

This thesis was submitted to the Institute of Mechanism Theory, Machine Dynamics and Robotics

Model to Enhance the Vision on Gauge Readers in Smoke Scenes

Master thesis

by:

Akshay Narendra SHEWATKAR B.Sc.

Matriculation number: 405 042

supervised by:

Dr. Oscar H. Ramírez-Agudelo, DLR

Nils Mandischer M.Sc., RWTH

Examiner:

Univ.-Prof. Dr.-Ing. Dr. h. c. Burkhard Corves

Prof. Dr.-Ing. Mathias Hüsing

Aachen, 5 April 2023

Master thesis

by Akshay Narendra SHEWATKAR B.Sc.

Matriculation number: 405 042

Model to Enhance the Vision on Gauge Readers in Smoke Scenes

The topic of this thesis is in the field of image de-smoking with the broader goal of improving vision in smoke filled buildings and to improve the safety and security of critical infrastructures such that the situational awareness is granted. The main objective is to de-smoke images of gauges obtained in smoky conditions, so as to get enhanced and more accurate pictures of the gauges in smoky environments. Images which are acquired in poor weather conditions such as fog, smoke, haze, mist, etc. are often severely degraded. The presence of such particles in air leads to reduced visibility, contrast, and colour degradation. Such acquired images are not suitable for many computer vision applications or for post processing operations. The post operations could be image segmentation, object detection, video surveillance, etc. Therefore, image de-smoking is required to mitigate these unwanted effects to ensure the accurate working of these computer vision applications. Smoke removal has been a challenging problem which refers to the process of retrieving clear images from the smoky ones. In addition, existing de-smoking methods have limitations due to hand-crafted features, homogeneous medium based model, and insufficient datasets. The thesis is divided into sub-sections considering the timeline of six months. First, both types of haze removal, namely prior based methods and learning based methods are assessed. The prior based methods depend on a physical scattering model whereas the learning based approaches are data driven and use deep learning networks to estimate the hazy to clear image translation. The state of the art algorithms are implemented and tested with benchmark dataset. Recent networks such as the Feature Fusion Attention network (FFA-Net) and the Autoencoder and Contrastive Regularization network (AE CR Net) will be used. Meanwhile, challenges to train a model on a custom dataset using a Nvidia Quadro RTX 8000 GPU are identified. Afterwards, datasets involving smoky and clear images of gauges and preprocessing of the data is acquired. Finally, the models are implemented and evaluated with the custom datasets using the SSIM (Structural similarity index) and PSNR (Peak signal to noise ratio) metrics which are widely used in image processing. Concluding, the findings and models are documented.

Supervisor: Dr. Oscar H. Ramírez-Agudelo, DLR

Nils Mandischer M.Sc., RWTH

Eidesstattliche Versicherung

Akshay Narendra SHEWATKAR

Matrikel-Nummer: 405 042

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Master thesis mit dem Titel

Model to Enhance the Vision on Gauge Readers in Smoke Scenes

selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Aachen, 5 April 2023

Akshay Narendra SHEWATKAR

Belehrung:

§ 156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Strafflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

Aachen, 5 April 2023

Akshay Narendra SHEWATKAR

The present translation is for your convenience only.
Only the German version is legally binding.

Statutory Declaration in Lieu of an Oath

Akshay Narendra SHEWATKAR

Matriculation number: 405 042

I hereby declare in lieu of an oath that I have completed the present Master thesis entitled

Model to Enhance the Vision on Gauge Readers in Smoke Scenes

independently and without illegitimate assistance from third parties. I have used no other than the specified sources and aids. In case that the thesis is additionally submitted in an electronic format, I declare that the written and electronic versions are fully identical. The thesis has not been submitted to any examination body in this, or similar, form.

Aachen, 5 April 2023

Akshay Narendra SHEWATKAR

Official Notification:

Para. 156 StGB (German Criminal Code): False Statutory Declarations

Whosoever before a public authority competent to administer statutory declarations falsely makes such a declaration or falsely testifies while referring to such a declaration shall be liable to imprisonment not exceeding three years or a fine.

Para. 161 StGB (German Criminal Code): False Statutory Declarations Due to Negligence

(1) If a person commits one of the offences listed in sections 154 to 156 negligently the penalty shall be imprisonment not exceeding one year or a fine.

(2) The offender shall be exempt from liability if he or she corrects their false testimony in time. The provisions of section 158 (2) and (3) shall apply accordingly. I have read and understood the above official notification: :

Aachen, 5 April 2023

Akshay Narendra SHEWATKAR

Contents

List of abbreviations	xi
1 Introduction	1
2 Literature Review	3
2.1 Prior Based Methods	4
2.2 Deep Learning based Methods	9
2.2.1 FFA Net	9
2.2.2 AEQR Net	14
2.3 Evaluation Metrics	18
2.4 Datasets used for Image desmoking/dehazing	20
3 Image Processing Basics	25
3.1 Image Processing	25
3.2 Computer Vision	25
3.3 Image Processing Techniques	28
4 Artificial Intelligence	33
4.1 Deep Learning	33
4.2 Convolutional Neural Networks(CNNs)	34
4.3 Residual Connections	36
4.4 Attention Mechanism	36
4.5 Training	37
5 Methodology	41
5.1 Dataset Acquisition	41
5.1.1 Datasets Available	41
5.1.2 Requirements for Custom Dataset	41
5.1.3 Synthetic Dataset	43
5.1.4 Summary	45
5.2 Implementation of Methods	46
5.2.1 BCCR	46
5.2.2 FFA Net and AEQR Net	48
5.2.3 Training and Testing	49
6 Results and Comparison	51
6.1 Realistic Dataset	51

6.2	Synthetic Haze Dataset	52
6.3	Synthetic Smoke Dataset	54
6.4	Factors Comparison	56
7	Discussion and Outlook	59
7.1	Comparison of the Methods	59
7.2	Autonomous Reading of Gauges	61
8	Summary and Future Work	63
	Bibliography	I
	List of Tables	VII
	List of Figures	IX

List of abbreviations

General abbreviations

AECR	Auto Encoder Contrastive Learning
FFT	Fast Fourier Transformation
BCCR	efficient image dehazing with Boundary Constraint and Contextual Regularization
FFA	Feaure Fusion Attention
FA	Feature Attention
CA	Channel Attention
MSE	Mean Squared Error
L1	L1 loss function
L2	L2 loss function
GAN	Generative Adversarial Network
MRF	Markov Random Field
LPIPS	Learned Perceptual Image Patch Similarity
Net	Network
PSNR	Peak Signal to Noise Ratio
SSID	Semi-Supervised Image Dehazing
SSIM	Structural Similarity Index
CNN	Convolutional Neural Network
ZID	Zero Shot Image Dehazing
SOTA	State Of The Art
TV	Total Variation
DFE	Dynamic Feature Enhancement

VGG	Visual Geometry Group
RGB	Red Green Blue
DL	Deep Learning
ML	Machine Learning
AI	Artificial Intelligence
UE	Unreal Engine

1 Introduction

This thesis' main goal is to create trained models to enhance gauge images which are corrupted. Many factors, such as fire, explosions, and other emergencies that could happen in buildings or tight spaces, might result in the presence of smoke or haze in critical scenarios. Accurate reading and interpretation of gauge data is even more important in these circumstances because it can give first responders the knowledge they need to stop further damage or harm from occurring. However, in many critical situations, smoke or haze obstructs the gauge images, making it challenging to read and understand the data from the gauges. This limits the possible uses of post-processing techniques like object detection and tracking and also presents a considerable barrier to gauge readers.

Thus, in critical situations where prompt and precise interpretation of gauge data can significantly affect averting or controlling the critical situation, the necessity for clear and enhanced images of gauges is vital. The thesis aims to achieve this through a model which seeks to generate clear and enhanced gauge images that may be easily interpreted and used in a variety of post-processing applications by removing smoke and haze from corrupted gauge images. This thesis aims to implement prior-based model (Efficient image dehazing with Boundary Constraint and Contextual Regularization) BCCR and train deep learning models, (Feature Fusion Attention Network) FFA Net and (Auto Encoder based Contrastive Regularization) AE CR Net on a custom dataset of gauge images with the objective of producing enhanced and clear gauge images from input images that are corrupted with light to dense smoke.

In the first part of this thesis, the relevant literature review, image processing techniques, and machine learning basics are presented in Chapters 2, 3, and 4. The new contribution to this field is presented in the second part of the thesis; which is the methodology of this work, the results and discussion, and are presented in Chapters 5, 6, and 7 respectively. Finally, Chapter 8 summarizes the findings and provides a future perspective on this work.

2 Literature Review

This chapter provides an overview of the current state of research in the field of image desmoking and image dehazing. In addition, the evaluation metrics and the datasets generally used for image dehazing are discussed.

Both atmospheric phenomena, smoke and haze can degrade image quality and negatively impact the performance of image processing systems. Similar to rain and snow, smoke and haze reduction is critical for various applications, including surveillance and relief systems. Smoke, composed of dust and particles, is an atmospheric phenomenon that can significantly affect image processing and decrease the clarity of photographic images [HST09]. Similar to other atmospheric phenomena such as rain, haze, and snow, smoke removal is important in surveillance and relief systems. While image dehazing algorithms aim to remove atmospheric haze, image desmoking algorithms specifically target the removal of smoke from images. There are several algorithms proposed for smoke removal in surgical images [VVL20], [PBV22], [CTJ18], such as leveraging the dark channel prior or convolutional neural networks. However, these methods were designed for surgical scenarios and may not perform well in real-world scenarios. It is important to note that smoke and haze differ, as haze has homogeneous particle density and uniform particle distribution, whereas the concentration of smoke is generally inconsistent and varies in a scene [PCC19].

Image desmoking and image dehazing are important tasks in computer vision that have been studied extensively in recent years [PCC19], [AJ22], [GCC21]. Two major categories of methods have been proposed to address these tasks: prior based methods and deep learning based methods. Prior based methods rely on handcrafted priors to model the characteristics of haze/smoke in images and estimate the transmission map or smoke density map. These priors include dark channel prior, color attenuation prior, and atmospheric scattering model [AJ22]. Prior based methods are computationally efficient and require minimal training data. However, they may not be as effective in dealing with complex or diverse scenes due to the limitations of handcrafted priors [GCC21].

On the other hand, deep learning-based methods use convolutional neural networks (CNNs) to learn the mapping between hazy/smoky and clear images directly from data. These methods require large amounts of training data and may be computationally expensive. However, they can achieve state-of-the-art performance on a wide range of scenes and conditions. In the following sections, the most popular methods for prior based methods

and deep learning based methods will be reviewed and explained.

2.1 Prior Based Methods

In this section, four main approaches of prior based methods are presented.

Single Image Dehazing

Fattal [Fat08] proposes a new approach for removing haze from single input images by estimating the optical transmission. The method accounts for surface shading and breaks the image into regions of constant albedo to resolve ambiguity in the data. It estimates the color of the haze and can be used for image refocusing and novel view synthesis. The image is factored into two components, the airlight contribution and the unknown surface radiance, which are combined by the transmission coefficient. Recovering a haze-free image requires determining the three surface color values and transmission value at every pixel. This method is passive, requires only a single image, and can handle discontinuities in the scene depth or medium thickness. It achieves a significant reduction of the airlight and restores contrasts in complex scenes, making it useful for other applications. Fig. 2.1 shows the output obtained using [Fat08] method.



Figure 2.1: **Left:** input image, **middle:** output image obtained, **right:** depth image obtained by using Fattal's approach [Fat08].

Visibility in Bad Weather from a Single Image

Several methods have been proposed to model the absorption and scattering processes, but most of them require multiple input images, which is often difficult to fulfill. To address this problem [Tan08] proposes a new automated method that only requires a single input image. This method is based on two observations: images with enhanced visibility have more contrast than images with bad weather, and airlight tends to be

smooth. A cost function is developed using Markov Random Fields (MRF), which can be optimized efficiently by various techniques. This proposed method does not require geometrical information or user interactions and is applicable for both color and gray images. This method estimates the atmospheric light, removes the light color of the input image, computes data and smoothness costs for every pixel, builds MRFs, and optimizes them using existing inference methods to produce the estimated values of the airlight. Finally, the direct attenuation is computed based on the estimated airlight to enhance the contrast of the input image and improve its visibility. The goal of this approach is solely to enhance the contrast of an input image and not to fully recover the scene's original colors or albedo. Fig. 2.3 shows the direct attenuation and the airlight estimations obtained for two input images using Tan et al method.



Figure 2.2: Top row: input images, Middle row: direct attenuation, Bottom row: airlight estimation. [Tan08].

Single Image Haze Removal using Dark Channel Prior

The dark channel prior was proposed by He et al. [HST09] for single image haze removal, based on statistics of outdoor haze-free images. It was observed that the dark pixels in local regions had low intensity in at least one RGB channel, which directly provided an accurate estimation of haze transmission. Combining a haze imaging model and soft

matting interpolation, a high-quality haze-free image and depth map is obtained. The approach is physically valid and works well for most outdoor hazy images, with few halo artifacts. However, it has limitations with scene objects similar to airlight.



Figure 2.3: Haze removal using single image. **Left:** input image, **middle:** output image, **right:** estimated depth map [HST09].

Efficient Image Dehazing with Boundary Constraint and Contextual Regularization

Meng et al. propose a method, Efficient image dehazing with Boundary Constraint and Contextual Regularization (BCCR). The method proposes three novel elements, namely, a constraint on scene transmission, contextual regularization and an efficient optimization scheme. To explain the formulation of an image affected by haze or smoke, the following equation is widely used [NN03], [Fat08], [HST09]:

$$\mathbf{I}(x) = t(x)\mathbf{J}(x) + (1 - t(x))\mathbf{A} \quad (2.1)$$

$$t(x) = \exp[-\beta d(x)] \quad (2.2)$$

Description of Eqns. where,

- x – pixel of the two dimensional image
- $\mathbf{I}(x)$ – observed image
- $\mathbf{J}(x)$ – scene radiance
- $\mathbf{A}(x)$ – atmospheric light
- $t(x)$ – scene transmission (*valid under the assumption that haze is homogeneous*)

- β – medium extinction or scattering coefficient of atmosphere
- $d(x)$ – scene depth

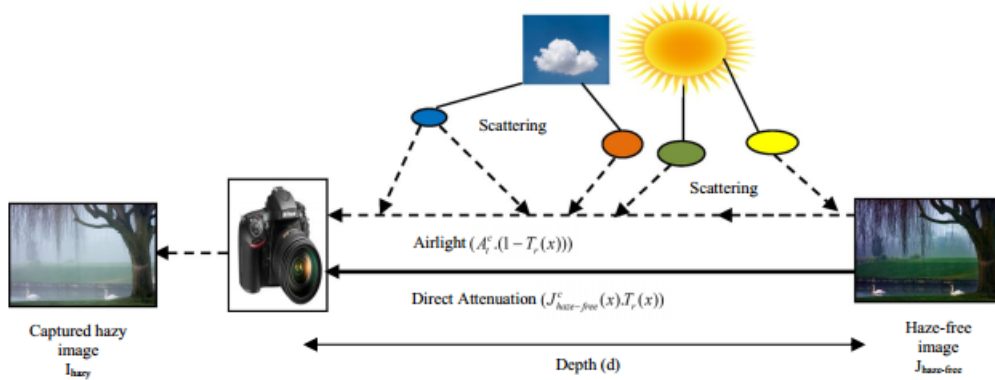


Figure 2.4: Physical model of hazy image formation [AJ22].

In Fig. 6.1 it is shown how various atmospheric phenomena affect the quality of a captured image. When incident light is reflected from the object, the reflected light is attenuated due to the distance between the camera and the object. Also, due to scattering of particles, A (Airlight) is also introduced into the camera. Finally, an image contaminated with haze or smoke is composed of two terms: direct attenuation and airlight.

The goal is to recover the scene radiance $\mathbf{J}(x)$ from $\mathbf{I}(x)$ with Eq. 2.2. For calculating $\mathbf{J}(x)$, estimation of $t(x)$ and \mathbf{A} is necessary. However, these two terms are unknowns, which makes the equation under constrained. Therefore, prior assumptions and some constraints need to be introduced in order to estimate the $\mathbf{J}(x)$. Eq. 2.3 is derived from 2.1 and shows the estimation of $\mathbf{J}(x)$.

$$\mathbf{J}(x) = \frac{\mathbf{I}(x) - \mathbf{A}}{[\max(t(x), \epsilon)]^\delta} + \mathbf{A} \quad (2.3)$$

To solve this under constrained problem of estimating $J(x)$, a boundary constraint from the radiance cube was introduced by Meng et al. [MWD13]. As shown in Fig. 2.5, the value of a hazy pixel $\mathbf{I}(x)$ is the linear interpolation between the clear pixel $\mathbf{J}(x)$ and the atmospheric light \mathbf{A} . The constraint is based on the observation that the scene radiance of an image is bounded. For the bounding, two constant vectors C_0 and C_1 are considered, where the parameters values would depend on the image that is to be cleared. The method proposed in [MWD13] is to push an image patch until it touches the boundary of the radiance cube and use the resulting estimate of the transmission map $t(x)$. This process is equivalent to using the dark channel prior, which assumes that the minimum

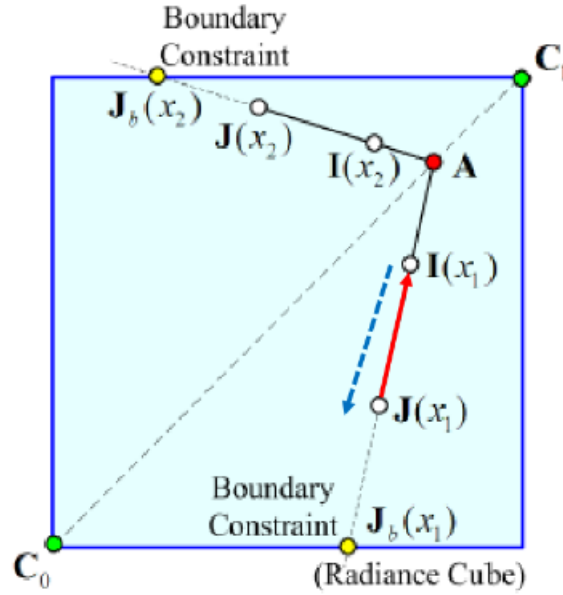


Figure 2.5: Radiance cube and boundary constraint [MWD13].

value in a local patch of a haze/smoke free image corresponds to the transmission map of the corresponding patch in the haze/smoke image.

The Dark Channel Prior

The Dark Channel Prior (DCP) equation is given by:

$$\mathbf{J}^{\text{dark}}(\mathbf{x}) = \min_{\mathbf{y} \in \Omega(\mathbf{x})} [\min_{\mathbf{c} \in \{r, g, b\}} \mathbf{J}^{\mathbf{c}}(\mathbf{y})] \quad (2.4)$$

The DCP assumes that within an image patch, at least one colored channel has very low intensity or a zero value of intensity. In Eq. 2.4, $\Omega(x)$ is a local image patch that is centred at x . $\mathbf{J}^{\mathbf{c}}$ is an intensity of a color channel among RGB channels. According to the equation, the minimum value among the three color channels within all pixels of the image patch region $\Omega(x)$ is chosen as the dark channel \mathbf{J}^{dark} .

The low intensities in the dark channel are due to shadows, colorful objects or surfaces or dark objects or surfaces. The approximation of $\mathbf{J}^{\text{dark}} \approx 0$ for the pixel value of the dark channel is called the DCP.

Compared to other approaches, such as Fattal's refined image formation model [Fat08], Tan's local contrast maximization method [Tan08], the BCCR method requires fewer general assumptions and achieves higher-quality results [MWD13]. Additionally, it benefits from incorporating a filter bank to attenuate image noise and enhance interesting image structures, such as jump edges and corners.

2.2 Deep Learning based Methods

This section provides a brief overview of the deep learning methods used for image dehazing/desmoking and their categorization into supervised, unsupervised, and semi-supervised methods. Each category includes several popular methods.

Supervised methods rely on a dataset of paired hazy and haze-free images to learn a mapping between them. One popular approach are the deep convolutional neural network (CNN) based methods, such as FFA [QWB19], AOD [LPW17], GridDehazeNet [LMS19] and AECR [WQL21]. These models learn an end-to-end mapping from hazy to haze-free images, using a loss function that measures the difference between the predicted and ground truth images.

Unsupervised methods do not require paired data and instead aim to learn from the image statistics. CycleGAN [EGE18] and ZID [LGL20] are examples of unsupervised methods used for dehazing/desmoking. Semi-supervised methods, as the name suggests, lie in between the supervised and unsupervised methods. These methods use a combination of paired and unpaired data to learn the mapping between hazy and haze-free images. One popular method is the SSID [LDR20].

Overall, the choice of deep learning method for image dehazing/desmoking depends on the availability of labeled data and the specific task at hand.

GridDehazeNet

GridDehazeNet [LMS19] is an end-to-end trainable CNN for single image dehazing that consists of three modules: pre-processing, backbone, and post-processing. The trainable pre-processing module generates diverse and pertinent features, unlike hand-selected methods. The backbone module implements attention-based multi-scale estimation on a grid network, which avoids bottleneck issues and employs a channel-wise attention mechanism. The post-processing module helps reduce artifacts. The proposed method does not rely on the atmosphere scattering model, and the network’s loss surface remains well-behaved without introducing heterogeneous components.

2.2.1 FFA Net

FFA (Feature Fusion Attention) network is a State Of The Art (SOTA) for image dehazing based on CNNs. It is an 'end-to-end' Convolutional Neural Network based architecture. This model does not directly attempt to estimate the scene radiance (see Eq. 2.3), instead

it learns the mapping from a hazy image to a clear image. The 'end-to-end' stands for supervised algorithm, that takes a single image, corrupted by haze and outputs an image which is enhanced by reducing the quantities of haze.

The feature fusion network (FFA Net) is made up of three components [QWB19]:

1. A novel **Feature Attention (FA)** module consisting of **Channel Attention (CA)** and **Pixel Attention (PA)** layers (see Figs. 2.6 and 2.7).
2. Basic blocks, where each basic block consists of *FA* Module and **local residual learning** (see Fig. 2.8).
3. An **Attention based** different levels feature fusion structure (see Fig. 2.9).

Feature Attention

The feature attention module used in FFA Net is mainly inspired by Zhang et al. [ZLL18]. Since the attention mechanism [VSP17] has been widely used in design of neural networks, it has played vital role in enhancing the performance of networks. The basic idea behind an attention mechanism is to selectively focus on the most relevant parts of an input data. In the context of neural networks, the attention mechanism involves computation of attention weights that indicate how important each element of the input should be for the current task. These weights can be learned from the data using various techniques such as softmax normalization, dot product, etc.

Most dehazing networks treat the channel wise and pixel wise features equally. Due to this equal treatment, hazy/smoky images which have *uneven haze distribution* and *weighted channel wise features* cannot be handled properly. The introduction of channel attention and pixel attention within *FA* Module of FFA Net addresses these problems by providing additional flexibility. In the FFA Net, the idea is to treat the different features in channels and pixels of an image, differently. The *CA* mainly relies on the fact that different channel features have different weighted information. The channel wise global spatial information is taken into a channel descriptor using global average pooling.

In Fig. 2.7, $X_c(i, j)$ is the value of the pixel at location (i, j) of the c^{th} color channel and H_p acts as the pooling function. To obtain the weights of the different channels, the output is passed through two convolutional layers, a ReLu and a sigmoid activation function. σ and δ are the sigmoid and ReLu activation functions, respectively. The final step is to do a element wise multiplication of the input F_c and the weights of the channel which is CA_c . It is considered that the haze has uneven distribution on different image pixels, and a single transmission map would not handle this information. Therefore, a

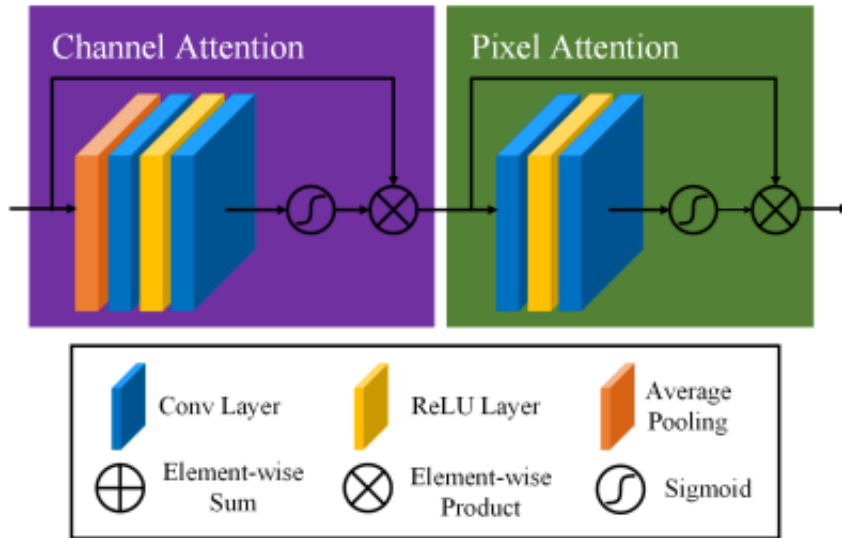


Figure 2.6: Feature Attention Module [QWB19].

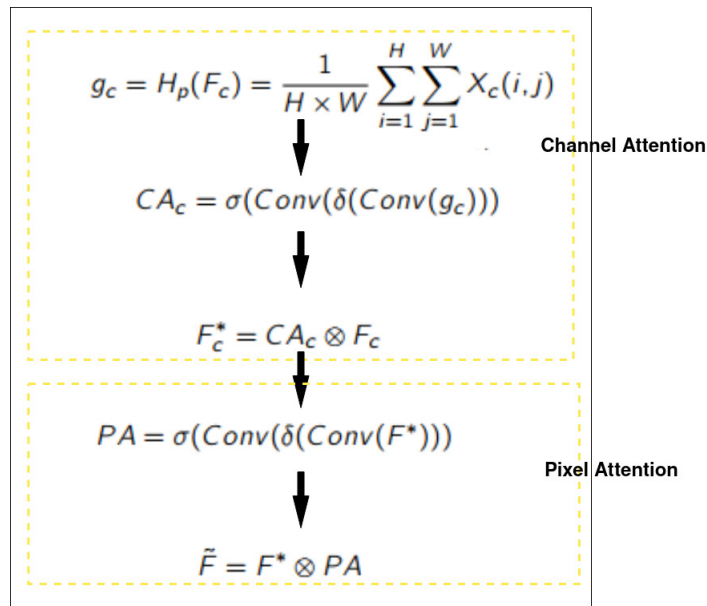


Figure 2.7: Equations of FA module [QWB19]).

pixel attention module PA is needed to pay more attention to areas with thick hazed pixels and high-frequency image region. Like the CA Layer, the input F_c^* is fed into two convolutional layers, with ReLU and sigmoid activation functions. Then, an element wise multiplication is performed between the input and the output.

Basic Block Structure

The basic block (as shown in Fig. 2.8) consists of the feature attention module (CA and PA layers) and local residual learning. The residual learning allows less important

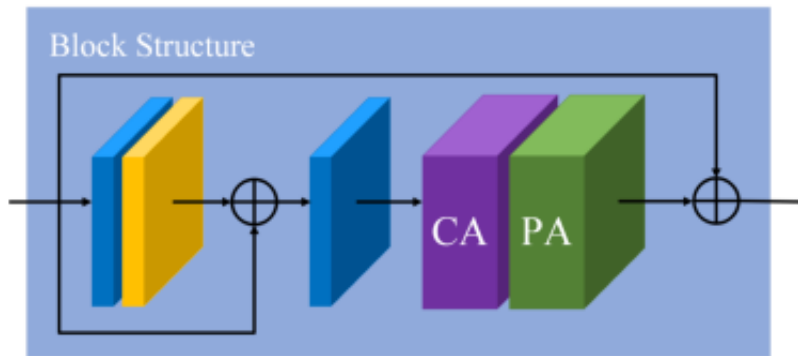


Figure 2.8: Basic block for FFA Net (Fig. 6 from [QWB19]).

features like thin haze regions to be bypassed to the final layers so that the main network focuses more on thick hazy regions. A group architecture is also present which combines B basic block structures with skip connections. These grouped blocks increase the depth of the network which enables it to capture information in depth. Also, the skip connections make the training process of the network faster.

Feature Fusion Attention

After passing through the G group architecture (as shown in Fig. 2.9) the feature maps are concatenated. The features are then fused by multiplying the adaptive learning weights obtained by the Feature Attention (CA and PA modules) mechanism.

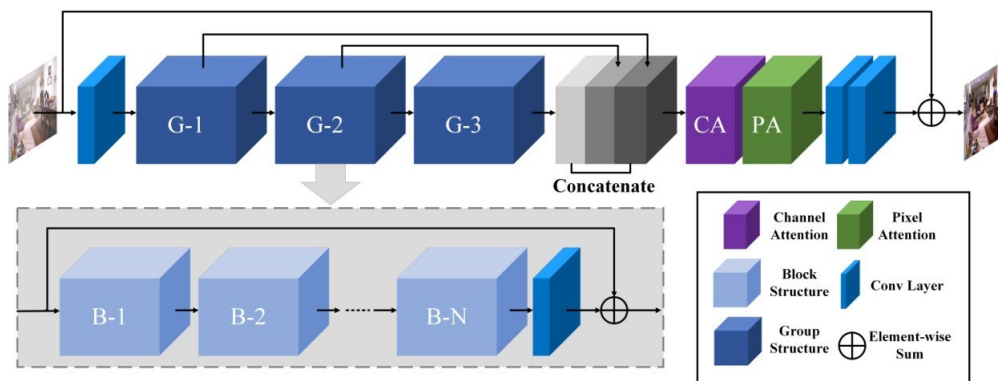


Figure 2.9: FFA Architecture [QWB19].

Loss Function

Generally, L2 loss, i.e. is the mean squared error, has been the widely used in image restoration, as well as other task such as regression. However, it has been shown in

[LSK17] and [ZGF17] that L1 loss gives better results than L2. This is because L2 Loss does not correlate well with human’s perception of image quality and is unable to capture the intricate characteristics of the human visual system (HVS) [ZGF17].

Therefore, in FFA Network, a L1 Loss is used. Here, the L1 Loss is optimised with:

$$L(\Theta) = \frac{1}{N} \sum_{i=1}^N \| I_{gt}^i - FFA(I_{haze}^i) \| \quad (2.5)$$

Here, Θ denotes the parameters of the FFA Net, which are to be optimized during the training process. These include the weights and the biases of the entire FFA architecture. I_{gt} is the ground truth (i.e. the clear images) and I_{haze} stands for the input (i.e. the hazy images).

CycleGAN

Engin, Genc, and Ekenel [EGE18] introduce a novel end-to-end network named Cycle-Dehaze, which does not require paired hazy and ground truth images for training. The proposed method is based on CycleGAN architecture and uses an unpaired training approach, where both clean and hazy images are fed into the network without any need for atmospheric scattering model parameters. The authors enhance CycleGAN formulation by incorporating cyclic perceptual-consistency loss in addition to cycle-consistency loss. The proposed approach (Fig. 2.10) enhances textural information recovery and produces visually better haze-free images.



Figure 2.10: Hazy and clean images [EGE18].

2.2.2 AE CR Net

The AE CR Net [WQL21] is a new method for image dehazing that utilizes autoencoders and contrastive regularization techniques. It is based on the FFA Net but is more compact and efficient due to its reduced network depth and additional modules. The contrastive regularization scheme provides a new perspective on dehazing images, allowing the network to better understand the relationship between hazy and clear images and produce more accurate results. The AE CR Net appears to be a promising approach for image dehazing that offers improved performance compared to existing methods.

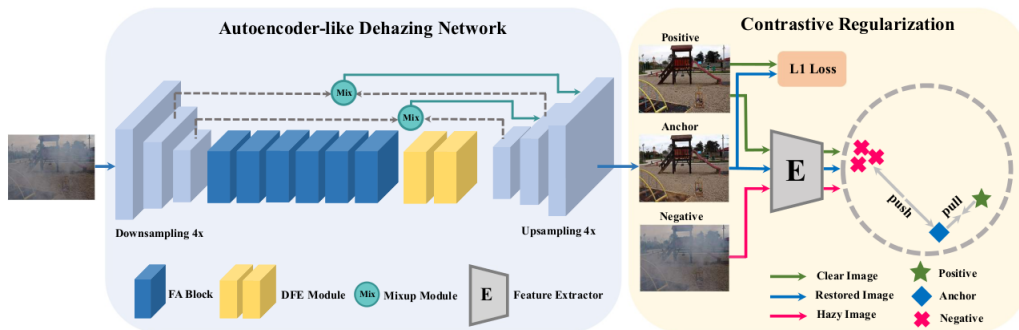


Figure 2.11: Architecture of AE CR Net [WQL21].

Autoencoder-like Dehazing Network

The AE CR Net [WQL21] is an autoencoder network that uses *FA* (Feature Attention) blocks (as shown in Fig. 2.6), which contain both channel attention and pixel attention layers, to learn the feature representation of images in a lower resolution space. The network consists of four downsampling operations, which reduce the resolution of the input image. These operations are followed by six dense *FA* blocks that learn the most significant features of the image in this lower resolution space.

The channel attention layers of the *FA* blocks allow the network to focus on the most relevant channels, while the pixel attention layers help the network to attend to the most relevant features in each pixel of the image. This attention mechanism improves the network's ability to capture the most salient features of the image and thus enhances its performance. After the feature representation has been learned, the network performs four upsampling operations to restore the image to its original size. Using the output of the final upsampling operation, a convolution operation is performed to obtain the final restored image.

Apart from the Autoencoder like network, there are two additional components added to improve the performance.

Adaptive Mixup & DFE

For better information flow between layers and fusion of spatial structured information, adaptive mixup and dynamic feature enhancement modules are used. For this, the AECR Net employs adaptive mixup (Fig. 2.12) and dynamic feature enhancement (DFE; Fig. 2.13) modules. The initial layers of the network capture the shallow features of the input image such as edges, intensity, and contours. However, as the layers progress, these features may be lost, which is addressed by using skip connections. In addition to the skip connections already used in the *FA* blocks, adaptive mixup operations are used within these skip connections, allowing the network to preserve the shallow information.

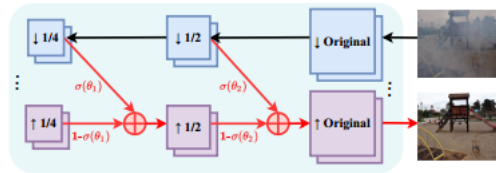


Figure 2.12: Adaptive Mixup Operation [WQL21].

Furthermore, the network employs two upsampling and downsampling layers, and a dilated convolutional layer is used to expand the receptive field in a flexible way. However, the deformable convolutional layer is introduced as an alternative to the rigid and fixed kernel. The deformable convolutional layer is more dynamic and flexible, which makes it better at capturing important information. The DFE module, which uses deformable convolutional layers, is introduced to enhance the model's transformation capability for better image restoration. By learning a dynamic weight map, the DFE module enhances the spatial structure of the feature maps, allowing the network to pay more attention to important regions while suppressing less important ones.



Figure 2.13: difference between rigid and deformable Kernels [WQL21].

Overall, the adaptive mixup and DFE modules, combined with skip connections and dilated convolutional layers, enable the AECR Net to produce high-quality restored images with preserved shallow information and enhanced spatial structure.

$$f_{\uparrow 2} = \text{Mix}(f_{\downarrow 1}, f_{\uparrow 1}) = \sigma(\theta_1) * f_{\downarrow 1} + (1 - \sigma(\theta_1)) * f_{\uparrow 1} \quad (2.6)$$

$$f_{\uparrow} = \text{Mix}(f_{\downarrow 2}, f_{\uparrow 2}) = \sigma(\theta_2) * f_{\downarrow 2} + (1 - \sigma(\theta_2)) * f_{\uparrow 2} \quad (2.7)$$

The Eqns. 2.6 and 2.7 show the adaptive mixup operations(see Fig. 2.12) used in the AE CR Net. $f_{\downarrow, i}$ and $f_{\uparrow, i}$ are the feature maps from the i^{th} downsampling and upsampling operations respectively. $\sigma(\theta_i)$ is the learnable factor which can be learned during training.

Contrastive Regularization

In the field of self-supervised representation learning for high-level vision tasks, contrastive learning is a commonly used technique. It involves training a model to pull an anchor point close to positive points and push it away from negative points in the representation space. This technique has been successfully applied in various vision tasks in previous works. However, there are few studies that have explored the application of contrastive learning in image dehazing, mainly due to the challenge of constructing contrastive samples and loss functions specific to this task.

To address this challenge, Wu et al. [WQL21] proposes a novel sampling method and a pixel-wise contrastive regularization for image dehazing. Unlike previous works, the approach uses a different sampling method and a novel loss function to improve the quality of image dehazing. The representation space for the specific task of removing smoke from gauge images, for contrastive learning would be:

- Clear gauge images as *Positive images*
- Input gauge images (corrupted with smoke/haze) as *Negative images*
- The output image of the AE CR Net model as *Anchor images*.

The positive set contains the input images and their corresponding ground truth images. The negative set contains input images that are corrupted by smoke or haze. Finally, the restored set contains the predicted images produced by the AE CR Net. These sets are then fed into the Visual Geometry Group 19 (VGG19) model, which is a widely used pre-trained convolutional neural network for image classification. The output of the VGG19 model is a high-dimensional latent feature space, which contains semantically meaningful representations of the images.

Within this latent feature space, the restored images are pulled towards the positive images and pushed away from the negative images (see Fig. 2.14). This is achieved by computing a loss function that encourages the distance between the restored images and positive images to be small, while simultaneously encouraging the distance between the restored images and negative images to be large. By doing so, the AECR Net is able to generate restored images that are semantically similar to the input images, leading to better image restoration results.

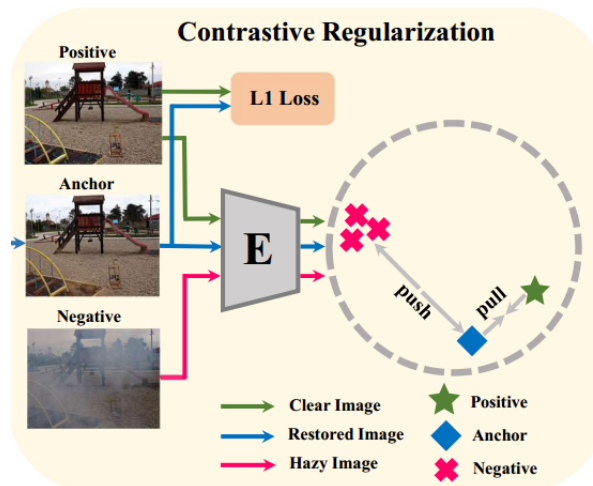


Figure 2.14: Contrastive Regularization: Anchor image pulled towards the Positive image and pushed away from the Negative image [WQL21].

Loss function

The final dehazing loss function looks like:

$$L1\|p - a\| + \beta \sum_{i=1}^n \frac{L1^*(p, a)}{L1^*(p, n)} \quad (2.8)$$

The first term in the loss function is a L1 loss between clear and restored image. The second term is the contrastive regularization term, where $L1^*$ stands for the distance between p (i.e. positive or ground truth) and a (i.e. anchor). $L1^*$ distance loss function, measures the absolute differences between the features. To extract the features, the first, third, fifth, ninth, and thirteenth layers of the VGG-19 model are used. β is the penalty parameter which is empirically set to 0.1.

2.3 Evaluation Metrics

Evaluation metrics are essential for assessing the effectiveness of dehazing and desmoking algorithms. Typically, two categories of quantitative metrics are used: full reference metrics and no-reference metrics. The full reference metrics require a ground truth image for comparison, while the no-reference metrics do not. The dehazing or desmoking process may result in various issues, such as residual smoke or haze, color distortions, halo artifacts, and edge preservation, among others. Therefore, numerous quality assessment methods have been introduced in the literature to measure these distortions.

Although both full reference and no-reference metrics are available, this thesis focuses only on full reference metrics since ground truth images are available for comparison (details about the dataset are presented in chapter 5). Furthermore, evaluating dehazing and desmoking algorithms using full reference metrics can provide better insights into the capabilities and limitations of these algorithms. Therefore, several well-known full reference metrics, such as Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM), will be used to evaluate the performance of the proposed algorithms in this thesis.

Full Reference Metrics

Full-reference metrics are used to evaluate a method when a ground truth (GT) image is available, which is particularly applicable to test the performance of synthetic images. Various metrics, such as PSNR, SSIM, Learned Perceptual Image Patch Similarity (LPIPS), Color Difference Equation (CIEDE2000), and Spatial High-frequency Reconstruction Quality (SHRQ) have been utilized in many recent works for evaluating the performance of image dehazing and desmoking algorithms.

Learned Perceptual Image Patch Similarity Metric

Pixel-wise metrics like PSNR and SSIM do not always agree with human judgment in assessing the perceptual quality of dehazed images. To address this, [LSZ17] introduced the Learned Perceptual Image Patch Similarity metric (LPIPS). It utilizes deep features trained on different deep learning frameworks to establish the perceptual similarity between two images. LPIPS can identify various distortions in the image, including photometric, noise, blur, and compression. The metric is calculated by a network F , which extracts features from multiple layers and computes the L2 norm after normalizing and scaling each channel. Another network, G , is trained to predict perceptual quality h from

distance pair d_0 and d_1 . The lower the LPIPS score, the higher the similarity between the two images.

Peak Signal to Noise Ratio

Peak Signal-to-Noise Ratio (PSNR) is a commonly used metric to measure the quality of a dehazed image obtained from a dehazing algorithm, with respect to a ground truth (GT) image [Zho06]. PSNR measures the degree of signal distortion between the two images and a higher value of PSNR signifies better quality of the dehazed or desmoked image. The PSNR can be calculated as follows:

$$PSNR = 10 \log_{10} \left(\frac{MAX^2}{MSE} \right) \quad (2.9)$$

where MAX is the maximum possible pixel value of the image and MSE is the mean squared error between the dehazed image (I_{hazed}) and the ground truth image (G). The MSE is computed as:

$$MSE = \frac{1}{N} \sum_{i=1}^N (G_i - I_{\text{hazed},i})^2 \quad (2.10)$$

where N is the total number of pixels in the image. The goal is to minimize the MSE value to obtain a high PSNR value.

Structural Similarity Index

When it comes to evaluating the visual quality of dehazed images from the perspective of human perception, the Peak Signal to Noise Ratio (PSNR) metric alone is not considered very effective. As a result, researchers have turned to the Structural Similarity Index Metric (SSIM) [WBS04] as an alternative. This metric compares the ground truth and dehazed images in terms of contrast, luminance, and structure.

The SSIM is calculated by taking into account the means and variances of the restored image and the ground truth image, as well as their cross-variance. Two constants, c_1 and c_2 , are included in the calculation, with default values of 0.01 and 0.03, respectively. The formula for calculating SSIM (also discussed in Chap. 6) is given by:

$$\text{SSIM}(GT, Out) = \frac{(2\mu_i\mu_r + c_1)(2\sigma_{ri} + c_2)}{(\mu_i^2 + \mu_r^2 + c_1)(\sigma_i^2 + \sigma_r^2 + c_2)} \quad (2.11)$$

The resulting SSIM score ranges from 0 to 1, with a score of 1 indicating that the two images being compared are identical. Since SSIM is highly sensitive to variations in contrast and illumination, it can effectively identify issues related to dehazing, such as incomplete haze removal or over-saturation of pixels.

CIEDE 2000

Color distortion is a common issue that arises during the dehazing or desmoking process. To address this, researchers in this field have turned to the CIEDE 2000 color difference metric [SWD05], which offers a more precise assessment of color restoration that is closer to human eye perception.

CIEDE 2000 yields values between 0 and 100, with smaller values indicating better color preservation, and values less than 1 corresponding to imperceptible differences to the human eye. A value of 100 indicates that the colors in two images are completely opposite to each other. This metric is considered an accurate way of evaluating color differences in dehazed images, making it a valuable tool for researchers in this area.

The evaluation metrics commonly used in popular image dehazing and image desmoking algorithms, namely PSNR and SSIM, will be employed in this thesis for evaluation purposes. The inclusion of these metrics will establish a benchmark for future research in this field. To ensure consistency with established practices, this approach has been adopted. Additionally, this will provide a standardized methodology for evaluating the performance of the proposed algorithms in comparison to existing methods.

2.4 Datasets used for Image desmoking/dehazing

Image desmoking, or the removal of smoke and haze from images, is an important problem in computer vision with many practical applications. Despite the significance of this problem, there are currently no dedicated datasets available that can be used to train models for image desmoking tasks. Previous attempts at image desmoking have been focused primarily on surgical images. In the initial stages of image dehazing research, there were limited datasets available, and they were typically small in size. Researchers relied on a small number of images to validate the performance of their proposed haze

removal algorithms, often downloading hazy images from the internet. Unfortunately, such images lacked ground truth data, making it difficult for researchers to evaluate their methods accurately. Today, researchers use two types of datasets in the field of image dehazing: natural hazy images without a reference image (i.e., real images) and synthetic hazy images with depth or ground truth information. Assessment methods vary for each type of dataset which are discussed in the following section.

Fattal's Dataset

Fattal's dataset [Fat08] is the most popular dataset for the assessment of dehazing capability. It provides 12 synthetic hazy images along with 31 realistic hazy images, containing various benchmarks hazy images, such as nighttime haze, heavily dense haze, white objects, depth discontinuities, different illumination conditions, and sky regions. Fig. 2.15 shows the Fattal's Dataset.

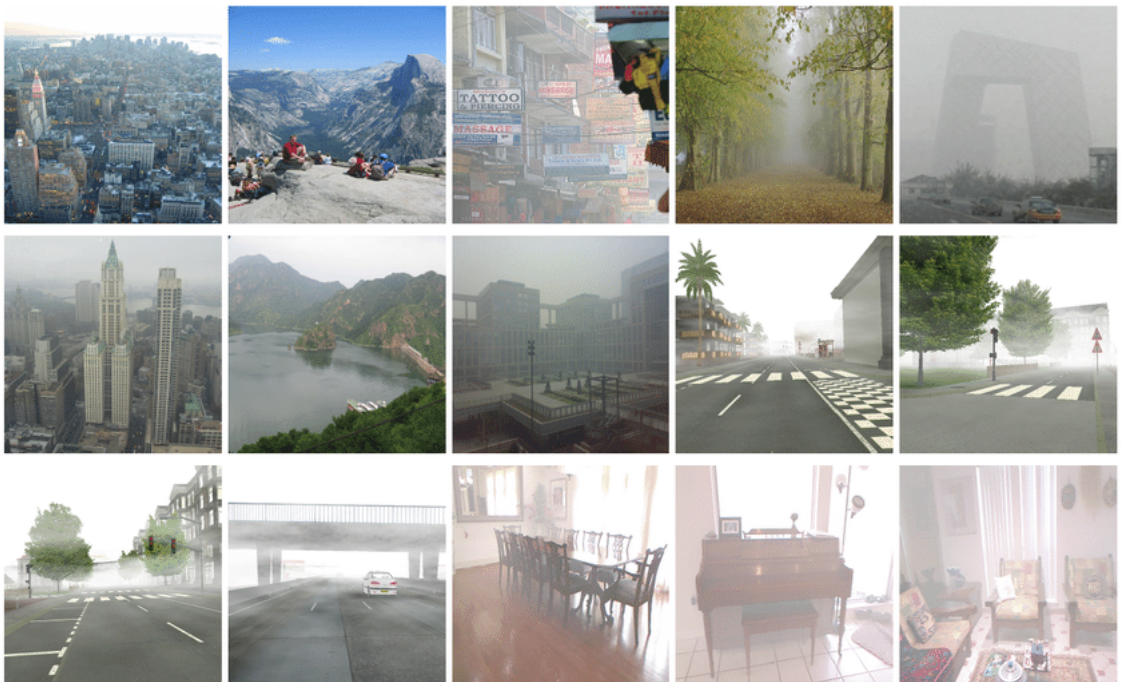


Figure 2.15: Sample images of Fattal's Dataset [Fat08].

D-Hazy Dataset

The D-Hazy dataset contains 1400+ pairs of synthetic hazy and haze-free images of indoor scenes. This dataset is generated using Middlebury [SS02] and NYU depth [SHK12] datasets, containing their corresponding depth maps. The transmission map is computed

for each image based on atmospheric light and the scattering coefficient. Fig. 2.16 shows the D-Hazy Dataset.



Figure 2.16: Sample images from D-Hazy dataset [AAD16].

Haze RD Dataset

This dataset includes 15 outdoor scenes with realistic hazy conditions, each simulated with five different weather conditions ranging from thin to dense haze, and a visible range from 50 to 1000 meters. The images have high resolutions and validate the scattering theory of the physical model. A depth map of each hazy scene is estimated by fusing structure from motion and lidar. Fig. 2.17 shows the Haze RD Dataset.

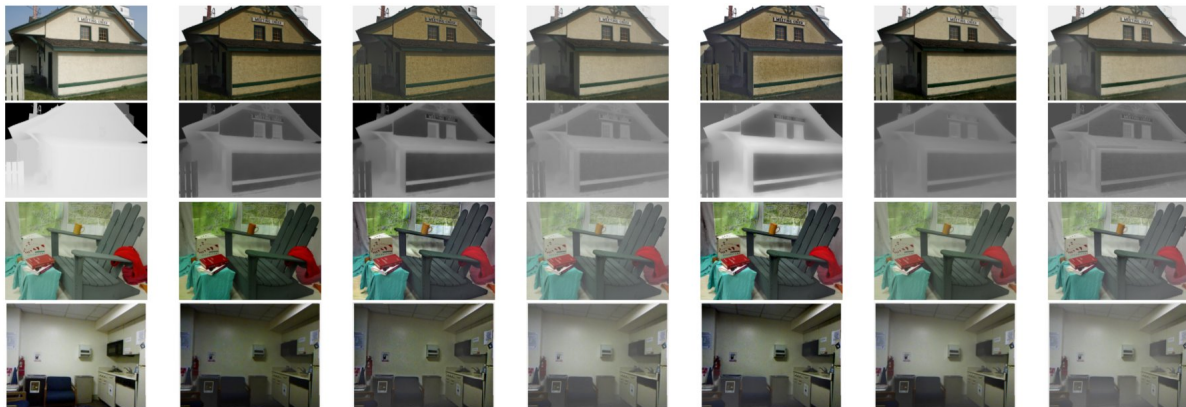


Figure 2.17: HazeRD samples from left to right, a Haze-free image, b depth map, simulated hazy images with the visual range of c 50 m, d 100 m, e 200 m, and f 500 m, respectively [ZDS17].

Dense-Haze

The Dense-Haze dataset proposed by Ancuti et al. [AAS19] includes real-world hazy images characterized by dense and homogeneous haze. It consists of 33 pairs of real hazy and their corresponding haze-free images. Fig. 2.18 shows the Dense-Haze Dataset.



Figure 2.18: Three examples of the Dense-Haze dataset that provides 33 pairs of hazy and corresponding haze-free (groundtruth) outdoor images [AAS19].

RESIDE

RESIDE [LRF19] is a large-scale dataset of hazy images, containing both synthetic and realistic hazy images for single image dehazing. The dataset is available in two versions: standard RESIDE and RESIDE- β . The standard RESIDE contains three subsets: indoor training test (ITS), synthetic objective testing set (SOTS), and hybrid subjective testing set (HSTS). ITS consists of 13,990 synthetic hazy images generated using 1399 haze-free images from NYU2 [SHK12] and Middlebury [SS02] stereo indoor datasets. For each haze-free image, 10 synthetic hazy images are generated with uniform random atmospheric light between $[0.7, 1.0]$ and scattering coefficient between $[0.6, 1.8]$. SOTS contains 500 different images with white scenes and dense haze synthesized from NYU2 which are not used in the training set. HSTS selects 10 synthetic outdoor hazy images, together with 10 realistic hazy images. RESIDE- β provides two more subsets: outdoor training set (OTS) and real-world task-driven testing set (RTTS). The OTS contains 72,135 hazy images and RTTS contains 4322 images. Fig. 2.19 shows the RESIDE Dataset. RESIDE is a large-scale dataset of hazy images that allows for comprehensive evaluation of dehazing methods using full reference, no-reference, and human subjective ratings. It has significantly contributed to the advancement of single image dehazing research.

Although there are a variety of datasets available for evaluating models on image dehazing, none of them fits the purpose of the thesis of image desmoking of gauge images. Therefore, a custom dataset, which fits the requirements of the task, was acquired (see Chapter 5 for more details).



Figure 2.19: Sample images from RESIDE dataset [LRF19].

3 Image Processing Basics

This chapter provides a comprehensive overview of the fundamental techniques used in image processing, as well as the different types of images and various image processing techniques that are commonly employed. This information is essential for understanding how digital images are processed or manipulated in the context of image desmoking and dehazing algorithms.

3.1 Image Processing

Digital Image Processing refers to the use of computer algorithms to perform various operations on digital images, such as enhancing image quality, removing noise, extracting information from the images, and manipulating the images in various ways. These techniques are crucial in various fields, including medical imaging, remote sensing, robotics, and computer vision. Before applying these techniques, it is necessary to preprocess the digital images, which involves various steps such as noise reduction, contrast enhancement, and image restoration. This is done to improve the overall quality of the image and make it suitable for further processing. Some applications of image processing are in face recognition, object detection systems, which are useful in various fields such as security systems, robotics, self-driving cars, and industrial automation.

This thesis focuses on the crucial application of digital image processing known as image desmoking, which enhances the visual quality of degraded images due to atmospheric phenomena such as smoke. This technique finds applications in various fields, including critical infrastructure monitoring, medical imaging, and remote sensing. Digital image processing is a fundamental tool for the analysis, visualization, and decision-making with images in various domains.

3.2 Computer Vision

Digital image processing falls under the category of computer vision. To gain a fundamental understanding of computer vision, it is essential to first consider the differences in how humans and computers perceive images. The Human Visual System (HVS) can process an image almost instantly with a single glance, while computers require significantly more processing time and effort to analyze and interpret the same image. Data scientists are

working to address this challenge by developing complex algorithms and models that can enable machines to see and interpret the world in a way that's similar to humans.

In recent years, the field of computer vision has experienced significant growth and development, largely due to advances in deep learning and artificial intelligence. In fact, computers are now capable of surpassing humans in some visual tasks. Computer vision is a subset of deep learning and artificial intelligence, where computers are instructed through algorithms to learn how to see and interpret the world around them.

Computer vision can be roughly summarized in three basic steps:

- **Image Acquisition:** Acquisition of images or datasets, which can be obtained through various means such as photos or videos.
- **Image processing:** This involves using various techniques to enhance, analyze, and extract useful information from the acquired images. Examples of image processing techniques include filtering, segmentation, feature extraction, and object recognition. (The deep learning models used in thesis perform automatic feature extraction from images of gauges.)
- **Decision making:** This step involves using the information extracted from the image to make decisions or take actions. Examples of decision-making tasks in computer vision include object tracking, navigation, and autonomous driving.

In digital image processing, images are treated as 2D or 3D matrices of numerical values, where each value in the matrix corresponds to the intensity or amplitude of a particular pixel in the image. The pixel intensity value represents the amount of light reflected or emitted from that pixel. Typically, 8-bit images are used, which have pixel values ranging from 0 to 255. A pixel value of 0 represents a completely black pixel, while a value of 255 indicates a completely white pixel. The range of values in between 0 and 255 represents various shades of gray, with lower values representing lighter shades and higher values representing darker shades. These numerical values of pixels form the basic building blocks for various image processing techniques such as filtering, segmentation, and feature extraction.

Computers work with different kinds of 'images' based on how they are represented. Following are some of the types.

Binary Image

Binary Images are a type of digital images in which each pixel has only two possible values, 0 and 1. These two values correspond to black and white colors respectively, where 0 represents the absence of color (black) and 1 represents the presence of color (white). A binary image is referred to as 1bit/pixel image, as it takes only 1 binary digit to represent a single pixel. Fig. 3.1 shows a clear gauge image obtained from the acquired realistic dataset, and Fig. 3.2 shows the binary image of the clear gauge image which has been obtained by using the OpenCV library.



Figure 3.1: RGB image of a gauge.

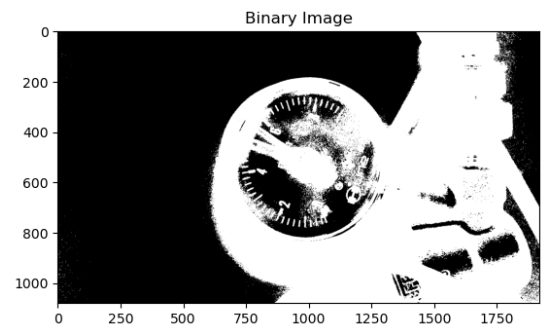


Figure 3.2: Binary output image of the gauge.

Greyscale Image

Grayscale images are images with a single channel, where each pixel has a value between 0 and 255 representing its intensity or brightness. The 0 value indicates a perfectly black pixel, while the 255 value represents a perfectly white pixel. A greyscale image contains 8 bit/pixel data, which allows to have (0-255) different gray levels. Fig. 3.1 shows a clear gauge image obtained from the acquired realistic dataset, and Fig. 3.3 shows its corresponding greyscale image.

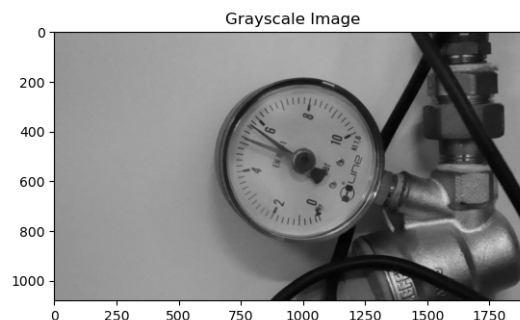


Figure 3.3: Output greyscale image of the image.

In addition, grayscale images are often used as a preprocessing step in image analysis and feature extraction algorithms. This is because they can be used to extract features that are related to the intensity or brightness of the image, such as edges or textures.

RGB Image

RGB images are widely used, where each image is represented as a 16-bit matrix. These matrices contain 65,536 different color combinations per pixel, and they are composed of three primary color channels: red, green, and blue (RGB). These channels are superimposed to produce a range of colors. Unlike grayscale and binary images, a pixel in an RGB image requires three coordinates to specify its location.

Each channel in an RGB image represents a specific color, and the intensity of the color can be determined by the value of the pixel in the matrix. It is essential to note that each channel in an RGB image has different histogram values, indicating that each channel has a different range of intensities. Understanding these histogram values is critical for image processing tasks, such as segmentation, object recognition, and image enhancement. In Fig. 3.4, splitting of a RGB gauge image into three channels is shown.

3.3 Image Processing Techniques

Image processing is a very vast and complex field where different techniques are used to manipulate the image based on the kind of result that is supposed to be achieved. These techniques could be utilized to enhance or improve the quality of image, removal of objects from an image or even generate new images from scratch. The following methods have been selected based on their relevance and utility in the field of image denoising and dehazing, either currently or potentially.

Image Generation

Image synthesis is a significant task in image processing, particularly in Deep Learning algorithms that require abundant labeled data for training. Image generation methods utilize Generative Adversarial Networks (GANs), which are unique neural network architectures that consist of two distinct models: the generator and the discriminator. Fig. 3.5 shows a general framework of GANs. The generator's role is to create synthetic images that closely resemble real images, while the discriminator's role is to differentiate between

real and synthetic images. The two models are then pitted against each other in an adversarial game, with the generator attempting to produce realistic images to deceive the discriminator, and the discriminator continually improving its ability to distinguish between real and synthetic images. Through this adversarial training process, the generator learns to create photo-realistic images that can be used for training other Deep Learning models. GANs have demonstrated remarkable success in generating images that are difficult to distinguish from real images, making them a powerful tool for various applications such as image editing, virtual reality, and computer vision.

Image Desmoking

Image desmoking or image dehazing, which is the focus of this thesis, also can be considered an image processing technique that is used to improve the visibility and quality of images that have been degraded by atmospheric haze or smoke. The image desmoking/dehazing process involves estimating and removing the smoke or haze from the image.

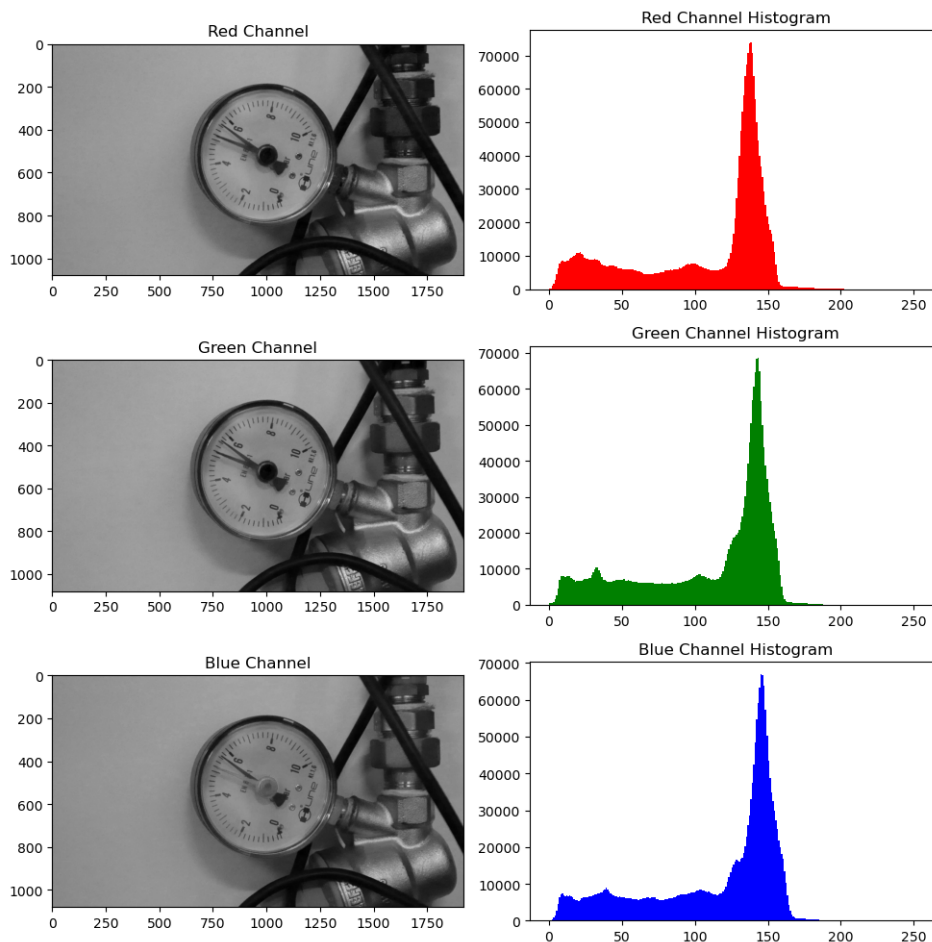


Figure 3.4: Splitting of the RGB gauge image into its RGB channels.

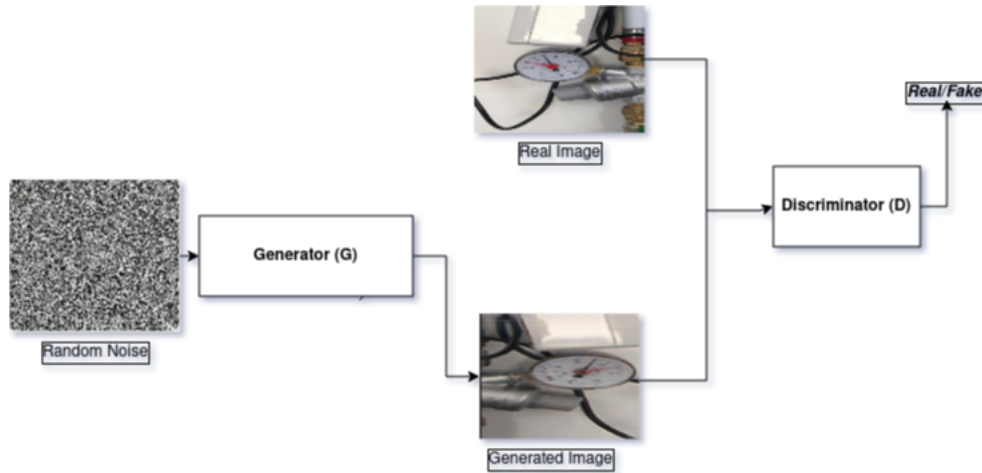


Figure 3.5: General framework of GANs.

The physical methods rely on estimation of transmission maps through atmospheric scattering models, whereas the Deep learning based models try to learn the mapping between a smoky image to a clear image directly (more explanation in chapter 5).

Image Denoising

Also, image denoising is another image processing technique. However, it is important to note that Image denoising and image desmoking or dehazing are not similar. Although both of the techniques aim to enhance images, the kind of degradation in images that they aim to reduce is fundamentally different. Noise in images refers to random variations in brightness or color that arise from limitations in the image capture device or transmission medium. It can take various forms, such as salt-and-pepper, Gaussian, or Poisson noise, and can be additive or multiplicative [VA13]. Image denoising algorithms focus on removing additive noise by estimating the original image from the observed noisy image. These algorithms assume that the underlying image is minimally distorted or changed and that noise is the primary source of degradation. By reducing the amount of noise, they aim to enhance the quality of the image [SYL14]. Haze or smoke refers to the degradation of an image caused by the scattering or absorption of light by particles or molecules in the atmosphere, resulting in reduced contrast, color saturation, and sharpness. These algorithms assume that the observed image is distorted due to the presence of atmospheric particles or molecules that have weakened the light, and work by estimating the original image from the observed hazy or smoky image by removing the atmospheric effects. Haze or smoke can result from factors like air pollution, fog, or wildfire smoke [JZW17].

In fig. 3.6, the histogram changes in a clear image, an image with haze, and an image with noise are shown.

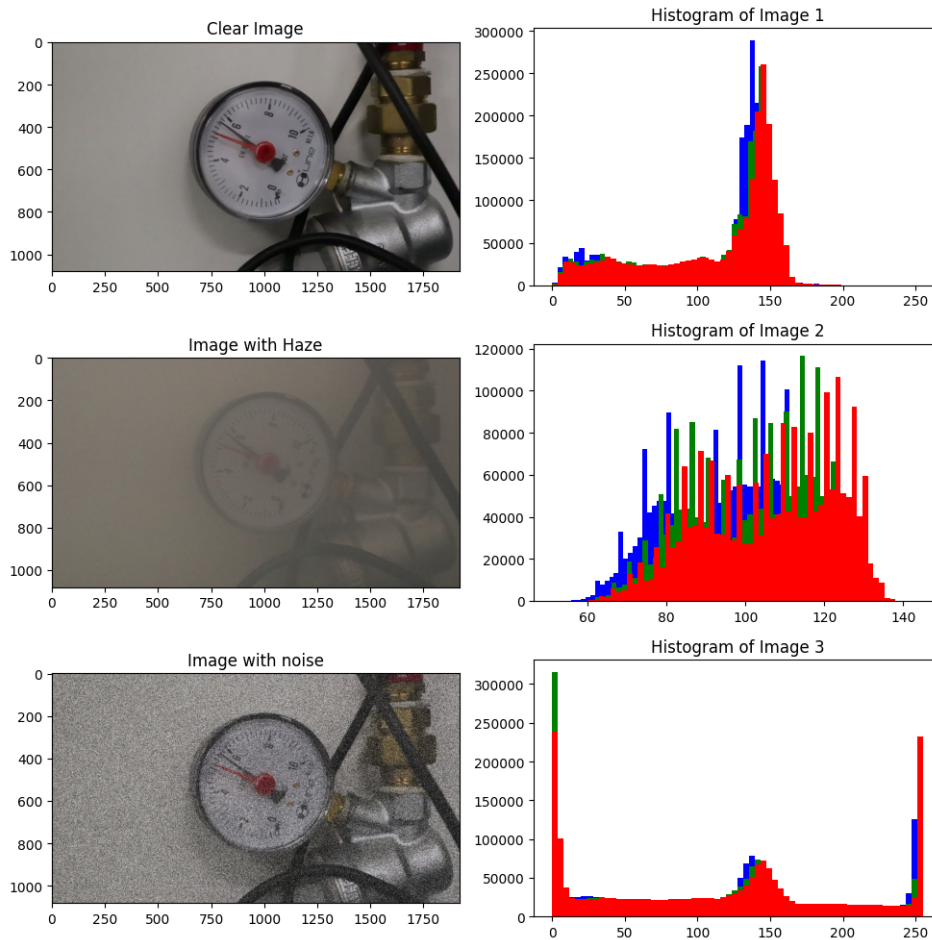


Figure 3.6: Left: images of gauges: clear (top-panel), haze (middle-panel), and with noise (bottom-panel). Right: histograms output of the respective gauge image on the left with two degradations, haze and noise.

It is crucial to observe the distinction between a noise image and a clear or hazy image (see Fig. 3.6). In the noise image, the majority of pixels tend to be either almost entirely white or entirely black, while the distribution of pixel intensities is more diverse in hazy and clear images.

The underlying assumptions and approaches used by image denoising and image dehazing/desmoking algorithms are fundamentally different, and the techniques that work well for one problem may not be suitable for the other. Therefore, specialized algorithms have been developed for image dehazing/desmoking that take into account the specific characteristics of atmospheric scattering and absorption.

4 Artificial Intelligence

This chapter presents fundamental concepts of deep learning, such as Convolutional Neural Networks (CNNs), attention mechanisms, and components of a training process. The comprehension of these concepts is essential to grasp the implementation and training of deep learning models for the tasks of image dehazing and image desmoking, which are thoroughly explained in Chapter 2 and Chapter 5. This chapter serves as an exhaustive introduction to the basic principles of deep learning, laying the foundation for researchers and practitioners to explore the intricacies of image dehazing and desmoking in subsequent chapters.

4.1 Deep Learning

The terms Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL) are often used interchangeably. DL is a subset of both ML and AI. AI aims to incorporate human intelligence into machines or systems, while ML automates model building by learning from data or experience [Sar21]. DL involves learning from data through multi-layer neural networks, which process data in multiple stages. The term "deep" refers to the use of multiple layers to build a data-driven model. Fig. 4.1 shows the relation.

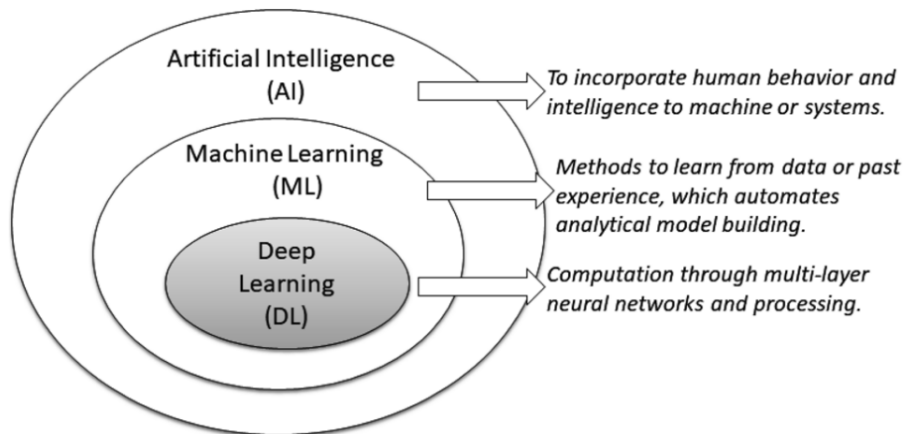


Figure 4.1: Relation between AI, ML, and DL [Sar21].

4.2 Convolutional Neural Networks(CNNs)

Throughout this thesis, the following terms will be used consistently to prevent any confusion: A "parameter" refers to a variable that the model learns during training, while a "hyperparameter" is a variable that must be set before the training process begins. The term "kernel" refers to a set of learnable parameters used in convolution operations, and "weight" is a term used interchangeably with "parameter," although it is preferred to use "weight" when referring to a parameter outside of convolution layers, such as in fully connected layers.

The Convolutional Neural Network (CNN) [LBB98] is a type of deep learning model inspired by the organization of animal visual cortex [HW68]. It is specifically designed to automatically learn spatial hierarchies of features in data with a grid pattern, such as images. The CNN is composed of three types of layers: convolution, pooling, and fully connected layers. The convolution layer is crucial to the CNN architecture, and it applies a small grid of parameters called a kernel at each image position to perform feature extraction. The extracted features become progressively more complex as they are fed into subsequent layers. The optimization of kernel parameters occurs during training, which involves minimizing the difference between the model's output and the ground truth labels through an optimization algorithm called backpropagation and gradient descent. Overall, CNNs are highly efficient for image processing because they can extract features from anywhere in the image.

Building blocks of CNN Architecture:

Convolution Operation

The size and number of kernels are key hyperparameters that define the convolution operation. The size is typically 3x3, 5x5 or 7x7, while the number determines the depth of the output feature maps. The output feature map size reduces compared to the input tensor after the convolution operation. Padding is a technique to retain the same in-plane dimension of the output feature map. The distance between two successive kernel positions is called a stride, which defines the convolution operation. Fig. 4.2 shows an example of a convolution operation.

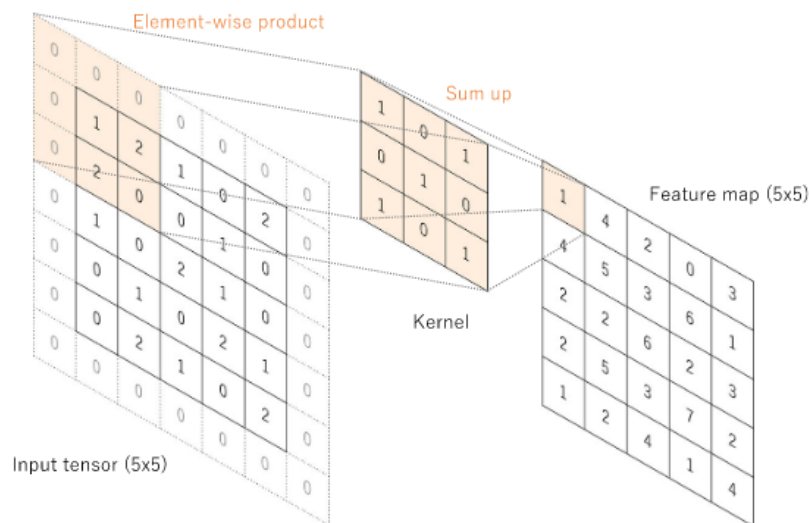


Figure 4.2: A convolution operation with zero padding so as to retain in-plane dimensions. Note that an input dimension of 5×5 is kept in the output feature map. In this example, a kernel size and a stride are set as 3×3 and 1, respectively [YND18].

Weight Sharing

The key feature of a convolution operation is weight sharing, where kernels are shared across all image positions. This leads to the extraction of local feature patterns that are translation invariant and capture increasingly larger fields of view through downsampling. Pooling operation, which summarizes the features within a feature map using a filter, is also done along with downsampling. Weight sharing also reduces the number of parameters to learn in comparison with fully connected neural networks.

Hyperparameters

The hyperparameters of the convolution operation, such as the kernel size, number of kernels, padding, and stride, are set before the training process starts. Kernels are the only parameters automatically learned during the training process in the convolution layer. Fig. 4.3 shows the parameters and hyperparameters in a typical CNN. Note that this table includes the CNN layer, the parameters and hyperparameters specific to each layer, and a brief description of each hyperparameter.

	Parameters	Hyperparameters
Convolution layer	Kernels	Kernel size, number of kernels, stride, padding, activation function
Pooling layer	None	Pooling method, filter size, stride, padding
Fully connected layer	Weights	Number of weights, activation function
Others		Model architecture, optimizer, learning rate, loss function, mini-batch size, epochs, regularization, weight initialization, dataset splitting

Figure 4.3: Parameters and hyperparameters in a CNN [YND18].

4.3 Residual Connections

Vanishing gradients is a problem that arises during the training of deep neural networks, where the gradients become very small as they propagate backwards through the network during backpropagation (see section 4.5). This can lead to very slow convergence during training or even cause the network to stop learning altogether. When the gradients vanish, the network parameters fail to update, and the network can become stuck at a suboptimal solution.

In FFA Net [QWB19] and AEER Net [WQL21], residual skip connections can help to improve the performance of these networks by allowing them to be trained more effectively. Both FFA Net and AEER Net are very deep networks with many layers, which makes them susceptible to the problem of vanishing gradients. By using residual skip connections, these networks can overcome this problem and learn more effectively, which can lead to better performance.

4.4 Attention Mechanism

Attention mechanism [VSP17] is a technique used in deep learning models that allows the model to selectively focus on certain parts of the input data while ignoring others. It works by assigning weights to different parts of the input data, based on their importance to the task at hand. These weights are learned by the model during training, and can be adjusted based on the model's performance.

In dehazing and desmoking architectures such as FFA Net, attention mechanism is used to selectively enhance or suppress certain features in the input image, in order to reduce the effect of haze or smoke. Specifically, FFA Net [QWB19] uses a novel Feature Attention (*FA*) module consisting of Channel Attention (*CA*) and Pixel Attention (*PA*) layers to selectively focus on important features in the input image. The *CA* layer computes a channel attention map, which assigns weights to different channels of the feature maps

based on their importance. The *PA* layer computes a pixel attention map, which assigns weights to different pixels in the feature maps based on their saliency.

The basic blocks in FFA Net consist of the *FA* module and local residual learning. The attention based different levels feature fusion structure is used to combine features from different levels of the network, and to selectively fuse them based on their importance. This allows FFA Net to selectively enhance or suppress features in the input image, in order to achieve better dehazing or desmoking results.

The attention mechanism is mathematically represented by the attention weights, which are learned during training. These weights can be computed using different methods, such as dot product, cosine similarity, or softmax function. The attention mechanism has been used in various applications, such as natural language processing, computer vision, and speech recognition.

4.5 Training

Training a neural network involves finding the optimal kernels in convolution layers and weights in fully connected layers that minimize the differences between the predicted outputs and the given ground truth labels on a training dataset. The backpropagation algorithm is the most commonly used method for training neural networks, where the loss function and the gradient descent optimization algorithm play critical roles. The model's performance under specific kernels and weights is calculated using a loss function through forward propagation on a training dataset, and the learnable parameters, namely kernels and weights, are updated according to the loss value using an optimization algorithm called backpropagation and gradient descent.

The loss function, also known as the cost function, measures the compatibility between the output predictions of the network through forward propagation and the given ground truth labels. Cross-entropy is commonly used for multiclass classification, while mean squared error is typically used for regression to continuous values. The choice of loss function is a hyperparameter and needs to be determined based on the given tasks. Gradient descent is a commonly used optimization algorithm that iteratively updates the learnable parameters, i.e., kernels and weights, of the network to minimize the loss. The gradient of the loss function provides the direction in which the function has the steepest rate of increase, and each learnable parameter is updated in the negative direction of the gradient with an arbitrary step size determined based on a hyperparameter called the learning rate (see Fig. 4.4). The gradient is mathematically a partial derivative of the loss with respect

to each learn-able parameter, and a single update of a parameter is formulated.

$$w \leftarrow w - \alpha * \frac{\partial L}{\partial w}, \quad (4.1)$$

where w stands for each learnable parameter, α stands for a learning rate, and L stands for a loss function. In practice, the learning rate is one of the most important hyperparameters to be set before the training starts. For practical reasons such as memory limitations, the gradients of the loss function with respect to the parameters are computed using a subset of the training dataset called mini-batch, and applied to the parameter updates. This method is called mini-batch gradient descent, also frequently referred to as stochastic gradient descent (SGD), and a mini-batch size is also a hyperparameter. Many improvements on the gradient descent algorithm have been proposed and widely used, such as SGD with momentum, RMSprop, and Adam [Sha20].

Loss:

The loss function is a crucial component in training any machine learning model. It represents the measure of how well the model is performing in its task. The goal is to minimize the loss function to improve the model's accuracy. In dehazing and desmoking architectures, various loss functions are used to estimate transmission maps, clear images, hazy image reconstruction, atmospheric light regression, etc. Examples of commonly used loss functions are L1 loss, L2 loss, perceptual loss, structure loss, gradient loss, and total variation loss [ZGF17]. These loss functions are used in combination to achieve better dehazing and desmoking results.

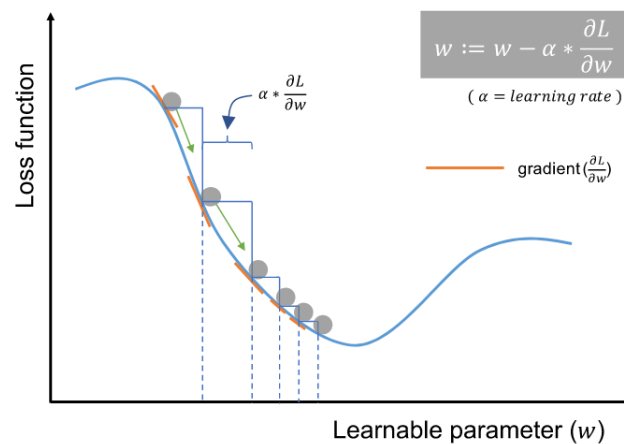


Figure 4.4: Gradient descent [YND18].

Activation:

Activation functions play a significant role in deep learning models. These functions introduce non-linearity to the network and help the model learn complex relationships between input and output. Some commonly used activation functions are sigmoid, ReLU, and tanh. In dehazing and desmoking architectures, activation functions are used in the neural network to improve its performance. For instance, ReLU is used in the encoder of the AEER Net architecture to introduce non-linearity and improve feature extraction.

Training:

Training is the process of updating the weights of the neural network to minimize the loss function. During training, the input data is fed into the network, and the output is compared with the ground truth to calculate the loss. The backpropagation algorithm is then used to update the weights of the network to minimize the loss. Training is an iterative process that continues until the network achieves the desired accuracy.

Batch Size:

Batch size refers to the number of samples that are fed into the neural network at once during training. A larger batch size can lead to faster training times, but it also requires more memory. A smaller batch size can lead to slower training times, but it allows for more frequent updates to the weights of the network.

Epochs:

An epoch is a single pass through the entire training dataset during training. In each epoch, the network is trained on every sample in the dataset. The number of epochs determines how many times the network is trained on the dataset. A larger number of epochs can lead to better accuracy, but it also increases the risk of overfitting.

5 Methodology

This chapter describes the methodology used in this work. It is split up in two different sections. First section 5.1 deals with the dataset acquisition procedures. The second section 5.2 focuses on the implementation of the three selected methods: BCCR, FFA Net and AECR Net(which were introduced in chapter 2).

5.1 Dataset Acquisition

Datasets are important for training any machine learning model. The more data a model has, the better predictions it can make. The focus of this section is to explain how the datasets were achieved for the task of removing smoke and haze from gauge images.

5.1.1 Datasets Available

For the BCCR [MWD13] approach, there is no need for a dataset since the model works on prior assumptions. Using these assumptions, the BCCR method tries to estimate the unknown scene transmission. However, for the deep learning approaches, a large enough dataset is necessary in order to train them for a specific task. For the task of enhancing gauge images, a dataset involving two types of images are needed. One, with a clear image of a gauge and with no smoke or haze in it. Second, the same image of gauge, but with smoke or haze in it. The objective is to make the model learn the transition from smoky gauge image to a clear gauge image. But, currently, there are no public datasets available which contain images of gauges. This posed a big challenge.

There are some datasets which are publicly available for training models for image dehazing. These datasets contain images of general indoor and outdoor environments (mentioned in 2). Also, these datasets were made to train models for image dehazing and not for the purpose of image desmoking.

5.1.2 Requirements for Custom Dataset

To train the FFA Net and the AECR Net models for the task of removing smoke from images of gauges, it was necessary to have a diverse dataset containing both clear, sharp RGB images of gauges as well as smoky images of gauges. Also, as an addition, a synthetic

haze dataset is also acquired to train models for dehazing tasks. This was important because the models required both an input image and a ground truth image to compute the loss, and the data needed to be diverse to prevent over-fitting of the models. To ensure diversity in the data, images of different gauges were captured from different viewpoints. Additionally, the input data needed to be corrupted with thin, medium, and heavy smoke/haze to better simulate real-world conditions and allow the models to learn to remove smoke/haze of any density.

To fulfill the requirement of mapping a single clear image of a gauge to 10 different smoky/hazy images, each containing smoke/haze of increasing density, it was necessary to capture a range of smoky/hazy images for each clear image. This ensures that the models were trained to handle varying degrees of smoke/haze and could effectively remove it from the images of the gauges.

In order to acquire a realistic dataset, an experimental setup was implemented within a confined space equipped with a smoke machine, a mounted pressure gauge, and an RGB camera. The experiment was conducted at the Jülich Solarzentrum DLR, on October 27th, 2022, as part of a campaign aimed at generating accurate and reliable data.

Setup and Components

To capture clear images of the pressure gauge, a Hurricane 1200 smoke machine was used to create smoky conditions. The Hurricane 1200 is a professional-grade smoke machine that produces dense fog and haze effects, which can simulate various environmental conditions. To capture images of the gauge without smoke, the smoke machine was turned off, and an EOS M50 Mark 2 RGB camera was fixed in place to capture the gauge's readings. It is a mirror-less camera that features a 24.1-megapixel APS-C CMOS sensor, which provides high-quality images with low noise levels. Fig. 5.1 shows the components used.

Subsequently, to obtain images of the gauge under smoky conditions, the smoke machine was activated inside the enclosed space. The room was filled with dense smoke, and the density of smoke was allowed to dissipate naturally over time, without any intervention or manipulation, in order to maintain the integrity and authenticity of the captured data. Throughout this process, the RGB camera remained mounted in the same orientation and captured images of the gauge under smoky conditions.

The realistic dataset acquired was unable to fully meet the requirement of mapping 10 distinct smoky/hazy images to a single clear image. This was due to the limited capacity to monitor and control the density of smoke generated by the available smoke machine.

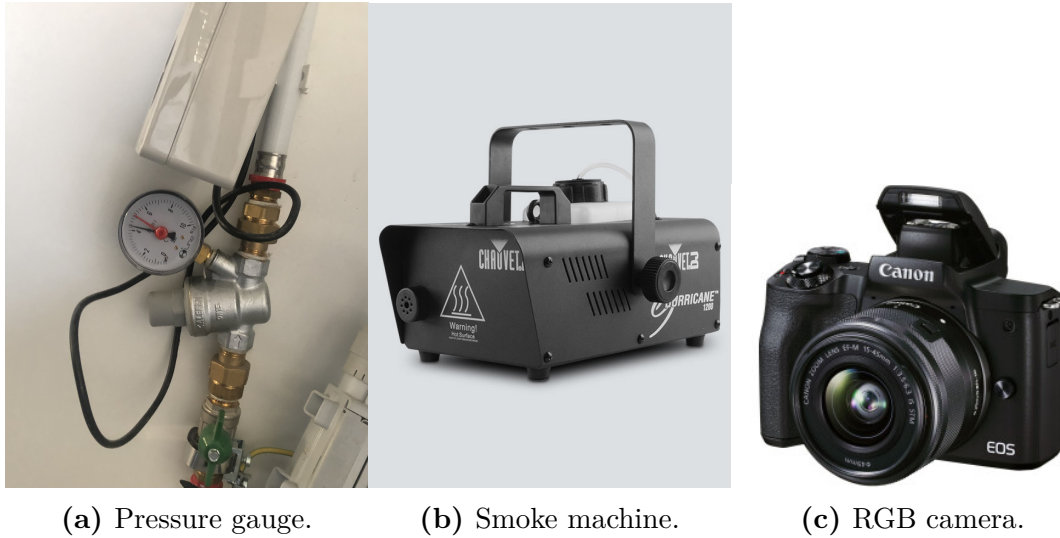


Figure 5.1: Components used in Realistic data acquisition.

As a result, the obtained dataset included a mixture of clear images and smoky/hazy images, rather than the desired mapping of 10 distinct smoky gauge images to one single clear gauge image.

5.1.3 Synthetic Dataset

Various methods for generating synthetic datasets are available, and one such approach involves the use of Generative Adversarial Networks (GANs) to create new images. However, GANs often lack control over the properties of generated images, such as smoke or haze levels. In addition, they require diverse training data that may not always be readily available.

In this thesis, the decision was made to utilize Unreal Engine 5.0 for synthetic image generation due to its robust capabilities. Unlike GANs, Unreal Engine 5.0 allows precise control over image attributes, such as different smoke or haze levels. Additionally, Unreal Engine 5.0 facilitates the generation of diverse images without necessitating an extensive dataset, making it an ideal tool for synthetic data acquisition. The selection of Unreal Engine 5.0 for image generation was motivated by its versatility and efficiency in producing high-quality synthetic data.

Unreal Engine

For synthetic data acquisition, the idea was to capture scenes featuring different gauges within a virtual office environment. The virtual environment, known as KAS (see Fig. 5.2),

was made available by a different group working at DLR¹. This KAS environment provided a highly realistic representation of an actual office space located in Sankt Augustin, DLR. The environment also included furniture, such as desks, chairs, and cabinets, positioned in a manner consistent with the actual office setting. This setting provides a highly controlled environment that enables the capture of scenes featuring different gauges with consistent lighting, positioning, and other environmental factors.

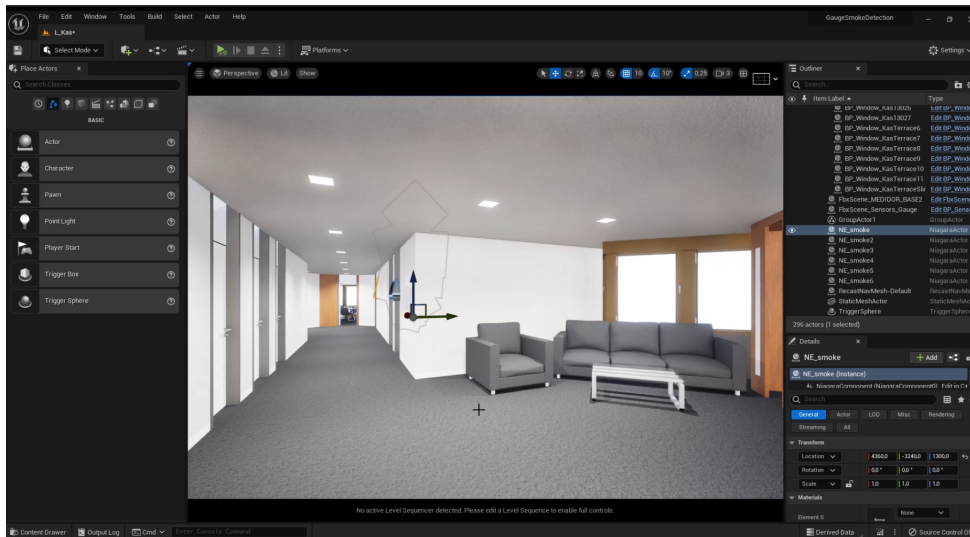


Figure 5.2: A snapshot of the virtual environment KAS in UE.

Additionally, virtual smoke components are generated within Unreal Engine 5.0 software to simulate real smoke. The use of virtual smoke components adds an additional layer of realism to the captured scenes, thereby enhancing the overall quality of the synthetic dataset. Overall, the use of Unreal Engine 5.0 in combination with the KAS environment has provided a powerful tool for generating synthetic data that can be used to train and test machine learning models in various domains.

Setup and Components

Three different 3D models of different gauges are acquired from free3d.com which are available for free. Then these three models are loaded into the KAS environment and mounted on walls of the virtual rooms at different locations. To simulate real smoke, a *Niagara smoke system* within the Unreal Engine is created.

The Niagara system in Unreal Engine 5 provides a powerful tool for creating dynamic and realistic smoke effects. By using SubUV materials with the sprite renderer, it is possible to create unique smoke patterns and animations. The process involves creating a

¹Deutsches Zentrum für Luft- und Raumfahrt

Niagara emitter, adjusting the spawn rate and velocity of particles, randomizing particle properties such as life span and rotation, and shaping the emitter to match the desired effect. Curve tools are also used to adjust particle size and opacity over time. Additionally, SubUV textures are applied to the particles through a Niagara material, allowing for more complex and detailed animations. The acceleration force of the particles was also controlled, which adds to the realism of the effect. Overall, the Niagara system and SubUV materials provided a flexible and customizable solution for creating impressive smoke effects in Unreal Engine 5. In order to achieve varying smoke densities, the spawn rate is increased from 2.5 to 25 with a step size of 2.5, resulting in ten different levels of smoke density, ranging from thin and light smoke to thick and dense smoke.

After having the smoke machines and the different gauges setup within the virtual environment, the next step is to capture images, which would form the synthetic dataset. For capturing the required clear images and smoky images and, a camera actor was placed was setup in front of every gauge. Also, a path is defined that the camera would follow when capturing a video. The parameters of the smoke components were changes for varying smoke densities and 10 sequences were recorded. This enabled the camera to follow a path as it recorded the video of the gauges while constantly changing the angle and distance of the recording. This variation was critical to ensure the dataset was realistic and diverse enough to be used for analysis. This path and video capturing was made possible through a plugin called *EasySynth* which was available within the Unreal Engine marketplace.

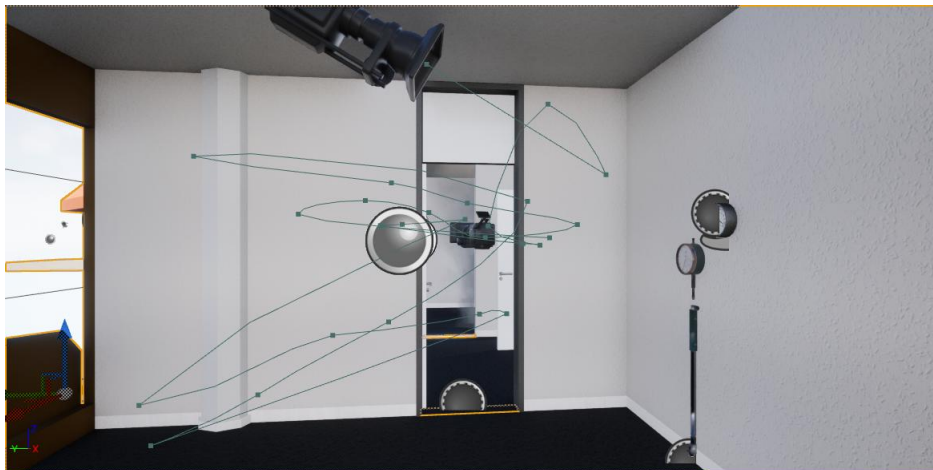


Figure 5.3: Recording of a video sequence for a single gauge.

5.1.4 Summary

The dataset acquisition part was completed and enough images were obtained so that the deep learning models could be trained with it. The realistic dataset consists of a total of 26000 images. This dataset consists of clear images, and smoky images with different

smoke densities. But as explained in Sec. 5.1.2 there is no mapping between a single clear image and a single smoky image as well as not so many variations of gauge images with different levels of smoke. Therefore, to obtain a more diverse dataset, the need of creating synthetic data as explained in Sect. 5.1.3).

Fig. 5.4 shows the ground truth and different levels of smoke for the realistic dataset obtained at the container in Jülich. Besides, Figs. 5.5 and 5.6 show the synthetic haze and synthetic smoke dataset, obtained with the UE. The ground truth (i.e. clear) is presented with the ten different levels of haze/smoke, respectively.

To conclude, three datasets were acquired in total, which are, Realistic dataset (Fig. 5.4; containing both real clear and smoky images of gauges), Synthetic Haze dataset (Fig. 5.5) and Synthetic Smoke dataset (Fig. 5.6).

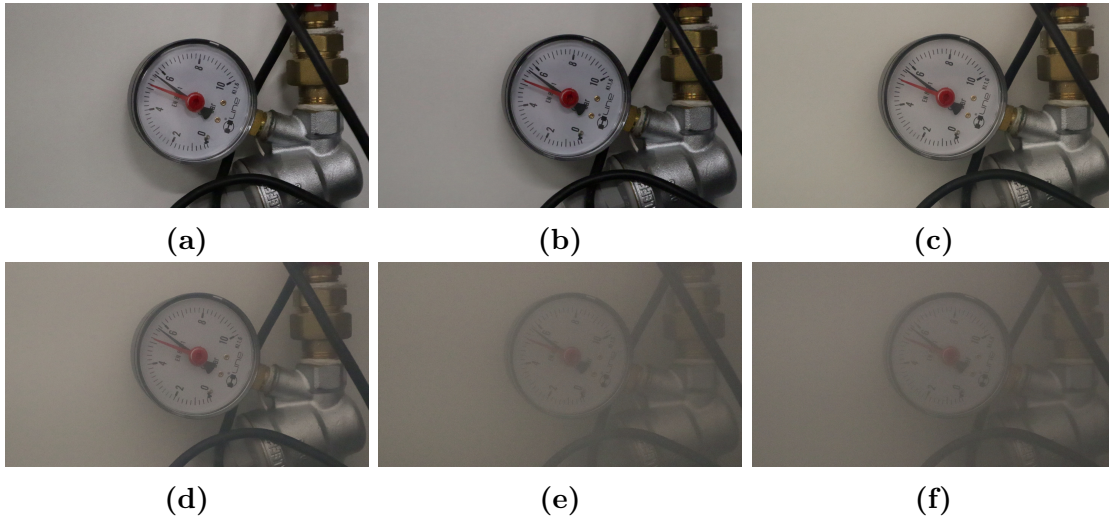


Figure 5.4: Sample Images from the realistic dataset.

5.2 Implementation of Methods

This section provides the implementation details for the three methods, namely, BCCR, FFA Net, and AECR Net. Also, the train, test, and validation sets are defined for the training of the deep learning based models.

5.2.1 BCCR

The libraries used for obtaining the enhanced image through this method was obtained from the git repository provided by Meng et al. [MWD13]. The ground truth image is used to evaluate the method in comparison to other methods using PSNR and SSIM



Figure 5.5: Two examples from the **Synthetic Haze** Dataset

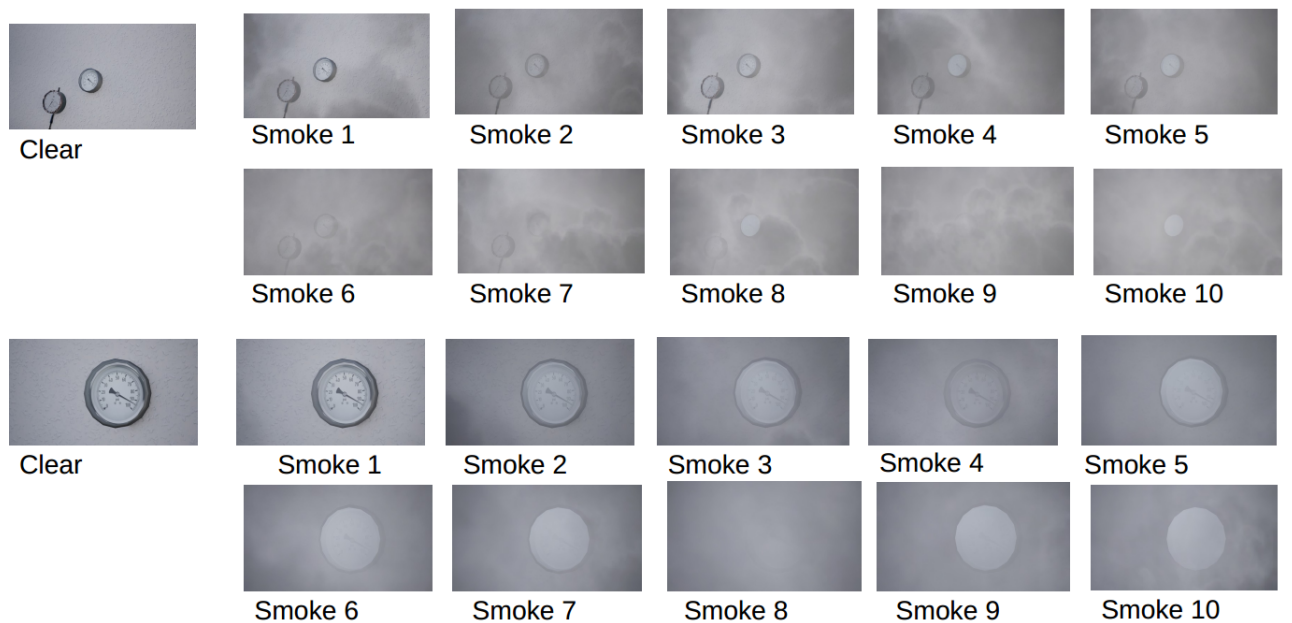


Figure 5.6: Two examples from the **Synthetic Smoke** Dataset

metrics. Default parameters (as in [MWD13]) are used. The parameters are shown in Table 5.1. These parameters are not necessarily optimal for every image. The values of these parameters need to be adjusted based on the characteristics of the input hazy image. These parameters play a crucial role in estimating the transmission and airlight maps from the input image, which are used to remove the haze and produce a clear output image. The results obtained using the BCCR method are shown in Chapter 6.

Parameters	Values
Airlight estimation window size	15
Boundary constraint window size	3
C_0	20
C_1	300
Regularization parameter λ	0.1
σ	0.5
δ	0.85
Show haze transmission map	True

Table 5.1: Parameters used for BCCR method, adopted from [MWD13].

5.2.2 FFA Net and AECR Net

For training the FFA Net and AECR Net, the train, validation, and test split is used as shown in tables 5.2 and 5.3. Two separate datasets, namely haze dataset and smoke dataset are used to train two models each for the FFA Net and AECR Net.

The training (train), validation (val), and testing (test) sets split was considered to be 80/10/10 percent, respectively.

Synthetic Haze Dataset	
Train	
Hazy Images	Clear Images
2180	218
Val	
Hazy Images	Clear Images
440	44
Test	
Hazy Images	Clear Images
440	44

Table 5.2: Total number of Images obtained for the Synthetic Haze dataset.

Synthetic Smoke Dataset	
Train	
Smoky Images	Clear Images
7670	767
Val	
Smoky Images	Clear Images
960	96
Test	
Smoky Images	Clear Images
960	96

Table 5.3: Total number of Images obtained for the Synthetic Smoke dataset.

Dataset preprocessing

Before the data is used for training the models, it needs to be preprocessed. This is a common step used in training of any machine learning algorithm since it helps to improve the performance and generalization ability of a model. For the two networks, FFA Net and AECR Net, some data augmentation and preprocessing transformations have been carried out which are common to both of the networks. Data augmentation involves applying random transformations to the input and target data to increase training data

diversity and reduce overfitting. Random horizontal flipping and rotation are used for this task. Preprocessing transformations, including center cropping, random cropping, and normalization, are applied to simplify the learning problem and make the data more compatible with the model architecture. The input and target data are normalized using mean and standard deviation values from the training datasets to improve model stability and reduce sensitivity to input variations. In Fig. 5.7, a sample of one the input case (out of the ten cases) and the ground truth images for training of the networks is shown. The input images are the images which are contaminated with various densities of smoke and the target images are the corresponding ground truth images.

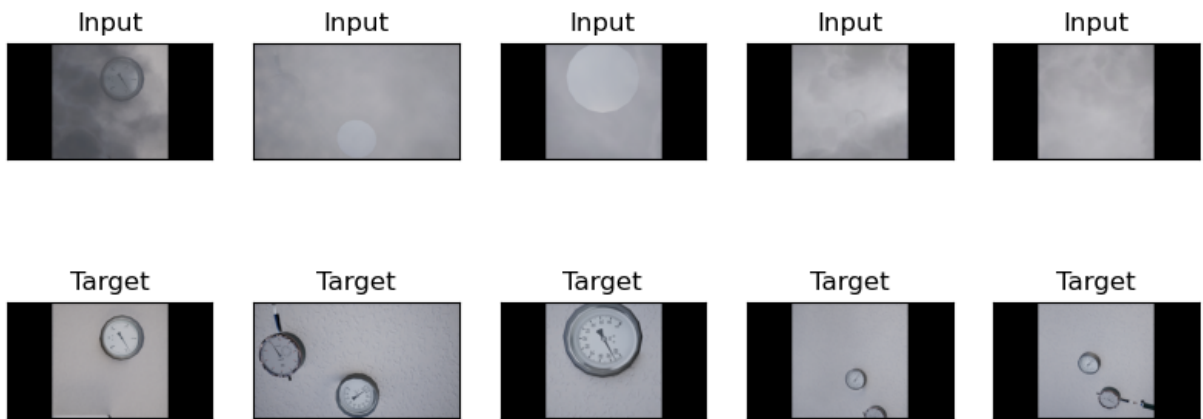


Figure 5.7: Sample input images and ground truth images for FFA Net and AECR Net.

5.2.3 Training and Testing

The code available from [QWB19] were used for training and testing the FFA Net. However, modifications were required to load the custom datasets (synthetic haze and synthetic smoke dataset) into the model. Specifically, changes were made to the code to ensure that the entire JPEG files within the custom datasets could be read correctly during the training process. These changes were necessary to ensure the proper functioning of the model and accurate evaluation of its performance.

The training code for the AECR Net was imported from [WQL21]. However, separate codes were developed to facilitate loading of the custom datasets and to evaluate the model's performance accurately. As the repository did not have any script for testing the performance of the model, then a test script for the AECR Net was also implemented. Note that the AECR Net, as well as FFA Net, are implemented in Python. Therefore, the implementation was carried out with the same open source programming language to be consistent with the networks. These changes were made to ensure compatibility with the unique requirements of the custom dataset and to optimize the training process using

the AECR Net architecture. Table. 5.4 shows the common parameters used for training both FFA Net and AECR Net.

Parameters	Values
Batch size	2
Step size	5000
Epochs	100
Image resolution	1366x768

Table 5.4: Parameters used for training FFA Net and AECR Net methods.

The training of the deep learning-based models, namely the FFA Net and AECR Net, utilized a Nvidia Quadro RTX 8000 GPU. The training process of the FFA Net was completed in approximately five days, while it took approximately 24 hours for the AECR Net. It is worth mentioning that the choice of GPU can significantly affect the training time, depending on its computational power. Additionally, other factors such as the size of the training dataset and the complexity of the network architecture can also influence the training time. Therefore, the training time provided here is specific to the given specifications and may vary in different settings. Though the training times may vary, the results obtained remain the same in different hyperparameter settings.

6 Results and Comparison

This chapter presents the results obtained from the three methods selected, i.e. BCCR, FFA Net, and AECR Net, on the obtained datasets. The following sections present the results with respect to each dataset; namely: realistic dataset (Sect. 6.1; see also Sect. 5.1.2), synthetic Haze dataset (Sect. 6.2), and synthetic Smoke dataset (Sect. 6.3; see also Sect. 5.1.3). The results are discussed quantitatively with respect to the PSNR and SSIM metrics. Finally, the comparison of the methods based on factors such as inference time and memory consumption is presented in Sect. 6.4.

6.1 Realistic Dataset

The results shown in this section are obtained over the realistic dataset for the three different methods. Given the low variety of examples with different levels of smoke only one example is investigated.

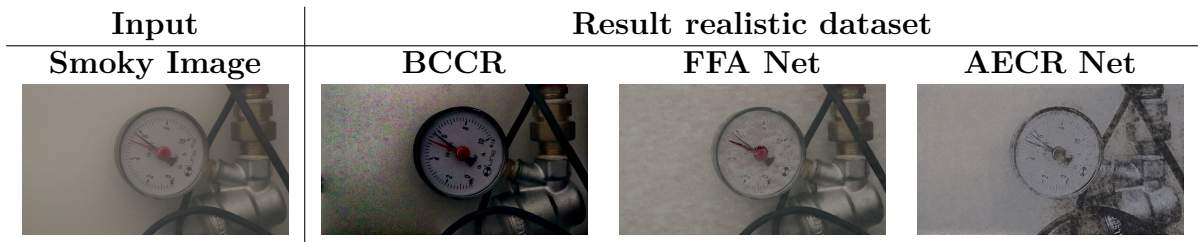


Table 6.1: Results of the three methods tested on a single smoky image from the **Realistic** Dataset.

Table 6.1 presents the results of applying different methods to enhance an input image containing a low level of smoke. The BCCR method produces a significantly clearer and enhanced image, but at the cost of higher levels of contrast and noise. While the output image from the FFA Net method maintains the color levels of the input, it is not as clear as the BCCR method. On the other hand, the AECR Net method produces an image with high levels of noise, and some pixels within the image are missing.

With respect to the metrics PSNR and SSIM, introduced in Sect. 2.3, the BCCR method produced output images with PSNR values ranging from 8 to 12. As for the SSIM, the values ranges from 0.40 to 0.60. The SSIM value ranges between zero (non-similar) and one (very similar; see Sect. 2.3). Note that, the BCCR method obtains a maximum PSNR of 20 dB and SSIM of around 0.70 when tested on the RESIDE dataset(see section 2.4). With respect to the aforementioned metrics, these results imply that the BCCR method

performs on smoky images when compared to the images from the benchmark RESIDE dataset. These metrics provide an objective measure of the similarity between the output images and the original images, with higher values indicating better image quality.

In the realistic dataset, independent of the method, the PSNR and SSIM values were found to perform poorly. The main reason of this behaviour does not rely on the methods but on the dataset itself, given the little variation obtained in the acquisition of the data (see Sect. 5.1.2).

6.2 Synthetic Haze Dataset

The results shown in this section are obtained over synthetic haze dataset. Table 6.2 summarizes the resulting output images obtained with the three different methods by using three input images with dense haze.





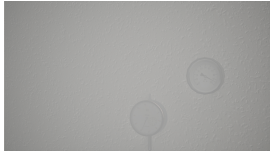



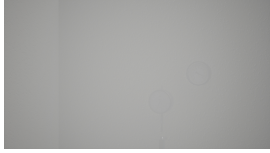



Input	Results Haze		
Hazy Images	BCCR	FFA Net	AECR Net
 (a) Input 1	 (b) BCCR 1	 (c) FFA Net 1	 (d) AECR Net 1
 (e) Input 2	 (f) BCCR 2	 (g) FFA Net 2	 (h) AECR Net 2
 (i) Input 3	 (j) BCCR 3	 (k) FFA Net 3	 (l) AECR Net 3

Table 6.2: Results of the three methods tested on three sample hazy images from **Synthetic Haze Dataset**. **First Column:** three input images with dense haze, **second column:** output images from BCCR method, **third column:** output images from FFA Net, **fourth column:** output images from AECR Net.

Table 6.2 displays the results obtained from different methods on input images containing denser levels of haze compared to the realistic dataset. The input images are denoted by a), e), and i), and the corresponding outputs from the BCCR method are denoted by b), f), and j). The BCCR method is observed to produce clear and enhanced outputs;

however, they display higher contrast and changed colors compared to the input images, making the output images less natural.

Both the FFA Net and the AECR Net methods demonstrate significant haze removal. However, the output images generated by the FFA Net method, denoted by c), g), and k), exhibit some artifacts. On the other hand, the AECR Net method produces the best results in this scenario, with color restoration and reduced artifacts observed in the output images, denoted by d), h), and l).

Overall, the results show that, for this particular dataset, the AECR Net approach removes dense haze from photos with the least amount of artifacts and while maintaining color information. Although delivering outputs that are clear and improved, the BCCR approach may not be appropriate if color preservation is significant. The FFA Net approach can be an alternative, although the produced photos might have artifacts. As for the metrics, Figs. 6.1 shows the results of the SSIM for the FFA Net and AECR Net during training. The maximum SSIM scores are 0.96 and 0.95, respectively. Given the proximity in these values, with respect to this metric, there is no significant difference. For the BCCR method, the average PSNR and SSIM scores were obtained as 12 dB and 0.65 respectively. As for the PSNR of deep learning based models, Fig 6.2 shows the results for the FFA Net and AECR Net during training. The maximum scores achieved are: 30.53 dB and 35.52 dB, respectively, making the AECR Net to stand out in performance with respect to the FFA Net. However, it is important to note that the FFA Net and the AECR Net achieve PSNR scores of 36.39 dB and 37.17 dB respectively when trained on the benchmark RESIDE dataset (See section 2.4). The PSNR scores obtained in this thesis are not as high as the ones obtained on the RESIDE dataset, but nevertheless, are a good measure to evaluate the performance for enhancing gauge images.

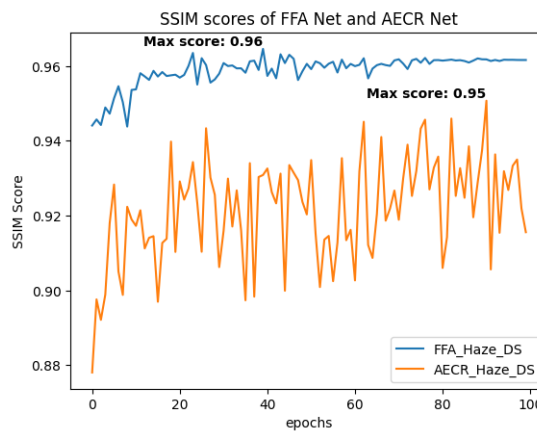


Figure 6.1: SSIM Plots for FFA Net and AECR Net trained on **Synthetic Haze** dataset.

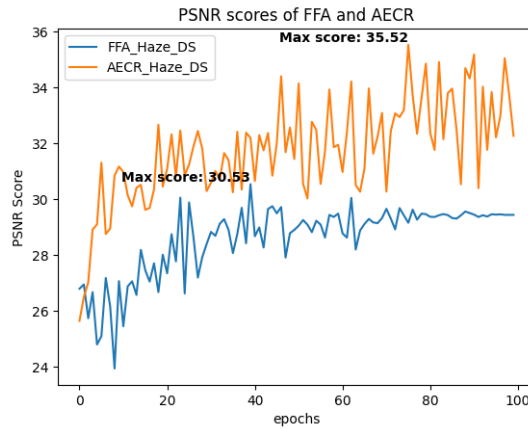


Figure 6.2: PSNR plots for FFA Net and AECR net trained on **Synthetic Haze** dataset.

6.3 Synthetic Smoke Dataset

The objective was to evaluate the performance of three different methods, namely BCCR, FFA Net, and AECR Net, in removing dense smoke from images.

Table 6.3 summarizes the results obtained by applying the three methods to three different input images with varying levels of dense smoke. The input images denoted by a), e), and i) contain heavy smoke, with i) being the densest where the gauge is not visible to the naked eye.

The output images produced by the BCCR method b), f), and j) were not clear and the dense smoke was not completely removed. Moreover, the output images from this method had color discoloration, making them unsuitable for practical post-processing applications. In fact, for the densest smoke, the output image j) amplified the dense smoke from the input image, making the results even worse. On the other hand, the deep learning-based methods, FFA Net and AECR Net, produced much clearer and enhanced output images compared to the prior-based method BCCR. However, the output images from the FFA Net g) and k) were missing some input pixels and still contained some residue of smoke, especially in cases where the input image had very dense smoke. The AECR Net, however, produced the best results when compared to the BCCR method and the FFA Net. Output image l) was able to remove the densest smoke from input image i). Moreover, the output images produced by the AECR Net preserved the colors and contrast levels and minimized artifacts in the output images.

As for the metrics, Figure 6.3 shows the results of the SSIM score, where the AECR Net achieved a maximum score of 0.93, which is higher than the FFA Net's maximum score of 0.84. This indicates that the AECR Net performs better than the FFA Net in terms of







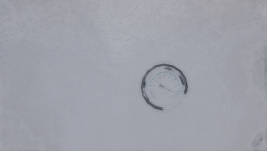





Input	Results Smoke		
Smoky Images	BCCR	FFA Net	AECR Net
			
(a) Input 1	(b) BCCR 1	(c) FFA Net 1	(d) AECR Net 1
			
(e) Input 2	(f) BCCR 2	(g) FFA Net 2	(h) AECR Net 2
			
(i) Input 3	(j) BCCR 3	(k) FFA Net 3	(l) AECR Net 3

Table 6.3: Results of the three methods tested on three sample smoky images from **Synthetic smoke** Dataset. **First Column:** three input images with dense smoke, **second column:** output images from BCCR method, **third column:** output images from FFA Net, **fourth column:** output images from AECR Net.

image structural similarity. As for the PSNR, Fig 6.4 the AECR Net achieved a maximum PSNR score of 26.91 dB, while the FFA Net's maximum PSNR score was 21.72 dB. The average PSNR and SSIM scores for the output images using BCCR method were 9 dB and 0.55 respectively. These results indicate that the AECR Net outperforms the FFA Net in terms of image quality and fidelity, as higher PSNR scores correspond to better image reconstruction accuracy.

Figs. 6.3 and 6.4 show the results of the SSIM and PSNR metrics for the FFA Net and AECR Net for the Smoke dataset during training.

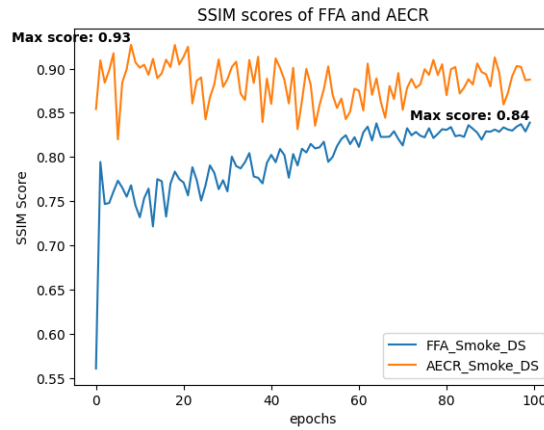


Figure 6.3: SSIM Plots for FFA Net & AECR net trained on **Synthetic Smoke** dataset.

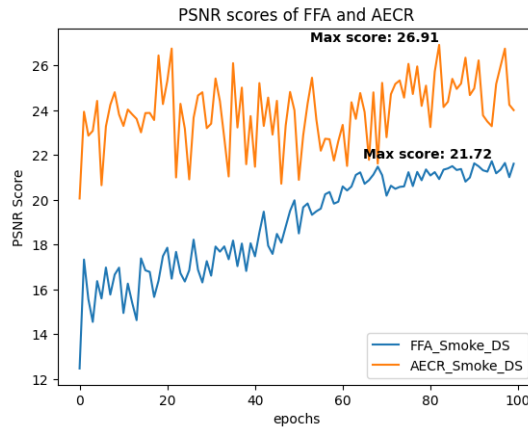


Figure 6.4: PSNR Plots for FFA Net & AECR net trained on **Synthetic Smoke** dataset.

6.4 Factors Comparison

Table 6.4 shows a performance comparison of the three different implemented methods: BCCR, FFA Net, and AECR Net, for six images selected randomly from both synthetic datasets. It includes information on the number of images used for testing, the average time taken for inference on a single image, and the total RAM consumption for each method.

The prior based model, BCCR, consumes lower memory compared to the deep learning approaches FFA Net and AECR Net. The BCCR method is computationally efficient in comparison to other methods, since it relies on usage of mathematical models and use handcrafted features like texture or edges to estimate the transmission map. The deep learning based models, however, use neural networks to learn the mapping between smoky and desmoked images (or hazy and dehazed images), which requires large computational resources.

Table 6.4: Performance comparison of the three methods

Method	No. of images	Avg. Time taken for inference on single image (s)	Total RAM Consumption (MB)
BCCR	6 images with resolution 1024x768	8.71	228.06
FFA Net	6 images with resolution 1024x768	0.989	3018.49
AEER Net	6 images with resolution 1024x768	0.100	2306.39

When comparing deep learning-based methods for image dehazing/desmoking, the FFA Net is observed to require longer inference time and greater RAM consumption compared to the AEER Net. This is primarily due to the increased depth and width of the FFA network architecture, which requires more computation and memory to operate. Specifically, the FFA Net has 57 FA blocks in its network architecture, while the AEER Net has only 6 [WQL21]. These blocks contribute to the increased complexity of the FFA Net, leading to higher computational and memory requirements.

The BCCR method requires only a few general assumptions and can restore a haze-free image to a moderate degree of image enhancement with relatively faithful colors and edge details. However, the inference time is longest when compared to the other two methods which makes this method unfit for practical usage.

7 Discussion and Outlook

This chapter discusses the results of the trained deep learning models and prior based method, their advantages and disadvantages are also described. Additionally, as an outlook, the enhanced images are fed to a gauge reader to check how well the images are enhanced.

7.1 Comparison of the Methods

For the synthetic haze dataset all three methods BCCR, FFA Net, and AEER Net were found to give better results in comparison to the synthetic smoke dataset. With FFA Net and AEER Net, the PSNR scores have increased by 10 dB in case of synthetic haze dataset than compared to the synthetic smoke dataset. The fact that these models were initially developed for the purpose of image dehazing rather than image desmoking explains why all of these techniques successfully perform better on the synthetic haze dataset. Originally, the networks were designed such that they could improve images by removing haze rather than smoke. However, through the work of this thesis, it is shown that the deep learning methods could in fact be adapted to the challenge of image desmoking as well, specifically at the task of desmoking gauge images. Due to the robust representation capabilities of CNNs, the deep learning methods could learn the complex mapping from a smoky gauge image to a clear gauge image. Given that for the distribution of smoke for the smoke dataset was not necessarily homogeneous, making the model obtained more robust.

The results on the synthetic smoke dataset showed that the deep learning techniques delivered excellent qualitative outcomes. The BCCR method, however, does not perform well on images having dense smoke because it works on prior assumptions. For instance, the parameters like airlight estimation could not always be known beforehand in realistic desmoking scenarios. Also, the prior based method BCCR does not have a learning component within it. Considering all these points, it can be concluded that the BCCR method is not appropriate to be used for the challenge of image desmoking.

For the deep learning based methods, although, the PSNR and SSIM scores are not as good as those from the synthetic haze dataset, the output images show that the trained models can remove even dense smoke (see table. 6.3). While being originally designed for picture dehazing, the FFA Net and the AEER Net's architectures were able to learn the thick smoke representations and also produced respectable results for the challenge of image desmoking. The input images in table. 6.3 were selected to be those images

that have densest levels of smoke. If the techniques are effective at enhancing images with dense smoke, it is safe to assume that the trained deep learning models can produce enhanced gauge images with light to even intense smoke levels. Among the deep learning methods, the FFA Net, on average takes 0.8 seconds more for inference when compared to the inference done by AECR Net (see table. 6.4). This is due to the fact that the FFA Net has more convolutional blocks than the AECR Net. To be precise, FFA uses 57 *FA* blocks while the AECR Net makes use of just 6 *FA* blocks [WQL21]. Also, the RAM consumption during inference for the FFA Net is greater than that of the AECR Net which proves that the AECR Net is faster and more compact than the FFA Net.

The increased depth in the FFA Net enables it to learn complex haze representations [QWB19]. However, it comes at the cost of increased inference times and memory consumption. The AECR Net is more compact, but still performs better than the FFA Net in most cases. The reason for this could be attributed to the fact that the AECR Net is an autoencoder based structure. This enables it to learn the complex haze and smoke representations from the gauge image, but within the latent space of the autoencoder network. The latent space serves as a compressed representation which significantly reduces the memory required by the network. The other reason for the superiority of AECR Net over the FFA Net is due to the loss functions that the AECR Net uses. While the FFA Net uses just a L1 loss, the AECR Net uses contrastive learning along with the L1 loss. Additionally, the contrastive loss used by AECR Net makes use of outputs from a fixed pretrained model (VGG 19). More loss functions do not always guarantee better results, however, in this particular case of image desmoking of gauge images, it is evident from results of this thesis that the addition of a contrastive loss produces better results (see Sect. 6.3).

On the other hand, deep learning methods have an advantage over BCCR when it comes to desmoking gauge images corrupted with dense smoke. These methods are also capable of generating equally good results in case of dehazing gauge images. However, to train these models, vast quantities of paired data are required, and the quality of the datasets used during the training significantly affects the quality of the results obtained. Moreover, one of the drawbacks of the deep learning methods is their black box nature. It is not possible to explain in depth what exact components or layers are performing the operations to learn the mapping from a smoky gauge image to a clear gauge image. This lack of interpretability may limit the practical applicability of these methods in certain scenarios.

7.2 Autonomous Reading of Gauges

In order to assess the effectiveness of the trained models in generating clear and enhanced images, a gauge reader application was utilized. This was done to check if the output enhanced images obtained from the selected deep learning methods are actually capable of post-processing applications. One such post-processing application is a gauge reader which reads analogue gauges from their images. The gauge reader [MRE22] used in this case, is an end to end computer vision system that is able to autonomously read analogic gauges with circular shapes and linear scales in unstructured environments. Fig. 7.1 shows the four main stages of the gauge reader application.

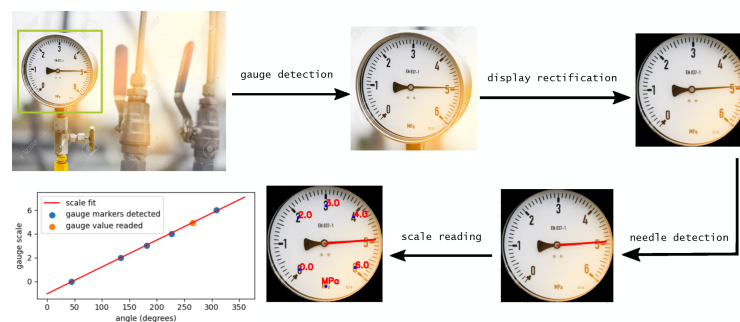


Figure 7.1: Gauge reader algorithm steps: gauge detection from scene, display rectification from a perspective angle, needle detection and scale reading and reconstruction [MRE22].

The gauge reader employs computer vision techniques to accurately read gauge measurements from images. However, it is essential that the gauge images be clear for the application to function optimally. The corrupted images (as shown in Fig. 7.2), when fed to the gauge reader application do not show any results which means that these corrupted images are not capable of being post processed. However, when the post-processed images obtained through the trained models, were fed into the gauge reader application, it was observed that the gauge reader could accurately read the measurements from these enhanced images as shown in Fig. 7.2. This demonstrates that the trained models are capable of generating clear images suitable for post-processing applications.

The potential applications of these models are particularly significant in real-world scenarios where clear gauge images may not be readily available. For instance, in critical environments where contaminated images could lead to life-threatening consequences, the use of these models could prove invaluable. Furthermore, with further improvements and modifications to the FFA Net and AECR Net models, their effectiveness in realistic scenarios could be further enhanced.

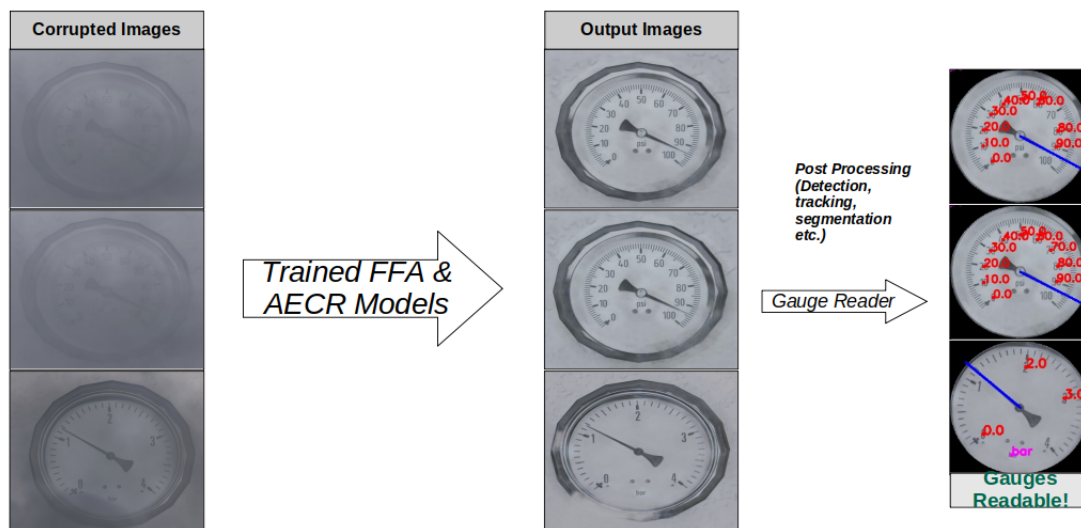


Figure 7.2: **Left:** Corrupted gauge images with dense smoke, **Middle:** Output images obtained through trained models, **Right:** Gauge reader readings on the enhanced images obtained by using the framework of [MRE22].

8 Summary and Future Work

The deep learning based models, FFA Net and AECR Net have successfully been implemented for the specific task of improving/enhancing gauge images which are captured in smoke environments. The acquisition of the datasets of gauge images was most important for training these deep learning models. A more diverse dataset containing real images of different gauges with different levels of smoke could have helped in generating better results. However, for this thesis, monitoring of different levels of smoke was not possible with the current equipment that were available. Therefore, it was decided to use the synthetic datasets where the different levels of smoke could be monitored through the Unreal Engine. As future work, more data could be obtained containing a variety of real gauge images with different gauges and varying levels of smoke, in hopes of producing better results. This would require an update in the sensor equipment of the Institute for the Protection of Terrestrial Infrastructures, DLR.

The atmospheric equation 2.1, which describes the creation of haze or smoke, could be enhanced in order to improve or adapt the prior based approaches, such as the BCCR method, to the problem of image desmoking for gauge images. A more thorough investigation of the atmospheric equation may be beneficial for developing better desmoking techniques. For instance, work can be done in having better approximations of the scene depth $d(x)$ and atmospheric light $A(x)$ parameters to make them more accurate with respect to the physics involved in the model.

Since the field of image dehazing is constantly improving through research, there are some latest works which promise good results. Song, He, Qian and Du [SHQ22] proposes a model called DehazeFormer, which is based on vision transformers. The vision transformers have emerged as a competitive alternative to CNNs [DBK20]. Therefore, it would be interesting to implement this network for the specific task of image desmoking of gauge images and compare how the results change. Also, Song, Zhou, Qian and Du [SZQ22], propose a model called gUnet which is a modified Unet for image dehazing. The Unet architecture was originally invented for biomedical image segmentation. The DehazeFormer and gUnet, reports state of the art PSNR and SSIM scores on the RESIDE dataset. Due to these reasons, it would be appropriate to implement these models for the specific purpose of image desmoking of gauge images using the datasets acquired through the work of this thesis.

Bibliography

- [AAD16] Ancuti, C.; Ancuti, C. O.; De Vleeschouwer, C.
D-HAZY: A dataset to evaluate quantitatively dehazing algorithms
In: 2016 IEEE International Conference on Image Processing (ICIP) (, 2016),
DOI 10.1109/ICIP.2016.7532754, pp. 2226–2230.
- [AAS19] Ancuti, C. O.; Ancuti, C.; Sbert, M.; Timofte, R.
Dense Haze: A benchmark for image dehazing with dense-haze and haze-free images
In: 2019 IEEE International Conference on Image Processing (ICIP) (, 2019),
<http://arxiv.org/abs/1904.02904>.
- [AJ22] Agrawal, S. C.; Jalal, A. S.
A Comprehensive Review on Analysis and Implementation of Recent Image Dehazing Methods
In: Archives of Computational Methods in Engineering, 29 (Nov. 2022) 7,
DOI 10.1007/s11831-022-09755-2, pp. 4799–4850.
- [CTJ18] Chen, L.; Tang, W.; John, N.
Unsupervised Learning of Surgical Smoke Removal from Simulation
In: (June 2018), DOI 10.31256/HSMR2018.38, pp. 75–76.
- [DBK20] Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; Houlsby, N.
An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale
In: CoRR (, 2020), arXiv: 2010.11929, <https://arxiv.org/abs/2010.11929>.
- [EGE18] Engin, D.; Genc, A.; Ekenel, H. K.
Cycle-Dehaze: Enhanced CycleGAN for Single Image Dehazing
In: (2018), DOI 10.1109/CVPRW.2018.00127, pp. 938–9388.
- [Fat08] Fattal, R.
Single Image Dehazing
In: (2008), DOI 10.1145/1399504.1360671, <https://doi.org/10.1145/1399504.1360671>.
- [GCC21] Gui, J.; Cong, X.; Cao, Y.; Ren, W.; Zhang, J.; Zhang, J.; Tao, D.
A Comprehensive Survey on Image Dehazing Based on Deep Learning
In: CoRR (, 2021), arXiv: 2106.03323, <https://arxiv.org/abs/2106.03323>.

- [HST09] He, K.; Sun, J.; Tang, X.
Single image haze removal using dark channel prior
In: IEEE Conference on Computer Vision and Pattern Recognition, Miami, 33 (June 2009), DOI 10.1109/CVPR.2009.5206515, pp. 1956–1963.
- [HW68] Hubel, D. H.; Wiesel, T. N.
Receptive Fields and Functional Architecture of Monkey Striate Cortex
In: Journal of Physiology (London), 195 (1968), <https://www.bibsonomy.org/bibtex/2872c68e0af1147a61f7cfc6ce053bc0e/schaul>, pp. 215–243.
- [JZW17] Ju, M.; Zhang, D.; Wang, X.
Single image dehazing via an improved atmospheric scattering model
In: The Visual Computer, 33 (Dec. 2017) 12, DOI 10.1007/s00371-016-1305-1, pp. 1613–1625.
- [LBB98] Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P.
Gradient-based learning applied to document recognition
In: Proceedings of the IEEE, 86 (1998) 11, DOI 10.1109/5.726791, pp. 2278–2324.
- [LDR20] Li, L.; Dong, Y.; Ren, W.; Pan, J.; Gao, C.; Sang, N.; Yang, M.
Semi-Supervised Image Dehazing
In: IEEE Transactions on Image Processing, 29 (2020), DOI 10.1109/TIP.2019.2952690, pp. 2766–2779.
- [LGL20] Li, B.; Gou, Y.; Liu, J. Z.; Zhu, H.; Zhou, J. T.; Peng, X.
Zero-Shot Image Dehazing
In: IEEE Transactions on Image Processing, 29 (2020), DOI 10.1109/TIP.2020.3016134, pp. 8457–8466.
- [LMS19] Liu, X.; Ma, Y.; Shi, Z.; Chen, J.
GridDehazeNet: Attention-Based Multi-Scale Network for Image Dehazing
In: CoRR (, 2019), arXiv: 1908.03245.
- [LPW17] Li, B.; Peng, X.; Wang, Z.; Xu, J.; Feng, D.
AOD-Net: All-in-One Dehazing Network
In: (2017), DOI 10.1109/ICCV.2017.511, pp. 4780–4788.
- [LRF19] Li, B.; Ren, W.; Fu, D.; Tao, D.; Feng, D.; Zeng, W.; Wang, Z.
Benchmarking Single-Image Dehazing and Beyond
In: IEEE Transactions on Image Processing, 28 (2019) 1, DOI 10.1109/TIP.2018.2867951, pp. 492–505.

-
- [LSK17] Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K. M.
Enhanced Deep Residual Networks for Single Image Super-Resolution
In: CoRR, abs/1707.02921 (2017).
- [LSZ17] Luan, Z.; Shang, Y.; Zhou, X.; Shao, Z.; Guo, G.; Liu, X.
Fast Single Image Dehazing Based on a Regression Model
In: Neurocomput. 245 (July 2017), DOI 10.1016/j.neucom.2017.03.024,
<https://doi.org/10.1016/j.neucom.2017.03.024>, pp. 10–22.
- [MRE22] Milana, E.; Ramírez-Agudelo, O. H.; Estevam Schmiedt, J.
Autonomous Reading of Gauges in Unstructured Environments
In: Sensors, 22 (2022) 17, DOI 10.3390/s22176681.
- [MWD13] Meng, G.; Wang, Y.; Duan, J.; Xiang, S.; Pan, C.
Efficient Image Dehazing with Boundary Constraint and Contextual Regularization
In: (Dec. 2013).
- [NN03] Narasimhan, S.; Nayar, S.
Contrast restoration of weather degraded images
In: IEEE Transactions on Pattern Analysis and Machine Intelligence, 25 (2003) 6, DOI 10.1109/TPAMI.2003.1201821, pp. 713–724.
- [PBV22] Pan, Y.; Bano, S.; Vasconcelos, F.; Park, H.; Jeong, T. T.; Stoyanov, D.
DeSmoke-LAP: improved unpaired image-to-image translation for desmoking in laparoscopic surgery
In: International Journal of Computer Assisted Radiology and Surgery, 17 (May 2022) 5, DOI 10.1007/s11548-022-02595-2, pp. 885–893.
- [PCC19] Pei, S.-C.; Chen, Y.-A.; Chiu, Y.-H.
Single image desmoking using haze image model and human visual system
In: Journal of Electronic Imaging, 28 (July 2019), DOI 10.1117/1.JEI.28.4.043007, p. 043007.
- [QWB19] Qin, X.; Wang, Z.; Bai, Y.; Xie, X.; Jia, H.
FFA-Net: Feature Fusion Attention Network for Single Image Dehazing
In: CoRR (, 2019), arXiv: 1911.07559.
- [Sar21] Sarker, I. H.
Machine Learning: Algorithms, Real-World Applications and Research Directions
In: SN Computer Science, 2 (Mar. 2021) 3, DOI 10.1007/s42979-021-00592-x, p. 160.

- [Sha20] Shaziya, H.
A Study of the Optimization Algorithms in Deep Learning
In: (Mar. 2020), DOI 10.1109/ICISC44355.2019.9036442.
- [SHK12] Silberman, N.; Hoiem, D.; Kohli, P.; Fergus, R.
Indoor Segmentation and Support Inference from RGBD Images
In: ECCV, 7576 (Oct. 2012), DOI 10.1007/978-3-642-33715-4_54, pp. 746–760.
- [SHQ22] Song, Y.; He, Z.; Qian, H.; Du, X.
Vision Transformers for Single Image Dehazing
DOI 10.48550/ARXIV.2204.03883.
- [SS02] Scharstein, D.; Szeliski, R.
A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms
In: International Journal of Computer Vision, 47 (Apr. 2002) 1, DOI 10.1023/A:1014573219977, pp. 7–42.
- [SWD05] Sharma, G.; Wu, W.; Dalal, E.
The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations
In: Color Research & Application, 30 (Feb. 2005), DOI 10.1002/col.20070, pp. 21–30.
- [SYL14] Shao, L.; Yan, R.; Li, X.; Liu, Y.
From Heuristic Optimization to Dictionary Learning: A Review and Comprehensive Comparison of Image Denoising Algorithms
In: IEEE Transactions on Cybernetics, 44 (2014) 7, DOI 10.1109/TCYB.2013.2278548, pp. 1001–1013.
- [SZQ22] Song, Y.; Zhou, Y.; Qian, H.; Du, X.
Rethinking Performance Gains in Image Dehazing Networks
DOI 10.48550/ARXIV.2209.11448.
- [Tan08] Tan, R. T.
Visibility in bad weather from a single image
In: 2008 IEEE Conference on Computer Vision and Pattern Recognition (, Feb. 2008), DOI 10.1109/CVPR.2008.4587643, pp. 1–8.
- [VA13] Verma, R.; Ali, J.
A comparative study of various types of image noise and efficient noise removal techniques

-
- In: International Journal of Advanced Research in Computer Science and Software Engineering, 3 (Jan. 2013), pp. 617–622.
- [VSP17] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I.
Attention Is All You Need
In: CoRR (, 2017), arXiv: 1706.03762.
- [VVL20] Vishal, V.; Venkatesh, V.; Lochan, K.; Sharma, N.; Singh, M.
Unsupervised Desmoking of Laparoscopy Images Using Multi-Scale Desmo-keNet
In: (2020), DOI 10.1007/978-3-030-40605-9_36, pp. 421–432.
- [WBS04] Wang, Z.; Bovik, A.; Sheikh, H.; Simoncelli, E.
Image quality assessment: from error visibility to structural similarity
In: IEEE Transactions on Image Processing, 13 (2004) 4, DOI 10.1109/TIP.2003.819861, pp. 600–612.
- [WQL21] Wu, H.; Qu, Y.; Lin, S.; Zhou, J.; Qiao, R.; Zhang, Z.; Xie, Y.; Ma, L.
Contrastive Learning for Compact Single Image Dehazing
In: CoRR (, 2021), arXiv: 2104.09367.
- [YND18] Yamashita, R.; Nishio, M.; Do, R. K. G.; Togashi, K.
Convolutional neural networks: an overview and application in radiology
In: Insights into Imaging, 9 (Aug. 2018) 4, DOI 10.1007/s13244-018-0639-9, pp. 611–629.
- [ZDS17] Zhang, Y.; Ding, L.; Sharma, G.
HazeRD: An outdoor scene dataset and benchmark for single image dehazing
In: (2017), DOI 10.1109/ICIP.2017.8296874, pp. 3205–3209.
- [ZGF17] Zhao, H.; Gallo, O.; Frosio, I.; Kautz, J.
Loss Functions for Image Restoration With Neural Networks
In: IEEE Transactions on Computational Imaging, 3 (2017) 1, DOI 10.1109/TCI.2016.2644865, pp. 47–57.
- [Zho06] Zhou Wang, A. C. B.
Modern image quality assessment
In: (2006), <https://doi.org/10.1007/978-3-031-02238-8>.
- [ZLL18] Zhang, Y.; Li, K.; Li, K.; Wang, L.; Zhong, B.; Fu, Y.
Image Super-Resolution Using Very Deep Residual Channel Attention Networks
In: CoRR (, 2018), arXiv: 1807.02758.

List of Tables

5.1	Parameters used for BCCR method, adopted from [MWD13].	48
5.2	Total number of Images obtained for the Synthetic Haze dataset.	48
5.3	Total number of Images obtained for the Synthetic Smoke dataset.	48
5.4	Parameters used for training FFA Net and AECR Net methods.	50
6.1	Results of the three methods tested on a single smoky image from the Realistic Dataset.	51
6.2	Results of the three methods tested on three sample hazy images from Synthetic Haze Dataset. First Column: three input images with dense haze, second column: output images from BCCR method, third column: output images from FFA Net, fourth column: output images from AECR Net.	52
6.3	Results of the three methods tested on three sample smoky images from Synthetic smoke Dataset. First Column: three input images with dense smoke, second column: output images from BCCR method, third column: output images from FFA Net, fourth column: output images from AECR Net.	55
6.4	Performance comparison of the three methods	57

List of Figures

2.1	Left: input image, middle: output image obtained, right: depth image obtained by using Fattal’s approach [Fat08].	4
2.2	Top row: input images, Middle row: direct attenuation, Bottom row: airlight estimation. [Tan08].	5
2.3	Haze removal using single image. Left: input image, middle: output image, right: estimated depth map [HST09].	6
2.4	Physical model of hazy image formation [AJ22].	7
2.5	Radiance cube and boundary constraint [MWD13].	8
2.6	Feature Attention Module [QWB19].	11
2.7	Equations of FA module [QWB19]).	11
2.8	Basic block for FFA Net (Fig. 6 from [QWB19]).	12
2.9	FFA Architecture [QWB19].	12
2.10	Hazy and clean images [EGE18].	13
2.11	Architecture of AECR Net [WQL21].	14
2.12	Adaptive Mixup Operation [WQL21].	15
2.13	difference between rigid and deformable Kernels [WQL21].	15
2.14	Contrastive Regularization: Anchor image pulled towards the Positive image and pushed away from the Negative image [WQL21].	17
2.15	Sample images of Fattal’s Dataset [Fat08].	21
2.16	Sample images from D-Hazy dataset [AAD16].	22
2.17	HazeRD samples from left to right, a Haze-free image, b depth map, simulated hazy images with the visual range of c 50 m, d 100 m, e 200 m, and f 500 m, respectively [ZDS17].	22
2.18	Three examples of the Dense-Haze dataset that provides 33 pairs of hazy and corresponding haze-free (groundtruth) outdoor images [AAS19].	23
2.19	Sample images from RESIDE dataset [LRF19].	24
3.1	RGB image of a gauge.	27
3.2	Binary output image of the gauge.	27
3.3	Output greyscale image of the image.	27
3.4	Splitting of the RGB gauge image into its RGB channels.	29
3.5	General framework of GANs.	30
3.6	Left: images of gauges: clear (top-panel), haze (middle-panel), and with noise (bottom-panel). Right: histograms output of the respective gauge image on the left with two degradations, haze and noise.	31

4.1	Relation between AI, ML, and DL [Sar21].	33
4.2	A convolution operation with zero padding so as to retain in-plane dimensions. Note that an input dimension of 5×5 is kept in the output feature map. In this example, a kernel size and a stride are set as 3×3 and 1, respectively [YND18].	35
4.3	Parameters and hyperparameters in a CNN [YND18].	36
4.4	Gradient descent [YND18].	38
5.1	Components used in Realistic data acquisition.	43
5.2	A snapshot of the virtual environment KAS in UE.	44
5.3	Recording of a video sequence for a single gauge.	45
5.4	Sample Images from the realistic dataset.	46
5.5	Two examples from the Synthetic Haze Dataset	47
5.6	Two examples from the Synthetic Smoke Dataset	47
5.7	Sample input images and ground truth images for FFA Net and AECR Net.	49
6.1	SSIM Plots for FFA Net and AECR Net trained on Synthetic Haze dataset.	53
6.2	PSNR plots for FFA Net and AECR net trained on Synthetic Haze dataset.	54
6.3	SSIM Plots for FFA Net & AECR net trained on Synthetic Smoke dataset.	56
6.4	PSNR Plots for FFA Net & AECR net trained on Synthetic Smoke dataset.	56
7.1	Gauge reader algorithm steps: gauge detection from scene, display rectification from a perspective angle, needle detection and scale reading and reconstruction [MRE22].	61
7.2	Left: Corrupted gauge images with dense smoke, Middle: Output images obtained through trained models, Right: Gauge reader readings on the enhanced images obtained by using the framework of [MRE22].	62