

THIN CLOUD REMOVAL USING LOCAL MINIMIZATION AND IMAGE TRANSFORMATION IN
HSI COLOR SPACE



A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Applied Mathematics and Computational Science
Department of Mathematics and Computer Science
Faculty of Science
Chulalongkorn University
Academic Year 2018
Copyright of Chulalongkorn University

การกำจัดเมฆแบบบางโดยใช้การหาค่าต่ำสุดเฉพาะที่และการแปลงภาพในปริภูมิสีเอชเอสไอ



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาคณิตศาสตร์ประยุกต์และวิทยาการคณนา ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2561
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Thesis Title THIN CLOUD REMOVAL USING LOCAL MINIMIZATION
AND IMAGE TRANSFORMATION IN HSI COLOR SPACE
By Mr. Thanet Markchom
Field of Study Applied Mathematics and Computational Science
Thesis Advisor Associate Professor Rajalida Lipikorn, Ph.D.

Accepted by the Faculty of Science, Chulalongkorn University in Partial
Fulfillment of the Requirement for the Master of Science

..... Dean of the Faculty of Science
(Professor Polkit Sangvanich, Ph.D.)

THESIS COMMITTEE

..... Chairman
(Associate Professor Anusorn Chonweerayuth, Ph.D.)

..... Thesis Advisor
(Associate Professor Rajalida Lipikorn, Ph.D.)

..... Examiner
(Assistant Professor Krung Sinapiromsaran, Ph.D.)

..... External Examiner
(Suriya Natsupakpong, Ph.D.)

5971980923 : MAJOR APPLIED MATHEMATICS AND COMPUTATIONAL SCIENCE

KEYWORD: cloud removal, HSI color space, satellite image, image processing

Thanet Markchom : THIN CLOUD REMOVAL USING LOCAL MINIMIZATION
AND IMAGE TRANSFORMATION IN HSI COLOR SPACE. Advisor: Assoc. Prof.
Rajalida Lipikorn, Ph.D.

In the use of satellite images, clouds can be an obstacle due to their opacity property that can block the visibility of ground objects in case of thick clouds and can also be blended with the underlying details in case of thin clouds. In this research, we propose a novel method to remove thin clouds and retrieve the actual information by taking the advantage of HSI color space instead of RGB color space. The proposed method aims to estimate the cloud appearance called the scattering light, remove thin clouds from intensity channel to avoid an effect to the original color, adjust the intensity with gamma correction to recover some information accidentally removed from the previous step, and restore obscure details by using contrast limited adaptive histogram equalization to enhance the intensity result. Furthermore, since thin clouds also affect saturation channel, we propose to increase saturation reduced as the result from thin clouds by using logarithm image transformation as well. From the results, the proposed method can remove clouds that are not extremely opaque and preserve the actual information such as color and texture according to the lowest mean squared error values between the outputs and their cloud-free images in the same position taken at the closest time, and the output images and their contrast gained values when compared to those obtained from other single-image methods.

Field of Study: Applied Mathematics and Computational Science Student's Signature

Academic Year: 2018 Advisor's Signature

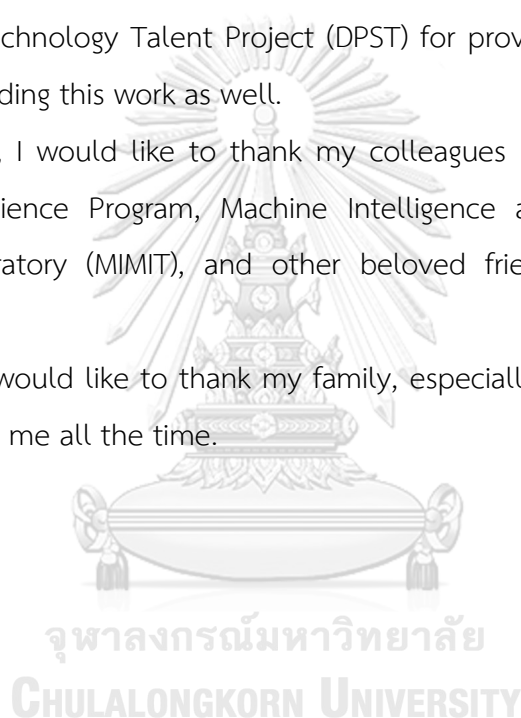
ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my thesis advisor, Associate Professor Rajalida Lipikorn, for giving me the great opportunity to pursue my master's degree under her supervision, and supporting me throughout this work. This thesis would have never been accomplished without her continuous guidance, motivation and encouragement.

I also would like to appreciate the support from Development and Promotion of Science and Technology Talent Project (DPST) for providing the scholarship for my education and funding this work as well.

Moreover, I would like to thank my colleagues in Applied Mathematics and Computational Science Program, Machine Intelligence and Multimedia Information Technology Laboratory (MIMIT), and other beloved friends who always help and encourage me.

Finally, I would like to thank my family, especially my parents, for supporting and understanding me all the time.



Thanet Markchom

TABLE OF CONTENTS

| | Page |
|---|------|
| | iii |
| ABSTRACT (THAI)..... | iii |
| | iv |
| ABSTRACT (ENGLISH)..... | iv |
| ACKNOWLEDGEMENTS..... | v |
| TABLE OF CONTENTS..... | vi |
| TABLE OF FIGURES..... | 1 |
| LIST OF TABLES..... | 3 |
| CHAPTER I Introduction..... | 4 |
| 1.1 Objectives..... | 6 |
| 1.2 Scopes and Assumptions..... | 7 |
| CHAPTER II Background Knowledge and Literature Reviews..... | 8 |
| 2.1 Background Knowledge..... | 8 |
| 2.1.1 HSI color space..... | 8 |
| 2.1.2 Image enhancement in spatial domain..... | 11 |
| 2.1.2.1 Power-Law function..... | 12 |
| 2.1.2.2 Logarithm function..... | 14 |
| 2.1.3 Histogram processing..... | 15 |
| 2.1.3.1 Histogram equalization..... | 16 |
| 2.1.3.2 Adaptive histogram equalization..... | 17 |
| 2.1.3.3 Contrast limited adaptive histogram equalization..... | 20 |

| | |
|--|----|
| 2.1.4 Cloud physical model..... | 21 |
| 2.2 Literature reviews..... | 23 |
| CHAPTER III Proposed Method..... | 26 |
| 3.1 Thin cloud removal step..... | 26 |
| 3.1.1 Scattering light term estimation..... | 26 |
| 3.1.2 Atmospheric light estimation | 29 |
| 3.1.3 Primary ground reflectance recovery | 31 |
| 3.1.4 Real ground reflectance recovery | 32 |
| 3.1.5 Real ground reflectance contrast enhancement | 33 |
| 3.2 Ground reflectance enhancement step..... | 34 |
| CHAPTER IV Results and Discussion..... | 36 |
| Chapter V Conclusions | 54 |
| 5.1 Future work | 55 |
| REFERENCES | 57 |
| VITA..... | 59 |

TABLE OF FIGURES

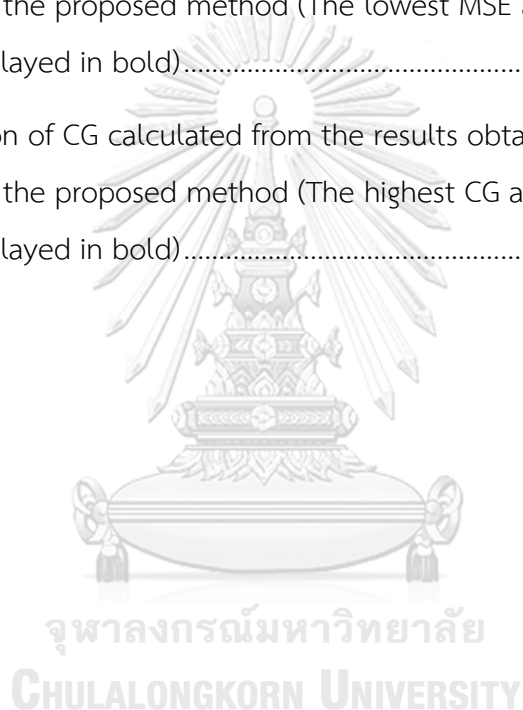
| | Page |
|---|------|
| Figure 1 Model of HSI color space..... | 9 |
| Figure 2 Plots of power-law curves with vary $\gamma = 0.25, 0.5, 0.75, 1, 1.25, 1.5,$ and 1.75 | 13 |
| Figure 3 Plots of logarithm curves with varying $c = 1, 1.5, 2, 2.5$ and 3 | 14 |
| Figure 4 An example of a gray scale image (left) and its corresponding histogram (right) | 15 |
| Figure 5 An example of the result image (left) and its corresponding histogram (right) after applying histogram equalization | 17 |
| Figure 6 An example image partitioned into small tiles with four regions (black, blue, green and red regions) for different interpolations | 19 |
| Figure 7 An illustration of clipping pixels that are over the selected maximum value and are redistributed to each gray-level | 21 |
| Figure 8 Cloud physical model of thin cloud distortion in a satellite image | 22 |
| Figure 9 Example of a process of the scattering light term estimation | 28 |
| Figure 10 An example of the final result of the scattering light term estimation process..... | 29 |
| Figure 11 (a) An RGB clouded image, (b) the corresponding intensity, (c) the corresponding estimated scattering light, (d) the intensity after subtraction, and (e) the primary ground reflectance intensity obtained from the method. | 31 |
| Figure 12 (a) An input clouded image, (b) the primary recovered ground reflectance image, and (c) the real ground reflectance image..... | 33 |
| Figure 13 The contrast-enhanced real ground reflectance | 34 |
| Figure 14 (a) The original saturation of clouded input image (b) The enhanced saturation after the ground reflectance enhancement step..... | 35 |

Figure 15 The CG values of the result images obtained from Liu's method, He's method, and the proposed method..... 50



LIST OF TABLES

| | Page |
|---|-------------|
| Table 1 A set of Landsat 8 satellite images used in the experiment..... | 36 |
| Table 2 The result images obtained from the proposed method and other single- image methods | 42 |
| Table 3 Comparison of MSE calculated from the results obtained from Liu’s method, He’s method, and the proposed method (The lowest MSE among the results in each input image is displayed in bold)..... | 47 |
| Table 4 Comparison of CG calculated from the results obtained from Liu’s method, He’s method, and the proposed method (The highest CG among the results in each input image is displayed in bold)..... | 48 |



CHAPTER I

Introduction

During the past decade, remote sensing technologies have been developing and have been using widely in many fields such as exploration, environment, military, civil engineering, and agriculture. These technologies usually involve with either aerial images taken by an aircraft or satellite images taken by a satellite depending on user's purposes and conveniences.

Satellite images are very useful in remote sensing. They are taken by the satellites that are launched around the world. Each satellite has different kind of sensor or device to receive electromagnetic radiation reflected from the ground, so it is possible to collect different types of satellite images not only in visible colors but also in other spectra such as infrared, ultraviolet, or microwave. The most important advantage of satellite images is the capability to collect data in wide areas from further distance when compared to aerial images, and data in those target areas are also updated precisely and continuously according to a constant orbit. However, since satellite images are taken from outer space through the atmosphere, the cloud appearance within satellite images is occasionally unavoidable. Therefore, the cloud appearance is one of the problems when satellite images are utilized.

A cloud is a hydrometeor consisting of liquid water, or ice, or both, suspended in the atmosphere. It probably includes larger particles of liquid water or ice, as well as some solid particles such as smoke or dust. Clouds form in every height level in the atmosphere so clouds can be distinguished by the heights above ground level at which they form. There are three main levels of cloud in the atmosphere: high-level, mid-level, and low-level. High-level clouds generally have a base at altitudes of 3,000 to 7,600 meters in the polar areas, 5,000 to 12,200 meters in the temperate regions, and 6,100 to 18,300 meters in the tropical region. Mid-level clouds appear from 2,000 meters above the Earth's surface and probably based as high as 4,000 meters around polar areas, 7,000 meters at mid latitudes, and 7,600

meters in the tropics. Meanwhile, low-level clouds are typically found at the height of 2,000 meters above the Earth's surface.

Clouds continuously transform and appear in various characteristics. However, clouds that are frequently observed can be classified into ten main genera and grouped into three previously mentioned levels: Cirrus, Cirrocumulus and Cirrostratus in high-level, Altcumulus and Altostratus in mid-level, Stratocumulus and Stratus in low-level, and Nimbostratus, Cumulus and Cumulonimbus which can be formed in all levels.

Besides height levels and cloud characteristics, clouds can be classified by opacity as well. Opacity-based classification consists of three types: opacus, perlucidus and translucidus. Opacus is a thick sheet of clouds, perlucidus is a sheet of clouds consisting of thick clouds patches with small translucent spaces between each patches, and translucidus is a thin translucent patch or sheet of clouds. Opacity-based varieties cannot be used to classify high-level clouds which are always translucent naturally as well as Cumulus, Cumulonimbus and Nimbostratus which are always opaque. For the rest of genera, Altcumulus, Altostratus, Stratocumulus, and Stratus, can appear in the form of either opacus or translucidus, depending on the quantity of their particles. In other words, if their particle densities are low, clouds are transparent. Conversely, if their particle densities are high, clouds are opaque. Therefore, it is possible to consider thin clouds as these genera in the form of translucidus. On the other hand, thick clouds as these genera in the form of opacus.

In the development of remote sensing technology, clouds are still a problem due to their opacity property which can block the visibility of ground objects and can also distort the actual data in case of thin clouds; therefore, a number of cloud removal methods have been proposed in order to obtain the real information without thin cloud degradation. These methods aim to eliminate cloud effect and retrieve the actual data with different prerequisites such as multiple input images taken by specific wavelengths or input images in the same position taken at different times. However, without complicated requirements, some methods can be used by using only a single input image itself, and they are called single-image methods. This

kind of methods is easier to be implemented and applied due to its lesser requirements. Most single-image methods usually involve with an image in RGB format or RGB color space which consists of three channels: red, green, and blue channels. These three channels represent the signals from objects in an image that response to each of the three wavelength ranges; i.e., red, green, and blue in white light. To remove thin clouds in an RGB image, each channel has to be processed individually before they are combined to form an RGB image. However, each channel of an RGB image is suffered differently by thin clouds because of the different scattering phenomenon in each wavelength of white light radiated from the sun. Hence, each channel should be considered separately to yield a better result; in other words, the additional steps, such as determining parameters depending on each channel, are needed in the process. As a result, it will increase the complexity of the method.

Besides RGB color space, there are many other color spaces that were proposed for different purposes. One of these color spaces is HSI color space which was designed to represent colors in a similar way to human vision. In this color space, there are three attributes; i.e., hue, saturation, and intensity. According to the observation, the intensity and the saturation relate to clouds due to the characteristics of clouds which are white and bright when appearing in an RGB image; hence, it is possible to reduce an effect of thin clouds and recover the clouded information by adjusting and enhancing these attributes in HSI color space instead of red, green and blue attributes in RGB color space as usual. Therefore, in this work, we propose a novel single-image cloud removal method based on HSI color space instead of RGB color space which is less complicated to implement and easier to use than other RGB-based methods.

1.1 Objectives

To propose an algorithm which can effectively remove thin clouds and enhance details distorted by thin clouds from satellite images and still retain important information such as texture and color

1.2 Scopes and Assumptions

1. This work focuses on removing thin clouds, particularly transparent clouds that we are able to see land information under the covered area and appear in an image.
2. The proposed method accepts a satellite image containing both thick clouds and thin clouds as an input.
3. The proposed method can preserve white objects such as roads, buildings and thick clouds that are nontransparent during the thin cloud removal process.
4. The details under thin clouds can be enhanced in term of texture; i.e., object edges and contrast between surrounding pixels, and also color.
5. The satellite images used in this work are taken above several areas in Thailand by Landsat 8.
6. The satellite images used in this work are RGB color images with .jpg format.



CHAPTER II

Background Knowledge and Literature Reviews

2.1 Background Knowledge

2.1.1 HSI color space

The most fundamental color space for color images is RGB color space which represents colors using a mixture of three components: red, green, and blue. These basic components can be combined together with different proportion to display each color. However, RGB is occasionally unsuitable for some applications, thus other color spaces are developed for this reason. In image processing field, many color spaces have been proposed and utilized for various objectives. One of them is HSI color space which was developed in the 1970s for describing colors as close as the way human vision perceives color-making attributes. Naturally, colors can be seen as a combination of the original color, the purity of it and the lightness. For example, if a pure red light is fused with a white light, it will be seen as a fade red color due to the loss of its purity. Meanwhile, the lightness can brighten the original color up to the completely white color when it is increased, and can alternately darken the original color when it is decreased. This phenomenon is presented in HSI color space with three attributes; i.e. hue, saturation, and intensity. Hue is the color attribute used to represent a pure color. It is determined by an angle from the reference point, which is usually a red color at 0° , between 0° to 360° . Saturation refers to the color purity or how much a pure color is diluted by a white light. The maximum of the saturation is 1 which means a color is absolutely pure without a white light mixture, and the minimum is 0 which means the original color is fully diluted and becomes absolutely white. Intensity is the amount of brightness. It is in the range between 0 and 1 where 0 implies the darkest black, and 1 implies the brightest white.

Figure 1 shows the model of HSI color space that is represented by circular color planes. The circles are perpendicular to the vertical intensity axis. The hue of any point is determined by an angle from the red axis designated as 0° , and the hue

increases counterclockwise from there. The saturation is the length of the vector from the origin to a point.

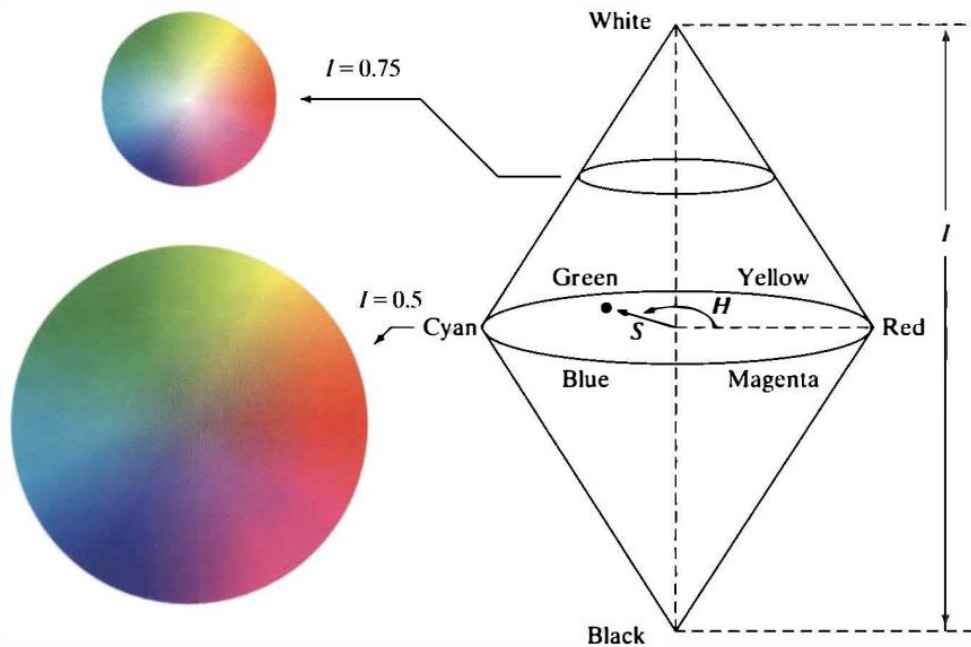


Figure 1 Model of HSI color space

Source: Adapted from [1]

Conversion from RGB color space to HSI color space is defined as follows:

$$H = \begin{cases} \theta & ; B \leq G \\ 360^\circ - \theta & ; B > G \end{cases} \quad (1)$$

$$\theta = \cos^{-1} \left\{ \frac{\frac{1}{2}((R-G) + (R-B))}{\left((R-G)^2 + (R-B)(G-B) \right)^{\frac{1}{2}}} \right\}$$

$$S = 1 - \frac{3}{(R+G+B)} \min\{R, G, B\} \quad (2)$$

$$I = \frac{1}{3}(R+G+B) \quad (3)$$

where H , S , and I are the hue, the saturation, and the intensity components of each pixel in HSI color space such that $0^\circ \leq H \leq 360^\circ$, $0 \leq S \leq 1$ and $0 \leq I \leq 1$, whereas, R , G , and B are the red, the green, and the blue components of each pixel in RGB color space which are normalized to the range $[0,1]$, respectively. It should be noticed that the hue component can be normalized by dividing by 360° as well, and the other HSI components are already in the range $[0,1]$ with the given normalized R , G , and B .

Similarly, an image in HSI color space can be converted back to RGB color space differently depending on the range of the hue which is divided into three sectors; i.e., RG sector, GB sector, and BR sector by the following equations.

RG sector ($0^\circ \leq H < 120^\circ$)

$$R = I \left(1 + \frac{S \cos(H)}{\cos(60^\circ - H)} \right) \quad (4)$$

$$G = 3I - (R + B) \quad (5)$$

$$B = I(1 - S) \quad (6)$$

GB sector ($120^\circ \leq H < 240^\circ$)

$$R = I(1 - S) \quad (7)$$

$$G = I \left(1 + \frac{S \cos(H - 120^\circ)}{\cos(60^\circ - (H - 120^\circ))} \right) \quad (8)$$

$$B = 3I - (R + G) \quad (9)$$

BR sector ($240^\circ \leq H < 360^\circ$)

$$R = 3I - (G + B) \quad (10)$$

$$G = I(1 - S) \quad (11)$$

$$B = I \left(1 + \frac{S \cos(H - 240^\circ)}{\cos(60^\circ - (H - 240^\circ))} \right) \quad (12)$$

Since HSI color space decouples the intensity and the saturation from the original color value (hue), it has a huge advantage when the color information is needed to be excluded or to be considered separately during a process.

2.1.2 Image enhancement in spatial domain

Image enhancement is basically a process for improving the original image in order to increase interpretability or perception of information so that the result will be more suitable for some specific purposes. Image enhancement methods can be performed in both spatial domain and frequency domain. In spatial domain, the image pixels are directly managed with a chosen transformation function and transformed to the result image which can be mathematically expressed as follows:

$$g(x, y) = T[f(x, y)] \quad (13)$$

where $f(x, y)$ denotes the original image, $g(x, y)$ denotes the transformed result image or the enhanced image, and T denotes a transformation operator on the original image $f(x, y)$.

Image enhancement in spatial domain can be performed on each pixel by applying an operator to a value of each pixel individually or applying an operator to each pixel's neighborhood. Normally, a neighborhood is defined by a square or a rectangle subimage area centered at a pixel (x, y) , which can be called a patch. A transformation operator T is applied at each location (x, y) to yield the output, $g(x, y)$, by considering only the pixels inside a neighborhood or a patch. In case that a neighborhood is of size 1×1 , $g(x, y)$ depends on only a pixel value at a position (x, y) of an image f , and the transformation can be seen as a gray-level or intensity transformation expressed as

$$s = T(r) \quad (14)$$

where r and s are variables denoting the gray-level of $f(x, y)$ and $g(x, y)$ at any point (x, y) , respectively.

In any color space, an image is composed of various components depending on a color space. Each component is separately presented in an individual image, which may be referred to as a channel. Therefore, without combining all of these images or channels together, each of them can be interpreted as a gray scale image containing gray scale values so that a gray-level transformation can be applied to each component of a color image to produce an enhanced result for each component before combining with other components of an image and displaying as a complete color image.

There are several types of functions that are used as transformation functions. Two functions are used in the proposed method; i.e., power-law and logarithm function.

2.1.2.1 Power-Law function

Power-law transformation function has the basic form:

$$s = r^\gamma \quad (15)$$

where γ is a positive constant.

Plots of s versus r for various values of γ are shown in Figure 2. Power-law curves with fractional values of γ map a narrow range of dark input values into a wider range of output values, with the opposite being true for higher input; that is, the darker parts in an input image will be brightened up and the contrast among those similar dark pixels will also be increased. Thus, the details that are used to be dim can be brighter and clearer, while the brighter parts are also brightened with the lower rate compared to the darker parts. Contrary, as shown in Figure 2, curves generated with values of $\gamma > 1$ have the opposite results from those generated with

values of $\gamma < 1$; in other words, input values will be mapped to lower values instead of higher values as in the case of fractional values of γ .

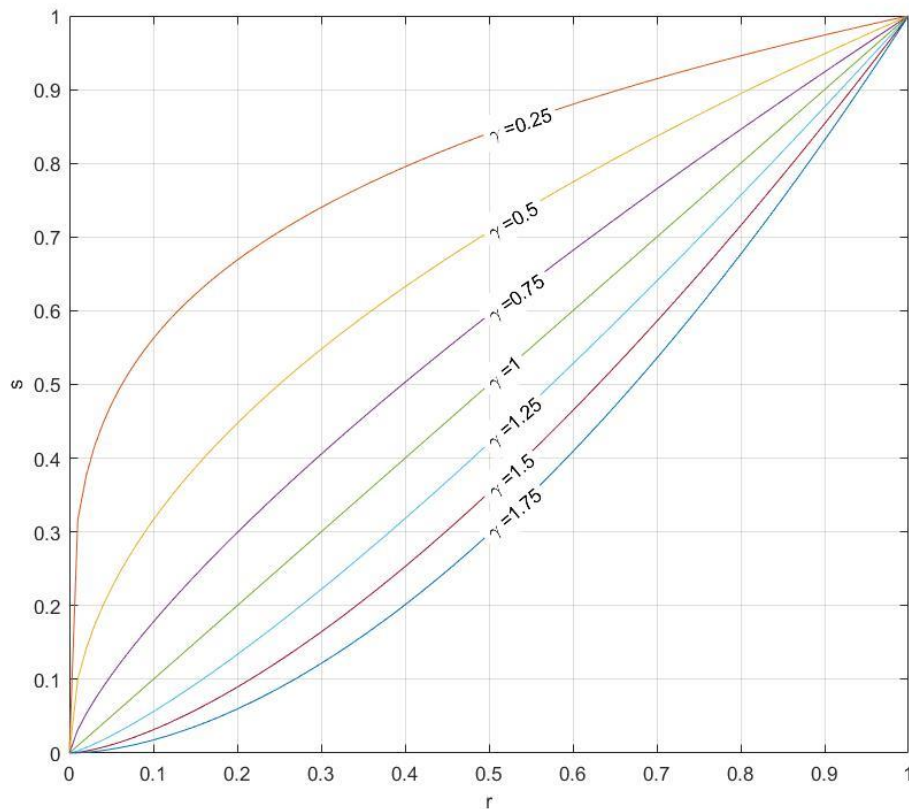


Figure 2 Plots of power-law curves with $\gamma = 0.25, 0.5, 0.75, 1, 1.25, 1.5,$ and 1.75

It should be noted that the exponent term in Eq.(15) is referred to as gamma by convention, thus the process that uses power-law transformation by correcting the exponent to obtain the desired result is called gamma correction which will be referred to as a power-law transformation in this work.

2.1.2.2 Logarithm function

The basic form of logarithm transformation function is

$$s = c \log(1+r) \quad (16)$$

where c is a positive constant. Logarithm transformation maps a narrow range of low gray-level values in an input image into a wider range of output levels. The opposite is true of higher values of input levels. This transformation is suitable for increasing the values of dark pixels and expanding the differences among those pixel values to gain contrast in an image while compressing the higher-level values.

The curves of logarithm transformation with varying c are shown in Figure 3.

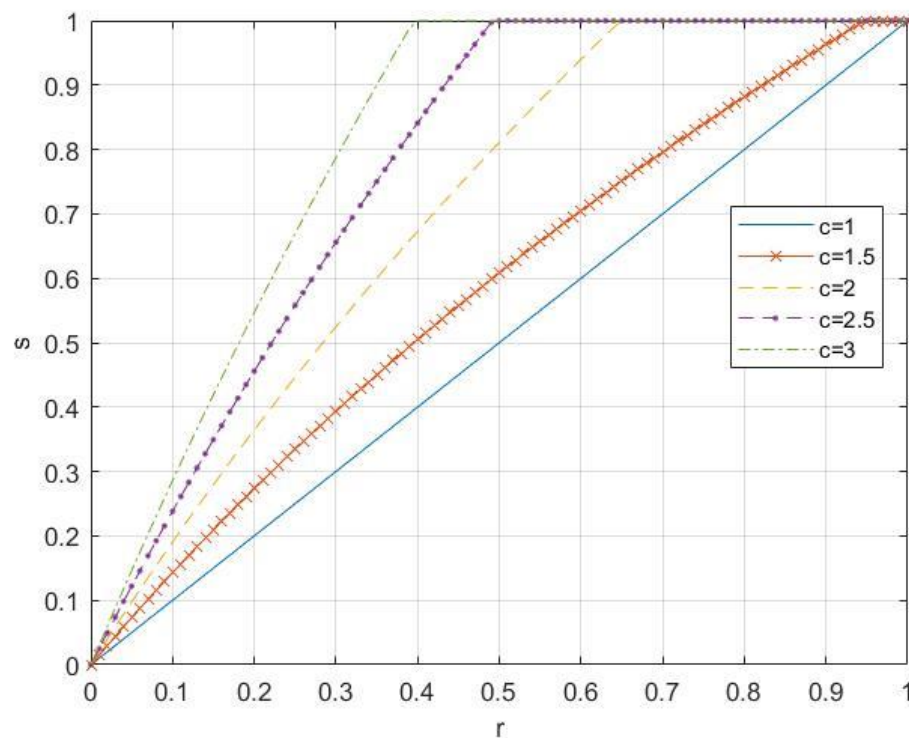


Figure 3 Plots of logarithm curves with $c = 1, 1.5, 2, 2.5$ and 3

2.1.3 Histogram processing

An image can be represented by a histogram which plots the number of pixels for each gray-level value. The horizontal axis of a histogram represents gray-level values depending on a type of an image. For example, if an image contains G gray-level values, i.e., $0, 1, 2, \dots, G-1$, then the horizontal axis of a histogram represents value $0, 1, 2, \dots, G-1$, while the vertical axis represents the number of pixels in each gray-level value. Figure 4 shows an example of a gray scale image which contains 256 possible gray-level values from 0 to 255 (left) and its corresponding histogram (right).

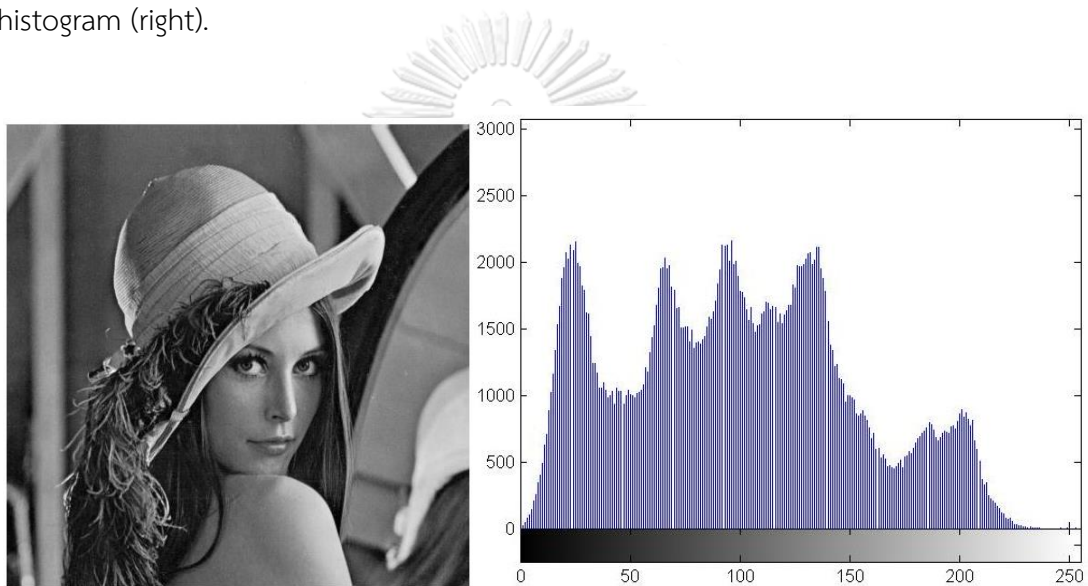


Figure 4 An example of a gray scale image (left) and its corresponding histogram (right)

The histogram of an image with G total possible gray-level values can be defined as the discrete function

$$h(r_k) = n_k \quad (17)$$

where r_k is the k th gray-level value for $k = 0, 1, 2, \dots, G-1$, and n_k is the number of pixels in an image whose gray-level is r_k .

An image histogram can be used for many purposes, such as enhancement, compression, segmentation, and description. The processes that involves with an image histogram is called a histogram processing.

2.1.3.1 Histogram equalization

One of the techniques that uses a histogram to enhance the contrast is histogram equalization. This process increases the contrast by spreading out the most frequent gray-level values, i.e. stretching out the gray-level range of an image, so that after applying this process, the gray-level values will be more distributed on a histogram, and the difference between pixels with similar gray-level values will be more obvious. This technique uses a cumulative distribution function (CDF) as a transformation function, T , to map an original gray-level value denoted by r_k to a new gray-level value denoted by s_k as follows:

$$s_k = T(r_k) = \left\lfloor (G-1) \sum_{i=0}^k p(r_i) \right\rfloor \quad (18)$$

for $k=0, 1, 2, \dots, G-1$, where G denotes the total number of gray-level value and $p(r_j)$ denotes the estimated probability of occurrence of gray-level value r_j which can be computed by

$$p(r_j) = \frac{h(r_j)}{n} \quad (19)$$

where n denotes the total number of pixels in an image.

Figure 5 shows the result of applying histogram equalization on an image shown in Figure 4 (left), and its corresponding histogram (right) that is more distributed than the original histogram as the result of histogram equalization.

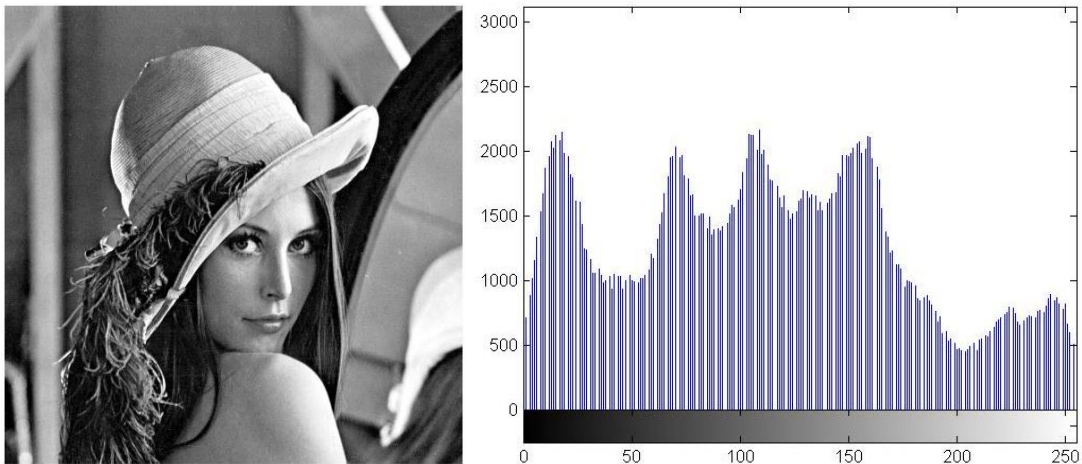


Figure 5 An example of the result image (left) and its corresponding histogram (right) after applying histogram equalization

2.1.3.2 Adaptive histogram equalization

Instead of computing histogram equalization for an entire image, adaptive histogram equalization (AHE) computes several histograms, each corresponding to a distinct section of an image called a tile, and uses them to redistribute the gray-level values in each tile. It is; therefore, suitable for improving the local contrast and enhancing the definition of edges in each region of an image. This technique consists of the following steps:

Step 1: an image is partitioned into equally sized rectangular tiles as shown on the right part of Figure 6.

Step 2: In each tile, histogram equalization is applied and a transformation function for each tile will be generated.

Step 3: Each pixel will be mapped by using up to four transformation functions from its neighborhood tiles depending on its position.

- I. For the tile center pixels displayed as black squares in the left part of Figure 6, the transformation functions derived from the tiles that these pixels belong are the most appropriate, and can be used normally.
- II. For pixels in the blue part of the image shown in Figure 6, they are bilinearly interpolated between four mapped values obtained from

four transformation functions derived from the tiles with center pixels closest to them.

Suppose that a pixel (x, y) is in the blue part of the image in Figure 6. Let (x_0, y_0) , (x_0, y_1) , (x_1, y_0) and (x_1, y_1) be the center pixels of the tiles closest to a pixel (x, y) ; and T_0 , T_1 , T_2 , and T_3 be the transformation functions derived from histogram equalization for the tiles with center pixels (x_0, y_0) , (x_0, y_1) , (x_1, y_0) , and (x_1, y_1) , respectively. The interpolated gray-level value of a pixel (x, y) denoted by s_k is

$$s_k = \frac{y_1 - y}{y_1 - y_0} \left(\frac{x_1 - x}{x_1 - x_0} T_0(r_k) + \frac{x - x_0}{x_1 - x_0} T_2(r_k) \right) + \frac{y - y_0}{y_1 - y_0} \left(\frac{x_1 - x}{x_1 - x_0} T_1(r_k) + \frac{x - x_0}{x_1 - x_0} T_3(r_k) \right) \quad (20)$$

where r_k is a gray-level value at a pixel (x, y) .

- III. For pixels that are close to the shaded green boundary in Figure 6, they are linearly interpolated between two mapped values obtained from two transformation functions derived from the tiles with center pixels closest to them.

Suppose that a pixel (x, y) is in the green region and in either the first or the last row of an image on the right side of Figure 6. Two tiles closest to a pixel (x, y) are aligned horizontally. Their center pixels are denoted by (x_0, y_0) and (x_0, y_1) , and their transformation functions generated from histogram equalization are T_0 and T_1 , respectively. The interpolated gray-level value of a pixel (x, y) is denoted by s_k , and can be computed as follows:

$$s_k = T_0(r_k) \left(1 - \frac{y - y_0}{y_1 - y_0} \right) + T_1(r_k) \left(1 - \frac{y_1 - y}{y_1 - y_0} \right) \quad (21)$$

where $r_{(x,y)}$ is a gray-level value at a pixel (x, y) .

For a pixel (x, y) in the green region in the first and the last column of the image on the right side of Figure 6, two tiles closest to a pixel (x, y) are aligned vertically. Their center pixels are denoted by (x_0, y_0) and (x_1, y_0) , and their transformation functions generated from histogram equalization are T_0 and T_1 , respectively. the interpolated gray-level value, $s_{(x,y)}$, can be similarly computed by

$$s_k = T_0(r_k) \left(1 - \frac{x - x_0}{x_1 - x_0} \right) + T_1(r_k) \left(1 - \frac{x_1 - x}{x_1 - x_0} \right) \quad (22)$$

- IV. Pixels near corners (shaded red in Figure 6) are transformed with the transformation function of the corner tile where they belong as same as the center pixels.

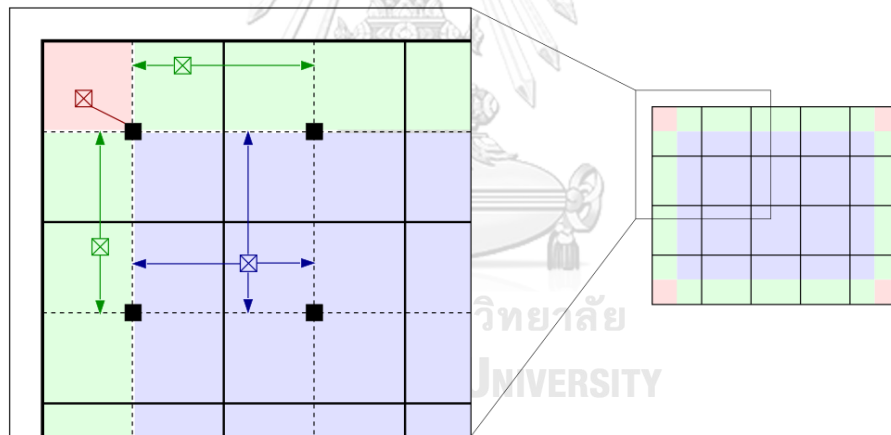


Figure 6 An example image partitioned into small tiles with four regions (black, blue, green and red regions) for different interpolations

Source: Adapted from [2]

This technique is highly suitable when an input image contains regions that are significantly lighter or darker than other regions in an image, and the contrast in those regions will not be enhanced sufficiently when using ordinary histogram equalization.

2.1.3.3 Contrast limited adaptive histogram equalization

Since AHE technique has a tendency to over-amplify noise in relatively homogeneous regions of an image; therefore, contrast limited adaptive histogram equalization (CLAHE) [3] is presented in order to prevent this problem by limiting the contrast amplification before the transformation function in each divided region is derived.

CLAHE method applies histogram equalization to each tile the same way as AHE, but limits the amplification by clipping the histogram at a predefined value and redistributing the clipped pixels to each gray-level value as illustrated in Figure 7 before computing the CDF for each tile. This clipping limits the slope of the CDF and that of the transformation function to avoid noise enhancement. The value at which the histogram is clipped is called the clip limit and can be calculated by

$$n_{CL} = n_{clip} \times \frac{n}{n_{gray}} \quad (23)$$

where n_{CL} is an actual clip limit, n_{clip} is a constant parameter in the range [0,1] indicating a normalized clip limit, n is the number of pixels in a tile and n_{gray} is the number of gray-level values in a tile. If the number of pixels in each gray-level value is greater than n_{CL} , the pixels will be clipped and the total number of clipped pixels will be distributed to each gray-level value equally. After distribution, if the number of pixels is over the clip limit illustrated as the green region on the right histogram of Figure 7, these pixels will be redistributed until the remaining pixels over the clip limit have been all distributed. Lastly, AHE will be applied to the distributed histogram to derive the transformation function and compute the new gray-level value as already mentioned.

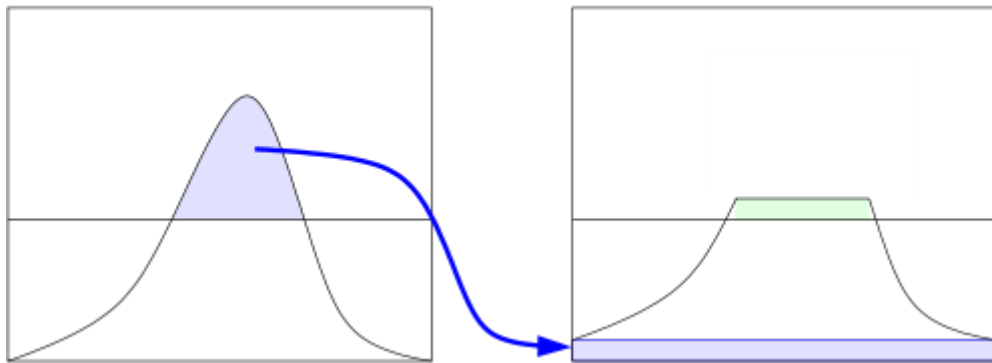


Figure 7 An illustration of clipping pixels that are over the selected maximum value and are redistributed to each gray-level

Source: Adapted from [2]

2.1.4 Cloud physical model

A cloud is a hydrometeor consisting of liquid water or ice, or both that suspends in the atmosphere. It probably includes larger particles of liquid water or ice, as well as some solid particles, such as smoke or dust. When satellite images are taken above a cloud covered area and if clouds are thick enough, sunlight will not be able to pass through those clouds, so the ground reflectance cannot be detected by a satellite. Similarly, although thin clouds are transparent and allow sunlight to pass through, some of the sunlight will be absorbed, and the transmitting light will be attenuated after all. Sunlight is also scattered by cloud particles, reflected back and causes the appearance of clouds as different level of white or gray sheets in an image. Thus, each clouded image is a combination of the ground reflectance from attenuated solar radiation and the scattering light caused by clouds which is illustrated in Figure 8.

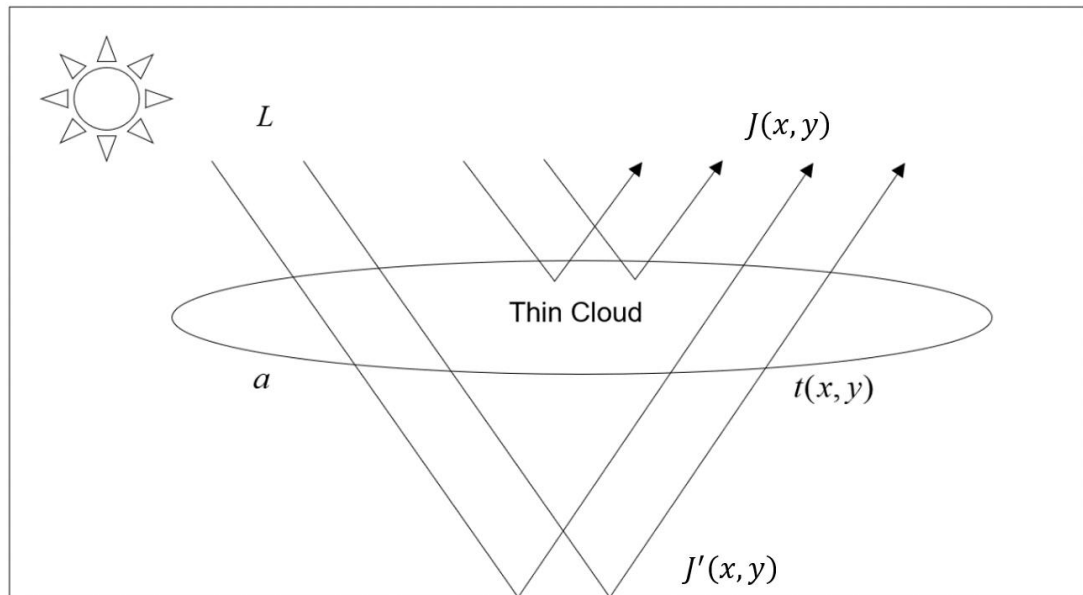


Figure 8 Cloud physical model of thin cloud distortion in a satellite image

By observing the effect of clouds in a satellite image, a cloud physical model [4] was proposed to describe a cloud distortion process as follows:

$$J(x, y) = aLJ'(x, y)t(x, y) + L(1 - t(x, y)) \quad (24)$$

where $J(x, y)$ is a clouded image, x and y are the pixel coordinates, L is the atmospheric light brightness, $J'(x, y)$ is the real reflectance of the ground object, $t(x, y)$ is the cloud transmission, and a is the attenuation coefficient of sunlight. The range of $J'(x, y)$, $t(x, y)$, L , and a is from 0 to 1. Considering the right hand side of Eq.(24), the first term $aLJ'(x, y)t(x, y)$ denotes the reflectance of the ground object which is incident by attenuated sunlight, aL , and passes through clouds with the rate of cloud transmission, $t(x, y)$; i.e., if the cloud transmission is maximum, the light can completely pass through, and the real ground reflectance is unaffected by the cloud transmission $t(x, y)$. In contrary, if the clouds have a thickness or opacity, the cloud transmission will be decreased, which means the light cannot pass through clouds, and will be absorbed and also reflected back partially, and will cause the white bright cloud appearance in an image. This reflected light is

represented in this model by the second term, $L(1-t(x, y))$, which can be called the scattering light term. When clouds are very thick, $t(x, y)$ approaches 0, and the scattering light will be as much bright as the atmospheric light brightness. On the other hand, if clouds are transparent or even do not exist, $t(x, y)$ will approach 1 and cause $L(1-t(x, y))$ to approach 0; that is, the whole light can pass through clouds and there is no scattering light appearing in an image.

According to this model, several cloud removal methods have been proposed in order to obtain an image without cloud degradation by recovering $J'(x, y)$ from a satellite image $J(x, y)$.

2.2 Literature reviews

Many cloud removal methods have been proposed over the last decade. Each method can be used with different inputs and requirements. These methods can be categorized by their required inputs into three groups: single-image methods, multi-spectral-image methods, and multi-temporal-image methods.

Single-image methods normally use only the data containing in an image itself. These methods usually consider clouds as noises and try to eliminate them by some techniques such as adaptive homomorphic filter [5] or Gaussian low-pass filter [6].

Unlike single-image methods, multi-spectral-image methods need multiple types of spectral images taken in various specific wavelengths such as images taken in cirrus band from Landsat 8 [7] which were used to correct the contamination of cirrus clouds in an image by using the relationship with other images taken by visible wavelength or infrared. However, cirrus band is available only for Landsat satellite, so this method has a massive condition for usage as well as other methods that need uncommon spectral images.

Similarly, multiple images are also needed in multi-temporal-image methods. These methods basically use partial cloud-free images from different periods to generate an image without clouds. For instance, satellite image time series succeed to generate cloud-free image by using Sparse Unmixing-based Denoising (SUBD) [8]. This method estimates the new value of a pixel covered by clouds from the

evolution in time of the values in non-corrupted pixels around it. Although most multi-temporal-image methods usually generate an image without any damages to the actual data because their processes only adjust clouded pixels while leaving other non-clouded pixels, but cloud-free images may not always be available in the expected areas.

In fact, single-image methods are usually not able to compete with multi-spectral-image and multi-temporal-image methods in term of effectiveness; however, due to ease of use and implementation, the results obtained from most single-image methods are quite acceptable compared to other types. On the other hand, single-image methods are still popular due to their simplicity with less preliminary conditions comparing to the other types since they use only the data containing in a single image which is commonly taken in a visible wavelength that almost every satellite is capable to operate.

Most of single-image methods usually follow the idea of the cloud physical model previously mentioned. Liu et al. [6] utilized the low-frequency characteristic of clouds in frequency domain. According to their work, areas covered by clouds are related to low-frequency components in frequency domain due to the fact that clouds generally appear as a white sheet and cause the underlying details to become blur or be blended with other details around them. As a result, they considered clouds as noises which are presented as the scattering light term in the model, and attempted to estimate this term using a low-pass filter in frequency domain. They suggested to apply Gaussian low-pass filter to three channels of a clouded RGB image separately, then defined each of them as approximated scattering light term for each channel, and subtract them from the original image. The result shows that most of the scattering lights are excessively removed, thus adding an offset value to retrieve a proper result is needed. The mean value of each band is chosen as an offset, and is added to each band. However, using the mean values as offsets is still not suitable for some cases, especially when clouds are very few, or spread partially in an image.

In the same year, Shen et al. [5] also considered thin clouds as low-frequency information and proposed a method based on the classic homomorphic filter, the

technique uses a logarithm property and high-pass filter in frequency domain to filter low-frequency components out while preserve only high-frequency components which correspond to the ground details. Moreover, since each channel in an RGB image is affected differently by thin clouds due to the fact that light is scattered unequally when passing through thin clouds [9], they also proposed a step to semi-automatically choose parameters in each channel in RGB color space. This method with this additional step actually yielded the better results compared to previous methods that used the same parameters for every channel, but it resulted in increasing complexity in their method as well.

Other color spaces besides RGB color space can be considered for thin cloud removal as well. HSI color space is one of the color spaces which can be used to remove thin clouds. Recently, HSI color space has been used in a haze removal method proposed by Zhu, Mai and Shao [10]. They examined the correlation between intensity, saturation, and haze concentration by collecting a large number of hazy images, and observed the effect of haze on an image in HSI color space. They found that saturation of the hazy area decreases sharply, and intensity increases at the same time producing high value of the difference between intensity and saturation; that is, the haze concentration is positively correlated with the difference between intensity and saturation. Therefore, the scene depth, which relates to the concentration of haze, was estimated by linear combination of intensity, saturation, and the difference between these two components and was used to retrieve the real ground reflectance. Since the model of a hazy image is similar to the thin cloud degradation model, it is possible to use HSI color space in thin cloud removal as it was used previously.

CHAPTER III

Proposed Method

In this work, we propose a novel single-image method for thin cloud removal in HSI color space instead of RGB color space as usual. The proposed method is mainly divided into two steps: 1) thin cloud removal and 2) ground reflectance enhancement. During the first step, the method aims to remove thin clouds from an input satellite image in the intensity channel by adjusting the result with gamma correction, and then the distorted details covered by thin clouds will be enhanced by applying logarithm image transformation to the saturation channel in the second step. The final result can be produced by combining the results obtained from these two steps with the original hue and converting back to RGB color space.

3.1 Thin cloud removal step

3.1.1 Scattering light term estimation

Considering the thin cloud degradation model expressed in Eq.(24), we can denote the scattering light term by $S(x, y)$ as

$$S(x, y) = L(1-t(x, y)) \quad (25)$$

In order to obtain the real ground reflectance, Eq.(24) can be rearranged as follows:

$$aJ'(x, y) = \frac{J(x, y) - L(1-t(x, y))}{Lt(x, y)} \quad (26)$$

From Eq. (25), we have

$$aJ'(x, y) = \frac{J(x, y) - S(x, y)}{L - S(x, y)} \quad (27)$$

In this step, we propose to remove thin clouds in the intensity channel which has the same model as thin cloud degradation model and can be expressed as follows:

$$aJ'_I(x, y) = \frac{J_I(x, y) - S_I(x, y)}{L - S_I(x, y)} \quad (28)$$

where $J'_I(x, y)$, $J_I(x, y)$ and $S_I(x, y)$ denote the cloud intensity channel, the real ground reflectance and the scattering light term in the intensity channel, respectively.

The process begins with estimating the scattering light term intensity denoted by $S_I(x, y)$, and also the atmospheric light, L , so that we can substitute these values into the rearranged Eq.(28) to firstly obtain the real ground reflectance intensity with sunlight attenuation, $aJ'_I(x, y)$.

By observation, the clouded pixels usually have unusual higher intensities than the cloud-free pixels due to the appearance of clouds as a white or grey sheet in an image. Since the intensity component in HSI color space relates to the brightness, it is possible to indicate the opacity of clouds which is represented by the scattering light term, $S_I(x, y)$, from the intensity component. However, some high intensity pixels in a satellite image are probably parts of buildings, roads, or even thick clouds which should not be removed in a thin cloud removal step. In order to neglect these bright spots, we suggested to find the minimum intensity in a patch for each HSI clouded image to estimate the scattering light term as follows:

$$S_I(x, y) = \omega \left(\min_{(x', y') \in \Omega(x, y)} J_I(x', y') \right) \quad (29)$$

where $S_I(x, y)$ is the estimated scattering light term in the intensity channel, $J_I(x, y)$ is an intensity value, ω is a constant parameter which can be adjusted in order to keep some details from subtraction, and $\Omega(x, y)$ denotes a patch centered at a pixel (x, y) . When $S_I(x, y)$ represents a minimum value in a patch $\Omega(x, y)$ the

bright spots in an image will be neglected; i.e., if each bright spot in an image is smaller than the size of a patch $\Omega(x, y)$, the higher value pixels of a bright spot will not be chosen as an estimated scattering light of a pixel (x, y) , and other lower values surrounding a bright spot will be chosen instead; that is, they are neglected.

For example, Figure 9 shows an example of the process when a pixel value of the estimated scattering light is presented. The left figure is an example of a satellite image with the bright object in the center of an image, and the right figure is the estimated scattering light image. Considering the pixel in a yellow square, its 3×3 patch is shown as a red rectangle. The pixel in the same position in the estimated scattering light (the right figure) is the minimum value of the patch on the left figure. This process is applied pixel by pixel for the entire image, and the final result is shown in Figure 10 where the estimated scattering light image is completely dark, so when a satellite image is subtracted by the estimated scattering light image as expressed in the denominator of the right hand side of Eq.(28), the bright object will not be subtracted, and will be preserved.

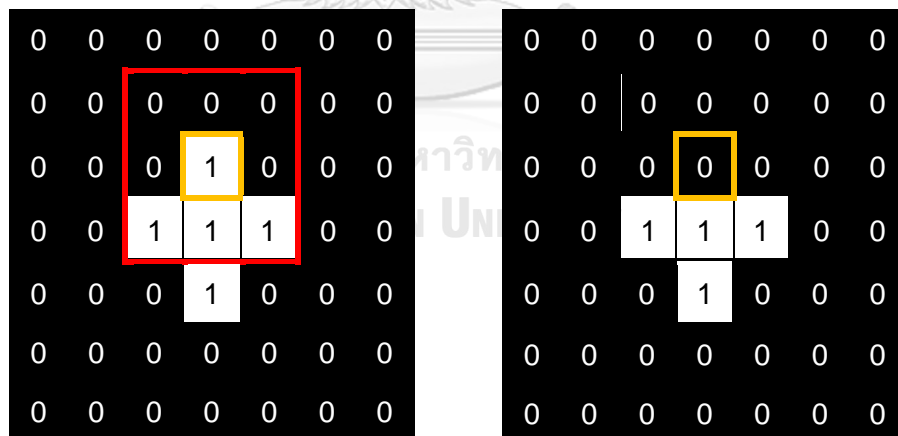


Figure 9 Example of a process of the scattering light term estimation

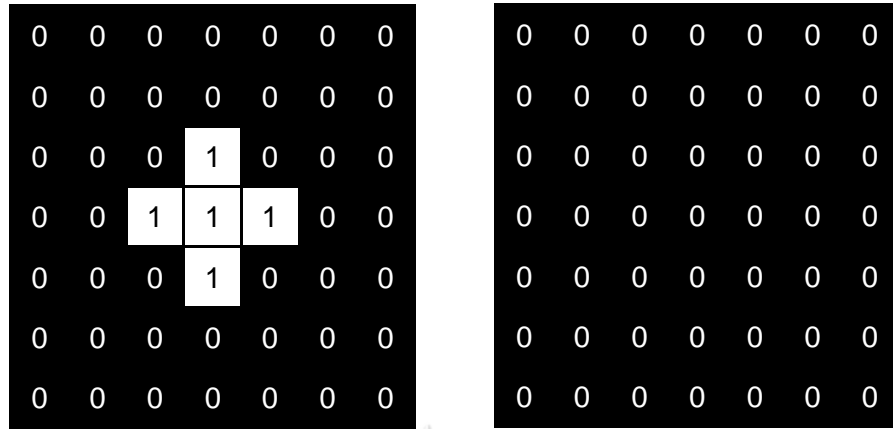


Figure 10 An example of the final result of the scattering light term estimation process

3.1.2 Atmospheric light estimation

In the previous step, the scattering light term in the intensity channel $S_I(x, y)$ is estimated first. Afterwards, the proposed method needs the atmospheric light, L , to be estimated. Many researchers have proposed distinct atmospheric light estimations; however, most of them usually follow the concept of most opaque clouded pixels which will be discussed in the following paragraph.

Suppose a pixel (x, y) is covered by the most opaque clouds, which cause $t(x, y)$ to approach 0, can be expressed mathematically as:

$$\lim_{t(x,y) \rightarrow 0} J_I(x, y) = \lim_{t(x,y) \rightarrow 0} (aLJ'_I(x, y)t(x, y) + L(1 - t(x, y))) \quad (30)$$

where the left hand side of the equation represents a pixel (x, y) in a satellite image such that it is covered by high opacity clouds, and the right hand side of the equation is from Eq.(24).

Thus, we have

$$\lim_{t(x,y) \rightarrow 0} J_I(x, y) = \lim_{t(x,y) \rightarrow 0} (aLJ'_I(x, y)t(x, y)) + \lim_{t(x,y) \rightarrow 0} L - \lim_{t(x,y) \rightarrow 0} Lt(x, y) \quad (31)$$

i.e.,

$$\lim_{t(x,y) \rightarrow 0} J_I(x, y) = L \quad (32)$$

Hence, it is possible to estimate the atmospheric light by using the intensity value of a pixel which is covered by the most opaque clouds in an image. However, finding the most opaque clouded pixels can be difficult because some bright spots such as buildings or roads can occasionally also appear in a satellite image, and their intensities are similar to thick clouds. In order to avoid this mistake, the most opaque pixel is needed to be identified implicitly. For our work, we chose to estimate the atmospheric light by taking an advantage of the estimated scattering light image already mentioned before. Since the estimated scattering light image, $S_I(x, y)$, directly correlates to the cloud thickness, and those bright spots are already neglected in this image, we therefore considered finding the most opaque pixel in this image instead of an original satellite image. However, it is possible that there are still some bright objects such as large buildings left in the estimated scattering light image, hence, we try to avoid choosing the pixels that belong to these left objects by first finding the 10% highest value pixels in $S_I(x, y)$; then, considering the pixel with the highest intensity in $J_I(x, y)$ among those 10% highest value pixels in $S_I(x, y)$ as in [11]. This process can be mathematically expressed as follows:

$$L = \max_{(x,y) \in A} \{J_I(x, y)\} \quad (33)$$

where A denotes a set of 10% highest value pixels in $S_I(x, y)$ image.

3.1.3 Primary ground reflectance recovery

After the scattering light term and the atmospheric light are estimated, they are substituted into Eq.(28) to produce $aJ'_I(x, y)$ which is called the primary ground reflectance in this work and denoted by:

$$J_I^*(x, y) := aJ'_I(x, y) = \frac{J_I(x, y) - S_I(x, y)}{L - S_I(x, y)} \quad (34)$$

where $J_I^*(x, y)$ is a primary ground reflectance in the intensity channel. Figures 11 shows the image as a result obtained after each step.

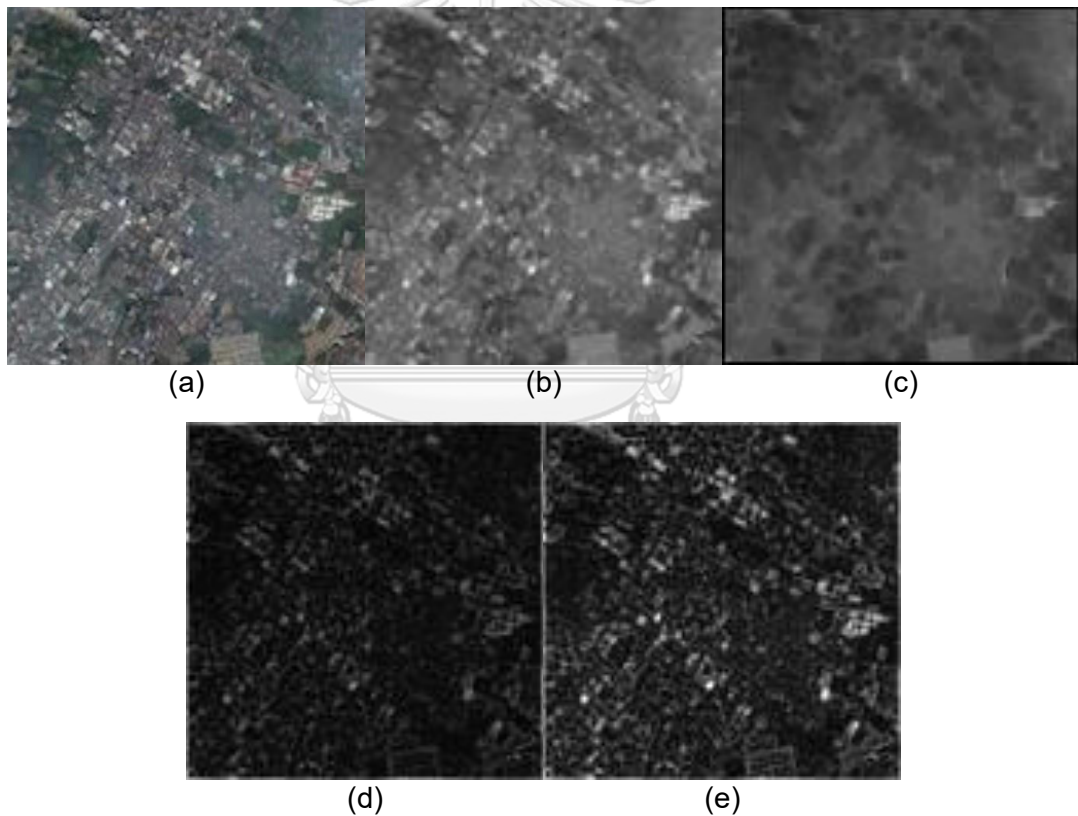


Figure 11 (a) An RGB clouded image, (b) the corresponding intensity, (c) the corresponding estimated scattering light, (d) the intensity after subtraction, and (e) the primary ground reflectance intensity obtained from the method.

3.1.4 Real ground reflectance recovery

In this step, we would like to obtain only the real ground reflectance, $J'_I(x, y)$. Although thin clouds in the primary ground reflectance obtained from Eq.(34) are already removed, there is still the sunlight attenuation coefficient in the recovered primary ground reflectance. This coefficient causes a satellite image to be dimmer than usual. Moreover, there might be an excessive deduction occurred during the previous step; therefore, we propose to slightly increase the intensity of the primary ground reflectance to reduce the effect of sunlight attenuation coefficient, and also regain the intensities that were excessively deducted as much as possible. A gamma correction is used in this step to slightly increase the intensity of $aJ'_I(x, y)$. We select the fractional gamma in order to map the original intensity to higher intensity, and also increase the contrast between similar pixels to gain the visibility of some details that are dark and blended to nearby pixels. However, increasing the intensity without a limitation may cause it to be over the range. Hence, we specify the interval of gamma correction by finding the maximum and minimum intensities of changed pixels in $J_I^*(x, y)$ as follows:

$$b = \max_{(x,y) \in \Delta} \{J_I^*(x, y)\} \quad (35)$$

and

$$a = \min_{(x,y) \in \Delta} \{J_I^*(x, y)\} \quad (36)$$

where $\Delta = \{(x, y) \mid J_I(x, y) - J_I^*(x, y) > 0\}$.

Thereafter, intensities in $J_I^*(x, y)$ will be increased by using a gamma correction within the range, $[a, b]$, as follows:

$$J'_I(x, y) = \begin{cases} (b-a) \left(\frac{J_I^*(x, y) - a}{b-a} \right)^\gamma + a & ; (x, y) \in \Delta \\ J_I^*(x, y)^\gamma & ; \textit{otherwise} \end{cases} \quad (37)$$

where γ is a constant parameter such that $0 < \gamma < 1$. The lower the constant γ is, the brighter the result will be.

From Eq.(37), the primary ground reflectance intensities of changed pixels will be increased up to the maximum value b because the original intensities should not be increased over the highest intensity values among changed pixels to avoid turning back to clouded pixels, and the contrast will also be increased as well. Meanwhile, other pixels that are not deducted in the previous step will be increased by gamma correction as usual.

Figure 12 shows an example of the result obtained from each step. Figure 12(a) is a sample RGB clouded image. After performing the thin cloud removal step, Figure 12(b) shows the result obtained from the primary ground reflectance recovery step in RGB color space which is dimmer than expectation and Figure 12(c) shows the final result in RGB color space produced by the real ground reflectance recovery step which is brighter and clearer in term of ground details.

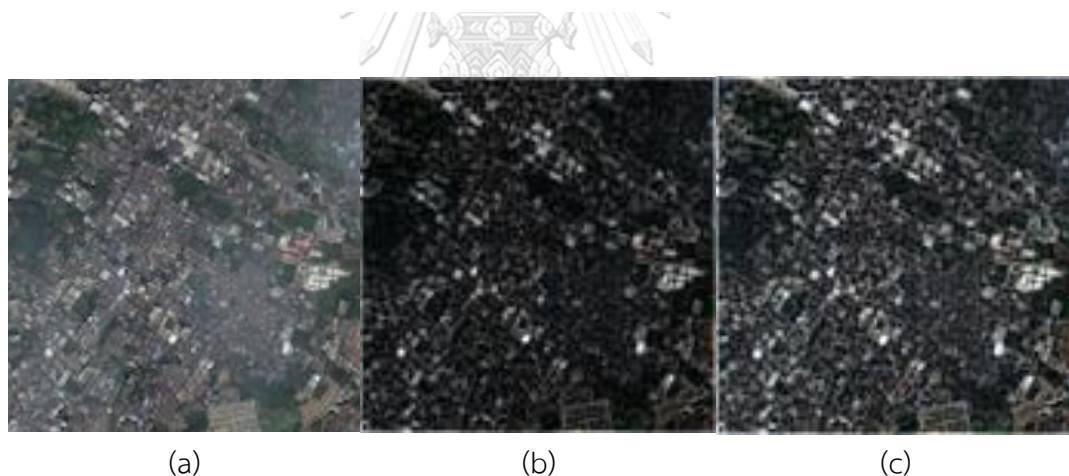


Figure 12 (a) An input clouded image, (b) the primary recovered ground reflectance image, and (c) the real ground reflectance image.

3.1.5 Real ground reflectance contrast enhancement

Due to the low contrast of the result obtained from the previous steps, we also propose to enhance the contrast of the result by using CLAHE in order to gain more details, such as edges and shadows, and also emphasize low intensity regions, such as water surfaces. The result image from previous step is equally partitioned

into 64 tiles (8×8 tiles) and n_{clip} is set to 0.01. The result after applying CLAHE to the result image in Figure 12(c) is shown in Figure 13.



Figure 13 The contrast-enhanced real ground reflectance

3.2 Ground reflectance enhancement step

According to the observation, the saturation channel is also affected by thin clouds. The appearance of thin clouds as a white bright sheet in a satellite image causes saturation value to be decreased due to the fact that saturation is lower when the color appearing in RGB color space is whiter. Therefore, during this step, the saturation channel will be increased in order to enhance the colors of ground information which are blended with thin clouds. However, some pixels with extremely low saturation values probably correspond with either thick clouds or white objects such as white buildings or roads; therefore, pixels with low saturation values are not supposed to be increased significantly so that the original white color in those pixels can be preserved. We thus use a logarithm image transformation function expressed as:

$$J'_s(x, y) = c \log(1 + J_s(x, y)) \quad (38)$$

to map the original saturation to the higher saturation with different rate; i.e., the saturation in the low range is increased with the rate less than the saturation in the high range where $J'_s(x, y)$ is the enhanced saturation image of an image J , $J_s(x, y)$ is the saturation of an image J , and c is a constant parameter where

$c > \frac{1}{\log(2)}$, and some thick clouds and white objects can be preserved by adjusting c to be close to $\frac{1}{\log(2)}$.

Finally, the recovered ground intensity $J'_1(x, y)$ and the enhanced saturation $J'_s(x, y)$ are combined together with its original hue, and the combined image in HSI color space is converted back to RGB color space as shown in the next chapter.

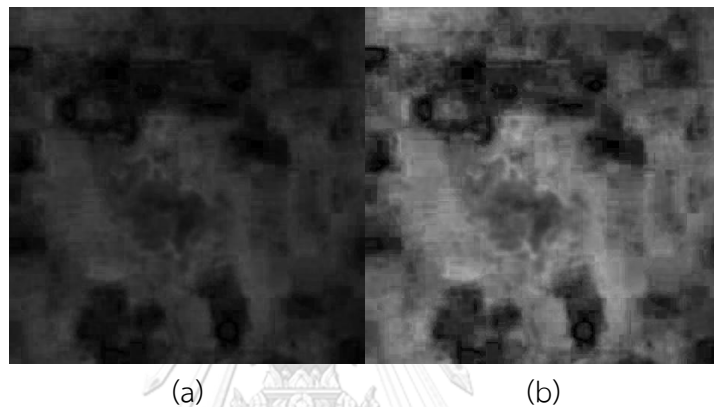


Figure 14 (a) The original saturation of clouded input image (b) The enhanced saturation after the ground reflectance enhancement step


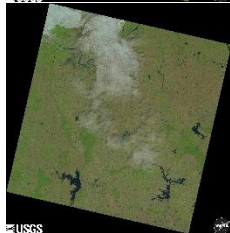
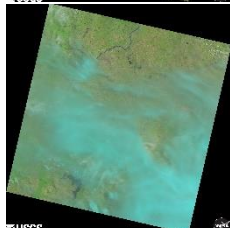
CHAPTER IV



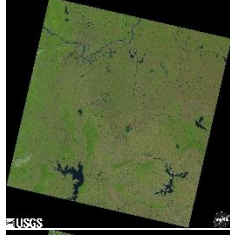




Results and Discussion

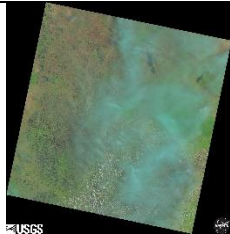

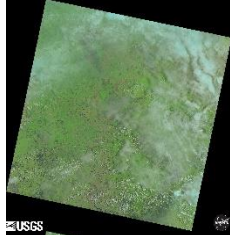
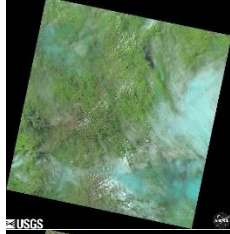
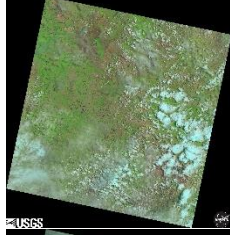
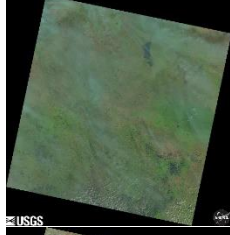

The proposed method was evaluated on a set of 26 Landsat 8 satellite images provided by the USGS EarthExplorer (EE) tool, available from U.S. Geological Survey (USGS). These images were taken over different areas of Thailand from 2013 to 2018. Each image has different amounts of thin clouds and thick clouds for testing the capability of the method in various scenarios.

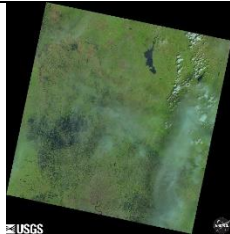



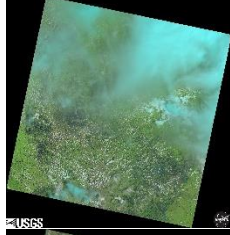

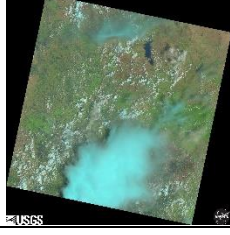
Table 1 shows a set of Landsat 8 satellite images used in the experiments. The first column shows an image number. The second column shows the RGB satellite image obtained from the USGS EarthExplorer (EE) tool. The third and fourth columns show the position of the area where a satellite image was taken in a format of path and row number used in Landsat 8 satellite system. The last column shows the acquisition date of each image in a format year-month-day.



Table 1 A set of Landsat 8 satellite images used in the experiment

| Image no. | Input | Path | Row | Acquisition date |
|-----------|---|------|-----|------------------|
| 1 |  | 128 | 048 | 2014-11-10 |
| 2 |  | 128 | 048 | 2014-12-12 |
| 3 |  | 128 | 048 | 2015-06-22 |

| Image no. | Input | Path | Row | Acquisition date |
|-----------|---|------|-----|------------------|
| 4 |  | 128 | 048 | 2015-11-13 |
| 5 |  | 128 | 048 | 2016-10-30 |
| 6 |  | 128 | 048 | 2016-12-01 |
| 7 |  | 128 | 048 | 2017-01-02 |
| 8 |  | 128 | 048 | 2018-04-11 |
| 9 |  | 128 | 048 | 2018-10-04 |
| 10 |  | 129 | 050 | 2013-10-29 |

| Image no. | Input | Path | Row | Acquisition date |
|-----------|---|------|-----|------------------|
| 11 |  | 129 | 050 | 2014-04-23 |
| 12 |  | 129 | 050 | 2014-05-25 |
| 13 |  | 129 | 050 | 2014-08-13 |
| 14 |  | 129 | 050 | 2014-09-14 |
| 15 |  | 129 | 050 | 2015-07-15 |
| 16 |  | 129 | 050 | 2016-01-07 |
| 17 |  | 129 | 050 | 2016-05-14 |

| Image no. | Input | Path | Row | Acquisition date |
|-----------|---|------|-----|------------------|
| 18 |  | 129 | 050 | 2016-11-22 |
| 19 |  | 129 | 050 | 2017-02-10 |
| 20 |  | 129 | 050 | 2017-02-26 |
| 21 |  | 129 | 050 | 2017-04-15 |
| 22 |  | 129 | 050 | 2017-09-22 |
| 23 |  | 129 | 050 | 2018-01-28 |
| 24 |  | 129 | 050 | 2018-04-02 |

| Image no. | Input | Path | Row | Acquisition date |
|-----------|---|------|-----|------------------|
| 25 |  | 129 | 050 | 2018-05-04 |
| 26 |  | 129 | 050 | 2018-05-20 |

The proposed method was compared with other single-image methods; i.e., Liu's method [6] and He's method [11]. In order to compare the effectiveness of these methods, we firstly compared the result images obtained from the proposed methods with cloud-free satellite images at the same positions. However, since it is difficult to obtain a cloud-free satellite image in exactly the same area and also exactly the same time, we thus use a cloud-free satellite image taken at the closest date to each of the input images to calculate the mean squared error (MSE) that can be expressed as:

$$MSE = \frac{1}{3MN} \left(\sum_{k=R,G,B} \sum_{i=1}^M \sum_{j=1}^N (H'_k(i,j) - H_k^{ref}(i,j))^2 \right) \quad (39)$$

where $H'_k(i,j)$ is a result image size of $M \times N$ in the k channel in RGB color space, $H_k^{ref}(i,j)$ is a reference image in the k channel in RGB color space with the same size as $H'_k(i,j)$, and MSE is a mean squared error calculated from a result image and a reference image.

Moreover, we also evaluated the methods' capability to retrieve distorted and lost details as a result from thin clouds by using Contrast Gain (CG) that was used in [12, 13] and defined as follows:

$$CG = \frac{1}{MN} \left(\sum_{i=1}^M \sum_{j=1}^N C^{H'}(i, j) - \sum_{i=1}^M \sum_{j=1}^N C^H(i, j) \right) \quad (40)$$

where $C^{H'}$ and C^H are contrast images of a result image, H' , and an input image, H , respectively, and can be calculated by

$$C^H(x, y) = \frac{S^H(x, y)}{m^H(x, y)} \quad (41)$$

where $m^H(x, y)$ and $S^H(x, y)$ are the mean and the variance of intensity values in a group of pixels within a patch of size $(2p+1) \times (2p+1)$ centered at pixel (x, y) , and can be expressed as:

$$m^H(x, y) = \frac{1}{(2p+1)^2} \sum_{i=-p}^p \sum_{j=-p}^p H(x+i, y+j) \quad (42)$$

and

$$s^H(x, y) = \frac{1}{(2p+1)^2} \sum_{i=-p}^p \sum_{j=-p}^p |H(x+i, y+j) - m(x, y)| \quad (43)$$




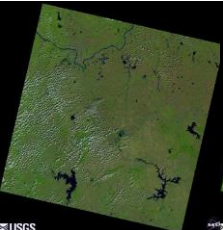



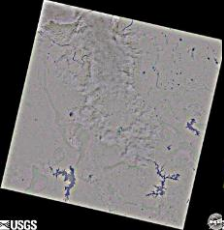
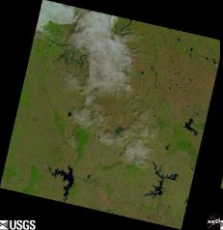
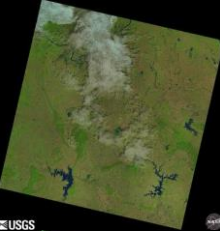
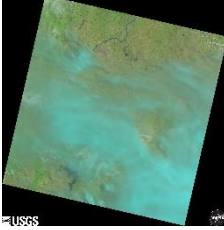

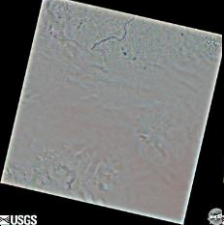
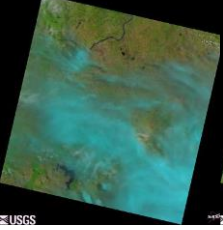
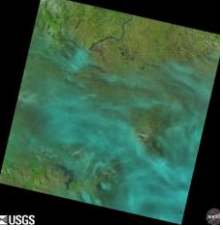
where p can be any integer number, and we set $p=2$ for every image in this work for simplicity. Similarly, $C^{H'}$ can also be calculated by using Eq.(41)-(43) with an image H' instead of an image H .

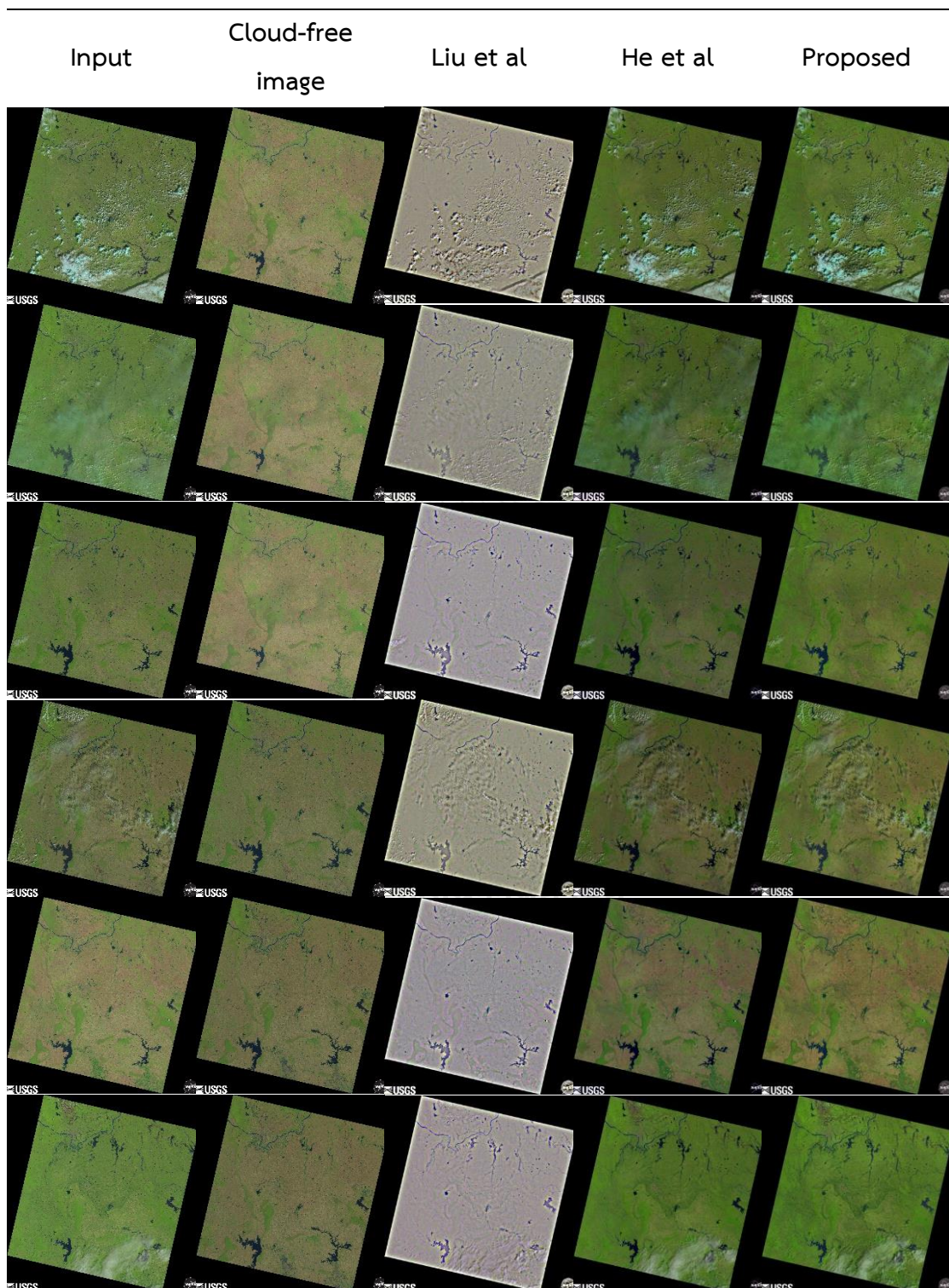
A contrast image is an image where each pixel indicates the contrast or the difference between a pixel itself and other neighbor pixels. The higher the value in each pixel is, the greater the difference will be; i.e., if there is a texture appearing in an area where pixels belong, the contrast value in those pixels will be higher than

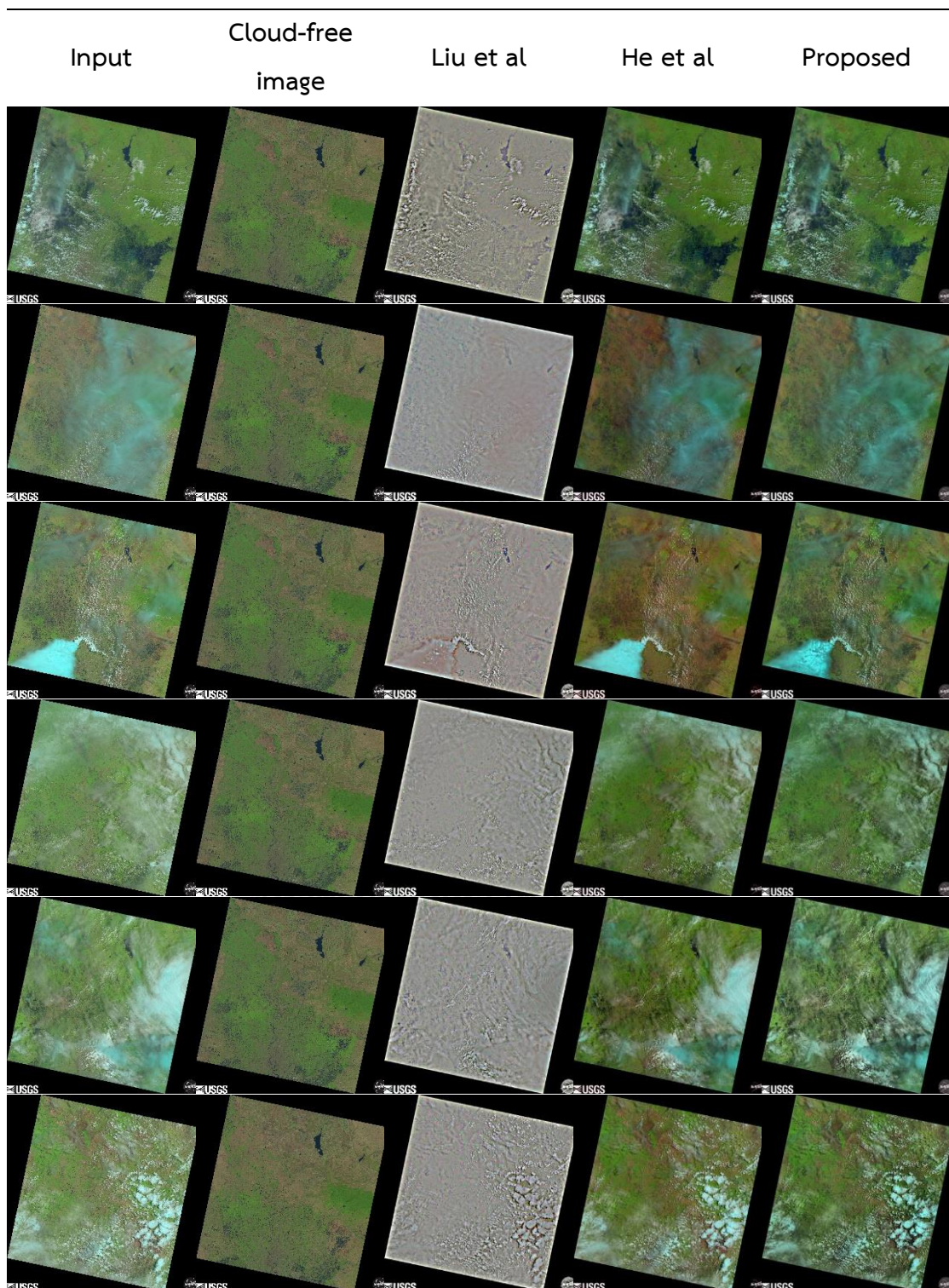
pixels that belong to other plain or smooth areas. Hence, a good result obtained from a cloud removal method should have a higher CG value because of more details that used to be blurred and blended with surrounding areas as a result of being covered by thin clouds.

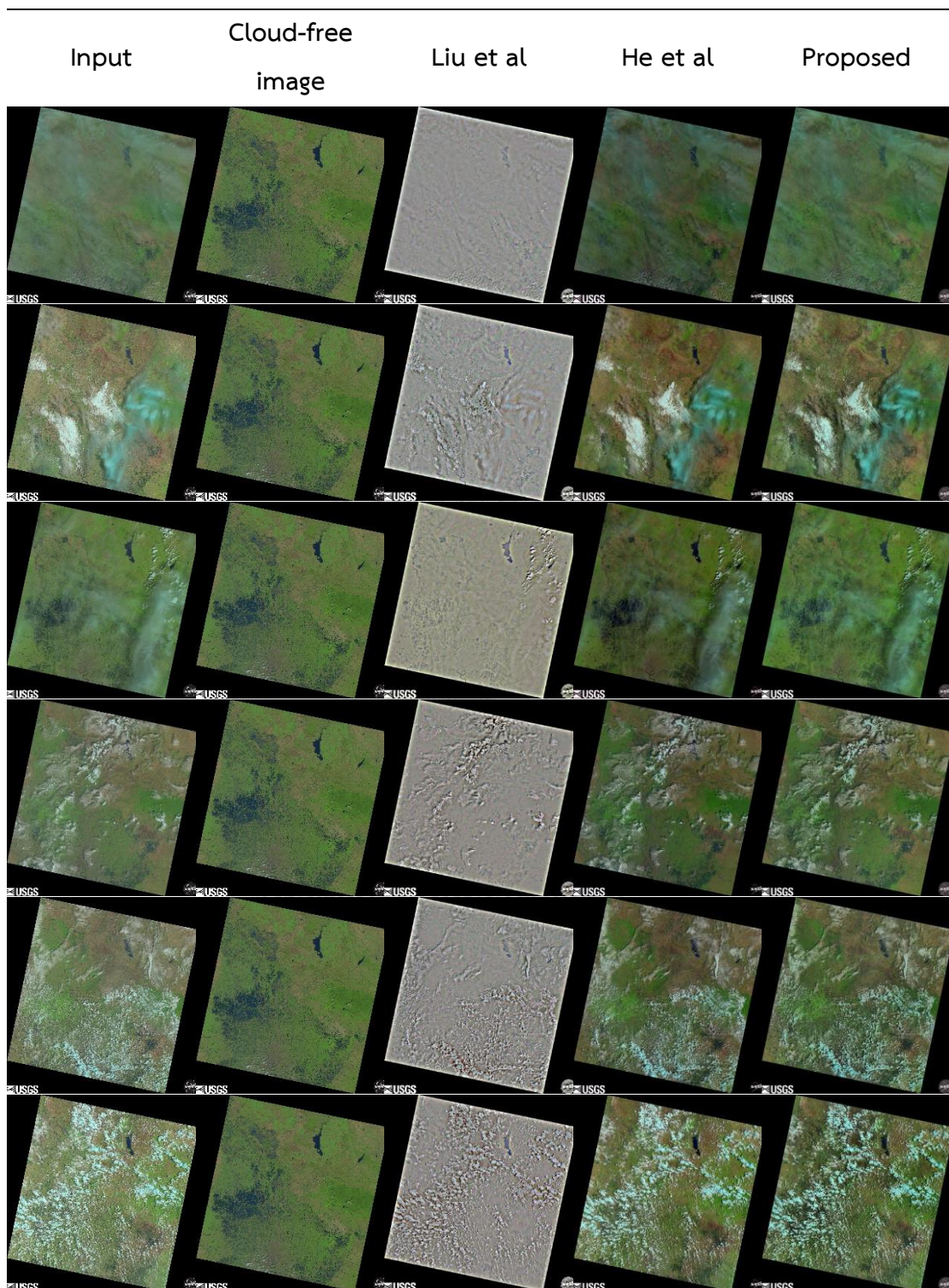
The results are shown in Table 2 where the first column shows the clouded input images, the second column shows the cloud-free images from the closest date to the input images, and the result images obtained from Liu's method, He's method and the proposed method are shown in the third, the fourth and the fifth columns, respectively.

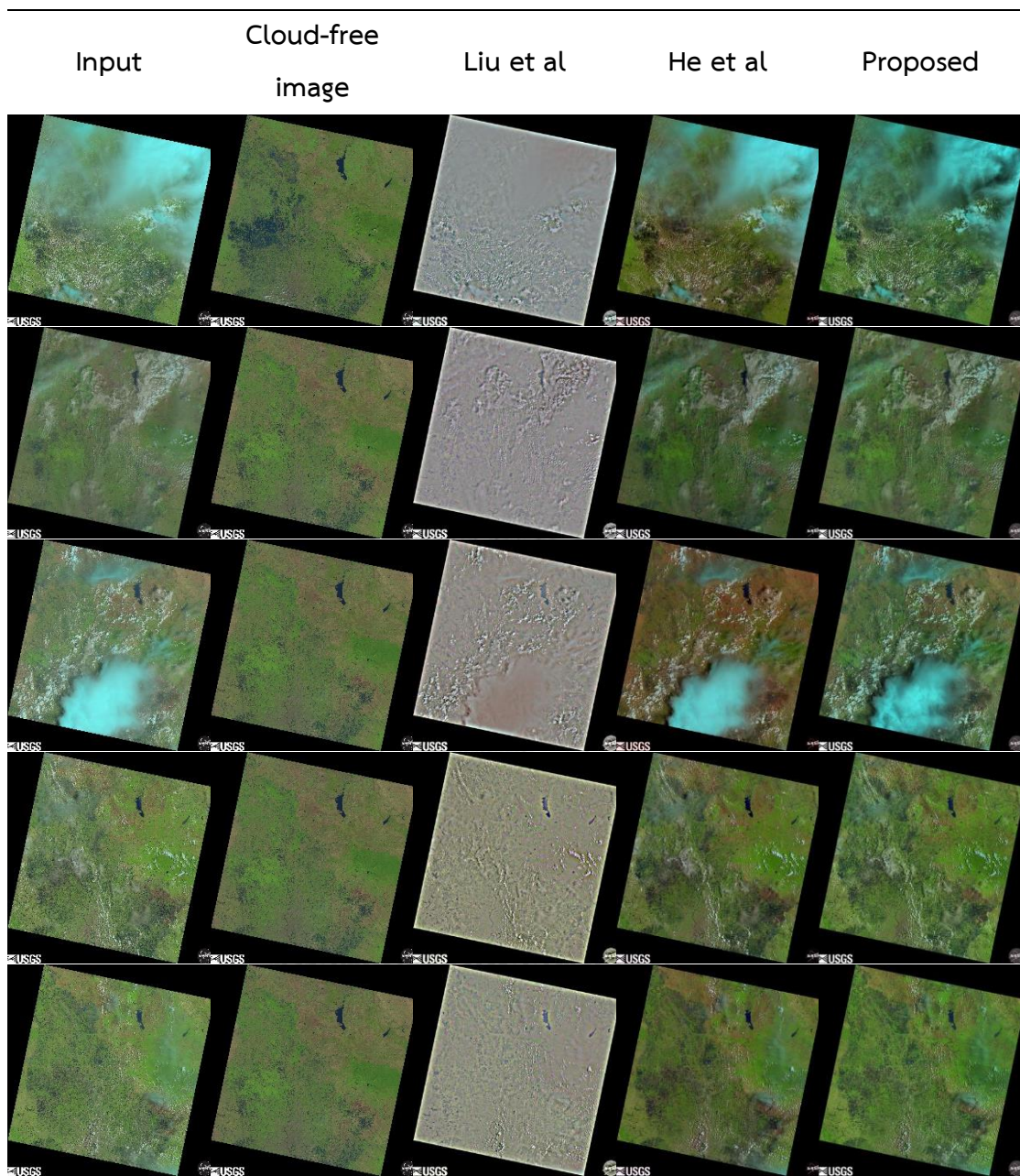
Table 2 The result images obtained from the proposed method and other single-image methods

| Input | Cloud-free image | Liu et al | He et al | Proposed |
|---|---|---|--|---|
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |









From Table 2, MSE and CG of each result image is calculated and shown in Table 3 and Table 4, respectively.

Table 3 Comparison of MSE calculated from the results obtained from Liu's method, He's method, and the proposed method (The lowest MSE among the results in each input image is displayed in bold)

| Image no. | MSE | | |
|-----------|-----------|----------|----------------|
| | Liu et al | He et al | Proposed |
| 1 | 0.01981 | 0.01851 | 0.01545 |
| 2 | 0.02842 | 0.01945 | 0.01314 |
| 3 | 0.02219 | 0.02183 | 0.01952 |
| 4 | 0.02195 | 0.02149 | 0.02000 |
| 5 | 0.02635 | 0.02833 | 0.02111 |
| 6 | 0.02910 | 0.02920 | 0.02003 |
| 7 | 0.03798 | 0.00677 | 0.00371 |
| 8 | 0.04788 | 0.00188 | 0.00178 |
| 9 | 0.04710 | 0.00703 | 0.00694 |
| 10 | 0.03706 | 0.01229 | 0.01087 |
| 11 | 0.04562 | 0.00769 | 0.00549 |
| 12 | 0.04307 | 0.01225 | 0.01011 |
| 13 | 0.04466 | 0.00895 | 0.00748 |
| 14 | 0.04485 | 0.01671 | 0.01385 |
| 15 | 0.04429 | 0.01591 | 0.01538 |
| 16 | 0.05021 | 0.00772 | 0.00314 |
| 17 | 0.05423 | 0.01222 | 0.01099 |
| 18 | 0.04330 | 0.00843 | 0.00490 |
| 19 | 0.04359 | 0.00846 | 0.00703 |
| 20 | 0.04775 | 0.01207 | 0.01188 |
| 21 | 0.05322 | 0.02338 | 0.02229 |
| 22 | 0.04853 | 0.01899 | 0.01587 |
| 23 | 0.04110 | 0.00943 | 0.00501 |
| 24 | 0.03656 | 0.01648 | 0.01495 |

| Image no. | MSE | | |
|-----------|-----------|----------|----------------|
| | Liu et al | He et al | Proposed |
| 25 | 0.02984 | 0.00699 | 0.00651 |
| 26 | 0.03450 | 0.00667 | 0.00569 |

Table 4 Comparison of CG calculated from the results obtained from Liu's method, He's method, and the proposed method (The highest CG among the results in each input image is displayed in bold)

| Image no. | CG | | |
|-----------|----------------|----------|----------|
| | Liu et al | He et al | Proposed |
| 1 | 0.10256 | 0.02442 | 0.07736 |
| 2 | 0.07621 | 0.01556 | 0.05182 |
| 3 | 0.06786 | 0.00757 | 0.04565 |
| 4 | 0.10733 | 0.01920 | 0.08086 |
| 5 | 0.08369 | 0.01757 | 0.05511 |
| 6 | 0.07267 | 0.01809 | 0.04808 |
| 7 | 0.08931 | 0.02028 | 0.06042 |
| 8 | 0.07854 | 0.01974 | 0.05440 |
| 9 | 0.07910 | 0.01318 | 0.05647 |
| 10 | 0.11001 | 0.02685 | 0.08450 |
| 11 | 0.07602 | 0.01657 | 0.05240 |
| 12 | 0.09173 | 0.01915 | 0.07444 |
| 13 | 0.08584 | 0.01829 | 0.07087 |
| 14 | 0.09163 | 0.02109 | 0.07898 |
| 15 | 0.11633 | 0.02357 | 0.10817 |
| 16 | 0.07452 | 0.01821 | 0.04744 |
| 17 | 0.09343 | 0.02178 | 0.07236 |
| 18 | 0.08184 | 0.02440 | 0.05855 |

| Image no. | CG | | |
|-----------|----------------|----------|----------|
| | Liu et al | He et al | Proposed |
| 19 | 0.10225 | 0.02596 | 0.07578 |
| 20 | 0.12632 | 0.03496 | 0.10536 |
| 21 | 0.14339 | 0.03674 | 0.14084 |
| 22 | 0.08620 | 0.02813 | 0.07677 |
| 23 | 0.09400 | 0.02504 | 0.06646 |
| 24 | 0.10061 | 0.02808 | 0.07815 |
| 25 | 0.09844 | 0.02525 | 0.07973 |
| 26 | 0.09028 | 0.02442 | 0.07383 |

From Table 3, the proposed method can outperform other methods in term of similarity to the cloud-free image of each result image which can be noticed from the lowest MSE value in every result.

In contrary, considering the CG values shown in Table 4, although our proposed method can produce the results with higher CG compared to those obtained from He's method, but their CG values are still less than CG values of the results obtained from Liu's method in all experiments as illustrated in Figure 15. In other words, Liu's method can raise edges of the ground objects much better than other methods. However, it can be seen from Table 2 that the actual information of the result images, especially color, obtained from Liu's method are lost severely, so the outputs from this method are clearly different from the others, and not appropriate for any uses. Thus, when result images and CG values are both considered together, it shows that He's method and our proposed method can produce the results with useable features, such as texture and color, but the proposed method can gain more contrast compared to He's method in every test image.

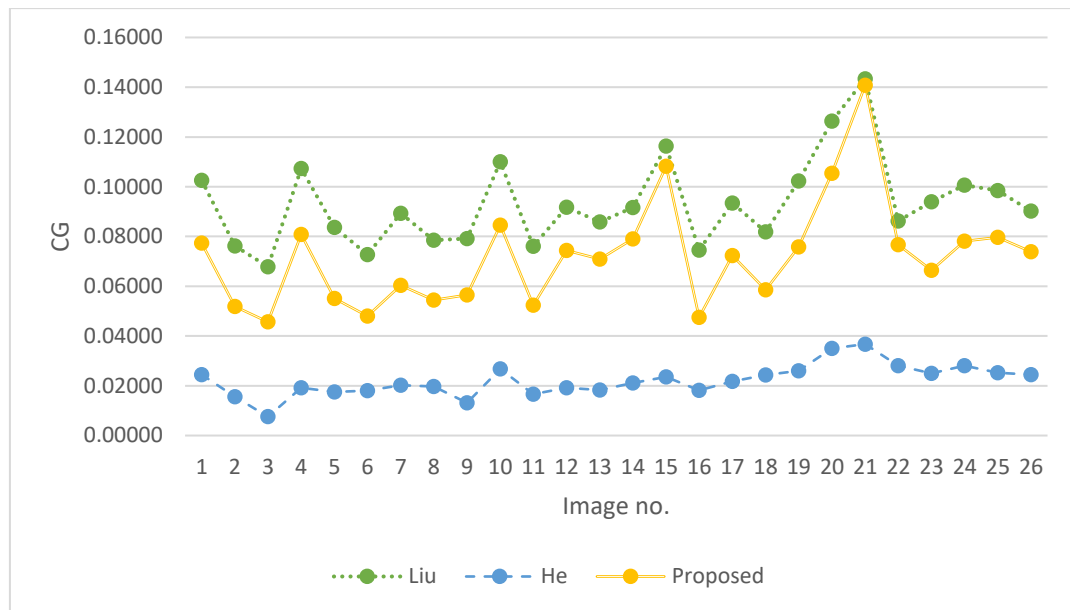


Figure 15 The CG values of the result images obtained from Liu's method, He's method, and the proposed method

Considering output images from our proposed method, there are some parts in each output image that are still blur and clouded due to being covered by too much opaque thin clouds to be deducted by the proposed method with parameters defined in this case. This outcome leads to necessity of identifying potential or limitation of the proposed method in term of the capability to remove thin clouds with different opacity.

For this reason, the input and output images are compared to identify clouded pixels that can be recovered by the proposed method. First, pixels in each image are classified into two groups: clouded pixel and non-clouded pixel. This classification is conducted on a criteria that is based on our observation of cloud-free images in the dataset. According to our observation, the thickness of cloud can be indicated by the difference between intensity and saturation; i.e., $I-S$, because clouds are generally white corresponding to high intensity and low saturation; that is, the higher the difference between these two values is, the whiter the color appearing in an RGB color image will be. Therefore, it is used in this case to examine cloud's thickness by first generating a histogram of $I-S$ from the cloud-free images which is shown in Figure 16. It should be noted that all negative values are

considered as zero in this histogram. According to this histogram, the majority of the differences lie between 0 to 0.3 whilst the counts in other bins are less than 0.1% of the total number of pixels in this histogram. Thus, in this work, a pixel that has the difference between intensity and saturation less than 0.3 will be considered as a non-clouded pixel; on the other hand, a pixel with the difference greater than 0.3 will be considered as a clouded pixel.

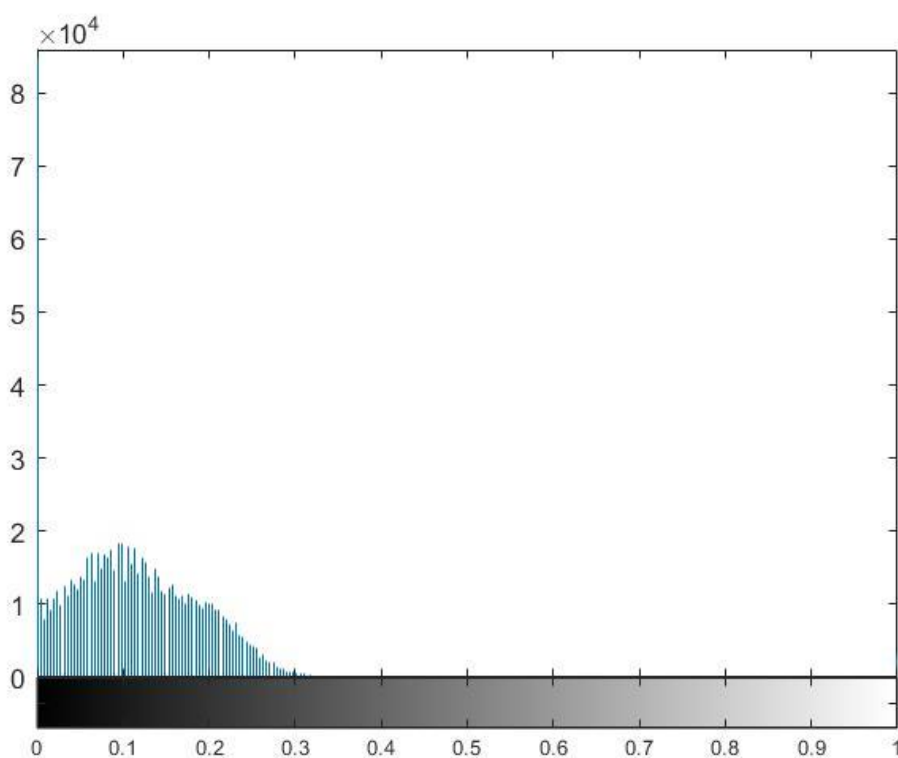


Figure 16 A histogram of I-S from the cloud-free images in the dataset

After clouded pixels are detected, each pair of the input and output images is compared to recognize clouded pixels that have been recovered by the proposed method. If the detected pixels in the input image disappeared in the output image, these pixels will be considered as pixels that can be handled by the proposed method with the parameters fixed in this experiment. The differences of these recovered pixels are then collected and represented by a histogram shown in Figure 17. This histogram shows that there is no pixel with the difference between intensity

and saturation higher than 0.74 and the number of pixels in each bin gradually drops as the differences increase from 0.3 to 0.74. Conversely, a histogram of clouded pixels remaining in output images in Figure 18 shows that a pixel count in each bin increases from 0.3 to 0.6 and keeps decreasing after that with a minor fluctuation until it reaches the maximum count at the difference equal to 1. These remaining pixels may correspond to pixels covered by thin clouds that cannot be recovered by the proposed method as well as thick clouds and bright objects.

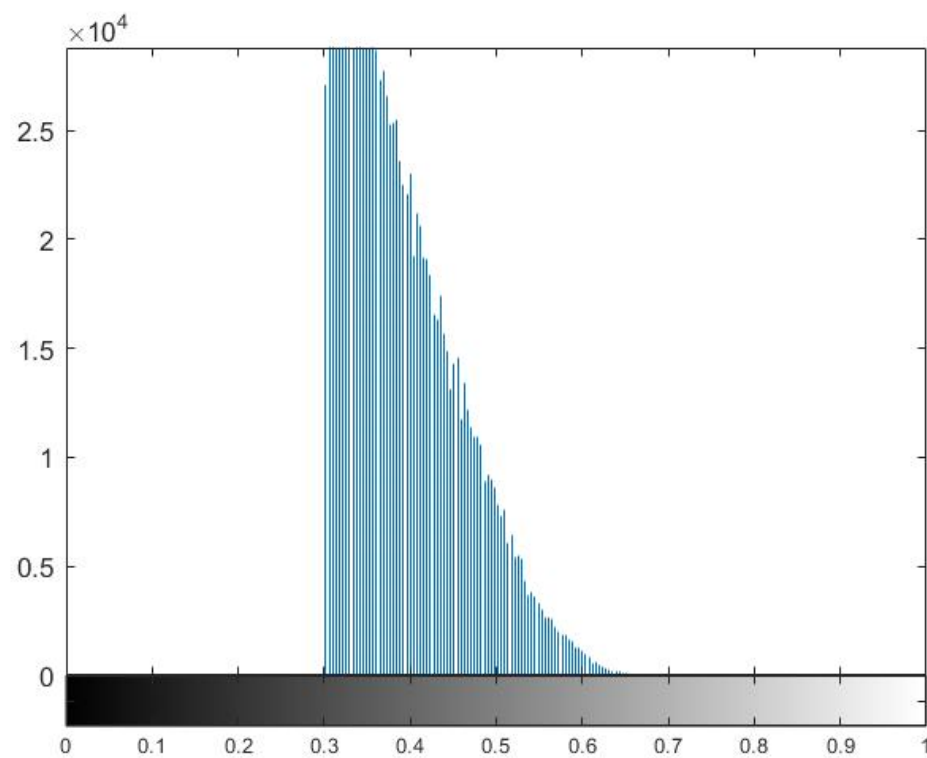


Figure 17 A histogram of I-S of pixels that have been recovered by the proposed method with the parameters fixed in the experiment

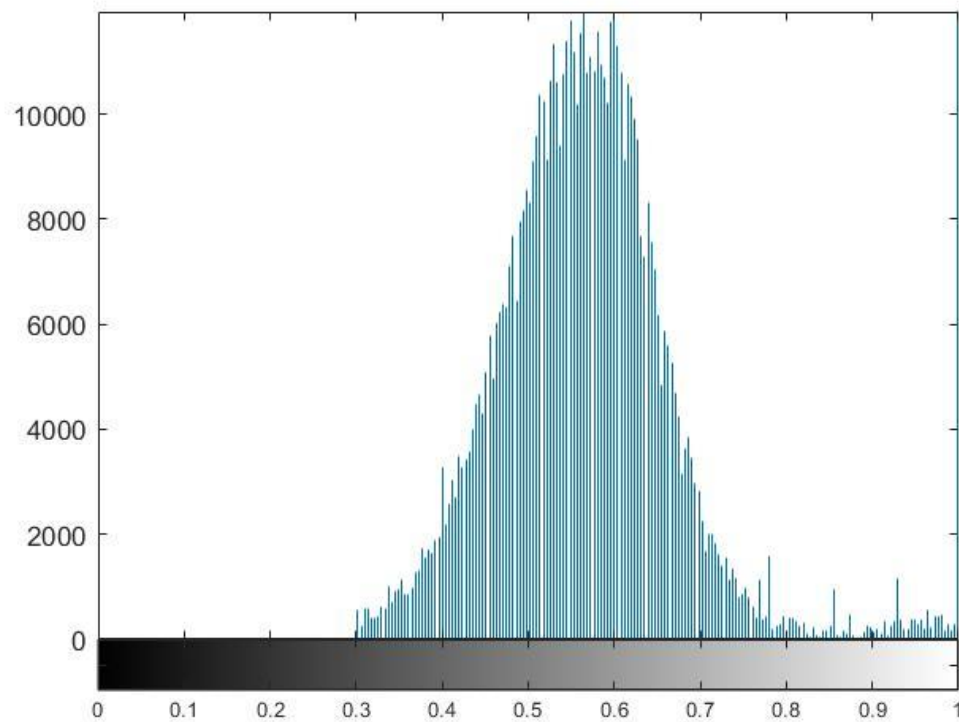


Figure 18 A histogram of I-S of clouded pixels remaining in output images

In other words, the proposed method with the fixed parameters can recover clouded pixels in an input clouded image if the pixel has the difference between intensity and saturation within the range from 0.3 to 0.74 with a decrement in cloud removal effectiveness when the difference between intensity and saturation increases up to 0.74.

Chapter V

Conclusions

This work presents a new method for thin cloud removal performing in HSI color space instead of RGB color space. The proposed method uses both intensity channel and saturation channel to remove thin clouds, and enhance the ground reflectance while keeping hue unchanged in order to preserve the original color of ground objects as much as possible. The proposed method was evaluated on a set of Landsat 8 satellite images taken above areas in Thailand. These images contain thin clouds that must be removed and also thick clouds as well as other bright objects, such as buildings and roads that should be avoided in thin cloud removal step. Our method shows that these thick clouds and bright objects can be neglected by using a minimum value in a small patch to represent each pixel of the estimated scattering light term. This can be seen from the output images in Table 2 where most thick clouds and bright objects still appear without any distortions.

The output images in Table 2 also reveal that the proposed method can remove most of the thin clouds in satellite images if the opacity is not very high, namely when the difference between intensity and saturation is not higher than 0.74, and it can perform more effectively when the difference is lower and is close to 0.3. Furthermore, the proposed method can retrieve some actual information such as color and texture which can be seen from more vibrant color of output images and the contrast gained after applying the method. Table 3 also shows that when the similarity between the output and its corresponding cloud-free image is considered by using MSE value between two RGB images as expressed in (39), the proposed method can generate outputs in RGB format that have red, green and blue component closest to their corresponding cloud-free images. In other words, the proposed method can outperform other methods in term of generating outputs that are closest to their corresponding cloud-free images which can be seen from the lowest MSE values in every result.

Nonetheless, considering the CG values shown in Table 4, even though our proposed method cannot produce the results with the highest CG value as Liu's method, but the result images obtained from Liu's method are significantly degraded which is caused by using the constant mean value as an offset to compensate the excessive thin clouds deduction as mentioned in Chapter II. On the other hand, our proposed method can yield the better results based on both output images' features; i.e., texture and color, and also the measurements, i.e., MSE and CG.

Furthermore, parameters in our proposed method can be adjusted in order to cope with thicker clouds. The parameter ω , used for keeping some details from subtraction during an estimation of the scattering light term, can be set to a higher value to increasingly deduct the scattering light term. Although the higher ω can remove higher opacity thin clouds, but it can also cause the output to become darker than expectation especially in most areas without thin clouds. Thus, before applying gamma correction in real ground reflectance recovery step, the parameter γ used for retrieving accidentally lost brightness should be set dependently on ω . In other words, as the higher the ω is set, the lower the γ should be selected so that the brightness of a darker image caused by the higher ω can be increased more by gamma correction.

5.1 Future work

In this work, the proposed method aims to remove thin clouds and also retain the actual information especially those in areas without thin clouds. However, the proposed method does not include a thin cloud detection step that can identify non-clouded pixels so that these pixels can be excluded from any transformation or calculation processes. Therefore, such thin cloud detection step can be incorporated as a preprocessing step in order to absolutely preserve cloud-free pixels and produce a more precise output.

Moreover, this work has also shown that thin cloud removal can also be performed in HSI color space as well as RGB color space. The proposed method can successfully reduce thin cloud distortion and enhance details covered by thin clouds by adjusting intensity and saturation in HSI color space. Alternatively, there are also

other color spaces having a potential to develop a thin cloud removal method such as HSV color space, which is similar to HSI color space but different in term of brightness component calculation, and CIELAB color space which also has a brightness component but may require a further investigation due to the lack of saturation component.



REFERENCES

- [1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Upper Saddle River: Prentice-Hall, Inc, 2006.
- [2] W. contributors. (2018, 5 November). *Adaptive histogram equalization*. Available:
https://en.wikipedia.org/w/index.php?title=Adaptive_histogram_equalization&oldid=862125899
- [3] K. Zuiderveld, "Contrast Limited Adaptive Histogram Equalization," P. Heckbert, Ed.: Academic Press Professional, Inc., 1994, pp. 474-485.
- [4] Z. K. Liu and B. R. Hunt, "A new approach to removing cloud cover from satellite imagery," *Computer Vision, Graphics, and Image Processing*, vol. 25, no. 2, pp. 252-256, 1984.
- [5] H. Shen, H. Li, Y. Qian, L. Zhang, and Q. Yuan, "An effective thin cloud removal procedure for visible remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 96, pp. 224–235, 2014.
- [6] J. Liu *et al.*, "Thin cloud removal from single satellite images," *Optics Express*, vol. 22, no. 1, pp. 618-632, 2014.
- [7] M. Xu, X. Jia, and M. Pickering, "Automatic cloud removal for Landsat 8 OLI images using cirrus band," *Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 2511-2514, 2014.
- [8] D. Cerra, J. Bieniarz, R. Muller, and P. Reinartz, "Cloud removal from Sentinel-2 image time series through sparse reconstruction from random samples," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 469-473, 2016.
- [9] S. G. Narasimhan and S. K. Nayar, "Vision and the atmosphere," *International Journal of Computer Vision*, vol. 48, no. 3, pp. 233-254, 2002.
- [10] Q. Zhu, J. Mai, and L. Shao, "A Fast Single Image Haze Removal Algorithm Using Color Attenuation Prior," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3522-3533, 2015.

- [11] K. He, J. Sun, and X. Tang, "Single Image Haze Removal Using Dark Channel Prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2341-2353, 2011.
- [12] A. K. Tripathi and S. Mukhopadhyay, "Single image fog removal using anisotropic diffusion," *IET Image Processing*, vol. 6, no. 7, pp. 966-975, 2012.
- [13] M. I. Anwar and A. Khosla, "Vision enhancement through single image fog removal," *Engineering Science and Technology, an International Journal*, vol. 20, no. 3, pp. 1075-1083, 2017/06/01/ 2017.



VITA

NAME Thanet Markchom

DATE OF BIRTH 18 January 1994

PLACE OF BIRTH Bangkok

INSTITUTIONS ATTENDED Chulalongkorn University

HOME ADDRESS 34/274 Moo 3, Khlong Sam, Khlong Luang, Pathum Thani,
12120



จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY