Adaptive Contrast Enhancement Involving CNN-based Processing for Foggy Weather Conditions & Non-Uniform Lighting Conditions

Christopher Schwarzlmüller*, Fadi Al Machot*, Alireza Fasih* and Kyandoghere Kyamakya*
*Alpen-Adria Universität Klagenfurt
9020 Klagenfurt, Universitätsstrasse 65–67, Austria
Email: forename.surname@uni-klu.ac.at

***Abstract*—Adaptive image processing in the context of Advanced Driver Assistance Systems (ADAS) is a crucial issue because bad weather conditions lead to poor vision. In a foggy weather, image contrast and visibility are low due to the presence of airlight that is generated by scattering light, which in turn is caused by fog particles. Since vision based ADAS are affected by inadequate contrast, a real-time capable solution is required.**

To improve such degraded images, a method is required which processes each image region separately. Hence, real-time processing is required, the method is realized with the CNN paradigm which claims the characteristic of real-time image processing.

To compare the proposed method with existing state-of-the-art methods the Tenengrad measure is applied.

***Index Terms*—Cellular Neural Network, CLAHE, weather degraded image restoration, real-time image processing, adaptive contrast enhancement.**

I. Introduction

Bad weather conditions in traffic lead to an increased risk of accidents. In such conditions, the contrast and color of images are drastically altered and/or degraded which means that these images suffer from poor contrast. Fog is among the most challenging of weather conditions. It is a phenomenon caused by tiny droplets of water in the air which not only blurs an image but also negatively affects its contrast.

Over the last decades there has be a growing need to develop automated systems for image interpretation. In order to adequately interpret an image, it is necessary that it is free from noise or other aberrations. Contrast is the local change in brightness and is defined as the ratio between average brightness of an object and the background [1]. Hence, it is necessary to remove disturbing weather effects from images in order to make vision systems more reliable [2]. This makes the development of an adaptive vision system, which detects "important" features, interesting. Such a system can enhance a sequence of low quality images to provide the data for Advanced Driver Assistance Systems (ADAS) sub systems. The aim of our contribution is, to enhance traffic images for further processing of vision based ADAS. Such systems are e.g. Lane Departure Warning Systems (LDWS) or Adaptive Cruise Control (ACC) which are very crucial on time.

Processing speed and image quality are two main factors in ADAS systems which must be provided by a video preprocessing unit. In this paper our main target is to show as to how CNN can be used to enhance the contrast of images taken in foggy weather. The effects that fog has on an image are neither uniform nor constant. This means that depending on the depth of the scene, the fog's intensity on the image will vary. The type of image degradation due to fog or mist is known as atmospheric degradation. It is not always possible to know the actual structure of a scene a priori and thus to adaptively handle the fog's influences. This fact leads to different approaches which either use a priori knowledge (information from sensors) or do not use any depth information at all. The intensity of fog effects varies across the scene and grows exponentially with respect to the depth of the scene. In other words, the conventional image enhancement techniques would not be suitable to obtain the original objects from the image. It is necessary to model the degrading mechanism of fog to reconstruct the scene [3]. Methods which restore image contrast under bad weather conditions are encountered more often in literature. Unfortunately, they are all rather restricted to be used aboard a moving vehicle.

This paper is organized as follows: In section II we discuss state-of-the-art approaches to restore images that have been degraded by foggy weather and methods to measure visible distance in the same conditions. Our proposed image processing architecture for fog removal will be presented in section III. In section IV we want to compare and contrast the results yielded by the different methods and present the results of our approach. Finally, we provide concluding remarks as well as an outlook in section V.

II. State-of-the-art

This section provides a critical review of relevant methods from literature which are used for enhancing contrast of degenerated images.

Alwani and Tiwari [4] propose a method for enhancing foggy degenerated images. The method does not require a model for eliminating the effects of fog and works in the *Hue Intensity Saturation* (HIS) color space. The authors argue, that the human visual system is highly sensitive to the variation of the intensity, while the change of saturation and hue is neglectable. The proposed algorithm processes the saturation and intensity layer of the input image, while the hue layer remains constant. The intensity layer is processed block wise, since foggy weather affects every image region in a different way (dependent on the scene depth). This procedure ensures local contrast enhancement for each region. The basic enhancement is based on thresholds, which are determined by experiments, and stretching. The saturation layer is modified by using gamma correction. The modified intensity and saturation layers are combined with the unaltered hue layer and converted in the RGB color space. The shortcoming of this method is that the thresholds and the gamma correction exponent are fixed. Due to the fact, that the authors chose a block-size of 64×64 pixels, we can not mimic this algorithm by using CNN, since such a template size is not feasible.

Wang *et al.*[5] propose a method which is based on the atmospheric scattering method and color analysis. First, the brightness of the degraded image is enhanced by using a fixed threshold. After that, three chromaticity components of the enhanced image are extracted. By this procedure it is possible to express an image with the atmospheric scattering model. For the further enhancement process, the approach uses a set of thresholds and histogram equalization for enhancing the image' brightness.

Jia *et al.* [6] developed a method for local contrast enhancement of foggy and rainy weather degenerated images. To achieve the local contrast enhancement, the method involves the CLAHE algorithm [7], [8]. To achieve good results for foggy and rainy weather to meet the *Human Visual System* (HVS), the histogram clipping is modified. The computation for the clipping limit involves the *Block-Edge Impairment Metric* [9]. Due to this, the clipping of the local histograms is well suited for foggy and rainy weather degenerated images. In addition, the authors use two post-processing steps. The purpose of the first step is to adjust the illumination of the actual frame. This step is required, due to the fact, that image sequences are processed. Yet, contrast enhancement techniques introduces noise in an image, the second step consists of smoothing the image. It is justified that CLAHE is able to process images in real-time [7]. Further, the results are promising and a model for the degeneration effect is not necessary.

In [10] a method is presented for a simple correction of contrast loss in foggy color images. Since the method works in RGB color space it applies a cost function to the RGB channel. The drawback of this method is that it is based on the assumption that airlight is constant across the whole image.

Oakley [11] presents a method which relies on a priori information about the scene depth to enhance the contrast of an image. He uses a physics-based method to restore scene contrast without any predicted weather information by approximating the distribution of radiances in the scene based on a single Gaussian with known variance.

Narasimhan *et al.* [2] presented a physics-based model which describes the appearances of scenes in uniform bad weather conditions. This solution does not necessitate a priori information about the scene structure or detailed knowledge of the particular weather condition and can be used for gray scale, RGB color or IR images. For contrast enhancement they use a method which segments the image in depth before applying contrast stretching or the histogram stretching method. The fact that a method based on physical model needs scene depth information is impractical in our case. Further, Narasimhan *et al.* [2] were able to restore an image using a scene-depth map. The disadvantage of this method is that every scene must be available in both bad weather conditions as well as in good weather. Hence, this approach is not suitable for an unknown outdoor environment.

Schechner *et al.* [12] present a polarization filtering method to reduce the effect of fog on images. The basic idea of this approach is based on the fact that the natural illuminating light scattered by atmospheric particles (airlight) is partially polarized. This method works well for hazy weather conditions but if fog appears, this method has either a limited effect or even fails altogether.

Hautière *et al.* [13] present a method for contrast enhancement and reduction of effects of foggy weather by using a single camera inside a moving vehicle. Weather conditions are estimated and used to restore the contrast according to a road scene structure, which is inferred a priori and then refined during the contrast restoration process. The authors assert that the scenes are flat and not too complex. So, they work with simple street scenes. One of the first steps to remove the effects of fog is to compute the extinction coefficient of fog using a single camera behind the vehicle's windshield. To acquire this extinction parameter, it is necessary to recover the parameters of Koschmieder's law. For reconstructing the scene each pixel exhibits its own depth information. Using this additional information it is possible to enhance the image depth dependently.

Zhai and Liu [14] present an algorithm for enhancing fog-degraded image contrast which does not take any scene depth information beforehand and which makes use of a moving mask. Pixels in such a mask have the same scene depth. The complexity of computation is very high. The idea of the authors is that the pixels in a mask have the same scene depth and the contrast has to be enhanced in every mask separately. The procedure involves two steps: firstly, the sky region is segmented in order to prevent over-enhancement in the flat region as well as to reduce noise; secondly the sky pixels in the masks are removed and the histogram information is modified in the mask. Thus, the modified partially overlapped histogram equalization function for enhancement can be gotten. The method does not require any scene depth information. Instead it solves the de-weathering image problem by a local image enhancement technique that is using a moving mask to segment the scene into different depths. Then each pixel is processed in the mask by sub-block overlapped histogram equalization. At the same time, in order to avoid bringing noise amplification into the sky region, the sky region is segmented before it is enhanced according to the characteristic of gray distribution within said region. We were inspired by this method because we apply the CNN paradigm it is possible to implement this method for real-time applications. This technique is accurate enough to be used for ADAS in foggy weather situations. In

2009 Zhai and Zhang [15] improved their algorithm by using a depth map of the scene as well as using a genetic algorithm to optimize the result. The algorithm still needs a single image for restoration.

John *et al.* [16] use wavelet fusion for enhancing weather degraded color images. This method is also well suited for video sequences because it is suitable for real time applications. The method firstly applies a contrast enhancement on the image using depth information and computes the value of airlight present in the image by optimizing a cost function. Next, the value of airlight in the image is computed. The last step of this method involves a wavelet fusion method to obtain the final result.

Xianqiao *et al.* [17], [18] use a Retinex based algorithm for foggy image enhancement. Every image can be decomposed into two different images, the reflectance image r and the illumination image S. The idea in [17] is to enhance the image with sky information. More precisely, the sky region is used to estimate the illumination of the whole image. In [18] an improved Single-Scale Retinex (SSR) algorithm is used to enhance the image. Both approaches seem to work well but even though there are limitations which will not be elaborated on further.

Dongjun *et al.* [19] invented a method for the enhancement of fog degenerated images based on the human visual model. They improve the method of Oakley [10] by using a human visual model which is based on a cost function. Furthermore, an airlight map is generated by modeling the relationship between pixel coordinates and the airlight. A fog degraded image is restored by subtracting the airlight map from the degraded map.

Yitzhaky *et al.* [20] demonstrate an image restoration approach of atmospherically degraded images. By knowing the weather conditions and some other parameters (i.e. type of ground surface, date) the method is capable to restore weather degraded images. Additionally, there are also a number of methods to detect fog in outdoor environments.

In [21] a method is presented which is able to measure the visible distance in foggy weather, using only one camera. The proposed method is able to detect fog and to compute visibility distances only in daylight conditions. The approach is based on a dynamic implementation of Koschmieder's law. The presented method works with a single camera and necessitates the presence of just the road and the sky in the scene. The aim of the system is to inform the driver about the visibility distance so that the driver is able to adjust the driving speed.

One characteristic of using only one camera is that it is not possible to compute the distance to a feature within only one frame. A solution which uses one camera is proposed in [21] is able to overcome these obstacles and to compute the parameter of Koschmieder's law and the visibility distance as well. The proposed method in [22] is able to judge fog density, using in-vehicle camera images and millimeter-wave (mm-W) radar data.

III. Concept description

The proposed method uses two existing approaches which appear in literature and partially mimics them with the Cellular Neural Network (CNN) paradigm. Since our goal is to enhance non-uniformly degraded poor contrast images, we need a method which operates locally. For this purpose, we consider the Contrast Limited Adaptive Histogram Equalization (CLAHE) [8], [7]. This method processes histogram equalization for each image region separately. The method is modified for enhancing foggy weather degenerated images by changing the computation of the clipping limit (including, the variance and the BIM [9]), which is used for the histogram equalization [6]. Since an interpolation step for each enhanced image region is required (we can distinguish regions due to observing the transition), we can observe blurring. In contrast to the original CLAHE, where bi-linear interpolation is applied, the interpolation approach of [23] is used. To remove that blur and to sharpen the image, we use an unsharp mask [24], [25]. The general principle is as follows. Firstly, a blurred version of the original (degraded) image must be generated. This is achieved by using a Gaussian kernel. Secondly, this blurred version is subtracted from the original image. The result is a difference image, which allows us to sharpen the image. Thirdly, we add this difference image to the original image and obtain a sharpened version of degraded image.

These methods are partially mimicked by calculating CNN templates, which are able to provide the same functionality. By cascading different CNN processors or by updating the templates of a single CNN processor, it is possible to achieve the discussed behavior of local contrast enhancement and image sharpening. A detailed description of the proposed method is given in the next section.

A. Explanation of the essential elements + Justification

This section illustrates the framework for local image contrast enhancement in detail. As mentioned in the previous section, the approach mainly consists of two principal methods; local contrast enhancement by using CLAHE and sharpening the image by unsharp masking.

The first step in the proposed approach is to mimic the functionality of CLAHE with CNN (see Fig. 1 and Fig. 2). We use the HSV (Hue, Saturation, Value) instead of the RGB color model due to the fact, that the color information is stored in the Value layer. Due to this, we only need to process this layer. The H and S layer remain constant. To mimic CLAHE, a deep understanding of the method is required. The main steps of CLAHE are as follows:

1) separate the image into tiles;
2) compute the clipping limit for each tile;
3) apply histogram equalization on each tile;
4) interpolate between neighboring tiles to reduce visible transitions.

1) Local Contrast Enhancement: The partition of an $M \times N$ image $\mathbf{I}$ defined as function of $\mathbf{I}(i,j) \mapsto [-1,1]$ (where $i,j \in \mathbb{N} \wedge 0 \leqslant i < M \wedge 0 \leqslant j < N$) is necessary to achieve local contrast enhancement. This is beneficial for non uniformly illuminated image regions or e.g. foggy weather degraded images. The resulting image segments (or tiles) are separately enhanced by using histogram equalization. Due to the fact, that each tile could consist of a different number of gray-scale values, it is essential for parallel processing, to equilibrate the number of gray-scale values. Therefore, the maximum count of gray-scale values of each tile is determined. The gray-scale values of each tile are supplemented by a number of specific gray-scale values which are not affecting the further processing.

For example, a gray-scale image consists of pixels with the intensity in the interval of $[-1,1]$. This image is separated into a fixed number of $m \times n$ (where $m,n \in \mathbb{N}\backslash\{0\}$) tiles, arranged in a rectangular grid of the dimension $\frac{M}{m} \times \frac{N}{n}$. For the sake of simplicity it is assumed, that $\frac{M}{m} = K$ and $\frac{N}{n} = L$ are natural numbers, otherwise there exist different sized tiles. The set of tiles is denoted by S. A single tile is represented by $\mathbf{T}_{i,j} \in S$ ($0 \leqslant i < K$ and $0 \leqslant j < L$) where the sub-indices denote the coordinates of the tile in the rectangular grid. The different gray-scale values of each tile are stored in a vector $\mathbf{g}_{i,j}$ and is ordered by the size. The maximum number of different gray-scale values is computed by (see equation (1))

$$g_{\max} = \max_{0\leqslant i<K \wedge 0\leqslant j<L}(|\mathbf{g}_{i,j}|) \tag{1}$$

where $|\mathbf{g}_{i,j}|$ provides the number of different gray-scale values of tile $\mathbf{T}_{i,j}$ and $g_{\max}$ denotes the maximum count. To ensure that each tile exhibits the same number of gray-scale values, the value -1 is appended at the front of each gray-scale vector until the number of value is equal to $g_{\max}$. The vector which is extended by this procedure is denoted by $\mathbf{gad}_{i,j}$. After balancing the number of gray-scale values for each tile the new distribution of the values is computed according to equation (2)

$$\begin{gathered} g'_{i,j}(r) = \frac{g_{\max}-1}{m\cdot n}\cdot\sum_{k=0}^{r}\mathbf{h}_{i,j}(k) \\ k = 1,2,3,\dots,g_{\max}-1 \end{gathered} \tag{2}$$

where $h_{i,j}(k)$ is the histogram and $g'_{i,j}(r)$ is the new pixel value of tile (i,j) [7]. Due to that, a higher contrast for each tile is ensured. This is achieved by involving a CNN processor with 3×3 templates. The calculation of the corresponding CNN templates is shown in the next section and is influenced by the work of [26].

The corresponding partial implementation with the CNN paradigm is shown in Fig. 1. According to the description before, the value (V) layer of an image is separated into tiles. To remove noise which could affect the further processing, a noise filter which is realized by using the CNN paradigm is applied. The following steps are also equal to the description before. The maximal number of gray-scale values is determined, the BIM value and the clipping limit for each tile are computed and the balanced gray-scale value vector is generated.

The processing of a single tile for a specific gray-scale value is illustrated in Fig. 2. The idea stems from [26] and is as follows; a specific gray-scale value is used as bias value for the CNN processing. The output of module (1) is a binary map TMap_k. Values which are smaller than the given bias value are now set to 1 otherwise 0. The next step is to calculate the pixels which are smaller than gray-scale value k but are greater than gray-scale value k-1 (at start-up, the corresponding threshold map is equal to 0). This is achieved by using the XOR function (2). The resulting map is denoted by SMap and is used for calculating new pixel intensities. Therefore, it is necessary to calculate the new gray-scale values $\hat{g}$ (3) which is used as template value in module (4). The final module (5) stretches the new gray-scale values to the interval of $[-1,1]$. The resulting output of module (5) is denoted by MapHE.

2) Interpolation: The next step is to calculate the clipping limit for the contrast expansion for each tile separately since pixels are enhanced to a maximum intensity. To compute a suited clipping limit for foggy weather degraded images, it is necessary to modify the clipping equation. The modification is done according to [6]. The formula to calculate the clipping limit is shown in equation (3)

$$\text{ClipLimit} = \text{avg.} + \lambda\cdot\sigma^2_{i,j}\cdot(\text{max.} - \text{avg.}) \tag{3}$$

where the variables $avg.$, λ, $\sigma^2_{i,j}$, and $max.$ denotes the average pixel value of the whole image, the BIM [9] value of the whole image, the variance in tile $T_{i,j}$ and the maximum pixel value of the whole image. The BIM value automatically determine the "blockiness" of the input image and the smaller the BIM value or the more block artifacts in an image occur, the less the contrast enhancement. The idea for involving the variance is determine the homogeneity of the tile. If the texture a tile is less (homogeneous), the lower the variance[6]. Relating to images taken in foggy weather, the variance of a tile decreases with growing distance to the observer. All values occurring in equation (3) are calculated without using the CNN paradigm. Now, the gray-scale values of each tile are clipped by using the corresponding clipping limit denoted by ClipLimit.

An image which arises after the above steps suffers from visible transitions in-between neighboring tiles (see Fig. 3) due to different shapes and equalized histograms in tiles. In the original CLAHE algorithm these disturbances are reduced by using a bi-linear interpolation in-between the neighboring blocks [8], [27]. The approach which is used in this paper is slightly different. The new pixel values of a tile are approximated by a Gaussian distribution and involving all direct neighboring tiles [28]. Further, the approximation depends on the Euclidean distance (see equation (4)) in-between two pixels (x_1,y_1) and (x_2,y_2).

$$d((x_1,y_1),(x_2,y_2)) = \sqrt{(x_1-x_2)^2 - (y_1-y_2)^2} \tag{4}$$

In this paper, distances between the pixel which has to be interpolated and the neighboring tiles are computed (see Fig. 4).

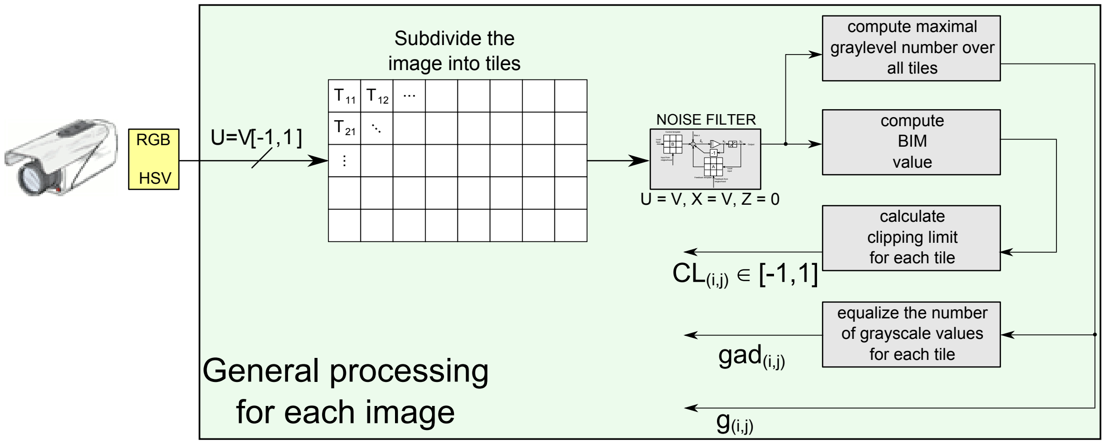

RGB
HSV
U=V[-1,1]
Subdivide the image into tiles
T11
T12
T21
NOISE FILTER
U = V, X = V, Z = 0
compute maximal graylevel number over all tiles
compute BIM value
calculate clipping limit for each tile
CL(i,j) ∈ [-1,1]
equalize the number of grayscale values for each tile
gad(i,j)
g(i,j)
General processing for each image

Fig. 1. Flow chart of the general processing of each image.

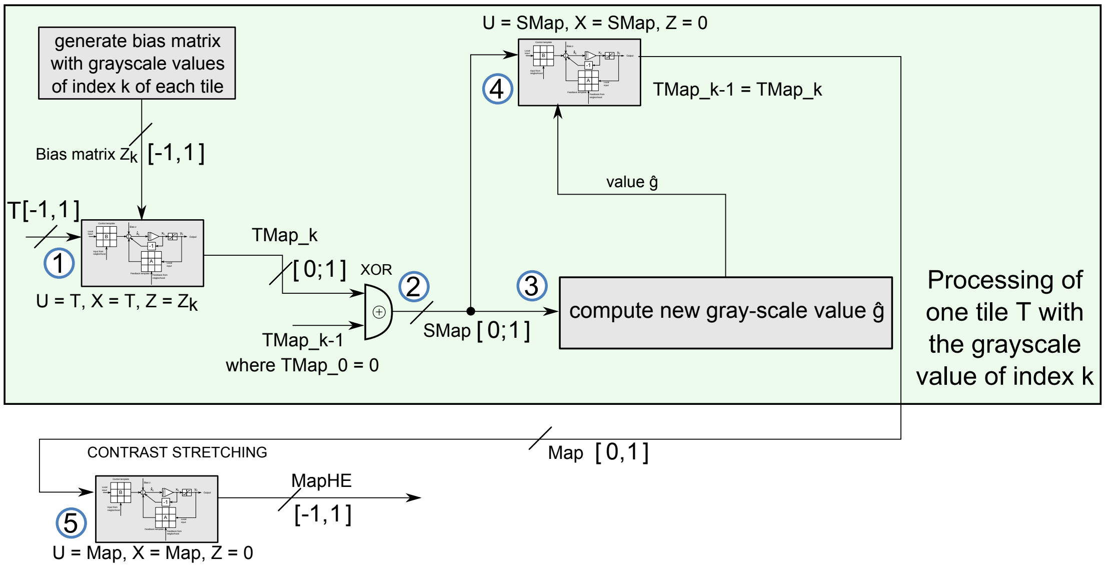


Fig. 2. Flow chart of a single tile for a specific gray-scale value with the index k.

Due to this a smoother transition in-between neighboring tiles is achieved.

These Euclidean distances d are used in formula for the Gaussian distribution shown in equation (5)

$$G(d) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{d}{\sigma}\right)^2\right) \qquad (5)$$

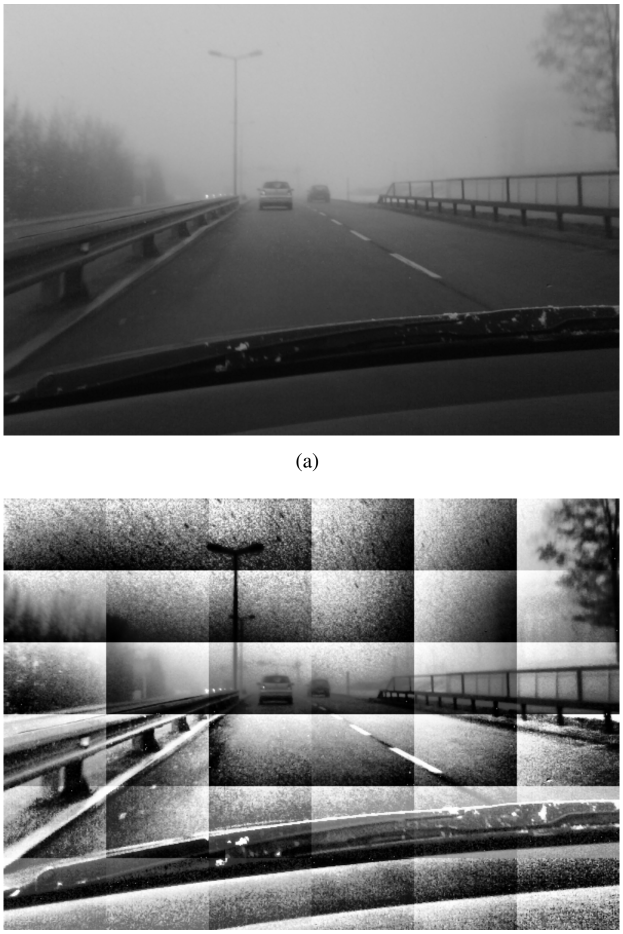
This distribution makes sure, that pixels with lower distances to the neighbor tiles gain a higher weighting. To have a greater influence on the weighting the high of the bell curve is controlled by an additional scaling parameter γ [28].

Figure 1 and Fig. 2 illustrates the first few steps of CLAHE processing by using the CNN paradigm. As a consequence, the interpolation formula for removing the transitions in-between tiles is as follows (equation (6) [28])

$$\hat{g}_{i,j}(r) = \frac{\sum\limits_{e=1}^{8} \left(g'_{i,j}(r) \cdot \gamma G(d_e)\right)}{\sum\limits_{e=1}^{8} \gamma G(d_e)} \qquad (6)$$

The variable $\hat{g}_{i,j}(r)$ denotes the new gray-scale value and the configurable parameters of equation (6) are the scaling parameter γ and the standard derivation σ. Both parameters control the size of the bell curve and how strong the histogram of the to interpolated pixel is compared to the neighboring tiles.

The aim of contrast enhancement is to separate an object from the background without inducting noise. Further, it is important that the quality of enhancement depends on the chosen size of the tiles and the parameters for interpolation. If the tiles are too large it is possible that occurring objects, consisting of rare gray-scale values, are hard to perceive in the equalized image. Therefore, the size of the tiles must be chosen



(a)

(b)

Fig. 3. Fig. (a) shows the original image and Fig. (b) shows the same image after applying the CLAHE method (with exception of the clipping and interpolation step) with too small chosen tiles. The transitions in-between tiles are visible.

well. The interpolation parameters are strongly dependent on the tile size and must be chosen in that way that no transitions in-between the tiles are visible.

3) Unsharp masking: The next step is to apply the unsharp masking (see Fig. III-A3) on the image for the purpose of sharpening. Since the unsharp masking method is highly sensitive to noise a CNN based noise filter is applied on the image before. After this step a 5×5 Gaussian kernel is used to blur the noise filtered image $I_{(2)}$. The smaller the dimensions of the kernel the sharper the resulting image. If the kernel is chosen larger, a local contrast enhancement is achieved. The resulting image is denoted by $I_{(3)}$. The next step unsharp masking method is to calculate the difference image $I_{(4)} = I_{(2)} - I_{(3)}$. The final step consists of combining the noise filtered image $I_{(2)}$ with image $I_{(4)}$ multiplied with a scaling parameter $k \in \mathbb{R}$. The final output of the proposed method is $I_{(5)} = I_{(2)} + k \cdot I_{(4)}$.

An abstract formulation of the algorithm is shown in algorithm 1. The next section illustrates the calculation of the CNN templates for the proposed method.

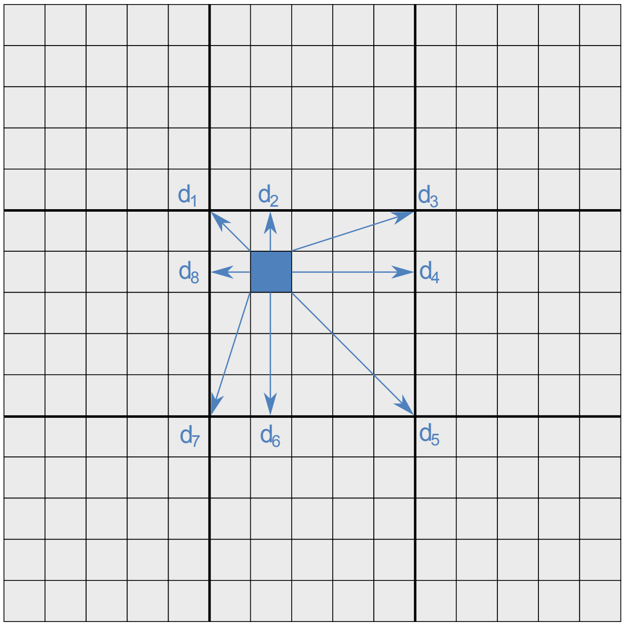


Fig. 4. Schematic illustration for differences in-between actual pixel and neighboring tiles (adapted from [28]). These distances are used to calculate the weight for interpolation.

B. Template calculation in detail

In this section the approach for calculating CNN templates for the proposed method is shown. First of all, the CNN templates for the noise removal are shown [29]. The approach for deriving the templates is based on the *maximum posterior probability* (MAP) approach. The general procedure is to minimize CNN energy function [29] and to apply the MAP estimation on the minimized Gibbs model cost function. This is followed by using the corrupt image model for noisy images and the maximum entropy (ME). Zhao *et al.* [29] analyzed the derived equations and calculated CNN templates which are shown in equation (7).

$$\mathbf{A} = \begin{pmatrix} 1.0 & 1.0 & 1.0 \\ 1.0 & 1.05 & 1.0 \\ 1.0 & 1.0 & 1.0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad z = 0 \quad (7)$$

Parameters such as the time step or iteration count are set to $\Delta h = 0.001$ and $t_{max} = 2$ respectively.

The template calculation for the thresholding is straight forward. The task of the template is, to remove all pixel values which are lower than a given gray-level threshold. The bias value Z_{ij_g} will fulfill the purpose of the threshold. The subindex g denotes the index of the actual element of the balanced sorted gray-level vector which is described in the last section. In case of p gray-levels the CNN processor requires p iterations since each gray-scale value act as threshold. Further, each tile has different gray-scale values in its balanced sorted vector. This is realized by using a threshold matrix for each time step. An image which is processed by a CNN processor using the templates in equation (8) is a binary image. Pixel

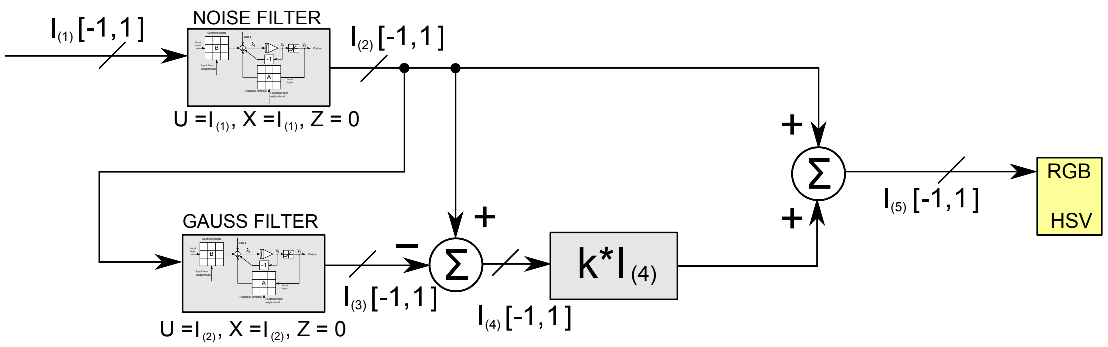


Fig. 5. Flow chart of unsharp masking.

Algorithm 1 Procedure for enhancing foggy weather degraded images

```
Subdivide the input image $I$ into $m \times n$ tiles
Apply a CNN based noise filter to remove noise
Compute the variable $\lambda$ which represents the $BIM$ of the whole image
Prepare histogram $h_{(x,y)}$ and a sorted gray-scale value vector $\mathbf{g}_{x,y}$ for each tile $T_{(x,y)}$
Extend the gray-scale value vector to a vector $\mathbf{gad}_{x,y}$
for $i = 0$; i $<$ number of tiles in a row; i++ do
  for $j = 0$; j $<$ number of tiles in a column; j++ do
    Redistribute histogram $h_{(i,j)}$ for tile $T_{(i,j)}$
  end for
end for
Compute the maximum pixel intensity $max$ of the image $I$
Compute the average pixel intensity $avg$ of the image $I$
for $i = 0$; i $<$ number of tiles in a row; i++ do
  for $j = 0$; j $<$ number of tiles in a column; j++ do
    Compute the variance $\sigma^2_{(i,j)}$ for tile $T_{(i,j)}$
    Compute the clipping limit $CL_{(i,j)}$ for tile $T_{(i,j)}$ by using the formula of [6] using the clipping limit.
    Compute new pixel values for each tile $T_{(i,j)}$ by using the clipping limit $CL_{(i,j)}$
  end for
end for
Generate eight $(m \times n)$ distance maps for the Euclidean distances which are require for the algorithm of [23]
for $i = 0$; i $<$ number of tiles in a row; i++ do
  for $j = 0$; j $<$ number of tiles in a column; j++ do
    Interpolate tile $T_{(i,j)}$ by using the distances to the neighbor tiles their maps, and the algorithm of [23]
  end for
end for
$I_{(1)}$ is the resulting image
Apply a CNN based noise filter on $I_{(1)}$ to obtain a noise reduced image $I_{(2)}$
Generate a blurred version $I_{(3)}$ of image $I_{(2)}$ by using a Gaussian kernel
$I_{(4)} = I_{(2)} - I_{(3)}$ is the difference image
Compute final image $I_{(5)} = k \cdot I_{(4)} + I_{(2)}$ where $k$ is a parameter to control the enhancement
```

values which are greater than the corresponding threshold value saturates to white (-1), otherwise to black ($+1$).

$$\mathbf{A} = \begin{pmatrix} 0.0 & 0.0 & 0.0 \\ 0.0 & 2.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad z = Z_{ij_g} \quad (8)$$

The task of counting pixels and calculating the new pixel values for the histogram equalization is done by conventional computation methods.

The last template for mimicking the CLAHE method is used for contrast stretching. Contrast stretching enhances the contrast within a sub-range of the intensity values by linearly remapping these intensity values to occupy the entire output range. Of course the disadvantage of the method is that a noisy image could corrupt the pre-processing steps (finding minimum and maximum intensity pixels). Hence, the image must be filtered in a preceding step — by using a noise filter. In the following the method of CNN based contrast stretching [30] is described. The general function of contrast stretching can be found in equation (9), where pv is the actual pixel value, pv_{min} is the minimum pixel value in the observed area (in this case, -1) and pv_{max} is the maximum pixel value in said area (in this case, $+1$).

$$f(pv) = -1 + 2\frac{pv - pv_{min}}{pv_{max} - pv_{min}} \quad (9)$$

For sake of simplicity, the minimum and the maximum value are calculated by using conventional methods. After having ascertained the minimum and the maximum we are able to proceed with the contrast stretching. To get the templates we must rewrite equation (9) in the following steps:

$$
\begin{aligned}
f(pv) = -1 + 2\frac{pv - pv_{min}}{pv_{max} - pv_{min}} &= \\
-\frac{pv_{max} - pv_{min}}{pv_{max} - pv_{min}} + \frac{2pv - 2pv_{min}}{pv_{max} - pv_{min}} &= \\
\frac{-pv_{max} + pv_{min} + 2pv - 2pv_{min}}{pv_{max} - pv_{min}} &= \\
\frac{-pv_{max} - pv_{min} + 2pv}{pv_{max} - pv_{min}} &= \\
pv \cdot \frac{2}{pv_{max} - pv_{min}} + \left(\frac{-pv_{max} - pv_{min}}{pv_{max} - pv_{min}}\right) & \qquad (10)
\end{aligned}
$$

The first term of equation (10) will be used as self-feedback whereas the second term acts as bias template. This leads to the following (11):

$$
\begin{aligned}
\mathbf{A} &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{2}{pv_{max} - pv_{min}} & 0 \\ 0 & 0 & 0 \end{pmatrix}, \\
z &= \frac{-pv_{max} - pv_{min}}{pv_{max} - pv_{min}} \qquad (11)
\end{aligned}
$$

Parameters such as the time step or iteration count are set to $\Delta h = 1.0$ and $t_{max} = 1$ respectively.

The next CNN templates are derived for the Gaussian kernel which used to blur the input image. The kernel is defined through equation (12)

$$
G(x,y;\sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \qquad (12)
$$

where $\sigma \neq 0$ denotes the standard deviation determining the width of the Gaussian kernel and the parameters x, y denote the pixel coordinates. The square of the standard deviation is known as variance σ^2. It is easy to implement the Gaussian kernel functionality by setting $\mathbf{A} = 0$ and $\mathbf{B} =$ "Gaussian kernel", and $z = 0$. To avoid loosing image information the templates are slightly modified compared to the mentioned convolution. Thus, a fixed input for each step is necessary. An example for a Gaussian kernel can be found in equation (12)

$$
\begin{aligned}
\mathbf{A} &= \frac{1}{K_a} \begin{pmatrix} 12 & 13 & 15 & 13 & 12 \\ 13 & 17 & 18 & 17 & 13 \\ 15 & 18 & 20 & 18 & 15 \\ 13 & 17 & 18 & 17 & 13 \\ 12 & 13 & 15 & 13 & 12 \end{pmatrix} \\
\mathbf{B} &= \frac{1}{K_b} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \\
z &= 0 \qquad (13)
\end{aligned}
$$

where K_a and K_b determine the width and height of the Gaussian filter [31]. Templates of the size of 5×5 are sufficient to blur the input image. Parameters such as the time step or iteration count are set to $\Delta h = 0.08$ and $t_{max} = 5$ respectively.

IV. Experimental performance analysis

In the following we present the results of each image contrast enhancement method, which we described above, and compare them by combining the Tenengrad method and Canny's Edge Detector. To evaluate the image quality after it has been enhanced by on the CNN and Coupled oscillator methods, we used the Tenengrad method (TEN) which is a well-known and traditional method for evaluating the quality (sharpness) of images [32]. Sharpness and edges of an image stand in direct correlation to each other which forms the basis for the Tenengrad method. The value yielded by this method quantifies the sharpness of an image. The main step to calculate the Tenengrad value involves the calculation of the gradient for each pixel. Sobel operator is a proper method to approximate the gradient $\nabla I(x,y)$ for each pixel by totaling all magnitudes greater than a threshold value. The Sobel operator, with the convolution kernels i_x and i_y is used as a high pass filter to provide approximation for the partial derivatives. The kernels are defined as follows [27], [3]:

$$
i_x = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}, \qquad i_y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \qquad (14)
$$

The kernel i_x is used for the horizontal and i_y is used for the vertical neighbors of a pixel (x,y). In equation (15). we describe the relationship between the Sobel operators and the gradient

$$
S(x,y) = \sqrt{(i_x * I(x,y))^2 + (i_y * I(x,y))^2} \qquad (15)
$$

The best focused point exhibits the greatest absolute gradient value $S(x,y) = |\nabla I(x,y)|$. In (16) the Tenengrad value is defined where X and Y are the dimensions of the image and T is a threshold value which is set in direct correlation with the amount of noise in the image.

$$
\mathrm{TEN} = \sum_{x=1}^{\mathrm{X}} \sum_{y=1}^{\mathrm{Y}} S^2(x,y) \text{ for } S(x,y) \geqslant \mathrm{T} \qquad (16)
$$

Instead of using common matrix multiplications, the multiplications in (15) and (16) are point-wise operations because they allow for each pixel to be analyzed separately. Note that pixel intensities in literature are in the interval of $[0,255]$. In our simulation, however, we have values in the interval of $[0,1]$.

This result strikes a cord with the visual evaluation done by the human eye. A further qualitative measurement which we want to use in this paper is the amount of extracted edges of an image. For this purpose we will use the Canny's Edge Detector [3]. Next, we want to quantify the appearance of edges by counting edge pixels. We denote this value as Sum of Value (SOV).

To combine both techniques we need a normalization step to achieve an image independent scale since both values are dependent to the dimensions of an image. In order to do that we have to normalize the Tenengrad value and SOV of a processed image in an interval between zero and one. Equation (17) shows the evaluation value for image quality comparison for our purpose which we denote as V.

$$V = 100 \cdot \mathrm{Norm(TEN)} \cdot \mathrm{NORM(SOV)} \tag{17}$$

In Fig. 6 we only present grayscale images because our nonlinear coupled oscillator implementation is not yet able to process color images (for now). The first image shows the original image which is the input for each algorithm. The original image has a Tenengrad value of 3683 and a SOV value of 2125. Our evaluation value which we described in (17) is 0.08. According to the results in Fig. 17 we put forth the following conclusion. The best result is yielded by method (m) which is proposed in this chapter. Adaptive histogram equalization with CNN (f) produces the worst result for our test-image. The maximum value of V is 100 and the minimum value is 0 – the greater the value, the better the image quality.

Due to this result we see that by using only the Tenengrad value to make a qualitatively decision between methods is insufficient. Therefore it is necessary to interpret the extracted edges with the Canny Edge Detector, too. These two measurement techniques together can provide a good qualitative benchmark, at least for our purposes. The results in Fig. 6 show, that our architecture behaves flexibly in changing conditions and promises adequate results. Further results of other methods can be found in [33].

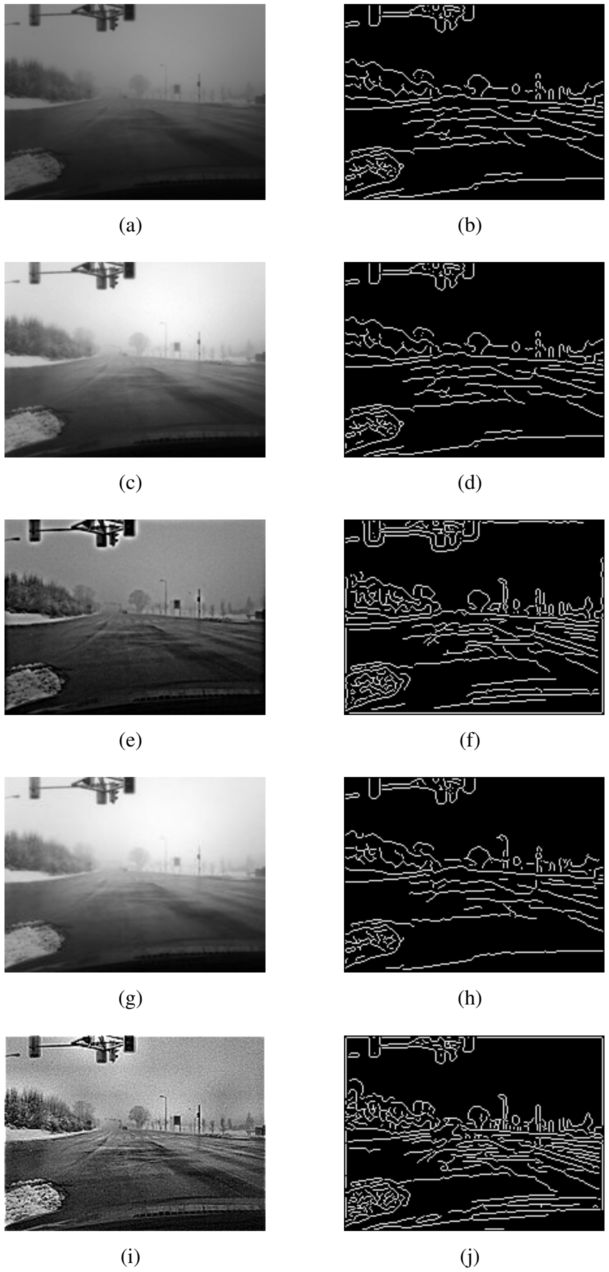
V. Conclusion and outlook

In this paper we presented a CNN-compatible method which does not adequately reflect the removal of fog but instead enhances images in real time. This method is based on two well known methods which are CLAHE and unsharp masking. These methods are partially mimicked with CNN to allow real-time processing. Our prototype implementation is designed in C++. But simulation of such a complex system demands a lot of resources in a computer. Due to this limitation we are going to implement this system on a Graphics Processing Unit (GPU). We will implement our system based on this architecture. This hardware emulation has the potential for working under real time conditions.

Further we showed, that the proposed method is able to enhance non uniformly degraded images and delivers better results comparing to state-of-the-art methods. Nevertheless, there is potential to enhance the method. Further investigation could lead to a CNN-based method which is able to realize the interpolation which is actually implemented without using the CNN paradigm.

Acknowledgment

This research is part of the project Adaptive - CV - Robust Adaptive Computer Vision. We acknowledge the financial support of the Government of Austria through the Oesterreichische Forschungsfoerderungsgesellschaft mbH (FFG).



(a) (b)

(c) (d)

(e) (f)

(g) (h)

(i) (j)

Fig. 6. Enhancement of fog degenerated image. The right hand side shows the extracted edges of the corresponding image on the left hand side: original image (200 × 150 pixel) (a) (TEN= 3683,SOV= 2125,V= 0.08), by using adaptive contrast stretching (CNN) (c) (TEN= 9875,SOV= 1894,V= 0.19), PDE based method [34] (CNN) with $\lambda = 1$ (e) (TEN= 10746,SOV= 3422,V= 0.38), nonlinear coupled oscillators (g) (TEN= 10087,SOV= 2066,V= 0.21), proposed method (k) (TEN= 25577,SOV= 3645,V= 0.96).

References

[1] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, 2nd ed. London, England, Great Britain: Chapman & Hall, 1998.

[2] S. G. Narasimhan and S. K. Nayar, "Contrast Restoration of Weather Degraded Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 1, pp. 713 – 724, Jun. 2003.

[3] T. Acharya and A. K. Ray, *Image Processing - Principles and Applications*. Wiley-Interscience, 2005.

[4] M. Alwani and A. K. Tiwari, “A Constrast Enhancement Based Algorithm To Improve Visibility Of Colored Foggy Images,” in *Proceedings of the 4th WSEAS International Converence on Business Administration (ICBA’10)*, 2010.

[5] D.-J. Wang, C.-H. Chen, T.-Y. Chen, and W.-W. Tsai, “Visibility Enhancement in the Foggy Environment Based on Color Analysis,” *Innovative Computing ,Information and Control, International Conference on*, vol. 0, pp. 342–345, 2009.

[6] Z. Jia, H. Wang, R. Caballero, Z. Xiong, J. Zhao, and A. Finn, “Real-time content adaptive contrast enhancement for see-through fog and rain,” in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, Mar. 2010, pp. 1378–1381.

[7] A. M. Reza, “Realization of the contrast limited adaptive histogram equalization (clahe) for real-time image enhancement,” *The Journal of VLSI Signal Processing*, vol. 38, pp. 35–44, 2004, 10.1023/B:VLSI.0000028532.53893.82. [Online]. Available: http://dx.doi.org/10.1023/B:VLSI.0000028532.53893.82

[8] K. Zuiderveld, *Contrast limited adaptive histogram equalization*. San Diego, CA, USA: Academic Press Professional, Inc., 1994, pp. 474–485. [Online]. Available: http://portal.acm.org/citation.cfm?id=180895.180940

[9] H. Wu and M. Yuen, “A generalized block-edge impairment metric for video coding,” *Signal Processing Letters, IEEE*, vol. 4, no. 11, pp. 317 –320, Nov. 1997.

[10] J. P. Oakley and H. Bu, “Correction of simple contrast loss in color images,” *IEEE Trans Image Processing*, vol. 16, no. 2, pp. 511–522, 2007. [Online]. Available: http://www.biomedsearch.com/nih/Correction-simple-contrast-loss-in/17269643.html

[11] J. Oakley and B. Satherley, “Improving image quality in poor visibility conditions using a physical model for contrast degradation,” *IEEE Transactions on Image Processing*, vol. 7, no. 2, pp. 167–179, Feb. 1998.

[12] Y. Schechner, S. Narasimhan, and S. Nayar, “Instant Dehazing of Images using Polarization,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. I, Dec. 2001, pp. 325–332.

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

[14] Y.-S. Zhai and X.-M. Liu, “An improved fog-degraded image enhancement algorithm,” in *International Conference on Wavelet Analysis and Pattern Recognition, 2007. ICWAPR ’07.*, vol. 2, Nov. 2007, pp. 522–526.

[15] Y.-S. Zhai and Y. Zhang, “Contrast Restoration for Fog-Degraded Images,” *Computational Intelligence and Security, International Conference on*, vol. 1, pp. 619–623, 2009.

[16] J. John and M. Wilscy, “Enhancement of weather degraded color images and video sequences using wavelet fusion,” in *World Congress on Engineering (Selected Papers)*, 2008, pp. 99–109.

[17] C. Xianqiao, W. Qing, Y. Xinping, and C. Xiuming, “The Enhancement for Foggy Traffic Image Based on EM Algorithm,” *Computational Intelligence and Natural Computing, International Conference on*, vol. 2, pp. 261–264, 2009.

[18] C. Xianqiao, Y. Xinping, and C. Xiumin, “Fast Algorithms for Foggy Image Enhancement Based on Convolution,” *Computational Intelligence and Design, International Symposium on*, vol. 2, pp. 165–168, 2008.

[19] K. Dongjun, J. Changwon, K. Bonghyup, and K. Hanseok, “Enhancement of Image Degraded by Fog Using Cost Function Based on Human Visual Model,” in *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Aug. 2008, pp. 64–67.

[20] Y. Yitzhaky, I. Dror, and N. S. Kopeika, “Restoration of atmospherically blurred images according to weather-predicted atmospheric modulation transfer function (mtf),” *Optical Engineering*, vol. 36, no. 11, pp. 3064–3072, Nov. 1997.

[21] N. Hautière, 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, Apr. 2006.

[22] 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*, Jun. 2007, pp. 87–92.

[23] J.-Y. Kim, L.-S. Kim, and S.-H. Hwang, “An Advanced Contrast Enhancement Using Partially Overlapped Sub-Block Histogram Equalization,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 11, no. 4, pp. 475–484, Apr. 2001.

[24] B. K. P. Horn, *Robot Vision*, 1st ed. McGraw-Hill Higher Education, 1986.

[25] G. X. Ritter and J. N. Wilson, *Handbook of Computer Vision Algorithms in Image Algebra*, 2nd ed. CRC Press, 2001.

[26] G. Cserey, C. Rekeczky, and P. Foldesy, “Pde based histogram modification with embedded morphological processing of the level-sets,” in *Cellular Neural Networks and Their Applications, 2002. (CNNA 2002). Proceedings of the 2002 7th IEEE International Workshop on*, Jul. 2002, pp. 315–322.

[27] K. D. Tönnies, *Grundlagen der Bildverarbeitung*. Munich, Germany: Pearson Studium, 2005.

[28] N. Kempe, “Analyse der Auswirkungen von Algorithmen zur Reduktion atmosphärischer Störungen am Beispiel von Fahrerassistenzsystemen,” Master’s thesis, Otto von Guericke Universität Magdeburg, May 2010.

[29] J. Zhao, Q. Ren, J. Wang, and H. Meng, “A New Approach for Image Restoration Based on CNN Processor,” in *Proceedings of the 4th international symposium on Neural Networks: Advances in Neural Networks, Part III*, ser. ISNN ’07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 821–827. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-72395-0_100

[30] M. Csapodi and T. Roska, “Adaptive Histogram Equalization with Cellular Neural Networks,” in *Fourth IEEE International Workshop on Cellular Neural Networks and their Applications*, 1996, pp. 81–86.

[31] S.-A. Chen, J.-F. Chung, S.-F. Liang, and C.-T. Lin, “Cellular neural network (CNN) circuit design for modeling of early-stage human visual system,” in *Biomedical Circuits and Systems, 2004 IEEE International Workshop on*, dec. 2004, pp. S1/4/INV – S1/49–12.

[32] H.-S. Le and H. Li, “Fused logarithmic transform for contrast enhancement,” *Electronics Letters*, vol. 44, no. 1, pp. 19–20, Mar. 2008.

[33] C. Schwarzlmüller, A. Fasih, M. A. Latif, and K. Kyamakya, “Adaptive Contrast Enhancement Involving CNN-based Processing for Foggy Weather,” *ISAST Transactions on Electronics and Signal Processing*, vol. 4, no. 1, pp. 24–35, Oct. 2010.

[34] A. Gacsádi, C. Grava, and A. Grava, “Medical image enhancement by using Cellular Neural Networks,” *Computers in Cardiology*, vol. 32, pp. 821–824, 2005.