

Old Dominion University

## ODU Digital Commons

---

Electrical & Computer Engineering Theses & Dissertations

Electrical & Computer Engineering

---

Spring 2012

# A Vision-Based Automatic Safe landing-Site Detection System

Yufei Shen

*Old Dominion University*

Follow this and additional works at: [https://digitalcommons.odu.edu/ece\\_etds](https://digitalcommons.odu.edu/ece_etds)



Part of the [Computer Engineering Commons](#), [Computer Sciences Commons](#), and the [Electrical and Computer Engineering Commons](#)

---

### Recommended Citation

Shen, Yufei. "A Vision-Based Automatic Safe landing-Site Detection System" (2012). Doctor of Philosophy (PhD), Dissertation, Electrical & Computer Engineering, Old Dominion University, DOI: 10.25777/36qe-eg15 [https://digitalcommons.odu.edu/ece\\_etds/133](https://digitalcommons.odu.edu/ece_etds/133)

This Dissertation is brought to you for free and open access by the Electrical & Computer Engineering at ODU Digital Commons. It has been accepted for inclusion in Electrical & Computer Engineering Theses & Dissertations by an authorized administrator of ODU Digital Commons. For more information, please contact [digitalcommons@odu.edu](mailto:digitalcommons@odu.edu).

**A VISION-BASED AUTOMATIC SAFE LANDING-SITE  
DETECTION SYSTEM**

by

Yufei Shen  
B.S. July 2006, Fudan University  
M.S. August 2009, Old Dominion University

A Dissertation Submitted to the Faculty of  
Old Dominion University in Partial Fulfillment of the  
Requirements for the Degree of

**DOCTOR OF PHILOSOPHY**

**ELECTRICAL AND COMPUTER ENGINEERING**

**OLD DOMINION UNIVERSITY**

May 2012

Approved by: 

\_\_\_\_\_  
Dean Krusienski (Director)

\_\_\_\_\_  
Jiang Li (Member)

\_\_\_\_\_  
Dimitrie Popescu (Member)

\_\_\_\_\_  
Roland Lawrence (Member)

# ABSTRACT

## A VISION-BASED AUTOMATIC SAFE LANDING-SITE DETECTION SYSTEM

Yufei Shen

Old Dominion University, 2012

Director: Dr. Dean Krusienski

An automatic safe landing-site detection system is proposed for aircraft emergency landing, based on visible information acquired by aircraft-mounted cameras. Emergency landing is an unplanned event in response to emergency situations. If, as is unfortunately usually the case, there is no airstrip or airfield that can be reached by the un-powered aircraft, a crash landing or ditching has to be carried out. Identifying a safe landing-site is critical to the survival of passengers and crew. Conventionally, the pilot chooses the landing-site visually by looking at the terrain through the cockpit. The success of this vital decision greatly depends on the external environmental factors that can impair human vision, and on the pilot's flight experience that can vary significantly among pilots. Therefore, we propose a robust, reliable and efficient detection system that is expected to alleviate the negative impact of these factors. In this study, we focus on the detection mechanism of the proposed system and assume that the image enhancement for increased visibility and image stitching for a larger field-of-view have already been performed on terrain images acquired by aircraft-mounted cameras. Specifically, we first propose a hierarchical elastic horizon detection algorithm to identify ground in the image. Then the terrain image is divided into non-overlapping blocks which are clustered according to a "roughness"

measure. Adjacent smooth blocks are merged to form potential landing-sites whose dimensions are measured with principal component analysis and geometric transformations. If the dimensions of a candidate region exceed the minimum requirement for safe landing, the potential landing-site is considered a safe candidate and highlighted on the human machine interface. At the end, the pilot makes the final decision by confirming one of the candidates, also considering other factors such as wind speed and wind direction, etc.

This dissertation is dedicated to my family. I sincerely appreciate and treasure your unconditional love and support in my life.

## ACKNOWLEDGMENTS

First, I express my most sincere gratitude to Dr. Zia-ur Rahman who passed away in December 2010. Dr. Rahman's kind advice and help guided me through difficulties during my Ph.D. study at Old Dominion University. I deeply cherish the memory of Dr. Rahman and his outstanding leadership and contribution in this dissertation.

I deeply appreciate Drs. Dean Krusienski and Jiang Li for their generous and timely assistance and advising which gave me courage, confidence and strength to finish my Ph.D. study. I am grateful to Drs. Dimitrie Popescu and Roland Lawrence for serving on my advisory and dissertation committee. I am thankful for Dr. Oscar Gonzalez's kind help that assists me to complete the program.

I also would like to thank my friends for their consistent support and encouragement in the past few years.

## TABLE OF CONTENTS

	Page
LIST OF TABLES .....	ix
LIST OF FIGURES .....	xi
 Chapter	
1. INTRODUCTION .....	1
1.1 PROBLEM STATEMENT .....	1
1.2 DEFINITION OF THE SAFE LANDING-SITE .....	2
1.3 NECESSITY OF THE VISION-BASED SYSTEM .....	2
1.4 RELATED WORKS .....	3
1.5 SYSTEM FRAMEWORK .....	5
1.6 CONTRIBUTIONS OF THIS STUDY .....	7
2. IMAGE ENHANCEMENT .....	9
2.1 CONTRAST STRETCHING .....	10
2.2 HISTOGRAM EQUALIZATION .....	12
2.3 GAMMA CORRECTION .....	13
2.4 INVERSE HYPERBOLIC TANGENT SHAPED CORRECTION ....	15
2.5 RETINEX .....	17
2.6 EXPERIMENTAL RESULTS .....	22
2.7 DISCUSSION .....	23
3. HORIZON DETECTION .....	28
3.1 INTENSITY-BASED HORIZON DETECTION .....	28
3.2 HOUGH TRANSFORM-BASED HORIZON DETECTION .....	29
3.3 A GREEDY SEARCH ALGORITHM .....	30
3.4 PROPOSED METHOD .....	31
3.5 EXPERIMENTAL RESULTS .....	34
3.6 DISCUSSION .....	35
4. LANDING-SITE DETECTION .....	40
4.1 ROUGHNESS ASSESSMENT .....	40
4.2 IMAGE SEGMENTATION .....	42
4.3 DIMENSION ASSESSMENT .....	43
4.4 HUMAN-MACHINE INTERFACE .....	45
4.5 PERFORMANCE METRIC .....	46

4.6	EXPERIMENTAL RESULTS .....	53
4.7	DISCUSSION .....	58
5.	TERRAIN ANALYSIS .....	61
5.1	TERRAIN FEATURES .....	61
5.2	OWO-BP-OR NEURAL NETWORK .....	63
5.3	CASCADED CLASSIFIER .....	66
5.4	EXPERIMENTAL RESULTS .....	70
5.5	DISCUSSION .....	71
6.	CONCLUSIONS .....	73
	REFERENCES .....	75
	APPENDICES .....	83
	APPENDIX A. TRAINING AND TESTING RESULTS OF CLASSIFIERS FOR ALL POSSIBLE COMBINATIONS.....	83
	VITA.....	89

## LIST OF TABLES

Table	Page
1. Four possible results of the detection of landing-sites .....	49
2. Distribution of normalized scores of experimental results .....	56
3 Training and Testing Results of Classifiers for All Possible Combinations .	83

## LIST OF FIGURES

Figure	Page
1. Flow diagram of the proposed automatic safe landing-site detection system	6
2. An example of successful contrast stretching .....	12
3. An example of failed contrast stretching .....	12
4. An example of histogram equalization .....	13
5. A group of gamma transforms .....	14
6. An example of gamma correction .....	15
7. An group of inverse hyperbolic tangent shaped correction functions .....	16
8. An example of inverse hyperbolic tangent shaped correction .....	17
9. An sample image enhanced by SSR in different scales and MSR .....	20
10. Gray world assumption and MSR .....	23
11. MSR application on aerial images .....	25
12. Comparison of enhancement results generated by four methods (1) .....	26
13. Comparison of enhancement results generated by four methods (2) .....	27
14. A failed example by using Hough transform only .....	30
15. A failed example by using the greedy search method .....	30
16. A sample image .....	37
17. Coarse adjustment .....	37
18. Fine adjustment .....	37
19. Detected horizon .....	37
20. Comparison of horizon detection results generated by four methods .....	38
21. Horizon detection results .....	38
22. A sample image .....	47

23. Edges found with Canny detector .....	47
24. Clustering result based on CHS.....	47
25. Multi-region growing result .....	47
26. Measurement of realistic dimensionality.....	47
27. Candidate sites for emergency landing .....	47
28. Simplified imaging model .....	48
29. Comparison between manual selection and automatic detection .....	55
30. Detection results of sequential images .....	59
31. Energy rings of frequency spectrum .....	63
32. Multi-layer perceptron .....	68
33. Structure of the cascaded classifier .....	69
34. A group of sample terrain patches .....	70

## CHAPTER 1

### INTRODUCTION

#### 1.1 PROBLEM STATEMENT

Emergency landing is an unplanned event in response to emergency situations. The top five leading factors of emergency landing or forced landing are: engine failure, running out of fuel, extremely bad weather, medical emergency, and aircraft hijack. Particularly, under the two most emergent situations, engine failure and running out of fuel, the aircraft may quickly lose flying power, and its maneuverability may be restricted to gliding. Once these happen, a forced landing process has to be immediately carried out. If, as is unfortunately often the case, there is no airport or even a runway that can be reached by the un-powered aircraft, a crash landing or ditching will be inevitable.

Finding a safe landing-site is vital to the survival of passengers and the pilot. Conventionally, the emergency landing-site is visually selected by the pilot looking at the terrain that is visible through the cockpit. This is a required fundamental skill in the flight training program, and every pilot is supposed to have the capability to do so. However, many external environmental factors, e.g., fog, rain, illumination, can significantly affect human vision so that the decision of choosing the optimal landing-site highly depends on the pilot's flight experience—the most significant internal factor—which can vary a lot among different pilots. In addition, the visual angle that the human eyes can cover simultaneously is limited: when the pilot looks to the left, what is on the right is missed, and vice versa. Since time is of supreme importance in the scenario under consideration, the inability to scan on both sides of

the cockpit simultaneously is a distinct disadvantage. Imaging sensors can alleviate this problem by creating panorama images that encompass the entire field-of-view (FOV) in front of the aircraft. In order to compensate for the natural inadequacies of human vision and also to alleviate the negative effects of both external and internal factors, a robust, reliable and efficient process for safe landing-site detection is greatly desirable. Therefore, the design of a vision-based automatic safe landing-site detection system [1–3] is proposed in this study.

## **1.2 DEFINITION OF THE SAFE LANDING-SITE**

Before introducing the design of system, the appropriate criteria to assess the safeness of landing-sites are first investigated. Two geographic concepts, elevation and landform, are taken into consideration. The gradient of elevation generally determines the roughness of the terrain. Landform describes terrain covering, e.g., forest, grass, water, rock, buildings. Smooth elevation gradient by itself is not sufficient to guarantee a safe landing-site, since the associated landform could be hazardous to the landing procedure. In addition, the landing-site has to have sufficient length and width—which can vary with the type of airplane—to enable a safe emergency landing. In summary, the “safeness” of a potential landing-site is evaluated by considering its surface roughness and its dimensions. Thus, a landing-site is considered safe only if its surface is smooth and its length and width are adequate. The proposed safe landing-site detection system is designed to automatically detect landing-sites that meet both of the two requirements.

## **1.3 NECESSITY OF THE VISION-BASED SYSTEM**

The safeness of an emergency landing-site is mainly determined by its surface roughness and dimensions. In general, the roughness of terrain can be measured by

the gradient of elevation, so if the elevation map of terrain is available, the gradient information can be easily found and the safeness can be accurately estimated. However, in this specific scenario, the safe landing is not only determined by the elevation variation of the land but also threatened by the hazards upon the ground, e.g., trees, rocks, vehicles, which are usually not captured in elevation maps. Therefore, a vision-based information channel is necessary, which provides a real-time imagery of the ground. Ideally, when the aircraft is flying in the upper air, it can be guided to a generally smooth area according to the gradient information extracted from the elevation map. Then, the proposed computer-aided detection (CAD) system leads the aircraft to a safe landing-site. In practice, most aircrafts do not have either the database of elevation maps or the LIDAR sensor system. Imagery captured by aircraft-mounted cameras is the only available information source so that the proposed CAD system will play a crucial role under this scenario.

#### 1.4 RELATED WORKS

Many achievements of autonomous landing have been accomplished [4–11] by utilizing vision-based approaches to guide unmanned aerial vehicles (UAVs) or helicopters to known landing-sites. Landing marks, which often appear high-contrast in the image as to be easily detected, play an important role in those approaches by providing relative position information for state estimation and control. Nevertheless, for a landing strategy to be feasible in unknown environments, which is usually the case for emergency landing, the dependence on known landing marks is limiting and, therefore, a flexible means of finding safe landing-sites is desired.

An extensive literature review was conducted but few publications are found on the topic of automatic aircraft safe landing-site detection. Garcia-Pardo et al. [12] stated that at their time of writing they were “unaware of a vision based safe landing system for an autonomous aerial robot.” Fitzgerald et al. [13] stated that “based

on an exhaustive literature review during the past two years, the group has found no forced landing research or forced landing system currently available.” Also, as stated in [14], “A thorough literature review revealed that there has been no previous research on automated forced landing systems.”

To date, there are relatively few publications on automatic aircraft safe landing-site detection. As stated in [12–14], no automated forced landing research or automated forced landing system were available at their time of writing. In [12], Garcia-Pardo et al. designed a two-step autonomous safe landing-site detection strategy. First, they applied a local contrast descriptor  $\mu/\sigma$ , which is derived by normalizing the neighborhood of the to-be-tested pixel and then calculating the mean  $\mu$  and the standard deviation  $\sigma$  of its neighborhood, to assess the roughness of the ground under the assumption that the boundaries of hazards appear as high-contrast edges in the image, reflected by small values of  $\mu/\sigma$ . A contrast threshold needs to be selected to differentiate smooth areas and boundaries. The optimal contrast threshold is found to have a linear relationship, with the ratio of mean and standard deviation of the whole image [12]. Then, round landing-sites with a sufficient size are found in the smooth areas. The system described in [12] was tested in an off-line fashion on 10 image sequences, which are captured by real flights over a synthesized environment, i.e. placing white boxes (obstacles) on grass covered ground. The detection results were evaluated by a “failure rate” defined as the percentage of images in which the system failed to find any safe landing-site.

Fitzgerald et al. also applied a two-step safe landing-site detection strategy [13]. First, Canny edge detector [15] is employed to describe edges in the image. This is computationally more efficient than the local contrast descriptor mentioned above. Second, the safe landing-sites are found by scanning the smooth area with a set of rectangular masks which are pre-defined in different scales and rotation angles. Three problems associated with the second step are: (1) It is inconvenient, or even

impossible, to pre-define a sufficiently large number of masks with all possible scales and angles. For example, if a potential safe landing-site has a shape which is not covered in the pre-defined mask set, the system is very likely to miss it. (2) Various aircrafts have different requirements of safe landing-sites in terms of the minimum length and width. Using a pre-defined set of masks is limiting to apply the system to different aircrafts and (3) It is computationally expensive to move all the masks over the smooth area, and the computational cost is proportional to the number of masks so that the requirement of time cost conflicts with the requirement of detection accuracy.

In addition to aircraft applications, related research of spacecraft landing has been conducted by many groups in recent years. NASA Jet Propulsion Laboratory (JPL) proposed a LIDAR-based hazard avoidance approach for safe landing on Mars [16]. They made use of elevation maps generated by scanning synthetic terrains with a simulated LIDAR model. Later, JPL introduced a fuzzy rule-based safety index to assess landing-sites [17, 18]. Furthermore, they brought multi-sensor images into their approach [19]. JPL also proposed a method to estimate the reachable area for the spacecraft [20] based on ballistic analysis. In addition to its application to landing on Mars, autonomous landing and hazard avoidance technologies (ALHAT) are also utilized for lunar landing [21–23] and UAV landing [13]. Therefore, the proposed system discussed here has a wide range of potential applications.

## 1.5 SYSTEM FRAMEWORK

The proposed safe landing-site detection system consists of seven main modules as shown in Figure 1. In the first module, images are acquired by aircraft-mounted cameras. Each camera looks in a specific direction covering a portion of the region in front of the airplane. Multi-spectrum sensors are preferred to obtain complementary information. In the second module, the separate images that are acquired at the same

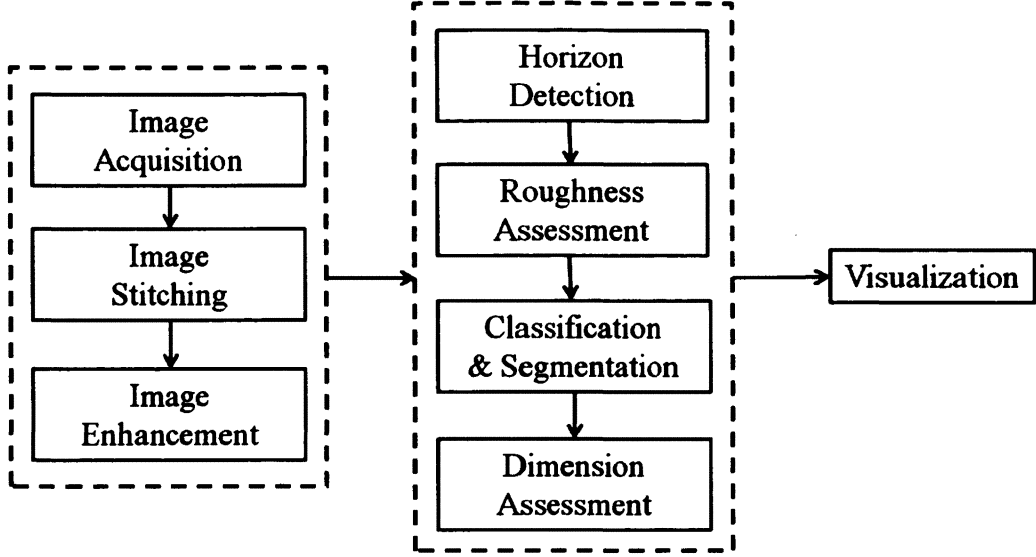


FIG. 1: Flow diagram of the proposed automatic safe landing-site detection system

time instant, are registered and stitched together to form a larger panorama image that covers the full FOV in front of the airplane, instead of regions under or behind it. This is reasonable because the aircraft is assumed to have lost its flying power in the scenario of this study. Landing to an area in front of the gliding aircraft is relatively easier than landing to areas under or behind it in terms of maneuverability. The imaging platform comprised by the first two modules is very important to generate high quality images. However, they will not be further discussed since the emphasis of this study is not on the development of the imaging platform, but on the computer-aided detection (CAD) system for finding safe emergency landing-sites.

To ameliorate the effect of environmental factors and improve the contrast and sharpness of images, which directly affects the performance of the subsequent modules, the non-linear retinex image enhancement method [24, 25] is utilized in the third module to images captured under poor illumination or weather conditions. Specifically, an elastic bound detection method is designed to position the horizon. The

terrain image is divided into non-overlapping blocks which are then clustered according to a “roughness” measure (the fourth module). Adjacent smooth blocks are merged to form potential landing sites (the fifth module) whose dimensions are measured with principal component analysis and geometric transformations (the sixth module). If the dimensions of the candidate region exceed the minimum requirement for safe landing, the potential landing site is considered a safe candidate and highlighted on the human machine interface (the seventh module). At the end, the pilot makes the final decision by confirming one of the candidates, also considering other factors such as wind speed and wind direction, etc.

## 1.6 CONTRIBUTIONS OF THIS STUDY

The contributions of the present dissertation consist of the following parts: (1) A delicate automatic safe landing-site detection mechanism is developed by seamlessly combining some existing image processing and analysis techniques, including block-wise roughness assessment, classification of blocks based on their edge strength, segmentation of candidate safe landing-sites, dimension assessment of candidate landing-sites, and visualization of detected safe landing-sites on human-machine interface. (2) A hierarchical elastic horizon detection algorithm is proposed to identify the ground/sky interface in the aerial image so that the camera is relieved from the limitation of looking straight down to the ground. In the forward-looking mode, the system can detect safe landing-sites in front of the aircraft providing more time to the pilot to prepare landing, which is especially helpful for un-powered aircraft in emergency situations. (3) The efficiency of the detection system is improved by applying Canny edge detector, instead of the local contrast descriptor [12], as a part of the roughness assessment algorithm and utilizing principle component analysis as the means to measure the dimension of smooth areas, instead of using a pre-defined set of masks [13]. And (4) a performance metric is developed to quantitatively evaluate

the detection results.

The remainder of the dissertation is organized as follows. In Chapter 2, the image enhancement module is described. In Chapter 3, a hierarchical horizon detection algorithm is presented, followed by Chapter 4, in which the proposed safe landing-site detection algorithm is described in detail. In Chapter 5, an attempt of terrain analysis is presented. Conclusions are drawn in Chapter 6.

## CHAPTER 2

### IMAGE ENHANCEMENT

Image enhancement is the process of improving the quality of images and often serves as a preliminary module at the front end of a CAD system so that interested objects become distinguishable from the surroundings. Contrast enhancement is one of the most fundamental tasks in image enhancement, which is very application dependent. Many image enhancement techniques [26–31] have been developed for a variety of applications. In this chapter, two widely used histogram modification based image enhancement methods, contrast stretching [28] and histogram equalization [29], are reviewed. Then, two non-linear image enhancement methods, gamma correction [31] and retinex algorithm [24, 25], are described, which are designed for improving the quality of low-contrast images captured under poor illumination or bad weather conditions. Results generated by the four methods are compared in Section 2.6.

A few preliminaries are firstly reviewed.

#### 2D Image

Let  $I(x_1, x_2)$  be the integer value at pixel  $(x_1, x_2)$  of image  $I$ , which has  $N_{row}$  rows and  $N_{col}$  columns ( $1 \leq x_1 \leq N_{row}$ ,  $1 \leq x_2 \leq N_{col}$ ), in the range of  $[0, L]$ .  $L$  is 255 if the bit depth is eight.

#### Histogram

Histogram is a statistical distribution of pixel values in the image. The histogram  $H(l)$  ( $l \in [0, L]$ ) of image  $I$  can be computed by the following pseudo code.

```

Initialize  $H$  with zeros;
for  $x_1 = 1 : N_{row}$ 
    for  $x_2 = 1 : N_{col}$ 
         $H(I(x)) = H(I(x)) + 1$ ;
    end
end
end

```

### Intensity Level Transform

The motivation of histogram modification based methods is to find a transform  $T(\cdot)$  that maps the original pixel values to transformed pixel values  $\tilde{I}(x_1, x_2) = T(I(x_1, x_2))$ .

### Cumulative Intensity Level Distribution

Cumulative intensity level distribution  $P$  of histogram  $H$  is defined as

$$P(l) = \frac{1}{N_{row} \times N_{col}} \sum_{k=0}^l H(k). \quad (1)$$

It is easy to prove that  $P(l)$  is in the range of  $[0, 1]$  and  $P(L) = 1$ .

## 2.1 CONTRAST STRETCHING

Peaks of histogram often concentrate at the lower end for dark images, and vice versa for bright images. When the peaks are close to one another, image contrast becomes low. Contrast stretching [28] transforms the over-concentrated histogram of the original image to the full dynamic range such that the contrast increases as the distances between peaks are enlarged. The transform of contrast stretching can

be written as

$$T(l) = \begin{cases} 0 & l < lo, \\ \left\lfloor \frac{l - lo}{hi - lo} \times L \right\rfloor & lo \leq l \leq hi, \\ L & l > hi, \end{cases} \quad (2)$$

where  $T(l)$  is the transformed value of  $l$ ,  $\lfloor \cdot \rfloor$  is the floor function, and  $lo$  and  $hi$  are two thresholds at lower and higher ends. The lower tail of histogram ( $l < lo$ ) and the higher tail of histogram ( $l > hi$ ) are clipped, and then the clipped histogram in the range of  $[lo, hi]$  is stretched to the range of  $[0, L]$ . In general,  $lo$  and  $hi$  can be determined in two ways. (1) Assign  $lo$  and  $hi$  with pre-defined intensity levels. (2) Adaptively determine  $lo$  and  $hi$  according to pre-defined percentage thresholds  $P_l$  and  $P_h$  ( $0 \leq P_l < P_h \leq 1$ ).  $lo$  satisfies  $P(lo - 1) < P_l$  and  $P(lo) \geq P_l$ , and  $hi$  satisfies  $P(hi - 1) < P_h$  and  $P(hi) \geq P_h$ .

The contrast stretching method works well when most of pixel values in the original image are concentrated in a relatively narrow range. In other words, the dynamic range of the histogram is not fully used. The unoccupied dynamic range provides the potential expansion space for the contrast stretching method. Figure 2 shows a good enhancement result by using contrast stretching. However, if most of pixel values in the original image already occupy the full dynamic range of the histogram, there will be no potential expansion space for contrast stretching, then the method fails. Figure 3 shows a failed example by contrast stretching. In summary, the contrast stretching method can provide the good global contrast in terms of the histogram of the whole image, but it fails to guarantee the good local contrast.

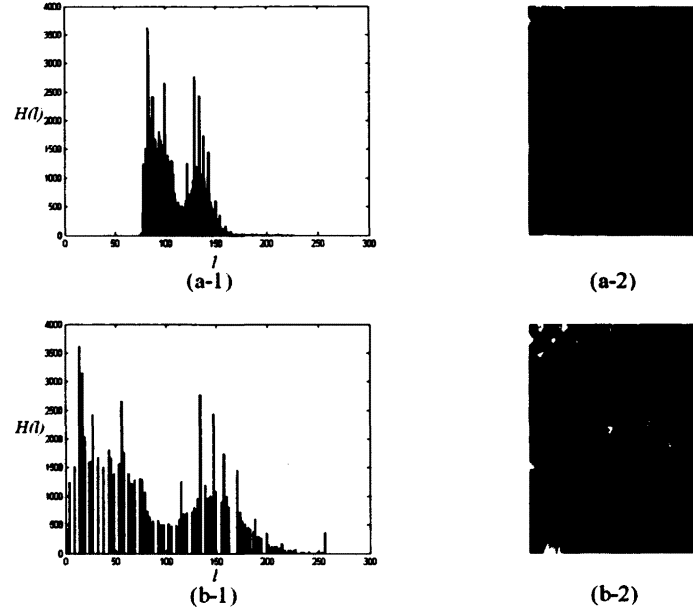


FIG. 2: (a-1) is the histogram of the original image (a-2); (b-1) is the histogram of the enhanced image (b-2). ( $P_l = 0.5\%$ ,  $P_h = 99.5\%$ )

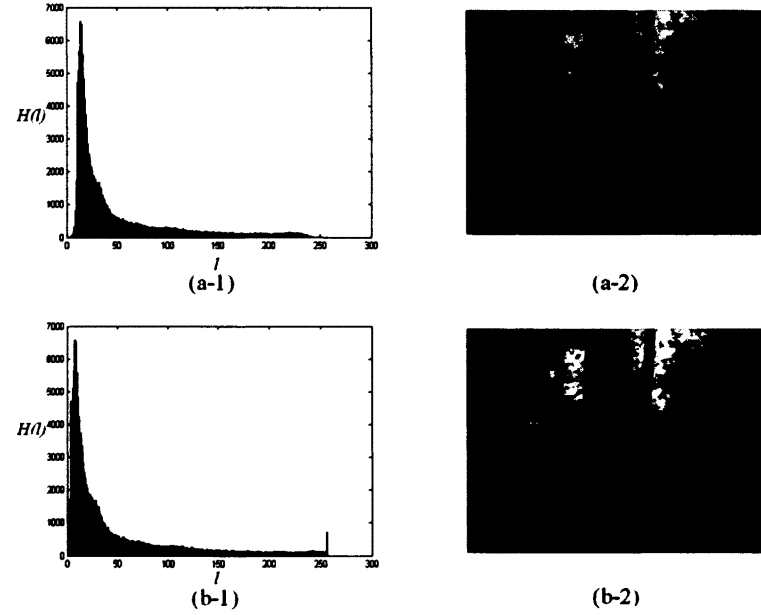


FIG. 3: (a-1) is the histogram of the original image (a-2); (b-1) is the histogram of the enhanced image (b-2). ( $P_l = 0.5\%$ ,  $P_h = 99.5\%$ )

## 2.2 HISTOGRAM EQUALIZATION

The transform of histogram equalization [29] is defined as

$$T(l) = \lfloor P(l) \times L \rfloor. \quad (3)$$

The advantage of histogram equalization is that it can be used to boost contrast by enlarging the distance between peaks in the histogram, especially at the lower end. However, its disadvantage is that the cumulative intensity level distribution  $P(l)$  may quickly climb to a relatively high value so that the dynamic range at the higher end of the histogram is actually suppressed. As shown in Figure 4, due to the major peak located at the lower end of the histogram,  $P(l)$  saturates quickly, so many places of the original image are over enhanced. Thus, the contrast of the brighter part of the image decreases, though the contrast of the darker part of the image increases.

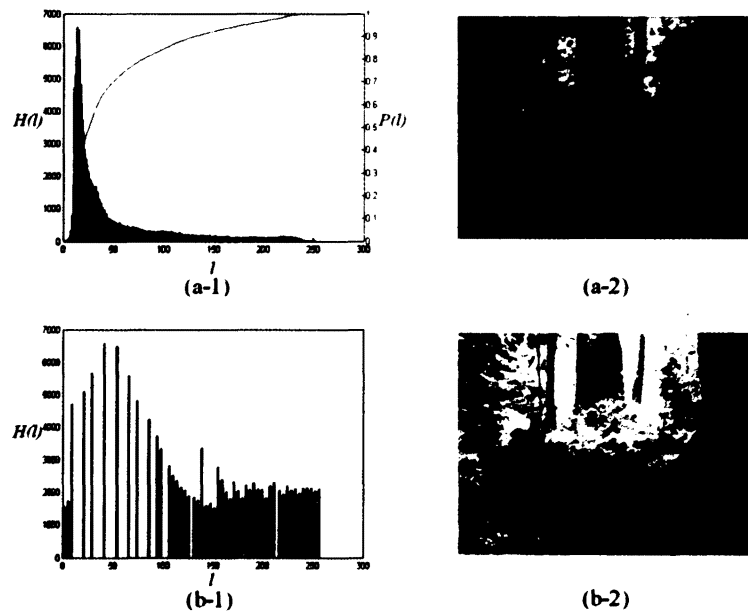


FIG. 4: An example of histogram equalization.

### 2.3 GAMMA CORRECTION

Gamma correction [31] is a transformation defined as

$$T(l) = \left(\frac{l}{L}\right)^{1/\gamma} \times L, \quad (4)$$

where  $\gamma$  is the parameter that determines the shape of the transfer curve, as shown in Figure 5. It can be used to enhance the brightness of the entire image, which is captured under poor illumination, when  $\gamma$  is less than 1, and vice versa. This method actually suppresses the global contrast of the image, though local contrast at the darker end of the histogram may increase due to the curvature of the gamma transformation. As shown in Figure 6, the dark part of the original image becomes greatly enhanced after the gamma correction and many more details are revealed after the enhancement.

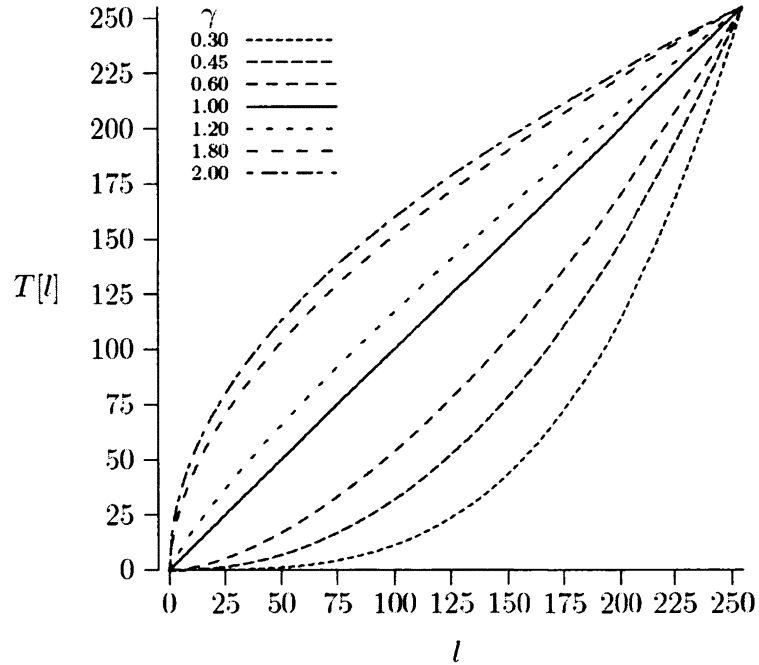


FIG. 5: A group of gamma transforms

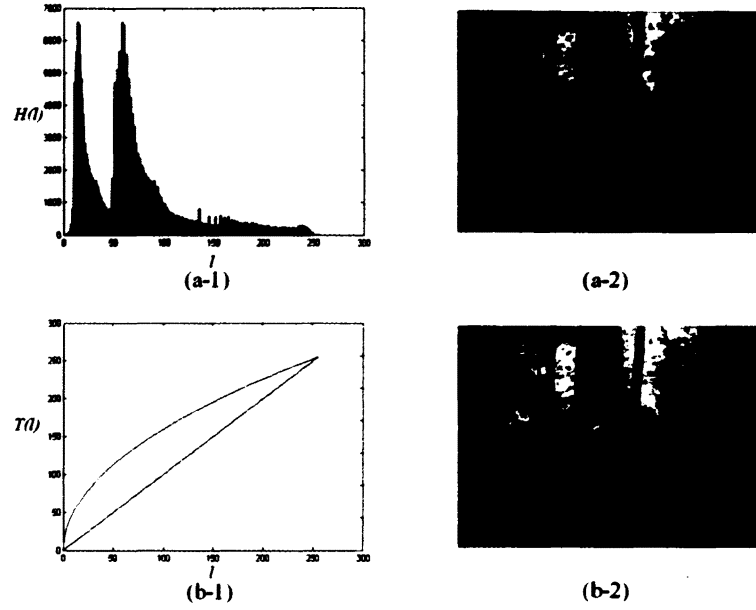


FIG. 6: An example of gamma correction.

#### 2.4 INVERSE HYPERBOLIC TANGENT SHAPED CORRECTION

To compensate the saturation problem of the gamma correction, an inverse hyperbolic tangent shaped correction is proposed [32]. The inverse hyperbolic tangent function is defined as

$$\tanh^{-1}(x) = \frac{1}{2} \log \frac{1+x}{1-x}, x \in (-1, 1). \quad (5)$$

It centers at the origin  $(0,0)$  and it is undefined at  $x = -1$  and  $x = 1$ . To make it applicable in the application of image enhancement, it must be normalized from the interval  $(-1, 1)$  to  $[0, L]$ . In addition, the values at  $l = 0$  and  $l = L$  are defined by the values of its neighboring points. Thus, the inverse hyperbolic tangent shaped

function can be written as

$$T(l) = \left\lfloor \frac{f(l) - f(0)}{f(L) - f(0)} \times L \right\rfloor, \quad (6)$$

$$f(l) = \begin{cases} 2f(1) - f(2) & l = 0, \\ \beta \log \frac{1 + \bar{l}}{(1 - \bar{l})^\alpha} & 1 \leq l \leq L - 1, \\ 2f(L - 1) - f(L - 2) & l = L, \end{cases} \quad (7)$$

$$\bar{l} = \frac{2l}{L} - 1, \bar{l} \in [-1, 1] \quad (8)$$

where the transformation  $T(l)$  is normalized from  $f(l)$ ,  $\alpha$  and  $\beta$  are two parameters used to tune the curvature of  $T(l)$ . When  $\alpha = 1$  and  $\beta = 0.5$ ,  $f(l)$  is the inverse hyperbolic tangent function. Figure 7 shows a group of inverse hyperbolic tangent shaped correction functions ( $\beta = 0.5$ ). A comparison of enhanced results with gamma correction is shown in Figure 8.

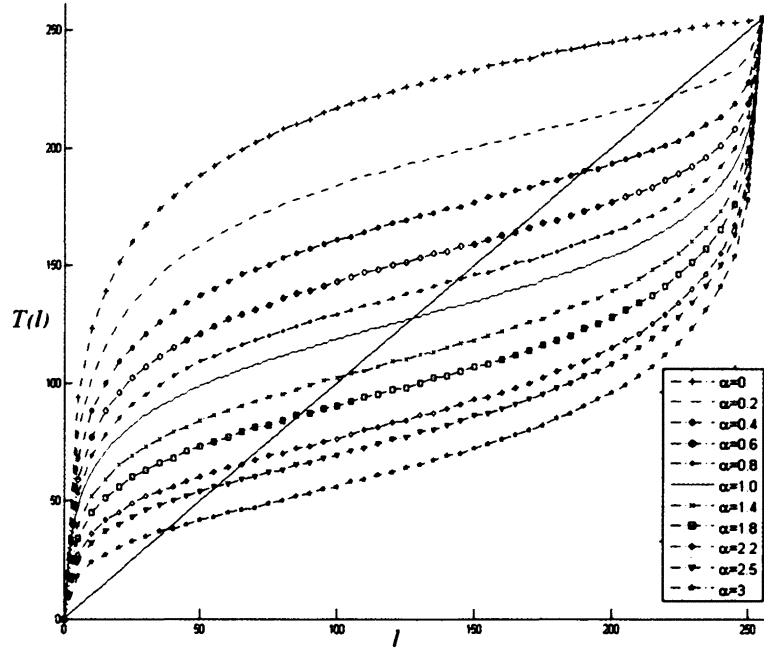


FIG. 7: A group of inverse hyperbolic tangent shaped correction functions

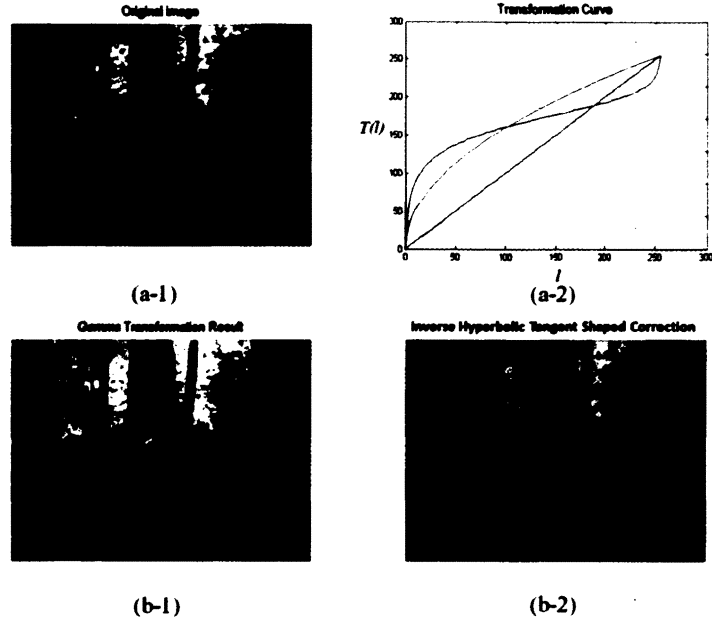


FIG. 8: An example of inverse hyperbolic tangent shaped correction

## 2.5 RETINEX

The non-linear image enhancement method multi-scale retinex with color restoration [24, 25] (MSRCR) is applied to ameliorate the effect of environmental factors, and improve the contrast and sharpness of images, if the images are captured under poor illumination or weather conditions.

### Single-Scale Retinex

The basic form of the single-scale retinex (SSR) is given by

$$R_i(x_1, x_2) = \log I_i(x_1, x_2) - \log[F(x_1, x_2) \otimes I_i(x_1, x_2)], \quad (9)$$

where  $(x_1, x_2)$  are coordinates in the two-dimensional image coordinate system, “ $\otimes$ ” represents the convolution operator,  $I_i$  ( $i = 1, 2, \dots, N$ ,  $N = 3$  for RGB images)

denotes the  $i$ th band of the  $N$ -band input image  $I$ ,  $R_i$  is the corresponding output to  $I_i$ , and  $F$  is the surround function, which is defined as,

$$F(x_1, x_2) = \kappa \exp \left[ -\frac{(x_1^2 + x_2^2)}{\sigma^2} \right], \quad (10)$$

$$\kappa = \frac{1}{\sum_{x_1} \sum_{x_2} F(x_1, x_2)}, \quad (11)$$

where  $\sigma$  is the scale that controls the extent of the surround and  $\kappa$  is the normalizing factor to make the sum of  $F(x_1, x_2)$  equal to one. Smaller values of  $\sigma$  lead to narrow surrounds, and vice versa.

A brief interpretation of the mechanism of SSR is provided here to explain how the enhancement is achieved. First, Equation (9) can be written in the following form,

$$R_i(x_1, x_2) = \log \frac{I_i(x_1, x_2)}{[F(x_1, x_2) \otimes I_i(x_1, x_2)]} = \log \frac{I_i(x_1, x_2)}{\tilde{I}_i(x_1, x_2)} = \log r_i(x_1, x_2). \quad (12)$$

For a pixel  $(x_1, x_2)$ , the convolution of the Gaussian surround function  $F$  and the input image band  $I_i$  can be interpreted as the weighted average value of the neighborhood of  $I_i(x_1, x_2)$ , and the ratio of the original value  $I_i(x_1, x_2)$  to the weighted average value  $\tilde{I}_i(x_1, x_2)$  embodies the “center/surround” concept and can be used to find details in the neighborhood of  $I_i(x_1, x_2)$ . For example, if the neighborhood of  $I_i(x_1, x_2)$  is very smooth, that is, the neighboring pixels of  $I_i(x_1, x_2)$  have similar values of  $I_i(x_1, x_2)$ , then  $\tilde{I}_i(x_1, x_2)$  will be almost identical to  $I_i(x_1, x_2)$ , resulting in  $r_i(x_1, x_2)$  being very close to one. On the other hand, if a change occurs at  $(x_1, x_2)$ , that is,  $I_i(x_1, x_2)$  is greater or less than the values in the neighborhood, then  $\tilde{I}_i(x_1, x_2)$  will be greater or less than one. Therefore,  $r_i(x_1, x_2)$  itself is an indicator of changes. Positive changes lead to values greater than one, and vice versa.

The usage of logarithm function is the critical point that leads the retinex algorithm to success. First, when  $r_i(x_1, x_2)$  is close to one, the linearity is preserved because the slope of the logarithm curve within a small range around one is about one. Second, when  $r_i(x_1, x_2)$  rapidly increases over one, the slope of the logarithm curve is much less than one, so  $R_i(x_1, x_2)$  will slowly increase. This compresses the dynamic range at the bright end to avoid saturation. Third, when  $r_i(x_1, x_2)$  fast drops below one, the slope of the logarithm curve is much greater than one, so  $R_i(x_1, x_2)$  will sharply decrease. As a result, the dynamic range at the dark end gets dramatically expanded. In summary, by using the properties of the logarithm curve, pixels in dark zones get boosted and those in bright zones do not get over-brightened so that a full dynamic range is achieved.

In addition, it is worth noting that the selection of the value of  $\sigma$  plays an important role in the surround function  $F$ . Figure 9 shows an example from the original paper. (a) is the input image. (b), (c), (d) are, respectively, the SSR outputs by using narrow, medium, and wide surround functions. The narrow surround function acts as a high-pass filter, capturing all the fine details in the image but at a severe loss of tonal information. The wide surround function captures all the fine tonal information but at the cost of dynamic range. The medium surround function acts as an intermediate between these two. Thus, multiple surrounds were found necessary in order to achieve a graceful balance between dynamic range compression and tonal rendition.

### Multi-Scale Retinex

The motivation of MSR is to combine the advantages of using different scales of surrounds in order to improve the contrast and keep the tonal information. Intuitively, an MSR output can be expressed in a weighted summation form of several

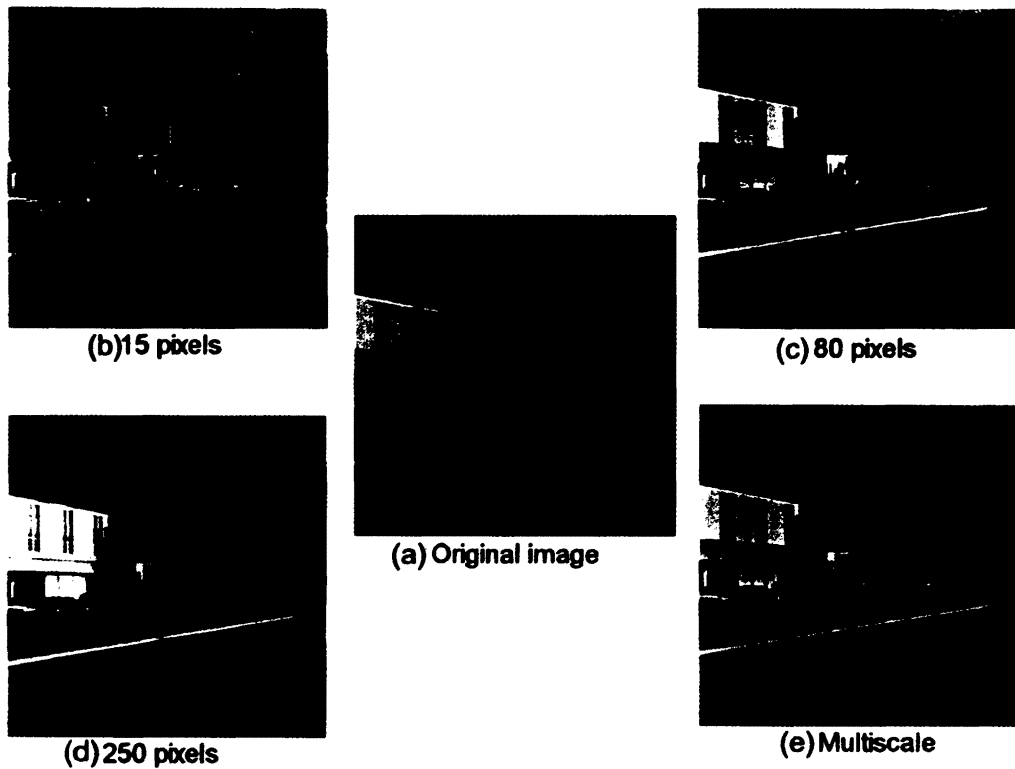


FIG. 9: (a) The original input (b) Narrow surround (c) Medium surround (d) Wide surround (e) MSR output. The narrow-surround acts as a high-pass filter, capturing all the fine details in the image but at a severe loss of tonal information. The wide-surround captures all the fine tonal information but at the cost of dynamic range. The medium-surround captures both dynamic range and tonal information. The MSR is the average of the three renditions.

SSR outputs as follows,

$$R_i(x_1, x_2) = \sum_{k=1}^K W_k (\log I_i(x_1, x_2) - \log [F_k(x_1, x_2) \otimes I_i(x_1, x_2)]), \quad (13)$$

where  $K$  is the total number of different scales of surround functions, and  $W_k$  is the weight assigned to the  $k$ th SSR output generated by using the  $k$ th surround function  $F_k$  ( $k = 1, 2, \dots, K$ ).  $I_i$  and  $R_i$  are, respectively, the  $i$ th spectral band of the input image  $I$  and the retinex output image  $R$ . The total number of scales used

for the MSR  $K$  is, of course, application dependent. Rahman et al. [24, 25] have found empirically, however, that a combination of three scales representing narrow, medium, and wide surrounds is sufficient to provide both dynamic range compression and tonal rendition. Figure 9 (e) shows the MSR output of (a) when three surround functions are applied together, which is obtained by equally weighting, i.e.  $W_k = 1.0$  for  $k = 1, \dots, K$  ( $K = 3$ ) in Equation (13). As is evident from Figure 9, none of the individual scales attain the goal, visual realism. The narrow and medium surround cases are self-explanatory. The wide-surround case deserves some discussion because it produces a reasonable output image. However, the lack of dynamic range obscures the features that were visible to the observer, hence it fails the test. The MSR processed image uses features from all three scales to provide simultaneous dynamic range and tonal rendition.

### Multi-Scale Retinex with Color Restoration

Due to the mechanism of retinex algorithm shown in Equation (9) and (13), tonal information is often lost when a relatively narrow surround function is used, because the retinex processing focuses on the ratio of the center value to the weighted average value of the surround. Thus, a “graying out” of the image, either in specific regions or globally, is the general effect of retinex processing on images with regional or global gray-world violations. This de-saturation of color can, in some cases, be severe. Therefore the desired color computation can be considered a color restoration, which should produce good color rendition for images with any degree of graying. In addition, the correction is expected to preserve a reasonable degree of color constancy since that is one of the basic motivations for the retinex. To achieve that goal, a color restoration factor  $\alpha$  is computed based on the following transform:

$$\alpha_i(x_1, x_2) = f[I_i(x_1, x_2) / \sum_{n=1}^N I_n(x_1, x_2)], \quad (14)$$

where  $\alpha_i(x_1, x_2)$  is the color restoration coefficient in the  $i$ th spectral band,  $N$  is the number of spectral bands,  $I_i$  is the  $i$ th spectral band in the input image, and  $f(\cdot)$  is some mapping function. In the original paper [25],  $f(\cdot)$  is picked to be a logarithm function, so  $\alpha$  is obtained as

$$\alpha_i(x_1, x_2) = \log[I_i(x_1, x_2) / \sum_{n=1}^N I_n(x_1, x_2)]. \quad (15)$$

In a purely empirical fashion, this was tested on several images and proved to restore color rendition, encompassing both saturated and less saturated colors. Adding this to Equation (13), the multi-scale retinex with color restoration (MSRCR) is given by:

$$R_i(x_1, x_2) = \alpha_i(x_1, x_2) \sum_{k=1}^K W_k (\log I_i(x_1, x_2) - \log[F_k(x_1, x_2) \otimes I_i(x_1, x_2)]). \quad (16)$$

The results of applying this transformation to the “monochrome” images are shown in Figure 10.

Some examples are shown in Figure 11, on the left column are original aerial images with low contrast and sharpness, and on the right column are results after enhancement. It is clear that the contrast and sharpness of the original images is significantly improved by using MSRCR. Therefore, more details can be seen on the ground by the pilot under poor weather conditions and illumination. The restored color information is also helpful to the pilot. In addition, this pre-processing step provides a good input for the subsequential landing-site detection algorithm.

## 2.6 EXPERIMENTAL RESULTS

To compare the performance of the four enhancement methods described above, each of them was applied on the same set of 10 aerial images. Two examples are

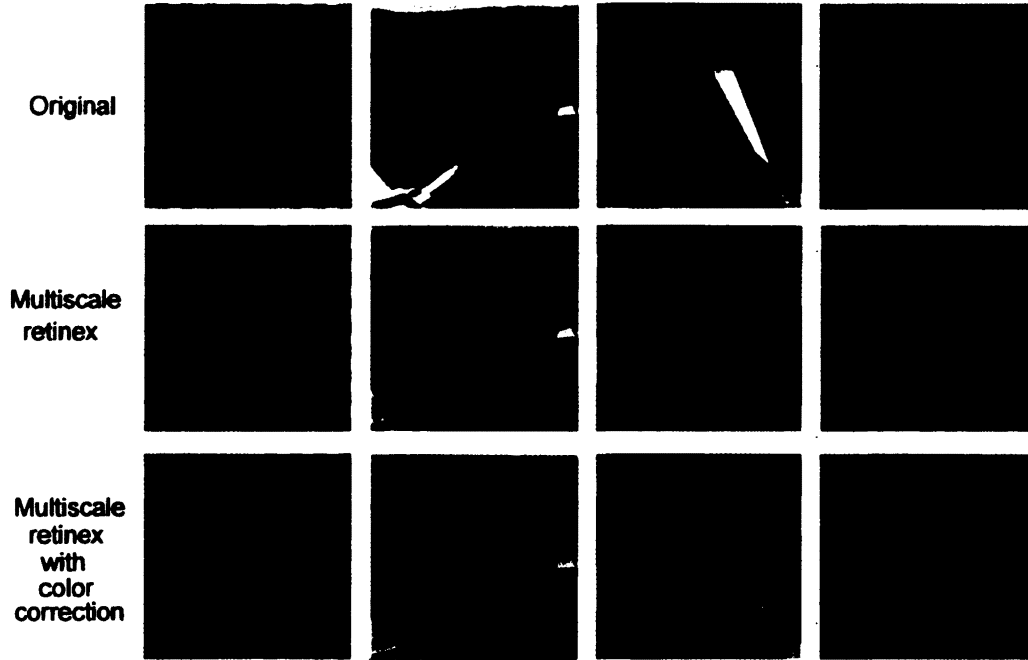


FIG. 10: (Top row) Scenes that violate the gray-world assumption; (Middle row) the MSR output.

shown in Figure 12 and Figure 13. The original image (see (a)) is foggy and the bounds of hazardous obstacles are blurred. The enhanced results by using contrast stretching (see (b)) and histogram equalization (see (c)) improved the global contrast in the whole image, but due to the intrinsic mechanism of these two methods, some bright areas in the original image become brighter in the enhancement results so that some details are lost. The results generated by using Gamma correction (see (d)) and inverse hyperbolic tangent shaped correction (see (e)) failed to eliminate the fog because the dynamic range is suppressed. The MSRCR produces relatively better enhancement results (see (f)) compared to afore mentioned methods. It improves the illumination and local contrast of the original image without introducing artifacts caused by over-enhancement.

## 2.7 DISCUSSION

Contrast stretching, histogram equalization, and Gamma transform are widely used effective image enhancement methods. They are easy to understand and implement. But the drawback of those global enhancement methods is that it does not fully utilize the local information. Multi-scale retinex with color restoration (MSRCR) is a relatively advanced image enhancement algorithm which makes use of local contrast information. Experimental results show that MSRCR performs better than the other three for the specific applications considered in this study.

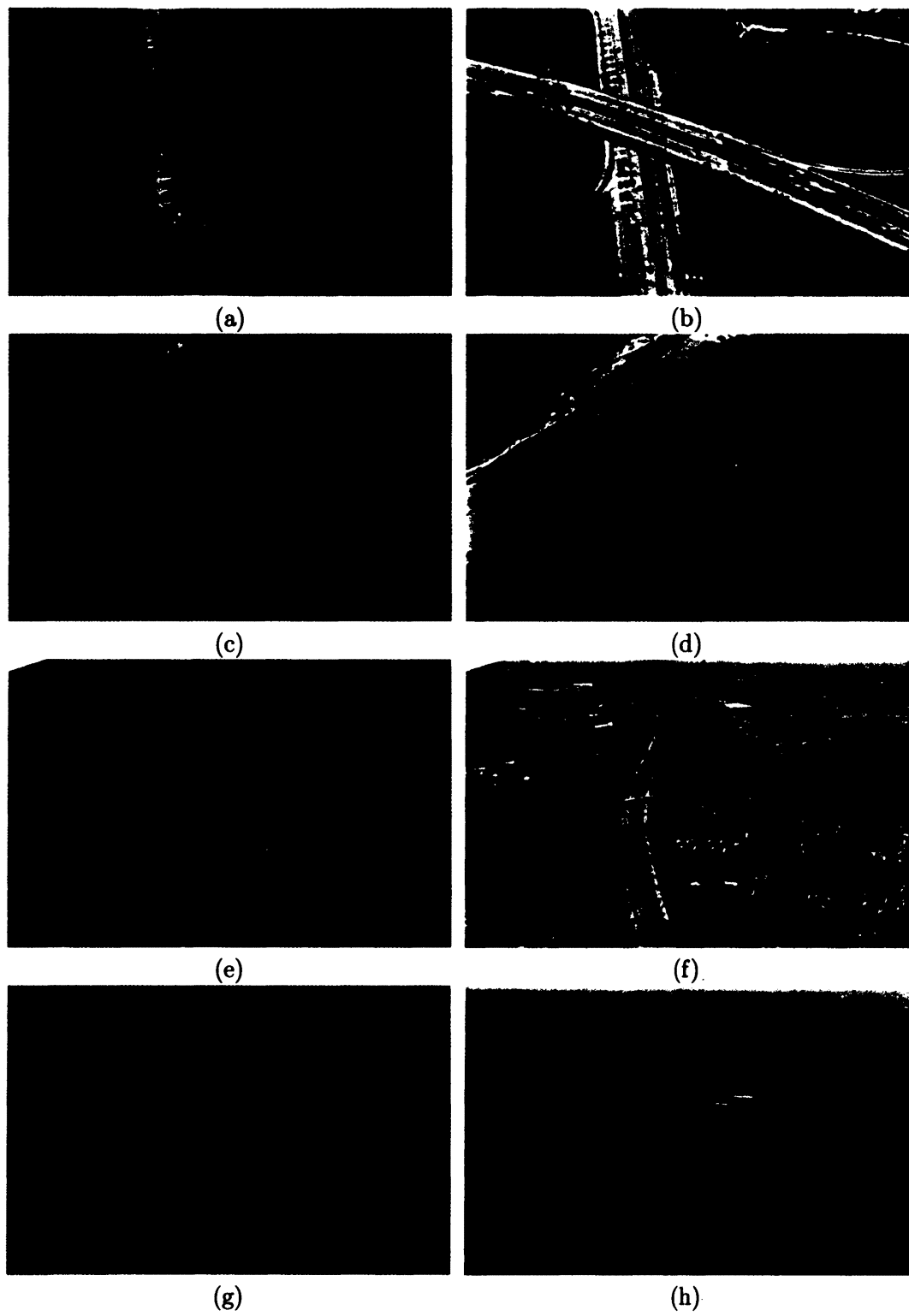


FIG. 11: MSR application on aerial images

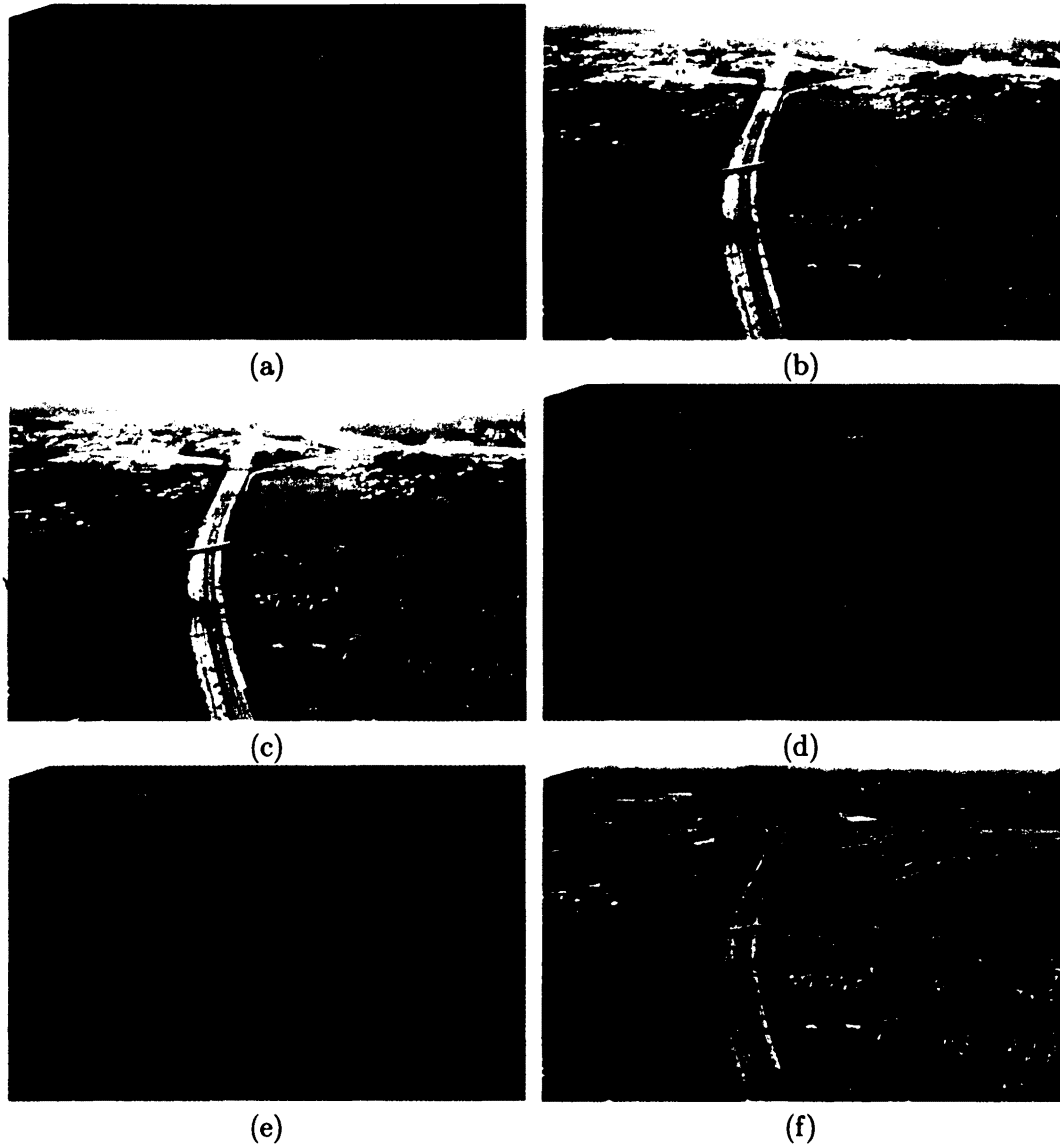


FIG. 12: Example 1: (a) Original aerial image; (b) Enhanced by contrasting stretching; (c) Enhanced by histogram equalization; (d) Enhanced by Gamma correction; (e) Enhanced by inverse hyperbolic tangent shaped correction; (f) Enhanced by MSRCR.

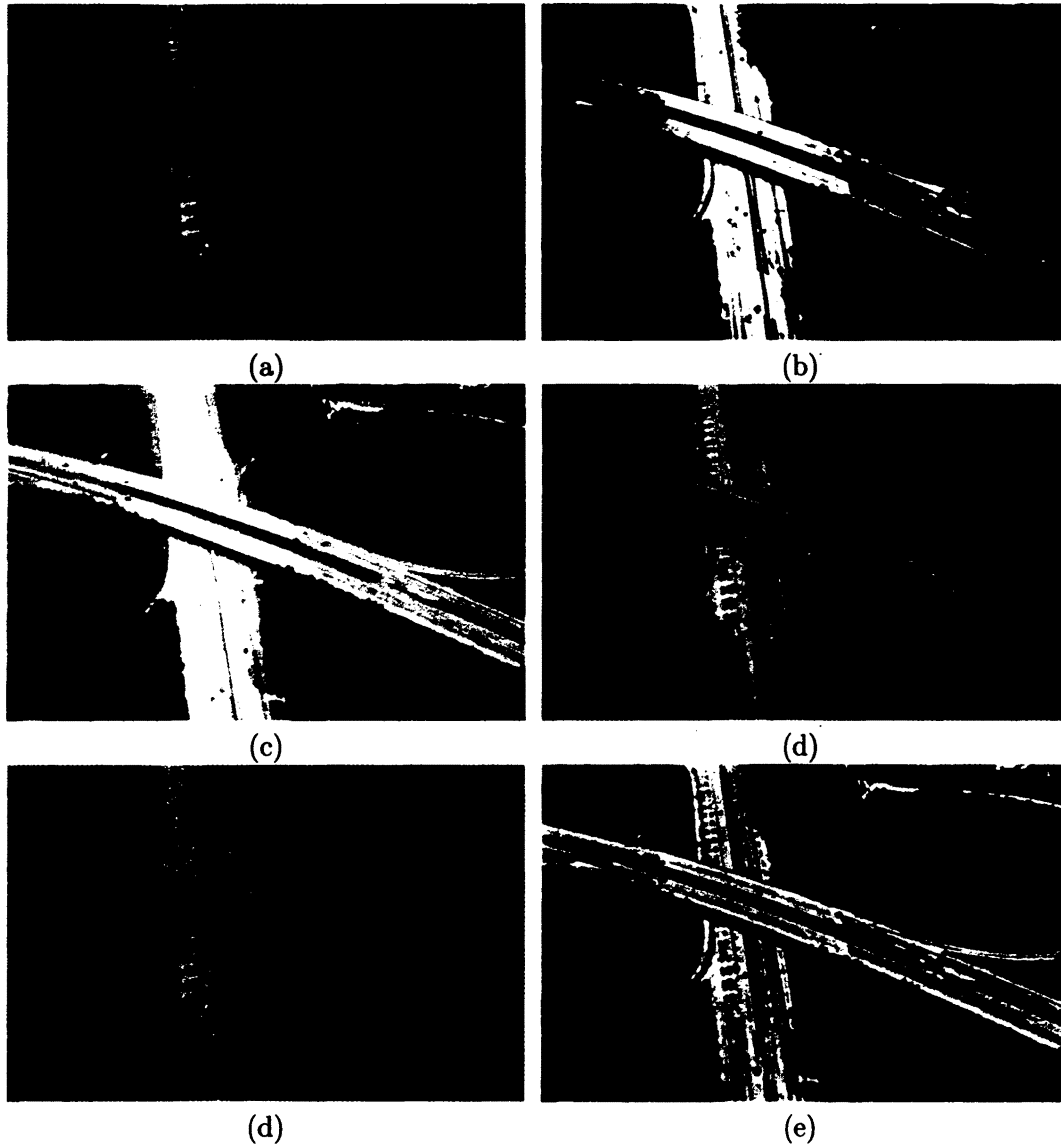


FIG. 13: Example 2: (a) Original aerial image; (b) Enhanced by contrasting stretching; (c) Enhanced by histogram equalization; (d) Enhanced by Gamma transform; (e) Enhanced by inverse hyperbolic tangent shaped correction; (f) Enhanced by MSRCR.

## CHAPTER 3

### HORIZON DETECTION

Horizon detection often serves as a preliminary step of many aerial or ground-based robotic systems [33–35]. A reliable segmentation of the sky and the ground plays an important role in subsequent steps. Much previous work has been conducted in horizon detection. In this chapter, three existing horizon detection algorithms are first reviewed and then the proposed algorithm is presented.

#### 3.1 INTENSITY-BASED HORIZON DETECTION

Williams and Howard proposed a horizon detection algorithm for a specific ground-based rover application to segment the foreground plane from distant mountains and the sky in glacial environments [36]. Due to the specialty of that application, two strong but reasonable assumptions are made: (1) It is assumed that the bottom third of image is ground, because the camera is mounted on a ground-based rover. (2) The ground is assumed to appear all white with very little variance, because the rover is in glacial environments. Based on those two assumptions, they proposed an intensity-based horizon detection algorithm, which can be summarized as:

**Step 1** Generate an edge map by applying Canny edge detector [15] to the original image.

**Step 2** Scan the edge map column by column. In each column, an edge point can be considered a point of the horizon only if pixels below it in that column appear all white with little variance.

**Step 3** Connect those horizon points detected in Step 2 to form the horizon after removing noisy points and interpolation.

The problem of this intensity-based algorithm is that the two assumptions often fail in other environments, thus it will not be further considered in the present dissertation.

### 3.2 HOUGH TRANSFORM-BASED HORIZON DETECTION

Dusha et al. [37] applied Hough transform [38] to detect the horizon based on the assumption that the horizon is the strongest boundary in the image. Their method can be summarized as:

**Step 1** Generate a binary edge map by applying Canny edge detector [15] to the original image.

**Step 2** Apply Hough transform to recognize straight lines.

**Step 3** Pick the line that has the highest votes by Hough transform as the horizon.

Hough transform is widely used to identify straight lines. However, that assumption is not always valid and that method can be easily disturbed by the presence of other strong edges. Therefore, a hard decision of selecting the highest voted line by Hough transform as the horizon is unreliable. One failed example is shown in Figure 14. A soft decision strategy is brought in to solve this problem and is discussed in Section 3.4.



FIG. 14: A failed example by using Hough transform only



FIG. 15: A failed example by using the greedy search method

### 3.3 A GREEDY SEARCH ALGORITHM

Ettinger et al. proposed a horizon detection method in a greedy search manner [35]. The method is based on two assumptions: (1) The horizon is a straight line that partitions the image into two regions, namely sky and ground. (2) There is little variance in either region, i.e. pixel characteristics of the sky region are consistent and different from the ground region, and vice versa. Thus, the detection of the horizon becomes to search for the optimal straight line such that the sum of the variances of both regions reaches the lowest value. Lines of all possible locations and angles are tested, and the optimal one that meets the above criterion is considered the horizon. The implementation procedure can be summarized as:

**Step 1** Down-sample the original image to a smaller size for computational concern.

**Loop:** for angle =  $0^\circ$  to  $180^\circ$

**Step 2** Compute the range of possible y-intercept based on the angle.

**Loop:** for y-intercept in the range computed in Step 2

**Step 3** Compute the position of that line segment in the image.

**Step 4** Partition the image into two parts and compute the pixel

intensity variance in each part.

**Step 5** Save the sum of results produced in Step 4.

**End**

**End**

**Step 6** Find the line with the largest value produced in Step 5.

**Step 7** Repeat the above double loops in the original image within a narrow dual side range of the angle and the y-intercept found in Step 6.

**Step 8** Select the line with the largest sum of two variances as the horizon.

Two problems of this method are: (1) It is computationally expensive due to the greedy search scheme. (2) The second assumption fails when large objects having similar colors to the sky appear in the ground region of the image, such as lakes, rivers, or other large bodies of water. One failed example is shown in Figure 15.

### 3.4 PROPOSED METHOD

In this study, an automatic horizon detection algorithm with a hierarchical strategy, in which the non-horizon pixels are gradually excluded step by step, is proposed [39]. The proposed algorithm is designed with several relatively soft decisions in order to adapt all possible situations. Specifically, it consists of a coarse level detection and a fine level adjustment. First, the original image is blurred by a large scale low-pass filter. Then, Canny Edge detector [15] and Hough transform [38] are successively utilized to find major edges in the image and identify lines associated with those major edges. The desired horizon is modeled by one of the lines. Since the horizon is often not a straight line, an elastic fine level adjustment is applied to capture the curvature of the horizon.

The proposed hierarchical horizon detection algorithm consists of a coarse level detection and a fine level adjustment. Three steps, namely, (1) major edge detection,

(2) line fitting, and (3) coarse adjustment, comprise the coarse level detection. (1) The original image as shown in Figure 16 is blurred by a large scale low-pass filter to remove relatively fine edges and retain relatively major edges in the image. In the present paper, a Gaussian low-pass filter is employed with a relatively large sigma  $\sigma$  which is adaptive to the dimension of the image, i.e.  $\sigma = N_V/50$ , where  $N_V$  is the number of rows in the image. Then, a Canny edge detector [15] is applied to the blurred image to find major edges. The Canny edge detector is used because it provides edge strength information in addition to edge location information. (2) An edge strength histogram is computed based on the edge strength produced by the Canny edge detector, and the top  $p\%$  of the points are obtained as possible points that comprise the horizon. On one hand,  $p$  should be a small number because the horizon is often the strongest edge in the image. On the other hand, it is not always the case, so  $p$  should not be too small. In other words, it would be better to conservatively include some non-horizon points in this step and exclude them in a later step than to frivolously lose some of the key horizon points in this step. Empirically,  $p\%$  is adaptively obtained as  $5/N_H$ , where  $N_H$  is the number of columns of the image. A binary edge map consisting of the top  $p\%$  strongest edge points can be generated. The standard Hough transform (SHT) [38] is then applied to fit probable lines in the binary edge map. Due to the presence of strong edges in addition to the horizon in the image, the highest peak of the voting result of the SHT does not necessarily correspond to the horizon. Thus, it is unreliable to simply take the highest peak as the horizon. Instead, the top  $N_L$  highest peaks are taken into account by comparing the average edge strength within the dual-side narrow bands along each of them. The line that has the highest average edge strength within its dual-side narrow bands is called the true peak of Hough transformation. Empirically,  $N_L$  is set to 5. This soft decision strategy makes the detection more reliable and robust. In Figure 17, the green line is the true peak of Hough transformation along with four yellow lines

that are among the top  $N_L$  highest peaks but not the true peak. (3) It is necessary to adjust the coarsely detected line, because the real horizon is often not perfectly straight. The coarse level adjustment is to search the dual-side neighborhood of each pixel of the detected line according to the edge map of the blurred image. As shown in Figure 17, yellow arrows denote the directions of the coarse adjustment and the red dots denote the position of each point after the coarse adjustment.

A fine adjustment takes place after the position of horizon is coarsely determined. As illustrated in Figure 18, the dual-side bands of the green coarsely determined horizon are explored based on a fine edge map computed from the original image in which fine edges are retained. Pixels of the coarsely detected horizon are then adjusted again to positions that meet certain criteria. The criterion used in [3] is to check edge pixels within the dual-side bands and find the one that has the largest edge strength. In general, it works well. However, strong edge patterns occasionally appear very close to the horizon, and the pixel of the horizon may be not the strongest edge point in its neighborhood. Therefore, in the present paper, a set of the strongest pixels  $EP$  is examined, which is defined as

$$EP = \{E_i | ES_i \geq ES_1 \times 0.8, i = 1, 2, \dots, N_E\}, \quad (17)$$

where  $N_E$  is the total number of edge pixels in that dual-side narrow band,  $E_i$  ( $i = 1, 2, \dots, N_E$ ) is the  $i$ th strongest edge pixel in that dual-side neighborhood, and  $ES_i$  is the edge strength of  $E_i$ . Let  $N_{TE}$  be the number of points in  $EP$  and  $TE_j$  ( $j = 1, 2, \dots, N_{TE}$ ) be the  $j$ th strongest edge point in  $EP$ . A modification of the idea of [35] mentioned above is employed by computing the variance of  $TE_j$ 's dual-side neighborhood. When validated on a test image database, in most cases, the true horizon pixel is the edge pixel that has lowest variances of its dual-side neighborhood.

This is formulated as

$$v_j = \sqrt{\text{var}(NB_j^+) \times \text{var}(NB_j^-)}, \quad (18)$$

where  $NB_j^+$  and  $NB_j^-$  are, respectively, the upper and lower neighborhoods of  $TE_j$ , and  $v_j$  is the variance measurement to be evaluated.  $TE_H$  is considered the pixel of the horizon when its corresponding  $v_H$  is the minimum value of  $v_j$ . Noisy points may emerge due to the discontinuity of the horizon in the fine edge map. These discontinuous points are removed by interpolation based on their neighboring points on the left and right. In this paper, the B-Spline interpolation method [40, 41] is used. After removing all the noisy points, a smoothing technique is applied to local segments of the detected horizon to get the smooth final detection result as shown in Figure 19.

### 3.5 EXPERIMENTAL RESULTS

The proposed algorithm is tested on 100 images provided by Google Earth®, which are captured in various view angles, over different types of terrains, and at different elevations that range from 1000 to 30000 ft. On the same image set, the greedy search method [35], the Hough transform method [37], and the hierarchical method [3] are tested. The quantitative performance measurement, average maximum bias (AMB) [3], is utilized. AMB is defined as

$$AMB = \frac{1}{H_M} \sum_{h=1}^{H_M} MB_h, H_M = L_H \times 0.1, \quad (19)$$

where  $L_H$  is the total pixel number of the detected horizon,  $H_M$  is one tenth of  $L_H$ ,  $MB_h$  ( $h = 1, 2, \dots, H_M$ ) is the  $h$ th maximum bias from the detected horizon to the position of the true horizon in the unit of pixel. Evaluating the average bias of the worst detected segment can reveal the true performance of the method, avoiding the significant bias to be canceled out by well-aligned segments, that is, if the most

biased segment of the detection result can be considered satisfactory, the rest of the detected horizon can be guaranteed to be better than the most biased segment. Figure 20 shows the horizon detection results by using the four methods. Five AMB intervals are used, which are, respectively, according to perfect detection ( $AMB \leq 1$ ), good detection ( $1 < AMB \leq 5$ ), acceptable detection, ( $5 < AMB \leq 10$ ), biased detection ( $10 < AMB \leq 20$ ), and false detection ( $AMB > 20$ ). The percentage shown above the bar of each AMB interval reflects the ratio of the number of images, of which the AMB value falls into that AMB interval, to the total number of tested images. The greedy search method finds the true horizon in 69% of the test (33% perfect detection and 36% good detection), though it fails in 25% of the test when sea, rivers or anything that has similar colors to the sky appear in the ground part of the image. It cannot achieve better detection because of the straight line assumption, though the general position of the horizon is detected. The Hough transform method, due to its intrinsic mechanism, is easily corrupted by other non-horizon edges shown in the image. This explains why it only achieves 5% perfect detection but a high rate of good and acceptable detection (43% and 45%). In particular, when the non-horizon edges are stronger than the horizon, it fails as shown in the 7% of false detection. The proposed method achieves 94% perfect detection and 4% good detection, showing the improvement to the hierarchical method [3], which generates 86% perfect detection and 9% good detection. It is worth noting that the latter two methods produced no false detections on the test images. Figure 21 shows a comparison of detection results generated by the four methods, respectively, in yellow, cyan, green, and red. All four methods detect the horizon in (a) and (b). In (c), the former two methods find the general position of the horizon but cannot fit its curvature due to the straight line assumption. The latter two methods accurately find the horizon. In (d), the former two methods fail. The hierarchical method [3] finds the true horizon with minor bias. The proposed method successfully detects the horizon.

### 3.6 DISCUSSION

The experimental results prove the feasibility of the proposed horizon detection algorithm. The accuracy and robustness of detection results generated by the proposed algorithm is achieved by the delicate detection mechanism. It provides a reliable tool for applications which require the horizon detection as a preliminary step.



FIG. 16: A sample image



FIG. 17: Coarse adjustment

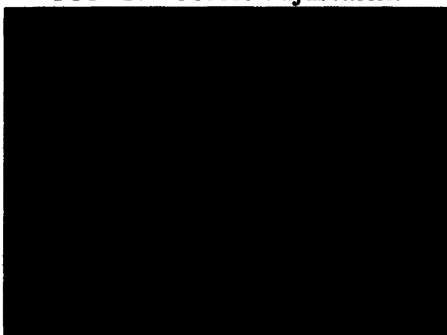


FIG. 18: Fine adjustment



FIG. 19: Detected horizon

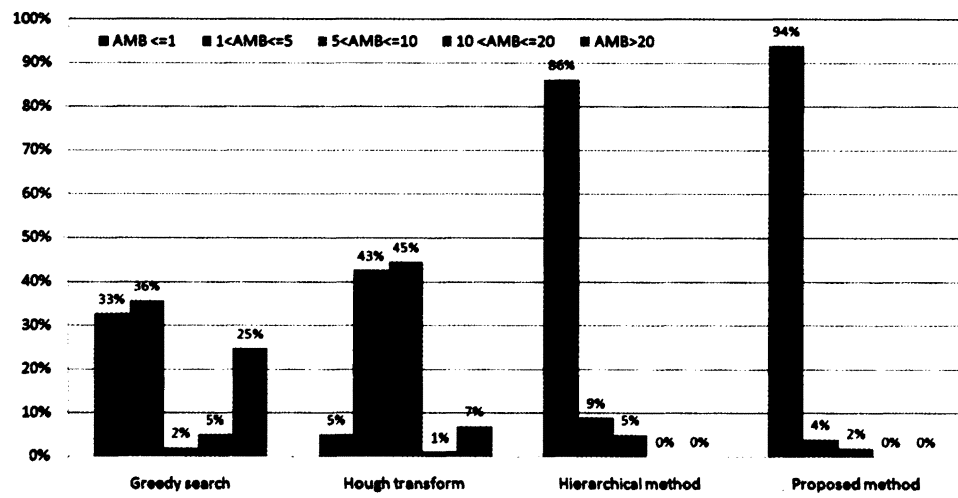


FIG. 20: Horizon detection results by using the greedy search method [35], Hough transform[37], hierarchical method[3], and the proposed method.

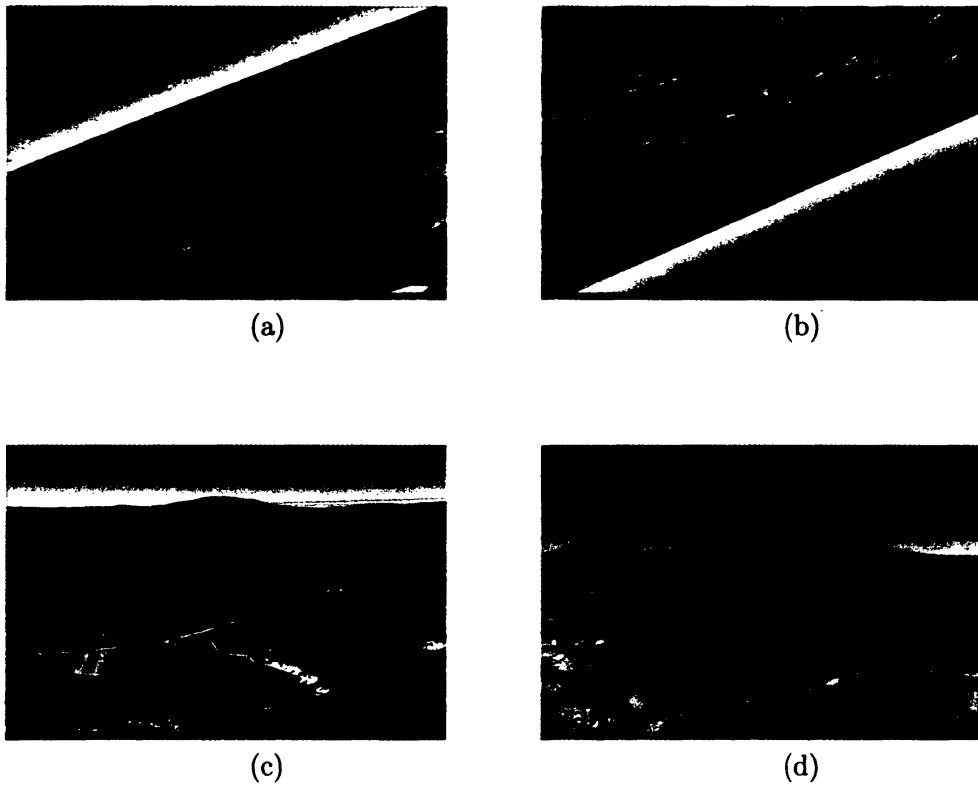


FIG. 21: Detection results by using the greedy search method [35] (yellow), Hough transform[37] (cyan), hierarchical method[3] (green), and the proposed method (red).

## CHAPTER 4

### LANDING-SITE DETECTION

#### 4.1 ROUGHNESS ASSESSMENT

##### 4.1.1 EDGE DETECTION

The roughness of ground and the presence of hazards are often reflected as boundaries and high variance of pixel intensity values in visible images. If high resolution elevation maps are not available, it is plausible to assume that identifying rough areas or hazardous objects on the ground is equivalent to the process of edge detection in visible images. Canny edge detector [15] is an efficient tool for computing the sharpness of edges, which is, from smoothest to sharpest, quantified to the range from 0 to 255. This method is applied at the beginning of the roughness assessment module. Figure 22 shows a sample image provided by Google Earth®, and its edge detection result using the Canny detector is shown in Figure 23. Brighter pixels represents sharper edges and vice versa.

##### 4.1.2 EDGE ANALYSIS

Different edge patterns appear among diverse regions in terms of the edge strength within a certain range. To characterize the difference, the edge map is first divided into un-overlapping blocks. The cumulative hazard strength (CHS) of each block is defined as follows,

$$CHS_B = \sum_{p \in B} H(ES_p), \quad (20)$$

$$H(ES_p) = \begin{cases} 1 & ES_p > T, \\ 0 & ES_p \leq T, \end{cases} \quad (21)$$

where  $ES_p$  is the edge strength of each pixel  $p$  in block  $B$ , and  $H()$  is the hazard indicator function. If  $ES_p$  is greater than the pre-specified safeness threshold  $T$ , the pixel  $p$  is considered hazardous and the CHS of block  $B$ ,  $CHS_B$ , is incremented by 1. In contrast, if  $ES_p$  is no greater than  $T$ , the pixel  $p$  is considered safe and  $CHS_B$  remains the same. Thus, blocks of smooth areas have a zero or low CHS value, but blocks of rough areas have a high CHS value. The block size (BS) in the unit of pixels is adaptively determined based on the height of the camera  $h_c$  in the unit of ft,

$$BS = \begin{cases} 20 \times 20 & h_c \leq 10000 ft, \\ 15 \times 15 & 10000 ft < h_c \leq 20000 ft, \\ 10 \times 10 & h_c > 20000 ft. \end{cases} \quad (22)$$

For example, if the aircraft is flying at a higher elevation, the image covers a relatively larger area on the ground and the realistic size of each pixel is relatively larger compared to the image captured at a lower height. As a result, to keep the consistency of the realistic area of each block to some extent, the block size is set as a smaller number when the image is captured at a higher height, and vice versa. In addition, the pre-specified safeness threshold  $T$  is related to the requirement of acceptable smoothness. A lower value of  $T$  means a stricter requirement for smoothness, because the edge strength of more pixels will be beyond the safeness threshold and they will be considered hazardous. It is more reasonable to utilize a unified strict safeness threshold rather than an adaptive safeness threshold according to the change of  $h_c$ , because loosening the requirement for smoothness as the  $h_c$  increases will bring risk to the landing process. It was empirically determined that  $T$  is not sensitive to the

final detection results if its value is picked in the range from 15 to 30, since the edge strength of hazards' boundaries is usually much higher than 30. In this study,  $T$  is selected to be 20 as a relatively stricter requirement for smoothness.

Two benefits are gained by introducing the CHS in a block process fashion. First, it improves the system reliability against sparsely located noise. For example, the real boundaries between objects usually appear as continuous points in the edge map. However, noise generated in the imaging process often appears as isolated sparse points in the edge map. By counting the number of edge points in each block, the effect of those sparse noisy points can be eliminated. Second, it can improve the computational efficiency for the sequential steps.

## 4.2 IMAGE SEGMENTATION

### 4.2.1 K-MEANS CLUSTERING

The classification module utilizes the K-means clustering method [42, 43] to classify the CHS of each block into a number of clusters. For example, if the number of clusters is specified as seven, the clusters can be interpreted as “very rough,” “rough,” “moderate rough,” “median,” “moderate smooth,” “smooth,” and “very smooth.” The number of clusters is firstly set to seven by default and then automatically reduced in the clustering procedure; that is, if any cluster loses all of its members, that cluster will be removed [42, 43]. Figure 24 shows the clustering results of the sample image shown in Figure 22. In this case, four clusters are obtained: “dark blue” renders the smoothest areas, “red” renders the roughest areas, and “green” and “light blue” represent the areas in between.

### 4.2.2 REGION GROWING

Based on the clustering result, adjacent “smoothest” blocks are merged to form

larger smooth areas by using the region growing method [44–47]. The result of connected areas is shown in Figure 25, where each area is labeled with a unique color. In the interests of efficiency, isolated tiny spots and narrow branches of merged areas can be removed by applying the morphological operation of image erosion [44, 48, 49] without assessing their dimensions, since they are obviously undersized.

### 4.3 DIMENSION ASSESSMENT

After the above steps, potential landing-sites are obtained, as shown in Figure 26. In this module, the realistic dimensions of those potential landing sites are measured to determine which of them are qualified to be candidate landing-sites. The realistic dimensionality of each potential landing-site is measured by converting its size from the image coordinate system to the realistic world coordinate system. In flight dynamics, changing the orientation of the aircraft to any direction can be decomposed to three kinds of rotations: yawing, rolling and pitching, which are respectively to rotate the aircraft along the vertical axis, the longitudinal axis, and the lateral axis. Given those three rotation angles, this procedure can be described by the intrinsic or extrinsic matrices composition [50, 51] with which one can map the world coordinate system to the aircraft coordinate system, and vice versa. In other words, two arbitrary points in an aerial image can be mapped to the world coordinate system so that the realistic distance between the two points on the ground is measurable if the three rotation angles are known. In practice, most aircrafts have the device to record the three angles so that they can be synchronically stored with real time aerial images. Images provided by Google Earth® are used in this pilot study. For this simulation, the imaging process is simplified with only pitching angle but no yawing and rolling angles because the actual information of all three rotation angles is unavailable. Therefore, the imaging model in the vertical direction of the image coordinate system can be described as shown in Figure 28, and the realistic size of

each pixel along the vertical direction of the image can be computed as follows,

$$\begin{aligned}
d_0 &= h_c \cdot \tan(\alpha - \frac{FOV_V}{2}), \\
d_1 &= h_c \cdot \tan(\alpha - \frac{FOV_V}{2} + \theta_1) - d_0, \\
d_2 &= h_c \cdot \tan(\alpha - \frac{FOV_V}{2} + \theta_1 + \theta_2) - d_0 - d_1, \\
&\vdots \\
d_i &= h_c \cdot \tan(\alpha - \frac{FOV_V}{2} + \sum_{j=1}^i \theta_j) - \sum_{k=0}^{i-1} d_k, \\
&\vdots \\
d_{N_V} &= h_c \cdot \tan(\alpha - \frac{FOV_V}{2} + \sum_{j=1}^{N_V} \theta_j) - \sum_{k=0}^{N_V-1} d_k,
\end{aligned} \tag{23}$$

where  $h_c$  is the height of camera,  $\alpha$  is the pitching angle,  $FOV_V$  is the field of view along the vertical direction of the image,  $N_V$  is the total number of pixels along the vertical direction,  $d_0$  is the distance between the vertical line and the first pixel,  $\theta_i$  and  $d_i$  ( $i = 1, 2, \dots, N_V$ ) are respectively the angle and the realistic distance corresponding to pixel  $p_i$  along the vertical direction. For large  $N_V$ ,  $\theta_i$  ( $i = 1, 2, \dots, N_V$ ) can be considered to have the same approximate value  $\theta$  so that Equation (23) can be simplified as

$$d_i = h_c \cdot \tan(\alpha - \frac{FOV_V}{2} + i\theta) - \sum_{k=0}^{i-1} d_k, \quad \theta = \frac{FOV_V}{N_V}. \tag{24}$$

In addition, since it is assumed that there is no yawing or rolling rotation, the realistic size of pixels along the horizontal direction of the image is the same,

$$d_H = \frac{2h_c}{N_H} \tan(\frac{FOV_H}{2}), \tag{25}$$

where  $FOV_H$  is the field of view along the horizontal direction of the image,  $N_H$  is the total number of pixels along the horizontal direction.

The dimensions of each potential landing-site are estimated by measuring the major axis and minor axis of its best fit ellipse, which are obtained using the principle component analysis method. Once the major axis and the minor axis are found, the realistic length  $L$  and width  $W$  in the unit of feet can be gained as,

$$\begin{aligned} L &= \sqrt{(d_H(x_{a2} - x_{a1}))^2 + \left(\sum_{k=y_{a1}}^{y_{a2}} d_k\right)^2}, \\ W &= \sqrt{(d_H(x_{b2} - x_{b1}))^2 + \left(\sum_{k=y_{b1}}^{y_{b2}} d_k\right)^2}, \end{aligned} \quad (26)$$

where  $(x_{a1}, y_{a1})$ ,  $(x_{a2}, y_{a2})$  are coordinates of two end points of the major axis and  $(x_{b1}, y_{b1})$ ,  $(x_{b2}, y_{b2})$  are coordinates of two end points of the minor axis, in the image coordinate system. Figure 26 shows the length and width of each potential landing-site in the unit of ft. Small areas with insufficient length or width are ruled out, and only large areas with sufficient length and width can be considered safe emergency landing-sites.

#### 4.4 HUMAN-MACHINE INTERFACE

All the candidate landing-sites are highlighted on a human machine interface and labeled with preference indices that are sorted in a descending order based on the respective region areas, as shown in Figure 27. This allows the pilot to efficiently evaluate the recommended candidates in the rational order, since the time cost of making a decision is very critical. The pilot will make his/her final decision by choosing one emergency landing-site from the recommended candidates as well as concerning other factors, e.g. wind direction, wind speed, maneuvering ability. In general, larger areas are preferable compared with smaller ones.

#### 4.5 PERFORMANCE METRIC

To quantitatively evaluate the results generated by the proposed system, two veteran professional pilots were asked to manually pick all the possible landing-sites in the original images. Their judgment is mainly based on the apparent smoothness of the areas shown in the images. Next, the realistic dimensions of these manually selected areas are measured by using the same dimension assessment module mentioned above. This step is necessary because it is hard to accurately estimate the length and width of candidate landing-sites in the images captured at different heights by just looking at them. As shown in the left column of Figure 29, if the dimensions of a selected area meet the minimum requirement, it is labeled in “green” as a safe landing site. Otherwise, it is labeled in “red” as an unsafe landing site. After manual selection, all the selected regions are sorted in a descending order according to area. To fully evaluate the performance of the proposed CAD system, those manually selected and labeled regions are utilized as the ground-truth candidate regions. The complete detection results produced by the proposed CAD system are compared to the ground truth.

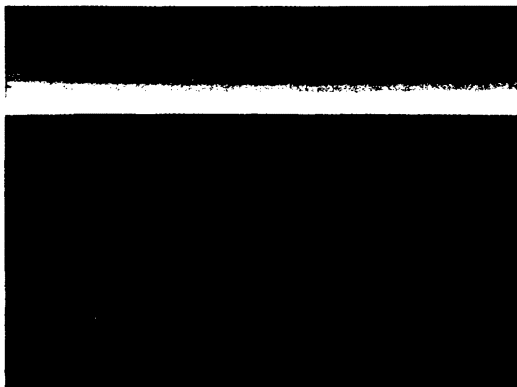


FIG. 22: A sample image



FIG. 23: Edges found with Canny detector



FIG. 24: Clustering result based on CHS

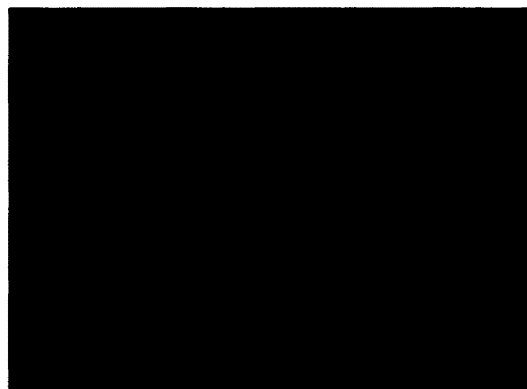


FIG. 25: Multi-region growing result

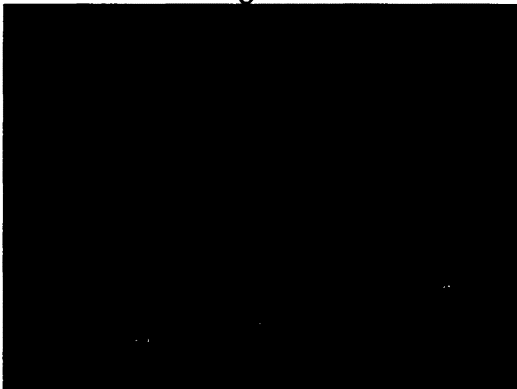


FIG. 26: Measurement of realistic dimensionality

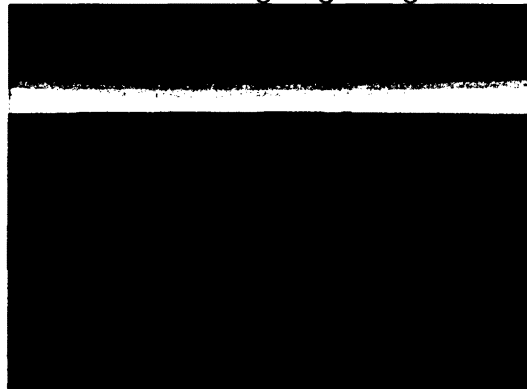


FIG. 27: Candidate sites for emergency landing

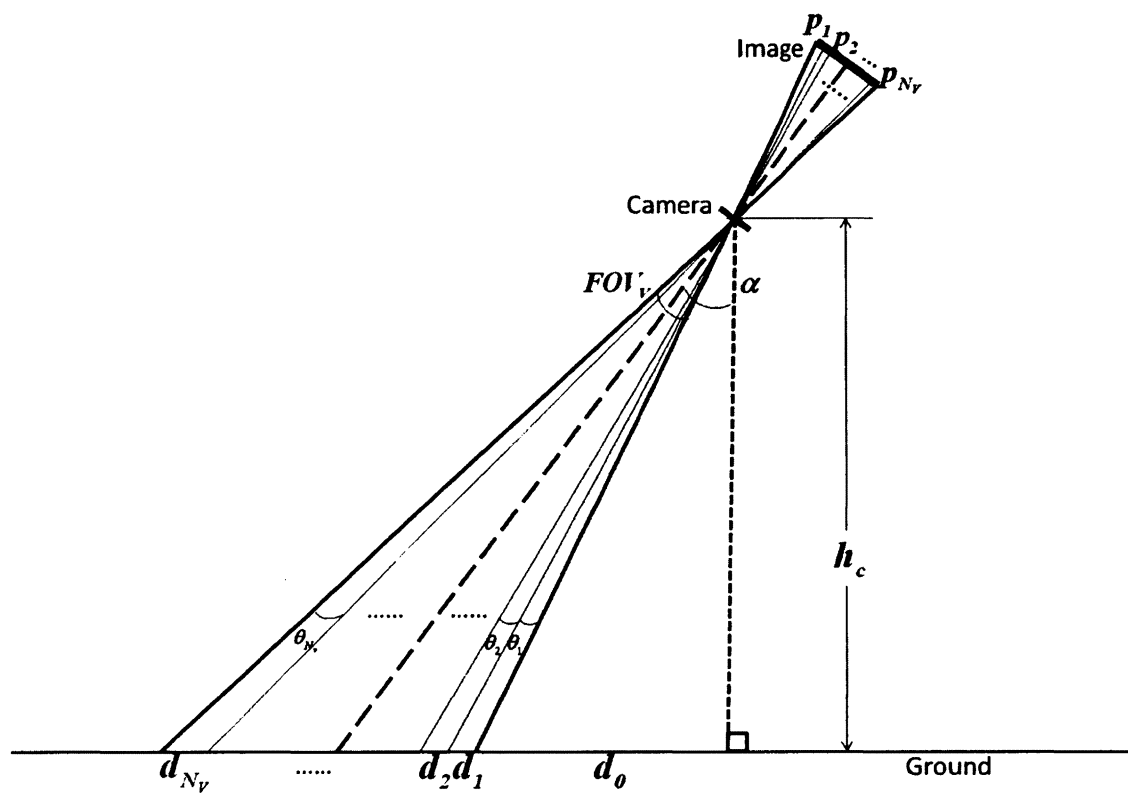


FIG. 28: Simplified imaging model

TABLE 1: Four possible results of the detection of landing-sites

#	Categories	Ground Truth Results	CAD System Results
1	True Positive	The area is <b>safe</b> for emergency landing and labeled as a candidate.	Hazards are <b>not found</b> in the area AND its dimensions <b>equal or exceed</b> the requirement.
2	False Positive	The area is <b>not safe</b> for landing.	Hazards are <b>not found</b> in that area AND its dimensions <b>equal or exceed</b> the requirement.
3	True Negative	The area is <b>not safe</b> for landing.	Hazards are <b>found</b> in that area OR its dimensions are <b>below</b> the requirement.
4	False Negative	The area is <b>safe</b> for emergency landing and labeled as a candidate.	Hazards are <b>found</b> in that area OR its dimensions are <b>below</b> the requirement.

If an area is found as a candidate landing site by the proposed CAD system, the detection result of this area is considered positive. Furthermore, if this detection result is consistent with the ground-truth, it is called a true positive (TP) detection. Otherwise, it is a false positive (FP) detection. Similarly, if an area is not selected as a candidate landing site by the proposed CAD system, the detection result of this area is considered negative. In addition, if this detection result is consistent with the ground-truth, it is called a true negative (TN) detection. Otherwise, it is a false negative (FN) detection. Table 1 lists the interpretations of the four exclusive and exhaustive situations. TP and TN are desired correct diagnoses. FP and FN are wrong diagnoses and have to be eliminated. It is worth noting that an FP is the worst situation, since it can mislead the aircraft to a dangerous place.

A scoring mechanism is proposed to quantitatively evaluate the performance of the proposed CAD system,

$$S = S_0 + S_B - S_P, \quad (27)$$

$$S_0 = \sum_{j=1}^N p_j, \quad S_B = \sum_{i=1}^M b_i T_i, \quad S_P = \sum_{j=1}^N p_j F_j.$$

For each test image, the score  $S$  consists of three parts: bonuses  $S_B$  for TP detections, penalties  $S_P$  for FP detections, and the base score  $S_0$ . In the ground-truth of each test image, there are  $M$  manually selected areas. According to the priority index, different bonus weights  $b_i$  ( $i = 1, \dots, M$ ) are given to these  $M$  areas in a descending order of size. If the CAD system successfully detects the  $i$ th largest safe landing-site in the ground truth, then bonus  $b_i$  is earned.  $T_i$  is the flag that indicates if the  $i$ th largest area in the ground-truth was successfully detected by the system. In addition to the reward mechanism, a punishment mechanism is also used. The complete detection results produced by the system before the visualization module contains  $N$  candidates, where  $N$  may be greater than five and different from

$M$ . Penalty  $p_j$  ( $j = 1, \dots, N$ ) is imposed if the  $j$ th recommended candidate landing site is an FP detection.  $F_j$  is the flag indicating if the  $j$ th detected area is an FP detection. The initiative of using bonus and penalties with different weights is to emphasize the priority of each safe landing-site in the ground truth. In general, the larger the dimensions of the landing-site are, the easier and safer the forced landing process is. Therefore, it is reasonable to give higher bonuses for true positive detection of larger safe landing-sites. Also, since a FP detection result labeled as a higher priority has more negative effect than a FP detection result labeled as a lower priority, it is reasonable to impose higher penalties to the former. In this dissertation,  $b_i$  is set, respectively, to be 30, 25, 20, 15, 10 for  $1 \leq i \leq 5$  and 5 for  $i \geq 6$ . In the same way,  $p_j$  is set, respectively, to be 30, 25, 20, 15, 10 for  $1 \leq j \leq 5$  and 5 for  $j \geq 6$ . The base score  $S_0$  is the potential maximum of penalties that a set of detection results can get. It is used to guarantee that  $S$  is non-negative, even if in the worst case all of the  $N$  detection results are FP,  $S = 0$ , the lowest score that the CAD system can get. Since  $N$  can significantly vary among images, the scores need to be normalized to a unified range. The normalized score is computed as follows,

$$\hat{S} = 100 \left( \frac{S}{S_{FM}} \right), \quad S_{FM} = S_0 + \sum_{i=1}^M b_i, \quad (28)$$

where  $\hat{S}$  is the normalized score that is obtained by normalizing  $S$  using  $S_{FM}$ .  $S_{FM}$  is the possible full score for each experiment when all detection results are consistent with the ground-truth. After normalization, scores range from 0 to 100: a score of 100 shows a perfect match while lower scores show decreasing matches between the

ground-truth and the CAD results. Equation (28) can be also interpreted as follows,

$$\hat{S} = 100 \left( \frac{S' - S_{min}}{S_{max} - S_{min}} \right) = 100 \left( \frac{S' + S_0}{S_{FM}} \right), \quad (29)$$

$$S_{min} = - \sum_{j=1}^N p_j, \quad S_{max} = \sum_{i=1}^M b_i, \quad S' = S_B - S_P, \quad S_{FM} = S_{max} - S_{min},$$

where  $S'$  is the actual performance of the  $N$  detection results,  $S_{min}$  and  $S_{max}$  are, respectively, the lower and upper bound of  $S'$ .  $S' - S_{min}$  is equivalent to  $S' + S_0$  shown in Equation (27), both of which transfer  $S'$  from the interval  $[S_{min}, S_{max}]$  to the interval  $[0, S_{FM}]$  so that  $S$  is guaranteed to be non-negative. Then,  $\hat{S}$  is obtained by normalizing  $S$  from the interval  $[0, S_{FM}]$  to the unified interval  $[0, 100]$ .

Equation (28) fails under two scenarios. (1) For images captured above rough terrains, there may be no safe landing-sites in the ground-truth ( $M = 0, S_{max} = 0$ ), so the best corresponding detection results should be no recommendations ( $N = 0, S_B = 0, S_0 = 0, S_P = 0$ ). For this special case, Equation (28) is ill-defined, since its denominator is 0; (2) When all the detection results are FP ( $S_B = 0, S_0 = S_P$ ), Equation (28) fails to differentiate results with different numbers of FPs, since the numerators of those situations are all zero. For example, suppose there is one safe landing-site in the ground truth ( $M = 1, S_{max} = 30$ ), the detection result of zero TP and zero FP ( $S_B = 0, S_0 = 0, S_P = 0$ ) should be better than the result of zero TP and two FPs ( $S_B = 0, S_0 = 55, S_P = 55$ ). However, based on Equation (28),  $\hat{S}$  of both above is 0, which means it fails to differentiate these two scenarios.

To solve these two problems, a correction is made to Equation (28) by adding a small augment  $\alpha$  to the numerator and denominator,

$$\hat{S} = 100 \left( \frac{S + \alpha}{S_{FM} + \alpha} \right), \quad S_{FM} = S_0 + \sum_{i=1}^M b_i. \quad (30)$$

$\alpha$  should be a relatively small number so that it has little effect on the ratio of the non-zero numerator and denominator when there is at least one TP in the detection results ( $S_B > 0$ ). In this dissertation,  $\alpha$  is 1. By using Equation (30), both problems are solved: in scenario (1),  $\hat{S}$  gets the expected score 100; In scenario (2) the normalized scores of the two scenarios are, respectively, 3.2 and 1.2. Thus the two scenarios get different scores, though both are relatively low, but at least “no recommendations” is better than false recommendations in this particular application.

## 4.6 EXPERIMENTAL RESULTS

The validation of the proposed system consists of three parts in this pilot study.

### Experiment 1

The reliability and accuracy of the proposed system was validated on independent static sample images with projection angles of  $0^\circ$  and  $60^\circ$ . A total of 169 images captured at 1,000–30,000 ft, and 100 images captured at 5,000 ft were used for testing the CAD algorithm. The left column of Figure 29 shows five samples of the labeled manual selection, and the right column shows the corresponding results produced by the proposed CAD system. Two performance metrics are applied to evaluate the detection results. (1) A group-wise TP detection rate (GTPR) is defined as,

$$GTPR = \frac{\sum_{i=1}^{N_{img}} DTP_i}{\sum_{i=1}^{N_{img}} MTP_i} \times 100\%, \quad (31)$$

where  $DTP_i$  is the number of TP detections in the  $i$ th image,  $MTP_i$  is the number of safe landing-sites in the  $i$ th image in the ground truth, and  $N_{img}$  is the total number of images in the testing set. The GTPRs of the 1,000–30,000ft testing set and the 5,000ft testing set are, respectively, 81.2% and 87.1%. (2) By using the performance metric presented in Section 4.5, normalized scores of

detection results of the two testing sets are obtained, and the distribution of the normalized scores is shown in Table 2. 69.2% of the experimental results of the first set and 74.0% of the experimental results of the second set completely match the ground truth ( $\hat{S} = 100$ ), and 80.6% of the first set and 86.0% of the second set generally match the ground truth ( $\hat{S} \geq 60$ ), which demonstrates the feasibility of the proposed CAD system. It is noticed that there were 18.3% failures in the first set and 14.0% in the second set ( $\hat{S} < 5$ ). Two major causes are found. (a) At higher elevations, fewer details are seen since the realistic area covered by each pixel in the image becomes larger when camera height increases. As a result, some areas appear to be smooth in the image, while in reality they are not. This explains why the results of the 5,000ft set is better than that of the 1,000–30,000ft set. This image resolution problem is, of course, a characteristic of the image-capture device. (b) Artificial or pseudo boundaries may cause the system to miss safe landing-sites. For example, an area may be flat but changes in soil color or soil texture appear as sharp edges that can confuse the CAD system. This can be attributed to the assumption that elevation changes can be mapped by edges in visible images. In the absence of elevation information (most small general aviation aircraft do not have an elevation database on board), imagery captured by aircraft cameras is the only source for computer algorithms to evaluate the surface roughness. The proposed CAD system can play an important role in this situation.

TABLE 2: Distribution of normalized scores of experimental results

Normalized Score	0	1	2	3	4	5-45	50	55	60	65	70	75	80	85	90	95	100
Percentage(%), 1k-30k ft	0	4.1	8.9	0	5.3	0	1.2	0	3.0	0	0.6	0.6	1.8	3.6	0.6	1.2	69.2
Percentage(%), 5k ft	0	0	9.0	0	5.0	0	0	0	4.0	0	1.0	2.0	5.0	0	0	0	74.0

## Experiment 2

The consistency of detection results is important, because the proposed system eventually will be utilized as a real time system to detect safe landing-sites as the aircraft moves. This is done by applying the proposed system to 10 sequences of images, each of which has 10 to 15 images. The motion of the aircraft is reflected by the relative position of the sequential images. For example, if the aircraft moves along the direction parallel to the ground, the corresponding sequence of images covers a band along the trace of the aircraft. The distance between the centers of every two consecutive images is determined by the flying speed. In other words, two consecutive images captured by the aircraft with lower speed have more overlap than those captured by one with higher speed. One sample sequence is shown in Figure 30(a), in which the distance between two consecutive images is about 2.5 seconds of latitude and 0 seconds of longitude, that is, approximately 253 feet or 77 meters. In addition, 10 sequences of images captured over a same spot but at different heights in the direction perpendicular to the ground are tested. Figure 30(b) shows a sequence of images along the vertical direction. A pair-wise consistency rate (CR) of detection results between two adjacent images and is defined as,

$$CR = \frac{CN}{CM} \times 100\%, \quad (32)$$

where  $CM$  is the number of common safe landing-sites in the ground truth between two adjacent images and  $CN$  is the number of common TP detection results between two adjacent images. The average consistency rate (ACR) is defined as,

$$ACR = \frac{1}{F-1} \sum_{f=1}^{F-1} CR_f, \quad (33)$$

where  $f$  is the index of images,  $F$  is the total number of images in one sequence, and  $CR_f$  is the consistency rate between the  $f$ th image and the  $(f+1)$ th image. The average ACR of tested 20 sequences is 84.1%.

#### 4.7 DISCUSSION

A vision-based real time information source is indispensable in the application of seeking safe emergency landing-sites. Although there are existing advanced equipments, that contain databases to indicate the locations of plain areas which are suitable for emergency landing, the system should possess the capability to identify transitory hazards or moving objects on a real time basis. The relatively low update frequency of the database mentioned above often cannot satisfy this expectation. Ideally, the proposed system can be combined with such a database to work together. For example, if the database can provide the location of a potential safe area and this area is within the reachable radius of the aircraft, the pilot can first follow the direction prompted by the database. As a result, the aircraft is expected to go in a generally correct direction. Then, by using the proposed system, safe landing-sites can be found after ruling out hazardous sites in that generally safe area. This hierarchical methodology is also applicable for exploratory landing on the moon, Mars, or other planets, to eliminate the damage to the spacecraft during the landing procedure.

Image quality is directly related to the reliability of the detection results. Given that the camera is at the same height, few details can be seen in a low resolution image, because each pixel of the image covers a large area on the ground. Similarly, for a given camera, fewer details can be seen from the higher elevations, so some areas appear to be smooth in the image, while in reality they are not. The resolution is, of course, a characteristic of the image-capture device, and a high quality imaging

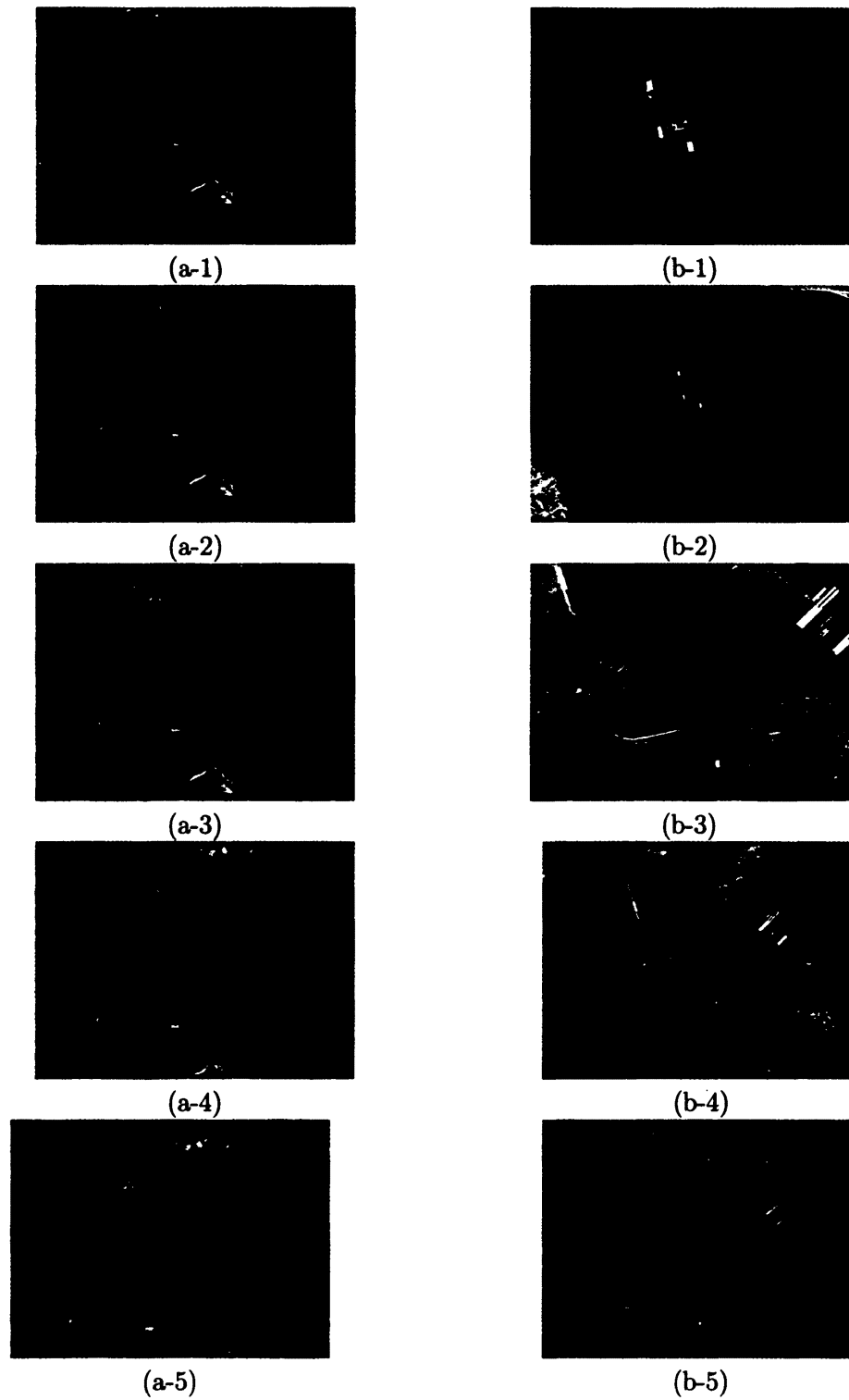


FIG. 30: Detection results of sequential images. (a) Aircraft moves along the direction parallel to the ground. (b) Images are captured over the same spot at different elevations.

device is always desirable.

In this study, the candidate landing-sites recommended to the pilot are sorted in a descending order only according to their areas. The factor of maneuverability has not been taken into account. The assumption used in this study is that a larger and wider landing-site can provide relatively more room for the emergency landing, which is indeed a positive factor for the emergency landing process. However, in reality the access to the largest candidate landing-site may not be safe and simple for the unpowered aircraft in terms of maneuverability. Therefore, if a method can be developed to estimate the degree of safeness, reliability and difficulty of landing at each candidate site by evaluating the recommended candidate landing-sites with other factors which are not reflected in the image, i.e., the controllability of the aircraft, wind direction, wind speed and so on, the automatic detection system can provide a comprehensive index of the priority of each candidate landing-site, which will be an advantage to the proposed system, although that is out of the scope of this dissertation. Currently this task is left to the pilot who will make the final decision by evaluating the recommended candidate landing-sites in an all-inclusive manner.

## CHAPTER 5

### TERRAIN ANALYSIS

Conventionally the task of terrain analysis is achieved by using multi-spectrum or hyper-spectrum remote sensing images [52–57]. In this chapter, an attempt is made to distinguish different types of terrains by using artificial neural networks. Selected features are first described for terrain analysis, and then the output weight optimization back propagation output reset (OWO-BP-OR) neural network [58–60] is introduced. Using this neural network as a unit, a cascaded neural network is designed. Experimental results are presented in Section 5.4.

#### 5.1 TERRAIN FEATURES

Each terrain patch to be classified,  $P$ , contains  $N_r$  rows,  $N_c$  columns, and  $N_b$  spectral bands ( $N_r = 100$ ,  $N_c = 100$ ,  $N_b = 3$  in this study). The feature vector  $V$  extracted from each terrain patch consists of (1) the standard deviation (std) of each band ( $V_1$  to  $V_{N_b}$ ), (2) the ratio between means of bands ( $V_{N_b+1}$  to  $V_{2N_b}$ ), (3) normalized energy in  $N_{ring}$  rings of the frequency spectrum ( $V_{2N_b+1}$  to  $V_{(2+N_{ring})N_b}$ ), (4) the ratio of (3) between channels ( $V_{(2+N_{ring})N_b+1}$  to  $V_{2(1+N_{ring})N_b}$ ). Given  $N_b = 3$  and  $N_{ring} = 5$ , the 36 elements of feature vector  $V$  can be written as follows.

- (1) The standard deviation (std) of each band ( $V_1$  to  $V_3$ )

$$V_i = \frac{1}{N_r N_c} \sum_{x_1=1}^{N_r} \sum_{x_2=1}^{N_c} (P_i(x_1, x_2) - M_i)^2, i = 1, 2, 3, \quad (34)$$

where  $P_i$  is the  $i$ th band of  $P$ , and  $M_i$  is the mean of  $P_i$  which is defined as,

$$M_i = \frac{1}{N_r N_c} \sum_{x_1=1}^{N_r} \sum_{x_2=1}^{N_c} P_i(x_1, x_2), i = 1, 2, 3. \quad (35)$$

(2) The ratio between means of bands ( $V_4$  to  $V_6$ )

$$V_4 = \frac{M_1}{M_2}, \quad (36)$$

$$V_5 = \frac{M_2}{M_3}, \quad (37)$$

$$V_6 = \frac{M_3}{M_1}. \quad (38)$$

(3) The normalized energy in rings of the frequency spectrum ( $V_7$  to  $V_{21}$ )

$$V_{6+5(i-1)+j} = \frac{\sum_{\nu_1=1}^{N_r} \sum_{\nu_2=1}^{N_c} |F_i(\nu_1, \nu_2)| R(\nu_1, \nu_2)}{\sum_{\nu_1=1}^{N_r} \sum_{\nu_2=1}^{N_c} |F_i(\nu_1, \nu_2)|}, i = 1, 2, 3, j = 1, 2, \dots, 5, \quad (39)$$

$$F_i(\nu_1, \nu_2) = \frac{1}{N_r N_c} \sum_{x_1=1}^{N_r} \sum_{x_2=1}^{N_c} P_i(x_1, x_2) \exp \left( -j2\pi \left( \frac{\nu_1 x_1}{N_r} + \frac{\nu_2 x_2}{N_c} \right) \right), \quad (40)$$

$$R(\nu_1, \nu_2) = \begin{cases} 1 & r_{j-1} \leq \sqrt{\left(\nu_1 - \frac{N_r}{2}\right)^2 + \left(\nu_2 - \frac{N_c}{2}\right)^2} < r_j, \\ 0 & \text{otherwise,} \end{cases} \quad (41)$$

$$r_0 = 0, r_j = \frac{j}{10} \min(N_r, N_c), \quad (42)$$

where  $F_i$  is the two-dimensional discrete Fourier transform (DFT) of  $P_i$ ,  $R$  is the mask of the  $j$ th ring of  $F_i$ ,  $r_j$  is the outer radius of the  $j$ th ring. In Figure 31, (a) shows a sample terrain patch, and (b) to (d) show the frequency spectrums of RGB bands of (a), labeled with normalized energy contained in each ring.

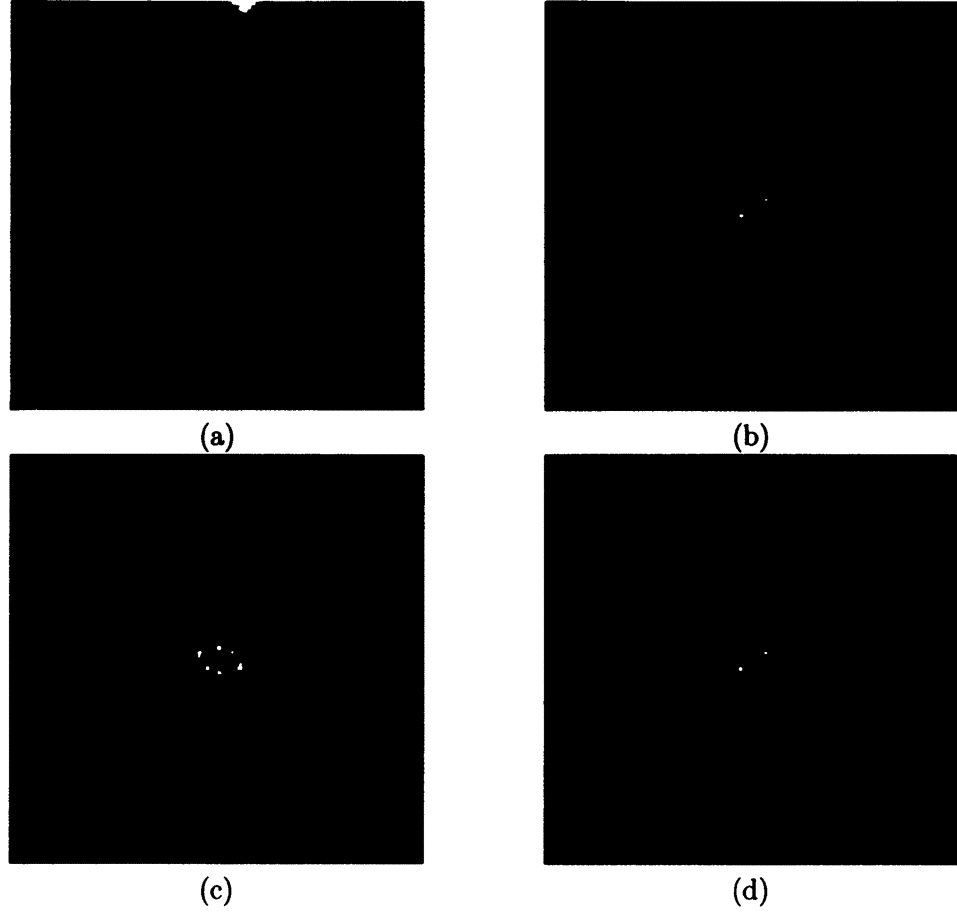


FIG. 31: (a) A sample terrain patch, (b)-(d) frequency spectrums of RGB bands of (a) labeled with normalized energy contained in each ring.

(4) The ratio of normalized energy between bands ( $V_{22}$  to  $V_{36}$ )

$$V_{21+5(i-1)+j} = \frac{V_{6+5\text{mod}(i-1, N_{ch})+j}}{V_{6+5\text{mod}(i, N_{ch})+j}}, i = 1, 2, 3, j = 1, 2, \dots, 5, \quad (43)$$

$$\text{mod}(a, b) = a - \left\lfloor \frac{a}{b} \right\rfloor \times b. \quad (44)$$

## 5.2 OWO-BP-OR NEURAL NETWORK

A multi-layer perceptron (MLP) is employed as the classifier. The output weight

optimization back propagation output reset (OWO-BP-OR) algorithm [58, 59] is implemented. The classifier is realized as a three-layer neural network with hidden units as shown in Figure 32, which contains an input layer, a hidden layer, and an output layer. The input layer contained  $N_{in}$  input units and a bias term. Each input unit corresponded to a feature item in the feature vector. The output layer contained  $N_{out}$  output units.

It is worth noting that the range of each element in the feature vector  $V$  can be significantly different. If  $V$  is directly utilized as the training input, some elements of  $V$  with relatively larger standard deviations will dominate the training, even if they might be actually relatively useless. To avoid this problem, the mean and the standard deviation of each element of  $V$  is first computed, so that the normalized feature vector  $\bar{V}$  is obtained as,

$$\bar{V}_m = \frac{V_m - \mu_m}{\sigma_m}, \quad (45)$$

where  $\mu_m$  and  $\sigma_m$  ( $m = 1, 2, \dots, N_{in}$ ) are, respectively, mean and standard deviation of  $V_m$ . The augmented input vector  $\bar{V}_a$  is defined as

$$\bar{V}_a = \begin{pmatrix} \bar{V} \\ 1 \end{pmatrix}. \quad (46)$$

The total number of vectors in the training set is  $N_v$ . For simplicity,  $s$  ( $s = 1, 2, \dots, N_v$ ) is used to index all the vectors in the training set.  $Y_s$  is defined as the corresponding classification output of the  $s$ th augmented input vector  $\bar{V}_{a,s}$ .  $T_{r,s}$  is defined as the classification ground truth of  $\bar{V}_{a,s}$ . All the input feature vectors  $\bar{V}_{a,s}$  ( $s = 1, 2, \dots, N_v$ ) form the input matrix  $X_a$ . Similarly, all the output vectors  $Y_s$  form the output matrix  $Y$ , and all the ground truth vectors  $T_{r,s}$  form the ground truth matrix  $T_r$ .

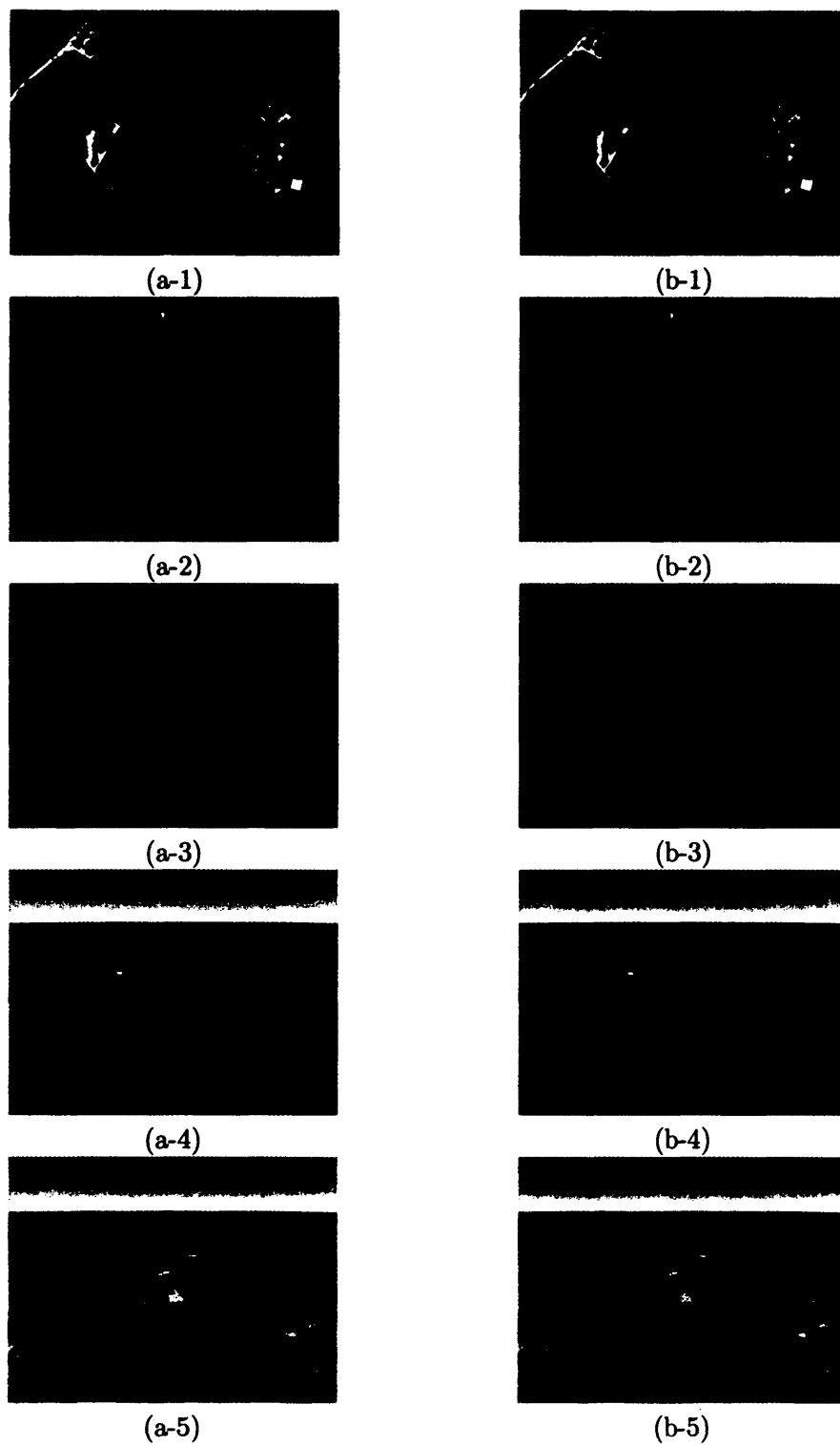


FIG. 29: Comparison between manual selection and automatic detection. (a): manually selected landing-sites; (b): recommended landing-sites detected by the proposed system

The goal of the training process is to minimize the error function  $E$ , which is defined as

$$E = \frac{1}{N_v} \sum_{s=1}^{N_v} \sum_{k=1}^{N_{out}} (T_{r,s}(k) - Y_s(k))^2, \quad (47)$$

where  $k$  denotes the index of classes. The relationship between the ground truth vector  $T_{r,s}$  and the ground truth class label  $f_s$  of vector  $\bar{V}_{a,s}$  can be written as

$$\begin{cases} T_{r,s}(k) = 1, & k = f_s; \\ T_{r,s}(k) = -1, & k \neq f_s. \end{cases} \quad (48)$$

For example, if  $N_{out}$  is 3, when  $f_s = 1$ ,  $T_{r,s} = (1 \ -1 \ -1)^T$ ; when  $f_s = 2$ ,  $T_{r,s} = (-1 \ 1 \ -1)^T$ ; and when  $f_s = 3$ ,  $T_{r,s} = (-1 \ -1 \ 1)^T$ .

Before the training,  $w_2$ , the weight matrix between the input layer to the hidden layer, was initialized with random numbers that follow a Gaussian distribution with a 0 mean and a standard deviation of 1. The dimensionality of  $w_2$  is  $N_h \times (N_{in} + 1)$ .  $w_3$ , the weight matrix between the input layer and the output layer, and  $w$ , the weight matrix between the hidden layer and the output layer, do not need to be initialized, since they will be solved by the OWO algorithm [58], [59] as follows,

$$w_c = (H_a^T H_a)^{-1} H_a^T T_r, \quad (49)$$

where  $H_a$  and  $w_c$  are respectively defined as

$$H_a = \begin{pmatrix} H \\ X_a \end{pmatrix}, w_c = (w \ w_3), \quad (50)$$

where  $H$  is constitutive of  $N_v$  hidden vectors  $H_s$  ( $s = 1, 2, \dots, N_v$ ). The dimensions of  $w_c$ ,  $w$ , and  $w_3$  are  $N_{out} \times (N_h + N_{in} + 1)$ ,  $N_{out} \times N_h$ , and  $N_{out} \times (N_{in} + 1)$ , respectively.

The output matrix  $Y$  is computed as

$$Y = w_c H_a. \quad (51)$$

The OWO algorithm [58], [59] solves for output weights  $w_c$ , or  $w$  and  $w_3$ , while the BP algorithm updates the hidden weights  $w_2$ , detailed as

$$w_2(j, i) = w_2(j, i) + Z \left( -\frac{\partial E}{\partial w_2(i, j)} \right), \quad (52)$$

where  $Z$  is a learning factor that will continuously update during the training process.

One possible problem of the cost function in (47) is that when a feature vector is already correctly classified, the cost function still may be increased. To solve this problem, the OR algorithm [58] is used. It improves the cost function as follows,

$$E = \frac{1}{N_v} \sum_{s=1}^{N_v} \sum_{k=1}^{N_{out}} (T_{rs}(k) + a_s + d_s(k) - Y_s(k))^2, \quad (53)$$

where  $a_s$  and  $d_s(k)$  are first initialized to zero and then updated during the training.

### 5.3 CASCADED CLASSIFIER

The OWO-BP-OR neural network described above ideally should be able to generate distinguishable prediction results, i.e. the output element according to the desired class is 1 and the rest output elements are  $-1$ . However, in practice, there could be more than one output elements having positive values, and sometime they are very close to each other so that the output element corresponding to the desired class might not have the highest value. If the decision is only based on the highest value, the desired classification result could be missed, though it might be just slightly lower than the highest value. Alternatively, if the least possible classes can first be excluded and the most possible classes retained, the classification process then can

be achieved within a narrowed range. This refined procedure can be carried out in an iterative fashion. As a result, a more reliable prediction result can be expected to be obtained.

Based on this idea, a cascaded classifier is designed by making use of the OWO-BP-OR neural network as its classification unit on each level. Instead of making a hard classification decision in one step by selecting the highest output element in the predicted result generated by a neural network, the cascaded classifier brings in a soft decision strategy. The classification process achieved by the cascaded classifier, which consists of multiple cascaded levels, becomes a refining procedure, in which a certain number of least possible classes are excluded on each level, as the number of candidate classes decreases. The desired classification result is gained on the final level. The structure of the proposed cascaded classifier is shown in Figure 33. On level one, an  $N_T$ -class classifier produces the output vector with  $N_T$  elements for each input feature vector. The least possible class can be found according to the lowest value of the elements and then the same feature vector is fed to a  $(N_T - 1)$ -class classifier on level two, which is trained to differentiate  $N_T - 1$  classes except the one that is just excluded. There are  $N_T$  classifiers on level two. Similarly, according to the output generated on level two, another least possible class gets excluded, and the same input feature vector is fed to a  $N_T - 2$  classifier on level three. This refining procedure is carried out until it reaches the final level. It is worth noting that there are  $N_T - 1$  levels in total and the number of classifiers on level  $k$  is determined by the number of combinations of  $N_T - k + 1$  classes out of  $N_T$  classes, which can be computed as

$$\binom{N_T}{N_T - k + 1} = \frac{N_T!}{(k - 1)!(N_T - k + 1)!}. \quad (54)$$

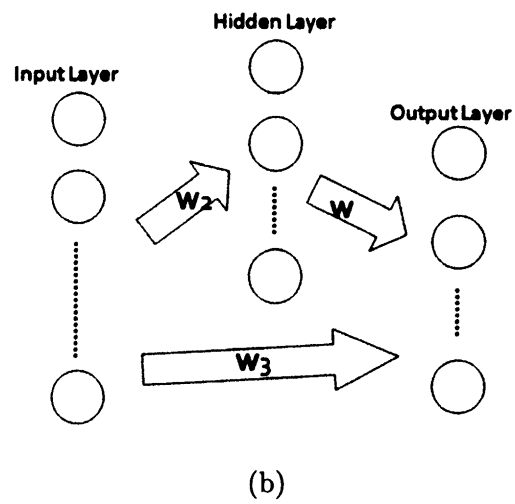
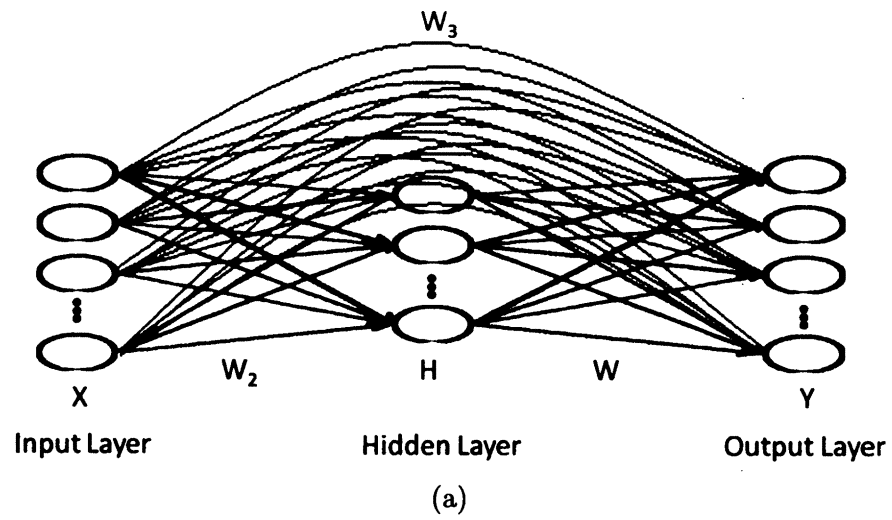


FIG. 32: (a) Multi-layer neural network representation; (b) Simplified representation

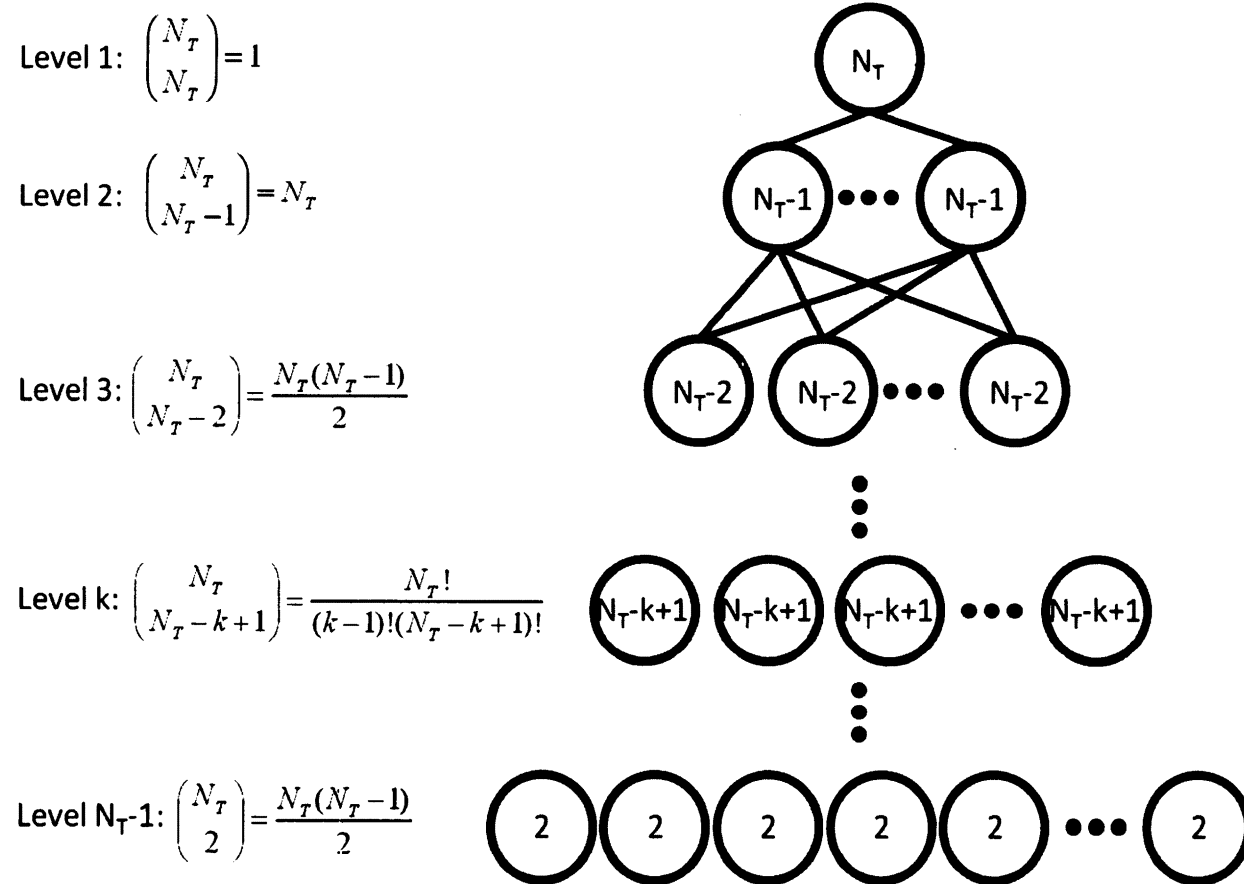


FIG. 33: Structure of the cascaded classifier

## 5.4 EXPERIMENTAL RESULTS

1,005 patches are manually extracted from Google Earth<sup>®</sup> covering 7 terrain classes as shown in Figure 34: (1) fresh water; (2) salt water; (3) green grass/farm; (4) yellow grass/farm; (5) rock; (6) dessert; (7) pavement. Only one type of terrain appears in each patch. The data set is split into two parts. 503 patches are used for training and the other 502 patches are used for testing.

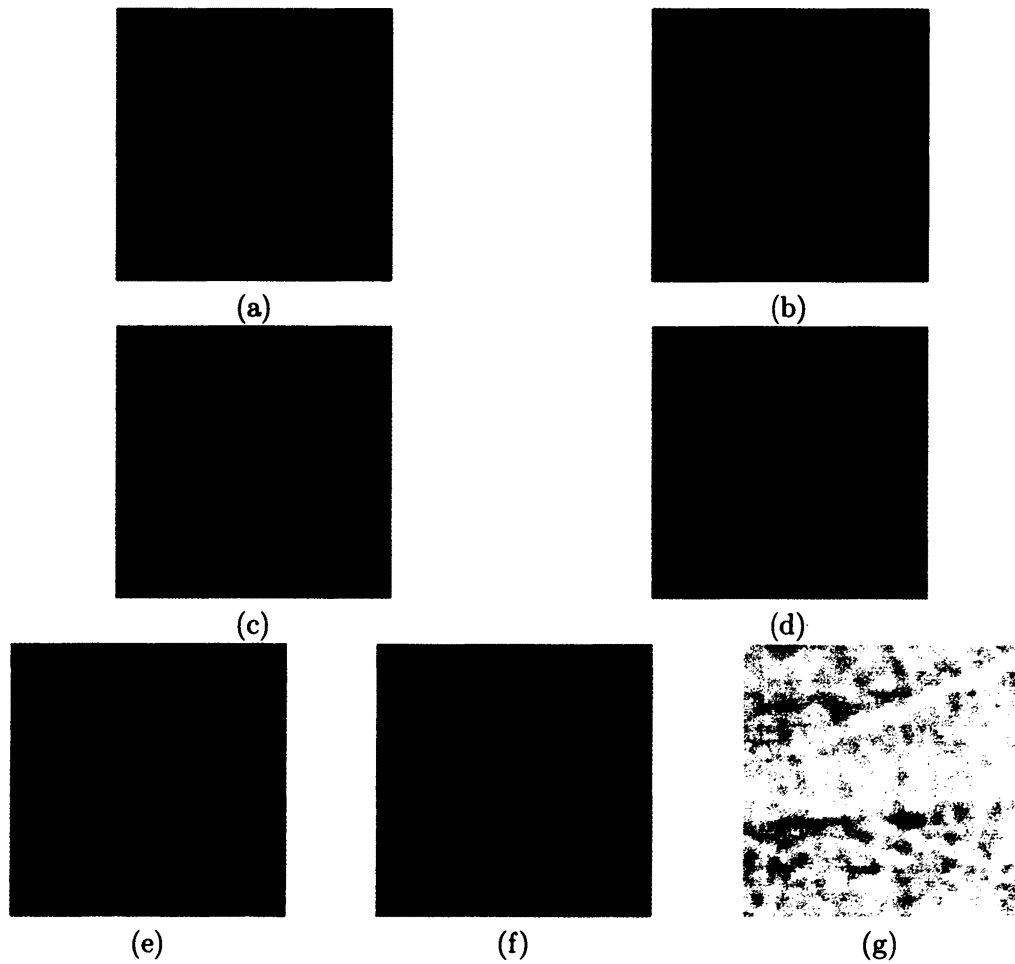


FIG. 34: (a) fresh water; (b) salt water; (c) green grass/farm; (d) yellow grass/farm; (e) rock; (f) dessert; (g) pavement.

### Experiment 1

The weights of the neural network are obtained by using the method described in Section 5.2 at the training phase. Then, the testing data set is tested by using the trained neural network. Empirically, the number of hidden units  $N_h$  is set to 15 in this study. The training accuracy is defined by the ratio of the number of correct classification results to the total number of training samples. The testing accuracy is defined by the ratio of the number of correct classification results to the total number of testing samples. For the 7-class classifier, its training accuracy is 62.4% and testing accuracy is 61.6%.

### Experiment 2

Each classification unit used to form the cascaded classifier is individually trained for each particular set of classes. In general, a classifier for fewer classes can generate better classification results, which is consistent with the idea mentioned above and the experimental results listed in Table 3 in Appendix A. It lists classifiers covering all the combinations of the 7 classes. For example, the training and testing accuracies of two-class classifiers reach over 95%. Because only two classes of patches get involved, the disturbance is relatively low. As the number of classes increases, the accuracy accordingly decreases. Therefore, a classifier with fewer classes may produce more reliable classification results than a classifier with more classes. The soft decision strategy is brought in by placing the 7-class classifier on the top and the 2-class classifier at the bottom. As shown in Experiment 1, the training and testing accuracies of the 7-class classifier are 62.4% and 61.6%. However, after the refining classification procedure, the overall training and testing accuracies are, respectively, improved by 14.6% and 14.1% to 71.5% and 70.3%.

## 5.5 DISCUSSION

The classification accuracy can be further improved by introducing new features, since most of the classification errors are found between the two classes of fresh water and green grass/farm. Since features are extracted from each terrain patch without any environmental context, they do not appear to be strong enough to successfully distinguish these two classes. In the future, other features will be applied to the classifier.

## CHAPTER 6

### CONCLUSIONS

A novel CAD prototype system for automatic safe landing-site detection is proposed in this dissertation, which is one of the few pilot studies in this area. By seamlessly combining a series of advanced image processing and analysis techniques and elaborately coordinating them to work together, the proposed system alleviates the limitations of human vision under poor illumination or weather conditions and primarily achieves the desired function, that is, to provide a robust, reliable, and efficient algorithm to automatically detect safe emergency landing-sites and to save time for the pilot under emergency situations to devote to other necessary actions. The promising experimental results show the feasibility of the design of the proposed system. It builds up a solid system framework which can be improved in the future.

A hierarchical elastic horizon detection algorithm is proposed to identify the ground in the aerial image so that the camera is relieved from the limitation of looking straight down to the ground. In the forward-looking mode, the system can detect safe landing-sites in front of the aircraft, which is significantly important for an unpowered aircraft during the gliding mode, because forward landing-sites provide more preparation time to the pilot to make right decisions and operations. Use of the forward-looking mode makes this dissertation distinguished from the few known studies in this area, in which use of the straight-looking mode is the convention.

A performance metric is developed to comprehensively and quantitatively evaluate the detection results, which fills the blank of a quantitative performance metric on this specific topic. In addition, some technical details are also delicately improved for the efficiency and reliability of the system, compared with the few known studies.

Several suggestions are listed as future work to make the system to better meet practical demands and applications. A denoising step might be needed to eliminate strong noise and artifacts, because the real world video data often has low signal to noise ratio and low resolution. A target tracking algorithm can be brought in to reduce the computational load of the video stream. The algorithm can be performed in a parallel fashion and implemented on a hardware platform. A shortwave near infrared sensor can be applied to obtain addition information. The proposed system has a wide range of potential applications, i.e. UAV autonomous landing and spacecraft landing, etc.

## BIBLIOGRAPHY

- [1] Y.-F. Shen and Z. Rahman, "An Automatic Computer-aided Detection System for Aircraft Emergency Landing," in *Proceedings of AIAA Infotech at Aerospace Conference, St. Louis, MO*, vol. 1, 2011, pp. 779–788.
- [2] Y.-F. Shen, Z. Rahman, D. Krusienski, and J. Li, "Automatic Detection of Aircraft Emergency Landing Sites," in *Proceedings of SPIE Defense, Security, and Sensing, Orlando, FL*, vol. 8056, 2011, pp. 8056–18.
- [3] Y. F. Shen, Z. Rahman, D. Krusienski, and J. Li, "A Vision-based Automatic Safe Landing-site Detection System," *IEEE Transactions on Aerospace and Electronic Systems*, submitted.
- [4] E. D. Dickmanns and F.-R. Schell, "Autonomous landing of airplanes by dynamic machine vision," in *IEEE Workshop on Applications of Computer Vision*, 1992, pp. 172–179.
- [5] A. Petruszka and A. Stentz, "Stereo vision automatic landing of VTOL UAVs," in *Proceedings of the 23rd Annual Association for Unmanned Vehicle Systems International Symposium and Exhibition (AUVSI '96), Orlando, FL, USA*, 1996, pp. 245–263.
- [6] I. Kaminer, A. Pascoal, and W. Kang, "Integrated vision/inertial navigation system design using nonlinear filtering," in *Proceedings of the American Control Conference*, vol. 3, 1999, pp. 1910–1914.
- [7] O. Shakernia, Y. Ma, T. J. Koo, J. Hespanha, and S. S. Sastry, "Vision guided landing of an unmanned air vehicle," in *Proceedings of the 38th IEEE Conference on Decision and Control*, vol. 4, 1999, pp. 4143–4148.

- [8] O. Shakernia, Y. Ma, T. J. Koo, and S. Sastry, "Landing an unmanned air vehicle: Vision based motion estimation and nonlinear control," *Asian Journal of Control*, vol. 1, no. 3, pp. 128–145, Sep 1999.
- [9] C. S. Sharp, O. Shakernia, and S. S. Sastry, "A vision system for landing an unmanned aerial vehicle," in *Proceedings of IEEE International Conference on Robotics and Automation, Seoul, Republic of Korea, 2001*, pp. 1720–1728.
- [10] S. Werner, S. Furst, D. Dickmanns, and E. D. Dickmanns, "A vision-based multi-sensor machine perception system for autonomous aircraft landing approach," in *Proceedings of the SPIE, the International Society for Optical Engineering, Orlando, FL, USA*, vol. 2736, 1996, pp. 54–63.
- [11] Z. F. Yang and W. H. Tsai, "Using parallel line information for vision-based landmark location estimation and an application to automatic helicopter landing," *Robotics and Computer-Integrated Manufacturing*, vol. 14, no. 4, pp. 297–306, 1998.
- [12] P. Garcia-Padro, G. Sukhatme, and J. Montgomery, "Towards vision-based safe landing for an autonomous helicopter," *Robotics and Autonomous Systems*, vol. 38, no. 1, pp. 19–29, Jan 2002.
- [13] D. L. Fitzgerald, R. A. Walker, and D. A. Campbell, "A Vision Based Forced Landing Site Selection System for an Autonomous UAV," in *Proceedings ISS-NIP, Melbourne, Australia, 2005*.
- [14] D. L. Fitzgerald, "Landing site selection for uav forced landings using machine vision," PhD thesis, Queensland University of Technology, 2007.
- [15] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

- [16] A. Johnson, A. Klumpp, J. Collier, and A. Wolf, "Lidar-based hazard avoidance for safe landing on Mars," in *Proceedings of the 11th Annual AAS/AIAA Space Flight Mechanics Meeting, Santa Barbara, CA*, 2001, pp. 323–337.
- [17] A. Howard and H. Seraji, "A fuzzy rule-based safety index for landing site risk assessment," in *Automation Congress, Proceedings of the 5th Biannual World*, 2002, pp. 579–584.
- [18] A. Howard, "A novel information fusion methodology for intelligent terrain analysis," in *Proceedings of IEEE International Conference on Fuzzy Systems*, 2002, pp. 1472–1475.
- [19] A. Howard and H. Seraji, "Multi-sensor terrain classification for safe spacecraft landing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 4, pp. 1122–1131, 2004.
- [20] S. R. Ploen, H. Seraji, and C. E. Kinney, "Determination of spacecraft landing footprint for safe planetary landing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 1, pp. 3–16, 2009.
- [21] S. Paschall, B. E. Cohanin, T. Brady, and R. Sostaric, "A self contained method for safe & precise lunar landing," in *IEEE Aerospace Conference, Big Sky, MT*, 2008, pp. 1–12.
- [22] J. O. Cappellari Jr, "Where on the moon? An Apollo systems engineering problem." *Bell System Technical Journal*, vol. 51, pp. 961–1126, 1972.
- [23] N. J. S. C. Solar System Exploration Division, "A site selection strategy for a lunar outpost: Science and operational parameters," *Conclusions of a Workshop*, Houston, USA, August 1990.

- [24] Z. Rahman, D. J. Jobson, and G. A. Woodell, "Retinex processing for automatic image enhancement," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 100–110, 2004. [Online]. Available: <http://link.aip.org/link/?JEI/13/100/1>
- [25] D. J. Jobson, Z. Rahman, and G. A. Woodell, "A multi-scale Retinex for bridging the gap between color images and the human observation of scenes," *IEEE Transactions on Image Processing: Special Issue on Color Processing*, vol. 6, no. 7, pp. 965–976, 1997.
- [26] A. Jain, *Fundamentals of Digital Image Processing*. Prentice-Hall, 1989.
- [27] D. Vernon, *Machine Vision*. Prentice-Hall, 1991.
- [28] E. Davies, *Machine Vision: Theory, Algorithms and Practicalities*. Academic Press, 1990.
- [29] Acharya and Ray, *Image Processing: Principles and Applications*. Wiley-Interscience, 2005.
- [30] Russ, *The Image Processing Handbook*, 4th ed. CRC, 2002.
- [31] C. A. Poynton, *Digital Video and HDTV: Algorithms and Interfaces*. Morgan Kaufmann, 2003.
- [32] C.-Y. Yu, Y.-C. Ouyang, Chuin-MuWang, and C.-I. Chang, "Adaptive Inverse Hyperbolic Tangent Algorithm for Dynamic Contrast Adjustment in Displaying Scenes," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, no. 485151, pp. 1–20, Feb 2010.
- [33] N. Pears and B. Liang, "Ground plane segmentation for mobile robot visual navigation," *IROS*, vol. 3, pp. 1513–1518, 2001.

- [34] Y. geun Kim and H. Kim, "Layered ground floor detection for visionbased mobile robot navigation," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, no. 26, 2004, pp. 13–18.
- [35] S. Ettinger, M. Nechyba, P. Ifju, and M. Waszak, "Towards flight autonomy: Vision-based horizon detection for micro air vehicles," in *Florida Conference on Recent Advances in Robotics*, 2002.
- [36] S. Williams and A. M. Howard, "Horizon line estimation in glacial environments using multiple visual cues," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 5887–5892.
- [37] D. Dusha, W. Boles, and R. Walker, "Attitude Estimation for a Fixed-Wing Aircraft Using Horizon Detection and Optical Flow," in *Digital Image Computing Techniques and Applications, 9th Biennial Conference of the Australian Pattern Recognition Society on Digital Object*, 2007, pp. 485–492.
- [38] R. O. Duda and P. E. Hart, "Use of the hought transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, January 1972.
- [39] Y.-F. Shen, D. Krusienski, J. Li, and Z. Rahman, "A Hierarchical Horizon Detection Algorithm," *IEEE Geoscience and Remote Sensing Letter*, submitted.
- [40] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes, "Nonrigid registration using free-form deformations: Application to breast mr images," *IEEE Transactions on Medical Imaging*, vol. 18, no. 8, pp. 712–721, Aug 1999.

- [41] S. Lee, G. Wolberg, and S. Y. Shin, "Scattered Data Interpolation with Multi-level B-Splines," *IEEE Transactions on Visualization and Computer Graphics*, vol. 3, pp. 228–244, 1997.
- [42] J. T. Tou and P. Gonzalez, Rafael C., *Pattern Recognition Principles*. Reading, MA: Addison-Wesley, 1974.
- [43] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York, NY: John Wiley and Sons, 1973.
- [44] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Upper Saddle River, New Jersey 07458: Pearson Prentice Hall, 2008.
- [45] W. K. Pratt, *Digital Image Processing*, 4th ed. Los Altos, California: John Wiley & Sons, Inc., 2007.
- [46] M. Petrou and P. Bosdogianni, *Image Processing the Fundamentals*. UK: Wiley, 2004.
- [47] Y.-F. Shen, J. Li, Y. Shen, and J. Wang, "MRI Brain Tumor Segmentation Using a Region Growing Method Based on Adaptive Thresholds," in *Proceedings of Modeling, Simulation and Gaming Capstone Conference*, April 2008, pp. 8–14.
- [48] P. Soille, *Image Processing the Fundamentals: Principles and Applications*. Springer-Verlag Telos, 1999.
- [49] J. Serra, *Image Analysis and Mathematical Morphology, Vol. 2: Theoretical Advances*. Academic Press, 1988.
- [50] H. Goldstein, *Classical Mechanics*, 2nd ed. Reading, MA: Addison-Wesley, 1980.
- [51] L. Landau and E. M. Lifshitz, *Mechanics*, 3rd ed. Oxford, 1996.

- [52] C.-I. Chang and S.-S. Chiang, "Anomaly Detection and Classification for Hyperspectral Imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 6, pp. 1314–1325, June 2002.
- [53] H. Seraji and N. Serrano, "A Multisensor Decision Fusion System for Terrain Safety Assessment," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 99–108, 2009.
- [54] L. Bruzzone, M. Marconcini, U. Wegmuller, and A. Wiesmann, "An advanced system for the automatic classification of multitemporal SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 6, pp. 1321–1334, 2004.
- [55] C.-I. Chang and H. Ren, "An experiment-based quantitative and comparative analysis of target detection and image classification algorithms for hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 38, no. 2, pp. 1044–1063, 2000.
- [56] J.-S. Lee, M. Grunes, E. Pottier, and L. Ferro-Famil, "Unsupervised terrain classification preserving polarimetric scattering characteristics," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 4, pp. 722–731, 2004.
- [57] J. S. Weszka, C. R. Dyer, and A. Rosenfeld, "A Comparative Study of Texture Measures for Terrain Classification," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 6, no. 4, pp. 269–285, 1976.
- [58] J. Li, M. T. Manry, L. M. Liu, C. Yu, and J. Wei, "Iterative improvement of neural classifiers," *Proceedings of the Seventeenth International Conference of the Florida AI Research Society*, pp. 700–705, May 2004.

- [59] M. T. Manry, X. Guan, S. J. Apollo, L. Allen, W. Lyle, and W. Gong, "Output weight optimization for the multi-layer perceptron," *IEEE Proc. the 26th Asilomar Conference*, vol. 1, pp. 502–506, Oct 1992.
- [60] Y.-F. Shen, "Brain Tumor Progression Assessment Using Multiple MRI Volumes," Master's Thesis, Old Dominion University, August 2009.

## APPENDIX A

### TRAINING AND TESTING RESULTS OF CLASSIFIERS FOR ALL POSSIBLE COMBINATIONS

TABLE 3: Training and Testing Results of Classifiers for All Possible Combinations

Classifier	No. Tra	R	W	Acc Tra	No. Tst	R	W	Acc Tst
1. 2.	138	128	10	92.8	138	127	11	92.0
1. 3.	201	187	14	93.0	201	185	16	92.0
1. 4.	174	169	5	97.1	172	166	6	96.5
1. 5.	89	89	0	100.0	88	87	1	98.9
1. 6.	96	96	0	100.0	95	95	0	100.0
1. 7.	120	112	8	93.3	118	112	6	94.9
2. 3.	213	213	0	100.0	215	215	0	100.0
2. 4.	186	186	0	100.0	186	186	0	100.0
2. 5.	101	101	0	100.0	102	102	0	100.0
2. 6.	108	108	0	100.0	109	109	0	100.0
2. 7.	132	132	0	100.0	132	131	1	99.2
3. 4.	249	239	10	96.0	249	239	10	96.0
3. 5.	164	163	1	99.4	165	163	2	98.8
3. 6.	171	171	0	100.0	172	172	0	100.0
3. 7.	195	192	3	98.5	195	194	1	99.5

*continued on next page*

*continued from previous page*

Classifier	No. Tra	R	W	Acc Tra	No. Tst	R	W	Acc Tst
4. 5.	137	132	5	96.4	136	130	6	95.6
4. 6.	144	144	0	100.0	143	143	0	100.0
4. 7.	168	167	1	99.4	166	165	1	99.4
5. 6.	59	59	0	100.0	59	55	4	93.2
5. 7.	83	83	0	100.0	82	81	1	98.8
6. 7.	90	90	0	100.0	89	89	0	100.0
1. 2. 3.	276	241	35	87.3	277	241	36	87.0
1. 2. 4.	249	223	26	89.6	248	223	25	89.9
1. 2. 5.	164	145	19	88.4	164	145	19	88.4
1. 2. 6.	171	158	13	92.4	171	155	16	90.6
1. 2. 7.	195	170	25	87.2	194	172	22	88.7
1. 3. 4.	312	258	54	82.7	311	257	54	82.6
1. 3. 5.	227	203	24	89.4	227	196	31	86.3
1. 3. 6.	234	209	25	89.3	234	206	28	88.0
1. 3. 7.	258	218	40	84.5	257	217	40	84.4
1. 4. 5.	200	175	25	87.5	198	171	27	86.4
1. 4. 6.	207	191	16	92.3	205	192	13	93.7
1. 4. 7.	231	208	23	90.0	228	212	16	93.0
1. 5. 6.	122	116	6	95.1	121	110	11	90.9
1. 5. 7.	146	137	9	93.8	144	133	11	92.4
1. 6. 7.	153	143	10	93.5	151	144	7	95.4
2. 3. 4.	324	271	53	83.6	325	279	46	85.8
2. 3. 5.	239	228	11	95.4	241	225	16	93.4

*continued on next page*

*continued from previous page*

Classifier	No. Tra	R	W	Acc Tra	No. Tst	R	W	Acc Tst
2. 3. 6.	246	241	5	98.0	248	240	8	96.8
2. 3. 7.	270	258	12	95.6	271	263	8	97.0
2. 4. 5.	212	186	26	87.7	212	187	25	88.2
2. 4. 6.	219	218	1	99.5	219	217	2	99.1
2. 4. 7.	243	229	14	94.2	242	225	17	93.0
2. 5. 6.	134	75	59	56.0	135	76	59	56.3
2. 5. 7.	158	155	3	98.1	158	153	5	96.8
2. 6. 7.	165	164	1	99.4	165	161	4	97.6
3. 4. 5.	275	251	24	91.3	275	250	25	90.9
3. 4. 6.	282	273	9	96.8	282	271	11	96.1
3. 4. 7.	306	278	28	90.8	305	284	21	93.1
3. 5. 6.	197	180	17	91.4	198	181	17	91.4
3. 5. 7.	221	212	9	95.9	221	214	7	96.8
3. 6. 7.	228	224	4	98.2	228	224	4	98.2
4. 5. 6.	170	150	20	88.2	169	146	23	86.4
4. 5. 7.	194	185	9	95.4	192	178	14	92.7
4. 6. 7.	201	198	3	98.5	199	196	3	98.5
5. 6. 7.	116	102	14	87.9	115	102	13	88.7
1. 2. 3. 4.	387	285	102	73.6	387	281	106	72.6
1. 2. 3. 5.	302	239	63	79.1	303	239	64	78.9
1. 2. 3. 6.	309	235	74	76.1	310	225	85	72.6
1. 2. 3. 7.	333	245	88	73.6	333	248	85	74.5
1. 2. 4. 5.	275	219	56	79.6	274	222	52	81.0

*continued on next page*

*continued from previous page*

Classifier	No. Tra	R	W	Acc Tra	No. Tst	R	W	Acc Tst
1. 2. 4. 6.	282	212	70	75.2	281	212	69	75.4
1. 2. 4. 7.	306	212	94	69.3	304	219	85	72.0
1. 2. 5. 6.	197	142	55	72.1	197	144	53	73.1
1. 2. 5. 7.	221	177	44	80.1	220	181	39	82.3
1. 2. 6. 7.	228	168	60	73.7	227	165	62	72.7
1. 3. 4. 5.	338	242	96	71.6	337	239	98	70.9
1. 3. 4. 6.	345	261	84	75.7	344	259	85	75.3
1. 3. 4. 7.	369	277	92	75.1	367	270	97	73.6
1. 3. 5. 6.	260	205	55	78.8	260	202	58	77.7
1. 3. 5. 7.	284	222	62	78.2	283	222	61	78.4
1. 3. 6. 7.	291	223	68	76.6	290	220	70	75.9
1. 4. 5. 6.	233	192	41	82.4	231	191	40	82.7
1. 4. 5. 7.	257	202	55	78.6	254	205	49	80.7
1. 4. 6. 7.	264	215	49	81.4	261	211	50	80.8
1. 5. 6. 7.	179	137	42	76.5	177	138	39	78.0
2. 3. 4. 5.	350	269	81	76.9	351	272	79	77.5
2. 3. 4. 6.	357	298	59	83.5	358	299	59	83.5
2. 3. 4. 7.	381	302	79	79.3	381	302	79	79.3
2. 3. 5. 6.	272	234	38	86.0	274	230	44	83.9
2. 3. 5. 7.	296	251	45	84.8	297	250	47	84.2
2. 3. 6. 7.	303	261	42	86.1	304	260	44	85.5
2. 4. 5. 6.	245	217	28	88.6	245	213	32	86.9
2. 4. 5. 7.	269	207	62	77.0	268	207	65	75.7

*continued on next page*

*continued from previous page*

Classifier	No. Tra	R	W	Acc Tra	No. Tst	R	W	Acc Tst
2. 4. 6. 7.	276	234	42	84.8	275	229	46	83.3
2. 5. 6. 7.	191	158	33	82.7	191	158	33	82.7
3. 4. 5. 6.	308	262	46	85.1	308	269	39	87.3
3. 4. 5. 7.	332	272	60	81.9	331	273	58	82.5
3. 4. 6. 7.	339	301	38	88.8	338	305	33	90.2
3. 5. 6. 7.	254	214	40	84.3	254	218	36	85.8
4. 5. 6. 7.	227	198	29	87.2	225	198	27	88.0
1. 2. 3. 4. 5.	413	289	124	70.0	413	282	131	68.3
1. 2. 3. 4. 6.	420	290	130	69.0	420	278	142	66.2
1. 2. 3. 4. 7.	444	297	147	66.9	443	296	147	66.8
1. 2. 3. 5. 6.	335	235	100	70.1	336	227	109	67.6
1. 2. 3. 5. 7.	359	261	98	72.7	359	253	106	70.5
1. 2. 3. 6. 7.	366	253	113	69.1	366	245	121	66.9
1. 2. 4. 5. 6.	308	219	89	71.1	307	218	89	71.0
1. 2. 4. 5. 7.	332	225	107	67.8	330	223	107	67.6
1. 2. 4. 6. 7.	339	233	106	68.7	337	235	102	69.7
1. 2. 5. 6. 7.	254	163	91	64.2	253	166	87	65.6
1. 3. 4. 5. 6.	371	264	107	71.2	370	261	109	70.5
1. 3. 4. 5. 7.	395	273	122	69.1	393	276	117	70.2
1. 3. 4. 6. 7.	402	296	106	73.6	400	297	103	74.3
1. 3. 5. 6. 7.	317	217	100	68.5	316	215	101	68.0
1. 4. 5. 6. 7.	290	228	62	78.6	287	221	66	77.0
2. 3. 4. 5. 6.	383	296	87	77.3	381	293	91	76.3

*continued on next page*

*continued from previous page*

Classifier	No. Tra	R	W	Acc Tra	No. Tst	R	W	Acc Tst
2. 3. 4. 5. 7.	407	299	108	73.5	407	303	104	74.4
2. 3. 4. 6. 7.	414	327	87	79.0	414	331	83	80.0
2. 3. 5. 6. 7.	329	258	71	78.4	330	271	59	82.1
2. 4. 5. 6. 7.	302	242	60	80.1	301	236	65	78.4
3. 4. 5. 6. 7.	365	292	73	80.0	364	306	58	84.1
1. 2. 3. 4. 5. 6.	446	288	158	64.6	446	279	167	62.6
1. 2. 3. 4. 5. 7.	470	306	164	65.1	469	299	170	63.8
1. 2. 3. 4. 6. 7.	477	318	159	66.7	476	310	166	65.1
1. 2. 3. 5. 6. 7.	392	250	142	63.8	392	244	148	62.2
1. 2. 4. 5. 6. 7.	365	244	121	66.8	363	242	121	66.7
1. 3. 4. 5. 6. 7.	428	286	142	66.8	426	281	145	66.0
2. 3. 4. 5. 6. 7.	440	322	118	73.2	440	328	112	74.5
1. 2. 3. 4. 5. 6. 7.	503	314	189	62.4	502	309	193	61.6

## VITA

Yufei Shen

Department of Electrical and Computer Engineering

Old Dominion University

Norfolk, VA 23529

Yufei Shen received his B.S. degree in electronic engineering from Fudan University, Shanghai, China, in 2006 and his M.S. and Ph.D. degrees in electrical and computer engineering from Old Dominion University, Norfolk, VA, in 2009 and 2012, respectively. His primary research focus is on the development of computer-aided detection/diagnosis systems. His research interests include digital signal/image processing and analysis, computer vision, machine learning, and pattern recognition.