

DOCTORAL THESIS

Automatic Defect Detection System for Automotive Painted Surface

Graduate School of Science and Engineering, Iwate University
Doctoral Course, Design & Media Technology
Jing Zhang

March 2022

Acknowledgement

I would like to express my deepest gratitude to associate Prof. Takuya Akashi for his excellent guidance, patience, endless support, and for providing me a comfortable atmosphere for doing my research. I would like to thank Prof. Kouichi Konno, Prof. Takamitsu Tanaka, who gave me many useful comments and advice since I started my research.

I would like to thank every member of the Smart Computer Vision Lab. of Iwate University for their help and patience. I would especially like to thank Dr. Chao Zhang for his help and advice. At least, I would like to thank my parents, who always support me in pursuing my dream without asking for any return. I can never forget that they gave all they have to support me to start my study in Japan.

Abstract

In recent years, with the development of industrial technology, competition in the automotive industry has become fiercer, and consumers have higher requirements for automobiles. The surface of the car is mostly coated, and the quality of the coating is the consumers' intuitive evaluation of the quality of the car. At the same time, the coating can improve the corrosion and oxidation resistance of automobiles, which is one of the main measures to prolong the life of automobiles. Therefore, the quality of coating can directly affect the market competitiveness of automobile manufacturers, and defect detection of coating surfaces is a very important work in the automobile manufacturing industry. In the process of car manufacturing, some minor errors, such as foreign objects mixed in the paint, dust in the air, etc., cause some bumps and dents on the surface of the car coating. These defects cannot be completely detected, and every year many cars are returned to the factory to repair, for this reason, causing a lot of damage to the automobile company. In the automotive industry, the detection of surface defects of automotive coatings is highly dependent on human vision and experience. The car company will arrange for some professional inspectors to inspect the coated car surface. However, the inspector's vision and attention will be affected by long-term work in this light environment. This manual defect detection method has many problems, such as

strong subjectivity, low stability, and low efficiency. Therefore, an automatic defect detection method for coated surfaces is required.

Automatic defect detection of automotive coated surfaces is usually based on the simulation of visual detection steps. There are three main steps: First, you must create an environment that is easy to detect defects. Then, collect the images. Finally, the defect is detected and identified by image processing. In the captured image, there is a lot of interference on the surface of the coating, which makes it difficult to detect. For example, during the drying process of the coating, uneven fine lines will be formed due to the drying speed and uniformity of the coating. These uneven, thin lines cause the surface drawn in the image to produce a point that looks like a cloud or orange peel. This is called the cloud or orange-peel effect and is not considered a defect. Because these noises reduce the accuracy of automatic defect detection, it is necessary to remove these noises. In this paper, the following studies have been done to remove the noise from the detection results. First, a method based on brightness difference to evaluate the difficulty of defect detection is presented. Using this method, the area most prone to orange peel effects in images were determined experimentally and statistically. Next, a method for automatically exploring the optimal threshold value for image binarization is presented to detect all possible candidates for defects. These candidates are filtered through the distribution area of orange peel effect and the brightness difference index to remove the noise. Finally, for the accuracy of detection results, the results are further filtered by continuous frames.

Many common automatic detection methods are based on a single image. Therefore, when noise is detected in the image, it is considered a defect, which is a prob-

lem encountered by common methods. In order to further screen for noise in defect candidates, some studies have proposed methods for merging continuous images to enhance defect performance, but these methods require the assistance of specific external equipment conditions. Other studies use the detection results of the previous frame to infer the results of the next frame and compare them with the actual results to determine whether the defect is true or false. However, this method requires not only a certain relative moving speed between the detected surface and the camera, but also an unstable effect on the curved surface. This study presents a new method based on continuous images. Specifically, many labels are attached to the test vehicle to simulate defects, and the label coordinates of each frame are captured and recorded. The results are then analyzed to obtain the calibration reference lines. The track of defect candidates in continuous images is compared with the calibration reference line to determine whether it is qualified. The experimental results demonstrate that, compared with the traditional method based on moving distance, the proposed method has higher stability and less strict requirements for the relative moving speed of the camera and detected surface. But this method still has a serious shortcoming. When a new type of car is detected, its calibration reference line needs to be retrieved.

Contents

1	Introduction	1
1.1	Background	1
1.2	Research Problem	5
2	Related Work	8
2.1	Defect Detection Techniques	8
2.1.1	Traditional Image Processing Technology	8
2.1.2	Deep Learning Technology	9
2.1.3	Defect detection of Mirror Surface	10
2.2	Noise Reduction and Removal Techniques	12
2.2.1	Noise Reduction and Removal Techniques with the Bina- rization	12
2.2.2	Noise Reduction and Removal Techniques Uses Multi-frame	13
2.3	Other	14
3	Automatic Defect Detection System	15
3.1	Hardware System	17
3.2	Software System	18

3.2.1	Detection of Defect Candidates	18
3.2.2	Verification of Defect Candidates Based on a Single Frame .	30
3.2.3	Verification of Defect Candidates Based on Consecutive Frames	41
4	Experiments	58
4.1	Datasets	58
4.1.1	Data Classified According to Paint Color and Defect Type .	58
4.1.2	Data Classified According to Paint Color and Car Model .	60
4.1.3	Ground Truth	62
4.2	Validation of Proposal Method	62
4.2.1	Validation of Differential Processing Method	62
4.2.2	Validation of Moving Trajectory Method	74
4.2.3	Validation of Exploring Binarization Method	79
5	Conclusion	84

Chapter 1

Introduction

1.1 Background

In recent years, with the development of industrial technology, competition in the automotive industry has become fiercer, and consumers have higher requirements for automobiles. The surface of the car is mostly coated, and the quality of the coating is the consumers' intuitive evaluation of the quality of the car. At the same time, the coating can improve the corrosion and oxidation resistance of automobiles, which is one of the main measures to prolong the life of automobiles. Therefore, the quality of coating can directly affect the market competitiveness of automobile manufacturers, and defect detection of coating surfaces is a very important work in the automobile manufacturing industry. In the process of car manufacturing, some minor errors, such as foreign objects mixed in the paint, dust in the air, etc., cause some bumps and dents on the surface of the car coating. These defects cannot be completely detected, and every year many cars are returned to the factory to repair, for this reason, causing a lot of damage to the automobile company. In

the automotive industry, the detection of surface defects of automotive coatings is highly dependent on human vision and experience. The car company will arrange for some professional inspectors to inspect the coated car surface. However, the visual acuity and attention of the inspector will be affected by long-term work in this light environment. There are four shortcomings of this manual detection:

- Strong subjectivity: Because each inspector relies on his own visual subjectivity, and its criteria are not uniform, this results in inconsistent criteria.
- Low stability: Due to the long time of testing, eye fatigue occurs, which may cause the inspector to miss some defects that should have been detected.
- Low efficiency: When the defect is not obvious, it needs external conditions (lighting environment) to make it easy to see. In some places where the observation distance is far, such as the top of a car, the inspector may need to rely on props (ladders, etc.) to detect the defect at a close distance, which results in longer detection time.
- High manpower costs: In order to improve the above problems, most automobile manufacturers will use the method of increasing the number of inspectors, which requires more manpower.

Therefore, it is necessary to have a device to assist inspectors in detecting defects in the coated surface.

General automobile coating can achieve an original metal surface coloring, anti-corrosion, anti-oxidation, and other functions, generally with a multi-layer coating structure. There are two main types of automotive coating: solid coating and metal coating. Coating surfaces have the following four optical properties:

- Mirror reflection of normal incident light on a smooth surface.
- Diffusion reflection of incident light on a rough surface.
- Directional reflection of incident light occurring on a glossy material with an inner layer.
- Diffusion and absorption of incident light occurring on the paint layer of the inner coating.

Among these four properties, the mirror reflection of the surface layer has the greatest influence.

Automatic defect detection of coated automobile surfaces is usually based on imitating the steps of manual detection. First, the defects on the car body are located in an environment where they are easy to detect. Then, images of the car body are collected. Finally, millimeter-sized defects are detected and distinguished through image processing and classification. In this study, only the areas with obvious concavity and convexity (more than 0.1 mm) on the detected surface are defined as defects. Therefore, the multi-layer transparent structure of the coating makes it difficult to detect defects after coating because it is impossible to accurately assess whether the change of reflected light is caused by inhomogeneity of the inner layer or by defects on the outer layer.

The main reasons for defects on the outer layer are as follows (also see Fig.1.1):

i) foreign bodies on the surface before painting, ii) fine dust in the air during painting, which will adhere to the paint, and iii) scratches or collision gaps caused by some erroneous operation after painting. Such defects are apparent only at certain

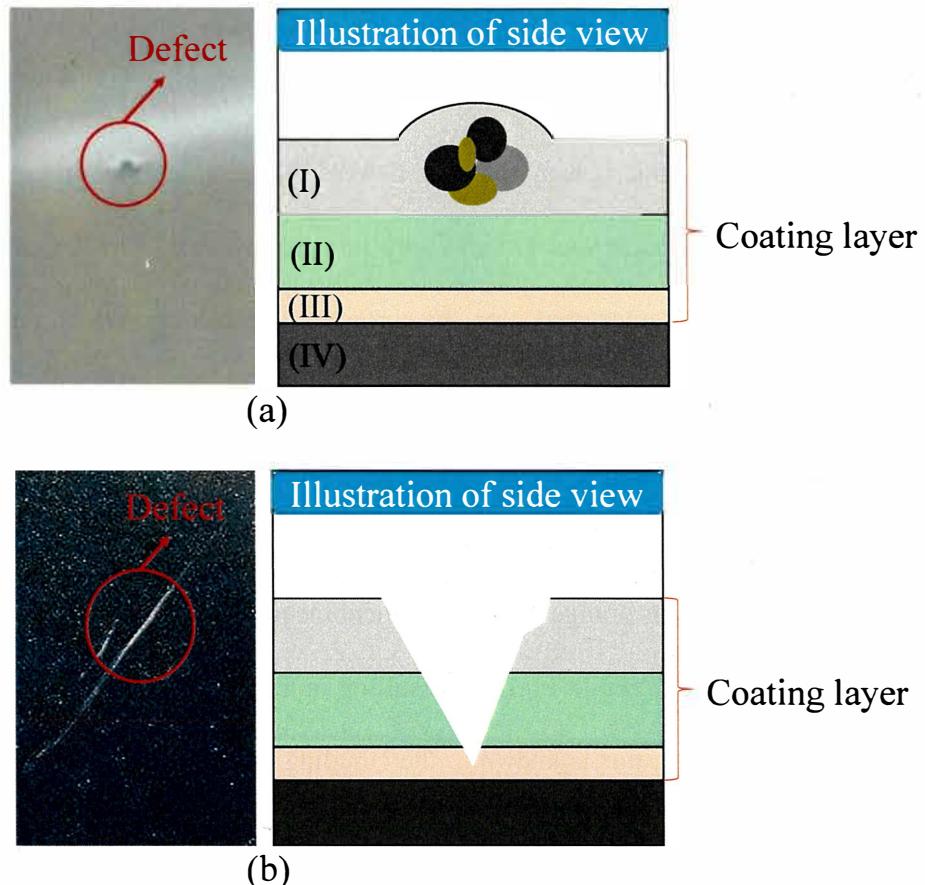


Figure 1.1: Examples of defects. (a) Defects due to dust, (b) Scratches due to collision, (I) Clear coat, (II) Base coat, (III) Primer, (IV) Metal/Body panel.

observation angles. In this study, an area was considered to be a defect through verification by experts.

1.2 Research Problem

Automatic defect detection of automotive coated surfaces is usually based on the simulation of visual detection steps. There are three main steps: First, you must create an environment that is easy to detect defects. Then, collect the images. Finally, the defect is detected and identified by image processing. In the captured image, there is a lot of interference on the surface of the coating, which makes it difficult to detect. For example, during the drying process of the coating, uneven fine lines will be formed due to the drying speed and uniformity of the coating. These uneven, thin lines cause the surface drawn in the image to produce a point that looks like a cloud or orange peel. This phenomenon causes the surface of the coating with light (as shown in Fig.1.2), resulting in a point light that looks like a cloud or an orange peel; this is known as the cludging or orange-peel effect and is not regarded as a defect in this study.

A new automatic defect detection system is proposed in this study. The hardware of this system consists of only fluorescent lamps, a single camera, and a computer for low cost, and the required setup space is small. It can be applied to most detection sites as a stand-in for a human inspector, which greatly reduces labor, time, and personnel costs. Moreover, because of the higher reproducibility of detection results, the individual measurement differences of human inspectors are eliminated. Our system aims to detect all defects and eliminate noise as much

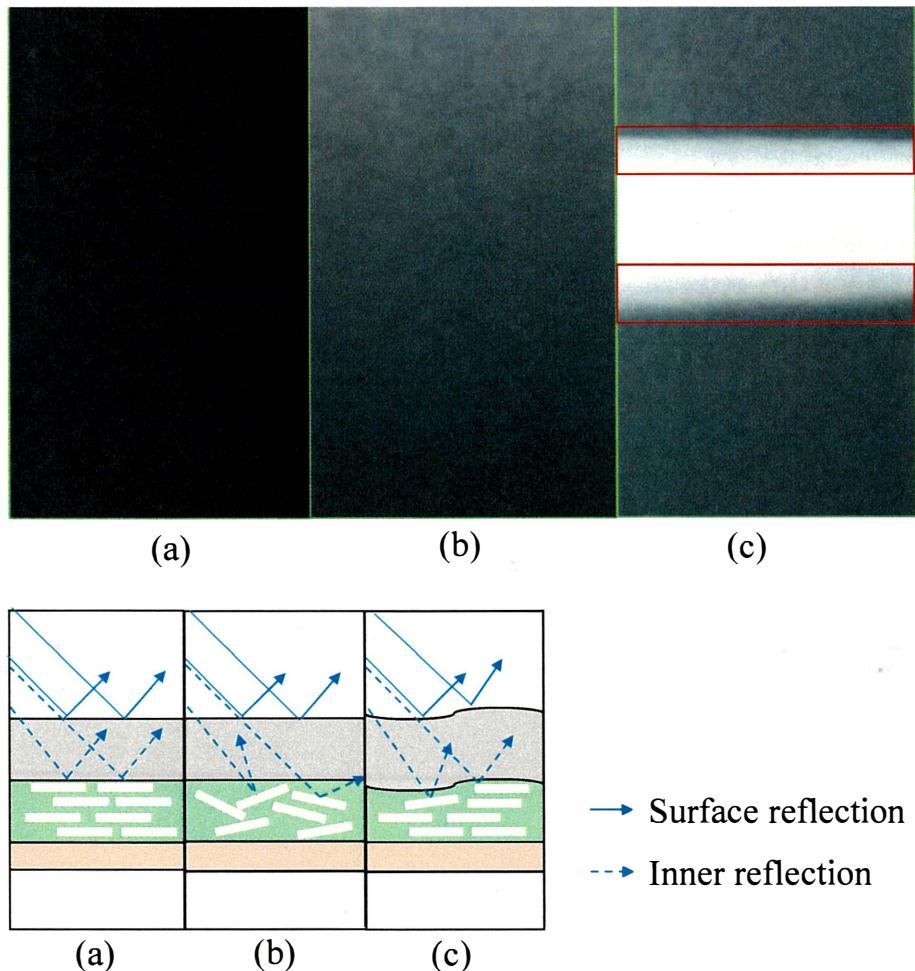


Figure 1.2: Causes of noise in automatic detection. (a) Normal, (b) Clouding, (c) Orange peel.

as possible as a result. Therefore, I use binarization to detect as many defect candidates as possible. To prevent the occurrence of missed detections, I use a wider detection range, that is, the reflection area of the fluorescent lamp and its surrounding area. Then, to eliminate the influences of the above-mentioned noises on defect detection results, we use two verification methods: one is the brightness difference of single-frame differential images and the other is the comparison of multi-frame motion trajectories.

Chapter 2

Related Work

2.1 Defect Detection Techniques

There have been a lot of studies on defect detection. However, the subjects of these studies are not limited to coated surfaces, such as steel [1, 2], magnetic tile [3, 4], road [5], fruit [6], textiles [7], etc. The detection methods of these studies can be divided into traditional image processing technology and deep learning technology.

2.1.1 Traditional Image Processing Technology

Image feature extraction can be understood as a mapping from high-dimensional image space to low-dimensional feature space. It is an important part of surface defect detection based on machine vision. Its effectiveness has a significant impact on subsequent defect target recognition accuracy, computational complexity, robustness, and so on. The basic idea of feature extraction is to make the target have smaller class cohesion and larger class-to-class divergence in the resulting subspace.

At present, the main image features are texture, color, shape, and so on.

The texture is an important feature of an image. It does not depend on color or brightness to reflect the homogeneity of the image. It also reflects important information about the organization of surface structures and their relationship with the surrounding environment. Unlike color and gray-scale features, texture features are not based on the characteristics of pixel points; they require statistical calculations, that is, locality, in areas containing multiple pixel points. At the same time, local texture information also has some degree of repetition, that is, global. Texture features are invariant in rotation and resistant to noise. The texture feature extraction methods based on are statistical, signal analysis, model, structure, and geometry.

Traditional image processing techniques describe and detect defects by extracting features, such as: the gray symbiosis moment proposed by Haralick et al. [8], which is a widely used method to describe textures using statistical features; the 14 texture features based on GLCM in literature[9, 10], which provide more effective texture feature quantities; and the template matching method proposed by Jian et al. [11]; Li et al. [12] propose a novel and robust fabric defect detection method based on the low-rank representation (LRR) technique; local binary patterns (LBP) [13] and a histogram of oriented gradient (HOG)[14] are also common features. However, due to the difference of illumination and background clutter, the above detection methods cannot be applied directly to the coated surface.

2.1.2 Deep Learning Technology

Due to the variety of defects, it is difficult for traditional machine vision algorithms to complete the modeling and migration of defect features, and the reusability is not large, requiring operating conditions to be distinguished, which will waste a lot of

human costs. Deep learning has achieved very good results in feature extraction and location. More and more scholars and engineers begin to introduce deep learning algorithms into the field of defect detection.

In deep learning, defect detection can be viewed as a problem of target detection. For example: Tao et al. [15] uses a compact convolution neural network (CNN) to accurately locate and classify metal defects; Li et al. [16] presents a surface defect detection method based on MobileNet-SSD; Azizah et al. [17] uses a CNN to detect surface defects of mangosteen fruit. However, the accuracy of deep learning is proportional to the number of data and cannot be completely detected. It is difficult to apply when you cannot get a lot of data in advance.

2.1.3 Defect detection of Mirror Surface

A coated automobile surface is regarded as a mirror, but because of the multi-layer structure of the coating, typical metal surface defect detection methods [18] are not directly applicable. At the same time, because vehicle surface defects have complex texture and shape features, the detection system excludes most object recognition methods [19]. Other defect detection methods using infrared and laser technology [20, 21] will increase the cost of defect detection.

Because of the mirror reflection on the coating surface [22], the traditional defect detection method can be used due to the deflectometry principle [21, 23]: the direction of the reflected light on concave or convex defects is irregular (i.e., does not follow specular reflection). For example, traditional automatic detection, as described in [24, 25], uses an external structure consisting of multiple high-frequency fluorescent tubes and a set of cameras, which are located at fixed positions around

the car body. Patents [26] proposed surface defect inspection apparatus based on images of the dark light pattern. In the literature [27], new strategies for painted surfaces based on image fusion technology have been presented and have a very powerful effect in the case of structured as well as smooth surfaces. Fan et al. [28], overcame the strong reflection of the coated surface by a closed indirect diffusion-lighting system. In another study [29], 3D technology was used to overcome deformation problems and to automatically detect and classify body defects.

Mirror detection problems are not only found in the surface of automobiles but also in other fields. One study [30] used an edge detector with image processing technology to detect scratches on painted tiles, and this was applied in a non-destructive manner to detect defects in compact ceramics [31]. However, there are more kinds of defects on the automobile surface, which can not be limited to scratches. Qu et al. [32] proposed an algorithm combining mesh element analysis with the improved penetration detection algorithm for crack detection of a concrete tunnel lining surface. Zou et al. [33] proposed Deep Crack, which is an end-to-end trainable deep convolutional neural network for automatic crack detection that learns high-level features for crack representation. Shi et al. [34] proposed a new road crack detection framework based on stochastic structured forest. Unlike in surface defects on automobiles, concrete roads are flat surfaces, which considerably simplifies the problem of deformation. In the field of agricultural, there are also relevant studies. Patrick et al. [35] developed a method based on an asymmetric second difference method; it uses a chlorophyll absorption waveband and two bands in the near-infrared region to detect the defective parts of an apple, which is not independent of the apple color. Because of the unique properties of plants, it is

difficult to apply this to the surface detection of automobiles.

2.2 Noise Reduction and Removal Techniques

With the improvement of automobile painting technology, the defects after coating have been greatly reduced. Most of the defects appear as dots. Also, noises appear as dots, produced by random reflection. To improve the accuracy of distinguish between these, many practical methods have combined reducing the noise techniques with the binarization.

2.2.1 Noise Reduction and Removal Techniques with the Binarization

Although image binarization has been studied for many years, image thresholding is still a problem, because there is no standard and it is difficult to apply one method to different types of images, such as uneven illumination, image contrast variation, etc. In coating surface detection, classical image segmentation methods, such as OTSU [36] binarization, are ineffective due to the non-uniformity of illumination. In order to solve the problem of binarization of images with non-uniformity illumination, there are two types of methods. One is to preprocess the image first and then use the global threshold. For example, Phong modeling is used to model the character of non-uniform illumination and then used to adjust the OSTU binarization threshold [37]. Research [38] proposed an improved method following recursive method. The global approach is usually simple and effective, however this approach is not suitable for our problem, because the preprocess will change the image information.

Another method is the local threshold method. For example, this commonly used thresholding methods (Niblack [39], Bernsen [40], and Sauvola[41]) and estimated local thresholds of images to produce better binarization results. However, these methods are easy to produce a lot of pseudo noise and are not suitable for defect detection. Research [42] presents an improved particle swarm optimization (IPSO) to solve the local binarization threshold and classify the defects of the steel billet. However, the IPSO method is not fast as binarization. In this paper, a new binarization method is presented, which is combined with the brightness difference method to detect the defect candidate.

2.2.2 Noise Reduction and Removal Techniques Uses Multi-frame

In order to improve the reliability of noise reduction, noise reduction technology usually uses multi-frame pictures to achieve the purpose. For example, in the patent[43], the images taken by multi-cameras are synthesized into one reference image and then compared as a difference image with the original image to obtain the defects. In our method, only a single camera is required. In order to reduce the influence of image noise on detection accuracy, patent[44] utilizes the information of the time axis to detect the noise by the moving distance of defect candidates between two frames. However, if the moving distance is changed, this method cannot be used. In our situation, the distance is not stable. I used the defect trajectory to overcome the noise caused by random reflection on the coating surface. The trajectory of the defect candidate will not change with the relative speed of the camera

and the detected surface. The stability of the trajectory makes this method possible for more objects to be detected.

2.3 Other

In addition, some manufacturers of industrial products claim to be able to inspect and detect defects on the coating surface, such as [45] [46]. However, they do not clarify or indicate how to overcome the problem of detecting unevenness on paint surfaces, nor do they have detailed information about their limitations (i.e., defect size, surface color, inspection coverage, etc.). Different environmental light contamination, material properties of paint, and spraying technologies may have very different effects. Molina et al. [47] introduced a new industrial automatic quality control system named QEYeTunnel, which is based on deflectometry and vision technology. The system meets the requirements of periodic production (approximately 30 s per car). It has been applied in the production line of the Mercedes-Benz factory in Victoria, Spain.

Chapter 3

Automatic Defect Detection System

The overall architecture of the proposed automobile defect detection system is illustrated in Fig.3.1. This system has two main parts: a hardware part for image acquisition and a software part for image processing. The hardware system mainly includes a light source, a camera, a guide rail, and a guide rail control system. The software system includes an image acquisition module, a standard movement trajectory database of automobile defects, and an image processing and result display module. The hardware system is introduced briefly below, and the image processing module of the software system will be introduced in detail.

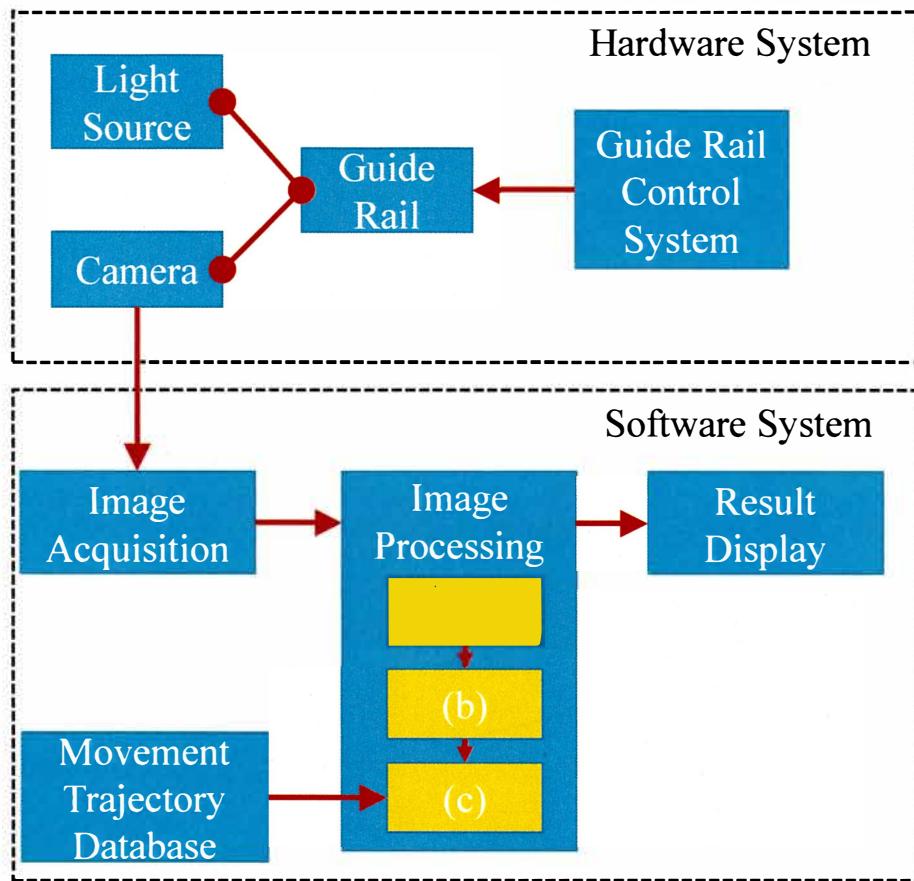


Figure 3.1: Overall architecture of automobile defect detection system. (a) Detection of Defect Candidates, (b) Verification of defect candidates based on single frame, (c) Verification of defect candidates based on consecutive frames.

3.1 Hardware System

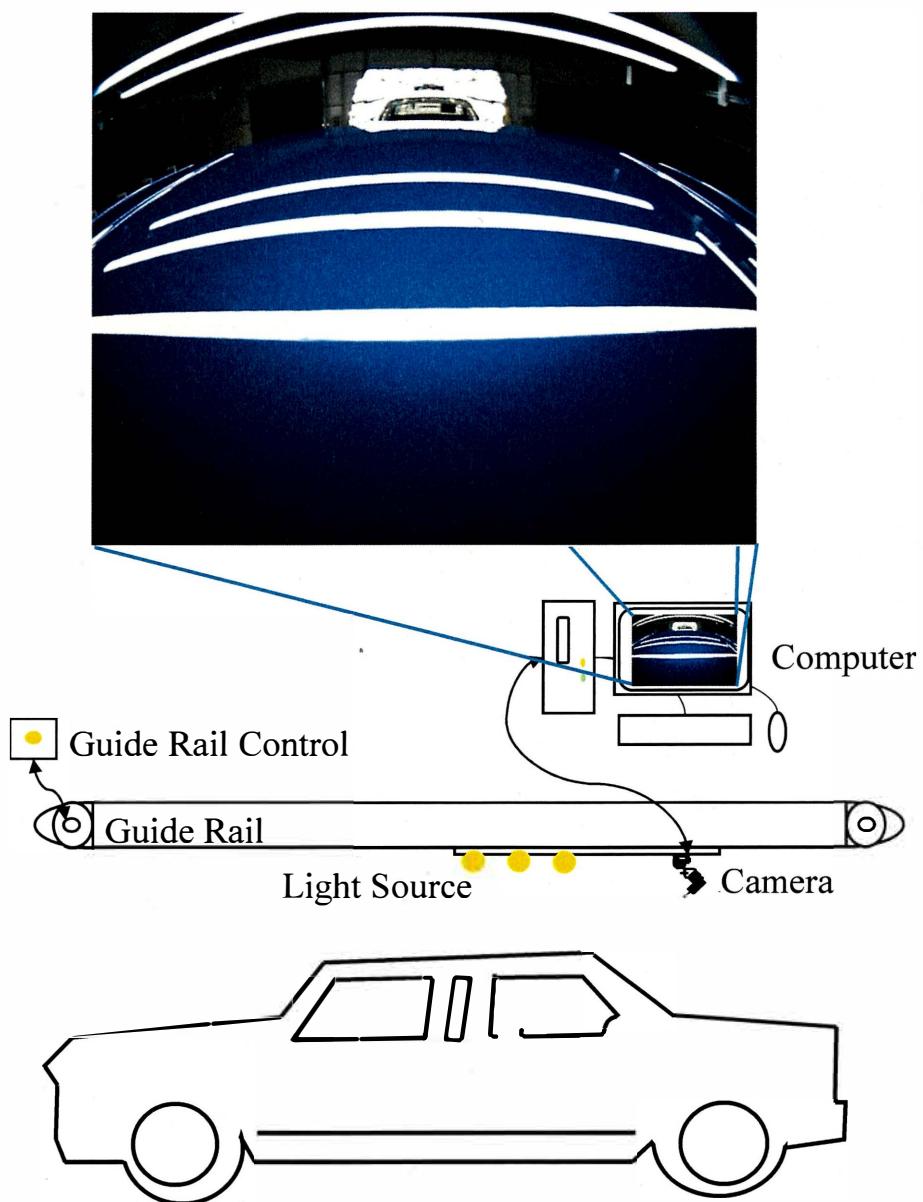


Figure 3.2: Overview of the hardware system.

Our hardware system settings are shown in Fig.3.2. I use LED tubes as the light source, and sequential frames are acquired using a color CMOS camera attached to a guide rail. The objective of this research is to inspect the car roofs, which are difficult to inspect manually, so I put the guide rail on the top of the car. Once the car

being inspected reaches a designated position, the guide rail starts and the camera starts to capture a picture and transmits it to the computer to begin detection. An example captured image is shown in Fig.3.2.

3.2 Software System

The image processing module can be divided into three main parts: detection of the defect candidates based on a single frame, verification of the defect candidates based on a single frame, and a final verification of the defect candidates based on the movement trajectory from multiple consecutive frames.

3.2.1 Detection of Defect Candidates

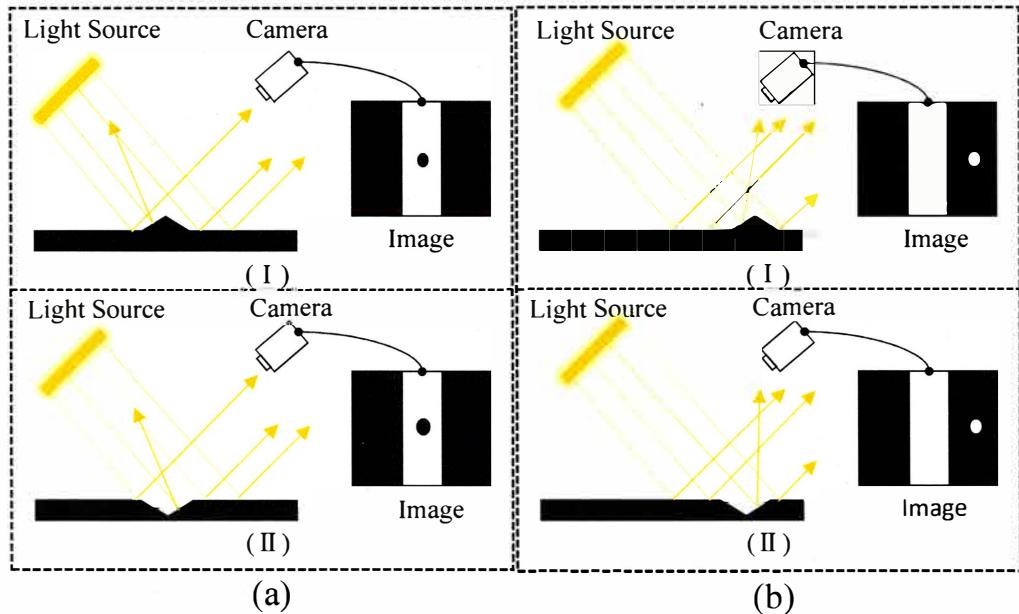


Figure 3.3: Specular reflection. (a) Bright-field illumination, (b) Dark-field illumination, (I) Convex, (II) Concave.

For the detection of scratches and dirt on smooth material surface, effective

polishing methods are coaxial light, high angle, low angle, back light, etc. In my study, defect candidate detection is based on direct reflection light. If I illuminate the part with a beam of light parallel to the scratch, the scratch will be visualized by the light and the effect will not be obvious in the image. If I illuminate the part with a beam of light perpendicular to the scratch, the scratch will be highlighted by the light and the effect will be obvious in the image. The principle of this method is very simple, as shown in Fig.3.3. The oblique direction of the light source illuminates the detected surface, and then the reflected light is captured by the camera to obtain a picture. The defective part of the picture will be different from its surrounding parts because of the chaotic reflection. When the defect is illuminated in the bright field, its pixel value is lower than that of the surrounding pixels, and when the defect is illuminated in the dark field, its pixel value is higher than that of the surrounding pixels.

In order to find the defect candidate in the picture, two important steps will be taken, one is to find the Region of Interest (ROI) and the other is to binarize the regions.

Detection of ROI

ROI detection is mainly divided into three steps: First, the input color image is converted to a gray-scale image, and the noise is reduced by a median filter. This is to remove the effects of uneven lighting and color in the image.

Secondly, the gray-scale image is binarized. Given the global binarization threshold, the bright area is identified as the reflective area of the LED. The global

binarization threshold is calculated by Eq.3.1.

$$Th = I_{mean} + \beta I_{std} + \alpha \quad (3.1)$$

where Th is threshold calculation of binarization. I_{mean} and I_{std} are the mean intensity and standard deviation of the gray-scale image, respectively, and α and β are constants.

Finally, based on the reflected area of the LED, its bounding rectangle is obtained and expanded to obtain the ROI in this study. The up and down expanding sizes are 1/12 of the image resolution, and the middle area is the middle value of the reflected area of the two adjacent LEDs. The ROI of each LED reflection is divided horizontally shown in Fig.3.4.

Exploring Binarization

From Fig.3.3, We can see that there are two cases of defect detection, one is to find the dark pixel group in the bright field, the other is to find the bright pixel group in the dark field. In the first case, the brightness value of the LED reflection (the reflective area of the LED tube) area approaches 255, and environmental changes have little effect on it. I just need to perform local binarization in the white areas and detect the black blocks that meet the size criteria. The second case is more complex and produces more noise than the first. For example: due to different colors of paint and uneven lighting problems, will cause different gray levels in areas outside the LED reflection area; due to the proximity of the LED tube area, the halo and distortion will be generated.

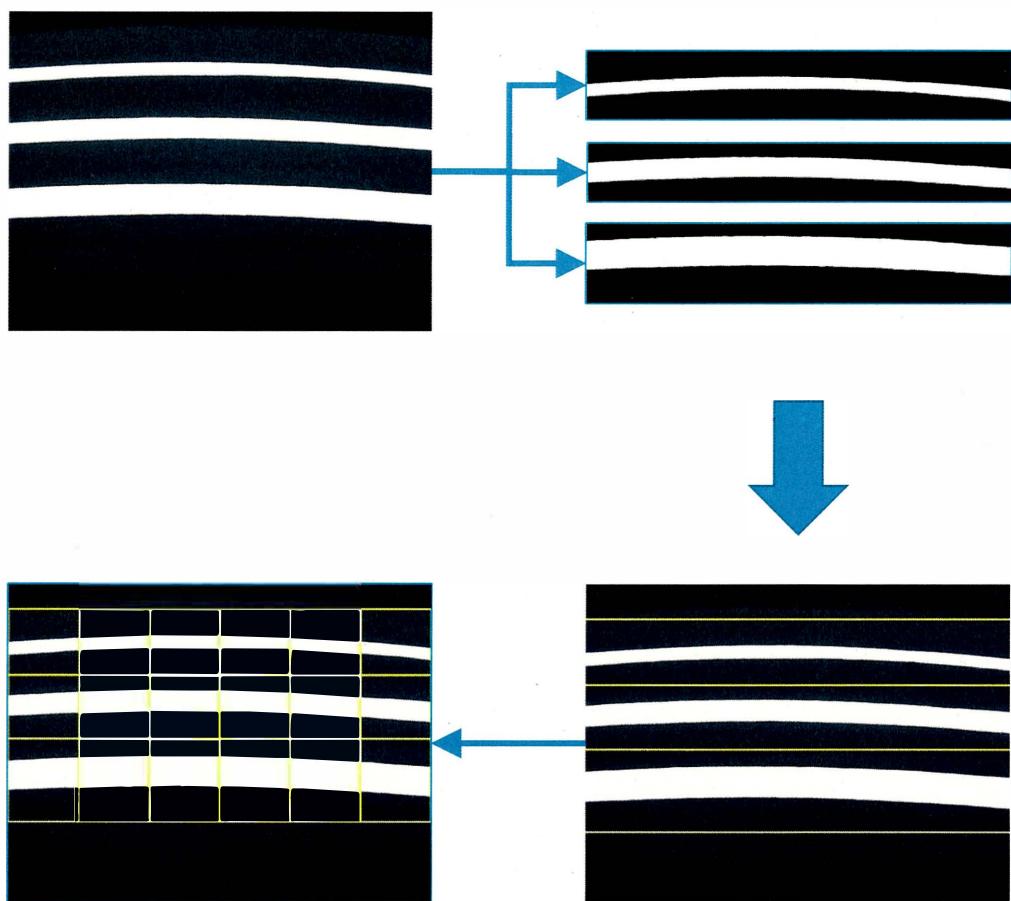


Figure 3.4: Detection of ROI

In the image, we are interested in the region where the product is located, which is called the target; The non-product area is the area that does not want to be processed, which is collectively referred to as the background. The purpose of image background segmentation is to remove the background region in the image. Background segmentation is the basic premise of image visual analysis and pattern recognition. The accuracy and speed of image feature extraction and classification depend on the quality of image background segmentation. Researchers have proposed many kinds of background segmentation algorithms.

Threshold segmentation is a widely used image segmentation technology. It makes use of the difference between the gray characteristics of the target and the background to be extracted in the image. By selecting the threshold, the image is divided into a combination of two types of regions (target and background) with different gray levels. The accuracy of threshold selection is directly related to the quality of segmentation.

How to determine the optimal threshold is a key problem in image segmentation. If the threshold is too small, some pixels that are not the background will also be regarded as the background, resulting in false recognition; If the threshold is too large, some background will be missed, resulting in missing recognition. Several threshold selection methods are analyzed below:

- (1) Threshold selection method based on image gray histogram

The method based on image gray level histogram analysis refers to determining the threshold directly from the original gray level histogram of the image.

It is the most intuitive and widely used image segmentation method. This

method is simple and easy, and achieves good segmentation effect. However, for the image with no obvious peak or wide and flat trough in the gray histogram, it is difficult to select the threshold according to the peak and valley of the histogram. At this time, the use of this method is limited. Moreover, the external brightness fluctuates under different experimental conditions, which will have a great impact on the threshold selection method of image gray histogram.

- (2) Threshold selection method based on image gap

The typical method of this kind of method is the maximum interclass variance method. This method can make the gap between the segmented target and the background very large, that is, there is a high contrast between the target and the background. This method is simple to calculate and is not affected by the change of image contrast and brightness under certain conditions. The maximum interclass variance method is a more successful one of these methods. This method is derived based on the principle of discriminant analysis and least square method. The optimal threshold is obtained when the discriminant reaches the maximum.

- (3) Threshold selection method based on Entropy

In the early 1980s, people began to consider using the concept of entropy in information theory for threshold selection. Pun first proposed the maximum a posteriori entropy upper bound method in 1980, and Kapur et al. Proposed the one-dimensional maximum entropy threshold method in 1985. According to the principle of maximum entropy, a threshold T is selected to maximize

the amount of first-order gray statistical information of the two parts of the image segmented by this threshold. That is, one-dimensional entropy is the largest. Compared with the maximum interclass variance method, the one-dimensional maximum entropy method involves logarithmic operation. The speed is slow and the real-time performance is poor.

- (4) Approximate iterative method

The principle of this method is to approximate the histogram with two or more probability density functions of normal distribution. The threshold is the nearest gray value corresponding to the minimum probability between the maximum values of two or more normal distributions, and the result has the segmentation of minimum error. Here, the error includes mistaking the target as the background and merging the background and noise as the target, Minimize the sum of the two. The iterative threshold segmentation effect of the target image is very good, and can better distinguish the main areas of the foreground and background of the image. However, the iterative algorithm has a large amount of computation, and often has no more obvious advantages than Otsu algorithm.

- (5) Other threshold selection methods

In recent years, with the development of genetic algorithm, wavelet transform, mathematical morphology and fuzzy set theory, these theories have also been introduced into the field of image threshold segmentation, resulting in many new thresholding methods. Although these algorithms have good computational effects, they are difficult to meet the requirements of real-time

detection because of their high computational complexity.

In image processing, it is generally greatly disturbed by external light. If a fixed threshold is used in threshold segmentation, the segmentation effect will be greatly affected when the environment changes. In order to avoid this effect, we must use dynamic threshold to automatically calculate the appropriate threshold to separate the target image from the background image. Image thresholding segmentation is one of the most commonly used, and it is especially suitable for images with different gray levels of target and background. Common image binarization algorithms can be roughly divided into two types: global threshold method and local threshold method.

The global threshold method refers to the method that only one global threshold T is used in the binarization process. It compares the gray value of each pixel of the image with T . If it is greater than t , it is taken as the foreground color (white); Otherwise, it is the background color. A threshold is determined according to the histogram or gray spatial distribution of the text image, so as to realize the transformation from gray text image to binary image. The global threshold method can be divided into point based threshold method and region based threshold method. The result of threshold segmentation method largely depends on the selection of threshold, so the key of this method is how to select the appropriate threshold. Typical global threshold methods include Otsu method, maximum entropy method and so on. The algorithm of global threshold method is simple. It has good effect on the image with obvious separation of target and background and bimodal histogram distribution, but the binarization effect is obviously worse for the image with non-

uniform illumination and large noise interference. In order to meet the automation and real-time requirements of image processing application system, the selection of image binarization threshold is best completed automatically by computer. Several automatic threshold selection algorithms are listed below:

- Average gray value method: take the average gray value of all pixels in the image as the threshold.
- Otsu method: also known as the maximum inter class difference method, it realizes the automatic selection of threshold based on the statistical characteristics of the whole image. The principle is to divide the image histogram into two categories with a certain gray value, calculate the pixel number and gray average value of the two categories respectively, and then calculate their inter class variance. When the variance between the two classes is the largest, the gray value is used as the threshold of image binarization. Otsu method is widely used. Whether there are obvious bimodals in the histogram of the image or not, it can get satisfactory results, and has been applied and developed in many fields. However, this method still has some shortcomings, mainly manifested in: if the gray difference between the target and the background is not obvious, large black areas may appear, and even the information of the whole image may be lost; Only one-dimensional gray histogram distribution is used, and the spatial correlation information of the image is not combined, so the processing effect is not good; When there is fracture in the image or there is some noise in the background, the expected effect cannot be obtained.
- Edge operator method: Laplace operator, Robert operator and sober operator

are used to enhance or weaken the gray level of pixels. For the pixels in the region with uniform gray distribution, these operators weaken the gray level; These operators enhance the gray level of pixels near the edge.

The local threshold method determines the threshold of the pixel by the current pixel gray value and the local gray characteristics of the points around the pixel. For example, the original image can be divided into some disjoint blocks, the gray mean of each block image can be used as the threshold of the block image, and the above overall threshold method can be used locally.

For the image with clear target and background, the global thresholding method can achieve better results. However, if the background of the image is uneven, or the change rate of the target gray is relatively large, the global method is no longer applicable. The local threshold method uses the pixel gray value and the local gray characteristics of the pixel neighborhood to determine the threshold of the pixel. When the illumination is uneven, there is sudden noise, or the background gray changes greatly, the local threshold determination technology must automatically determine different thresholds according to the coordinate position relationship of pixels and implement dynamic adaptive binarization. Local threshold selection generally divides the image into several sub images, and uses the overall threshold method on each sub image area, so as to form the local threshold method of the whole image (the corresponding threshold is determined according to each sub image, and the specific threshold determination method is similar to the global threshold determination). The segmented image has discontinuous gray distribution at the boundary of different sub images, so smoothing technology must be used to

eliminate gray discontinuity.

The local threshold method is generally used to identify the images with serious interference and poor quality. Compared with the overall threshold method, it is more widely used, but it also has disadvantages and problems, such as slow implementation speed, unable to ensure the connectivity of character strokes, and prone to artifacts (that is, the stroke results are obtained by noise interference in the background domain). Typical local binarization algorithms include Bernsen method, multi threshold gradient intensity method, texture image-based method, maximum variance method and so on.

In this study, the method shown in Fig.3.5 is used.

The best binarization thresholds are explored for each region after segmentation, and the results are used for binarization. The exploration of the optimal threshold for binarization uses a threshold-decreasing approach. The threshold decreases from the maximum until the number of white blocks in the binary image increase substantially.

Finally, just pull out the defect candidate that meets the requirements inside the ROI. My goal is to detect defects in two situations: when a black pixel is surrounded by a white area, and when a white pixel is surrounded by a black area. In the first case, I locally binarize the above approach and detect defects that meet the size criteria. In the second case, I first remove the LED tube area and then explore the remaining area. In my method, you can reverse and save the pixels in the white area (white pixels become black, black pixels become white) so that the two cases after binarization are the same.

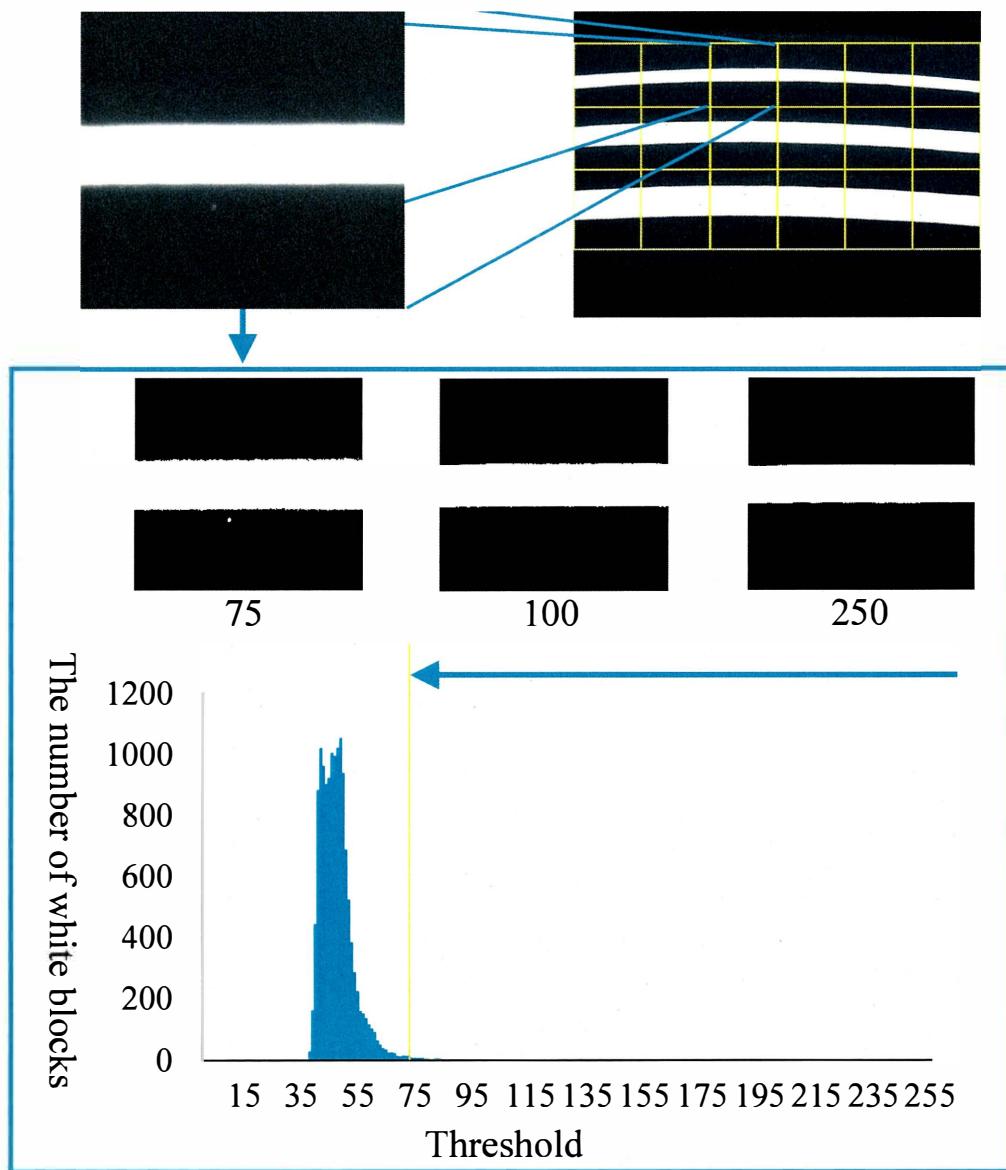


Figure 3.5: Exploring binarization

3.2.2 Verification of Defect Candidates Based on a Single Frame

In the last step, we use binarization to determine the defect candidates. The binarization method is implemented by selecting a single threshold value from the information of a gray-level histogram, which is effective in defect detection on a surface without noise influence. However, there is fine unevenness over the whole coating surface, such as orange peel and clouding, which cannot be regarded as defects so that they do not affect the detection results. Next, I must select a high-quality defect candidate from the detected candidates.

In this study, two screening methods are proposed, and their effects are compared, and the most effective method is selected.

Differential Processing

Image filtering can greatly reduce the external noise interference caused by the image acquisition process, enhance the edges of the image, and improve the quality of the original image. The segmentation algorithm may also introduce noise due to changes in the external environment. Therefore, it is necessary to use a filter algorithm to remove noise and improve the image quality. In the process of background segmentation, the contrast method is sensitive to the change of illumination. If the external conditions change a lot, the image containing noise will also be obtained, which will inevitably affect the whole process of recognition processing and the output of the result, and even make a wrong recognition. Therefore, image filtering is particularly important and necessary to obtain the ideal image. There are now many ways to filter images. Distribution by scope can be divided into two types: spatial filter and transform filter:

- (1) Spatial Domain Filtering Method Spatial domain filtering is accomplished by using a template to perform neighborhood operations in the image spatial domain. According to the characteristics of the structure, it can be divided into two categories: linear filter and non-linear filter. Linear spatial filter system mainly relies on transfer function, pulse function or point spread function to filter neighborhood. Nonlinear filter systems mostly process neighborhoods directly. The spatial filter algorithm uses small-area template convolution. This template convolution operation. It is a very basic processing method in digital image processing.
- (2) Frequency domain filtering method Frequency domain filtering is also known as transformation domain filtering. It is an important means of image processing in frequency domain. Most of the processing is done in the transformation domain. Common frequency domain filters include Butterworth, Chebyshev, Server filters, etc.

Because the frequency domain filtering method needs to be done in the transformation domain, and this frequency domain change often requires complex calculation, which cannot meet the real-time requirements of the detection system, this paper uses the spatial domain filtering method. Neighborhood averaging (linear filtering) and median filtering (non-linear filtering) are the most widely studied and applied methods in the spatial domain filtering methods. Firstly, the filtering effects of the two methods in the injection system are analyzed and compared. Then, based on the advantages and disadvantages of each method, a filter method suitable for this system is designed.

Neighborhood averaging is a simple spatial smoothing noise processing method and belongs to linear filter. The basic idea of this method is to replace the gray value of a pixel with the average value of several pixels. Assuming that each pixel (j, k) in the image $F(x, y)$ has a gray value of $F(j, k)$, centered on it, A represents the point set of window pixels, and L represents the number of pixels in the set. Thus, after filtering by the neighborhood averaging method, the corresponding output of the pixel $F(j, k)$ is:

$$g(j, k) = \frac{1}{L} \sum_{(x,y) \in A} F(x, y) \quad (3.2)$$

When smoothing with the neighborhood averaging method, there are usually two ways to select a neighborhood: radius per unit distance or radius $\sqrt{2}$ times the distance per unit. Taking a 3×3 window as an example, when the unit distance is the radius, its neighborhood is:

$$A_4 = \{(j, K - 1), (j + 1, k), (j, k + 1), (j - 1, k)\} \quad (3.3)$$

Called a four-point neighborhood, $L = 5$ when four-point neighborhood.

When the radius is $\sqrt{2}$ times the unit distance, the neighborhood is:

$$A_8 = \begin{bmatrix} (j - 1, K - 1) & (j - 1, k) & (j - 1, k + 1) \\ (j, k - 1) & & (j, k + 1) \\ (j + 1, k + 1) & (j + 1, k) & (j + 1, k + 1) \end{bmatrix} \quad (3.4)$$

Called an eight-point neighborhood, $L = 9$ when eight-point neighborhood.

Assume that noise n is additive noise and is not correlated with each other in space, and the expected value is 0, the variance is σ^2 , and g is an uncollected image. The image f containing noise is averaged by the neighborhood:

$$\bar{f}(m, n) = \frac{1}{N} \sum f(i, j) = \frac{1}{N} \sum g(i, j) + \frac{1}{N} \sum n(i, j) \quad (3.5)$$

From the formula, the mean value of noise is unchanged and the variance is reduced from σ^2 to $N^{-1}\sigma^2$ by the neighborhood averaging method. It shows that the intensity of the noise is reduced, that is, the noise is strongly suppressed. However, at the same time, image blurring (widening) due to the average effect also occurs, which affects the positioning of edges in subsequent processing, and the degree of image blurring is proportional to the radius of neighborhood.

To improve it, a weighted average filter can be obtained by modifying the mean filter mentioned above. The commonly used weighted mean filters are as follows:

$$H_1 = \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (3.6)$$

$$H_2 = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad (3.7)$$

$$H_3 = \frac{1}{8} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (3.8)$$

The main problem of neighborhood averaging filter operation is to fuzzify all sharp and discontinuous parts of the image, that is, it is possible to remove edge information points in the image as noise points and lose some edge information. This is a big disadvantage in high-precision image measurement system, which requires better preservation of image edge details. The median filter algorithm can reduce the loss of edge information and overcome the blurred image details while filtering noise. Here we will focus on this filtering method.

Image median filtering is a non-linear signal processing method proposed by Turky in 1971. Its principle is to use a sliding window with odd points to replace the value of the center point with the median value of each point in the window. The result obtained by this method will not be affected by points whose pixel gray values in the neighborhood are very different from those of the current processing point. Therefore, this method can remove impulse noise, salt and pepper noise while better preserving the edge details of the image. It can overcome image detail blurring caused by mean filter method. In one-dimensional form, median filter is a sliding window with odd number of pixels.

The concept of one-dimensional median filter can be easily extended to two-dimensional. At this time, take some form of two-dimensional window, sort the pixels in the window, and generate monotonic two-dimensional data series. Generally speaking, a two-dimensional median filter suppresses noise better than a one-

dimensional median filter. The window shapes of a two-dimensional median filter are linear, square, cross, circular, diamond, and so on. Filter windows of different shapes produce different filtering effects. Using median filter must be selected according to the content of the image and different requirements. Based on previous experience, square or circular windows are suitable for images of objects with long outlines, while cross-shaped windows are preferred for images of objects with sharp top angles.

Median filter can remove salt and pepper noise, impulse noise and maintain image edge details. However, it cannot be adapted to changes in local noise levels. Furthermore, fixed square or rectangular templates cannot adapt or mediate the direction of the change in stripes. For a large area of noise pollution, the median filter is less capable of smoothing noise than the mean filter. This is because if most of the image points in the filter window (i.e. the neighborhood) are polluted by noise, the output of the median filter is still a noise-polluted pixel, while the mean filter averages the noise, smoothing the noise to some extent.

Differential imaging is one of the simplest and most commonly used methods in the field of image processing. It can be used to remove unwanted additive patterns in an image, which may be slowly changing background shadows, periodic noise, or additional contamination known at every pixel of the image. Differential imaging is used in dynamic detection and recognition, angiography, and diagnosis of printed circuit board mask defects. It can be used to detect changes between two images in the same scene.

Because the difference method depends on the selection of the threshold value, the background segmented image can easily produce noise when the environment

changes. This paper chooses a differential method consisting of multiple filters.

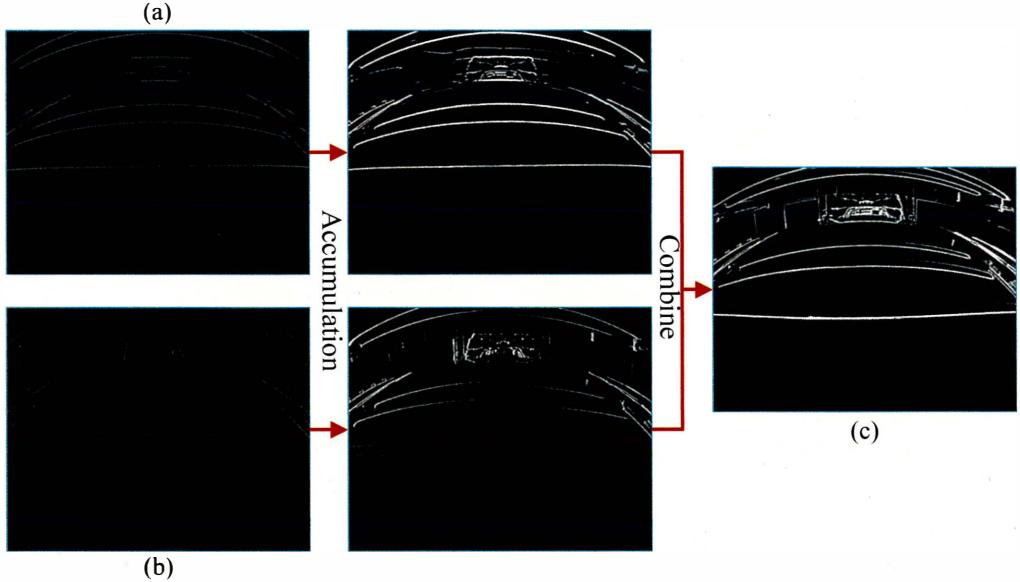


Figure 3.6: Integration of differential image. (a) Differential image in horizontal, (b) Differential image in vertical, (c) Differential image in two different directions.

Differential processing is a commonly used method for extracting contours and calculating features of objects. A previous study [48] provided us with a multi-directional differential method. It can effectively reduce the impact of the orange peel effect by using the composite differential image of intensity changes obtained from the subtraction of two different directions to verify the defect candidate.

Specifically, differential images of input images in horizontal and vertical directions are first calculated by Eq.3.9 and Eq.3.10. In the calculation results, the positive values indicate that the intensity of the area around the edge changes from bright to dark, and the negative values indicate that the intensity of the area around the edge changes from dark to bright, as shown in Fig.3.6-(a) and Fig.3.6-(b). Then, Eq.3.11 is used to combine them into a two-directional differential image, as shown in Fig.3.6-(c). Using the above method, I can obtain the upper left and lower right

differential images and synthesize them. For the convenience of observation, the values of red and green are given respectively, as shown in Fig.3.7-(c).

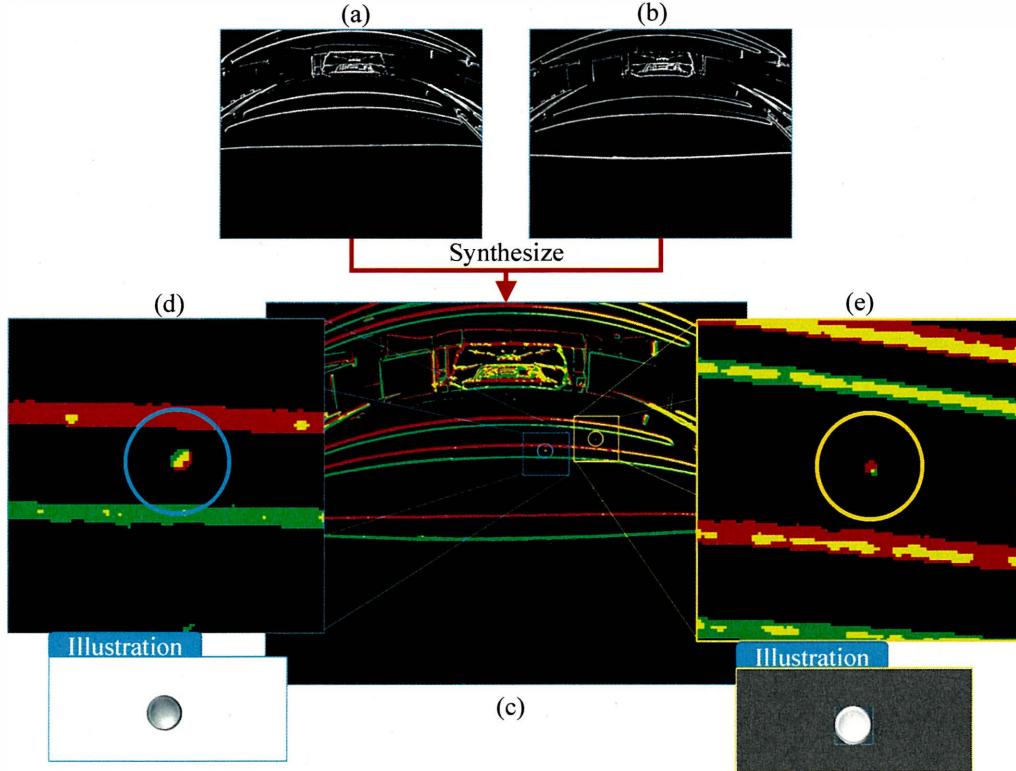


Figure 3.7: Integration and utilization reference image. (a) Upper left differential image, (b) Lower right differential image, (c) Reference image, (d) Defects in LED tube, (e) Defects outside LED tube.

$$G_{x_n}(x, y) = (-1)^n \sum_{j=-\frac{k}{2}}^{\frac{k}{2}} [I(x - \gamma s, y + \omega j - \gamma(1 - \omega)s) - I(x + \theta s, y + \omega j + \theta(1 - \omega)s)] \quad (3.9)$$

$$G_{y_n}(x, y) = (-1)^n \sum_{i=-\frac{k}{2}}^{\frac{k}{2}} [I(x + \omega i + \gamma(1 - \omega)s, y - \gamma s) - I(x + \omega i - \theta(1 - \omega)s, y + \theta s)] \quad (3.10)$$

$$G_n(x, y) = \sqrt{(\varphi(G_{x_n}(x, y)))^2 + (\varphi(G_{y_n}(x, y)))^2} \quad (3.11)$$

where $G_{x_n}(x, y)$ and $G_{y_n}(x, y)$ correspond to the horizontal and vertical difference in two orthogonal directions, respectively. (x, y) represents the coordinates of the target pixel, and $I(x, y)$ is the intensity value of the target pixel. Parameter n represents the control variable of the difference direction, k and s are parameters for controlling the size of the mask that takes the difference. ω is a variable that controls the orthogonal axis, and γ and θ are variables that control the positional relationship between two pixels; γ and θ cannot both be zero at the same time. $\varphi(t)$ is computed using Eq.3.12.

$$\varphi(t) = \begin{cases} t, & \text{if } t > 0 \\ 0, & \text{otherwise.} \end{cases} \quad (3.12)$$

The method of reconfirmation of the defect candidate is shown in Fig.3.7-(d) and Fig.3.7-(e). According to the order of the defect edges in the LED tube from top left to bottom right, the color changes from light to dark and then to light, and

the defects in the reference image go from green to red. Similarly, the defect edges around the LED tube change from dark to light and then to dark from the top left to bottom right. The defect in the reference image is shown as red to green. Only when one of the above two conditions is satisfied can the candidate defect go to the next step.

Segmenting Noise Areas and Brightness Difference

Under the current hardware conditions of image processing, the processing and analysis of color images directly is complex and time-consuming, so the processing of color images is usually converted into gray images first, and then processed according to the gray image processing method.

To extract the feature of the defect, a method for calculating the target white block brightness difference in an area is presented. The calculation method of brightness difference refers to Eq.3.13.

$$Gh(x, y) = V(x, y) - \frac{\sum_{x+n}^{i=x-n} \sum_{y+n}^{j=y-n} V(x, y)}{(2n + 1)^2 - 1} \quad (3.13)$$

Where $Gh(x, y)$ is the brightness difference (x, y) of the target pixel, and $V(x, y)$ is the gray value of the target pixel. To eliminate the effects of orange peels and clouds around the LED reflection, the detection area was divided into six parts, as shown in the figure. Because white and metal lacquers are areas where orange peel is seriously affected, the brightness difference of the defects of the two lacquers was investigated.

The table above is a typical example. Tables 3.1 to 3.4 are white paint and

Table 3.1: Investigation Results of Defects in White Coating

Brightness Difference	Area 1	Area 2	Area 3	Area 4	Area 5	Area 6
0 9	0	0	0	0	0	0
10 19	0	0	0	0	0	0
20 29	0	0	0	0	0	0
30 39	0	1	0	0	0	0
40 49	0	2	0	0	0	0
50 59	0	1	0	0	0	0
60 69	0	0	0	0	0	0
70 79	0	0	0	0	0	0
80 89	0	1	0	0	0	0
90 255	0	0	0	0	0	0

Table 3.2: Investigation Results of Noise in White Coating

Brightness Difference	Area 1	Area 2	Area 3	Area 4	Area 5	Area 6
0 9	3080	1452	1743	31	37	20683
10 19	6993	4705	1583	28	22	902
20 29	2947	2643	94	0	0	0
30 39	271	2002	37	0	0	0
40 49	275	1766	12	0	0	0
50 59	533	1516	2	0	0	0
60 69	614	798	0	0	0	0
70 79	385	283	0	0	0	0
80 89	185	102	0	0	0	0
90 255	87	49	0	0	0	0

Table 3.3: Investigation Results of Defects in Metallic Coating

Brightness Difference	Area 1	Area 2	Area 3	Area 4	Area 5	Area 6
0 9	0	0	0	0	0	0
10 19	0	1	0	0	0	0
20 29	0	0	0	0	0	2
30 39	0	1	0	0	0	16
40 49	0	0	0	0	0	8
50 59	0	0	0	0	5	2
60 69	0	0	0	1	0	0
70 79	0	0	0	3	0	0
80 89	0	0	1	0	0	0
90 255	0	3	2	0	0	0

Table 3.4: Investigation Results of Noise in Metallic Coating

Brightness Difference	Area 1	Area 2	Area 3	Area 4	Area 5	Area 6
0 9	1	35154	2903	262	155	11034
10 19	5	20887	972	648	525	4452
20 29	19	163	19	7	2	16
30 39	46	24	0	0	0	0
40 49	52	41	0	0	0	0
50 59	50	28	0	0	0	0
60 69	28	12	0	0	0	0
70 79	10	2	0	0	0	0
80 89	5	0	0	0	0	0
90 255	68	4	0	0	0	0

metal paint survey results. The results show that the white coating in the image has a higher pixel value, is close to the defect location, has a smaller brightness difference and is not easy to observe. Regarding noise, in area 2 near the LED area, all colors have the greatest noise. Based on the above results, it can be determined that area 2 is a noisy area and area 3-6 is a detection area.

3.2.3 Verification of Defect Candidates Based on Consecutive Frames

Many traditional methods treat still images one by one [49] [50]. Therefore, when there is noise in an image, it is recognized as a defect, which is a problem of the conventional method. To solve this problem, a study [25] proposed a method based on the defect augmentation phenomena. The method causes shadows to surround defects when merging consecutive images. However, the realization of defect augmentation phenomena requires the assistance of specific external equipment conditions. A study [44] proposed another method, which infers the result of the current frame from the detection result of the previous frame and compares it with the ac-

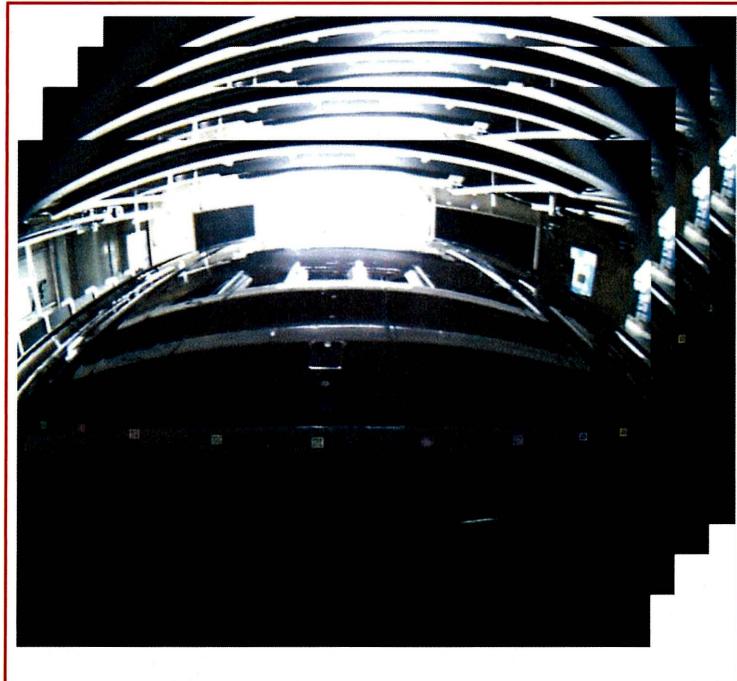
tual result to judge the authenticity of the defect. This method not only requires a certain relative moving speed between the detected surface and the camera but also has an unstable effect on the curved surface.

In this paper, a new defect verification method based on continuous frame is proposed. I put many labels on the tested car to simulate defects and photographed and recorded the label coordinates of each frame. The results are then analyzed to obtain a calibration reference line, as shown in Fig.3.8. Our method is to reconfirm the defect candidate according to these regular trajectories. The specific steps are shown below.

First, I must obtain the reference line of the movement trajectory. I suggest using a checkerboard to automatically detect the reference line, which is a crossover point as the reference point has a very good accuracy and is more convenient to make, and to provide help for the location of defect detection results, as shown in Fig.3.9.

Then, I must find the trajectory of the defect candidate in continuous frames and compare it with the reference trajectory to judge whether it qualifies or not. Because of the curvature of the roof of the car being inspected and the fish-eye lens problem of the camera, the reference trajectory is not a straight line but a curve, which makes it difficult to judge. Therefore, I obtain an approximate curve of the polynomial curve fitting method and minimize the sum of the errors between the calculated data and the actual data by the ordinary least squares [51] method, as

(a)



(b)

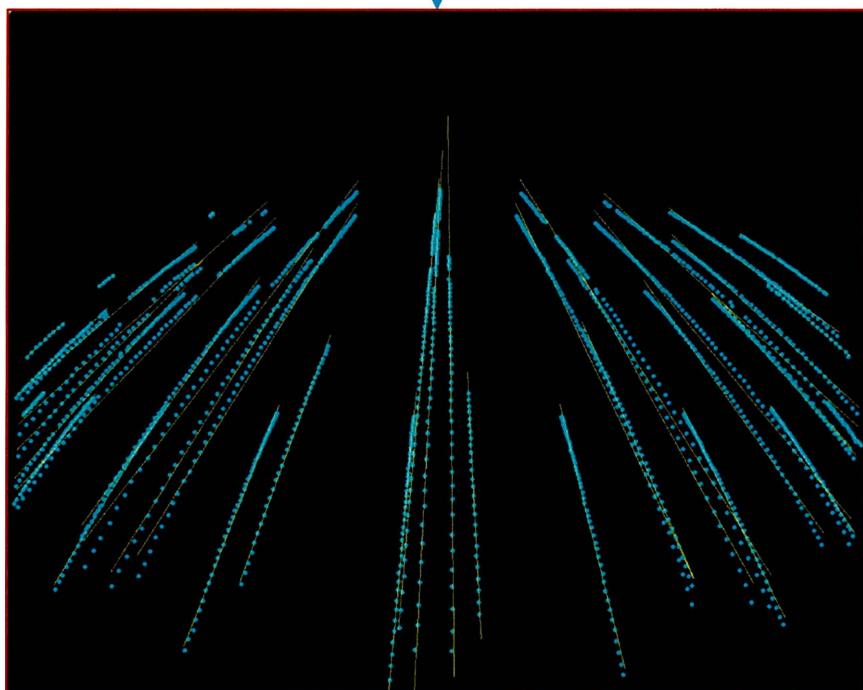


Figure 3.8: Label-based trajectory. (a) Labeled surface images, (b) Regular trajectory.

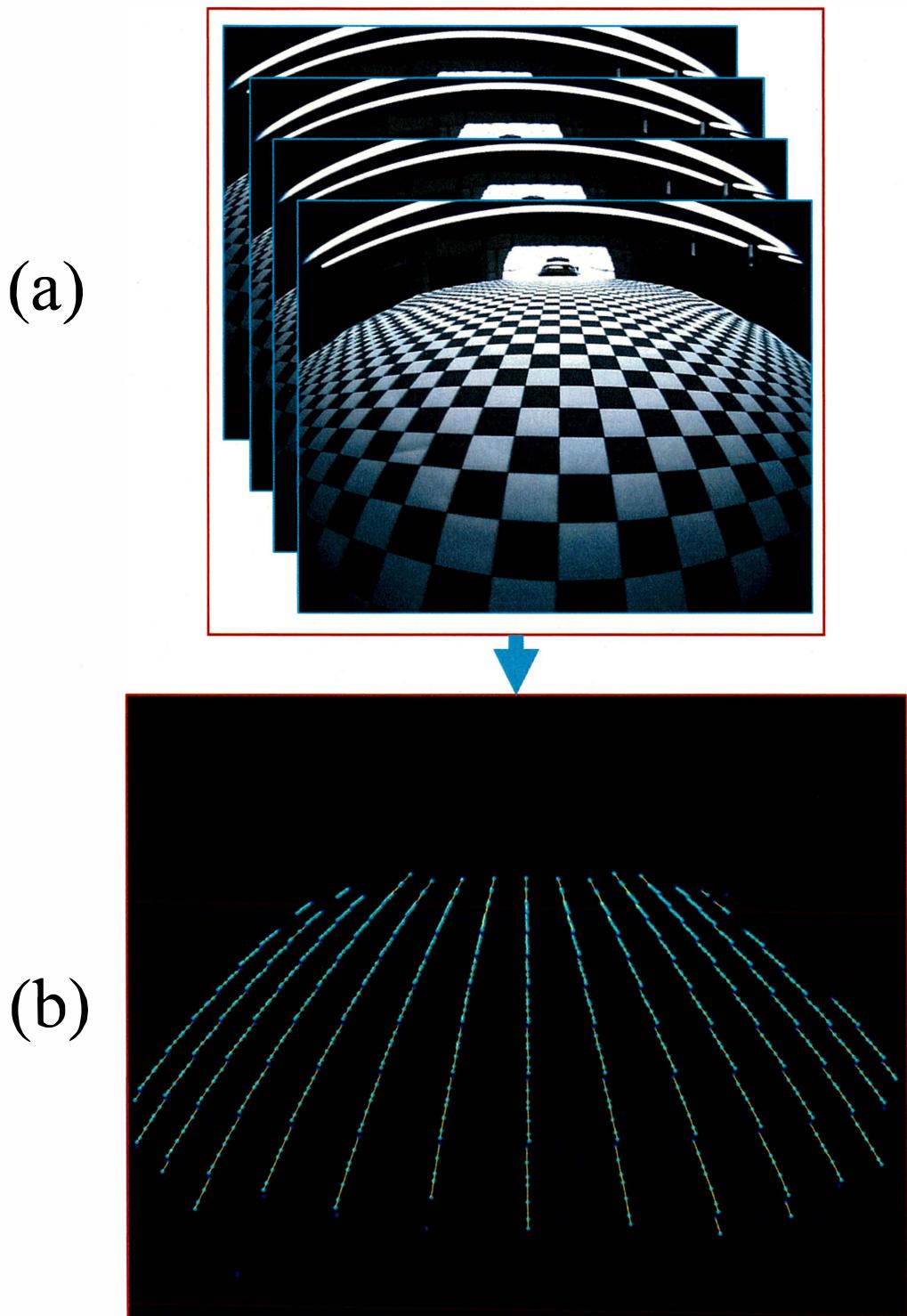


Figure 3.9: Reference line based on chessboard. (a) Chessboard images, (b) Reference line.

shown in Eq.3.14.

$$\begin{cases} \min = \sum_{i=1}^n (p(x_i) - y_i)^2 \\ p(x) = a_0 + a_1x + \dots + a_kx^k \end{cases} \quad (3.14)$$

To obtain condition a , the equation can be expanded and sorted as shown in Eq.3.15.

$$\begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{pmatrix} = \begin{pmatrix} n & \sum_{i=1}^n x_i & \dots & \sum_{i=1}^n x_i^k \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \dots & \sum_{i=1}^n x_i^{k+1} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^n x_i^k & \sum_{i=1}^n x_i^{k+1} & \dots & \sum_{i=1}^n x_i^{2k} \end{pmatrix}^{-1} \begin{pmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i y_i \\ \vdots \\ \sum_{i=1}^n x_i^k y_i \end{pmatrix} \quad (3.15)$$

However, there are two major problems in trajectory estimation. First, different degrees of the equation lead to different inference results. Even if the defect location is correctly detected, the filling on the inside of the defect moving trajectory is essentially the same, but the inference on the outside is completely different owing to the different degrees of the equation, as shown in Fig.3.10. Secondly, a different number of defects leads to different inference results. Even if the degree of the equation is fixed, the trajectory varies with the number of defects, as shown in Fig.3.11.

In this study, I solve this problem by simplifying the curve problem to a multi-segment straight line problem. Firstly, I make a new moving trajectory reference, as shown in Fig.3.12. According to the position of the LED tube in the picture, I divide the ROI into three areas. Further, according to Eq.3.16, the approximately

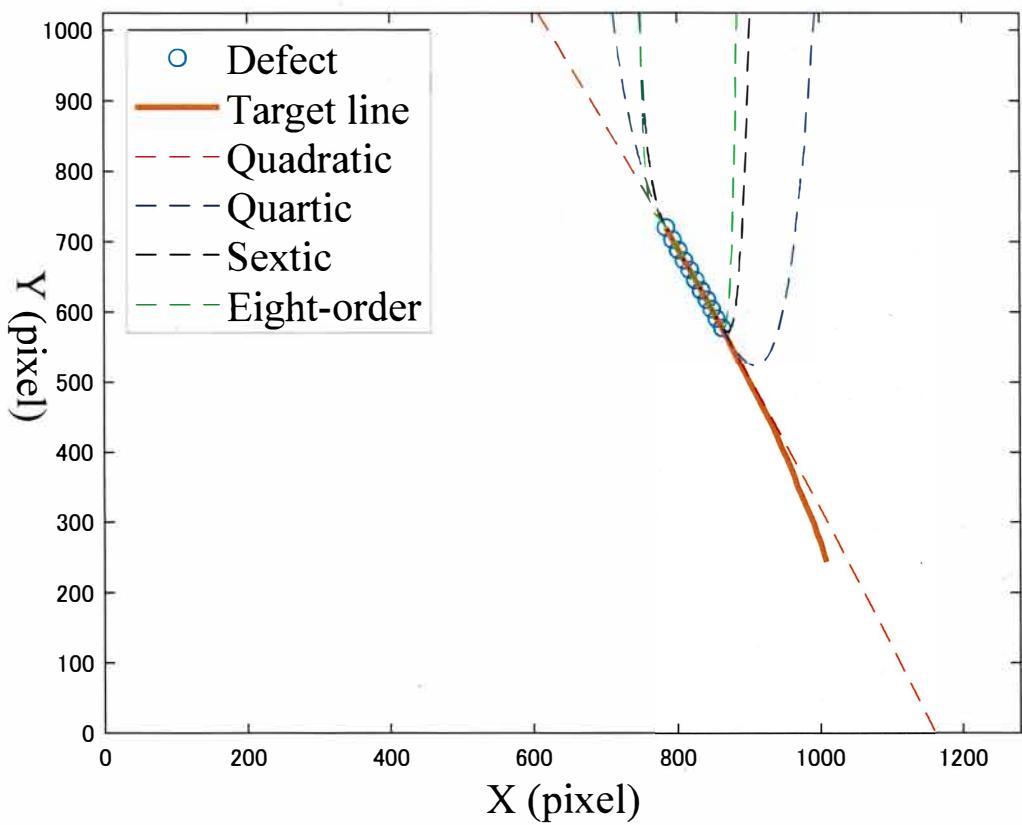


Figure 3.10: Trajectory estimation of equations with different degrees.

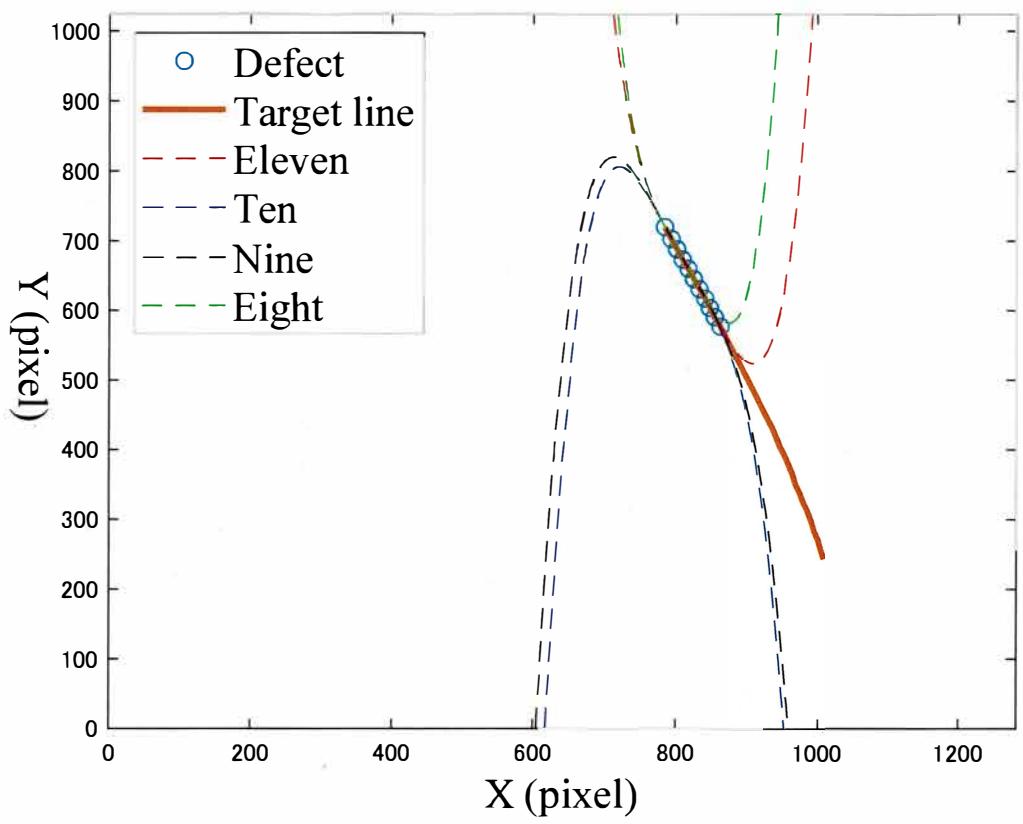


Figure 3.11: Trajectory estimation of different number of defects.

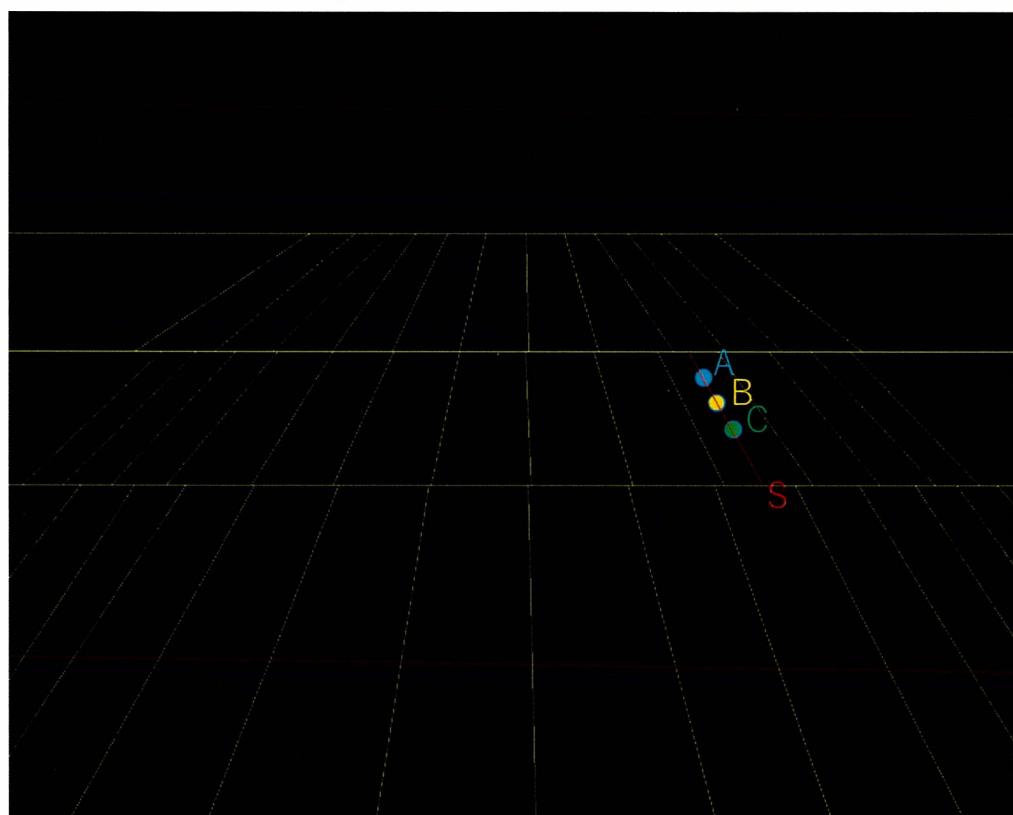


Figure 3.12: Verification of defect candidates based on trajectory.

straight lines of corners in the three areas of the checkerboard images are obtained.

$$\begin{cases} k = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i^2)} \\ b = \bar{y} - k\bar{x} \end{cases} \quad (3.16)$$

where k is the slope of the approximated straight line, b is the constant term of the approximate straight line, and \bar{x} and \bar{y} are the averages of the X and Y coordinates, respectively.

Then, the defect candidates are tracked, grouped and their approximate lines are calculated. By Eq.3.17, two defect candidates that meet the requirements in consecutive frames can be combined into one group. As shown in Fig.3.12, A and B, B and C can each be regarded as a group. When two groups are extending forward and backward of the same defect candidate, and the inclination angles of the approximate straight lines are similar, three consecutive candidate defects can be obtained by combining the two groups, as shown in Fig.3.12, A, B, and C. Their approximate straight line can be obtained by Eq.3.16.

$$\begin{cases} kV_{x_{max}} \geq |x_i - x_{i-k}| \geq kV_{x_{min}} \\ kV_{y_{max}} \geq y_i - y_{i-k} \geq kV_{y_{min}} \end{cases} \quad (3.17)$$

Here, k is the spacing between two frames. $V_{x_{min}}$ and $V_{x_{max}}$ are the minimum distance and maximum distance that can be moved on the X-axis per frame, similarly $V_{y_{min}}$ and $V_{y_{max}}$. The reference distance can be obtained from the checkerboard

data.

Finally, I compare it to the reference trajectory image. When the angle of this straight line is within the range defined by the reference line, it is determined that the defect candidate C is a defect existing on the inspected surface. Here, I judge by the intersection and inclination of them with the dividing line. As shown in Figure 13, the approximate straight line S intersects the dividing line W with an angle of α . When the angle α is between the angle β and β' formed by the reference line and the partition line on both sides, it is judged to be qualified.

Fabrication of Reference Line

The movement track used in the previous text is obtained through the corner detection of the chessboard. Compared with the marked center of gravity, the corner is more stable, as shown in the fig.4.1. Image corner detection is a classical research

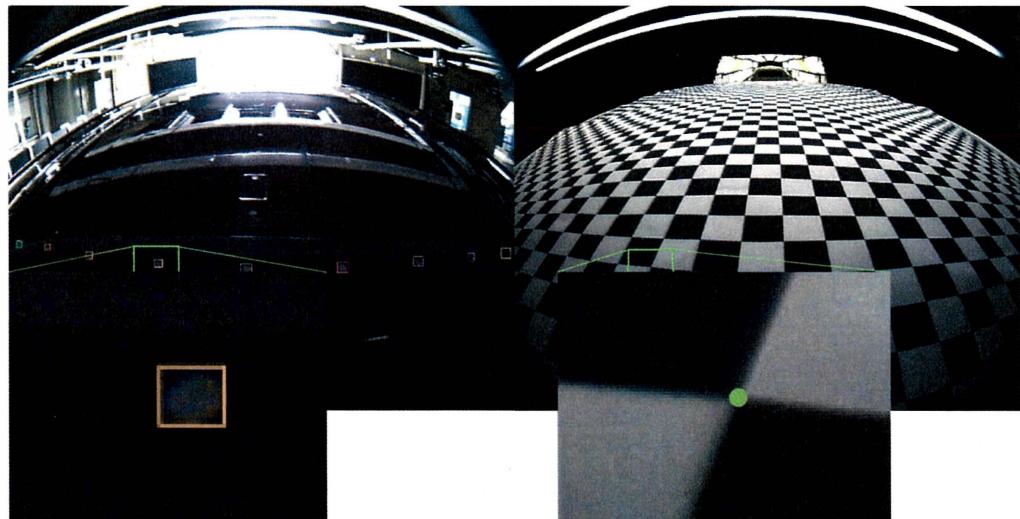


Figure 3.13: Mark points and checkerboards

direction in the field of image vision. In view of the different characteristics of different types of corners, checkerboard corners, as a special corner, are widely used

in camera calibration, and corner detection also plays a very important role in the whole process of camera calibration. At present, there are mainly corner detection based on image gray and corner detection based on image edge extraction. The latter one depends on the quality of image edge extraction, and its practicability is not very strong. Therefore, the most widely used detection method is based on the change of image gray curvature. The commonly used operators include Moravec, Harris, Forstner and Susan. Among them, Harris has good positioning performance and robustness. In addition, Harris algorithm has simple principle and is not affected by camera attitude and illumination. It has become one of the most widely used corner extraction algorithms. Therefore, many improved algorithms have been proposed for Harris algorithm: sub-pixel corner detection algorithm combined with scale space, adaptive corner extraction algorithm, detection algorithm proposed according to the geometric characteristics of chessboard, etc., but most of them are at the cost of increasing computation or low efficiency. At the same time, there are three deficiencies in using Harris algorithm for detection:

- The algorithm has no scale invariance;
- The corners extracted by the algorithm are pixel level;
- The detection time of the algorithm is not very satisfactory.

Generally, the corner detection of checkerboard can be divided into four steps.

In the first step, the local average adaptive thresholding method has strong adaptability to the uneven brightness. Therefore, after binarization and equalization of the image, the ideal threshold is obtained.

In the second step, the image expansion separates the connection of each black block quadrilateral. Because the expanded pixels are white pixels, the black block quadrilateral can be reduced and the connection can be broken.

The third step is to detect the quadrilateral, calculate the convex hull of each contour, polygon detection, and judge whether there are only four vertices. If so, it is a quadrilateral, and then remove some interference quadrilateral with constraints such as length width ratio, perimeter and area.

The fourth step is to take each quadrilateral as a unit, which has adjacent quadrilateral, interference quadrilateral without adjacent quadrilateral, two adjacent quadrilateral as boundary quadrilateral, and four adjacent quadrilateral as internal quadrilateral. The sequence number of each quadrilateral can be sorted according to the adjacent relationship, and then the middle point of its connecting line is taken as the corner point according to the two opposite points of the two diagonal quadrilaterals.

However, although this method is simple, there are the following problems in practical use:

- You need to specify the checkerboard size in advance. This is difficult to do in many automation applications.
- Poor robustness. If the chessboard has interference (such as slight occlusion), the detection will fail, and the detection will fail if the chessboard has a large inclination angle.
- Cannot handle incomplete chessboard.

As shown in the figure, what needs to be detected is an irregular and incomplete

chessboard. Therefore, a unique checkerboard detection method is needed to meet this requirement.

The proposal method is mainly divided into three steps.

The first step, locate the corner position of the chessboard.

Define two different corner prototypes. One is used for corner points parallel to the coordinate axis, and the other is used for corner points rotated by 45° . According to practical experience, it can be found that these two simple prototypes are sufficient for corner detection of large-scale deformation caused by perspective transformation. Each prototype consists of four filter cores a, B, C, D for convolution with the image.

Next, we use these two corner prototypes to calculate the corner likelihood of each pixel. Let's see the following definitions:

$$\left\{ \begin{array}{l} c = \max(s_1^1, s_1^2, s_2^1, s_2^2) \end{array} \right. \quad (3.18)$$

$$\left\{ \begin{array}{l} s_i^1 = \min(\min(f_A^i, f_B^i) - \mu, \mu - \min(f_C^i, f_D^i)) \end{array} \right. \quad (3.19)$$

$$\left\{ \begin{array}{l} s_i^2 = \min(\min(\mu - f_A^i, f_B^i), \min(f_C^i, f_D^i) - \mu) \end{array} \right. \quad (3.20)$$

$$\left\{ \begin{array}{l} \mu = 0.25(f_A^i + f_B^i + f_C^i + f_D^i) \end{array} \right. \quad (3.21)$$

Where f_A^i represents the convolution response of convolution kernel A and prototype $i(i = 1, 2)$ at a pixel. s_i^1 and s_i^2 represent two possible flippings of prototype I, that is, considering that the corner of the chessboard may appear, the left diagonal is black and the right diagonal is white; Or the left diagonal is white and the right diagonal is black. It is not difficult to find that if the response value of any of the four cores is small, the value C of corner likelihood will be very small. This feature is important to remove a large number of non chessboard corners. After the above calculation for each pixel, a corner likelihood graph is obtained.

The next step is to use non maximum suppression (NMS) to obtain candidate points on the corner likelihood graph. As the name suggests, non maximum suppression is to suppress elements that are not maxima and search for local maxima. This local represents a neighborhood. The neighborhood has two variable parameters, one is the dimension of the neighborhood, and the other is the size of the neighborhood. Here, I use NMS to select the maximum pixels with the highest scores in the neighborhood and suppress the pixels with low scores. Then, the gradient statistical method is used to verify these candidate points in a local $N \times n$ neighborhood. Firstly, the local gray image is Sobel filtered, and then the weighted direction histogram (32bins) is calculated. The mean shift algorithm is used to find the two main modes α_1 and α_2 . According to the direction of the edge, a template T is constructed for the desired gradient intensity $\|\nabla I\|_2$. The product of $T * \|\nabla I\|_2$ (* for cross-correlation operator) and corner likelihood is taken as the corner score, and then the corner is obtained by judging with the threshold. The results are shown in the fig.3.14.

The second step is the refinement of sub-pixel corners and directions.

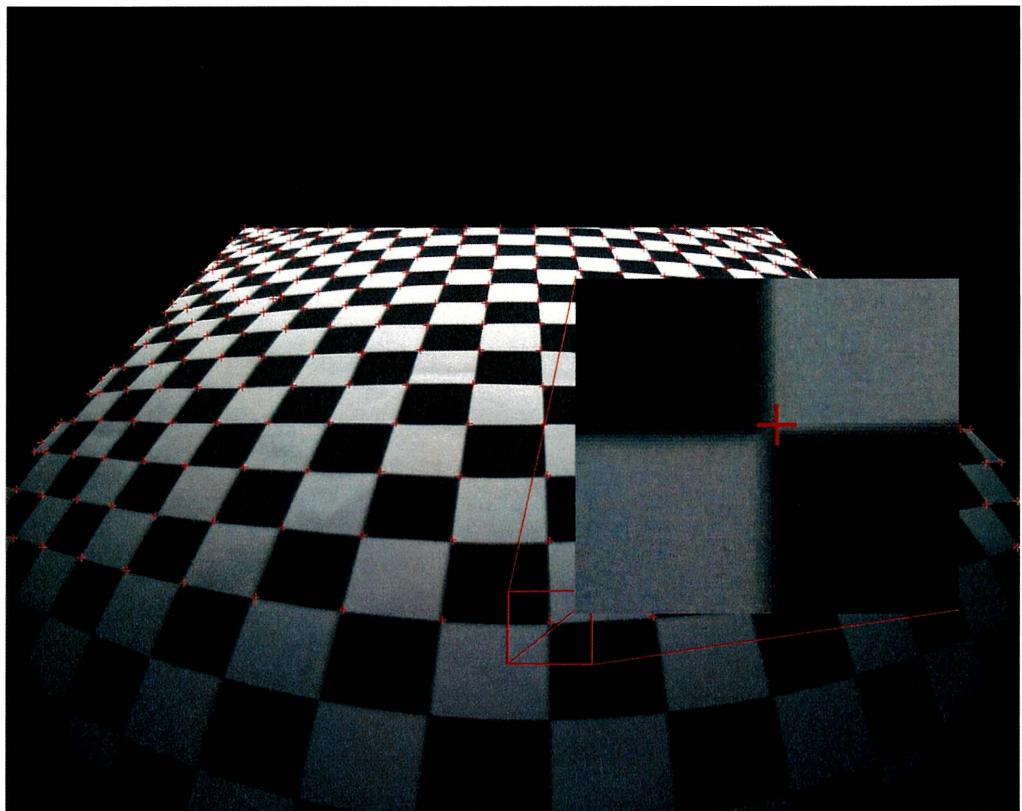


Figure 3.14: Corner detection result

The corner points obtained above are generally not very accurate, and sub-pixel refinement is required for the position and edge direction of the corner points. Suppose c is the ideal corner position, p is a pixel point in the local neighborhood of c , and g_p is the image gradient vector of point p , then $g_p^T (p - c) = 0$. Briefly explain the reason as shown in the figure: the green point in the center represents the ideal corner c . if the pixel p is not on the boundary, the following icon number is 1, and the gradient g_p in the flat area is the zero vector, so $g_p^T (p - c) = 0$. If the pixel p is on the boundary, At the position with the following icon number 2, the gradient g_p vector direction is vertically downward, $(p - c)$ direction is horizontally left, and the two vector directions are perpendicular to each other, so $g_p^T (p - c) = 0$. The above figure is an ideal checkerboard, but in fact, the edge cannot be so sharp (only one pixel size), and the gradient direction is not so ideal, The most ideal position of corner c is that we find c' satisfying the following formula in the neighborhood $N_I(c')$ of corner candidate c' :

$$\left\{ \begin{array}{l} c = \varphi \min \sum_{p \in N_1(c')} (g_p^T (p - c))^2 \end{array} \right. \quad (3.22)$$

The pixels in the neighborhood are automatically weighted by the gradient amplitude. The derivative of c' on the right of the above formula and make it equal to 0 can obtain the analytical solution:

$$\left\{ \begin{array}{l} c = \left(\sum_{p \in N_1} g_p g_p^T \right)^{-1} \sum_{p \in N_1} (g_p g_p^T) p \end{array} \right. \quad (3.23)$$

The next step is to refine the edge direction vectors $e1$ and $e2$, which can be achieved

by minimizing the deviation of normals and the corresponding gradient:

$$\left\{ \begin{array}{l} e_i = \varphi \min \sum_{p \in N_1(c')} (g_p^T e_i')^2 \end{array} \right. \quad (3.24)$$

$$\left\{ \begin{array}{l} M_i = \{p | p \in N_1 \wedge |m_i^T g_p| < 0.25\} \end{array} \right. \quad (3.25)$$

The result is shown in the fig.3.15.

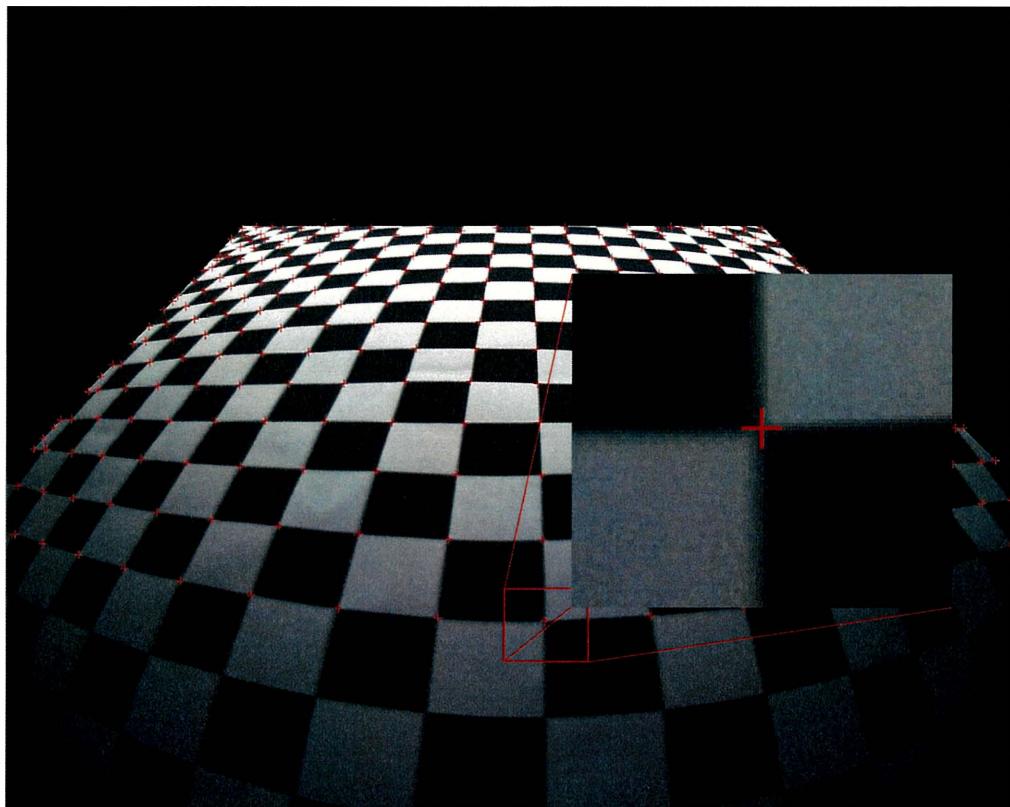


Figure 3.15: Corner refinement result

Last step, a reference track can be obtained by tracing the corner points.

Chapter 4

Experiments

4.1 Datasets

To detect the robustness of this method, I prepared two sets of data with different picture sizes. The first set of data is classified according to the type of paint color and defects, and the second set is classified according to the paint color and the car model.

4.1.1 Data Classified According to Paint Color and Defect Type

The first set of data is obtained by shooting a SUV roof with a camera with anti-flicker capability. The experimental video is 30 frames per second and the image size is 1280×1024 pixels. In particular, the actual width of the roof portion of the image is approximately 1000 mm, and the detection range of our method is 1 × 1 pixel to 40×40 pixels. Therefore, the detectable size of the defect of this method is within [1mm,40mm]. In order to evaluate the detection performance of the method, I photographed 15 cars with defective roofs. Each video data is 9 seconds long and

is Indicates the color of the paint and the type of defect, as shown in Table 4.1.

Fig.4.1 is a supplemental description of the color of the paint and the type of defect.

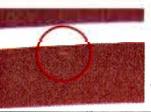
Painting color		Defect type	
Black		Dot	
Red		Line	
Blue		Cotton wool	
Original		Deep hole	
White			
Silver			

Figure 4.1: Paint color and defect type for the first set of data

Table 4.1: Data Classified According to Paint Color and Defect Type

No.	Color	#Total frames	Defect type	#Total defects
1	Black	270	Dot	1
2	Black	270	Cotton wool	1
3	Black	270	Cotton wool	3
4	Black	270	Dot + Cotton wool	3
5	Black	270	Dot + Cotton wool	11
6	Red	270	Cotton wool	2
7	Red	270	Dot	1
8	Red	270	Line	1
9	Blue	270	Dot	1
10	Blue	270	Dot	1
11	Blue	270	Dot + Cotton wool	3
12	Brown	270	Dot + Cotton wool	3
13	White	270	Dot	1
14	White	270	Dot	1
15	Silver	270	Deep hole	1

4.1.2 Data Classified According to Paint Color and Car Model

The second set of data was taken from 23 defective roofs. Each group of experimental data is 30 frames and 16 seconds video, and the image size is 1600×1200 pixels. In particular, the actual width of the roof portion of the image is about 1000mm, and our method detects a range of 1×1 pixel to 40×40 pixels. Therefore, the defect detection size of this method is within [0.6mm, 25mm]. The 23 roofs come in nine different colors from two different SUV models. As shown in the Fig.4.2 and Table 4.2.

Painting color	
White	
Metallic	
Black	
Red	
Brown	
Silver	
Blue	
Original	
Gray	

Figure 4.2: Paint color for the second set of data

Table 4.2: Data Classified According to Paint Color and Defect Type

No.	Color	#Total frames	Car model	#Total defects
1	Metallic	270	A	1
2	White	270	A	1
3	Black	270	A	2
4	Blue	270	A	1
5	Brown	270	A	1
6	Red	270	A	1
7	Blue	270	B	1
8	White	270	B	1
9	Gray	270	B	1
10	Black	270	B	2
11	Silver	270	B	2
12	Red	270	B	1
13	Original	270	B	1



- Defect area
- Ground truth bounding box
- Detected defect area and its center
- Marker for detected defect

Figure 4.3: Criteria of correct detection. (a) True-positive detection, (b) False-positive detection.

4.1.3 Ground Truth

Ground truth is created by placing a mark around a defect that appeared in the sequence data based on a visual inspection. When the coordinates of the center of gravity of the detected defects overlap with the coordinates of the label, it is assessed to be true-positive detection. When they do not, it is false-positive detection, as shown in Fig.4.3.

4.2 Validation of Proposal Method

Using the above data, the methods proposed in this paper are validated. The validation experiment is divided into three parts.

4.2.1 Validation of Differential Processing Method

First, the experiment was performed using the first set of data. In the step of detecting defect candidates, in order to include all possible candidates, I made the following choices related to the detection method and ROI. First, the traditional binarization method is chosen as the main detection method, which is based on the change of intensity of the image, to avoid the omission of detection caused by other influences such as color. Secondly, through observation of the experimental video, we found that some defects are easier to observe in the bright field of vision environment, and others are easier to observe in the dark field of vision environment. Therefore, we set the ROI as the LED tube reflection area and its surroundings to ensure that all defects can be detected. The defect candidates detected were analyzed, which included true-positive deflections and a large number of false-positive

defections. Since each true defect occurs many times in a video data, it can be regarded as detected when it is detected more than once. As can be obtained from Table 4.3, although the accuracy is very low, every true defect is detected. In particular, the recall is the number of true detected defects divided by the total number of defects. The precision is the number of times of true detection divided by the total number of times it was detected. The F-measure score is calculated by the harmonic mean of precision and recall.

Table 4.3: Analysis of defect candidates

No.	#Total defects	Recall (%)	Precision (%)	F-measure (%)
1	1	100.00	0.42	0.84
2	1	100.00	0.97	1.93
3	3	100.00	0.59	1.18
4	3	100.00	0.42	0.84
5	11	100.00	4.08	7.85
6	2	100.00	0.32	0.63
7	1	100.00	0.41	0.82
8	1	100.00	0.47	0.94
9	1	100.00	0.25	0.50
10	1	100.00	0.19	0.39
11	3	100.00	0.31	0.61
12	3	100.00	1.10	2.17
13	1	100.00	0.12	0.25
14	1	100.00	0.15	0.29
15	1	100.00	0.40	0.81

In the next step of defect candidate verification in a single frame, I use a multi-directional differential image to verify defect candidates. The results of verification are shown in Table 4.4. While keeping the Recall from decreasing, the precision has increased significantly, indicating that this method eliminates a large number of false detection defects. Further, by observing Table 4.4, I can find that the results of the coating color are black, red, and brown is much better than blue, white and silver. I analyzed the experimental data and concluded that it was due to the different

reflectivity of different colors. This phenomenon causes the reflected light to be more concentrated in the imaging of the black-coated surface under the illumination of the LED tube. Conversely, in the imaging of a white painted surface, the reflected light is more dispersed. As shown in Fig.4.4. This produces more noise on the painted side of the highly reflective color. Failure example is shown in Fig.4.5.

Table 4.4: Analysis of Verification Results Based on Single Frame

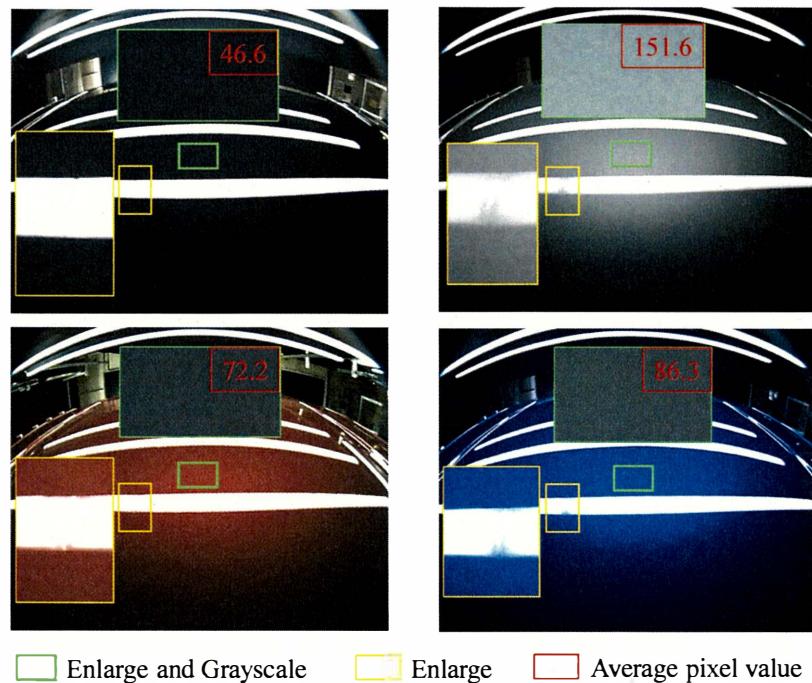
No.	#Total defects	Recall (%)	Precision (%)	F-measure (%)
1	1	100.00	41.07	58.23
2	1	100.00	39.44	56.57
3	3	100.00	26.92	42.42
4	3	100.00	25.26	40.33
5	11	100.00	85.07	91.94
6	2	100.00	66.67	80.00
7	1	100.00	93.55	96.67
8	1	100.00	88.00	93.62
9	1	100.00	2.62	5.11
10	1	100.00	3.36	6.51
11	3	100.00	5.71	10.81
12	3	100.00	25.14	40.18
13	1	100.00	0.90	1.78
14	1	100.00	0.76	1.50
15	1	100.00	1.64	3.22

The results of the difference methods are evaluated according to the different colors of the coating. The results are shown in the Table 4.5 - 4.11. Part of the non-checkout in the table is due to connectivity issues, as shown in the Fig.4.6.

Table 4.5: Evaluation results for white coatings

Brightness difference	#Total defects	#Detect defects	#Undetected defects
0 ~ 9	1	0	1
10 ~ 19	42	2	40
20 ~ 29	45	36	9
30 ~ 39	35	30	5
40 ~ 49	38	27	11
50 ~ 59	27	18	9
60 ~ 69	19	11	8
70 ~ 79	21	6	15
80 ~ 89	9	2	7
90 ~ 100	8	3	5
100 ~ 255	5	4	1

The differential filter used in the above experiments is shown in the fig.4.14. At the same time, six other groups of differential filters are used for comparison



□ Enlarge and Grayscale □ Enlarge □ Average pixel value

Figure 4.4: Reflective effects of different colors.

Table 4.6: Evaluation results for black coatings

Brightness difference	#Total defects	#Detect defects	#Undetected defects
0 ~ 9	0	0	0
10 ~ 19	71	55	16
20 ~ 29	178	146	32
30 ~ 39	58	44	14
40 ~ 49	47	43	14
50 ~ 59	29	25	4
60 ~ 69	23	21	2
70 ~ 79	20	15	5
80 ~ 89	23	22	1
90 ~ 100	15	11	4
100 ~ 255	126	69	57

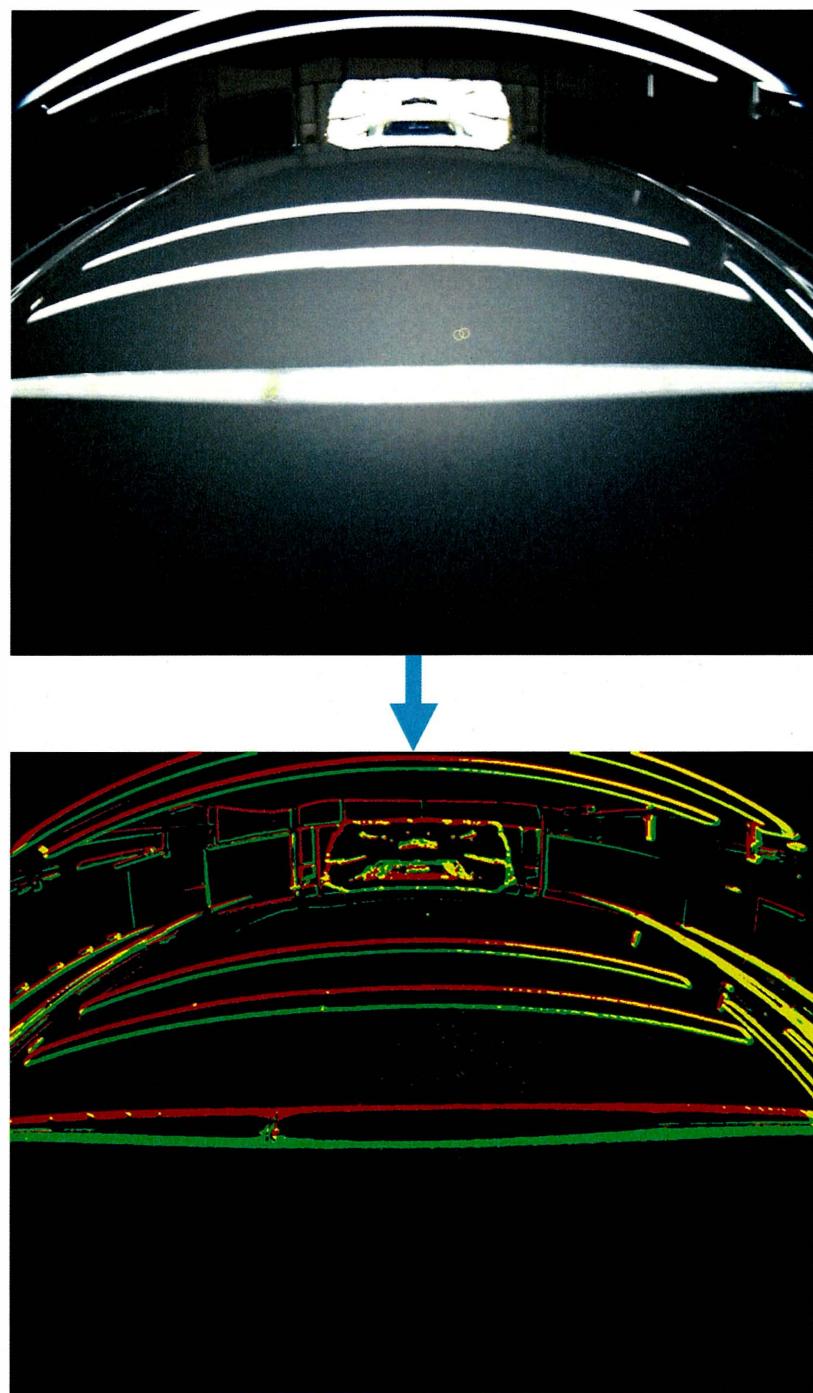


Figure 4.5: Failure example.

Table 4.7: Evaluation results for silver coatings

Brightness difference	#Total defects	#Detect defects	#Undetected defects
0 ~ 9	0	0	0
10 ~ 19	20	5	15
20 ~ 29	42	35	7
30 ~ 39	27	15	12
40 ~ 49	17	13	4
50 ~ 59	13	4	9
60 ~ 69	12	7	5
70 ~ 79	12	2	10
80 ~ 89	6	3	3
90 ~ 100	4	1	3
100 ~ 255	8	4	4

Table 4.8: Evaluation results for grey coatings

Brightness difference	#Total defects	#Detect defects	#Undetected defects
0 ~ 9	0	0	0
10 ~ 19	6	1	5
20 ~ 29	2	0	2
30 ~ 39	7	5	2
40 ~ 49	4	4	0
50 ~ 59	4	3	1
60 ~ 69	1	1	0
70 ~ 79	3	3	0
80 ~ 89	1	1	0
90 ~ 100	2	0	2
100 ~ 255	6	0	6

Table 4.9: Evaluation results for red coatings

Brightness difference	#Total defects	#Detect defects	#Undetected defects
0 ~ 9	0	0	0
10 ~ 19	4	1	0
20 ~ 29	7	5	2
30 ~ 39	5	4	1
40 ~ 49	6	6	0
50 ~ 59	10	8	2
60 ~ 69	9	9	0
70 ~ 79	6	4	2
80 ~ 89	9	7	2
90 ~ 100	18	15	3
100 ~ 255	73	56	17

Table 4.10: Evaluation results for blue coatings

Brightness difference	#Total defects	#Detect defects	#Undetected defects
0 ~ 9	1	0	1
10 ~ 19	30	3	27
20 ~ 29	25	14	11
30 ~ 39	23	23	0
40 ~ 49	18	17	1
50 ~ 59	17	15	2
60 ~ 69	14	10	4
70 ~ 79	11	7	4
80 ~ 89	10	4	6
90 ~ 100	2	0	2
100 ~ 255	20	9	1

Table 4.11: Evaluation results for orange coatings

Brightness difference	#Total defects	#Detect defects	#Undetected defects
0 ~ 9	0	0	0
10 ~ 19	3	3	0
20 ~ 29	4	3	1
30 ~ 39	1	0	1
40 ~ 49	0	0	0
50 ~ 59	1	0	1
60 ~ 69	0	0	0
70 ~ 79	1	0	1
80 ~ 89	1	0	1
90 ~ 100	0	0	0
100 ~ 255	0	0	0

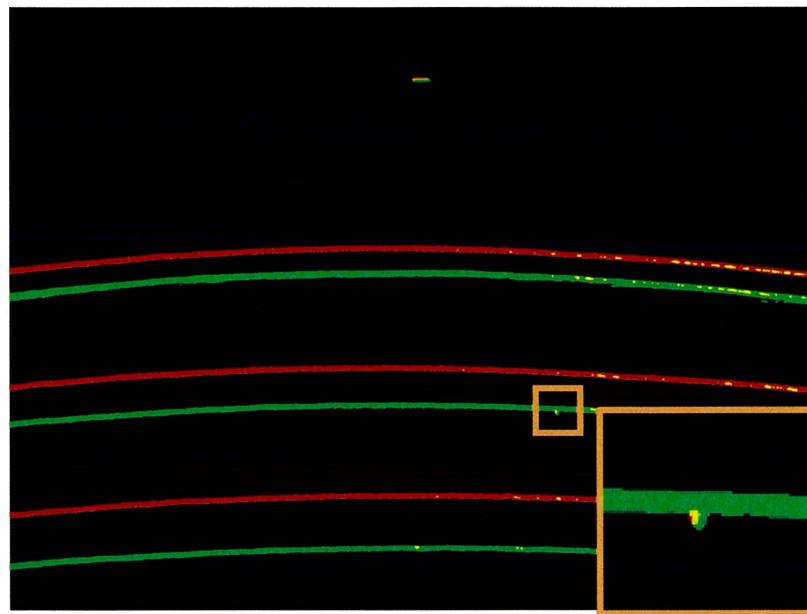


Figure 4.6: Examples of failures due to connection problems.

<table border="1"><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	1	0	-1	0	0	0	<table border="1"><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	-1	0	1	0	0	0
0	0	0																	
1	0	-1																	
0	0	0																	
0	0	0																	
-1	0	1																	
0	0	0																	
Left	Right																		
<table border="1"><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>-1</td><td>0</td></tr></table>	0	1	0	0	0	0	0	-1	0	<table border="1"><tr><td>0</td><td>-1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr></table>	0	-1	0	0	0	0	0	1	0
0	1	0																	
0	0	0																	
0	-1	0																	
0	-1	0																	
0	0	0																	
0	1	0																	
Above	Below																		

Figure 4.7: The differential filter.

<table border="1"> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>-1</td></tr> </table>	1	0	0	0	0	0	0	0	-1	<table border="1"> <tr><td>-1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> </table>	-1	0	0	0	0	0	0	0	1
1	0	0																	
0	0	0																	
0	0	-1																	
-1	0	0																	
0	0	0																	
0	0	1																	
Upper left	Lower right																		
<table border="1"> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>0</td><td>0</td></tr> </table>	0	0	1	0	0	0	-1	0	0	<table border="1"> <tr><td>0</td><td>0</td><td>-1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> </table>	0	0	-1	0	0	0	1	0	0
0	0	1																	
0	0	0																	
-1	0	0																	
0	0	-1																	
0	0	0																	
1	0	0																	
Upper right	Lower left																		

Figure 4.8: Differential filter No.1 for comparative experiments.

<table border="1"> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	-1	0	0	1	0	0	0	0	<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> </table>	0	0	0	0	1	0	0	-1	0
0	-1	0																	
0	1	0																	
0	0	0																	
0	0	0																	
0	1	0																	
0	-1	0																	
Below	Above																		
<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	-1	1	0	0	0	0	<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>-1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	1	-1	0	0	0
0	0	0																	
-1	1	0																	
0	0	0																	
0	0	0																	
0	1	-1																	
0	0	0																	
Right	Left																		

Figure 4.9: Differential filter No.2 for comparative experiments.

Above	Below																		
<table border="1"> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	1	0	0	-1	0	0	0	0	<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> </table>	0	0	0	0	-1	0	0	1	0
0	1	0																	
0	-1	0																	
0	0	0																	
0	0	0																	
0	-1	0																	
0	1	0																	
Left	Right																		
<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	1	-1	0	0	0	0	<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	-1	1	0	0	0
0	0	0																	
1	-1	0																	
0	0	0																	
0	0	0																	
0	-1	1																	
0	0	0																	

Figure 4.10: Differential filter No.3 for comparative experiments.

Below	Above																		
<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> </table>	0	0	0	0	-1	0	0	1	0	<table border="1"> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	1	0	0	-1	0	0	0	0
0	0	0																	
0	-1	0																	
0	1	0																	
0	1	0																	
0	-1	0																	
0	0	0																	
Right	Left																		
<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	-1	1	0	0	0	<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	1	-1	0	0	0	0
0	0	0																	
0	-1	1																	
0	0	0																	
0	0	0																	
1	-1	0																	
0	0	0																	

Figure 4.11: Differential filter No.4 for comparative experiments.

-1	-2	-1
0	0	0
1	2	1

Below

1	2	1
0	0	0
-1	-2	-1

Above

-1	0	1
-2	0	2
-1	0	1

Right

1	0	-1
2	0	-2
1	0	-1

Left

Figure 4.12: Differential filter No.5 for comparative experiments.

-1	-1	-1
0	0	0
1	1	1

Below

1	1	1
0	0	0
-1	-1	-1

Above

-1	0	1
-1	0	1
-1	0	1

Right

1	0	-1
1	0	-1
1	0	-1

Left

Figure 4.13: Differential filter No.6 for comparative experiments.

experiments. The experimental results are shown in the table 4.12 - 4.18. The results show that either differential filter is not ideal for white coating.

Table 4.12: Results of experimental differential filters

Colour	#Total defects	#Detect defects	#Error detected
White	36	4	79
Silvery	54	18	225
Black	294	51	18
Red	30	14	13
Blue	179	81	16
Bright	27	11	62
Brown	107	8	39

Table 4.13: Results of Differential Filter No.1 for Experimentation

Colour	#Total defects	#Detect defects	#Error detected
White	36	4	21
Silvery	54	16	1226
Black	294	49	18
Red	30	8	10
Blue	179	74	38
Bright	27	8	294
Brown	107	39	7

4.2.2 Validation of Moving Trajectory Method

Using the newly acquired defect candidate, the continuity of the defects between multiple frames is used to validate the candidate defects in the step of using multiple frames to validate them. I employed previously proposed methods [44] that use the defect detection results of the previous frame to predict the defect location in the current frame, and I compared these results with the defect location of detection to verify the correctness of the detection results. The detection results are shown

Table 4.14: Results of Differential Filter No.2 for Experimentation

Colour	#Total defects	#Detect defects	#Error detected
White	36	1	267
Silvery	54	17	7
Black	294	20	6
Red	30	10	18
Blue	179	50	14
Bright	27	5	16
Brown	107	42	3

Table 4.15: Results of Differential Filter No.3 for Experimentation

Colour	#Total defects	#Detect defects	#Error detected
White	36	0	225
Silvery	54	17	13
Black	294	14	6
Red	30	10	13
Blue	179	41	9
Bright	27	3	27
Brown	107	41	3

Table 4.16: Results of Differential Filter No.4 for Experimentation

Colour	#Total defects	#Detect defects	#Error detected
White	36	2	66
Silvery	54	18	15
Black	294	49	18
Red	30	13	12
Blue	179	73	12
Bright	27	8	45
Brown	107	44	5

Table 4.17: Results of Differential Filter No.5 for Experimentation

Colour	#Total defects	#Detect defects	#Error detected
White	36	1	89
Silvery	54	18	5
Black	294	29	8
Red	30	11	12
Blue	179	55	9
Bright	27	6	27
Brown	107	40	3

Table 4.18: Results of Differential Filter No.6 for Experimentation

Colour	#Total defects	#Detect defects	#Error detected
White	36	4	39
Silvery	54	18	332
Black	294	46	8
Red	30	10	9
Blue	179	75	16
Bright	27	11	64
Brown	107	42	6

Table 4.19: Analysis of verification results based on moving distance

No.	#Total defects	Recall (%)	Precision (%)	F-measure (%)
1	1	100.00	100.00	100.00
2	1	100.00	100.00	100.00
3	3	66.67	100.00	80.00
4	3	66.67	85.71	75.00
5	11	100.00	100.00	100.00
6	2	50.00	100.00	66.67
7	1	100.00	100.00	100.00
8	1	100.00	100.00	100.00
9	1	100.00	36.36	53.33
10	1	100.00	50.00	66.67
11	3	100.00	64.71	78.57
12	3	66.67	85.71	75.00
13	1	100.00	5.71	10.81
14	1	0.00	0.00	-
15	1	0.00	0.00	-

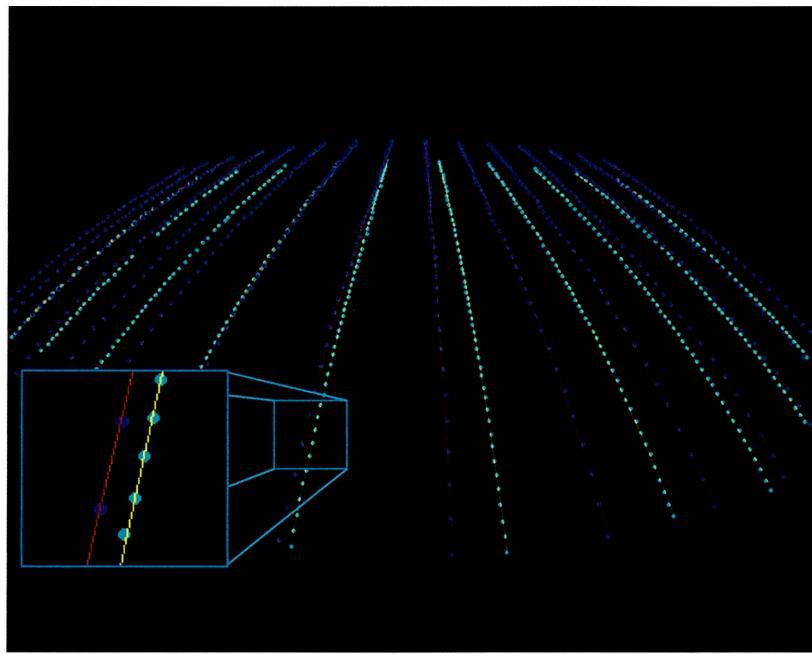


Figure 4.14: Different defect detection results due to unstable moving speed.

in Table 4.19. While many false-positive detections are removed, true-positive detections are also removed. By analyzing the results, I find that this method requires a very stable moving speed to ensure that the moving distance of defects in each frame remains unchanged, which requires a powerful hardware environment. The movement speed is slightly changed, and the movement of defects between frames is more than twice the error (as shown in Fig.4.14).

Therefore, I take the motion trajectory with high stability as the basis of detection, and the detection results are shown in Table 4.20. Compared to Table 4.19, this method has higher recall and precision. By comparing with Table 4.4, it can be known that the higher the precision of the candidate detected in a single frame (the less the false detection is), the higher the final precision. In addition, the recall of data numbered 4 declined. The reason is that the true defect is not detected in the defect candidate for three or more consecutive frames. Moreover, as shown in the

Table 4.20: Analysis of verification results based on moving trajectory

No.	#Total defects	Recall (%)	Precision (%)	F-measure (%)	Time (ms)
1	1	100.00	100.00	100.00	154.58
2	1	100.00	100.00	100.00	155.05
3	3	100.00	100.00	100.00	150.62
4	3	66.67	100.00	80.00	156.87
5	11	100.00	100.00	100.00	166.86
6	2	100.00	100.00	100.00	157.60
7	1	100.00	100.00	100.00	162.16
8	1	100.00	100.00	100.00	159.07
9	1	100.00	36.84	53.84	166.52
10	1	100.00	62.50	76.92	167.40
11	3	100.00	65.85	79.41	161.80
12	3	100.00	92.86	96.29	156.85
13	1	100.00	9.30	17.02	171.08
14	1	100.00	33.33	50.00	166.30
15	1	100.00	14.55	25.40	161.21

table 4.20, each frame takes about 155 ms. According to the length of the video, the approximate time cost can be calculated. The time cost of the video used in this experiment is about 41.85 seconds.

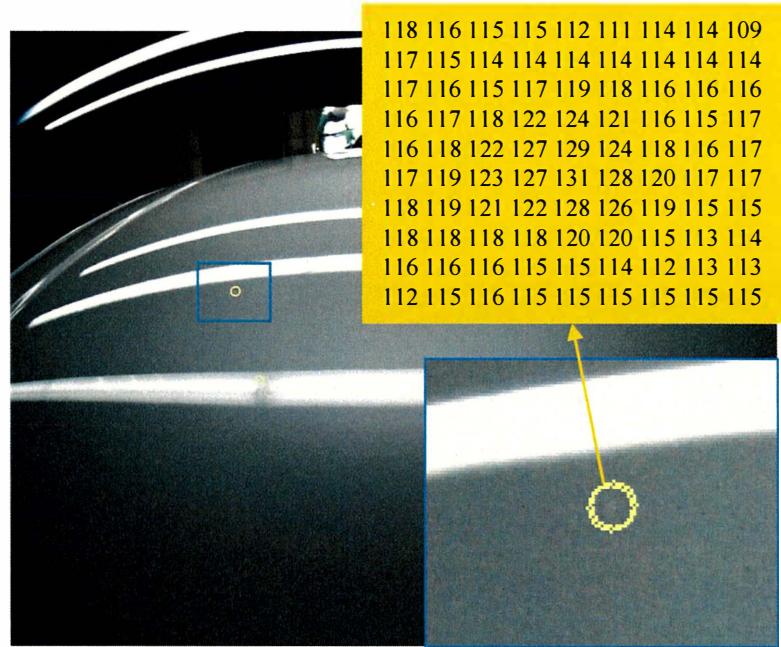
Further, the experimental results show that there are two major problems in this method. First, when a coating color having high light reflectivity, there are many false detections that are difficult to be removed. The reason for the analysis is mainly that the difference between the false defects caused by the diffuse reflection and the true defects is small and difficult to be distinguished. As shown in Fig.4.15, their grayscale distribution is very close. Second, Fig.4.16-f shows that some false-positive detections can also satisfy the requirements of the trajectory by coincidence, thus escaping the screening. This proves that our method also requires more strict criteria, such as more inter-frame comparison and more precise motion trajectory criteria. However, with more strict standard setting, the possibility of real detection being eliminated will also increase, so further experiments are required to

determine reasonable standards.

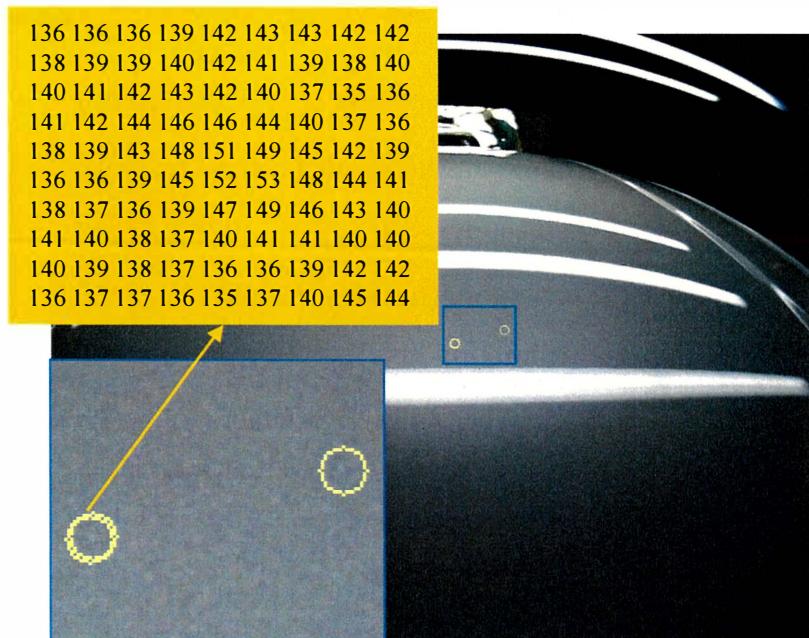
4.2.3 Validation of Exploring Binarization Method

Validation using the second set of data. For the validity of the proposed binarization method, we compare it with other binarization methods. The comparison methods are global binarization (Otsu, Triangle) and adaptive threshold (Mean, Gaussian). Local binarization (Niblack [39], Bernsen [40]) produces a lot of pseudo-noise, so it is not included in the comparison experiment. The results are shown in Table 4.21. In particular, the F-measure score of candidates is calculated by the harmonic mean of precision and recall. The recall is the number of defect candidates actually detected divided by the total number. The precision is the number of true candidate detections divided by the total number of times it was detected. Processing time is the average time consumed per frame. The best results are marked in red in the table. As shown in the results, compared with other binarization, this method is slow, but its detection accuracy is higher.

Finally, we add the multi-frame verification method to get the final detection results. The results are shown in Table 4.224.23. In particular, the F-measure score of defects is calculated similarly to the F-measure score of candidates, and the value used changes from the candidate correlation to the defect correlation. The experimental results show that this is an effective defect detection technique, and excellent results can still be obtained for the difficult-to-detect white coatings.



(a)



(b)

Figure 4.15: Gray value between true and false defects. (a) True defect, (b) False defect.

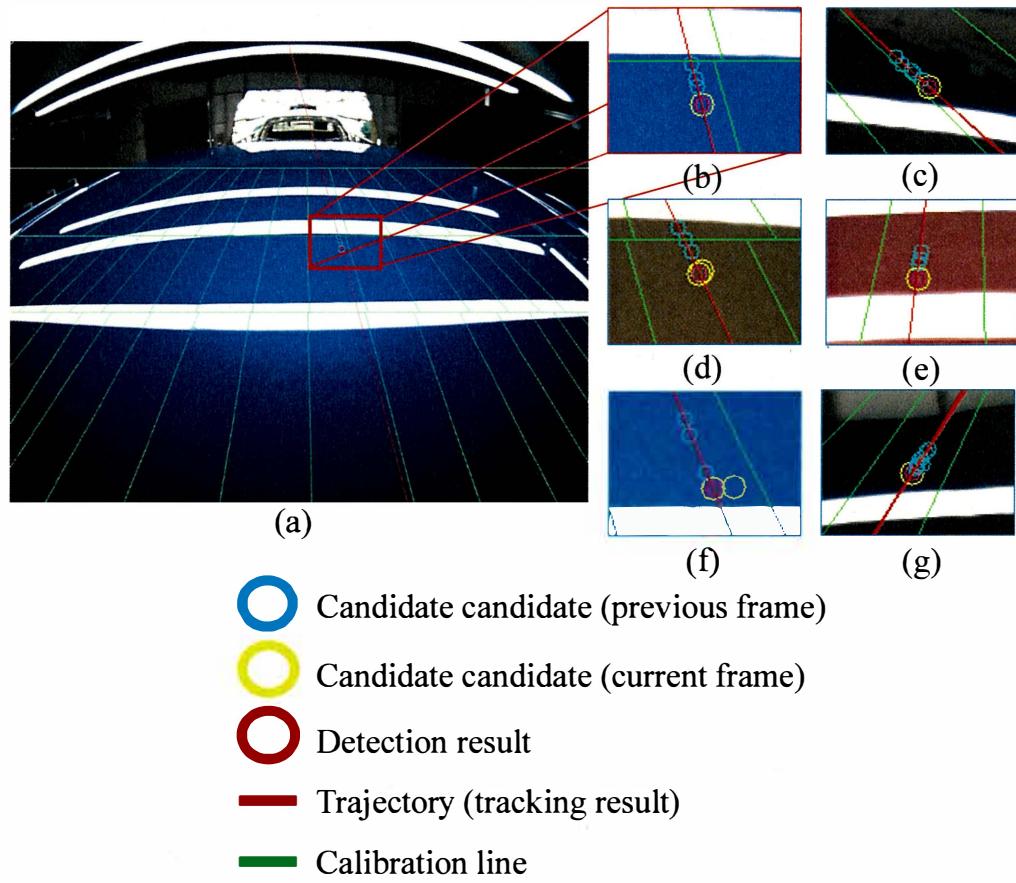


Figure 4.16: Detection results. (a) Full display of detection results, (b) Expanded display of Defects in (a), (c)(d)(e)(g) Expanded display of true-positive detection results of different colors, (f) Expanded display of false-positive Detection.

Table 4.21: F-measure of candidates (%)

Model	Color	Otsu	Triangle	ATM	ATG	Proposal
A	Metallic	82.5	75.9	92.0	88.7	93.0
A	White	34.3	0.0	23.6	26.0	34.9
A	Black	65.7	60.5	69.8	68.5	71.5
A	Blue	67.5	29.7	67.5	68.5	72.8
A	Brown	63.5	52.8	74.6	73.4	74.2
A	Red	62.3	52.3	70.0	66.9	70.8
B	Blue	58.3	17.8	64.5	46.5	67.0
B	White	26.0	26.0	16.4	13.6	40.1
B	Gray	59.7	0.0	58.8	58.8	62.3
B	Black	82.6	53.3	85.9	83.0	87.0
B	Silver	29.0	0.0	16.9	11.8	41.2
B	Red	67.9	60.3	62.8	59.0	76.9
B	Original	77.7	60.2	86.7	84.3	98.8

Adaptive Threshold Mean (ATM), Adaptive Threshold Gaussian (ATG)

Table 4.22: Processing time (ms)

Model	Color	Otsu	Triangle	ATM	ATG	Proposal
A	Metallic	22	19	3235	2485	326
A	White	115	120	8420	10121	362
A	Black	20	19	4207	2754	239
A	Blue	20	20	3023	2083	281
A	Brown	19	23	2853	2399	310
A	Red	20	22	1899	1251	231
B	Blue	22	22	3449	2256	308
B	White	178	318	11222	12980	509
B	Gray	21	21	1712	1153	265
B	Black	20	25	3790	2680	253
B	Silver	75	134	4567	4466	269
B	Red	20	22	858	639	264
B	Original	22	20	3495	2371	238

Adaptive Threshold Mean (ATM), Adaptive Threshold Gaussian (ATG)

Table 4.23: F-measure of defects (%)

Model	Color	Otsu	Triangle	ATM	ATG	Proposal
A	Metallic	100.0	100.0	100.0	100.0	100.0
A	White	33.0	0.0	51.4	50.9	100.0
A	Black	100.0	93.8	100.0	100.0	100.0
A	Blue	100.0	50.0	100.0	100.0	100.0
A	Brown	100.0	100.0	100.0	100.0	100.0
A	Red	100.0	100.0	100.0	100.0	100.0
B	Blue	100.0	33.3	100.0	66.7	100.0
B	White	50.0	50.0	52.6	52.7	100.0
B	Gray	100.0	0.0	100.0	100.0	100.0
B	Black	100.0	75.0	100.0	100.0	100.0
B	Silver	50.0	0.0	55.5	54.6	100.0
B	Red	100.0	100.0	100.0	100.0	100.0
B	Original	100.0	100.0	100.0	100.0	100.0

Adaptive Threshold Mean (ATM), Adaptive Threshold Gaussian (ATG)

Chapter 5

Conclusion

In this study, a system for automatically detecting defects in coated surfaces is presented. The system can be applied to the end of a painting project in automobile production engineering. The hardware of the system consists of only one LED tube, a camera, and a computer, making it a low-cost system. At the same time, the installation takes less space because of the small hardware composition. On the other hand, the software part of the system, through three steps to achieve the screening and detection of defects. They are detection of defect candidates, verification of defect candidates based on a single frame, and verification of defect candidates based on consecutive frames.

The detection of defect candidates is mainly based on exploring binarization methods. Experiments show that compared with the traditional global binarization and residence binarization, this binarization can detect all defects and produce the least noise.

In the defect candidate verification based on a single frame, two proposed methods are experimented. Firstly, the differential processing method is validated, and

six different combinations of filters are used for experiments. Experiments show that the differential treatment method is not effective in detecting white coatings, and can be applied to defect detection of coatings with colors other than white. Secondly, the brightness difference method is experimented. The experimental results show that this method can be applied to all colors of the coating, and the effect of noise removal will not be different for different colors.

Finally, the detected defects are verified several times based on the multi-frame motion track data, which effectively reduces the occurrence of false detection. Compared with the traditional motion distance-based method, this method is more stable and does not require much relative motion speed for the camera and the measured surface. However, the best parameters of this method depend on the brightness difference between the defect and noise in the photographic environment. When the brightness difference of the defect is low, it is difficult to detect the noise. In the correction of detection accuracy, the optimization of photographic mode is especially important. In addition, changes in photography may result in changes in motion trajectories. Therefore, other forms of motion tracks need to be examined to propose a more robust classification method.

Bibliography

- [1] Q. Luo, X. Fang, L. Liu, C. Yang, and Y. Sun, “Automated visual defect detection for flat steel surface: A survey,” *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 3, pp. 626–644, 2020.
- [2] N. Neogi, D. K. Mohanta, and P. K. Dutta, “Review of vision-based steel surface inspection systems,” *EURASIP Journal on Image and Video Processing*, vol. 2014, no. 1, pp. 1–19, 2014.
- [3] C. Yang, P. Liu, G. Yin, H. Jiang, and X. Li, “Defect detection in magnetic tile images based on stationary wavelet transform,” *Ndt & E International*, vol. 83, pp. 78–87, 2016.
- [4] Y. Huang, C. Qiu, and K. Yuan, “Surface defect saliency of magnetic tile,” *The Visual Computer*, vol. 36, no. 1, pp. 85–96, 2020.
- [5] W. Cao, Q. Liu, and Z. He, “Review of pavement defect detection methods,” *IEEE Access*, vol. 8, pp. 14 531–14 544, 2020.
- [6] O. O. Arjenaki, P. A. Moghaddam, and A. M. Motlagh, “Online tomato sorting based on shape, maturity, size, and surface defects using machine vision,” *Turkish Journal of Agriculture and Forestry*, vol. 37, no. 1, pp. 62–68, 2013.

- [7] S. Radovan, G. D. Papadopoulos, M. Georgoudakis, and P. Mitropoulos, “Vision system for finished fabric inspection,” in *Machine Vision Applications in Industrial Inspection X*, M. A. Hunt, Ed., vol. 4664, International Society for Optics and Photonics. SPIE, 2002, pp. 97 – 103. [Online]. Available: <https://doi.org/10.1117/12.460200>
- [8] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, “Textural features for image classification,” *IEEE Transactions on systems, man, and cybernetics*, no. 6, pp. 610–621, 1973.
- [9] A. Baraldi and F. Pannigiani, “An investigation of the textural characteristics associated with gray level cooccurrence matrix statistical parameters,” *IEEE transactions on geoscience and remote sensing*, vol. 33, no. 2, pp. 293–304, 1995.
- [10] F. T. Ulaby, F. Kouyate, B. Brisco, and T. L. Williams, “Textural information in sar images,” *IEEE Transactions on Geoscience and Remote Sensing*, no. 2, pp. 235–245, 1986.
- [11] C. Jian, J. Gao, and Y. Ao, “Automatic surface defect detection for mobile phone screen glass based on machine vision,” *Applied Soft Computing*, vol. 52, pp. 348–358, 2017.
- [12] P. Li, J. Liang, X. Shen, M. Zhao, and L. Sui, “Textile fabric defect detection based on low-rank representation,” *Multimedia Tools and Applications*, vol. 78, no. 1, pp. 99–124, 2019.

- [13] X. Zhu and H. Wang, “An improved lbp method for feature extraction and classification of metal defects based on gabor filter and 2dPCA,” in *2020 5th International Conference on Control, Robotics and Cybernetics (CRC)*. IEEE, 2020, pp. 98–102.
- [14] S. Chatterjee, P. Saeedfar, S. Tofangchi, and L. Kolbe, “Intelligent road maintenance: A machine learning approach for surface defect detection,” in *ECIS*, 06 2018.
- [15] X. Tao, D. Zhang, W. Ma, X. Liu, and D. Xu, “Automatic metallic surface defect detection and recognition with convolutional neural networks,” *Applied Sciences*, vol. 8, no. 9, p. 1575, 2018.
- [16] Y. Li, H. Huang, Q. Xie, L. Yao, and Q. Chen, “Research on a surface defect detection algorithm based on mobilenet-ssd,” *Applied Sciences*, vol. 8, no. 9, p. 1678, 2018.
- [17] L. M. Azizah, S. F. Umayah, S. Riyadi, C. Damarjati, and N. A. Utama, “Deep learning implementation using convolutional neural network in mangosteen surface defect detection,” in *2017 7th IEEE international conference on control system, computing and engineering (ICCSCE)*. IEEE, 2017, pp. 242–246.
- [18] L. A. Martins, F. L. Pádua, and P. E. Almeida, “Automatic detection of surface defects on rolled steel using computer vision and artificial neural networks,” in *IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society*. IEEE, 2010, pp. 1081–1086.

- [19] Q. Li and S. Ren, “A real-time visual inspection system for discrete surface defects of rail heads,” *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 8, pp. 2189–2199, 2012.
- [20] X. Chaolin, “On-line detection technique of tiny surface defects for metal plates and strips based on photometric stereo,” *J. Mech. Eng.*, vol. 4, no. 006, 2013.
- [21] S. Höfer, J. Burke, and M. Heizmann, “Infrared deflectometry for the inspection of diffusely specular surfaces,” *Advanced Optical Technologies*, vol. 5, no. 5-6, pp. 377–387, 2016.
- [22] C. McCamy, “Observation and measurement of the appearance of metallic materials. part i. macro appearance,” *Color Research & Application*, vol. 21, no. 4, pp. 292–304, 1996.
- [23] M. C. Knauer, J. Kaminski, and G. Hausler, “Phase measuring deflectometry: a new approach to measure specular free-form surfaces,” in *Optical Metrology in Production Engineering*, vol. 5457. International Society for Optics and Photonics, 2004, pp. 366–376.
- [24] J. Tornero, L. Armesto, M. C. Mora, N. Montes, Álvaro Herráez, and J. Asensio, “Detección de defectos en carrocerías de vehículos basado en visión artificial: Diseño e implantación,” *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 9, no. 1, pp. 93–104, 2012. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1697791211000793>

- [25] L. Armesto, J. Tornero, A. Herraez, and J. R. Asensio, “Inspection system based on artificial vision for paint defects detection on cars bodies,” *2011 IEEE International Conference on Robotics and Automation*, pp. 1–4, 2011.
- [26] Y. K. Imanishi M, A. T, S. Y, C. S, and W. M, “Surfacedefect inspection apparatus,” *USPatent*, p. 5726705A, 1998-3-10.
- [27] F. P. León and S. Kammel, “Inspection of specular and painted surfaces with centralized fusion techniques,” *Measurement*, vol. 39, no. 6, pp. 536–546, 2006.
- [28] W. Fan, C. Lu, and K. Tsujino, “An automatic machine vision method for the flaw detection on car’s body,” in *2015 IEEE 7th International Conference on Awareness Science and Technology (iCAST)*, 2015, pp. 13–18.
- [29] V. Borsu, A. Yogeswaran, and P. Payeur, “Automated surface deformations detection and marking on automotive body panels,” in *2010 IEEE international conference on automation science and engineering*. IEEE, 2010, pp. 551–556.
- [30] N. Ahamad and B. Jana, “Analysis and detection of surface defects in ceramic tile using image processing techniques,” *Microelectronics, Electromagnetics and Telecommunications: Proceedings of ICMEET 2015*, pp. 575–582, 12 2016.
- [31] W. A. Ellingson and M. P. Brada, “Optical method and apparatus for detection of surface and near-subsurface defects in dense ceramics,” *US Patent 5,426,506/A*, 6 1995. [Online]. Available: <https://www.osti.gov/biblio/87733>

- [32] Z. Qu, L. Bai, S.-Q. An, F.-R. Ju, and L. Liu, “Lining seam elimination algorithm and surface crack detection in concrete tunnel lining,” *Journal of Electronic Imaging*, vol. 25, no. 6, p. 063004, 2016.
- [33] Q. Zou, Z. Zhang, Q. Li, X. Qi, Q. Wang, and S. Wang, “Deepcrack: Learning hierarchical convolutional features for crack detection,” *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1498–1512, 2018.
- [34] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, “Automatic road crack detection using random structured forests,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3434–3445, 2016.
- [35] P. M. Mehl, Y.-R. Chen, M. S. Kim, and D. E. Chan, “Development of hyperspectral imaging technique for the detection of apple surface defects and contaminations,” *Journal of food engineering*, vol. 61, no. 1, pp. 67–81, 2004.
- [36] N. Otsu, “A threshold selection method from gray-level histograms,” *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [37] X. Liu, B. Zhang, B. Liu, and Y. Lan, “Phong model based binarization and its application to billet images with non-uniform illumination,” in *2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2017, pp. 1763–1767.
- [38] M. S. Sindhuri and N. Anusha, “Text separation in document images through otsu’s method,” in *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*. IEEE, 2016, pp. 2395–2399.

- [39] W. Niblack, *An introduction to digital image processing*. Strandberg Publishing Company, 1985.
- [40] J. Bernsen, “Dynamic thresholding of gray-level images,” in *Proc. Eighth Int'l conf. Pattern Recognition, Paris, 1986*, 1986.
- [41] J. Sauvola and M. Pietikäinen, “Adaptive document image binarization,” *Pattern recognition*, vol. 33, no. 2, pp. 225–236, 2000.
- [42] B. Zhang, X. Liu, and S. Wu, “Ipso based binarization processing in uneven illumination images for billet defect detection,” in *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2018, pp. 1923–1928.
- [43] O. M. Nishiguchi M, K. K, F. Y, and H. Y, “Vehicle body surface inspection device and surface inspection method,” *Japanese Patent*, p. 6052590, 2016-12-27.
- [44] K. N. Teruo A, M. I, Y. S, and S. K, “Inspection system and process,” *US Patent*, p. 5734742, 1998-3-31.
- [45] Micro-Epsilon, “Fully automatic surface inspection of painted car bodies,” in *Fully automatic surface inspection of painted car bodies*, 2021. [Online]. Available: <https://www.micro-epsilon.com/measurement-systems/Paint-Inspection/karosserie/>
- [46] Stemmer-Imageing, “Defects in automotive panels,” in *Types of Defects - Automotive - Panels*, 2021. [Online]. Available: <https://www.stemmer-imaging.com/en-gb/knowledge-base/types-of-defects-automotive-panels/>

- [47] J. Molina, J. E. Solanes, L. Arnal, and J. Tornero, “On the detection of defects on specular car body surfaces,” *Robotics and Computer-Integrated Manufacturing*, vol. 48, pp. 263–278, 2017.
- [48] A. Yimit, A. Itou, Y. Matsui, and T. Akashi, “Automatic visual inspection method for a coated surface having an orange peel effect: Automatic visual inspection method for coated surface,” *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 14, 10 2018.
- [49] Y. Caulier, “Inspection of complex surfaces by means of structured light patterns,” *Opt. Express*, vol. 18, no. 7, pp. 6642–6660, Mar 2010. [Online]. Available: <http://www.osapublishing.org/oe/abstract.cfm?URI=oe-18-7-6642>
- [50] L. Arnal, J. E. Solanes, J. Molina, and J. Tornero, “Detecting dings and dents on specular car body surfaces based on optical flow,” *Journal of Manufacturing Systems*, vol. 45, pp. 306–321, 2017.
- [51] G. H. Golub and C. F. Van Loan, “An analysis of the total least squares problem,” *SIAM journal on numerical analysis*, vol. 17, no. 6, pp. 883–893, 1980.