

# Single image dehazing using deep neural networks

Cameron Hodges<sup>a,\*</sup>, Mohammed Bennamoun<sup>a</sup>, Hossein Rahmani<sup>b</sup>

<sup>a</sup>Department of Computer Science and Software Engineering, University of Western Australia, 35 Stirling Hwy, Crawley WA, Australia

<sup>b</sup>School of Computing and Communications, Lancaster University, Bailrigg, Lancaster, UK

## 1. abstract

The rapid growth in computer vision applications that are affected by environmental conditions challenge the limitations of existing techniques. This is driving the development of new deep learning based vision techniques that are robust to environmental noise and interference. We propose a novel deep CNN model, which is trained from unmatched images for the purpose of image dehazing. This solution is enabled by the concept of the Siamese network architecture. Using object performance measures of image PSNR and SSIM we are able to demonstrate a quantitative and qualitative improvement in the network dehazing performance. This superior performance is achieved with significantly smaller training datasets than existing methods.

## 2. Introduction

Computer vision is fast becoming a ubiquitous technology serving many areas of modern technology. While current applications have benefited from the availability of high quality imagery, industries such as driverless vehicles and outdoor surveillance which adopt computer vision require systems that deal with poor quality data and sensor degradations due to environmental conditions are becoming mandatory. The current literature on single image dehazing relies on an atmospheric scatter model pro-



Figure 1: Examples of a Self Driving Vehicle Application, Original (top) and Dehazed images using our method (bottom)

posed by [15] in order to determine how the observed light from an image subject is attenuated by atmospheric conditions. The model can be summarised by Equation 1:

$$I = J.t + A(1 - t) \quad (1)$$

In this model the observed scene, represented as  $I$ , is composed of two elements. The clear image  $J$  that has been attenuated by aerosols in the air; the attenuating effect is described by the transmission map  $t$ . In addition to this, there is the light in the scene that is scattered, this is described by  $A$  which is known as the global atmospheric light multiplied by  $(1 - t)$ . In areas of the image where haze is heavy, the transmission map is close to 0. This almost entirely attenuates the clear image  $J$  and the scattered light  $A(1 - t)$  is approximately equal to  $A$ , which returns almost no information about the clear image.

While this process is simple in concept, its implementation using single images is very problematic. The major challenge is to estimate the transmission map from the hazy image. Previously, this was through conventional computer vision methods such as dark channel [12] or other manual heuristics [5]. While those methods have

\*Corresponding author: Tel.: +0-000-000-0000; fax: +0-000-000-0000;

Email address: Cameron.Hodges@Research.uwa.edu.au  
(Cameron Hodges)

demonstrated to be effective, due to the non-linear complex nature of haze it is difficult to make further improvements with conventional techniques alone. Therefore, newer research is turning to machine/deep learning techniques. Current machine learning (ML) based methods can be considered in two broad categories.

The majority rely on having matched training data, including images of both clear and hazy views of the exact same scene [4] [24] [32] [34]. This is a significant limitation as this type of dataset is difficult to obtain. This makes it a challenge to create a model that generalises to many situations. While there is an emerging body of work considering methods that do not rely on matched data [29] [9], the current implementations of this approach are still significantly limited by slow dehazing speeds and difficulty in recovering fine details.

Considering these challenges we have proposed a new approach which fits this later category and is intended to create an independent measure of haze in an image in order to drive the machine learning process. A consistent challenge of machine learning is to create suitable training data sets. In light of this challenge, the improved characteristic of our method is the use of unmatched training images, which instead of attempting to solve directly this challenge we circumvent it entirely. We have based our method on a two-network structure, a dehazing network and a discriminator network. This enables a novel fine tuning process, which is a particularly important contribution as it allows our method to achieve results comparable or exceeding the state-of-the-art performance with a significantly smaller data set. Our method is trained using a set of 5,968 foggy images and 5,968 clear images, in comparison to AOD-Net [19] which uses a training set of 27,256 images or in the case of [29] which uses 1,000,000 image patches.

### 3. Related Work

This work builds on a number of core techniques from existing computer vision and ML literature. This includes Siamese network architectures and Convolutional Neural Networks.

The fundamental concept underpinning Siamese Networks was first demonstrated in [3] for the problem of customer signature verification. In this work, a customer's signature is fed to a network along with a reference signature to determine whether it is genuine or a forgery.

With the proposed neural network architecture composed of two identical networks that are linked by a shared set of weights, the authors in [3] demonstrate a methodology for extracting learned features from input data and outputting a metric of similarity between the two signatures. In this case, the cosine between the extracted feature vectors was used as the similarity metric. The true value of this technique is in its ability to determine the similarity between two unlike samples based on learned features, without having to classify these into discrete classes. Since then, this property of Siamese networks has been utilised to approach verification problems including signature verification [3], pedestrian re-identification [8], and data dimension reduction [10].

In the field of deep learning, Convolutional Neural Networks (CNN) have been of great interest for decades now. While the fundamental techniques, as laid out in [18] and [17] have existed since the 1980's, it has only recently entered broad use. This is driven by the arrival of cost effective GPUs, methods to support training of deeper network architectures and large datasets that the real value of this ML technique has been realised. The majority of the current research can be traced to [16] and be seen as the turning point for this area of work.

In the specific area of single image dehazing, research methods have predominately focused on analytical methods to dehaze and filter images. A method known as Dark Channel Prior has become a foundation of the current accepted methods [12] [26] to resolve a transmission map  $t$  of haze within a given image. These methods have proven to be an effective starting point but have several limitations. High contrast surfaces such as white objects in the scene or a bright sky are seen as haze by the DCP method. For example, [6] attempt to improve the performance of these methods with an optimised recovery of  $J$  to achieve a more accurate result for high textured image patches. The work in [2] demonstrates an alternative analytical approach in which the colour relationships between clusters pixels are used to resolve the original unhazed image.

The works in [4] [32] [19] [33] demonstrate that deep learning could be an effective platform for the development of dehazing techniques. The hierarchical nature of the CNN architecture allows for the extraction of haze relevant features. This provided a number of key improvements. Deep learning methods proved a more versatile solution in a wide variety of image types, scene distances

and lightening, as well as an improvement in processing speeds. Additionally in the case of [33], Deep Learning also allows the inclusion of priors. For this reason, they have become the accepted standard for all new dehazing methods. Work in [4] demonstrated a method for using a CNN based architecture with a supervised training approach. It uses the CNN to generate transmission map values for image patches taken directly from the hazy image. Our work uses this as a building block to create an end-to-end solution that allows fine tuning of the model with a relatively small training dataset. Our discriminator follows a similar structure as WaterGan [21], which uses a Siamese network based discriminator to generate realistic synthetic training data for a training pipeline. Although we have a significant distinction. In our proposed method, we use the output of the Siamese network (discriminator) as a direct training signal during the dehazing process, which consequently increases the performance of the dehazing network in a single process. This is a departure from previous work, which have used the discriminator output to effectively exclude or keep already generated images to form a dataset for a later training process. We are broadly differentiated from Generative Adversarial Networks such as [22] and [9] as we are attempting to use a determinative model [15] to recover pixel based information from the source image as opposed to generating new data from learned parameters.

In addition to removal of as much haze as possible from images, we also aim to maintain two other critical characteristics including process speed and retention of fine detail. When comparing our method to Generative methods such as [9] while in some cases the GAN based method removes more haze, it is not able to return fine detail that an atmospheric scatter model based method can. This can be seen in Figure 2.

Work shown by [29] demonstrates an alternative method for training with unmatched data. The authors propose a new network layer for sorting convolutional layers, known as a ranking layer. During training, the network is then trained initially as in image classifier to allow for the recovery of image detail without matched clear images. While this method shows promise, it is limited in its processing speed. We have aimed to produce a network with high speed performance. For this we have also chosen to use all convolutional layers and simple patched approach to create the full image transmission map. This is

significantly quicker to dehaze the complete image when compared to the Random-Forest Regression method used in [29]. Although at this time source code for [29] is not available for direct comparison.



Figure 2: Comparison of fine detailed returned in the dehazing process. Left: Our Proposed Model; Right: CycleDehaze [9]. Our model is able to dehaze the source image while maintaining color accuracy and fine detail.

#### 4. Proposed Solution

To address the limitations of the current literature, we create a learning based architecture that is built on two separate networks: a dehazing network and a discriminator network. The function of this structure is to create an objective measure of the performance of the dehazing model, which can be used to fine tune its weights and improve performance. This approach of training a neural network without matching ground truth data is considered as the main contribution of this work. The dehazing network takes an input image of 380x620 pixels and first reshapes it into 589 patches (20x20x3) and estimates the transmission map (single value) for each patch. This transmission map and the estimated atmospheric light value (constant for the entire image) are then used to solve for  $J$  in Equation 1. Each dehazed patch is then used to reform the full image to be fed to the discriminator network. The architecture of our dehazing network is based on the architecture in [4]. This network is initially trained as a supervised model using synthetic data to produce coarse weights.

The second network, which forms the architecture, is a discriminator network. Due to the lack of hazy image data with matching clear ground truth images, there is a need for an approach which allows for the detection and quantification of haze in an image that is independent of the image scene. In order to achieve this invariant determination, we are proposing a generalised network design shown in Figure 3. This structure is intended to allow the network to learn to differentiate between images based on whether they are hazy or clear even if their content is

not matching. The two sub networks (CNN1 and CNN2) in Figure 3 are of an identical structure and share common weights and biases. When fed with image pairs of  $380 \times 620 \times 3$ , this allows for the extraction and similarity comparison of only the features that describe the haze in the image. Later, when paired with the dehazing network, this allows the deep network optimiser to determine how well an image was dehazed, and therefore fine tune to improve the performance of the dehazing network.

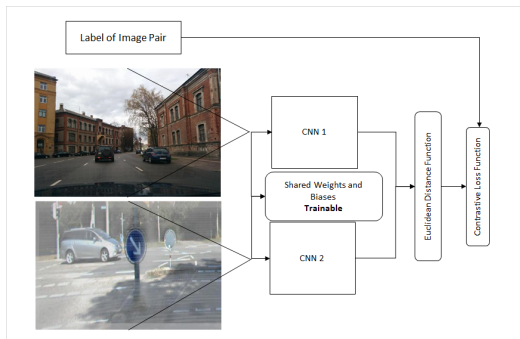


Figure 3: Discriminator block diagram of the training configuration. Each training pair consists of two complete images and a label based on logic in Section 5

Once the dehazing network and the discriminator network are trained individually using supervised datasets, they are combined and fine tuned to improve the performance of the dehazing network. Pairs of images from different scenes are fed into the network in the following manner. A hazy image is fed to the dehazing network, while a known clean image is fed to one of the inputs of the discriminator network. This can be seen in Figure 4.

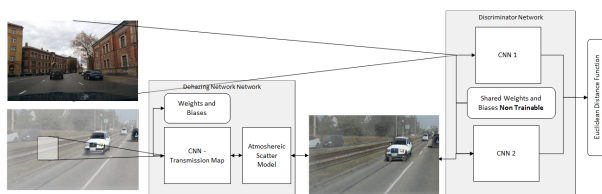


Figure 4: Complete network block diagram of the training configuration. Each hazy image is reshaped into 589 patches, the patches are then dehazed before being reformed into a complete image. This dehazed complete image is then sent to the discriminator with an unmatched clear image.

The intention here is for the discriminator network to

measure how similar the dehazing network output is to a known clean image, with respect to the haze relevant features. If the dehazing network performs well and produces a dehazed image, then the output of the discriminator will be very low. This methodology allows a training improvement of the dehazing network to be driven by the output of the network loss curve.

## 5. Datasets

As with all ML based research, finding suitable datasets is an immense challenge. In order to overcome this, we have drawn data from a variety of sources. The breakdown of our datasets is from the following sources: 1) Google Image search (Key words: Haze, Fog, Smog, city streets); 2) Kitti Driving Dataset (Including Matching Lidar Scans) [30]; 3) Mapillary Academic Dataset [27]; 4) Middlebury Stereo Image Dataset [28]; 5) Dashboard Camera Footage of Foggy Driving; 6) RESIDE hazy data set [20]; 6) NTIRE 2018 Challenge Dataset [1].

In order to create a dataset for training, validation, and testing, we needed to collect a large number of images. Considering the applications of the technology, we focused on collecting images that are typical of a self driving vehicle application that were both clear and hazy. Unfortunately foggy/hazy images taken from vehicle mounted cameras are difficult to find. Therefore, we have supplemented the dataset with a large number of synthetically produced images. The datasets were used in the following way.

### 5.1. Dataset for Pre-training the Dehazing network

22 sample depth maps were taken from the Middlebury Stereo Image Dataset [28] and scaled to  $600 \times 600$  pixels. They were then broken into 19,200  $20 \times 20$  pixel depth patches and applied to city street scape type images from Google searches to produce 148,500 unique  $20 \times 20 \times 3$  hazy image patches. These patches were then divided up on a 60-20-20 basis to produce training, validation and testing sets of 89,100, 29,700 and 29,700 patches, respectively. Each patch was given a label of a single value equal to the mean transmission map value derived from the depth map that was used to create it.

### 5.2. Dataset for pre-training the Discriminator network

The dataset for pre-training was built up based on a variety of data sources. In total 5,968 clear images have been collected from the following sources: 1) 763 images from Google searches (keywords City streets, clear weather city streets); 2) 5,205 images from Mapillary Self Driving Dataset [27]. The hazy images (5,968 images in total) have been collected from the following sources: 1) 971 images from Google Searches (Keywords Fog, Smog, City Haze); 2) 4,400 synthetic haze images created with clean images from the Kitti dataset (none of the base clean images appear in the clean dataset) [30]; 3) 597 images taken from foggy driving YouTube videos. These images are randomised and then pairs of images are drawn from them. The label for each pair is based on whether the image classes match (hazy or clear). The logic for the label of pairs is as follows:

Image Pair Combination	Label
Hazy - Hazy	1
Hazy - Clear	0
Clear - Hazy	0
Clear - Clear	1

An important characteristic of this is that the pairs are of unrelated scenes and randomised, the only factor considered is whether the image is considered hazy or clear.

### 5.3. Dataset for Testing and Performance Assessment

Our final testing was conducted using three separate datasets. The RESIDE/SOTS dataset [20] containing 500 outdoor synthetic fog images and the NTIRE 2018 Challenge Dataset [1] containing 5 images, provide an ideal dataset for the final testing. The inclusion of matching clear and hazy images allows for a direct quantitative analysis of the dehazing performance. Following this, we used the RESIDE/RTTS dataset which include real world hazy images with object annotations for testing the effects of haze on object detection models.

## 6. Implementation

The network was implemented in Tensorflow [25] with GPU support and trained using a Nvidia Geforce 1080 Ti GPU. The complete network can be considered in its dehazing network and discriminator network components.

### 6.1. Dehazing Network Architecture

The CNN implementation for the dehazing network must extract haze relevant features from the input images. This has to be in a manner that is invariant to the subject matter and then map these features to a transmission map value. The work in [4] is particularly relevant for this application. The use of parallel Convolutional layers of different filter sizes is ideal for the extraction of features that appear at varying scales in the input image. This is a unique requirement of hazy image processing. The network design takes the images patches and feeds them through a number of Convolutional layers to extract haze relevant features. The network’s distinctive use of parallel Convolutional layers is ideally suited to extract features that can appear on a variety of scales within the target image. In addition to this, the use of an activation layer called BReLU proposed by the authors of [4] creates a more consistent output of transmission values. This layer is built on the common ReLU activation with an additional upper limit which truncates the output to a range of 0 to 1. The weights were randomly initialised with a  $\mu = 0$  and  $\sigma = 0.1$ , and the bias values were set to 0.

### 6.2. Discriminator Network Architecture

The implementation of the discriminator network is based around two identical CNNs, which share a common set of weights and biases. The detailed structure of each CNN is similar in layer structure to [21]. This structure outputs a feature vector of size  $64 \times 1$  from each side of the Siamese network. These feature vectors are then compared using a similarity module. For this work, we have chosen a Euclidean distance measure. While previous work e.g., [8] has shown that more complex approaches, such as a second order similarity function, can perform better, they on the other hand introduce additional complexity. These methods require an additional set of parameters which need to be learned. In contrast, the Euclidean distance is simple to implement and computationally efficient. The weights of all layers were randomly initialised with a  $\mu = 0$  and  $\sigma = 0.03$ , and the biases values were set to 0. The similarity function of two feature vectors  $\mathbf{x}_1$  and  $\mathbf{x}_2$  is given in Equation 2 where vectors the are of size  $64 \times 1$ .

$$s(\mathbf{x}_1, \mathbf{x}_2) = \sqrt{\frac{1}{64} \sum [\mathbf{x}_1 - \mathbf{x}_2]^2} \quad (2)$$

### 6.3. Complete Network Architecture

Once the two networks (Sec. 6.1 and Sec. 6.2) are successfully implemented, they are combined to form a complete end-to-end architecture as shown in Figure 4. In order to improve the efficiency of the batch training, the dehazing network was repeated to form 32 parallel networks which allow the training of batches of 32 images. This can be seen in Figure 5. Initial testing found that as the entire image is dehazed in patches of  $20 \times 20$  pixels, it resulted in an artifacted appearance in the final image. To resolve this, we applied a guided filter [11] to the transmission map before it is used to resolve  $J$ .

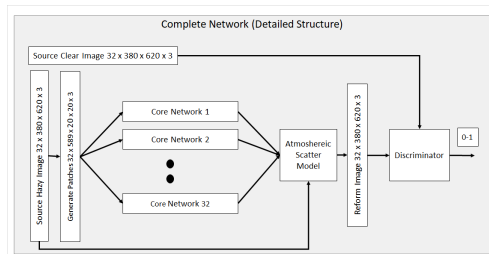


Figure 5: 32 parallel networks allow for batch training during end-to-end training, each network processes 589 patches.

## 7. Training Methodology and Performance

As per the model implementation, the training of the model is carried out in three distinct phases: Dehazing network pre-training (Sec. 7.1), Discriminator network pre-training (Sec. 7.2), and fine tuning of the complete network (Sec. 7.3).

### 7.1. Dehazing Network Pre-training

In order to pre train the dehazing network, we use the supervised dataset described in Section 5.  $20 \times 20 \times 3$  pixel patches are fed into the network in batches of 32 at a time. A mean squared error cost function and Adam Optimizer [14] are used to train the network. For each training example the input patch is fed into the CNN which outputs a single value for the predicted transmission map value, this is compared to the known value for that example. Once trained, the network weights are suitable for determining the transmission map value for the given image patch to an accuracy comparable with the existing literature [4].

This provides a suitable starting point to move to the fine tuning using the complete network architecture.

### 7.2. Discriminator Network Pre-training

In order to facilitate the fine tuning of the dehazing network, the discriminator network needs to be trained to allow it to assess complete images for their level of haze. This was achieved through a supervised learning process. Training images of size  $380 \times 620 \times 3$  from unrelated scenes are fed to the network in pairs with a label of either 1 or 0, specifying if the pairs are matching or contrasting, respectively. We have used a contrastive loss with a spring function as (described in [10]) and shown in Equation 3. The model weights are optimised using the Adam Optimizer [14] with batches of 128 pairs at a time.

$$loss = \frac{1}{128} \sum y \times y_{pred}^2 + (1 - y) \times \max(0, (1 - y_{pred}))^2 \quad (3)$$

### 7.3. Fine Tuning of the Complete Network

Once the dehazing network has reached the performance benchmarks set by the existing literature, the training process moves to a fine tuning stage. For this stage we attempted two alternative training methods. **The first training method** focuses on increasing the contrast between training images. 3000 images pairs are fed into the network where both images in the pair are hazy. As the dehazing network dehazes one of the images in the pair, the discriminator loss should increase to indicate that haze in the image is being removed and the contrast between the dehazed image and the original hazy image is increasing. For **the second training method**, 9920 training pairs are randomly drawn from the discriminator dataset (listed in Section 5), where the first image in each pair is known to be a hazy image and the second image is a known clear image. In this case as the dehazing network removes haze from the known hazy image the output will become closer to the known clear image. Therefore the discriminator output will decrease.

In both training methods the weights learned during the pre-training process are loaded as a starting point. The network is then trained on these pairs in batches of 32 pairs at a time. Once these pre-trained weights are loaded, the discriminator weights and the first Convolutional layer of the dehazing network are fixed, leaving

only the remaining dehazing weights to be further updated. An Adam Optimizer [14] is used to reduce the mean output of the discriminator for each batch by adjusting the dehazing network weights through back propagation.

## 8. Results

In order to demonstrate the proof of our concept, we focused on demonstrating the success of each sub-network, as well as the end-to-end architecture. The first milestone of this work was to demonstrate that a CNN could effectively assess the level of haze within an image in a manner that is invariant to the image background, lighting or pose. It was found from the training and validation results of the discriminator network that this was possible. The decreasing of both the training loss and the validation loss confirm that the network is correctly determining whether pairs are matching or contrasting with respect to the presence of haze. This determination was made with an accuracy of 95% after 77 training epochs.

The dehazing network architecture is based on the architecture in [4], which demonstrates its effectiveness. We replicated this work and achieved an equivalent performance. The linear clustering of the data points indicates that the predicted transmission map value correlates with the true value for the test dataset described in Section 5. The mean square error of our predicted values from the test dataset was  $1.9 \times 10^{-3}$ . While this final MSE performance is an improvement on [4], it is difficult to make a direct comparison as we were not able to test on an identical dataset. A more important result was to demonstrate the viability of the fine tuning methodology. The end-to-end network performance for each training method are as follows:

### 8.1. Training Methods: Hazy Pairs vs Hazy-Clear Pairs

The first training method discussed in Section 7.3 was considered a failure. While we believe this would prove to be an effective way to train the network, it was found that while the loss decreased rapidly, the actual image output becomes a completely black scene. This black output image had a high measure contrast with the hazy image but it was a completely undesirable outcome. This result is important to the concept development as it further demonstrates that the dehazing network performance can

Table 1: Time to dehaze a Single 380 x 620 Image

	Time (Seconds)
Our Model	0.5
Cycledehaze [9]	3.12
AOD-Net [19]	0.49
Zhang [34]	7.09

be driven effectively by the discriminator network. In the results of the second training method discussed in Section 7.3, the discriminator loss decreased at significantly slower rate than seen in the first method although the image quality improved as an outcome of training. We can be confident in the connection between discriminator loss and the actual dehazing performance on the results of our pre-training.

### 8.2. Process Speed

While the availability of high speed GPU based operation of Neural Network improves the speed of DL based function, these processing resources are not always available in the end use application. Despite the comparative slow computing times for parallel networks, our work places this entire burden on the training stage. Therefore, our final dehazing network is still able to process a  $380 \times 620$  pixel image in approximately 0.5 seconds using an Intel i7 Laptop CPU. While not all comparative authors made their source code available for testing, we were able to demonstrate the performance of our model against several alternatives. Results are seen in Table 1.



Figure 6: Dehazing progress: Sample images taken from the RE-SIDE dataset, Original Hazy Image (Left Lower), Ground truth (Left Upper), Dehazed with Coarse Weights (Right Upper) Dehazed with Fine Tuned Weights (Right Lower)

### 8.3. Comparative testing

Further to these training results, comparative testing of our method using Structural Similarity Index (SSIM) [31] and Peak Signal to Noise Ratio were conducted. These objective metrics applied to two hazy image datasets, firstly the RESIDE Dataset [20] and then to the NTIRE Dehazing 2018 Challenge Dataset [1]. The results for the RESIDE dataset are shown in Table 2 which show a superior performance to existing state-of-art methods. The use of unmatched true clear images in our method allows the training process to produce more natural appearing dehazed images. Following this, we aimed to test how well our model transferred to new datasets. We tested our trained model on the test set of the NTIRE 2018 dataset without doing any training on the NTIRE training set. Our proposed method achieved a PNSR/SSIM of **16.07/0.78** for the indoor dataset and **17.66/0.69** for the Outdoor dataset, and thus, ranked 5 out of 17 for the indoor dataset and 12 out of 18 submissions for the Outdoor dataset. We did not produce competitive PSNR results; this is due to dense haze in this dataset, possibly indicating some amount of over fitting of our model to lighter haze conditions. It is also important to note that our proposed method is designed to train an image dehazing model which does not need the images of both clear and hazy views of the exact same scene during training phase. Thus, the comparison with the submitted works to the NTIRE 2018 Dehazing Challenge is not fair as they extremely rely on having matched training data, including images of both clear and hazy views of the exact same scene.

### 8.4. Coarse Weights vs Fine Tuned Model

Prior to testing our fine tuned model, we carried out additional testing of our course weights in the model. These coarse weights are determined in a similar supervised manner to [4] and [34]. The results seen in Table 2 and images in Figure 6 show a quantifiable improvement in the performance of the model when fine tuned with the Siamese network based discriminator. This demonstrates a key element of our work.

### 8.5. Dehazing in Object Detection

In order to further evaluate our proposed dehazing method, we looked at the end use applications of Single

Table 2: Comparison to existing methods on SOTS/RESIDE Dataset published in [20]

	SSIM	PSNR
DCP [12]	0.8179	16.62
GRM [6]	0.8553	18.86
CAP [24]	0.8364	19.05
Dehazenet [4]	0.8472	21.14
MSCNN [32]	0.8102	17.57
AOD-Net [19]	0.8504	19.06
<b>Our Method - Coarse Weights</b>	0.8518	24.499
<b>Our Method - Fine Tuned Weights</b>	0.8564	24.7645

Table 3: Dehazing effects on Object Detection: COCO MAP performance on select images from RTTS dataset [20]

	COCO Map [7]
Original Hazy Images	21.74
Cycledehaze [9]	20.17
AOD-Net [19]	21.60
Dehazenet [4]	24.04
<b>Our Method - Coarse Weights</b>	23.40
<b>Our Method - Fine Tuned Weights</b>	24.73

Image Dehazing. For purposes of demonstrating the improvement that our fine tuning process produced, we applied a pre-trained object detection model to our original hazy images, images dehazed with our coarse weights and images dehazed with our final fine tuned model. For this testing, we chose the Single-Shot Detection mobilenet V1 [13] trained on the COCO Image dataset [23], as it is a typical model that would be used in a light weight surveillance or real time detection application. The results as seen in Table 3 and Figure 7 show a significant improvement of the performance (Mean-Average Precision) of the detection model when hazy images are first dehazed using a dehazing algorithm. Secondly it shows that our fine tuned model outperforms our coarse weights model and achieves the highest performance.

## 9. Conclusion

We have presented an end-to-end method for the fine tuning of a neural network based on unmatched training data in the context of single image dehazing. The results



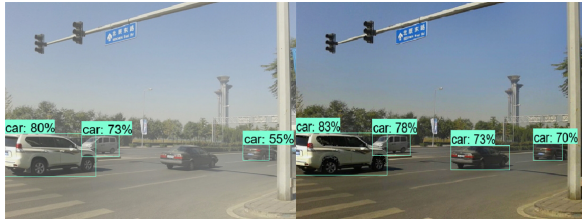


Figure 7: Dehazing process demonstrates an increase in object detection performance. Detection confidence scores improve and a fourth car is detected that was missed in the original image

demonstrate that the Siamese network architecture can be used to compare unmatched images in order to learn haze relevant features that are invariant to a number of image variable eg. scene, lightening and pose. The quantitative results in section 8 show a significant improvement when compared to existing methods found in academic literature. While this work was successful as a proof of concept and shows great potential, there are two areas of further development. Research into single image dehazing often employs subjective testing by human observers. This method of assessment allows people to rank dehazed images in order to determine the effectiveness of the dehazing methods. Our work did not have the resources required to use this type of methodology but it could be used in the future to complement the SSIM and PSNR measure.

## References

- [1] Cosmin Ancuti, Codruta O. Ancuti, and Radu Timofte. “NTIRE 2018 Challenge on Image Dehazing: Methods and Results”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2018.
- [2] D. Berman, T. Treibitz, and S. Avidan. “Non-local Image Dehazing”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. 00. June 2016, pp. 1674–1682. DOI: 10.1109/CVPR.2016.185. URL: doi.ieeecomputersociety.org/10.1109/CVPR.2016.185.
- [3] Jane Bromley et al. “Signature verification using a” siamese” time delay neural network”. In: *Advances in neural information processing systems*. 1994, pp. 737–744.



Ground Truth - Hazy Image



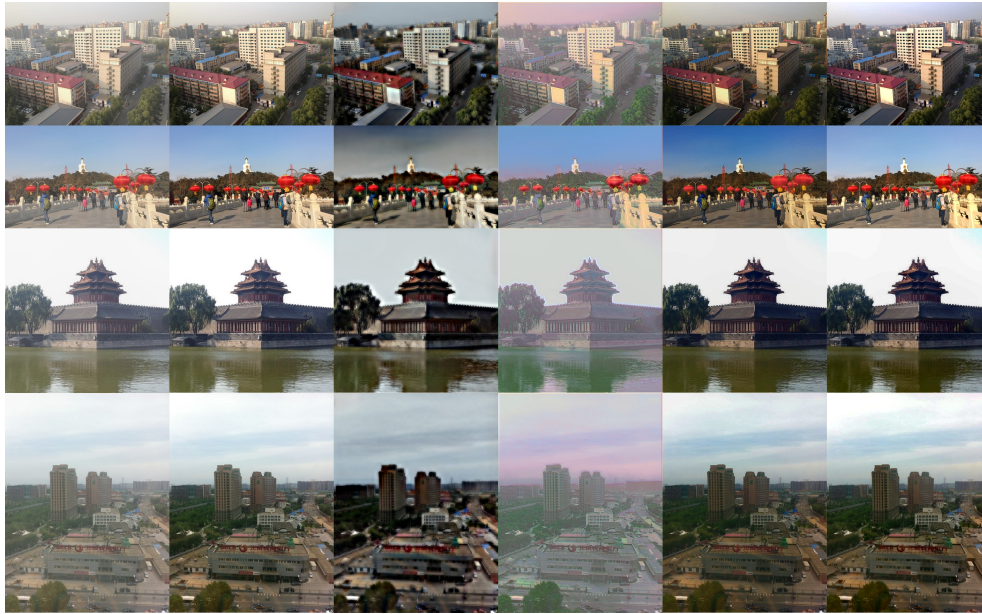
Dehazenet [4] - AOD-Net [19]



Zhang [34] - Our Model

Figure 8: Examples from Real-World Task-Driven Test Set (RTTS) [20] dehazed with multiple methods. This is especially evident around the roof and flags which show less hazy and more fine detail than comparative methods.

- [4] Bolun Cai et al. “Dehazenet: An end-to-end system for single image haze removal”. In: *IEEE Transactions on Image Processing* 25.11 (2016), pp. 5187–5198.
- [5] Peter Carr and Richard Hartley. “Improved Single Image Dehazing Using Geometry”. eng. In: IEEE Publishing, 2009, pp. 103–110. ISBN: 978-1-4244-5297-2.
- [6] Chen Chen, Minh N. Do, and Jue Wang. “Robust Image and Video Dehazing with Visual Artifact Suppression via Gradient Residual Minimization.” In: *ECCV (2)*. Ed. by Bastian Leibe et al. Vol. 9906. Lecture Notes in Computer Science. Springer, 2016, pp. 576–591. ISBN: 978-3-319-46474-9. URL: http://dblp.uni-trier.de/db/conf/eccv/eccv2016-2.html#ChenDW16.



Original Image - Ground Truth - CycleDehaze [9] - Zhang [34] - AOD-Net [19] - Ours

Figure 9: Additional examples dehazed with multiple methods

- [7] COCO Consortium. *Evaluation*. 2019. URL: <http://http://cocodataset.org/#detection-eval> (visited on 06/11/2019).
- [8] X. Deng et al. “Deep second-order siamese network for pedestrian re-identification”. In: vol. 10112. Springer Verlag, 2017, pp. 321, 337. ISBN: 9783319541839.
- [9] Deniz Engin, Anil Genc, and Hazim Kemal Ekenel. “Cycle-Dehaze: Enhanced CycleGAN for Single Image Dehazing”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. 2018.
- [10] Raia Hadsell, Sumit Chopra, and Yann LeCun. “Dimensionality reduction by learning an invariant mapping”. In: *null*. IEEE. 2006, pp. 1735–1742.
- [11] Kaiming He, Jian Sun, and Xiaoou Tang. “Guided Image Filtering”. eng. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 35.6 (June 2013), pp. 1397, 1409. ISSN: 0162-8828.
- [12] Kaiming He, Jian Sun, and Xiaoou Tang. “Single image haze removal using dark channel prior”. In: *IEEE transactions on pattern analysis and machine intelligence* 33.12 (2011), pp. 2341–2353.
- [13] Andrew Howard et al. “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications”. In: *arXiv.org* (2017). URL: <http://search.proquest.com/docview/2074236657/>.
- [14] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).
- [15] H. Koschmieder. “Theorie der horizontalen sichtweite: kontrast und sichtweite”. In: (1925), pp. 1–11.
- [16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.
- [17] Yann LeCun et al. “A theoretical framework for backpropagation”. In: *Proceedings of the 1988 connectionist models summer school*. Vol. 1. CMU, Pittsburgh, Pa: Morgan Kaufmann. 1988, pp. 21–28.
- [18] Yann LeCun et al. “Backpropagation applied to handwritten zip code recognition”. In: *Neural computation* 1.4 (1989), pp. 541–551.

- [19] Boyi Li et al. “AOD-Net: All-In-One Dehazing Network”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Oct. 2017.
- [20] Boyi Li et al. “RESIDE: A Benchmark for Single Image Dehazing”. In: *arXiv preprint arXiv:1712.04143* (2017).
- [21] Jie Li et al. “WaterGAN: Unsupervised generative network to enable real-time color correction of monocular underwater images”. In: *IEEE Robotics and Automation Letters* 3.1 (2018), pp. 387–394.
- [22] Runde Li et al. “Single Image Dehazing via Conditional Generative Adversarial Network”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), pp. 8202–8211.
- [23] Tsung-Yi Lin et al. “Microsoft COCO: Common Objects in Context”. In: *CoRR* abs/1405.0312 (2014). arXiv: 1405.0312. URL: <http://arxiv.org/abs/1405.0312>.
- [24] Zhu Qingsong & Mai Jiaming & Shao Ling. “A Fast Single Image Haze Removal Algorithm Using Color Attenuation Prior.” In: *IEEE transactions on image processing* : a publication of the IEEE Signal Processing Society. 2015.
- [25] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: <https://www.tensorflow.org/>.
- [26] Gaofeng Meng et al. “Efficient Image Dehazing with Boundary Constraint and Contextual Regularization”. In: *Proceedings of the 2013 IEEE International Conference on Computer Vision. ICCV '13*. Washington, DC, USA: IEEE Computer Society, 2013, pp. 617–624. ISBN: 978-1-4799-2840-8. DOI: 10.1109/ICCV.2013.82. URL: <http://dx.doi.org/10.1109/ICCV.2013.82>.
- [27] Gerhard Neuhold et al. “The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Oct. 2017.
- [28] Daniel Scharstein et al. “High-resolution stereo datasets with subpixel-accurate ground truth”. In: *German Conference on Pattern Recognition*. Springer. 2014, pp. 31–42.
- [29] Yafei Song et al. “Single Image Dehazing Using Ranking Convolutional Neural Network”. In: *IEEE Transactions on Multimedia* PP (Nov. 2017), pp. 1–1. DOI: 10.1109/TMM.2017.2771472.
- [30] Jonas Uhrig et al. “Sparsity Invariant CNNs”. In: *International Conference on 3D Vision (3DV)*. 2017.
- [31] Zhou Wang et al. “Image Quality Assessment: From Error Visibility to Structural Similarity”. In: *IEEE TRANSACTIONS ON IMAGE PROCESSING* 13.4 (2004), pp. 600–612.
- [32] Ren Wenqi et al. “Single Image Dehazing via Multi-Scale Convolutional Neural Networks”. In: *European Conference on Computer Vision*. 2016.
- [33] Dong Yang and Jian Sun. “Proximal Dehaze-Net: A Prior Learning-Based Deep Network for Single Image Dehazing”. In: *The European Conference on Computer Vision (ECCV)*. Sept. 2018.
- [34] Shengdong Zhang et al. “Joint learning of image detail and transmission map for single image dehazing”. In: *The Visual Computer* (Nov. 2018). DOI: 10.1007/s00371-018-1612-9. URL: <https://doi.org/10.1007/s00371-018-1612-9>.