denominator in the above equations is used in order to scale the



(a) f_s graph for $a = 1$

(b) f_m graph for $a = 1$

Figure 4.17: The two modified partition of unity functions

values for our exponential functions f_s and f_m in the $[0, 1]$ interval.

Taking into account the previous presented properties of foggy images, means that our exponential function must have the maximum value (1) in the sky regions (above the horizon line) and it must have a minimum value in the near vicinity of the camera, up to a certain distance. Thus, a translation of our modified exponential functions (along the x axis from figure 4.17) has to be applied. A linear isomorphism

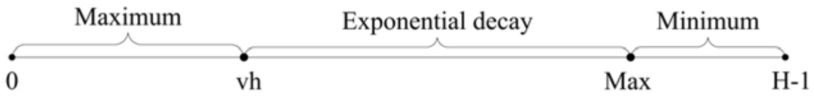


Figure 4.18: Segment for translation of the final exponential functions

$A : [vh, Max] \to [0, H-1]$ is used for this purpose:

$$A(x) = ax + b \tag{4.54}$$

having the following properties:

$$\begin{cases} A(vh) & = & 0 \\ A(Max) & = & H-1 \end{cases} \tag{4.55}$$

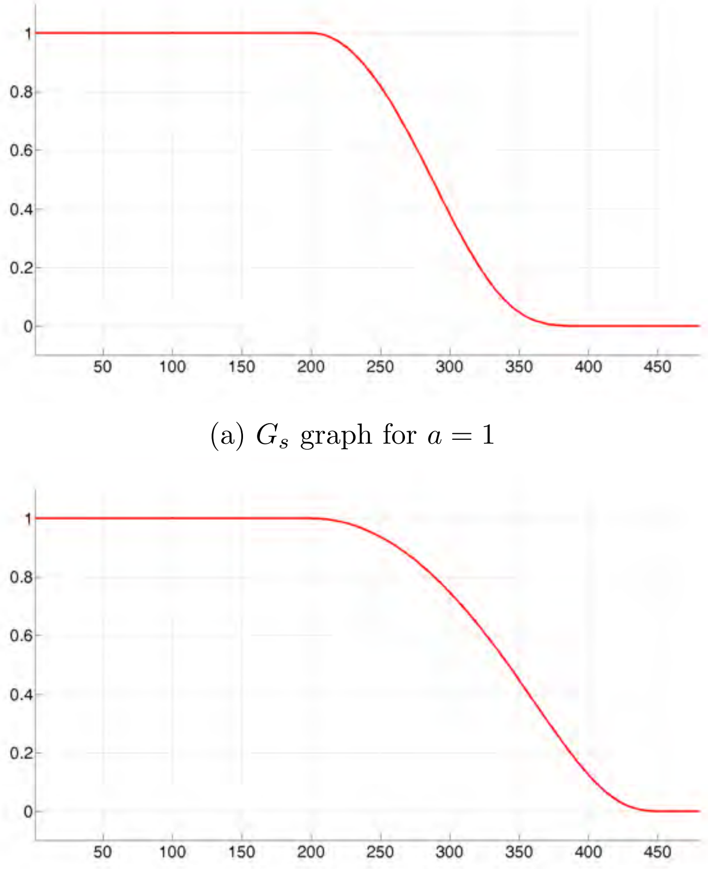
such that by composing the modified partition of unity functions with the function A we obtain our desired exponential functions.

Figure 4.18 presents a line segment for translating our exponential functions. The values on this line segment represent image lines. H is the height of the image, vh and Max can be arbitrarily chosen such that our functions will be defined on three intervals($[0, vh]$, (vh, Max), $[Max, H-1]$). For example we could assign one third of the image lines for each interval. A more robust solution would be to take into account the position of the horizon line (vh) and the thickness of the fog in the scene.

By solving the system of equations presented in equation (4.55) the final exponential function becomes:

$$G_s(x) = \begin{cases} 1 & \text{if } x \leq vh \\ f_s(c(x - vh)) & \text{if } x \in (vh, Max) \\ 0 & \text{otherwise} \end{cases}$$

where $c = \frac{H-1}{Max-vh}$. Depending on the value chosen for the Max parameter a new translation will be obtained. Typical values are $\frac{1}{4}$, $\frac{1}{2}$, $\frac{3}{4}$ of the $[vh, H-1]$ interval or even the maximum value ($Max = H-1$). Figure 4.19 presents the final allure of our translated exponential functions. A similar function G_m is obtained by using f_m instead of f_s.



(a) G_s graph for $a = 1$

(b) G_m graph for $a = 1$

Figure 4.19: The two translated exponential functions. Used values are $H = 480$, $vh = 200$, $Max = H - 1$.

In figure 4.20 we present the atmospheric veils obtained by using our approach. The first column represents the original image from the FRIDA data set [85]. The second column represents the image with homogenous fog added, while the next columns present different atmospheric veils for the following algorithms: modified no black pixel constraint (NBPC) with no exponential function used, modified no black pixel constraint with planar assumption (NBPC+PA) [85] and our algorithm with the following exponential functions: f_s and f_m exponential functions and G_s and G_m translated exponential functions.

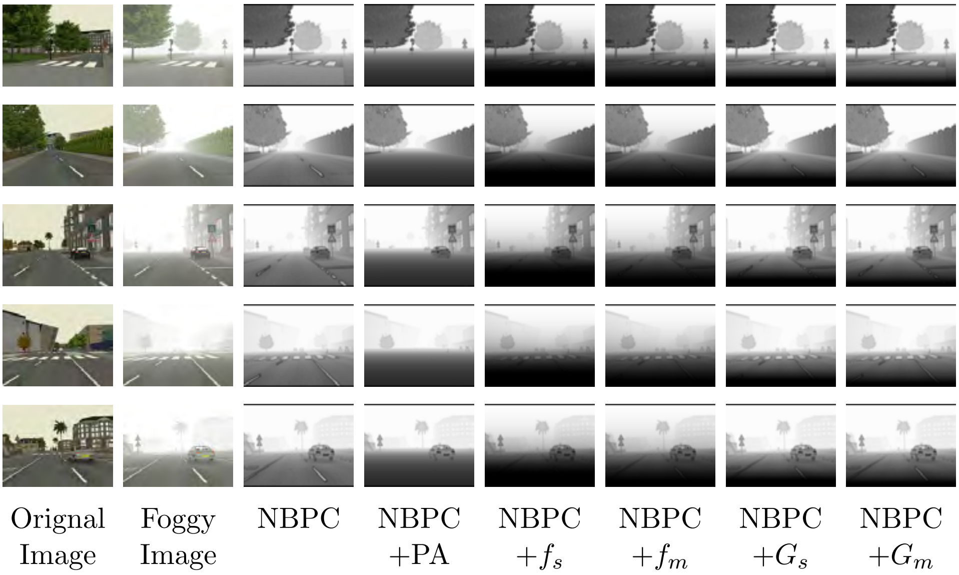


Figure 4.20: Obtained atmospheric veil. From left to right, the original image without fog, the original image with synthetically added fog and the atmospheric veil obtained using the filtering along columns for the following algorithms: NBPC, NBPC+PA, NBPC + f_s, NBPC + f_m, NBPC + G_s and NBPC + G_m (the used parameters are $k = 15$, $p = 95\%$, $a = 3$).

The contrast restoration results are presented in the next section.

4.4.4 Contrast Restoration Results

In this section we perform a quantitative and qualitative evaluation of our image defogging algorithm on both synthetic and real camera images. Furthermore, a comparison of our algorithm with the method presented in [85] is also performed.

The synthetic images used in our evaluation come from the FRIDA data set [85]. This data set contains synthetic images that try to simulate real traffic scene. Four different types of fog were added to these images: homogenous fog, heterogeneous fog, cloudy homogeneous fog and cloudy heterogeneous fog. We have presented the highlights of this data set in section 4.3 of this chapter.

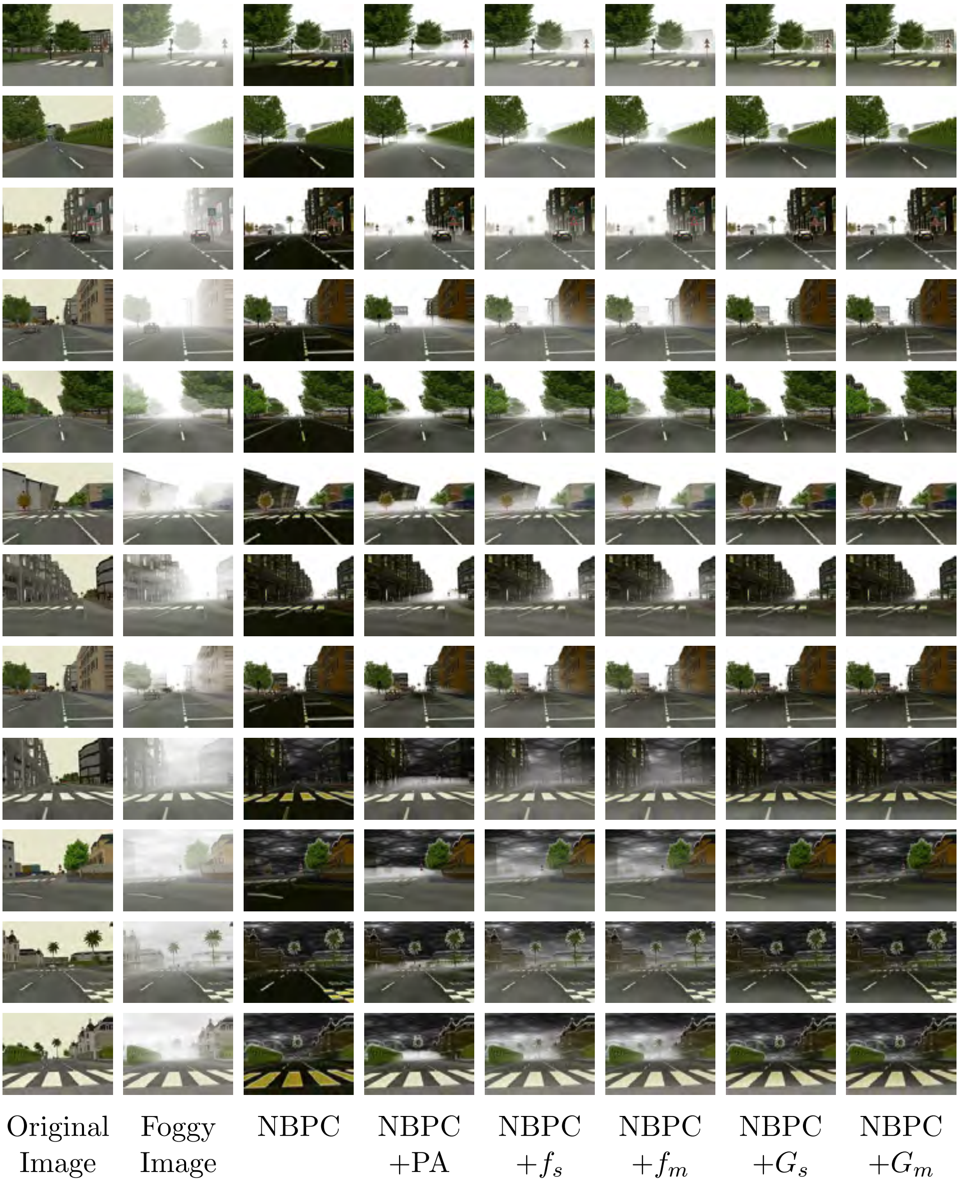


Figure 4.21: Contrast restoration results on synthetic images. Form left to right, the original image without fog, the original image with synthetically added fog and the restoration results with the following algorithms: NBPC, NBPC+PA [85], NBPC + f_s, NBPC + f_m, NBPC + G_s and NBPC + G_m.The used parameters are k = 15, p = 0.99% and a = 3.5

Figure 4.21 presents twelve images from this data set that were enhanced by using our image enhancement algorithm, using six methods for computing the atmospheric veil. The columns represent the original image without fog, the original image with synthetically added fog and the six enhanced images by using the following algorithms: modified no black pixel constraint (NBPC), no black pixel constraint with planar assumption (NBPC+PA) [85], no black pixel constraint with the squared exponential function (NBPC+f_s), no black pixel constraint with the modulus exponential function (NBPC+f_m), no black pixel constraint with the translated squared exponential function (NBPC+G_s) and no black pixel constraint with the translated modulus exponential function (NBPC+G_m). All the above mentioned algorithms use the filtering method applied on the columns of the image. On the lines from figure 4.21 we have chosen four images for the homogeneous and heterogeneous fog scenarios and two images for cloudy homogeneous and cloudy heterogeneous fog scenarios. By carrying out a first visual analysis it can be seen that all the above methods are suitable for image enhancement. The first method, NBPC (column 3) yields a very dark image when comparing to the original image without fog. The idea behind contrast restoration is to remove the fog from the foggy image such that more features are visible in the enhanced image. So, objects that could not be seen in the original foggy images can be clearly identified in the enhanced images. The second method, (NBPC+PA) [85], gives good results in the vicinity of the camera, but it cannot restore the contrast at medium to great distances. Some objects are not visible in the enhanced image: traffic signs, cars, etc. Another problem of this method is that the restoration is not smooth, it can be seen that a portion of the image is not correctly restored. The methods using the squared and modulus exponential functions (NBPC+f_s and NBPC+f_m) are smooth and can restore the image very good up to a certain distance. The last two methods (NBPC+G_s and NBPC+G_m) give the best results. The restoration is smooth on all the image and the enhancement is resolved even at great distances from the camera.

Algorithm	**Hom. Fog**	**Het. Fog**	**Cloudy Hom. Fog**	**Cloudy Het. Fog**	**All Types of Fog**
No	71.83 ± 10.23	50.84 ± 10.36	58.02 ± 10.53	39.98 ± 9.84	55.16 ± 10.24
NBPC	49.54 ± 6.35	47.55 ± 6.94	84.32 ± 21.32	82.43 ± 21.26	65.96 ± 13.96
NBPC+PA	35.99 ± 8.52	30.74 ± 7.56	65.35 ± 23.25	63.31 ± 20.48	48.84 ± 14.95
NBPC+f_s	40.16 ± 8.18	27.57 ± 5.16	57.26 ± 14.28	49.58 ± 16.12	43.64 ± 10.93
NBPC+f_m	37.13 ± 7.35	26.94 ± 4.68	53.98 ± 15.06	48.88 ± 15.19	41.73 ± 10.57
NBPC+G_s	32.40 ± 5.79	30.38 ± 6.32	66.02 ± 21.63	64.75 ± 20.52	48.38 ± 13.56
NBPC+G_m	32.31 ± 5.94	31.06 ± 6.02	65.93 ± 21.84	65.48 ± 20.45	48.69 ± 13.56

Table 4.4: Average absolute difference between the original synthetic images without fog and the enhanced images for the four types of synthetic fog (Homogeneous, Hetero- geneous, Cloudy Homogeneous and Cloudy Heterogeneous Fog) and for the whole data set

Algorithm	**Hom. Fog**		**Het. Fog**		**Cloudy Hom. Fog**		**Cloudy Het. Fog**		**All Types of Fog**	
	e	σ	e	σ	e	σ	e	σ	e	σ
NBPC	37.23	0.0005	19.45	0.0003	77.51	0.0019	48.63	0.0014	45.71	0.0010
NBPC+PA	27.68	0.0016	15.47	0.0003	67.10	0.0025	44.69	0.0014	38.73	0.0014
NBPC+f_s	24.89	0.0000	12.64	0.0000	61.97	0.0026	40.39	0.0011	34.97	0.0009
NBPC+f_m	27.24	0.0002	13.68	0.0001	64.03	0.0023	40.81	0.0011	36.44	0.0009
NBPC+G_s	34.16	0.0013	17.34	0.0002	74.34	0.0024	46.55	0.0013	43.09	0.0013
NBPC+G_m	34.69	0.0012	17.58	0.0002	74.71	0.0024	46.89	0.0013	43.46	0.0012

Table 4.5: Mean value produced by the e and σ metrics for the synthetic foggy images and the enhanced images, for the four types of fog (Homogeneous, Heterogeneous, Cloudy Homogeneous and Cloudy Heterogeneous Fog) and for the whole data set

One quantitative comparison of these algorithms is presented in table 4.4 and consists in computing the average absolute difference between the original images without fog and the enhanced images [84]. In order to asses the enhancement improvements we also present the average absolute difference between the initial images and the foggy images and compare this to the obtained results. By carrying out an analysis of table 4.4 it would result that the algorithm using the modulus exponential function gives the best results, but this is not the case, since a part of the image is still foggy. In our opinion some of the algorithms preserve a good reconstruction at small distances (road surface) (NBPC+PA [85], NBPC+f_s and NBPC+f_m) but are not able to enhance the portions of the image that are located at greater distances. This problem is resolved by the last two algorithms (NBPC+G_s and NBPC+G_m). They perform better at medium and great distances, being able to enhance more features such as traffic signs, cars, details on buildings, etc, but the image is a little darker than the original image without fog. This is the reason why we do not obtain the best result by using this metric (average absolute difference) for our translated exponential functions.

In the previous section we have presented two other metrics for image enhancement: rate of new visible edges in the enhanced images e and the percentage of pixels that become completely black or completely white after restoration σ. These two metrics are more relevant for assessing the quality of restoration. It can be seen in table 4.5 that the image enhancement using the translated exponential functions G_s and G_m are able to produce the greatest number of new visible edges, maintaining a very small percent (0.0012%) of bad pixels (underexposed or saturated pixels), thus providing a very good enhancement rate.

In figure 4.22 we present the results of image enhancement on real traffic images obtained in different driving scenarios in fog conditions in the neighborhood of Cluj-Napoca. Although the original images are in gray scale the results obtained demonstrate that the enhanced images provide a more detailed and consistent view of the analyzed scene. Table 4.6 presents the average enhancement results for the e and σ metrics on the traffic images. It can be observed that the algorithms using the translated exponential functions produce the greatest number of new edges in the enhanced images, while providing the smallest number of underexposed or saturated pixels.

We have acquired stereo images using two CV-A10-CL monochrome cameras and performed a dense stereo reconstruction on the foggy images using the SORT-SGM algorithm; a GPU based implementation presented in [94]. After performing the image enhancement of the foggy images using all the previous presented algorithms we have repeated the stereo reconstruction process for each method. In our work we have focused on four different test cases having the following parameters:

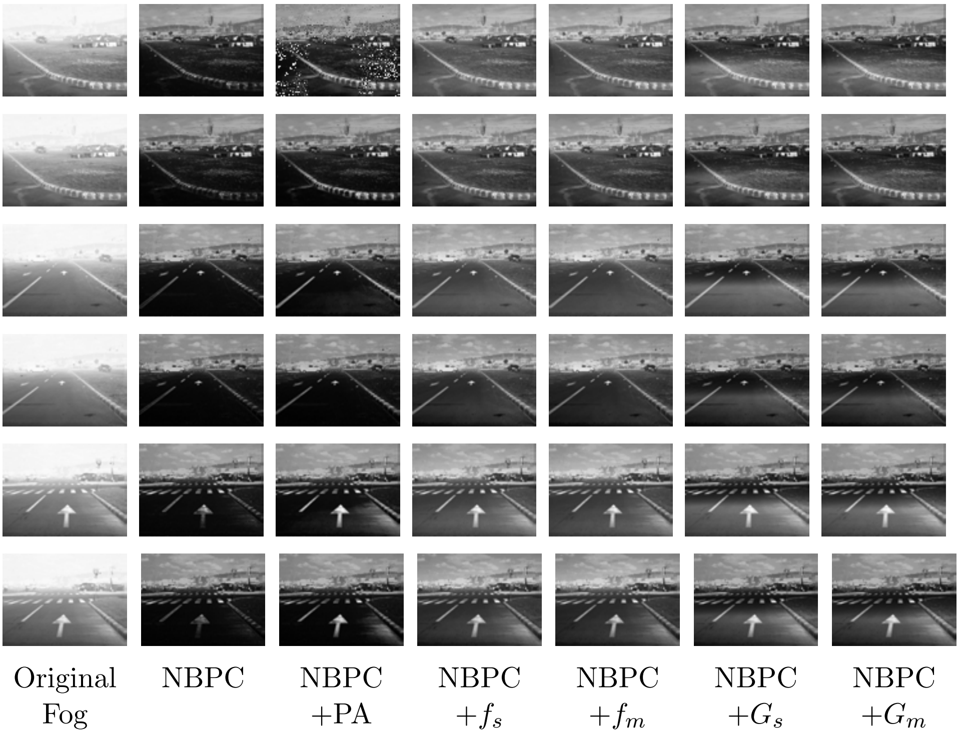


Figure 4.22: Image Enhancement Results on Traffic Images. From Left to right, the original image captured in fog conditions, and the enhanced images with the following algorithms: NBPC, NBPC+PA [85], NBPC + f_s, NBPC + f_m, NBPC + G_s and NBPC + G_m. All the presented algorithms use the median filtering along the columns of the image (the used parameters are $k = 15$, $p = 0.95\%$ and $a = 3.5$)

- 1: $k = 05$, $p = 0.95\%$, $a = 3.5$ and $Max = \frac{H+vh}{2}$
- 2: $k = 05$, $p = 0.95\%$, $a = 3.5$ and $Max = H - 1$
- 3: $k = 15$, $p = 0.95\%$, $a = 3.5$ and $Max = \frac{H+vh}{2}$
- 4: $k = 15$, $p = 0.95\%$, $a = 3.5$ and $Max = H - 1$

The differences between these test cases appear only for the enhancement algorithms using the translated exponential functions. We can observe a boost in the number of 3D points in the cases where the Max parameter is considered to be in the middle of the $[vh, H - 1]$

interval. The odd lines form figure 4.22 present the left images, while the even lines present the right ones. The percentage of new $3D$ points obtained through image enhancement, with each method is presented in table 4.7 and the average number of $3D$ points obtained through stereo reconstruction for test case 3 is presented in figure 4.23. We can observe that the methods using the squared and modulus exponential functions are able to provide more 3D points than all the other methods.

Algorithm	Left		Right	
	e	σ	e	σ
NBPC	63.68	0.0006	66.97	0.0066
NBPC+PA	63.67	0.0006	66.86	0.0066
NBPC+f_s	53.61	0.0035	59.10	0.0064
NBPC+f_m	54.69	0.0016	59.44	0.0064
NBPC+G_s	65.18	0.0006	65.29	0.0064
NBPC+G_m	65.33	0.0006	65.07	0.0064

Table 4.6: Mean values for e and σ metrics, for stereo traffic images.

Algorithm	Case 1	Case 2	Case 3	Case 4
NBPC	-4.06	-4.06	1.87	1.87
NBPC+PA	-8.42	-8.42	0.44	0.44
NBPC+f_s	4.89	4.89	5.57	5.57
NBPC+f_m	4.38	4.38	5.80	5.79
NBPC+G_s	3.33	2.20	5.05	4.48
NBPC+G_m	3.29	1.89	5.04	4.34

Table 4.7: Average Percent of new 3D points in the enhanced traffic images for the four test cases

The used stereo reconstruction algorithm [94] is able to perform $3D$ reconstruction up to 40 meters with an average error of 3%. Since we cannot estimate the reconstruction errors in fog conditions we propose to evaluate the accuracy of our image enhancement algorithms not only by performing 3D reconstruction on stereo images, but also by performing object detection and classification in fog conditions. In

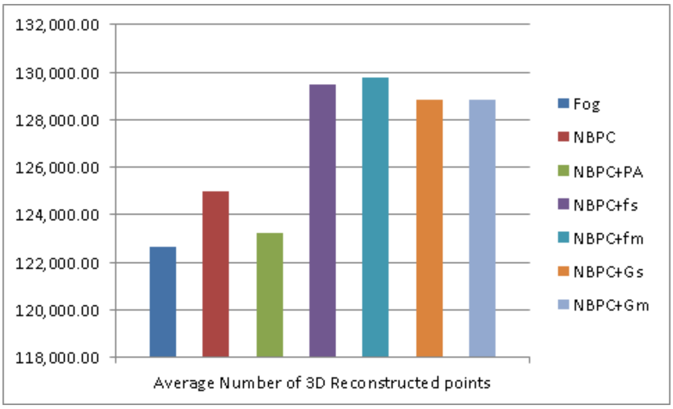


Figure 4.23: Average Number of $3D$ Reconstructed Points

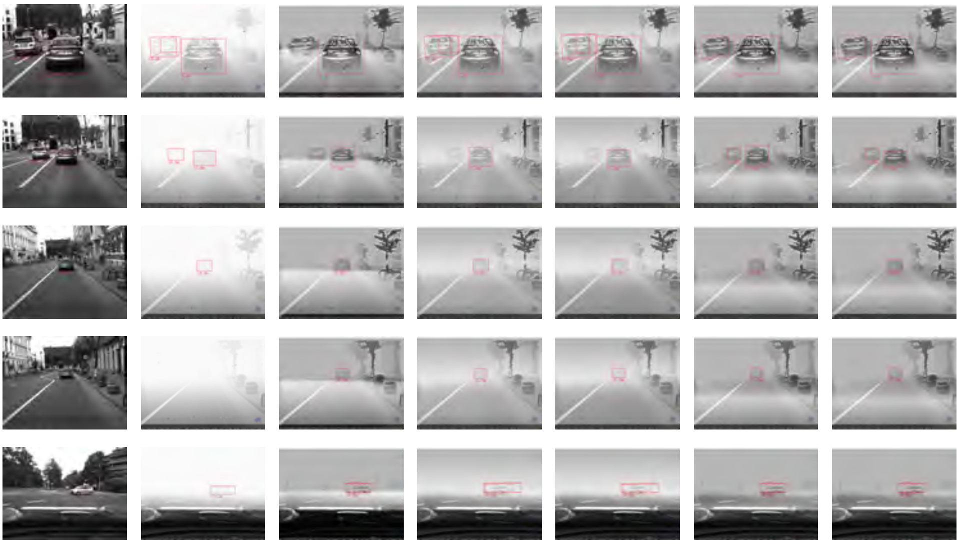


Figure 4.24: Object classification results on enhanced images. From left to right, the original image, the image with added homogeneous fog, and the enhanced images with the following algorithms: NBPC+PA [85], NBPC + f_s, NBPC + f_m, NBPC + G_s and NBPC + G_m (the used parameters are $k = 15$, $p = 0.95\%$ and $a = 3.5$).

order to achieve this, we have added fog to a stereo image sequence (visibility distance below 15 meters) and performed stereo reconstruction on the original fog free images, on the images with added fog and on the enhanced images. Next we perform object detection and classification in all these scenarios. The target object used in our tests is a vehicle (displayed with a red box). The results are presented in figure 4.24. At small distances (< 10) m all the algorithms are able to correctly classify the car in front of the ego vehicle and to accurately estimate it's distance, but the algorithm using NBPC+PA is not able to detect the second vehicle. As the distance increases the objects tend to disappear in the fog until they are no longer detected and classified. The detection and classification error increases with the distance; for example in the second image the detected vehicle is situated at 11.2 m and is perceived as being 0.4 m further in the foggy image than in the original one, in the third case 1.5 m further, while in the fourth the object is no longer detected and classified. Table 4.8 presents the obtained distances to the target vehicle after performing object detection and classification in the above presented scenarios. By analyzing the object detection and classification results presented in figure 4.24 we can state that the image enhancement algorithms using the translated exponential functions are the ones that yield the best results, very close to the fog free scenarios; the average detection errors are below 5.5% in the cases where the target vehicle are located beyond the visibility distance.

Algorithm	**Case 1**	**Case 2**	**Case 3**	**Case 4**	**Case 5**
Original	4.9	11.2	17.0	20.0	12.9
Fog	5.0	11.6	18.5	no	13.2
NBPC+PA	5.0	11.4	17.4	22.7	12.9
NBPC+f_s	5.0	11.4	17.9	21.7	13.2
NBPC+f_m	5.0	11.4	17.7	22.5	12.9
NBPC+G_s	5.0	11.4	17.3	21.7	12.9
NBPC+G_m	5.0	11.4	17.3	21.7	12.9

Table 4.8: Distance to Target Vehicle obtained after object detection and classification

Our exponential functions use 3 parameters: the position of the horizon line in the image v_h, max represents the line from which the translated exponential function has the value 0 and a is the parameter that models the shape of the exponential function. The value of v_h can be estimated from the camera parameters or it can be computed with the algorithm described in [6], max $= H - 1$ since our exponential function decreases asymptotically towards zero, and the a parameter is chosen accordingly to the thickness of the fog in the scene. By applying a fog detection on the input images we can estimate the fog category based on the visibility distance obtained from the images [6]. If the visibility distance is below $50m$, very dense fog scenario, we have used $a = 4.5$ in the evaluation of the distances to the target vehicle based on stereo reconstruction. When the visibility is between 50 and $100m$ we have chosen $a = 3.5$. Thus, a look up table based approach can be employed for choosing the value of this parameter. If there is no fog in the scene by selecting $a = 0$ the exponential function will have the value 0, and thus the estimated atmospheric veil will be null.

Another important factor for image enhancement is the processing time. Our method is robust and scalable, and the complexity of the presented algorithm varies linearly with the dimension of the input image. The method was implemented and tested in C++ on an i7-based PC. Table 4.9 presents the average processing time of our algorithm on the three test sets: synthetic images, natural images and traffic scenes. We can observe that the processing time is below 25 milliseconds in the real traffic scenarios, so we are able to process at least 30 frames per second, which makes our algorithm suitable for real time image processing.

4.4.5 Optimizing Contrast Restoration on Embedded Hardware

A big issue for vision based advanced driving assistance functions or for fully automated vehicles is the ability to run image processing based algorithms on embedded low power devices. This is due to the

Image Resolution	**Image Type**	**Image Color**	**Processing Time (ms)**
640 x 480	Synthetic	RGB	32.4
604 x 404	Natural	RGB	17.9
512 x 383	Traffic	GRAY	24.2

Table 4.9: The average processing time of our algorithm on the three tested scenarios: synthetic images from the frida data set, natural images and traffic images.

high demand of computational resources needed by such applications. For this purpose we have chosen to deploy our contrast restoration solution on an Xilinx Zynq FPGA embedded platform [95]. In this section we present the methodology for reaching this goal and the increase in performance that can be achieved with an FPGA device when comparing to the solution implemented on an ordinary off the shelf processor. The ZYNQ Programmable System on Chip (SoC) is a Xilinx FPGA device, that not only provides a powerful FPGA (Artix-7/Kintex-7) but also a 1.2 GHz ARM dual core processor. In addition, such a SoC combines the flexibility of writing code in a high level programming language (C/C++) with the massive parallelism and computational power obtained inside an FPGA device. This SoC provides multiple advantages:

- performance – the performance gain that can be achieved by exploiting the full capabilities of the SoC is very high. Our embedded solution is more than 10 times faster than the PC based one.

- cost – the embedded solution proves to be not only faster than other solutions, but it is also more cost effective.

- form factor – the solution is run on a SoC that can easily be integrated inside a vehicle's embedded camera.

- power consumption – the power consumption of such a Xilinx SoC is in the range of a few Watts ($<$ 10 W).

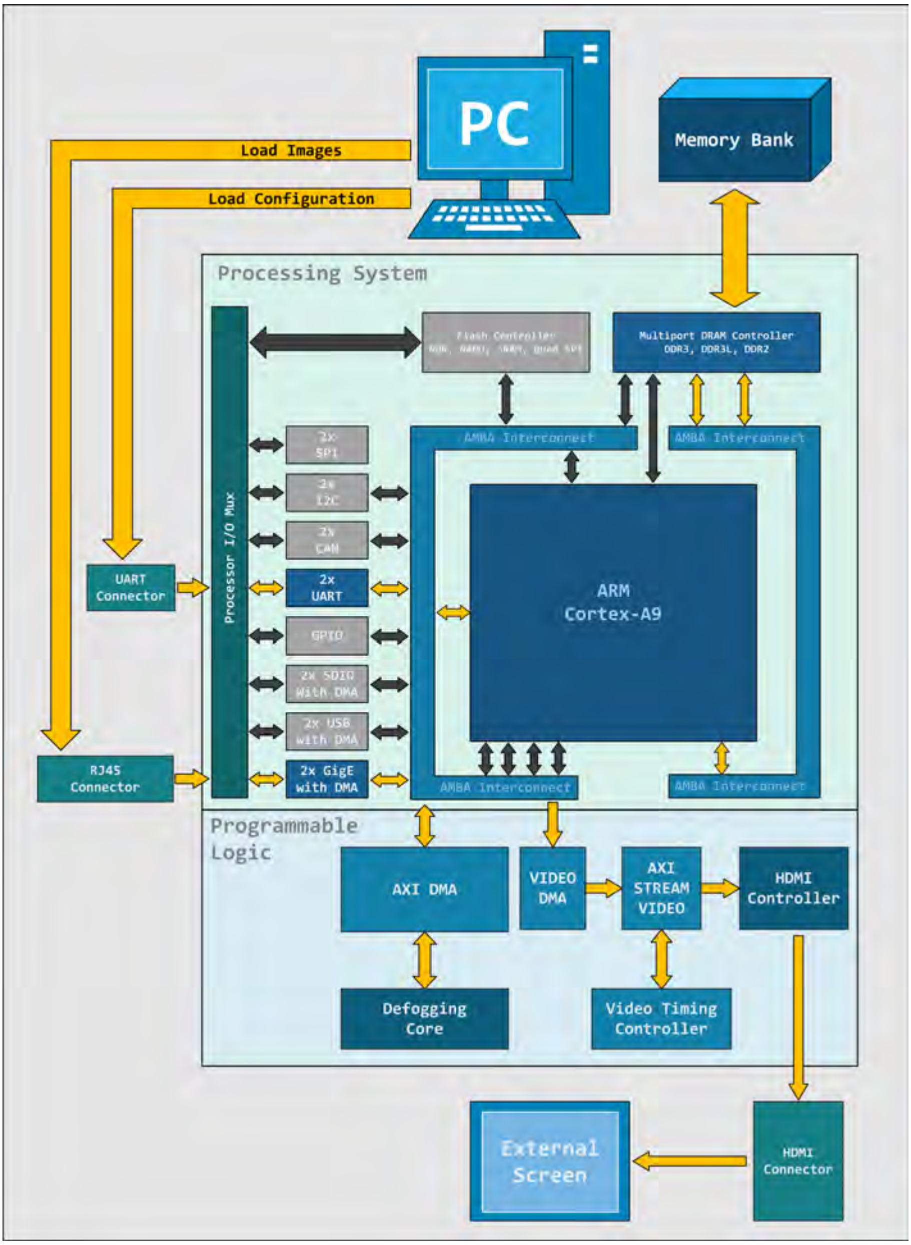


Figure 4.25: Global processing diagram for embedded contrast restoration on Xilinx ZYNQ FPGA [95].

All these characteristics make the Xilinx ZYNQ architecture ideal for real time embedded image processing in automotive applications.

The prototype / test board used in these experiments (Digilent ZYBO board) is equipped with a ZYNQ Z-7010 SoC and has a 512 MB DDR Memory. The hardware version of our contrast restoration design is spread in both the Programmable Logic (FPGA) and the Processing System (ARM CPU). The Xilinx ZYNQ architecture provides multiple ports that facilitate the communication between these two main components - AXI Protocol [96]. This protocol proposed by ARM is part of the AMBA specification and incorporates interfaces like AXI Lite, AXI4 and AXI Stream each suited for specific use cases, facilitating the communication between the FPGA and the ARM processor. A high level view of this SoC and the the cores implemented in the programmable logic are depicted in figure 4.25.

The image enhancement algorithm presented in the previous sections was the basis for our FPGA based implementation. Given the hybrid nature of the Xilinx ZYNQ embedded platform we have experimented with different hardware implementations of this algorithm. Taking into account the layout of the development board used, the images were loaded inside the ZYNQ platform through an Ethernet interface. So the connection between the PC and the FPGA SoC is made via the Ethernet interface, which in the SoC is interfaced directly in the ARM processor. This approach made the image acquisition a very simple process; it is done by the ARM CPU. The images coming from the PC are stored in the DDR memory available on the development board. For displaying the results, a monitor is connected directly to the development board through an HDMI interface. Since the HDMI interface pins are connected to the programmable logic in the FPGA, we have designed a custom HDMI controller for displaying the enhanced images. In order to put as little pressure as possible on the CPU we used one of the DMA engines available on the board, the Video Direct Memory Access IP (VDMA) provided by Xilinx. After the foggy images are processed and stored in the external DDR memory we have serialized the defogged images, using AXI Stream to the programmable logic. We have used a VTC (Video Timing Controller) IP for synchronizing the stream and then converted it the RGB24 format. Using a custom VHDL based VIDHDMI core, the pixel data is encoded as well as serialized according to the HDMI specification and finally connected to the output ports.

Full Software Approach
The first approach when porting an algorithm to embedded hardware is first to implement it only in software. So the entire defogging algorithm was ported in C and run on the ARM CPU, using the FPGA just for the displaying the results through the HDMI interface. Although the resulting images are good, these first results were poor from the performance point of view. The processing of one frame took around 1 second, which was nowhere near the desired throughput. However, after further software optimizations we managed to get that number down to around 180 ms per frame. This proves to be the most straightforward and accessible method, but unfortunately it poses some problems when it comes to its effectiveness in real-time systems even after extensive software optimizations. The ARM CPU on our ZYNQ works at a frequency of only 650 MHz and it should come as no surprise that the performance is nowhere near a conventional high performance PC with clock frequencies in the 3–4 GHz range. In addition, the throughput offered by this approach is not usable in a real world situations, yielding only about 5 frames per second. Figure 4.26 present the hardware architecture for the full software solution.

Hybrid Approach
The second step in our optimization strategy was to take advantage of the parallelization and pipelining abilities of the FPGA device and thus to harness its capabilities. The performance gains that can be obtained in the programmable logic, especially when it comes to image processing are considerable. The downside of this is that code written for the hardware is harder to develop and more difficult to maintain, but it makes up for that in terms of the customization and control one has on the entire process. As such, we decided to first port our computationally intensive tasks to the programmable logic. More specifically we targeted the median and the standard deviation filters used in the defogging process, equation (4.46). These filters are implemented in

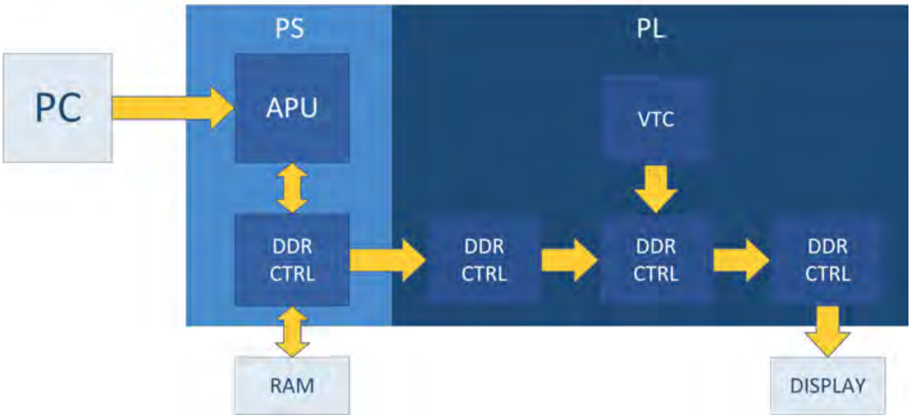


Figure 4.26: Full software solution. The image defogging algorithm is implemented completely on the ARM processor from the ZYNQ embedded platform [95]

the FPGA using Vivado HLS with modified hardware orientated C code. While Vivado HLS provides a high level language in which to develop our Image Processing Core, the standard sequential C mindset proves ineffective. In this environment we must always keep in mind the consequences of our code in terms of used area, parallelization and pipelining potential. Using these three principles we obtained a faster and efficient hardware design of the image defogging algorithm. The transfer of the foggy image from the DDR memory through the Filter Core (median and std deviation) is done with minimum CPU utilization, using the DMA Engine which operates on the High Performance AXI Ports residing between the two integrated components of the ZYNQ SoC. The image is serialized using an AXI Stream interface through our pipelined Filter Core and back into main memory where the subsequent contrast restoration steps are performed. Only by moving these performance critical parts into the FPGA logic yielded more than double the performance when comparing to the pure CPU one, giving us around 70 ms for the defogging of one frame. However, what is truly notable here is the fact that the operations done in the programmable logic part of the SoC only took around 4 ms, including the transfer. This incredible performance was achieved due to the proper pipelining of the algorithm.

Full Hardware Approach

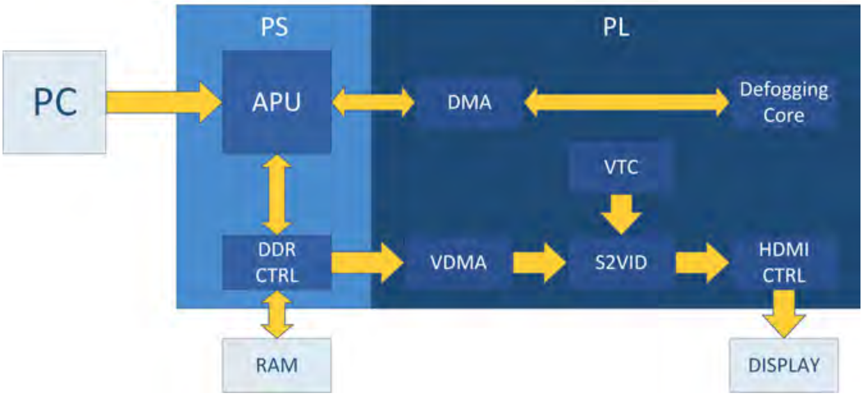


Figure 4.27: Full Hardware solution. The image defogging algorithm is implemented entirely in the FPGA logic of the ZYNQ embedded platform [95]

Taking into account this amazing potential of our pipeline solution, we have then decided to move the entire algorithm in the FPGA logic. Working with a modest ZYNQ Z-7010 SoC we are very limited in terms of logic area. Even so, we managed to optimize our code and make the algorithm fit with room to spare, keeping the cost advantage. Figure 4.27 presents the top-level architecture of the full hardware solution. Some decisions had to be made in order to save up some area for the complete hardware solution:

- We have used a look-up-table for storing the values of the translated exponential function G_s and G_m, equations (4.52 and 4.53).

- In order to free up resources and to be able to fit the whole design in the FPGA programmable logic, we have used single precision floating point calculations instead of double precision floating point (Vivado HLS uses different Cores for float and double).

- We took full advantage of the DSPs cores available in the FPGA device.

Results on the embedded FPGA platform

The used embedded FPGA platform is the ZYNQ Z-7010 SoC from Xilinx. This SoC has the following hardware configuration: only 28.000 Logic Cells, 2.1 Mb Block Ram and 80 DSP slices. It is the entry level configuration for the ZYNQ platforms. As in the previous sections we have used the FRIDA (Foggy Road Image Database) data set [86] for assessing the contrast restoration results on this embedded platform.

As in the previous sections, we present the contrast restoration results with our defogging algorithm on the ZYNQ embedded device in figure 4.28. The first column presents synthetic images without fog from the FRIDA data set [86]. The second column presents the same images with added homogenous fog, while next two columns present our restoration results in the full software, full hardware implementations when using the translated exponential function G_s. The parameters used for the image restoration process are: k= 3 (size of the median filter), p = 99% (percent from equation (4.46)), a = 3.5, $v_h = 260$, $Max = H - 1$ ($H = 480$ — image height). By carrying out a visual analysis of the results in Figure 4.28, it can be seen that all the presented implementations are good candidates from the results quality point of view. The images in the full software approach and the full hardware implementation are very similar. The average absolute difference between these images is less than 2% due to the optimizations required in order to be able to fit the full hardware design on our test board. The restoration results are significant, superior to other state of the art algorithms, as it was presented in [91] and [97]. The visibility distance in the scene is drastically increased, such that a driver can actually see the obstacles that are in front of the vehicle even in very dense fog conditions.

When programming for such an embedded FPGA device, an important assessment contains the device utilization for the obtained hardware architectures. In the full software configuration (the whole defogging algorithm runs on the ARM CPU) only the HDMI display component is implemented in the programmable logic. When comparing the hybrid and full hardware solutions only the number of DSP

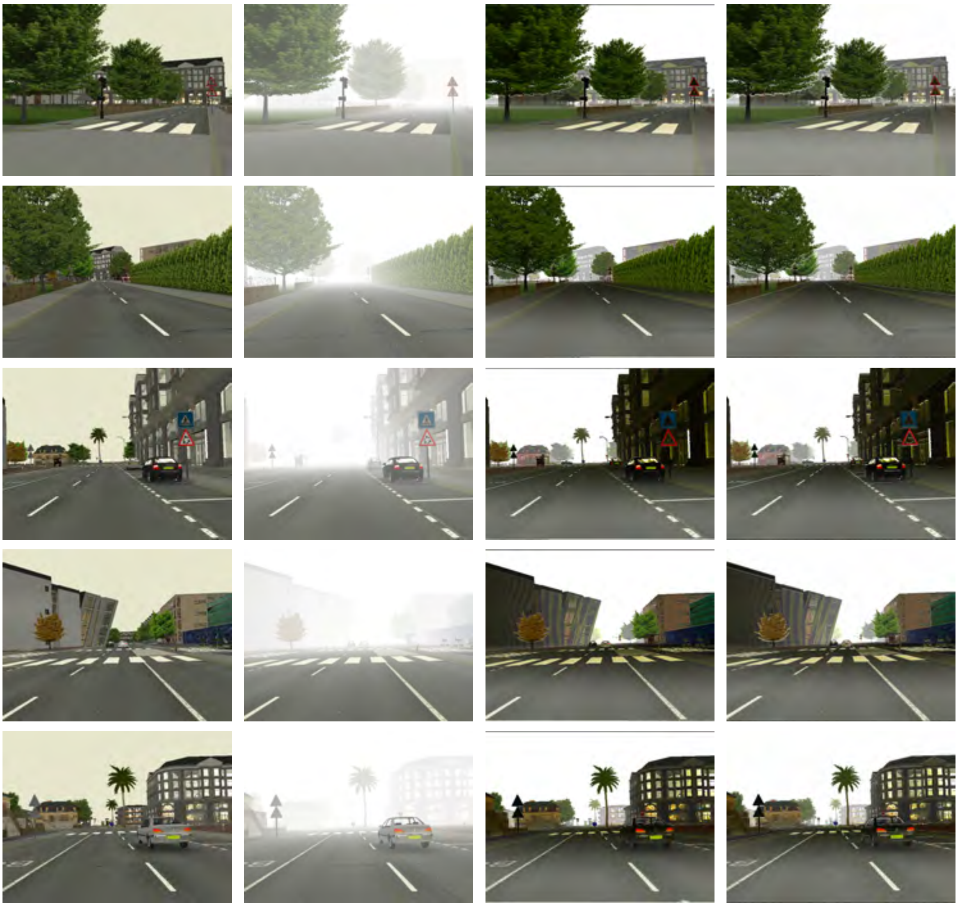


Figure 4.28: Comparison of contrast restoration results between the PC / ARM implementation and the embedded ZYNQ FPGA implementation. Form left to right, the original image without fog, the original image with synthetically added homogeneous fog and the restoration results with the algorithm run on Intel or ARM processor and the FPGA implementation. The used parameters are $k = 15$, $p = 0.99\%$ and $a = 3.5$

slices and LUTs (Look Up Tables) is greatly increased, due to the fact that in the full hardware implementation we use a LUT for the exponential function of the atmospheric veil and have also moved the final computation of the restored image, equation 4.40, in the FPGA logic. Figure 4.29 presents the resource utilization for the most rel-

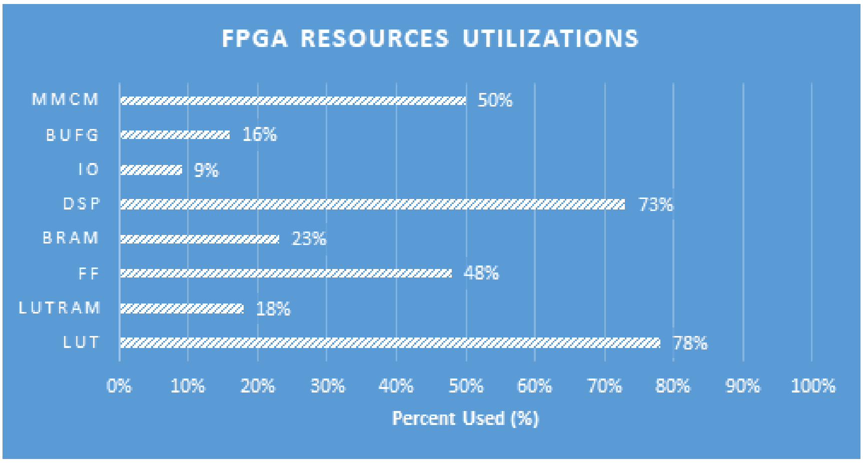
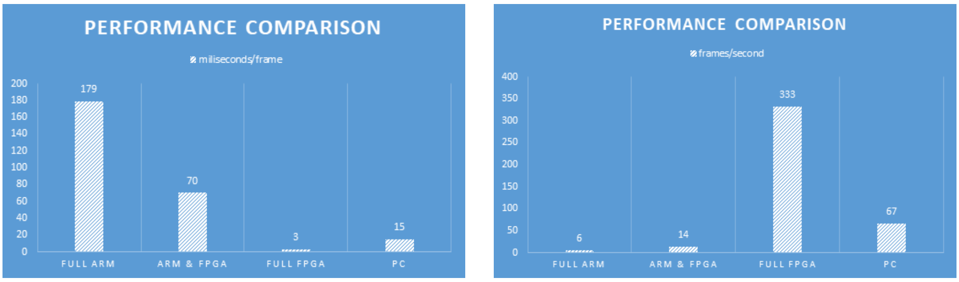


Figure 4.29: ZYNQ FPGA Resources used for Image Defogging

evant implementation – the full hardware one. We present here the actual utilization for each type of hardware component present in the FPGA fabric. It is worth mentioning that more than 50% of the used logic belongs to the embedded contrast restoration algorithm.

The final assessment carried out in this section is the performance evaluation. In the figure below we present the average processing time and the maximum number of frames per second that can be achieved by our three implementations on the ZYNQ embedded platform and we compare the performance of these embedded implementations to the reference PC implementation (standard PC equipped with an Intel



Run-time evaluation

Number of frames per second

Figure 4.30: Performance Analysis of the three embedded solutions and the reference PC one.

i7 3770K processor). In the PC based implementation the defogging algorithm is able to obtain a maximum throughput of about 15frames per second [97] while our full hardware solution is almost 5 times faster, making it a very good candidate to be embedded directly in the camera hardware. Having obtained such results on an entry level FPGA based SoC, we have proved that an embedded image processing solution can outperform a PC based solution, not only in terms of performance but also in terms of cost effectiveness.

4.5 Conclusions

This chapter presented the most relevant methods used for contrast restoration of fog degraded images. We have categorized these methods into the following three classes: non-model based methods and model based methods with and without depth information. The non-model based methods usually achieve poor results, because they try to change the image histogram's shape by only using image related information and thus introduce noise in the resulting images or create different artifacts that were originally not present in the analyzed scene. Other non-model based methods use either the wavelet transform or multi-scale retinex and they require multiple frames of the same scene. This is not possible in dynamic traffic scenes, so these methods are not suitable for advanced driving assistance functions. The model based methods with given depth information would be able to provide satisfactory results, heavily relying on the accuracy of the depth information. Since for the images captured in fog conditions it is very hard to obtain a reliable depth map all these methods try to approximate it, but the results of these approximations are not mathematically solid. Furthermore such approximations are not possible in real time driving situations when the observed scene is constantly changing.

The last category of methods, the model based contrast restoration methods without depth information try to use a single image in order to obtain the fog free representation of the input scene. All these methods use a mathematical model of the observed scene in fog conditions. Some methods are only suited for restoring the contrast of the images when the depth is rather constant, failing to correctly enhance images of road scenes. Other methods try to infer a transmission map or an approximation of the depth in the scene and subtract it from the original foggy image. These methods achieve better enhancement results, but the processing time is rather high, so they are not good candidates for real time driving assistance applications or autonomous driving. Methods based on Koschmieder's law try to restore the contrast of images by inferring the atmospheric veil or the fog addendum in the image. The results are quite good on natural images. For traffic images, new constraints can be added: planar road constraint or to take into account the exponential decay of the fog. The planar road constraint has some drawbacks: it does not produce a smooth atmospheric veil over the whole image (there exists a cut off of the atmospheric veil, which yields poor results for objects that are further away from the camera. Our solution for image defogging is modeling the exponential decay of the fog by using two mathematical functions inspired from the partition of unity in the differential geometry field [93],[92]. Our algorithm not only produces better visual results but it also performs in real time. It is based on the median filtering and standard deviation computed on the columns of the input image (which makes these filters consistent with the presence of fog in the scene), and makes an exponential inference of the atmospheric veil or fog addendum in the image. The results are better especially in the distance since our algorithm can adapt to the density of the fog present in the scene.

We have also tested the quantitative results of several image defogging solutions. We mention here: the rate of new visible edges, the percentage of pixels that become completely black or completely white after restoration, the ratio of gradients at visible edges and the average absolute difference between the images without fog and the restored images (this metric can only be used for synthetic images); other metrics specific for stereo systems: the number of $3D$ reconstructed points and the distance estimation of the object detection and classification in fog conditions.

Another important aspect to consider for autonomous driving or advanced driving assistance functions is the ability to perform image processing functions on specialized embedded hardware. In this chapter we have presented a very robust solution for deploying our state of the art image defogging algorithm to an FPGA based embedded system – the Xilinx ZYNQ SoC. The obtained results were remarkable, the run-time gain was 5 times faster than the original PC based solution with the same accuracy.

Chapter 5

Conclusions

This book presents the authors view about how an image processing based advanced driving assistance function or a fully autonomous vehicle must cope with one of the most severe weather phenomenon that can severely affect the perception task in driving situations. This last chapter presents the main conclusions that can be drawn from the book, together with some general guidelines to be considered for designing autonomous vehicles that should also be capable of driving in fog conditions (even in extreme foggy conditions at reduced speed).

For advanced driving assistance functions or autonomous vehicles the camera, or several cameras, will be one of the main sensors used for the perception task, i.e. to understand what happens in the vehicle's surroundings. For achieving this, several image processing functions are mandatory in order to be able to assess whether the image processing tasks are able to perform with reasonable performance, due to the weather and environment conditions. If this is not the case then the camera must signal that is inoperable.

One of the most dangerous weather phenomena for driving scenarios is fog. So an autonomous vehicle must be able to cope and perform well in such driving situations. A first task that must be performed in an autonomous vehicle is to detect the presence of fog in the scene. A good fog detection algorithm has to provide some relevant information like: the presence of the fog in the scene, the type or category of the fog (the fog's density), an estimation of the visibility distance in the scene, i.e. how far away is the camera able to perceive objects. The majority of the fog detection algorithms for driving assistance are based on the well known Koschmieder law. A fog detection solution must be stable, with as minimum oscillations as possible, providing accurate results. Temporal filtering and integration must also be added to increase the robustness of the fog detection solution. In addition, a good solution in fog conditions should also incorporate the reliable computation of the maximum safe traveling speed on the given road segment. If this is not possible than the autonomous vehicle must rely on other sensors, like radars or laser scanners, in or to proceed. Furthermore, the detected information can be uploaded in the cloud, or sent to regional traffic information system, in order to aid in high level navigation functions that are also necessary in autonomous driving.

The next task to be performed by an autonomous vehicle in fog scenarios is to infer the fog free representation of the scene, i.e. to perform an image enhancement algorithm or image defogging. An important aspect in image defogging is to consider a model based algorithm, since the depth in fog conditions cannot be reliably inferred by a monocular or stereo vision system. In this book, we have presented a state of the art algorithm based on Koschmieder's law and a mathematical model for inferring the atmospheric veil that also takes into account the exponential decay of the fog present in traffic scenes. The derived exponential functions were designed to be applied in the image domain and they provide superior restoration results and higher quality images. The clarity of the reconstructed scene is higher especially in the regions of the image with many details, such as traffic signs, cars and buildings. Furthermore, our algorithm is able to provide increased clarity, even for objects that are further away in the scene. Recent approaches in literature try to reliably perform semantic semantic segmentation tasks directly on the foggy images, and provide pixel wise segmentation of the scene in front of the vehicle [98]. These approaches are new and can be considered as a redundant image processing path for an autonomous vehicle. Nevertheless, these approaches must also be tested on real world traffic scenarios.

Safety is an important issue, not only in autonomous vehicles, but also in advanced driving assistance functions. For this reason the automotive community is considering not only a redundancy in sensors, but also an algorithmic redundancy. This implies that also image processing functions must be dual and a fusion of the output of several image processing algorithms must be performed, together with the fusion of information provided by all relevant sensors in the vehicle (cameras, laser scanners, radars, etc.). This is crucial for achieving automotive safety.

Last but not least, a robust, real time, complete, embedded and low power solution is needed that must incorporate all the above mentioned ideas. In this book, we have presented our algorithmic variants for performing real time image processing on embedded platforms: the fog detection algorithm and driving assistance in fog conditions was successfully ported on an embedded ARM processor, while the image defogging solutions was ported to a Xilinx ZYNQ SoC.

Bibliography

[1] N. Mihai, *IMAGE PRE-PROCESSING SOLUTIONS FOR DRIVING ASSISTANCE IN ADVERSE WEATHER CONDITIONS*. Technical University of Cluj-Napoca, 2015.

[2] William T. Buller, "Benchmarking Sensors for Vehicle Computer Vision Systems ," http://mtri.org/automotivebenchmark.html.

[3] World Health Organization, "The top 10 causes of death," http://www.who.int/mediacentre/factsheets/fs310/en/.

[4] M. Pines, "Top 25 causes of car accidents," http://seriousaccidents.com/legal-advice/top-causes-of-car-accidents/.

[5] Sam Francis, "SAE's full list of levels for autonomous vehicles," http://roboticsandautomationnews.com/2017/06/05/saes-full-list-of-levels-for-autonomous-vehicles/12669/.

[6] M. Negru and S. Nedevschi, "Image based fog detection and visibility estimation for driving assistance systems," in *Intelligent Computer Communication and Processing (ICCP), 2013 IEEE International Conference on*, Sept 2013, pp. 163–168.

[7] M. Green, "Contrast detection and direction discrimination of drifting gratings," *Vision Research*, vol. 23, no. 3, pp. 281–289, 1983.

[8] K. Heidorn and I. Whitelaw, *The Field Guide to Natural Phenomena: The Secret World of Optical, Atmospheric and*

Celestial Wonders. Firefly Books, Limited, 2010. [Online]. Available: http://books.google.ro/books?id=oyXeRgAACAAJ

[9] K. Heidorn, *And Now - The Weather.* Fifth House, 2005. [Online]. Available: http://books.google.ro/books?id=TqrgAAAAMAAJ

[10] Weather Almanac, "Weather Almanac for September 2002 - THE FOG ROLLS IN," http://www.islandnet.com/~see/weather/almanac/arc2002/alm02sep.htm.

[11] C. Starr, C. Evers, and L. Starr, *Biology: Concepts and Applications*, ser. Brooks/Cole biology series. Thomson, Brooks/Cole, 2006. [Online]. Available: http://books.google.ro/books?id=3u1J5v06Tm4C

[12] N. Hautiere and D. Aubert, "Contrast restoration of foggy images through use of an onboard camera," in *Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE*, Sept 2005, pp. 601–606.

[13] W. Middleton, *Vision through the atmosphere.* University of Toronto Press, 1952. [Online]. Available: http://books.google.ro/books?id=DHx6AAAAIAAJ

[14] International Commission on Illumination, *International Lighting Vocabulary.* CIE, 1987, no. 17.4.

[15] N. Hautiere, J.-P. Tarel, J. Lavenant, and D. Aubert, "Automatic fog detection and estimation of visibility distance through use of an onboard camera," *Machine Vision and Applications*, vol. 17, no. 1, pp. 8–20, 2006. [Online]. Available: http://dx.doi.org/10.1007/s00138-005-0011-1

[16] Computer Vision Online, "Camera Calibration," http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT9/node2.html.

[17] E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision.* Upper Saddle River, NJ, USA: Prentice Hall PTR, 1998.

[18] R. C. Gonzalez and R. E. Woods, *Digital Image Processing (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006.

[19] M. Berman, J. Liu, and L. Justison, “Caltrans Fog Detection and Warning System,” ICX technologies, White Paper, 2009.

[20] D. Pomerleau, “Ralph: rapidly adapting lateral position handler,” in *Intelligent Vehicles '95 Symposium., Proceedings of the*, Sep 1995, pp. 506–511.

[21] ——, “Visibility estimation from a moving vehicle using the ralph vision system,” in *Intelligent Transportation System, 1997. ITSC '97., IEEE Conference on*, Nov 1997, pp. 906–911.

[22] N. Hautiere, R. Labayrade, and D. Aubert, “Detection of visibility conditions through use of onboard cameras,” in *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*, June 2005, pp. 193–198.

[23] N. Hautière, R. Labayrade, and D. Aubert, “Estimation of the visibility distance by stereovision: a generic approach.” in *MVA*, 2005, pp. 590–593. [Online]. Available: http://dblp.uni-trier.de/db/conf/mva/mva2005.html#HautiereLA05

[24] N. Hautiere and D. Aubert, “Contrast restoration of foggy images through use of an onboard camera,” in *Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE*, Sept 2005, pp. 601–606.

[25] R. Kohler, “A segmentation system based on thresholding,” *Computer Graphics and Image Processing*, vol. 15, no. 4, pp. 319 – 338, 1981. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0146664X81800159

[26] K. Mori, T. Kato, T. Takahashi, I. Ide, H. Murase, T. Miyahara, and Y. Tamatsu, “Visibility estimation in foggy conditions by in-vehicle camera and radar,” in *Innovative Computing, Information*

and Control, 2006. ICICIC '06. First International Conference on, vol. 2, Aug 2006, pp. 548–551.

[27] K. Mori, T. Takahashi, I. Ide, H. Murase, T. Miyahara, and Y. Tamatsu, "Recognition of foggy conditions by in-vehicle camera and millimeter wave radar," in *Intelligent Vehicles Symposium, 2007 IEEE*, June 2007, pp. 87–92.

[28] V. Cavallo, M. Colomb, and J. Doré, "Distance perception of vehicle rear lights in fog." *Human Factors*, vol. 43, no. 3, pp. 442–451, 2001. [Online]. Available: http://dblp.uni-trier.de/db/journals/hf/hf43.html#CavalloCD01

[29] M. Gabb, S. Krebs, O. Lohlein, and M. Fritzsche, "Probabilistic inference of visibility conditions by means of sensor fusion," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, June 2014, pp. 1211–1216.

[30] A. Westenberger, M. Gabb, M. Muntzinger, M. Fritzsche, and K. Dietmayer, "State and existence estimation with out-of-sequence measurements for a collision avoidance system," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*, June 2013, pp. 612–617.

[31] M. Gabb, O. Lohlein, R. Wagner, A. Westenberger, M. Fritzsche, and K. Dietmayer, "High-performance on-road vehicle detection in monocular images," in *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, Oct 2013, pp. 336–341.

[32] C. Zhao-zheng, L. jia, and C. Qi-mei, "Real-time video detection of road visibility conditions," in *Computer Science and Information Engineering, 2009 WRI World Congress on*, vol. 5, March 2009, pp. 472–476.

[33] N. Hautière, J.-P. Tarel, H. Halmaoui, R. Brémond, and D. Aubert, "Enhanced fog detection and free-space segmentation

for car navigation," *Machine Vision and Applications*, vol. 25, no. 3, pp. 667–679, Apr 2014. [Online]. Available: https://doi.org/10.1007/s00138-011-0383-3

[34] S. Bronte, L. Bergasa, and P. Alcantarilla, "Fog detection system based on computer vision techniques," in *Intelligent Transportation Systems, 2009. ITSC '09. 12th International IEEE Conference on*, Oct 2009, pp. 1–6.

[35] S. Nedevschi, *Prelucrarea imaginilor si recunoasterea formelor.* Ed. Microinformatica, 1997.

[36] R. Gallen, A. Cord, N. Hautiere, and D. Aubert, "Towards night fog detection through use of in-vehicle multipurpose cameras," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, June 2011, pp. 399–404.

[37] R. Gallen, A. Cord, N. Hautiere, E. Dumont, and D. Aubert, "Nighttime visibility analysis and estimation method in the presence of dense fog," *Intelligent Transportation Systems, IEEE Transactions on*, vol. PP, no. 99, pp. 1–11, 2014.

[38] S. Nayar and S. Narasimhan, "Vision in bad weather," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 2, 1999, pp. 820–827 vol.2.

[39] S. Narasimhan and S. Nayar, "Removing weather effects from monochrome images," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 2, 2001, pp. II–186–II–193 vol.2.

[40] S. Narasimhan, C. Wang, and S. Nayar, "All the Images of an Outdoor Scene," in *European Conference on Computer Vision (ECCV)*, vol. III, May 2002, pp. 148–162.

[41] N. Kawasaki, T. Miyahara, and Y. Tamatsu, "Visibility condition determining device for vehicle," Jan. 10 2008, uS Patent

App. 11/820,224. [Online]. Available: http://www.google.com/patents/US20080007429

[42] M. Pavlic, H. Belzner, G. Rigoll, and S. Ilic, "Image based fog detection in vehicles," in *Intelligent Vehicles Symposium (IV), 2012 IEEE*, June 2012, pp. 1132–1137.

[43] M. Pavlic, G. Rigoll, and S. Ilic, "Classification of images in fog and fog-free scenes for use in vehicles," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*, June 2013, pp. 481–486.

[44] N. Hautiere, J. P. Tarel, and D. Aubert, "Mitigation of visibility loss for advanced camera-based driver assistance," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 11, no. 2, pp. 474–484, June 2010.

[45] R. Deriche, "Using canny's criteria to derive a recursively implemented optimal edge detector," *International Journal of Computer Vision*, vol. 1, no. 2, pp. 167–187, 1987. [Online]. Available: http://dx.doi.org/10.1007/BF00123164

[46] S. Se, "Zebra-crossing detection for the partially sighted," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2, 2000, pp. 211–217 vol.2.

[47] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981. [Online]. Available: http://doi.acm.org/10.1145/358669.358692

[48] M. Nagao and T. Matsuyama, "Edge preserving smoothing," *Computer Graphics and Image Processing*, vol. 9, no. 4, pp. 394 – 407, 1979. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0146664X79901023

[49] N. Hautiere, R. Labayrade, and D. Aubert, "Real-time disparity contrast combination for onboard estimation of the visibility

distance," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 7, no. 2, pp. 201–212, June 2006.

[50] F. Jimenez, F. Aparicio, and J. Paez, "Evaluation of in-vehicle dynamic speed assistance in spain: algorithm and driver behaviour," *Intelligent Transport Systems, IET*, vol. 2, no. 2, pp. 132–142, June 2008.

[51] R. Gallen, N. Hautiere, and S. Glaser, "Advisory speed for intelligent speed adaptation in adverse conditions," in *Intelligent Vehicles Symposium (IV), 2010 IEEE*, June 2010, pp. 107–114.

[52] The Engineering Toolbox, "Friction and coefficients of friction," http://www.engineeringtoolbox.com/friction-coefficients-d_778.html.

[53] M. Negru and S. Nedevschi, "Assisting navigation in homogenous fog," in *International Conference on Computer Vision Theory and Applications (VISAPP 2014)*, 2014, pp. 619–626.

[54] S. Bota and S. Nedevschi, "Glsceneint: A synthetic image generation tool for testing computer vision systems," in *Intelligent Computer Communication and Processing (ICCP), 2006 IEEE International Conference on*, Sept 2006, pp. 39–46.

[55] M. Negru, V. Popescu, and S. Nedevschi, "Fog assistance on smart mobile devices," in *Intelligent Computer Communication and Processing (ICCP), 2014 IEEE International Conference on*, Sept 2014, pp. 197–204.

[56] M. Lan, M. Rofouei, S. Soatto, and M. Sarrafzadeh, "Smartldws: A robust and scalable lane departure warning system for the smartphones," in *Intelligent Transportation Systems, 2009. ITSC '09. 12th International IEEE Conference on*, Oct 2009, pp. 1–6.

[57] P. Jeong, A. V. Vesa, A. Rarau, and S. Nedevschi, "Real-time driving assistant application for android-based mobile devices,"

in *Intelligent Computer Communication and Processing (ICCP), 2014 IEEE International Conference on*, Sept 2014, pp. 205–210.

[58] R. Itu and R. G. Danescu, "An efficient obstacle awareness application for android mobile devices," in *Intelligent Computer Communication and Processing (ICCP), 2014 IEEE International Conference on*, Sept 2014, pp. 157–163.

[59] F. Oniga, A. Trif, and S. Nedevschi, "Stereovision for obstacle detection on smart mobile devices: First results," in *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, Oct 2013, pp. 342–347.

[60] A. Trif, F. Oniga, and S. Nedevschi, "Stereovision on mobile devices for obstacle detection in low speed traffic scenarios," in *Intelligent Computer Communication and Processing (ICCP), 2013 IEEE International Conference on*, Sept 2013, pp. 169–174.

[61] A. Petrovai, A. Costea, F. Oniga, and S. Nedevschi, "Obstacle detection using stereovision for android-based mobile devices," in *Intelligent Computer Communication and Processing (ICCP), 2014 IEEE International Conference on*, Sept 2014, pp. 141–147.

[62] R. Varga, A. V. Vesa, P. Jeong, and S. Nedevschi, "Real-time pedestrian detection in urban scenarios," in *Intelligent Computer Communication and Processing (ICCP), 2014 IEEE International Conference on*, Sept 2014, pp. 113–118.

[63] J. Oakley and H. Bu, "Correction of simple contrast loss in color images," *Image Processing, IEEE Transactions on*, vol. 16, no. 2, pp. 511–522, Feb 2007.

[64] Y. Schechner, S. Narasimhan, and S. Nayar, "Instant dehazing of images using polarization," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, 2001, pp. I–325–I–332 vol.1.

[65] S. Shwartz, E. Namer, and Y. Schechner, “Blind haze separation,” in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2, 2006, pp. 1984–1991.

[66] S. Nayar and S. Narasimhan, “Vision in bad weather,” in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 2, 1999, pp. 820–827 vol.2.

[67] S. Narasimhan and S. Nayar, “Chromatic framework for vision in bad weather,” in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 1, 2000, pp. 598–605 vol.1.

[68] ——, “Contrast restoration of weather degraded images,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 6, pp. 713–724, June 2003.

[69] K. Zuiderveld, “Contrast limited adaptive histogram equalization,” in *Graphics Gems IV*, P. S. Heckbert, Ed. San Diego, CA, USA: Academic Press Professional, Inc., 1994, pp. 474–485. [Online]. Available: http://dl.acm.org/citation.cfm?id=180895.180940

[70] Z. ur Rahman and G. A. Woodell, “Retinex processing for automatic image enhancement,” *Journal of Electronic Imaging*, vol. 13, pp. 100–110, 2004.

[71] J. Stark, “Adaptive image contrast enhancement using generalizations of histogram equalization,” *Image Processing, IEEE Transactions on*, vol. 9, no. 5, pp. 889–896, May 2000.

[72] J. J. Rodriguez and C. C. Yang, “High-resolution histogram modification of color images,” *Graphical Models and Image Processing*, vol. 57, no. 5, pp. 432 – 440, 1995. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1077316985710374

[73] D. Jobson, Z.-u. Rahman, and G. Woodell, “A multiscale retinex for bridging the gap between color images and the human observation of scenes,” *Image Processing, IEEE Transactions on*, vol. 6, no. 7, pp. 965–976, Jul 1997.

[74] L. L. Grewe and R. R. Brooks, “Atmospheric attenuation reduction through multisensor fusion,” in *Proc. SPIE*, vol. 3376, 1998, pp. 102–109. [Online]. Available: http://dx.doi.org/10.1117/12.303670

[75] K. Tan and J. Oakley, “Enhancement of color images in poor visibility conditions,” in *Image Processing, 2000. Proceedings. 2000 International Conference on*, vol. 2, Sept 2000, pp. 788–791 vol.2.

[76] S. G. Narasimhan and S. Nayar, “Interactive deweathering of an image using physical models,” in *IEEE Workshop on Color and Photometric Methods in Computer Vision, In Conjunction with ICCV*, October 2003.

[77] J. Kopf, B. Neubert, B. Chen, M. Cohen, D. Cohen-Or, O. Deussen, M. Uyttendaele, and D. Lischinski, “Deep photo: Model-based photograph enhancement and viewing,” *ACM Trans. Graph.*, vol. 27, no. 5, pp. 116:1–116:10, Dec. 2008. [Online]. Available: http://doi.acm.org/10.1145/1409060.1409069

[78] R. Tan, “Visibility in bad weather from a single image,” in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, June 2008, pp. 1–8.

[79] K. He, J. Sun, and X. Tang, “Single image haze removal using dark channel prior,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 12, pp. 2341–2353, Dec 2011.

[80] J. P. Tarel and N. Hautiere, “Fast visibility restoration from a single color or gray level image,” in *Computer Vision, 2009 IEEE 12th International Conference on*, Sept 2009, pp. 2201–2208.

[81] K. He, J. Sun, and X. Tang, “Single image haze removal using dark channel prior,” in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, June 2009, pp. 1956–1963.

[82] N. Hautiere, J. P. Tarel, and D. Aubert, “Towards fog-free in-vehicle vision systems through contrast restoration,” in *Computer Vision and Pattern Recognition, 2007. CVPR ’07. IEEE Conference on*, June 2007, pp. 1–8.

[83] N. Hautiere and D. Aubert, “Free space detection for autonomous navigation in daytime foggy weather,” in *IAPR Conference on Machine Vision Applications (MVA’09*, 2009.

[84] J. P. Tarel, N. Hautiére, A. Cord, D. Gruyer, and H. Halmaoui, “Improved visibility of road scene images under heterogeneous fog,” in *Intelligent Vehicles Symposium (IV), 2010 IEEE*, June 2010, pp. 478–485.

[85] J. P. Tarel, N. Hautiére, L. Caraffa, A. Cord, H. Halmaoui, and D. Gruyer, “Vision enhancement in homogeneous and heterogeneous fog,” *Intelligent Transportation Systems Magazine, IEEE*, vol. 4, no. 2, pp. 6–20, Summer 2012.

[86] J. P. Tarel, A. Cord, H. Halmaoui, D. Gruyer, and N. Hautiere, “Frida - fog road images database,” http://www.lcpc.fr/english/products/image-databases/article/frida-foggy-road-image-database.

[87] K. Perlin, “An image synthesizer,” *SIGGRAPH Comput. Graph.*, vol. 19, no. 3, pp. 287–296, Jul. 1985. [Online]. Available: http://doi.acm.org/10.1145/325165.325247

[88] ——, “An image synthesizer,” in *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH ’85. New York, NY, USA: ACM, 1985, pp. 287–296. [Online]. Available: http://doi.acm.org/10.1145/325334.325247

[89] M. Negru, S. Nedevschi, and R. I. Peter, “Image enhancement in daytime fog conditions,” *Automation, Computers, Applied Mathematics (ACAM)*, vol. 21, no. 2, pp. 129–136, 2012.

[90] N. Hautiére, J.-P. Tarel, D. Aubert, and E. Dumont, “Blind contrast enhancement assessment by gradient ratioing at visible edges,” *Image Analysis & Stereology*, vol. 27, no. 2, 2011. [Online]. Available: http://www.ias-iss.org/ojs/IAS/article/view/834

[91] M. Negru, S. Nedevschi, and R. I. Peter, “Exponential image enhancement in daytime fog conditions,” in *Intelligent Transportation Systems - (ITSC), 2014 17th International IEEE Conference on*, Oct 2014, pp. 1675–1681.

[92] M. Spivak, *A Comprehensive Introduction to Differential Geometry. Vol. I*, 2nd ed. Wilmington, Del.: Publish or Perish Inc., 1979.

[93] ——, *Calculus on Manifolds: A Modern Approach to Classical Theorems of Advanced Calculus*, 2nd ed. Menlo Park,Calif.: W.A.Benjamin, 1968.

[94] C. Pantilie and S. Nedevschi, “Sort-sgm: Subpixel optimized real-time semiglobal matching for intelligent vehicles,” *Vehicular Technology, IEEE Transactions on*, vol. 61, no. 3, pp. 1032–1042, March 2012.

[95] B. Coseriu, M. Negru, and S. Nedevschi, “Contrast restoration of foggy images on the zynq embedded platform,” in *2016 IEEE 12th International Conference on Intelligent Computer Communication and Processing (ICCP)*, Sept 2016, pp. 207–214.

[96] Xilinx Inc., “Axi reference guide,” https://docslide.net/documents/vivado-design-suite-axi-reference-guide-ug1037.html, 2015.

[97] M. Negru, S. Nedevschi, and R. I. Peter, "Exponential contrast restoration in fog conditions for driving assistance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2257–2268, Aug 2015.

[98] C. Sakaridis, D. Dai, and L. Van Gool, "Semantic foggy scene understanding with synthetic data," *International Journal of Computer Vision*, vol. 126, no. 9, pp. 973–992, Sep 2018. [Online]. Available: https://doi.org/10.1007/s11263-018-1072-8