**DEPT. OF CEN**
**AMRITA SCHOOL OF ENGINEERING**
**III-B. TECH**
**CSE-AI**

**SEMESTER-5**
**21AIE303**
**SIGNAL AND IMAGE PROCESSING**
**END SEMESTER PROJECT**
**DATE: 23-01-2023**


**TEAM MEMEBERS:**

**S.K Eswar Sudhan CB.EN.U4AIE20015**

**G.Venkata Krishna Rayalu CB.EN.U4AIE20016**

**R.Gokul CB.EN.U4AIE20018**

**K.Likhitha CB.EN.U4AIE20030**

**Nadella Manish CB.EN.U4AIE20037**

## Abstract:

This report gives insights about various dehazing algorithms and how these algorithms can be applied for dehazing remote sensing images.

The report also gives details about metric analysis for the algorithms that have been used with their drawbacks when applied on satellite images.

## Introduction:

Haze and smoke can significantly degrade the visual quality of images, making it difficult for humans to interpret and for computers to analyze. Image dehazing is a technique used to remove the effects of haze and smoke from images, thus improving their visual quality and making them more useful for analysis. In this report, we will provide an overview of various dehazing algorithms and their applications in dehazing remote sensing images. The report will also include a detailed analysis of the performance of these algorithms using metrics and their drawbacks when applied to satellite images.

Remote sensing images are often affected by atmospheric scattering, which causes haze and reduces visibility. Dehazing these images is crucial for their interpretation and analysis. The most common approach to dehazing is to estimate the transmission map of the haze, which represents the proportion of light that reaches the camera without scattering. Once the transmission map is estimated, the dehazed image can be obtained by applying a correction factor to the original image.

There are various dehazing algorithms that have been proposed in recent literature, including the Dark Channel Prior (DCP) algorithm, haze removal using DCP and Morphology, the Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm and its extension to retinex theory and dehazing using color attenuation prior. Each algorithm has its own strengths and weaknesses, and the choice of algorithm will depend on the specific application and the type of image being dehazed.

In this report, we will provide a detailed comparison of these dehazing algorithms using metrics such as the Peak Signal-to-Noise Ratio (PSNR), and the Structural Similarity Index (SSIM), We will also analyze the drawbacks of these algorithms when applied to satellite images and discuss the limitations of the current state-of-the-art dehazing techniques.

In conclusion, this report aims to provide a comprehensive overview of the various dehazing algorithms and their applications in dehazing remote sensing images. It also aims to provide a detailed analysis of the performance of these algorithms using metrics and their drawbacks when applied to satellite images. With the increasing use of remote sensing images in various applications, dehazing techniques play an important role in improving the visual quality and analysis of these images. By understanding the different dehazing algorithms and their limitations, we can make better decisions on which algorithm to use for specific applications and improve the overall dehazing performance.

## Literature:

### Mathematical Equations for Homomorphic Filtering in Frequency Domain: A Literature Survey:

One of the key methods for digital image enhancement, particularly when the input image is affected by bad lighting circumstances, is the homomorphic filtering technique. Numerous imaging applications, such as biometric, medical, and robotic vision, have adopted this filtering method. In order to lessen the importance of low frequency components, homomorphic filtering employs a high-pass type filter in the frequency domain. Few sets of equations are discussed.

The technique of using a combination of Contrast Limited Adaptive Histogram Equalization (CLAHE) and a Guided Filter for image dehazing is a method that aims to improve the visibility of hazy images by increasing the contrast and reducing noise. The CLAHE algorithm is used to enhance the local contrast of the image, while the guided filter is applied to smooth out the noise and preserve the edges. The combination of these two techniques can result in a clearer and more detailed image compared to using either one alone.

### Single-Image Dehazing Using Color Attenuation Prior Based on Haze-Lines:

This paper discusses of we can change the base color Attenuation prior algorithm and include Haze Lines Prior algorithm to estimate airLight in order to get a little bit more accurate dehazed Image. This paper used PSNR and SSIM for evaluation for algorithms.

### A Fast Image Dehazing Algorithm Using Morphological Reconstruction:

This paper discusses an alternative methos to refine the transmission map using morphological concepts. The refined transmission map is observed to be better than the one generated by DCP algorithm. The proposed algorithm is compared with the present dehazing algorithms and it is compared using PSNR and SSIM metrics.

## Objective:

Metric analysis is an important step in evaluating the performance of image dehazing algorithms. It allows us to quantify the effectiveness of the algorithm in terms of the visual quality and accuracy of the dehazed image. By comparing the performance of different algorithms using metrics, we can identify the strengths and weaknesses of each algorithm and make an informed decision on which algorithm to use for a specific application.

The most commonly used metrics for evaluating image dehazing algorithms are the Peak Signal-to-Noise Ratio (PSNR), the Structural Similarity Index (SSIM).

The PSNR is a measure of the peak error between the original and dehazed images. It is calculated as the ratio of the maximum possible pixel value to the root mean squared error between the original and dehazed images. A higher PSNR value indicates that the dehazed image is of higher quality and has a smaller error compared to the original image.

The SSIM is a metric that measures the structural similarity between the original and dehazed images. It takes into account the luminance, contrast, and structure of the images. A higher SSIM value indicates that the dehazed image is more similar to the original image in terms of structure and visual quality.

By comparing the performance of different algorithms using these metrics, we can identify which algorithm performs best in terms of visual quality and accuracy. For example, if one algorithm has a higher PSNR value than another, it indicates that the dehazed image produced by that algorithm is of higher quality and has a smaller error compared to the original image. Similarly, if one algorithm has a higher SSIM value than another, it indicates that the dehazed image produced by that algorithm is more similar to the original image in terms of structure and visual quality.

In addition to these metrics, it is also important to analyze the drawbacks of the algorithms when applied to satellite images. For example, some algorithms may struggle with images that have a low contrast or a low signal-to-noise ratio. Additionally, some algorithms may be computationally expensive and may not be suitable for real-time applications.

By comparing the performance of different algorithms using these metrics, we can identify which algorithm performs best in terms of visual quality and accuracy. For example, if one algorithm has a higher PSNR value than another, it indicates that the dehazed image produced by that algorithm is of higher quality and has a smaller error compared to the original image. Similarly, if one algorithm has a higher SSIM value than another, it indicates that the dehazed image produced by that algorithm is more similar to the original image in terms of structure and visual quality.

Overall, metric analysis is an important step in evaluating the performance of image dehazing algorithms. By comparing the performance of different algorithms using metrics, we can identify the strengths and weaknesses of each algorithm and make an informed decision on which algorithm to use for a specific application. Additionally, it is important to analyze the drawbacks of the algorithms when applied to satellite images in order to understand their limitations and improve the overall dehazing performance.

## Theoretical Background:

Image dehazing is a technique used to remove the effects of atmospheric haze from an image. The main goal of image dehazing is to recover the original hazy-free image, which is often more visually pleasing and easier to interpret than the hazy image.

This can be especially important in applications such as surveillance, remote sensing, and autonomous driving, where clear images are crucial for object detection, tracking, and navigation.

Haze can also be a problem for outdoor photography, making it difficult to capture clear images of distant landscapes or buildings. Image dehazing can help to improve the visual quality of these images and make them more suitable for printing or sharing online.

In addition to improving the visual quality of images, image dehazing can also have a significant impact on the performance of computer vision algorithms. Haze can cause objects in an image to appear distorted, blurred, or partially occluded, making it difficult for algorithms to accurately detect or recognize them. Image dehazing can help to improve the performance of these algorithms by removing the effects of haze and providing a clearer view of the scene.

The main idea behind image dehazing is to estimate the scene transmission (also known as the medium transmission or atmospheric veil) and then use it to recover the original hazy-free image. In general, the process of image dehazing can be divided into two steps: estimating the scene transmission and then using the transmission to recover the original hazy-free image. The specific algorithm used will depend on the type of image and the desired level of dehazing.

Another approach is based on the atmospheric scattering model, which describes the formation of a hazy image as the result of the scattering of light by atmospheric particles. This model can be used to estimate the scene depth and the transmission using a single image.

## Methodology:

## Dark Channel Prior:

- This algorithm has three main objectives:
- Estimate atmospheric light
- Find Transmission map
- Smoothen Transmission map using regularization

The dark channel prior is a method used in image dehazing to estimate the atmospheric transmission map, which is used to remove haze from an image. The basic idea behind the dark channel prior is that, in most natural images, there is a small patch of pixels that have very low intensity values in at least one color channel. This patch of pixels is called the dark channel, and it is used to estimate the atmospheric transmission map

$$I(x) = t(x)J(x) + (1 - t(x))A$$

This is the equation represents the image captured where $I(x)$ is Image captured $J(x)$ is the actual scene radiance without haze A is atmospheric light and $t(x)$ is transmission

The atmospheric light is a scalar value representing the amount of light that is scattered by the atmosphere and reaches the camera. In the dark channel prior algorithm, atmospheric light is calculated from the highest intensity value of the pixels in the dark channel. Once the dark channel of an image is calculated, where each pixel represents the minimum intensity value of a patch centered on that pixel, the atmospheric light is calculated by taking the highest intensity value of the pixels in the dark channel.

In the dark channel prior (DCP) method, the transmission map is estimated by first calculating the dark channel of the image. The dark channel is a 2D array, where each pixel represents the minimum intensity value of a patch centered on that pixel. The patch size is typically 3x3, 5x5, or 7x7 pixels. With the atmospheric light, the transmission map can be calculated using the formula:

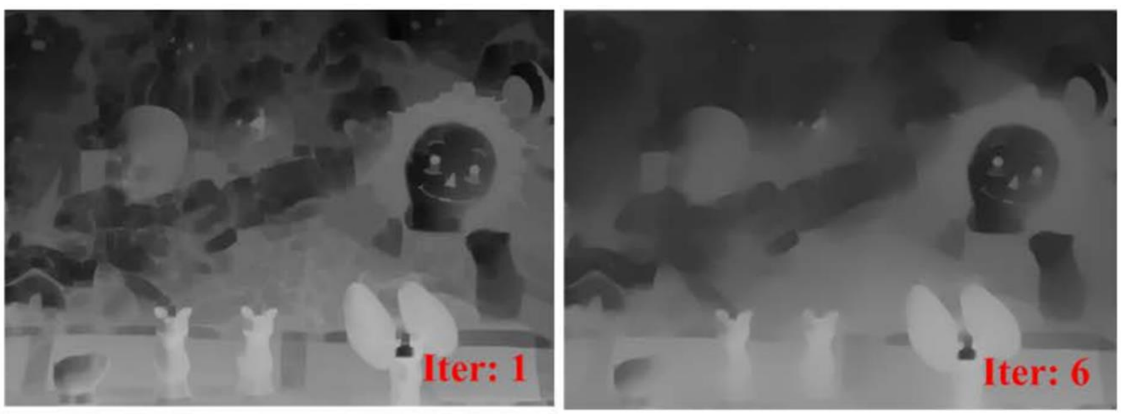
$$t(x) = 1 - \min(I(x))/A$$

Where x is a pixel in the image, I(x) is the intensity of the pixel and A is the atmospheric light

Later we perform regularization and use the equation below to find the scene without haze

$$J(x) = I(x) - A/( [\max (t(x), \mu)]^\delta) + A$$

Where μ is constant 0.001 to avoid division by zero




$J(x) = I(x) - A [\max (t(x), \mu )]^\delta + A$

## Dark Channel Prior using Morphology:

This algorithm has three main objectives:

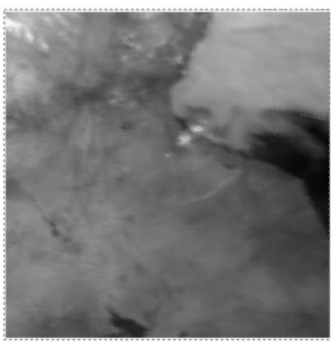
- Estimate atmospheric light
- Find the transmission map
- Refine the transmission map

Atmospheric light and the transmission map are computed using concepts similar to Dark Channel Prior.

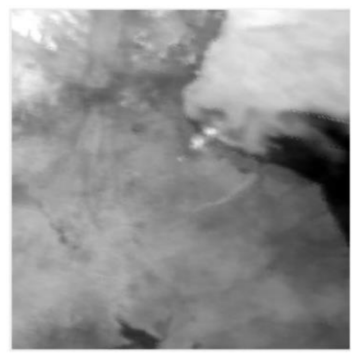
But to refine the transmission map, we used the concepts of morphology.

Refining the transmission map:

- Perform closing operation on initial transmission map and reconstruct the image. This operation removes the small dark elements from the image.
- Perform opening reconstruction on the transmission map. This operation removes small objects which are smaller than structuring element. On doing the this some small useful data might be lost in order to save it we store the removed objects.
- Recover ranges from original image and add the removed small objects in order to get the refined transmission map
- Feed the atmospheric light and refined transmission amp to ASM model to get the Dehazed image.



InitialTransmission Map



RefinedTransmission Map

Refined transmission map after performing morphology.

## Color attenuation prior using Haze lines prior:

This algorithm is mixture of CAP and Haze line prior algorithms.

The main objective of this algorithm:

- Find the scene depth
- Find the scaterring coeffecient
- Estimate the airLight

How to find scene depth:

In order to find the scene depth we use CAP algorithm. This algorithm states that:

'If the thickness of the haze increases the scene depth also increases'.

Hence the scene depth of the image will be linearly dependent on the difference between brightness and saturation in the image. Using this we estimate the scene depth.

How to find scattering coefficient:

In general scattering coefficient is kept constant as in most of the images the haze is evenly spread. But in case of satellite images the haze is inconsistent and unevenly spread. So the scattering coefficient must keep changing.

Scattering coefficient directly depends on scene depth. Hence with an increase in scene depth, scatering coefficient increases exponentially.

How to find airlight:

In order to estimate the airLight we use Haze lines Prior.

In this algorithm, we map the pixels in the image to RGB colorspace. Then the model takes these pixels as lines and these lines converge at a point. This point is airLight. Then we use a feature extraction model in order to find the origin point for this airLight.

After finding scene depth,scaterring coefficient and estimating the airlight, we send these values to ASM model in order to get the dehazed image.

## CLAHE Algorithm and Mix CLAHE:

This algorithm defines two functions, clahe and clahe2, which can be used to apply the Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm to an image.

The clahe function takes three parameters:

image: The input image, which should be a 2D array.

clipping_limit: A float value that sets the limit for contrast enhancement. Higher values allow for more contrast enhancement, but can also amplify noise in the image. The default value is 2.0.

grid_size: A tuple that sets the size of the contextual regions, which are the small regions of the image that the algorithm adjusts the contrast of separately. The default value is (8, 8).

The function first gets the dimensions of the input image, and then divides the image into non-overlapping contextual regions based on the grid_size parameter. It then creates an output image (clahe_image) that is the same size as the input image, but filled with zeros.

The function then performs CLAHE on each grid. For each grid, it:

- Gets the grid region of the image by slicing the input image with the grid's row and column indices
- Gets the histogram of the grid region, which is a distribution of the number of pixels of each intensity level in the grid region
- Clips the histogram by a certain value. Clipping the histogram means that it limits the number of pixels in the histogram that can have a high intensity level. This is done to avoid amplifying noise in the image. The clip limit is determined by the clipping_limit parameter, which is a float value. The clip limit value multiplied by the total number of pixels in the histogram gives the maximum number of pixels that can have high intensity levels.
- Calculates the average number of pixels to redistribute, which is the total number of pixels in the histogram minus the number of clipped pixels, divided by the number of intensity levels.
- Redistributes the clipped pixels by adding them to the neighboring intensity levels. This is done by starting from the highest intensity level and adding the clipped pixels to the intensity levels in descending order until all the clipped pixels have been added.
- Calculates the cumulative density function (CDF) of the histogram, which is a distribution of the cumulative number of pixels of each intensity level in the grid region
- Maps the original pixel values to the new values using the CDF. This is done using the numpy's interp() function, which maps the original pixel values to new values according to the CDF.
- Inserts the grid region back into the output image by slicing the clahe_image with the grid's row and column indices.
- After all the grids have been processed, the function returns the output image, clahe_image, which contains the result of the CLAHE algorithm applied to the input image.

Now we get the output that is dehazed image

**Working of Mix CLAHE:**

This works similar to CLAHE but works in HSV space, HSV stands for Hue, Saturation, and Value. It is a color space that represents colors using three channels: Hue, Saturation, and Value. Hue represents the color itself, and is typically represented as a value between 0 and 360. Saturation represents the intensity of the color, and is typically represented as a value between 0 and 100. Value represents the brightness of the color, and is typically represented as a value between 0 and 100. The clahe2 function first converts the image from the BGR color space to the HSV color space using the cv2.cvtColor() function. This allows the function to separate the color information (Hue and Saturation) from the brightness information (Value). By isolating the brightness information, the function can apply the CLAHE algorithm specifically to the V channel of the image, which represents the Value or the brightness. This can help to remove haze more effectively since haze mainly affects the visibility by reducing the brightness of the image. After applying the CLAHE to the V channel of the image, the function converts the image back to the BGR color space using the cv2.cvtColor() function. Finally, it applies the CLAHE on each channel of the image using the clahe function, which helps to further improve the contrast and visibility of the image. The final output of the function is the image with CLAHE applied to each channel.

**The two parameters that control the dehazing:**

The clip_limit parameter sets the limit for contrast enhancement. A higher value allows for more contrast enhancement, but can also amplify noise present in the image. A lower value will limit the contrast enhancement and will result in a less contrasted image. Increasing the clip_limit value allows more pixels to be included in the histogram thus resulting in a more contrasted image but it also amplifies noise in the image.

The grid_size parameter sets the size of the contextual regions, which are the small regions of the image that the algorithm adjusts the contrast of separately. A smaller grid size will result in more regions, and therefore more local adjustments to the contrast of the image. A larger grid size will result in fewer regions, and therefore fewer local adjustments to the contrast of the image. Increasing the grid size will result in more general contrast enhancement while decreasing the grid size results in more local contrast enhancement.

**Homomorphic Filter:**

A homomorphic filter is a type of image processing filter that is used to enhance the brightness of an image while preserving its overall contrast. It is based on the idea of performing mathematical operations on the image in the frequency domain (using the Fourier or Wavelet transform) rather than the spatial domain. The basic idea of the homomorphic filter is to adjust the image's brightness by applying a non-linear transformation to the image's intensity values. The transformation is typically applied to the logarithm of the image intensity values. The filter is called "homomorphic" because the logarithm function is applied to both the image and the illumination, which results in a filter that is independent of the image's brightness.

One of the main benefits of the homomorphic filter is that it can be used to enhance images that have uneven lighting, such as images taken in low-light conditions or images with reflections. Additionally, it can be used to remove noise from an image, by suppressing the noise in the low-frequency portion of the image's frequency spectrum. It's important to note that the homomorphic filter is sensitive to the choice of parameters, such as the cutoff frequency and the parameters of the non-linear transformation. Therefore, it is important to carefully adjust these parameters to obtain the best results for a specific image and application.

The Homomorphic filter is a signal processing method that allows to enhance the low frequency components of an image while suppressing the high frequency components. The filter is applied to the image in the frequency domain, which can be achieved by performing a Fourier or Wavelet transform on the image. The filter is defined by a transfer function, which is applied to the magnitude of the frequency components. The transfer function typically has the form:

$$\boxed{H(u,v) = (Y_H - Y_L)\left[1 - e^{-C\left(\frac{D^2(u,v)}{D_0^2}\right)}\right] + Y_L}$$

Where H(u,v) is the transfer function, (u,v) are the spatial frequencies, and c is a parameter that controls the cutoff frequency.

The filter is applied to the image by multiplying the magnitude of the frequency components by the transfer function, and then performing an inverse Fourier or Wavelet transform to get the filtered image in the spatial domain. One of the main advantages of the homomorphic filter is that it can be used to enhance images that have uneven lighting. By removing the effect of the lighting from the image, the filter allows to see details that would be otherwise obscured by shadows or reflections. Additionally, it can be used to remove noise from an image, by suppressing the noise in the high-frequency portion of the image's frequency spectrum.

It's important to note that the homomorphic filter can be sensitive to the choice of parameters, such as the cutoff frequency and the parameters of the non-linear transformation. Therefore, it is important to carefully adjust these parameters to obtain the best results for a specific image and application. Also, it's important to mention that the homomorphic filter is not only used for image dehazing but also used in areas such as image enhancement, image restoration, and image analysis.

## Multi Scale Retinex Algorithm:

The Multi-Scale Retinex algorithm is a computational method for color image enhancement that attempts to improve the visual perception of images by adjusting the color balance and luminance levels.

The algorithm is based on the idea that the perceived color of an object is influenced by the color of the surrounding environment, and that the human visual system is able to adapt to different lighting conditions.

The Multi-Scale Retinex algorithm uses a multi-scale image decomposition to separate the image into different frequency bands, which are then processed separately to adjust the color balance and luminance levels. The algorithm is commonly used in image processing and computer vision applications, such as image enhancement, color constancy, and automatic white balancing

This algorithm can be used to remove haze from an image by removing the effect of atmospheric scattering. The basic idea is that the MSR algorithm can estimate the global atmospheric light and the transmission map of the image, which can then be used to dehaze the image.

The algorithm is based on the Retinex theory, which states that the perceived brightness of a pixel in an image is the logarithm of the product of the scene radiance and the camera sensitivity. The MSR algorithm separates the illumination of the scene from the reflectance of the scene by taking the logarithm of the original image, and then subtracting the logarithm of the convolved image from it. The convolved image is obtained by convolving the original image with a Gaussian kernel at different scales.

The algorithm can be broken down into the following steps:

1. Convert the input image to a specified precision (8, 16 or 32 bit) and scale the image by adding 1 before taking the logarithm since the logarithm of zero and negative values are not defined.
2. Convolve the image with a Gaussian kernel of size sigmaS at each scale specified in the sigmaS vector.
3. For each scale, calculate the logarithm of the original image minus the logarithm of the convolved image, and divide the result by the number of scales.
4. Take the exponential of the result, subtract 1 and rescale the result to the range [0, 1].

5. Apply the simplest color balance algorithm like piece wise contrast streching to get better output by setting the lower and upper cutoff values.
6. rescale the output image to the precision scale.

In step 3 of the MSR algorithm, the logarithm of the original image is subtracted from the logarithm of the convolved image (after convolution with a Gaussian kernel of a specific scale). The reason for this is that the Retinex theory states that the perceived brightness of a pixel in an image is the logarithm of the product of the scene radiance and the camera sensitivity. So, by taking the logarithm of the original image, it is possible to separate the illumination of the scene from the reflectance of the scene.

By subtracting the logarithm of the convolved image, which represents the illumination of the scene, from the logarithm of the original image, it is possible to obtain an estimate of the reflectance of the scene, which is the part of the image that carries the information about the actual colors and textures of the scene.

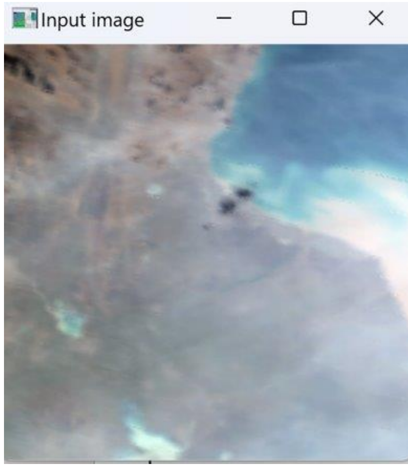
So, in simple terms, using logarithm in this step is to separate the illumination from the reflectance of the scene, which is the core of Retinex theory.

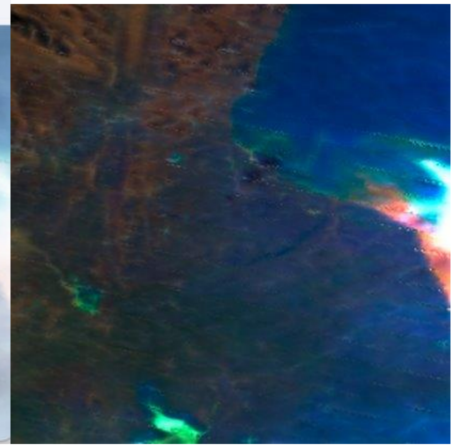
It is also divided by the number of scales as the algorithm is run for multiple scales, so this step is averaging the reflectance for all scales.

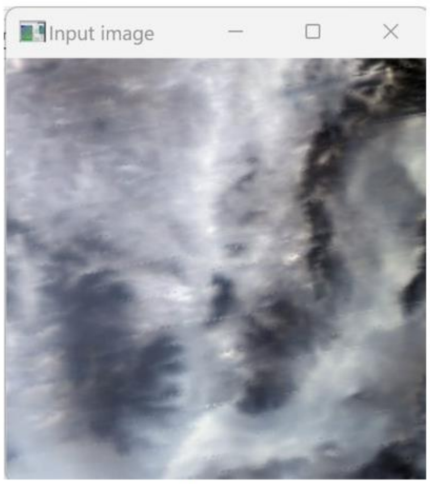
## Experimental Results and Discussion:

**DCP:**

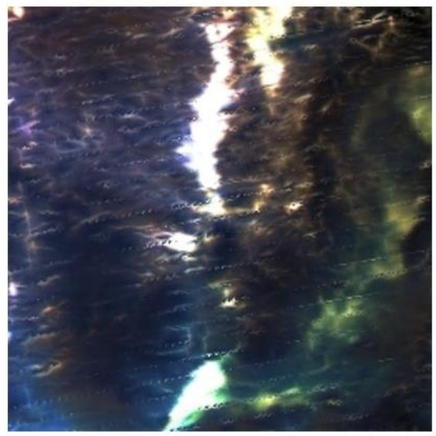
**INPUT and OUTPUT:**






Input Hazy Image

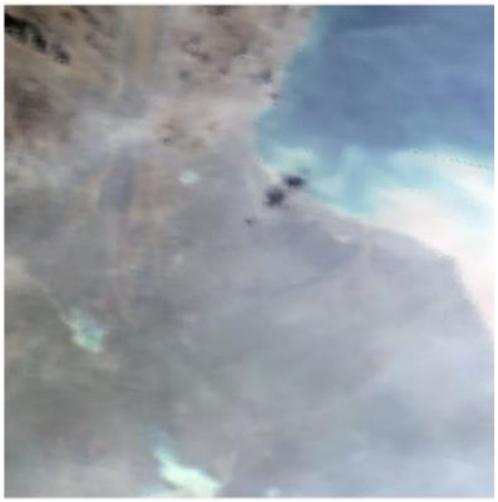


Output Dehazed Image

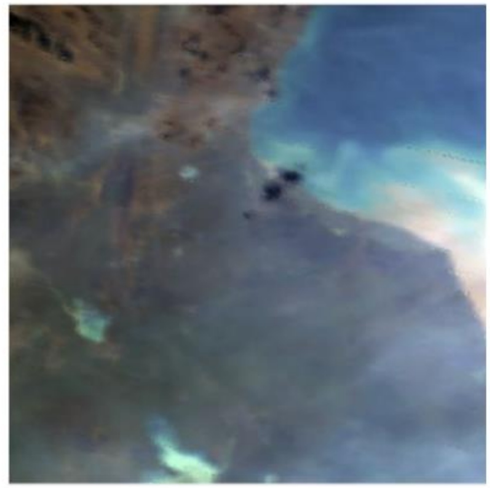

Input Hazy Image

Output Dehazed Image

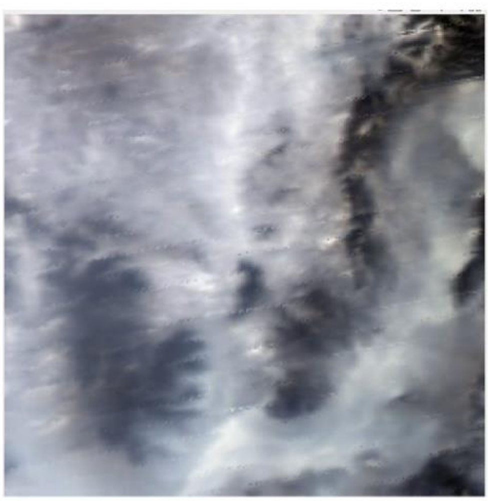
**DCP using Morphology:**



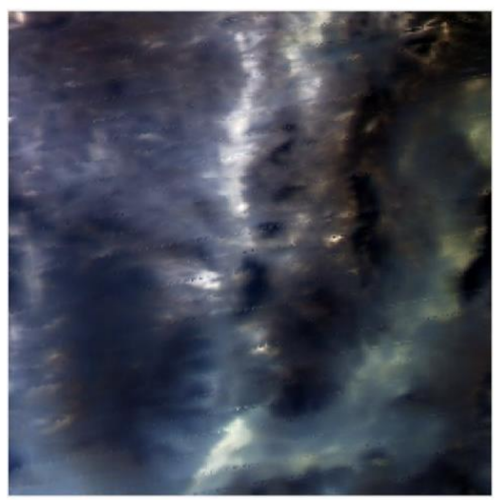
Input Hazy Image



Output Dehazed Image



Input Hazy Image

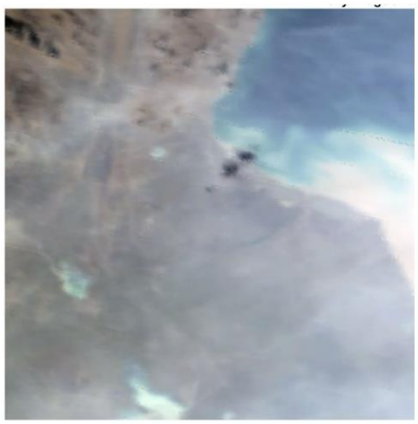


Output Dehazed Image

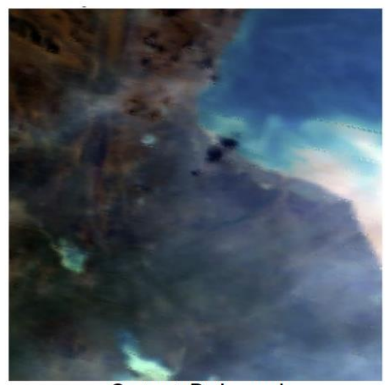
Input Hazy Image

Output Dehazed Image

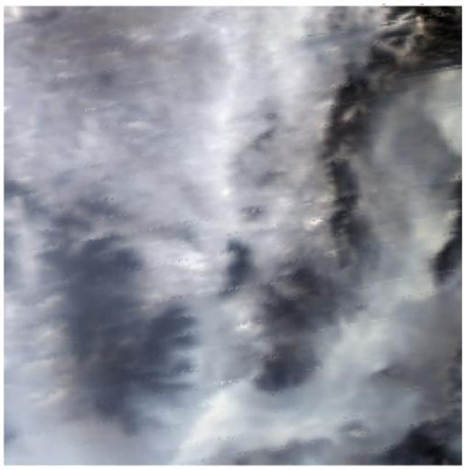
**Color attenuation prior:**



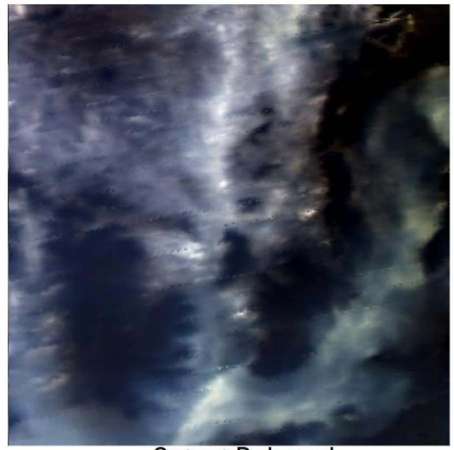
Input Hazy Image



Output Dehazed Image



Input Hazy Image

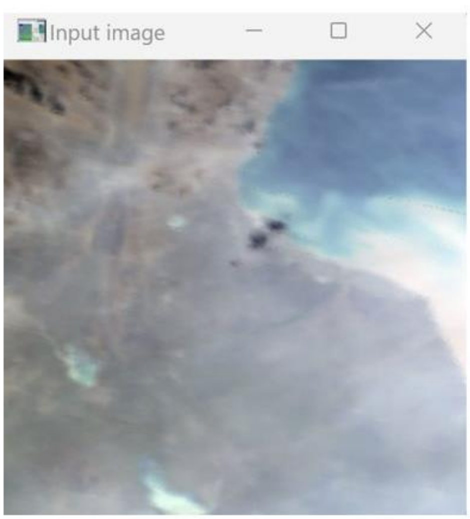


Output Dehazed Image

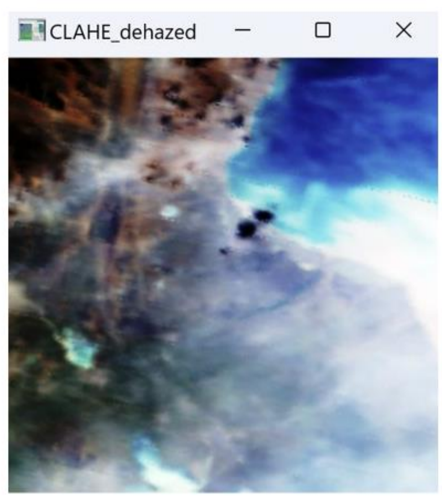
Input Hazy Image

Output Dehazed Image

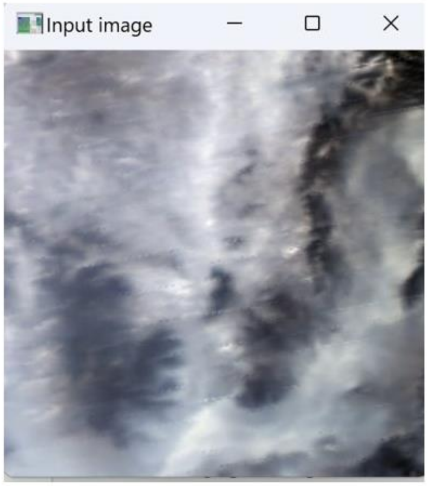
**CLAHE:**



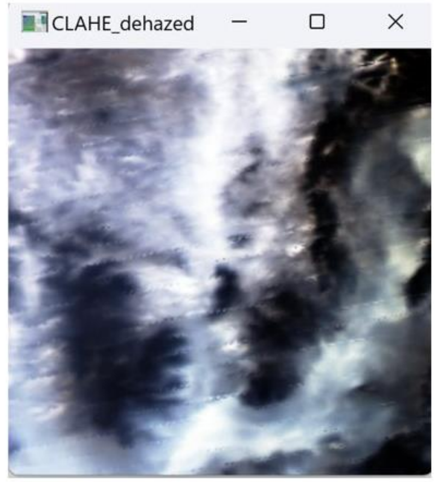

Input Hazy Image




Output Dehazed Image




Input Hazy Image



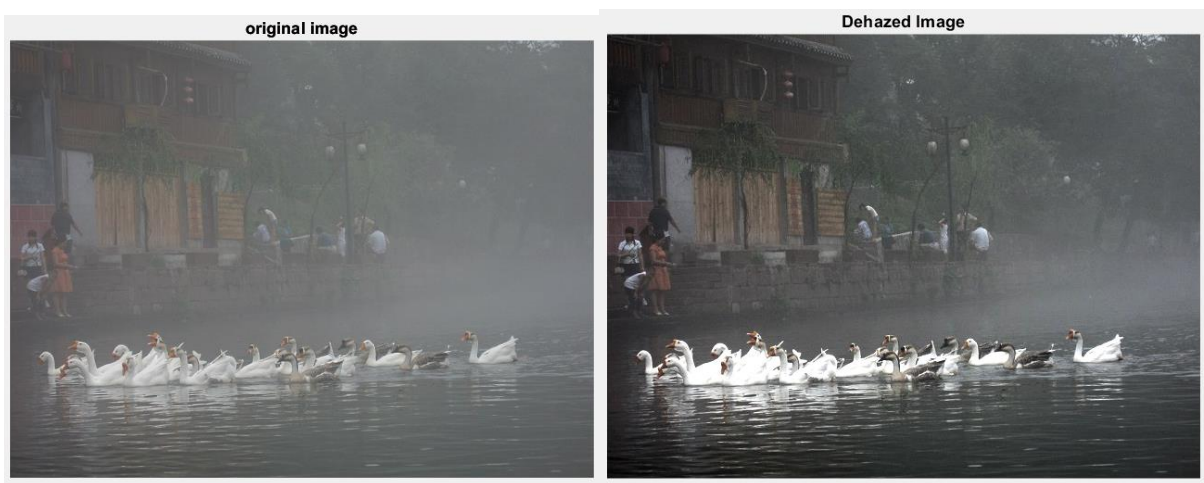

Output Dehazed Image


Input Hazy Image


Output Dehazed Image

**Homomorphic Filter:**

Original Image Dehazed Image




**Mutli scale retinex:**

**Original Image**

**Dehazed Image**

**Original Image**

**Dehazed Image**

**Satellite Image Dataset:**

Image 1

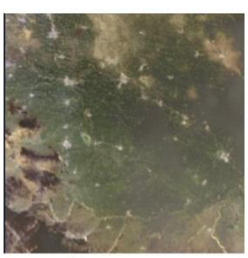


Image 2

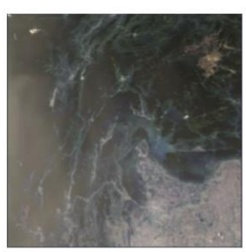


Image 3

Image 4

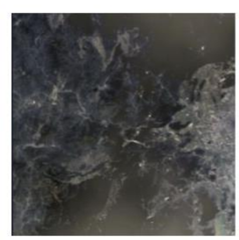


Image 5

**Below is the comparative table which contains the metric analysis (PSNR) of the above algorithms for the above satellite images:**

| Sno. | DCP | DCP morphology | CLAHE | MIX CLAHE | CAP | MSR | Homomorphic filter | Algo |
|---|---|---|---|---|---|---|---|---|
| 1 | 11.92 | 10.16 | 16.03 | 15.89 | 12.28 | 14.97 | 18.11 | 9.03 |
| 2 | 14.58 | 15.61 | 13.05 | 12.89 | 16.42 | 16.02 | 17.45 | 14.80 |
| 3 | 15.30 | 17.54 | 11.86 | 11.71 | 19.27 | 10.90 | 19.89 | 18.16 |
| 4 | 20.05 | 14.75 | 9.79 | 9.83 | 17.36 | 10.99 | 21.78 | 22.60 |
| 5 | 18.83 | 18.14 | 9.61 | 9.60 | 23.05 | 11.17 | 21.67 | 22.60 |

**Below is the comparative table which contains the metric analysis (SSIM) of the above algorithms for the above satellite images:**

| Sno. | DCP | DCP morphology | CLAHE | MIX CLAHE | CAP | MSR | Homomorphic filter | Algo |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.49 | 0.632 | 0.635 | 0.62 | 0.711 | 0.78 | 0.78 | 0.45 |
| 2 | 0.59 | 0.717 | 0.58 | 0.584 | 0.776 | 0.57 | 0.78 | 0.70 |
| 3 | 0.54 | 0.747 | 0.67 | 0.695 | 0.757 | 0.53 | 0.72 | 0.74 |
| 4 | 0.80 | 0.45 | 0.483 | 0.471 | 0.629 | 0.48 | 0.83 | 0.87 |
| 5 | 0.68 | 0.696 | 0.491 | 0.480 | 0.805 | 0.46 | 0.77 | 0.82 |

**Normal Image Dataset:**

Image 1

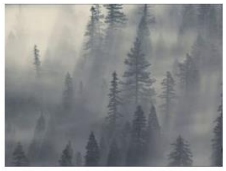


Image 2

Image 3

Image 4

Image 5

**Below is the comparative table which contains the metric analysis (PSNR) of the above algorithms for the above normal images:**

| Sno. | DCP | DCP morphology | CLAHE | MIX CLAHE | CAP | MSR | Homomorphic filter | Algo |
|---|---|---|---|---|---|---|---|---|
| 1 | 28.33 | 14.83 | 13.09 | 13.09 | 12.79 | 14.69 | 19.94 | 15.01 |
| 2 | 27.66 | 15.78 | 16.64 | 15.11 | 14.17 | 15.12 | 18.83 | 10.80 |
| 3 | 27.58 | 16.71 | 16.7 | 16.53 | 11.35 | 12.987 | 19.41 | 8.98 |
| 4 | 27.52 | 17.59 | 16.58 | 16.46 | 13.65 | 14.6 | 18.22 | 8.83 |
| 5 | 29.18 | 17.31 | 17.83 | 15.71 | 15.34 | 16.3561 | 20.19 | 17.39 |

**Below is the comparative table which contains the metric analysis (SSIM) of the above algorithms for the above satellite images:**

| Sno. | DCP | DCP morphology | CLAHE | MIX CLAHE | CAP | MSR | Homomorphic Filter | Algo |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.74 | 0.532 | 0.748 | 0.748 | 0.489 | 0.742 | 0.86 | 0.77 |
| 2 | 0.52 | 0.854 | 0.781 | 0.713 | 0.802 | 0.715 | 0.89 | 0.60 |
| 3 | 0.58 | 0.815 | 0.784 | 0.772 | 0.627 | 0.58 | 0.88 | 0.42 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 4 | 0.62 | 0.799 | 0.782 | 0.773 | 0.709 | 0.5289 | 0.84 | 0.42 |
| 5 | 0.76 | 0.714 | 0.866 | 0.709 | 0.737 | 0.8331 | 0.80 | 0.80 |

## Conclusion:

Based on the various metric analysis these are the concerns that are associated with most of the dehazing algorithms:

- Limited generalizability: Many image dehazing algorithms are designed to work well on specific types of hazy images, such as those with a specific type of atmospheric scattering. They may not perform well on other types of hazy images, such as those caused by other types of atmospheric conditions or man-made pollution.
- Computational complexity: Some dehazing algorithms require significant computational resources, making them impractical for real-time applications or for use on low-powered devices.
- Color distortion: Some dehazing algorithms can introduce color distortion in the recovered image, which can make the image appear unnatural.
- Requirement of extra information: Some dehazing algorithms require additional information, such as the scene depth map or the atmospheric light, which may not be available in all cases.
- Over-enhancement: Some dehazing algorithms may over-enhance the image, which can lead to an artificially bright image that does not accurately reflect the true scene.

Therefore, it is not possible to stick to one algorithm for all types of hazy images or situations. Depending on the specific scenario and requirements, different algorithms may be more or less suitable, and it may be necessary to use multiple algorithms or to combine them in some way to achieve the best results.

## References:

1. Mathematical Equations for Homomorphic Filtering in Frequency Domain: A Literature Survey
2. An Improved Image Dehazing Technique using CLAHE and Guided Filter
3. Single-Image Dehazing Using Color Attenuation Prior Based on Haze-Lines
4. A Fast Image Dehazing Algorithm Using Morphological Reconstruction.
5. Efficient Image Dehazing with Boundary Constraint and Contextual Regularization - Gaofeng MENG