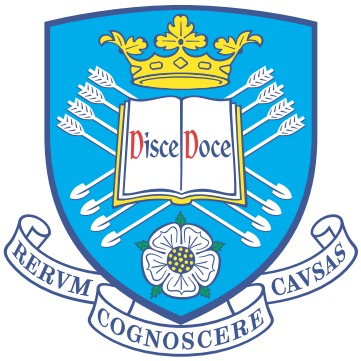
The University of Sheffield

Dept. of Automatic Control and Systems Engineering
Faculty of Engineering



Computer Vision Methods for Autonomous Remote Sizing in Manufacturing

Yueda Lin

May 2023

Supervisor: Prof. Lyudmila S. Mihaylova

Abstract

In the grand scheme of Industry 4.0, the employment of modern intelligent digital technology has been utilised to facilitate industrial production, leveraging automation to elevate production efficiency. Building upon this, Industry 5.0 takes a step forward, accentuating the concept of human-machine symbiosis. It directs its focus on augmenting human performance within the industry, mitigating errors made by workers, and honing the overarching performance of human-machine systems. Across various manufacturing domains, an escalating demand for this level of automation has been noticed. One such area is the speciality steel industry, whose tasks are the primary consideration of this dissertation.

Speciality steel rolling forms the backbone of industrial sectors as diverse as aerospace and oil and gas. The key to the sustained survival of steel plants hinges on the digitalisation of the rolling process. Despite this, a significant number of steel rolling plants in the present day continue to place a heavy reliance on human operators to oversee and regulate the manufacturing process.

With a view to securing the safety of workers in high-risk factory environments and optimising the control of steel production, this dissertation puts forth machine vision approaches. These are aimed at supervising the direction of hot steel sections and remotely gauging their dimensions, both conducted in real-time. This dissertation further contributes a novel image registration approach founded on extrinsic features. This approach is then amalgamated with frequency domain image fusion of optical images. The resultant fused image is designated to evaluate the size of high-quality hot steel sections from a remote standpoint.

With the integration of the remote imaging sizing module, operators can stay abreast of the section dimensions in real time. Concurrently, the mill stands can be pre-adjusted to facilitate quality assurance. The efficacy of the developed approaches has been tested over real data, delivering an accuracy rate exceeding 95%. This suggests that the approach not only ensures worker safety but also contributes significantly to the enhancement of production control and efficiency in the speciality steel industry.

Acknowledgements

Many people have helped me in the process of completing this thesis and studying. First, my tutor, Professor Mila, is the most important person, who is very grateful for her guidance and help. Secondly, I would like to thank Dr Peng Wang. During the whole project, we solved many problems together, and I learned a lot from him. At the same time, I would like to thank Liberty Speciality Steels for inspiring this research, for the support and for providing the real data. I would like to express my gratitude to Simon Pike and Ree Muroiwa for the valuable discussions and advises. Finally, I would like to thank my family for supporting me in all aspects.

Contents

List of Nomenclature

Abbreviations

2D	Two-dimensional
3D	Three-dimensional
BRIEF	Binary Robust Independent Elementary Features
CBOI	Current Boundary Of Interest
CNN	Convolutional Neural Network
CT	Computed Tomography
CV	Computer Vision
DOG	Difference of Gaussian
FAST	Features from Accelerated Segment Test
FFT	Fast Fourier Transform
GAN	Generative Adversarial network
HOG	Histogram of gradients
HRB	Hot Rolled Bar
LiDAR	Light Detection And Ranging
MRI	Magnetic Resonance Imaging
MST	MultiScale Transformation
NCC	Normalized Cross-Correlation
NMS	Non Maximum Suppression
ORB	Oriented FAST and Rotated BRIEF

PCNN	Pulse Coupled Neural Network
SIFT	Scale invariant feature transformation
SPECT	Single-Photon Emission Computerized Tomography
SR	Sparse Representation

List of Figures

List of Tables

Chapter 1

Introduction

Computer Vision (CV) is one of the popular fields in machine learning. With the development of deep learning, some deep learning methods are used in the computer vision field. However, some traditional feature extracted methods and decision-making methods still have some advantages compared to deep learning methods. My work aims to create and develop the vision algorithms, which are applied in the industry and manufacturing fields.

Automation has become one of the essential aspects in modern industry and manufacturing ever since the first moving assembly line built by Ford in 1913 [64]. In recent years, machine learning becomes more and more popular. In the industrial 4.0 vision, modern intelligent digital technology can enable industrial production, automate production, and increase productivity. On this basis, Industry 5.0 emphasizes the symbiosis of human-machine cooperation, focusing on improving human performance in the industry, preventing worker errors, and optimizing the overall performance of human-machine [31]. As an embracement of machine learning, computer and machine vision technologies have witnessed fast applications in a wide range of areas - from the automobile production to aircraft manufacturing, etc. They have made quite good results in practical applications in the industry due to the real-time, highly efficient capabilities to deliver accurate results by fusing various sensor data. The next following subsections will cover the different popular research directions of computer vision and their applications in manufacturing fields.

1.1 Motivation

Speciality steels rolling plays a critical role in industrial sectors such as aerospace, oil and gas. Speciality steels also referred to as alloy steel include additional alloyed elements that grant distinct properties to the final product. These steels are specifically engineered to offer superior performance under particular conditions according to ISO 4948/21981 standard.

Digitalisation of the rolling process is considered as a key to the long-term viability of

steel plants.

The quality standards for *ingots*, as well as steel sections which are manufactured from ingots, strictly set an upper limit on the final quality of all products in downstream industries. Despite the wave of digitalisation, a lot of nowadays steel rolling plants are still heavily relying on human operators to control and monitor the manufacturing/rolling process. It has been shown that the long-term exposure to a high-temperature, intense light environment in steel factories could cause injuries, particularly to eyes [25].

The production of Hot Rolled Bar (HRB) has both high capital and operational costs, which are further increased by yield losses when dimensions do not conform to tight specifications. Traditionally, contact measurement techniques such as callipers, mechanical gauges or chalk-marked rods are used to measure the HRBs. Although these methods can get the dimension with a certain accuracy, they are most likely to put human workers in a hazardous environment. One additional drawback is that the measurement is usually performed at the very last stage of rolling.

In the process of automatic rolling, the real-time and automated non-contact detection and measurement of steel sections on the production line can be taken as a way to assure quality and eventually avoid hazards for operators and financial loss due to errors during the manufacturing process. For instance, Zhou et al. [73] propose an approach for online diameter estimation of hot forgings. A fast measurement method based on feature line reconstruction of stereo vision is developed to increase the calculation speed. A considerable measurement accuracy is demonstrated, but the approach requires good lighting conditions and high-precision camera calibration.

1.2 Thesis Structure and Contributions

The chapters of the thesis are laid out as the following order.

- **Chapter 2** - Literature Review.

  Chapter 2 is the literature review, which first introduces the technology related to the steel production process. Then the relevant algorithms of visual measurement are introduced and discussed. The image edge detection, registration, and fusion algorithms are also analyzed.

- **Chapter 3** - Edge Detection.

  Chapter 3 introduces two experiments of visual measurement with monocular cameras. The images taken by thermal and optical cameras are used, respectively, for visual measurement to measure the diameters of the steel sections on the production line of the steel plant.

- **Chapter 4** - Image Registration.

  Chapter 4 introduces the experimental process of image registration using dual optical cameras and the virtual checkerboard. Through this method, the measurement plane and object plane can be matched adaptively.

- **Chapter 5** - Image Fusion.

  Chapter 5 is the image fusion experiment after image registration. Different image fusion methods are tested to improve the effect of visual measurement.

- **Chapter 6** - Pattern Recognition

  Chapter 6 introduces a classification case in an actual steel plant. For the purposes of automating the steel rolling process, the direction of the ingots placed on the production line needs to be determined. If the top of the ingot (the one with the 'Brain' pattern) is facing the camera, it needs to alert the staff to change the ingot direction in time. Wrong ingot direction can affect the performance of finished steel and lead to a scrap of steel.

- **Chapter 7** - Conclusions

  Chapter 7 is the conclusions and the future works.

Main Contributions

The main contributions of this dissertation are the following:

i) A new approach for remote steel dimension measurement is introduced, which utilises a monocular thermal imaging camera as the input source. By incorporating environmental reference objects as points of reference, this method effectively bypasses the need for the traditional calibration process. The introduction of this non-conventional approach demonstrates a significant step towards enhancing the efficiency and accuracy of steel dimension measurements.

ii) An alternative method for the remote measurement of steel dimensions is suggested, deploying a monocular optical camera as the primary input. The camera is pre-calibrated using a checkerboard, ensuring precise measurements. Additionally, the study proposes an analytical approach based on weighted variance, providing a robust method for scrutinising the measurement process. This dual-pronged approach effectively enhances the accuracy and reliability of remote steel dimension measurements.

iii) A new two-camera-based approach for hot steel section remote sizing is proposed, by embedding efficient image fusion methods. The approach is robust to environmental changes, which include high temperature, evaporation and other sources of noise. It achieves high precision results for non-contact measurements in medium-range distances.

iv) A new image registration approach is proposed that uses extrinsic features from a virtual checkerboard and this also improves the system's robustness against environmental changes.

v) An efficient image recognition approach is developed for ingot direction recognition, which provides a new perspective on automating the recognition of steel ingot orientation.

vi) The proposed framework was validated using real-world data collected from a high-quality steel manufacturing plant, demonstrating the efficiency of the proposed approach and its potential for industrial applications. The achieved remote sizing accuracy is above 95% with a tolerance range of 2 mm which is a significant technical advance in the remote measuring of steel sections.

1.3 Publications

The author's publications associated with the thesis topic are outlined below:

Peer Reviewed Conference Proceedings

1. Peng Wang, Yueda Lin, Ree Muroiwa, Simon Pike,Lyudmila Mihaylova, "Computer Vision Methods for Automating High Temperature Steel Section Sizing in Thermal Images", In *Proceedings of the 2019 Sensor Data Fusion*, Bonn, Germany, October 2019, pages 1-6.

2. Peng Wang, Yueda Lin, Ree Muroiwa, Simon Pike, Lyudmila Mihaylova, "A Weighted Variance Approach for Uncertainty Quantification in High Quality Steel Rolling", In *Proceedings of the IEEE 23rd International Conference on Information Fusion*, Rustenburg, South Africa, July 2020, pages 1-7.

3. Yueda Lin, Peng Wang, Ree Muroiwa, Simon Pike, Lyudmila Mihaylova, "Image Fusion for Remote Sizing of Hot HighQuality Steel Sections", In *Proceedings of the UKCI 2021. Advances in Intelligent Systems and Computing*, vol 1409. Springer Nature , pages 357-368.

Peer Reviewed Journal Paper

1. Yueda Lin, Peng Wang, Sardar Ali, Ree Muroiwa, Lyudmila Mihaylova, Towards Automated Remote Sizing and Hot Steel Manufacturing with Image Registration and Fusion, *Journal of Intelligent Manufacturing*, under review, 2023.

Chapter 2

Related Work

2.1 Background

2.1.1 Demands in the Steelmaking Industry

In numerous manufacturing settings, there is a need for remote evaluation of the sizes of produced items, a process known as '*remote sensing*'. This term refers to the technology that utilises sensors to detect targets from a distance without direct cont

One significant industry with such a demand is the steel manufacturing sector, where the high temperatures of hot steel (exceeding 1000 degrees Celsius) necessitate autonomous, non-contact operations.

Steel rolling is a crucial component of the supply chain in various industries, including automobile manufacturing, transportation, and infrastructure construction. The quality of steel ingots effectively determines the maximum quality of all downstream products. At present, a large portion of steel casting and processing is still manually executed. However, operators working in high-temperature steel factory environments are not only at risk of physical injury, but the intense light during the steel rolling process can also cause considerable damage to human eyes. [25].

Steel, an alloy material with iron as the primary element and a carbon content below 2%, has variable compositions depending on specific material property requirements. The content of carbon elements in steel can be adjusted, and other elements like chromium, manganese, molybdenum, nickel, silicon, tungsten, vanadium, phosphorus, and sulfur can be added. Advancements in smelting technology have led to significant improvements in steel's output and performance in modern society, making it widely used in infrastructure, manufacturing, and daily life.

The production of large-scale steel typically incurs high costs, from raw materials to the energy consumed during the production process. In the course of automatic rolling, real-time, automated non-contact measurement/detection of steel sections on the production line is a

potent method of process control. Real-time sizing significantly aids product quality control and fosters improvements and innovation in the manufacturing process. However, errors in measurement or faulty detection can lead to wastage of raw materials, and in some cases, even result in the scrapping of raw materials.

2.1.2 Computer Vision Methods for Non-Contact Measurements

In the early 1980s, Marr introduced the first comprehensive vision system framework from an information processing perspective, now known as Marr's vision theory [42]. According to this theory, the system processes visual information through three stages, moving from the raw data (two-dimensional (2D) image data) to the final representation of the 3D environment.

In the realm of non-contact measurement methods, techniques based on camera use [8] and Light Detection and Ranging (LiDAR) [74] are prominent areas of research. While camera-based methods, in the absence of on-site calibration, generally lack the precision of LiDAR-based methods, the cameras' compact form factor, ease of portability, and cost-effectiveness make them popular choices in numerous environments, even those involving high-temperature forging [17] and welding [65]. Structured light-based dimension measurement methods [58] effectively merge the benefits of both LiDAR and passive camera-based approaches. They have shown superior performance compared to contact measurements obtained with calipers, particularly in scenarios involving hot and large scale forging.

Camera-based systems are classified as either monocular or binocular. Binocular systems have found extensive application in manufacturing processes, ranging from measurement [61] to inspection [57] and fatigue crack detection [28]. They achieve high measurement accuracy through the complementary information they gather and precise depth estimation. However, there are still high-accuracy manufacturing applications for monocular systems. For instance, in [8], a real-time vision-based method was developed to monitor a workpiece's diameter during the turning process. The results, compared with manual measurements using a digital caliper, demonstrated the method's effectiveness, with an error of less than 0.6%. Additionally, Bi et al. [5] proposed using a single Charge Coupled Device (CCD) camera to measure the dimension of forgings with temperatures ranging from 800 to 1200 degrees Celsius, achieving accurate edge extraction through the use of both a digital and a physical filter.

An Automated, Intelligent, Online-decision-making, and Non-contact (AI-ON) measuring system is integral to the automation of steel rolling processes and the reduction of operational costs. Such systems present higher efficiency compared with traditional rolling systems that employ contact-based measuring methods and safeguard human operators from hazardous environments.

During automatic rolling, real-time, automated non-contact measurement of steel sections

on the production line is a potent method for process control. This real-time sizing significantly facilitates product quality control and fosters improvements and innovation in the manufacturing process.

A myriad of three-dimensional (3D) measurement and estimation methods based on vision have been studied to obtain the physical sizes of objects [18]. These 3D vision measurement methods can be broadly divided into three categories: time of flight methods, structured light methods, and stereo vision methods.

The *time of flight method* [22] uses a particular time of flight camera. In addition to capturing the colour information of pixels, the camera also records the time from the pixel light source to the camera. Therefore, the time of flight camera needs an individual artificial light source. The time is obtained by calculating the phase difference of light, and then the propagation speed of light is used as the reference to calculate the depth map. The error of the time-of-flight method is affected by many aspects, such as colour and material of the object, the distance to the camera and illumination of the environment. The depth error in $5m$ is less than $4.6mm$.

The *structured light method* [40, 72] relies on projecting the inferred light with specific structural characteristics onto the object, then collect the reflected structured light pattern by the infrared camera, and calculate the 3D information according to the principle of triangulation. Since the encoded structured light image or speckle spots are easily confused by intense natural light outdoor, the structured light scheme is not easy to use outdoor. When the object is far away from the camera, the intervals between light points projected on the object increases and lead to a decrease in accuracy. The structured light camera is also easily affected by the reflection of a smooth plane, such as a mirror.

Stereo vision measurement systems have also been proposed [43, 73]. In the analysis of two images taken by the binocular camera in the same scene, the parallax image is obtained by the stereo matching algorithm. Then the depth map and depth information are obtained by including geometry information. For the binocular stereo vision system, the camera calibration is the most crucial aspect that affects the measurement accuracy. The number of calibration images and the position of the calibration plate plays a significant role [68]. For instance, Zhou et al. [73] propose an approach for online diameter estimation of large hot forgings. A fast measurement method based on feature line reconstruction of stereo vision is developed to increase the calculation speed. Their measurement accuracy is considerable, but the method needs good lighting conditions and high-precision camera calibration.

Among non-contact measuring methods, Light Detection And Ranging (LiDAR) based methods such as [74], have demonstrated high accuracy. However, this comes with a significant economical cost. Alternatively, camera based methods provide high accuracy and are inexpen-

sive. For instance, by projecting structured light onto steel bar surfaces, the method developed in [59] is able to measure the diameter of a 53 tonnes round steel bar with a maximum error of 0.38%. Liu et al. [35] propose an approach for online diameter estimation of cylindrical forgings, obtaining relative errors less than 0.7%. Similarly, Yang et al. [67] propose a method that shows improved measurement accuracy of rectangular forgings, attaining an average 0.48% estimation error. Since the encoded structured light can be easily perceived as intense light, this approach cannot be used easily in rolling applications where there are various intense light sources.

To avoid the disadvantages of the aforementioned measuring systems, Zatočilová et al. [70] design a passive, stationary multi-image system for fast measuring of dimensions and straightness of rotationally-symmetric forgings. An edge detection algorithm that exploits simple shapes of the forging is developed. After extracting four boundary curves of the forging in a pair of images, a 3D model reconstruction is performed where the length, diameter, and straightness of the forgings are calculated. The system is proved capable of performing diameter measuring with deviations less than 1%. Wu et al. [65] propose a monocular-vision-based method for online measuring of a weld stud. An accurate mathematical model constrained by the measuring principle is developed. Based on the model, a further calibration is proposed to optimise the projective transformation parameters. They show that the model is flexible, fast, and capable of achieving high-precision measurements of the weld stud. Nevertheless, the temperature in these two cases is not as high as to 1000 - 1500°C, at which the intense radiation of the Hot Rolled Bar (HRB) can cause overexposure problems easily.

2.2 Edge Detection

For visual perception, an essential feature in obtaining an object's position information is its contour and edge features.

For visual measurement, the accuracy of the measurement algorithm is fundamentally determined by the ability to obtain the edge information of the object clearly and accurately. In the medium and long-distance vision measurement, the pixel-level edge recognition error will lead to significant errors in the measurement results after scale conversion. Therefore, choosing an appropriate edge detection algorithm is the cornerstone of an excellent visual measurement algorithm.

2.2.1 Edge Detection Based on Differential Operators

Some basic edge detection filters are based on differential operators, such as Sobel, Prewitt and Roberts operators [1]. The image edge is captured using the gradient information of the image by convoluting and manipulating the mask as a kernel with each image's pixel. Differentiator

calculates the gradient change of the grey scale or RGB value of its pixel point, and the higher the value is, the more likely it is to be the edge. Then an appropriate threshold is selected to filter the extracted edges. The following filters are used:

$$R_x = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, R_y = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}. \tag{2.1}$$

Equation (2.1) shows the Roberts operator, it contains a pair of 2×2 convolution kernel. It uses the difference between two adjacent pixels in the diagonal direction to approximate the gradient amplitude to detect the edge. In principle, the output of the Roberts operator is located in the middle of four rectangular connected pixels. Since the Roberts operator uses 2×2 convolution kernel, its computing speed is quite fast, but at the same time, it is quite sensitive to local noise.

The Prewitt operator and Sobel operator are given respectively with the following equations:

$$P_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, P_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}, \tag{2.2}$$

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, S_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \tag{2.3}$$

The Prewitt and the Sobel operators are composed of two 3×3 convolution kernels that respond to changes in intensity in the horizontal and vertical directions, respectively. By plane convolution of the image with these two convolution kernels, approximate values of the transverse and longitudinal intensity differences can be obtained, respectively. Prewitt and Sobel operators respond more to the vertical and horizontal edges relative to the pixel grid when the Roberts operator is more sensitive to the oblique edges. Prewitt and Sobel operators have 3×3 convolution kernels, so they have a longer computational time compared to the Roberts operator which has 2×2 kernels. However, their larger convolution kernels cover more pixel points, making the input image smoother and less noise-sensitive.

The main difference between the Sobel operator and the Prewitt operator is that the Sobel operator adds the concept of weight on the basis of the Prewitt operator. It considers that the distance between neighbouring points has a different effect on the current pixel point. The closer the pixel point is, the greater the effect on the current pixel, thus sharpening the image and highlighting the edge outline. The output of the Sobel operator has some isotropy, whereas the Prewitt and Roberts operators have no isotropy.

2.2.2 Canny Edge Detection

The Canny operator is a multi-level edge detection algorithm [7]. In the paper [7], Canny proposed some standards for edge detection. The goal of edge detection is to find an optimal edge. The definitions of the optimal edge are:

- The algorithm can mark the images' actual edges as much as possible without omitting small edges.
- The identified edge should be as close as possible to the actual edge in the actual image. That is, the positioning of the edge needs to be accurate.
- Edges in the image should not be repeatedly marked, and image noise should not be identified as edges as much as possible.

Algorithm 1: Canny Edge Detection

Input: Original Image I, Gaussian Kernel K_G, Double Threshold $[\theta_l, \theta_h]$

Output: Detected Edges E

1: Use Gaussian convolution smooth the image with Gaussian Kernel K_G
2: Use differential operator to calculate the gradient of each pixel
3: Use non maximum suppression thin the detected edges
4: Use double threshold θ_l, θ_h to classify the edges
5: Weak edge suppression

As demonstrated in Algorithm 1, the initial step of the Canny edge detection process involves smoothing the image using Gaussian convolution. Gaussian filtering is arguably the most popular denoising filtering algorithm. The process begins with the generation of a convolution kernel according to the Gaussian formula. This Gaussian kernel is then used to average the gray values of the pixel points targeted for filtering along with those of the neighboring pixels. This procedure effectively filters out the high-frequency noise that may have been superimposed on the ideal image, thus enhancing the image quality.

After the image smoothing, the Canny edge detection algorithm applies two thresholds to detect edges. These thresholds, often selected through trial and error, help to differentiate true edges from noise or other insignificant features. This way, the algorithm ensures that only the most salient and relevant features are considered during edge detection, thus improving the overall accuracy and robustness of the process.

The 2D-Gaussian kernal, is in the form:

$$G(x, y; \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \tag{2.4}$$

$$G_{33} = \begin{bmatrix} 0.0751 & 0.1238 & 0.0751 \\ 0.1238 & 0.2042 & 0.1238 \\ 0.0751 & 0.1238 & 0.0751 \end{bmatrix}, \tag{2.5}$$

$$G_{55} = \begin{bmatrix} 0.0030 & 0.0133 & 0.0219 & 0.0133 & 0.0030 \\ 0.0133 & 0.0596 & 0.0983 & 0.0596 & 0.0133 \\ 0.0219 & 0.0983 & 0.1621 & 0.0983 & 0.0219 \\ 0.0133 & 0.0596 & 0.0983 & 0.0596 & 0.0133 \\ 0.0030 & 0.0133 & 0.0219 & 0.0133 & 0.0030 \end{bmatrix}, \tag{2.6}$$

where σ is the standard deviation of the distribution. Equations 2.5 and 2.6 show the discrete approximation of the Gaussian kernels 3×3, 5×5 with $\sigma = 1$. The larger the σ, the lower the weight of the central pixel, and the more vulnerable the value of the current pixel is to the influence of the surrounding pixels. The larger the kernel size, the better the filtering effect, but some details may be missing.

The algorithm's second step is calculating the gradient of the filtered image. The gradient calculation uses the differential operator mentioned above in section.2.2.1, and generates the gradient's magnitude and angle (Eq.2.7)through the edge detector's convolution operation. Here, θ is the angle of gradient

$$\theta = \arctan\left(G_y / G_x\right), \tag{2.7}$$

where G_y and G_x are the magnitude of gradient in vertical and horizontal directions. The gradient direction is calculated for a non maximum suppression algorithm to be used later.

The gradient image obtained from the second step has many problems, such as wide and weak edges. Now non-maximum suppression is used to find the local maximum of pixel points, and the grey value corresponding to the non-maximum value is set to 0, which can remove a large part of non-edge pixel points. The non-maximum suppression algorithm compares the gradient magnitude of adjacent pixels along the positive and negative gradient direction of the target pixel, which retains the pixel if it is an extreme value or discards if it is not. The edges of the final image generated after non-maximum suppression are ideally single-pixel edges.

After these three steps, the edge quality is already high, but there are still many false edges. Hence, the Canny algorithm uses the double threshold method to filter the edges further. The thresholds are typically set by the user based on the desired balance between edge detection sensitivity and noise reduction. Pixel points with gradient intensity below the low threshold are suppressed and not used as edge points. Pixel points above the high threshold are defined as strong edges and retained as edge points. Between high and low thresholds is defined as a weak

edge, which remains to be further processed. The edges are linked into contours from high-threshold images. When the endpoint of the contour is reached, the algorithm searches for a point satisfying the low threshold among the eight neighbour pixels of the breakpoint pixel and then collects new edges from this point until the whole image is closed.

2.2.3 Structured Random Forest

Decision Tree Learning

The fundamental function of a decision tree is to classification. 'Classification' task is to decide which preset groups should the input sample belongs to. A decision tree has a tree structure, which has nodes and directed edge (branch). There are two kinds of nodes, internal nodes and leaves. An internal node contains thresholds of a feature and split the input data into several branches according to the thresholds. The leaves are the classification results and the output layer of a decision tree, and the input variable will be put into one of the leaves and labelled.

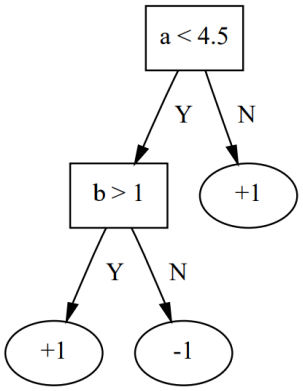


Figure 2.1: A Simple Decision Tree Model [16]

As figure 2.1 shows, for a input $x_i = [a_i, b_i]$, this decision tree can classify the input x into label -1 and label 1. For decision tree learning, how to decide the split function is the main topic. For the existing decision tree learning algorithm, information entropy is the main standard of computing split function.

Information entropy H is introduced by Shannon in 1948 with the following expression [53]

$$H = -K \sum_{i=1}^{n} p_i \log p_i, \tag{2.8}$$

where K is a positive constant and p_i is the probability of case i occurs in the system.

In decision tree learning algorithm proposed by Ross Quinlan, the optimization object of his decision tree learning algorithm is to maximum the gain of information when producing each

split function

$$info_X(T) = \sum_{i=1}^{n} \frac{|T_i|}{|T|} \times info(T_i), \tag{2.9}$$

$$gain(T) = info(T) - info_X(T). \tag{2.10}$$

In the Quinlan's notation, $info(x)$ is the information entropy of x, T is the set of data in the current node, T_i is the subset data which is partitioned after passing through current node to the sub-nodes. For a binary tree like figure 2.1, n is equal to 2. The information gain expression can be written as:

$$InfoGain(T) = H(T) - \frac{N_{left}}{N_T} H(T_{left}) - \frac{N_{right}}{N_T} H(T_{right}), \tag{2.11}$$

where N_{left} is the number of data been partitioned to the left branch.

When the decision tree reach a preset depth, the training of decision tree will end. If the depth of a decision tree becomes too deep, the model may face the problem of overfitting.

Random Forest

Random forest is one of the supervised training model, which is first developed by Tim Kam Ho in 1995 [24]. Random decision forest is easy to produce, and the computation power used is low. At the same time, the random forest model has a significant performance on classification and regression problems.

A random forest model is the integration of multiple decision trees. The idea of ensemble learning is to construct a strong classifier by combining several weak classifiers. Its generalization error is small and recover the overfitting problem from decision trees.

In the training process, a small training set is generated by sampling from the training data. Then, training the decision tree by using the newly generated small training set. Repeat the previous process, and multiple decision trees can be produced. By integrating the output of multiple decision trees, we can get a final random forest output. For random forests, many studies have focused on how to sample from the in the original training set to produce the new training sets for decision trees in order to weaken the correlation between decision trees [12].

Structured Random Forest

The edge extraction method used in this dissertation is based on the structured random forest algorithm developed by Pdollar et al. [13], which introduces a "structure" to the traditional

random decision forest. The main idea of the random decision forest is to produce decision trees and train the split function

$$h(x, \theta_j) \rightarrow \{0, 1\}, \tag{2.12}$$

where x is the input, θ_j is the trained parameter at node j of the tree, and $\{0, 1\}$ indicates the input x is split left or right to the subsequent nodes. When the inputs reach to the leaves of the trees, they are labelled as $y \in \mathcal{Y}$. The training process of the decision trees is to maximize the information gain I_j [13] of the given node j

$$I_j = I\left(S_j, S_j^L, S_j^R\right), \tag{2.13}$$

where $S_j \in \mathcal{X} \times \mathcal{Y}$ is the training dataset with $\mathcal{X}$ the sample space, $S_j^L = \{(x, y) \in S_j | h(x, \theta_j) = 0\}$, $S_j^R = S_j \backslash S_j^L$, $x \in \mathcal{X}$ is a sample, and L and R indicate the left and right branches.

After the decision trees are trained, the structured random forest framework further maps the labels $y \in \mathcal{Y}$ into a discrete label set $c \in \mathcal{C}\{1, ..., k\}$. The similarity of the labels are measured by a intermediate mapping

$$\Pi : \mathcal{Y} \rightarrow \mathcal{Z}. \tag{2.14}$$

These labels y with the similar $z \in \mathcal{Z}$ are mapped into the same discrete label c. With the hierarchical label mappings, the structured random forest manages to label each pixel and determine whether the pixel is part of an edge. By assembling a large number of trees with each response to a different feature channel (colour and image gradient, etc.), the final edge detection results are generated by considering the votes among all the trees in the structured forest. More details can be found in [13].

The structured random forest employed in this dissertation has been pre-trained using the Berkeley Segmentation Dataset and Benchmark (BSDS500). This pre-training enables the algorithm to produce fast and highly accurate edge detection results, not only on the test sets but also on our thermal steel section images. The model is trained with 8 trees and 1 million patches per tree, and further details regarding the hyper-parameters can be found in [13]. Figure 2.2 showcases the edges extracted from one of the steel section images using the structured random forest algorithm. It is evident that the algorithm successfully extracts the edges of the steel section with clarity.

When a single image is not enough to obtain enough information, multiple images will be taken, and the information in the images will be fused to enhance information acquisition. Before fusion, images need to be registered in position.

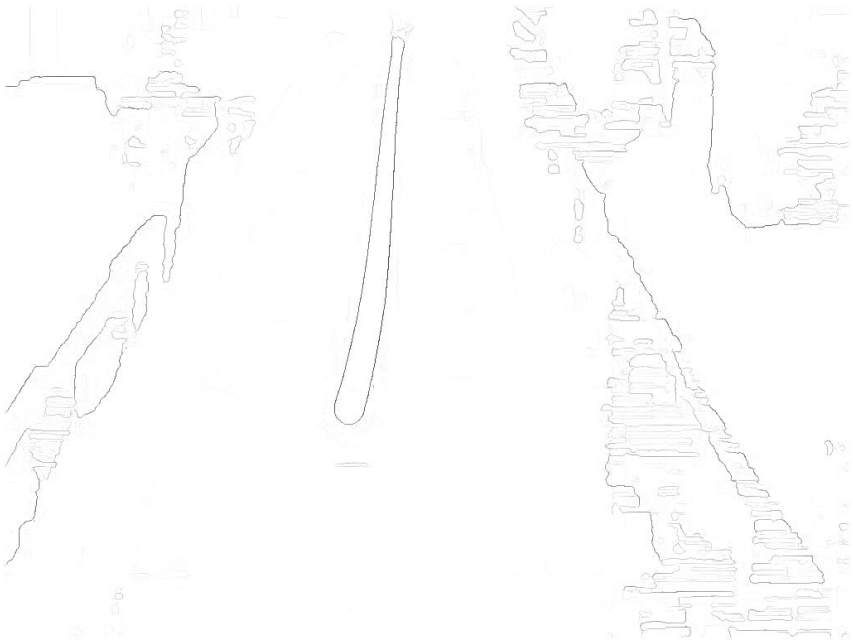


Figure 2.2: Structured Forest Edge Extraction

2.3 Image Registration

Image registration has many practical applications in medical image processing and analysis. With the advancement of medical imaging equipment, images containing accurate anatomical information such as CT and MRI can be acquired for the same patient; at the same time, images containing functional information such as SPECT can also be acquired. However, diagnosis by observing different images requires spatial imagination and the subjective experience of the doctor. Using the correct image registration method, various information can be accurately fused into the same image, making it easier and more accurate for doctors to observe lesions and structures from various angles. At the same time, through the registration of dynamic images collected at different times, changes in lesions and organs can be quantitatively analyzed, making a medical diagnosis, surgical planning, and radiation therapy planning more accurate and reliable. In the field of computer vision, registration methods can be used for video analysis, pattern recognition, and automatic tracking of object motion changes. In material mechanics, registration is often used to study mechanical properties, known as digital image correlation. By fusing and comparing the information (shape, temperature, etc.) collected by different cameras and sensors, values such as strain field and temperature field can be calculated. By bringing into the theoretical model, parameter inverse optimization can be performed. Image registration is not only to register two or more images taken by different devices at different angles, but also to match images taken in the same environment at different times.

Traditional image registration can be divided into area-based registration and feature-based registration [75]. The process of these registration methods can also be divided into four

steps: feature detection, feature matching, transform function estimation and image resampling.

The classic method in area-based registration is the normalized cross-correlation (NCC) method. The NCC is used to describe the degree of correlation between two targets, i.e. it can be used to describe the similarity between images or between image parts. In image registration, a certain template will be set. Then, a search of the area which has the highest NCC is performed with a template in an image as the corresponding matching, and then the whole image is aligned. The expression of the NCC is:

$$\gamma(u,v) = \frac{\sum_{x,y}\left[f(x,y) - \bar{f}_{u,v}\right]\left[t(x-u,y-v) - \bar{t}\right]}{\left\{\sum_{x,y}\left[f(x,y) - \bar{f}_{u,v}\right]^2 \sum_{x,y}[t(x-u,y-v) - \bar{t}]^2\right\}^{0.5}}, \tag{2.15}$$

where u, v are the height and width of the template, $\bar{f}_{u,v}$ is the mean of $f(x,y)$ in the region u, v, $\bar{t}$ is the mean of template image [69].

Obviously, normalized cross-correlation (NCC) is not an ideal feature tracking method because it is not invariant in image scale, rotation and perspective distortion. These limitations have been addressed in various solutions, including some that make NCC an integral part. These follow-up algorithms and NCC are widely used in the field of feature tracking for tracking particular objects in images.

Apart from the direct matching of an image, feature point matching is a very common registration method. Generally, the selected feature points are pixels with some singularity relative to their field. Feature points are often easy to extract, but the information contained in feature points is relatively small, which can only reflect their position and coordinate information in the image. Therefore, the key is finding the matching feature points in the two images.

The key point is also known as the point of interest. It defines essential and characteristic places (such as corners, edges, etc.) in an image. Each key point is represented by a descriptor (eigenvector containing the essential characteristics of the key point). The descriptor should be robust to image transformations (such as position transformation, scaling transformation, brightness transformation, etc.). Therefore, many algorithms provide key point detection and feature description.

Generally, image registration techniques with features include four aspects: transformation model, feature space, similarity measure, search space and search strategy. According to these four characteristics, the steps of image registration can generally be divided into the following five steps:

1. Select the appropriate transformation model according to the actual application

2. Select a suitable feature space, based on grayscale or based on features

3. According to the parameter configuration of the transformation model and the selected

features, determine the possible range of parameter changes, and select the optimal search strategy

4. The similarity measure is used to search according to the optimization criterion in the search space to find the maximum correlation point, so as to solve the unknown parameters in the transformation model

5. The images to be registered correspond to the reference images according to the transformation model to realize the matching between the images

Therefore, selecting appropriate features for matching is the key to registration.

2.3.1 Histogram of gradients (HOG)

The histogram of gradients is widely spread and used after Navneet Dalal and Bill Triggs 's work on human detection had been published. [10]. In HOG, the gradients of pixels along x and y axis are extracted. The edges and corners of the image are shown clearly through the HOG features. Analysis the gradients of pixels in the image will lead some advantages compared to directly analysis the color and intensity of the pixels. For example, the results of HOG are not influenced by the brightness of image.

During the algorithm, the original image is cropped into several 'cells'. The histogram is firstly calculated from these cells. The size of the cells can be changed. In order to have better invariance to light and shadows, it is necessary to normalize the contrast of histogram, which can be achieved by making cell units into larger blocks and normalizing all cell units in the block.

The gradients are calculated in the horizontal and vertical directions. This can be realized by using the following convolution kernel

$$\begin{bmatrix} -1 & 0 & 1 \end{bmatrix}, \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}^T. \tag{2.16}$$

After the gradients of pixels in x and y direction have been calculated, the magnitudes and the directions of gradients are determined by the following equations

$$M(x,y) = \sqrt{M_x(x,y)^2 + M_y(x,y)^2}, \tag{2.17}$$

$$\theta(x,y) = tan^{-1}\frac{M_y(x,y)}{M_x(x,y)} \in [0^\circ, 360^\circ) \, or \, [0^\circ, 180^\circ), \tag{2.18}$$

$$M_x(x,y) = f(x+1,y) - f(x-1,y), \tag{2.19}$$

$$M_y(x,y) = f(x,y+1) - f(x,y+1), \tag{2.20}$$

where $M_x(x,y)$ is the magnitude of vertical gradient at pixel (x,y), $M_y(x,y)$ is the magnitude of horizontal gradient at pixel (x,y), $\theta(x,y)$ is the direction of gradient at pixel (x,y), and $f(x,y)$ is the magnitude of intensity at pixel (x,y).

The gradients of pixels in the cells are then used to create cell histogram. In Dalal's paper, the pixels in a 8×8 cell are divided into a 9 channel histogram with a bin size of $40°$. As an output, the final magnitude and direction of a cell is voted by the histogram. Combined the outputs from cells will provided a HOG feature map of the image.

2.3.2 Scale invariant feature transformation (SIFT)

Scale invariant feature transformation (SIFT) finds the scale and orientation of the interesting points from image [36, 37]. The description and detection of local image features can help identify objects. Sift extracts the local features of the image, finds the extreme points in the scale space, and extracts the position, scale and direction of them by using Gaussian filters and down sampling methods. Under the current computer hardware speed and small feature database conditions, the recognition speed can be close to real-time operation. SIFT features a large amount of information, suitable for fast and accurate matching in massive databases. The advantage of SIFT is

- The SIFT features are the local features of image. It is invariant to the scaling, rotating and intensity changing.
- Calculation speed

The SIFT algorithm is started with building a Gaussian pyramid for the input image. And the pyramid is built by following two methods.

- Applying different scales Gaussian filter to the image
- Down sampling the original image

Start with Gaussian filter, the basic idea of the algorithm is to search feature points in different scale-space of the image. By applying different Gaussian filter $G(x,y,\sigma)$ to the image, the image $I(x,y)$ is changed into variant scale-space and is represented as

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y),, \tag{2.21}$$

where $*$ denotes the convolution operation for x and y pixel coordinates, $G(x,y,\sigma)$ is the variable-scale Gaussian

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \tag{2.22}$$

With different standard deviations σ, the original input image $I(x, y)$ can be transferred to different variant scale-space. σ becomes the scale coordinate, or scale space factor, and the size of σ determines the degree of smoothness. The large scale corresponds to the overview features of the image, and the small scale corresponds to the detailed features of the image. A large σ value corresponds to a coarse scale (low resolution) and, conversely, corresponds to a fine scale (high resolution).

The first octave of the pyramid consists of images with original size but in variant scale-space, which means using different Gaussian filters. For the next octave, the original image is down sampled by a factor of 2 and the down sampled image is also changed into different scale-space.

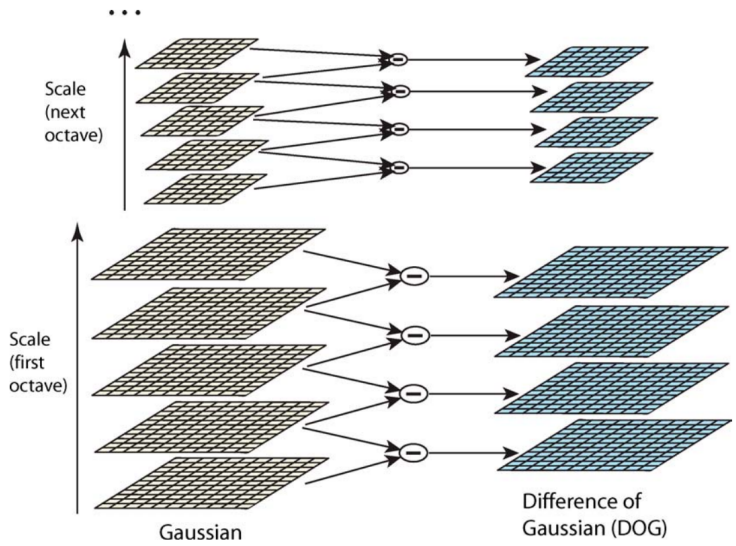


Figure 2.3: SIFT Pyramid Schematic Diagram [37]

As Figure 2.3 shows, for each octave, there are several images with the same size but in variant scale-space. After the pyramid is built, the important feature points' locations are be extracted by using Difference of Gaussian (DOG) algorithm which relies on the following

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma), \end{aligned} \tag{2.23}$$

where, $D(x, y, \sigma)$ is the DOG function. The key feature points are then find in the DOG space. In order to find local extreme points, a comparison among one point and its 26 neighbours is introduced.

As figure 2.4 shows, the extrema is found among a $3 \times 3 \times 3$ area, where contains eight adjacent points from the same scale-space, and other 18 adjacent points from the neighbour scale-space. The key points found by local extreme method are then be further selected and adjusted. The key points' locations are shifted in order to increase the stability. The offset

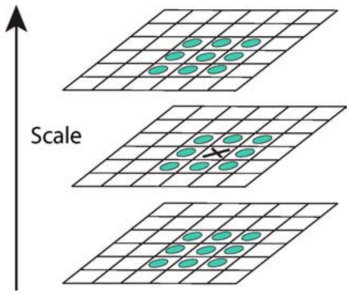


Figure 2.4: SIFT Local Extrema Detect [37]

shifted is calculated by expending the Taylor expansion of DOG functions and find the zero gradient point, which is a kind of interpolation methods [**?**]. The points with edge responses caused by DOG is eliminated by applying a threshold which is calculated from a 2×2 Hessian matrix.

When the key points are adjusted, the magnitude $m(x, y)$ and the orientation $\theta(x, y)$ of key points are calculated with the similar way in HOG, which are represented with the expressions

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}, \tag{2.24}$$

$$\theta(x, y) = tan^{-1} \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}. \tag{2.25}$$

The histogram is used to statistically analyze the direction of the pixels and decide a main and sub directions of the key points. There features are finally ready for being used in image comparison or object tracking etc.

2.3.3 Features from Accelerated Segment Test (FAST)

Features from Accelerated Segment Test (FAST) algorithm is a feature point detection algorithm, which does not involve the feature description of feature points. The specific feature point description can be realized by cooperating with other algorithms. [51]

The proponents of FAST define FAST corner as if a pixel has a significant difference from enough pixels in its surrounding neighbourhood, and the pixel may be a corner.

As shown in Algorithm 2 and Figure 2.5, in order to decide whether a given pixel is a FAST feature point, a Bresenham circle with radius of 3 is created. There are 16 pixels on this Bresenham circle as Figure 2.5 shows. The first step is compare the intensity of pixel 1 and pixel 9 to the chosen pixel p_i with the threshold θ. If the difference less than θ, the chosen pixel p_i cannot be a feature point and is discarded directly. If the difference is larger than θ, the chosen pixel p_i is compared to 2 more pixels. Calculate the intensity differences between $p_1, 5, 9, 13$ and p_i. If 3 of 4 are larger than θ, the chosen pixel p_i comes to the final checking. Compare p_i with

Algorithm 2: Features from Accelerated Segment Test

Input: The pixels p_i in input image I, the threshold θ of FAST intensity.

Output: FAST feature points p_i

1: Find 16 surrounding pixels of a given pixel p_i on the Bresenham circle with radius equal to 3.
2: **if** $|p_{1,9} - p_i|$ larger than θ **then**
3: **if** $|p_{1,5,9,13} - p_i|$ 3 of 4 larger than θ **then**
4: **if** $|p_{1,\ldots,16} - p_i|$ 9 of 16 larger than θ **then**
5: p_i may be a corner points
6: **end if**
7: **end if**
8: **end if**
9: Use the non-maximum suppression algorithm to further screen feature points

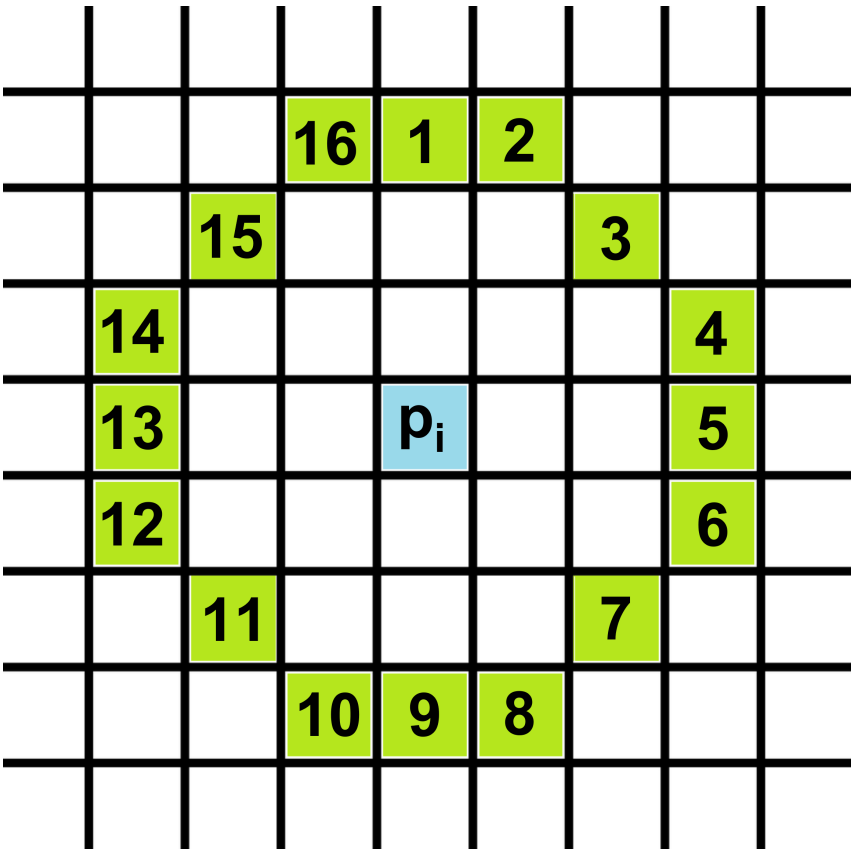


Figure 2.5: FAST Algorithm

all 16 pixels around, if at least 9 of the intensity differences larger than θ, p_i is considered as a FAST feature point.

For the adjacent corner candidate points, the non-maximum suppression algorithm is used to eliminate some points. The remaining corner points are FAST feature points.

Binary Robust Independent Elementary Features (BRIEF)

The Binary Robust Independent Elementary Features (BRIEF) describes the detected feature points. It is a binary coded descriptor. It abandons the traditional method of describing feature points by using a regional grey histogram, which significantly speeds up the establishment of feature descriptors and dramatically reduces the time of feature matching [6]. It is a high-speed and potential algorithm.

The classic image feature descriptors SIFT and SURF use 128 dimensional (SIFT) or 64 dimensional (SURF) feature vectors. Each dimensional data generally occupies 4 bytes, and the feature description vector of a feature point needs to occupy 512 or 256 bytes. Therefore, if an image contains many feature points, the feature descriptor will occupy much storage, and the process of generating the descriptor will be quite time-consuming. In the practical application of SIFT features, PCA, LDA and other feature dimensionality reduction methods can be used to reduce the dimension of feature descriptors, such as PCA-SIFT. In addition, some Locality-Sensitive Hashing (LSH) methods can be used to encode the feature descriptors into binary strings. Then the Hamming distance can be used for feature point matching, which can be realized quickly through XOR operation, which significantly improves the efficiency of feature matching.

BRIEF uses binary coding to generate feature descriptors and Hamming distance for feature matching based on this idea. However, because BRIEF is only a feature descriptor, feature points must be detected and located in advance. For example, feature points extracted from FAST, SIFT and SURF algorithms can be used.

Take the feature point as the centre and take the window of $S \times S$. The pixels in the window are convoluted with $\sigma = 2$ Gaussian kernel.Then, n_d groups of point pairs are randomly selected at location (x_i, y_i) in this sampling window. Then compare the intensity values of these point pairs. If $I(x_i) > I(y_i)$, it is encoded as 1, otherwise, it is encoded as 0. In this way, a binary coding string of specific length n_d is given, that is, the BRIEF feature descriptor.

Five sampling methods of random point pairs are tested in [6]. The best results are obtained through experiments by using the sampling method conforming to Gaussian distribution $[0, S^2/25]$ for X and Y. Figure 2.6 [6] shows the point pairs used when $n_d = 128$.

BRIEF algorithm reduces the storage space requirement of features and improves the speed of feature generation by detecting random sampling point pairs and establishing feature

Figure 2.6: BRIEF Sampling Point Pairs with $n_d = 128$

Algorithm 3: Binary Robust Independent Elementary Features

Input: the given feature point p, the size of search window S, the number of point pairs n_d.

Output: BRIEF Descriptor

1: Apply Gaussian smoothing to window $S \times S$.
2: Random select n_d point pairs (X, Y)
3: **for** $i = 1, .., n_d$ **do**
4: **if** $I(x_i) > I(y_i)$ **then**
5: $\tau(p; x_i, y_i) = 1$
6: **else**
7: $\tau(p; x_i, y_i) = 0$
8: **end if**
9: **end for**
10: BRIEF Descriptor $= \tau(p; X, Y)$

descriptors by binary coding. The measurement of Hamming distance is convenient for fast matching of feature points. The Hamming distance of mismatched feature points will be much greater than that of matching points.

However, BRIEF algorithm does not have scale invariance and rotation invariance. Therefore, matching accuracy decreases significantly after the image is scaled and rotated.

2.3.4 Oriented FAST and Rotated BRIEF (ORB)

Oriented FAST and Rotated BRIEF (ORB) combines the FAST feature point extraction algorithm and BRIEF feature description algorithm and modifies them to become an algorithm with feature extraction and feature description like SIFT and SURF [52].

The FAST features have no rotation invariance and scale invariance. Therefore, the ORB algorithm uses a multi-scale image pyramid to create a multi-scale representation for a single image as shown in Figure 4.3. The pyramid consists of a sequence of images, all of which are versions of images with different resolutions. Each layer in the pyramid contains a lower sampled version of the image than the previous layer. The FAST feature point extraction algorithm will extract the key points on different scales when the multi-scale pyramid is created. By detecting the key points of each scaled image, the ORB algorithm effectively locates the feature points on different scales. Through this method, the ORB realizes scale invariance [52].

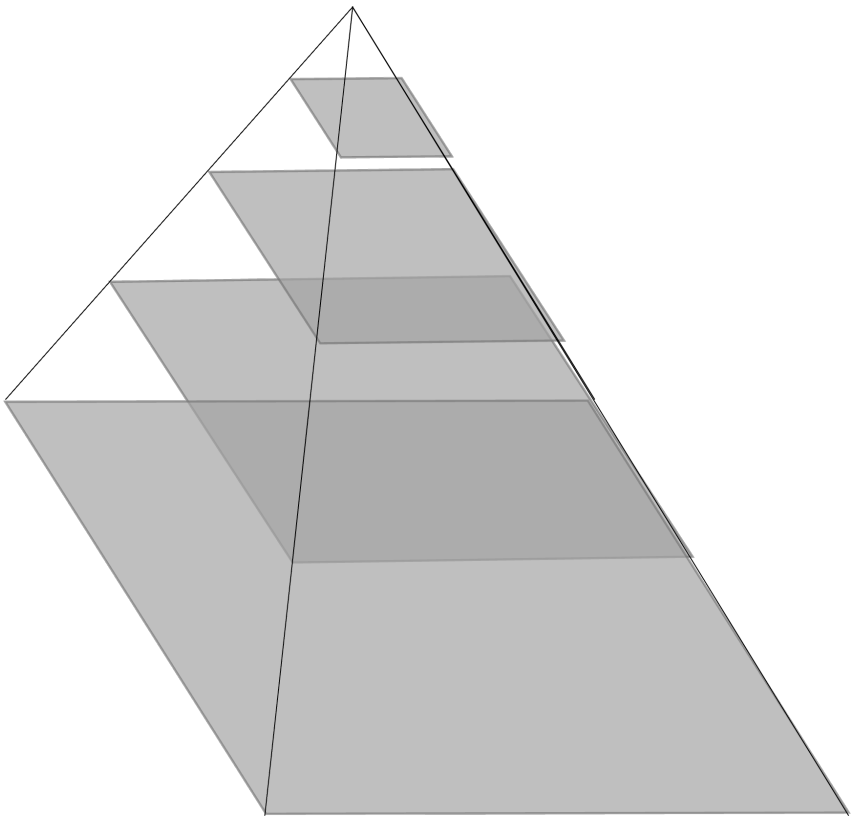


Figure 2.7: Pyramid Sampling

In order to make the algorithm rotation invariant, the ORB algorithm calculates the

intensity centroid of the current feature point patch. Then, the direction vector of the current feature point is from the point to the intensity centroid. According to Equation (2.26), the geometric moment m_{pq} of a certain patch around feature point is defined [50] as

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y), \tag{2.26}$$

where x and y are the coordinate in the specific patch, $I(x,y)$ is the intensity level at pixel (x,y) and p,q should be 0 or 1.

The coordinate of the centroid is then be calculated by the equation Eq. (2.27)

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right). \tag{2.27}$$

After calculating the centroid of the patch, the feature point is given a direction by creating a vector from the feature point to the centroid. From the above, we can know that the BRIEF algorithm chooses the points according to the X, Y axis of the image. Therefore, if the image content rotates, but the coordinate axis remains unchanged, the BRIEF descriptors for the same feature points are different. This situation will cause the failure of feature point matching.

The ORB algorithm rotates the coordinate axis of the brief descriptor to make it consistent with the direction of the centroid line of the feature point to make the brief descriptor obtain rotation invariance. Make the BRIEF descriptor produce the same or approximate description of the rotated feature points. The rotation angle θ is defined as follow:

$$\theta = \operatorname{atan} 2 \left(m_{01}, m_{10} \right) \tag{2.28}$$

ORB algorithm also optimizes the BRIEF descriptor in other aspects to improve the efficiency of the original algorithm.

In order to fuse the images of two cameras, it is necessary to convert the image of one camera to the perspective of another camera. Since the factory's shooting environment and lighting problems, using the images' intrinsic feature points is not very good. The shooting interval between two cameras leads to a significant difference in the subject's position in the images, further increasing image registration difficulty. To solve this problem, extrinsic feature points are used by the system. Checkerboard is used in the camera calibration process, and it is also used as external feature points for image registration.

2.4 Image Fusion

Image fusion [15, 20, 26, 27] refers to the use of different sources of image information collected for the same target to aggregate and filter to produce higher-quality or user-friendly images.

The image source can be from different types of cameras, such as infrared and optical cameras, or can also be using different lenses, focal lenses, or shooting angles. The ultimate goal is to improve the inadequacy of a single sensor, improve the sharpness and information content of the resulting image, and help users or computers obtain more accurate, reliable and comprehensive information about the target or scene. Image fusion can be divided into three levels from the information dimension, pixel-level fusion, feature-level fusion and decision-level fusion.

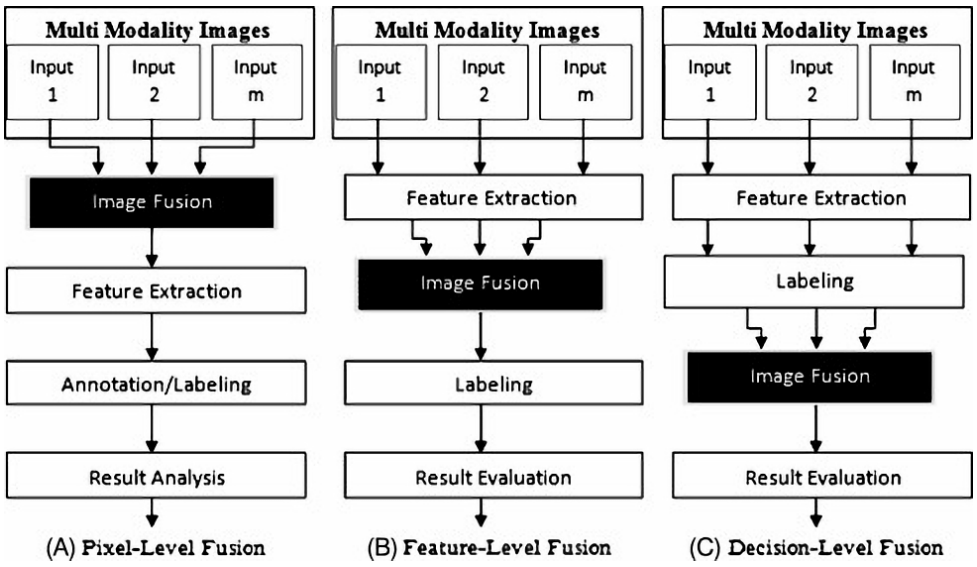


Figure 2.8: Different Fusion Levels [56]

Pixel-level image fusion refers to the process of directly processing data collected from multiple sensors to obtain a fused image. It is expected to provide more information for human or machine perception than any input image. Because of this advantage, pixel-level image fusion has achieved significant results in remote sensing, medical imaging and other fields. Pixel-level image fusion uses information from multiple sensors to aggregate in one image, often enhancing details such as edges, corners, and textures. It is helpful for subsequent image processing, such as edge extraction and target recognition. The amount of data in the fused image is much higher than the initial image, which can only be achieved by pixel-level image fusion. Feature-level and decision-level image fusion can cause the image to lose some details.

Feature-level image fusion is to extract the feature information from the source image first, then analyze, process and integrate the feature information to get the fused image features. These image features can be edge information, high-brightness parts, or parts of the image that the viewer is interested in. After feature-level image fusion, only feature information is preserved in the fused image, while the details in the original image are discarded. Because feature-level image fusion filters information and compresses image information before fusion, the computational speed of feature-level image fusion is also faster than pixel-level fusion. Fusion will be relatively real-time.

Decision-level image fusion is a cognitive-based method. It is not only the highest-level image fusion method but also the highest level of abstraction. According to the specific requirements of the question, the feature information obtained from the feature-level image is used. Then the decision is made directly according to a specific algorithm.

Feature-level and decision-level image fusion are generally designed for specific practical purposes, and the design of fusion rules is directly serving the project objectives. Pixel-level image fusion has been developed as a basic way to fuse the most information. Next, several different kinds of pixel-level image fusion algorithms are introduced.

Generally, pixel-level image fusion methods are divided into two categories: spatial domain and transform domain according to domain choice. However, this classification method is too general. Therefore, according to the fusion strategy, the pixel-level image fusion methods can be further divided into four categories [32, 71]:

- Spatial Domain Methods: Directly manipulating pixels.
- Multi-Scale Transformation: a multi-scale and multi-resolution transform domain method for processing image transformed coefficients.
- Model-Based Fusion Method: Specific mathematical feature extraction models for fusion.
- Hybrid: The combination of two or more methods.

2.4.1 Spatial Domain

The fusion process of the image fusion method based on the spatial domain directly operates the pixel values of the image. The most straightforward fusion strategies are the average, minimum, maximum, max-min and weighted average methods. These methods are simple in the calculation, fast in operation, and can better preserve the overall effect of the resulting image. However, the details such as edges and contours are seriously lost. Therefore, people have proposed activity level methods based on information statistics, such as using spatial frequency to complete image fusion [33]. Furthermore, considering the correlation of pixels in the local region, image fusion based on region segmentation also came into being, significantly improving the extraction effect of detailed information. For example, [3] used a differential evolution algorithm to adapt the fusion method to determine the size of image blocks and achieved good fusion results.

2.4.2 Multi-Scale Transformation

The processing object of transform domain fusion is the decomposition coefficient of the source image after transformation. The fusion process mainly includes three steps:

- The transform algorithm decomposes the source image into high and low frequency coefficients.

- Different fusion strategies are adopted for different coefficients, and the fusion is completed in layers and sub directions.

- Image fusion is realized by inverse transform.

Pyramid transform is the first multi-scale transform method used in image fusion. A series of image sets with gradually reduced resolution obtained by 2-step sampling and low-resolution coefficients and high-resolution coefficients obtained by decomposition highlight the essential features and details of the image. Many pyramid algorithms, such as Laplacian pyramid LP, contrast pyramid, gradient pyramid and morphological pyramid, have achieved good fusion results. The fusion method based on pyramid transform has high computational efficiency and ideal fusion effect and is still widely used. However, pyramid decomposition also has the following disadvantages: redundant decomposition and non-directionality. Furthermore, with the gradual increase of the decomposition layer, the resolution will become smaller and smaller, and the boundary will become more and more unclear.

Subsequently, wavelet transform was introduced. It decomposes the image into low-frequency approximate coefficients representing the contour and high-frequency detail coefficients representing the image details in three directions (horizontal, vertical and diagonal), which fully reflect the local variation characteristics of the source image. The advantage is that the decomposed information has no redundancy and directionality, which overcomes the shortcomings of the pyramid transformation method. However, the importance of wavelet translation invariance is verified in the multi-scale transform experiment. The method without translation invariance does not work well when matching incomplete images [34]. Therefore, scholars have proposed improved wavelet transforms, such as multi wavelet, dual-tree complex wavelet transform, contour wave, curve wave and shear wave. They not only have translation invariance but also have direction selectivity.

2.4.3 Model-Based

Because the traditional multi-scale transformation method uses predefined fixed vectors to extract features, such as spatial frequency and gradient energy, it lacks the generalization of features. In order to obtain adaptive features, some new mathematical models for adaptive extraction of image features are proposed, such as sparse representation (SR), pulse coupled neural network (PCNN), and the newly proposed deep learning model like convolutional neural network (CNN), generative adversarial network (GAN).

Sparse Representation

Sparse representation is a method to effectively decompose the image into a group of non-zero atoms while preserving the details of the image. An overcomplete dictionary and sparse representation model are the core content of the sparse representation. Generally speaking, there are two ways to obtain overcomplete dictionaries. First, for a specific type of image, using the existing ancient signal model to construct atoms is simple and easy to implement. The second is to use learning methods, such as singular value decomposition K-SVD algorithm and PCA, to train a large number of experimental samples to construct a dictionary, which is a self-learning dictionary with higher redundancy.

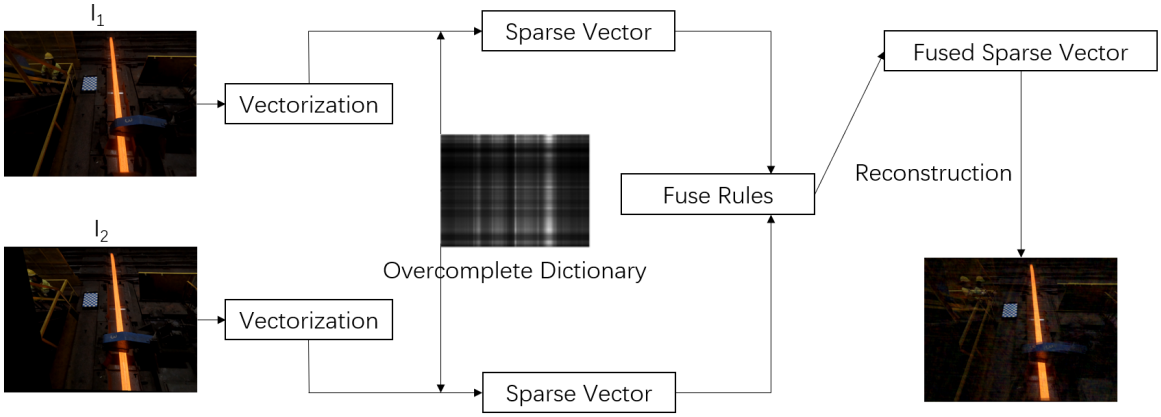


Figure 2.9: Sparse Representation Fusion

The process of spares representation image fusion is:

- Construct an overcomplete dictionary
- Convert the source images into single-scale feature vectors which linearly combined by atoms in the dictionary according to the overcomplete dictionary
- Fused the feature vectors based on fusion rules and activity level
- Reconstruct the vectors to image

[66] is the early use of sparse representation for multifocal image fusion. In the paper, a multi focus image fusion method based on sparse representation is proposed. Compared with some transform domain fusion methods, better fusion results are obtained.

The advantage of sparse representation is that the model construction is simple, easy to understand, and the processing of noise error is ideal. However, the sparse representation method has high complexity and low computational efficiency, blurring the details of the source image, such as edges and textures.

Convolutional Neural Network

In 2012, Alex Krizhevesky participated in the ImageNet image recognition competition and won the championship through his CNN network 'AlexNet' [29]. After that, CNN has attracted the attention of many researchers.

When using traditional methods to deal with camera based problems, feature points are extracted from the image, and then the extracted features are used for modelling the classifier, and the results are output through the classifier. In the deep learning approach, the problem is solved by using an end-to-end black box approach. The entire process from extracting features to making the final output is completed within the same network.

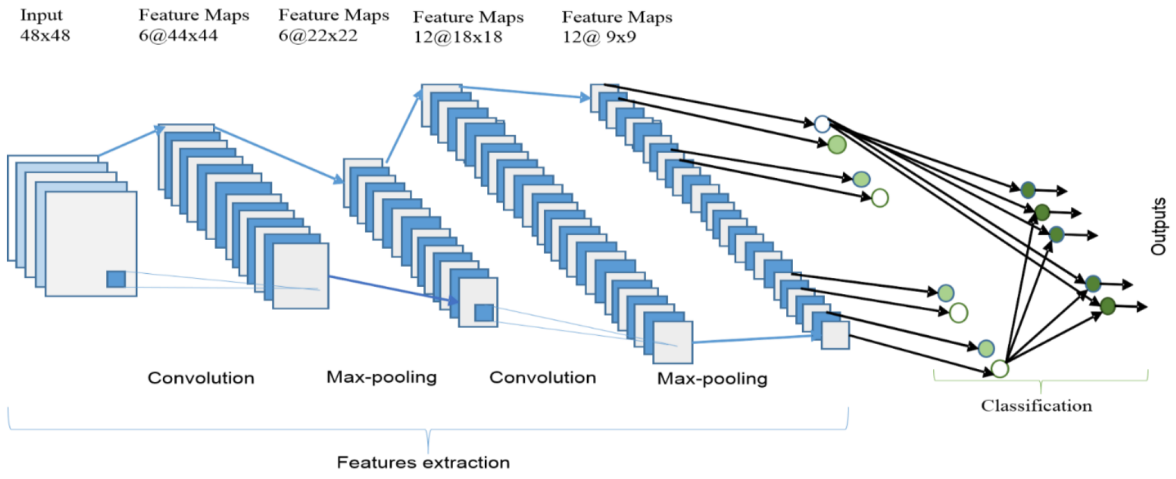


Figure 2.10: The Structure of a CNN [2]

Figure 2.10 shows the structure of a convolution neural network. A convolution neural network contains a input layer, several convolution-pooling layers, a series of classification layer.

At convolution layer, a convolution kernel convolved with part of previous inputs in sequence. For the first convolution layer, the original input image is convolved with the kernel. For the other convolution layer, is the feature map.

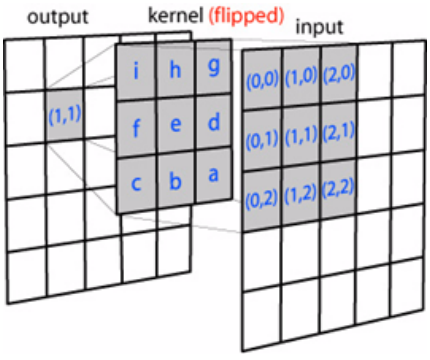


Figure 2.11: Convolution Kernel

Figure 2.11 shows how the convolution kernel works in convolution layer. The output of a convolution layer is called feature map. The kernels used in convolution neural network are learnable, and there are always more than one kernels used to extracted features. Different kernels are used to extract different kinds of features. Since the kernels are trained during training the network, the preset values of kernels need to be decided to make the kernels become independent with each other.

The convolution layer is followed by a pooling layer. Pooling layer can be seen as a subsampling process. For a max pooling layer, the output will be the maximum number among the pooling window.

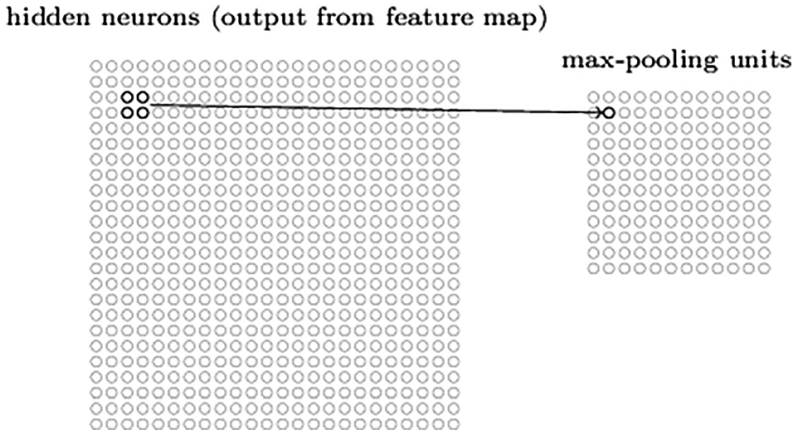


Figure 2.12: Pooling Layer [54]

As figure 2.12 shows, the size of feature map will decrease after processed by pooling layer. Through downsampling by pooling layer, the dimension of feature map significantly decreases, which decrease the computation cost in the following layer. At the same time, downsampling can increase the robustness of the network, since some noises are eliminated at pooling layer.After the features are extracted from the previous convolution and pooling layers, the features are passed to the classification part of CNN.

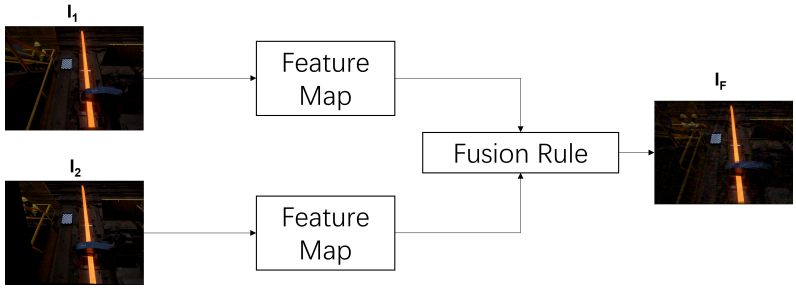


Figure 2.13: CNN Fusion

Convolutional neural network CNN is one of the most popular models in image processing. As a deep learning model, CNN is a high-speed and efficient parallel computing based on GPU. Its feature extraction is data-driven. After a large number of data samples are trained, param-

eter values are automatically generated. Usually, the order of magnitude is tens of thousands. Therefore, the features extracted by CNN fusion method have strong generalization. And with the deepening of the network, the influence of physical characteristics is gradually abandoned. Its features are more and more abstract and accurate, with translation, rotation and scaling invariance. However, the training effect of CNN is determined by the data set samples, and the model usually trains specific images. The effect of CNN may not be ideal when encountering images or environments that are not in the training set. At the same time, training set labeling and training is a time-consuming and laborious process.

2.5 Summary

This chapter primarily delves into pertinent literature methods. Initially, it provides an overview of the iron and steel manufacturing industry, which holds a vital role in contemporary industry. Subsequently, several widely-used vision measurement algorithms are discussed, including the time of flight method, structured light method, and binocular system algorithm. Given the stringent environmental requirements of both the time of flight and structured light methods, their performance is subpar in the extreme conditions of steel plants. Hence, capturing steel images with a camera and conducting measurements is proposed as an effective strategy for steel size measurement.

To pinpoint the location of steel within an image, the deployment of an edge detection algorithm becomes indispensable. Section 2.2 outlines various edge recognition algorithms, ranging from edge detection algorithms based on a differential operator to the contemporary Canny edge algorithm and structured random forest algorithm. As inferred from the experimental results, the edge detection algorithm of the structured random forest outperforms other methods in terms of edge effects.

When a single image falls short of requirements, information from multiple images is integrated. The preliminary step for image fusion is image registration. Section 2.3 introduces numerous distinct feature extraction and registration algorithms. Through the extraction of feature points in the image and the calculation of the gradient direction of these points, pairing of feature points is achieved, facilitating image registration.

Once the image is registered accurately, image fusion technology can be employed to blend the information within the image. Section 2.4 discusses image fusion algorithms at different pixel levels, starting from the most straightforward spatial domain method to multi-scale transformation fusion, and then to various fusion methods utilizing models.

Chapter 3

The Developed Edge Detection Algorithms for Steel Remote Sizing

3.1 Practical Industrial Task

Speciality steels rolling plays a critical role in industrial sectors such as aerospace, oil and gas. Digitalisation of the rolling process is considered as a key to the long-term viability of steel plants. The quality standards for *ingots*, as well as steel sections which are manufactured from

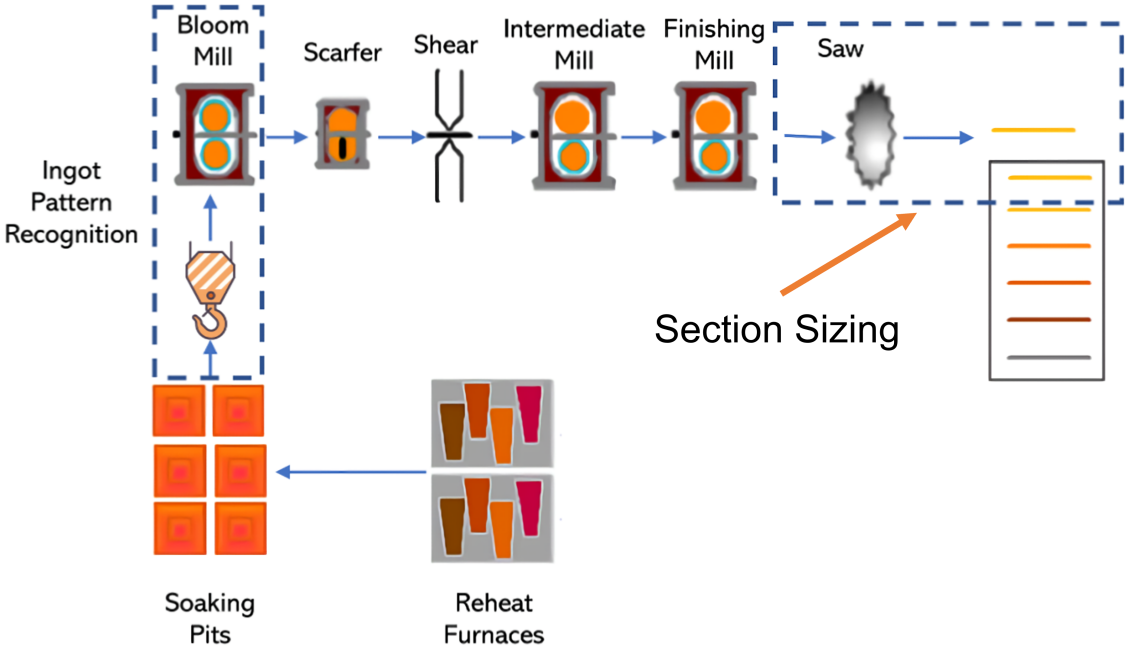


Figure 3.1: Steel rolling schematic diagram

ingots, strictly set an upper limit on the final quality of all products in downstream industries. Despite the wave of digitalisation, a lot of nowadays steel rolling plants are still heavily relying on human operators to control and monitor the manufacturing/rolling process. It has been shown that the long-term exposure to a high-temperature, intense light environment in steel

factories could cause injuries, particularly to eyes [25].

In the process of automatic rolling, the real-time and automated non-contact detection and measurement of steel sections on the production line can be taken as a way to assure quality and eventually avoid hazards for operators and financial loss due to errors during the manufacturing process. For instance, Zhou et al. [73] propose an approach for online diameter estimation of hot forgings. A fast measurement method based on feature line reconstruction of stereo vision is developed to increase the calculation speed. A considerable measurement accuracy is demonstrated, but the approach requires good lighting conditions and high-precision camera calibration.

A structural diagram of the whole production system is shown in Figure 3.1. This dissertation focuses on the starting and the finishing stages, i.e., providing machine vision techniques for the hot steel section direction recognition stage (upper left corner) and the section sizing stage (upper right corner). Ingots are reheated and moved to the rolling line, where steel sections will be processed by a few mills to change their size and shape. Dimension measurement during rolling plays a key factor for quality assurance, and it should be performed wherever necessary. In the considered industrial case study, sections are measured after the finishing mills. Laser range finder measurements are provided only at the last mill and these measurements will be used as ground truth to assess the performance of the developed computer vision remote sizing measurement system.

The initial plan is to use a monocular camera and perform a vision measurement task without pre-calibration. The edge of the conveyor belt of the production line is used as a reference to calculate the steel section's size indirectly. Therefore, the edge detection task should be split into two parts. The first part needs to extract the contour of the conveyor belt of the background production line, and the second part is to extract the edge of the high-temperature steel section on the production line.

Figure 3.2 shows the production line in the industry. It can be seen from the image that the illumination on the whole production line is not uniform. Complex illumination conditions also bring great difficulties to edge recognition. The two green lines in the figure identify the edges at the left and right ends of the production line and is also the boundary planned to be used as the visual measurement reference. An ideal edge detection algorithm needs to display these two boundaries as completely as possible. At the same time, the red lines in the figure identify the edges of the conveying rollers on the production line. Extracting their edges can help correct the image and keep the image and the production line horizontal.

Figure 3.3 shows the results of identifying the production line background map using Sobel edge detection. By adjusting the algorithm's threshold, the quality and quantity of the recognized edges are quite different. When a large threshold is used in (a), the recognized edges

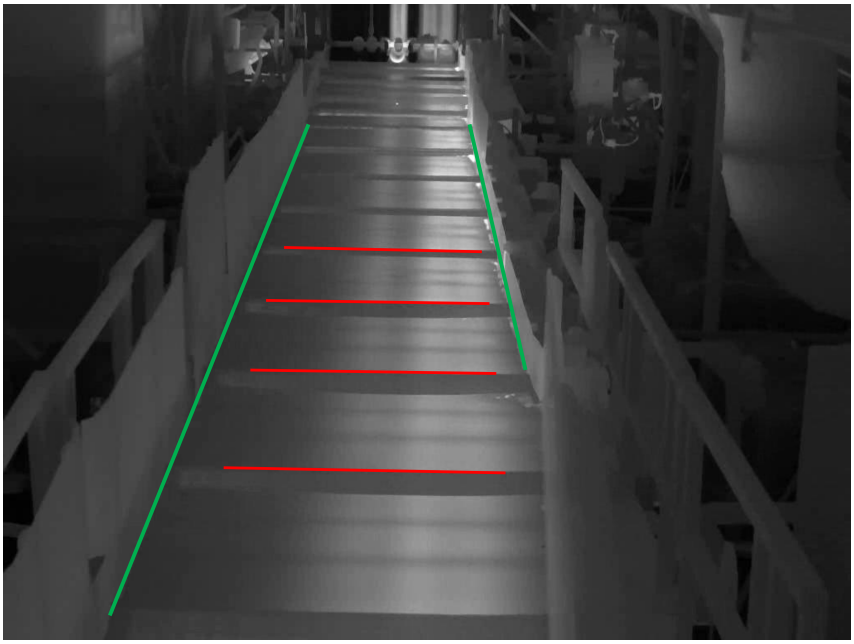
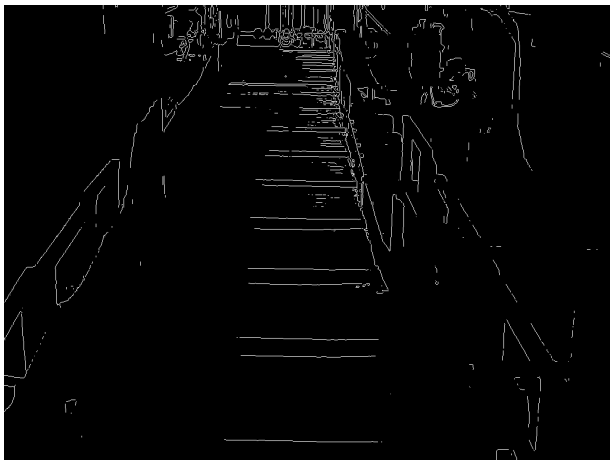


Figure 3.2: Background Production Line

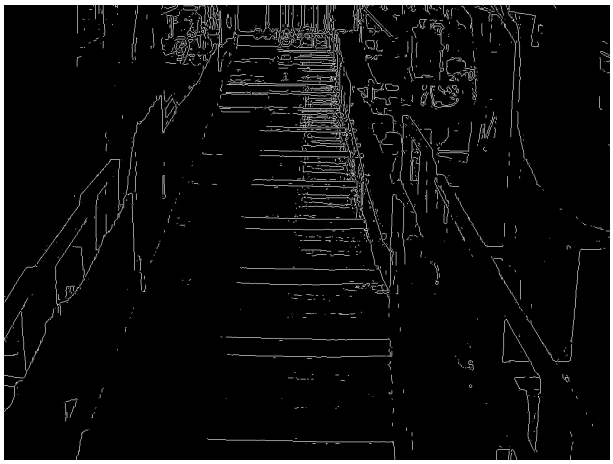
are all composed of pixels with high gradient intensity, and the recognized edges are true edges. However, most edges were not recognized, especially the left edge of the production line, which was completely ignored. As the threshold is lowered, in (b), the edge on the left side of the production line begins to be recognized, but then the noise is spread over the upper right part of the image. As the threshold value decreases again, the left edge can be connected into a line, but the increase of noise pixels makes the edge quality of the right half poor. And real edges and noise become indistinguishable.

Figure 3.4 shows the results of identifying the production line background using Canny edge detection. Compared with Sobel edge detection, the edge result generated by Canny looks more integrated. Because of the characteristics of the algorithm, the edges of Canny are basically connected. When the higher threshold $[[0.04, 0.1]$ is used, the edge on the left side of the production line is still not recognized, but a considerable part of the edge of the conveyor drum on the production line is recognized. In figure (b), when the threshold value reaches $[0.02, 0.5]$, the edge on the left side of the production line is relatively completely recognized. There are fewer breakpoints in the middle with the help of non-maximum suppression. And the right half edge of the image begins to generate some noise textures. When the threshold value is further lowered, although there are no separate noise pixels in the image, the wrong edges generated by the noise are connected and generate more textures. However, at this time, almost all required edges have been detected.

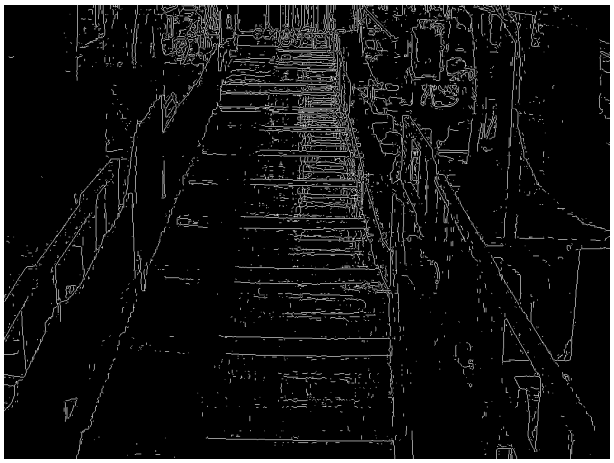
Figure 3.5 shows the image of hot steel on the production line taken by the thermal imaging camera. As the temperature of high-temperature steel is far higher than that of the surrounding environment, the difference between the brightness of steel and the surrounding



(a)

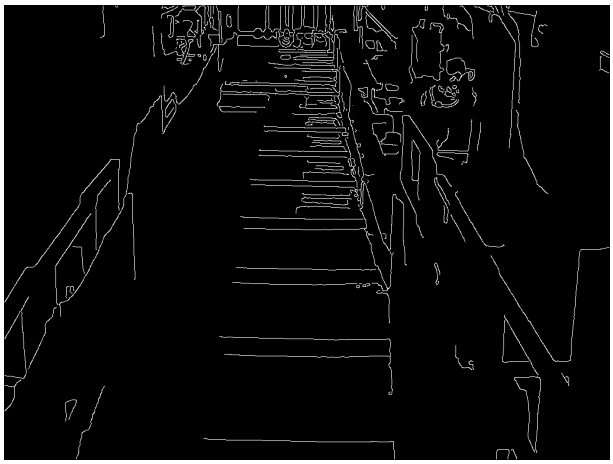


(b)

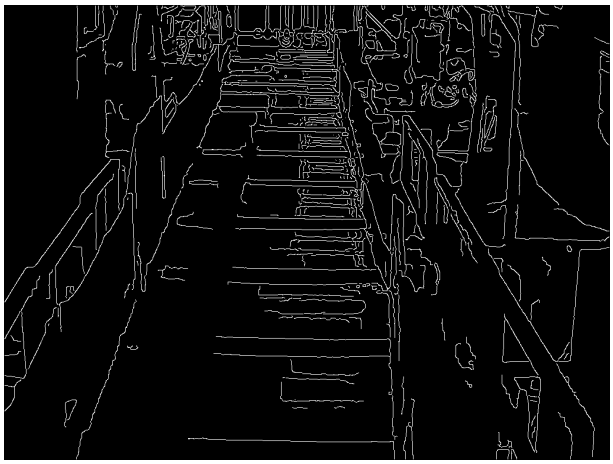


(c)

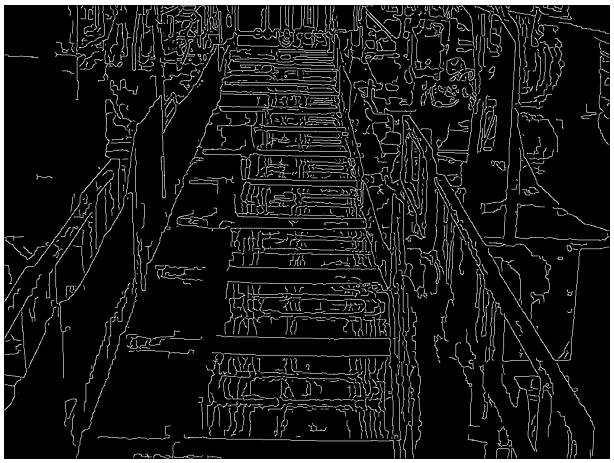
Figure 3.3: Sobel edge detection of background with threshold (a) $\theta = 0.029$, (b) $\theta = 0.0015$, (c) $\theta = 0.01$



(a)



(b)



(c)

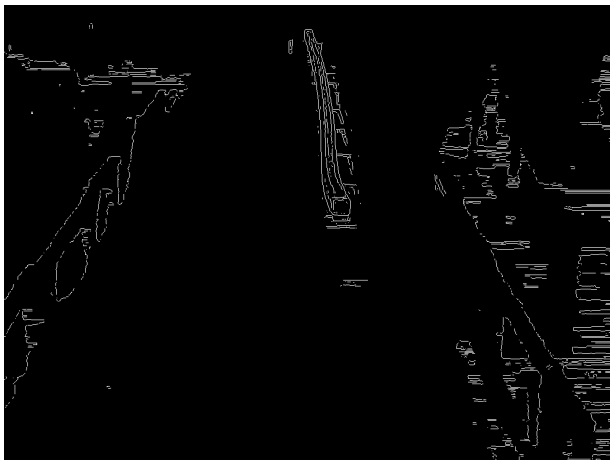
Figure 3.4: Canny edge detection of background with threshold (a) $\theta = [0.04, 0.1]$, (b) $\theta = [0.02, 0.5]$, (c) $\theta = [0.012, 0.03]$

Figure 3.5: Steel Section (Thermal Cam)

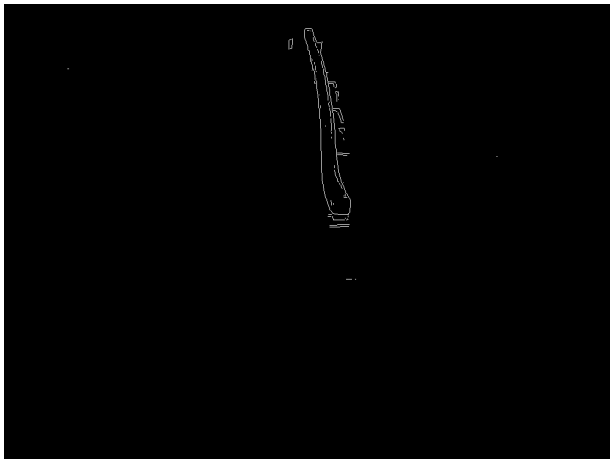
environment is noticeable. Therefore, the edge detector will have different performances in such different environments.

Figure 3.6 presents the edges of high-temperature steel extracted using Sobel. In this task, the edges other than the high-temperature steel section should be filtered out, and the edges of the steel should be completely retained. When a low threshold is used, as shown in figure (a), the detected edge can clearly find the position of the steel, but there are many wrong edges on the steel, and many edges are detected on the left and right of the production lines. Figure (b) shows that with the increase of the threshold, the edges on the production line are basically eliminated, leaving only the edges near the steel, but the edges are not clean, and there is some noise. If the threshold is further raised, the noise edge around the steel will be less and less, but if the threshold is too high, the steel edge will be broken and incomplete.

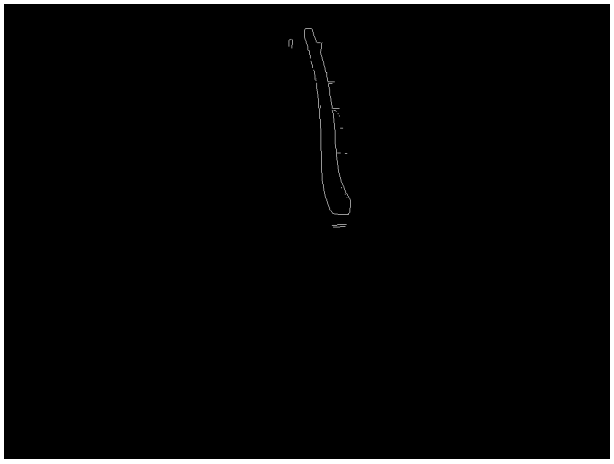
Figure 3.7 uses the Canny algorithm for edge detection. Like the Sobel algorithm, too many erroneous and cluttered edges are detected at lower thresholds. Because of the large gradient of brightness change near the high-temperature steel section, many edges are defined as strong edges, and many weak edges connected to strong edges are misidentified. At low thresholds, the edges of the Canny algorithm appear less effective than those of Sobel. As the threshold value increases to $[0.04, 0.1]$ in figure (b), the noise decreases and the detected edges are relatively concentrated. However, the edges near the high-temperature steel section are still somewhat cluttered. When the threshold value is reached to $[0.22, 0.55]$ in figure (c), the edges of hot steel are well extracted, and only a few edges are detected incorrectly. Furthermore, because of the Canny algorithm, steel edges will not break or be missing as long as the threshold is not raised too much.



(a)



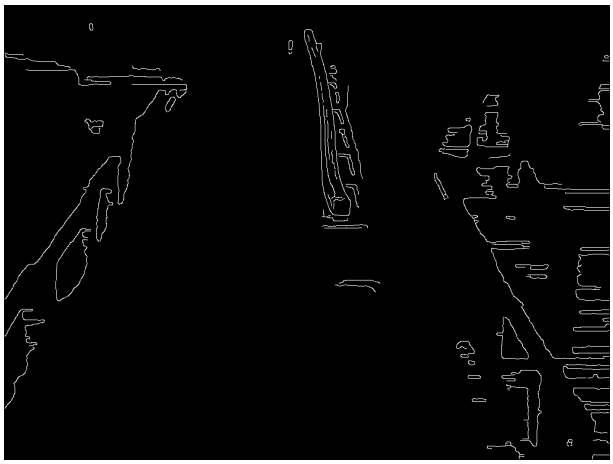
(b)



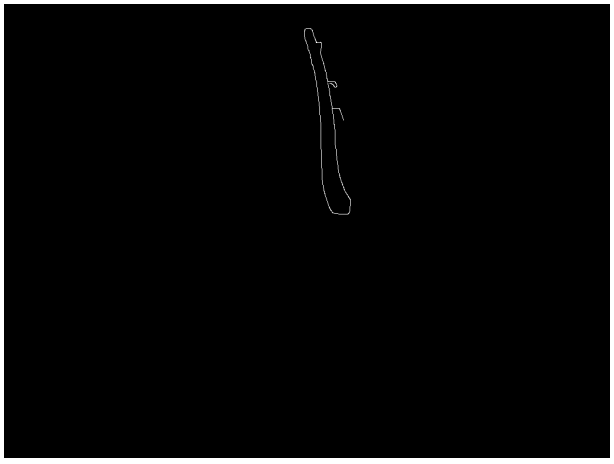
(c)

Figure 3.6: Sobel edge detection of steel section with threshold (a) $\theta = 0.02$, (b) $\theta = 0.05$, (c) $\theta = 0.09$

(a)



(b)



(c)

Figure 3.7: Canny edge detection of steel section with threshold (a) $\theta = [0.0064, 0.016]$, (b) $\theta = [0.04, 0.1]$, (c) $\theta = [0.22, 0.55]$

Although the traditional edge detection algorithm can also obtain acceptable results when a certain threshold is selected, the selection of the threshold is not adaptive but is adjusted by human judgment. When the ambient lighting conditions change, the previously selected threshold will no longer apply, so it is inappropriate to use Sobel and Canny or other similar edge detection algorithms in this case.

Figure 3.8: Structured Forest Background Edge Detection

Figure 3.8 uses the structural random forest algorithm for edge recognition. Different confidence levels of the algorithm generate the edges of different intensities in the graph. Edges with higher confidence levels are brighter, and lower confidence edges are darker. Almost all background objects are recognized in the image, except no edges are identified on a portion of the left side of the production line where the pixel intensities do not have a significant difference. The centre pixels of the edges identified by the structured random forest algorithm have a high confidence level, and the surrounding connected pixels will have some low confidence to form a wide edge with the true edge. At this point, non-maximum suppression is used to thin the edges to locate the true edges. Figure 3.9 shows that the edges identified become sharper when they become one pixel thick using non-maximum suppression.

However, because some edges are weak (pixels with lower confidence level), they appear pale in the image. Filter and capture the desired edges by binarizing the image. For a production line background map, as many edges as possible need to be identified, especially the left edge. The binarisation threshold is set to 2%, and the edge of the lower 2% confidence level is ignored, as shown in Figure 3.10.

The binarized edge graph produces some noise because of the low threshold set to identify as many edges as possible. Next, the edge image is denoised using a denoising algorithm for

Figure 3.9: Structured Forest Background Edge Detection with NMS

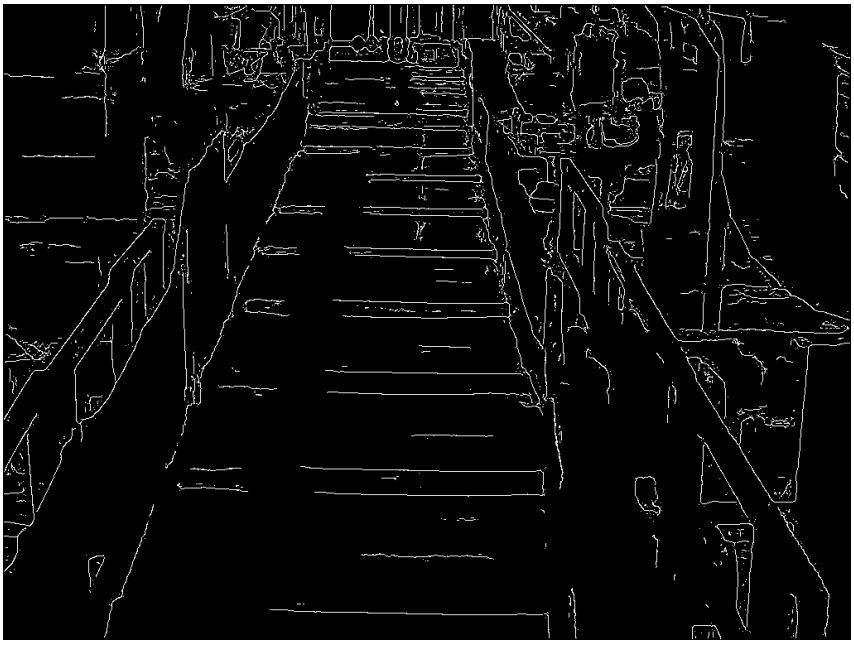


Figure 3.10: Structured Forest Background Edge Detection with Binarization

binary images.

Algorithm 4: Binarized Image Denoising

Input: Binarized Edge Image I_{bw}

Output: Denoised Image $I_{bw_{denoised}}$

```
Get the size [m, n] of I_bw
for 1 < i < m − 1,1 < j < n − 1 do
    if p_ij is edge then
        Check the number n_edge of edge pixels in the adjacent 8 pixels
        if n_edge > 1 then
            p_ij is the edge
        else
            p_ij is not the edge
        end if
    end if
end for
```

Algorithm.4 is the binarization denoising algorithm used in this dissertation. The algorithm runs through the whole image and calculates each edge pixel and its surrounding eight adjacent pixels. If the current pixel is an edge, and there are two or more edge pixels in the surrounding eight adjacent pixels, the pixel is retained as an edge pixel. Otherwise, it is removed. Such an algorithm can effectively eliminate fragmentary noise. The result is shown in Figure 3.11

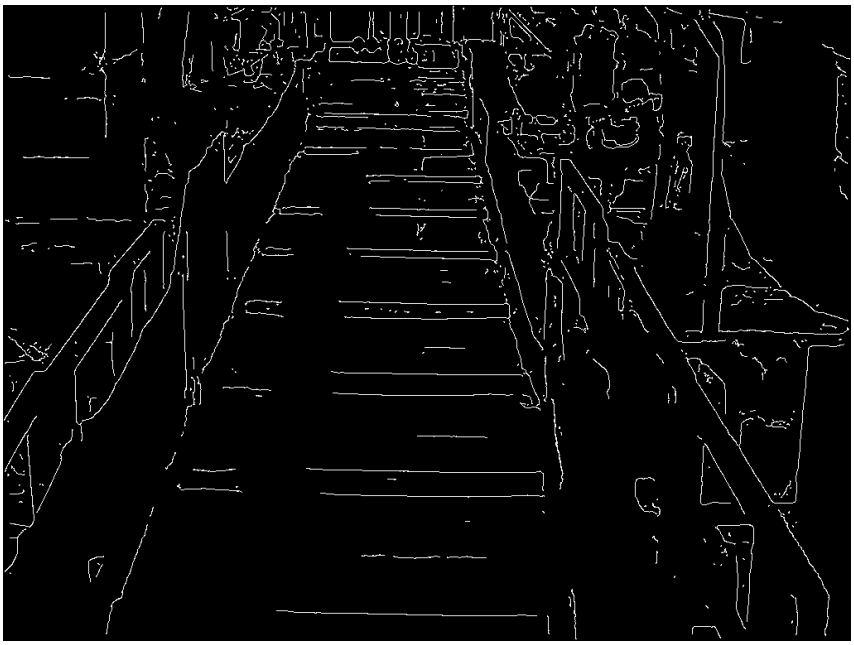


Figure 3.11: Structured Forest Background Binarized Edges with Denoise

Figure 3.12: Structured Forest Steel Section Edge Detection

Figure 3.13: Structured Forest Steel Section Edge Detection with NMS

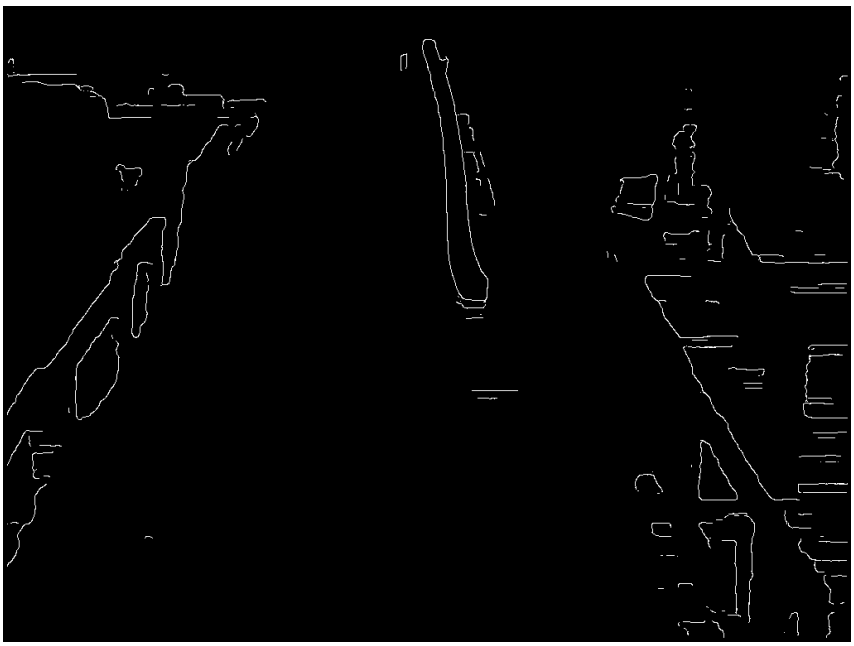


Figure 3.14: Structured Forest Steel Section Binarization Edge Detection with Denoised

Figure 3.12,3.13 and 3.14show the images using the structured random forest edge detection algorithm, non-maximum suppression and noise reduction binarization. Since the edge of high-temperature steel is relatively obvious, the threshold value used for binarization is 20%. It can be seen from the figure that the edges of the steel are completely identified. Although there will be background edges around the production line and some erroneously recognized edges near the section, the interference of these erroneously recognized edges can be eliminated through subsequent recognition algorithms.

3.2 Steel Section Detection

In order to measure the size of the steel section, the target section should be identified and positioned from the image. The detection of the steel section can be classified into two steps: First, extract all edges in the image as shown in previous section; Then, the background is eliminated by the selected area of interest and the colour difference. In these two steps, the steel section edges are extracted completely, and the edge information is used to measure the size of the steel section.

3.2.1 Background Subtraction

In order to measure the size, the background part in the image needs to be removed, and only the steel section part remains. Due to the steel section's high-temperature property, steel's brightness and colour differ from the background in the image taken with optical or infrared thermal cameras. Therefore, the RGB image is first converted into a grayscale image, and then

the grayscale image is binarized using the threshold $thrs$ obtained from histogram information proposed by [46].

Pixels with intensities less than $thrs$ are set to 0 and 1 otherwise. Directly binarizing an image with a threshold value will generate noise or speckle in the binarization result due to noise and some special textures. Therefore, in order to eliminate noise and retain the most significant area, that is, the steel section part, morphological methods are used to process the binarized image.

Algorithm 5 shows the pseudo-code for **Edge Extraction** and **Background Substraction**. I_{rgb} is the input image. It is processed by the structured random forest edge detection algorithm and I_{edge} contains all the edges extracted. To improve the dimension measurement accuracy, the Non-Maximum Suppression (NMS) method is applied to sharpen the edges extracted to one pixel. In parallel, I_{rgb} is first binarised as I_{bw} and then the morphological method is applied on I_{bw} to remove the imperfections caused by thresholding, which results with I_{mor}. The final detected steel section with edges extracted is denoted as $E_{section}$, which is generated from

$$E_{section} = I_{mor} \odot I_{edge} \tag{3.1}$$

where $\odot$ indicates the element-wise production of two matrices.

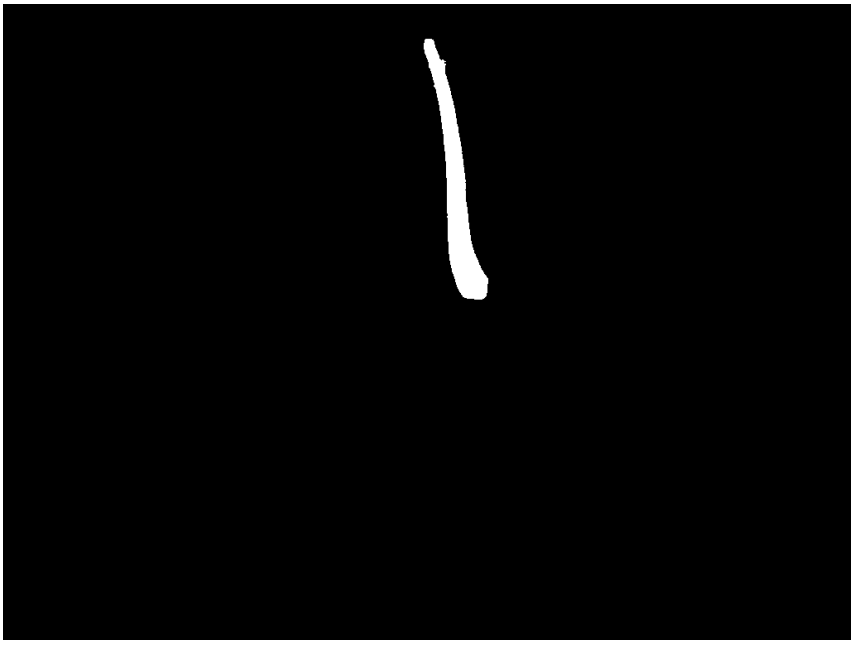


Figure 3.15: I_{mor}

3.3 Mapping from Image Space to Physical Space

3.3.1 Spacial Resolution Information

Algorithm 5: Edge Extraction and Background Subtraction

Input: I_{rgb}

Output: Detected section with edges $E_{section}$

1: Edge Extraction

Structured Forests based Edge Detection

Non-Maximum Suppression $\rightarrow I_{edge}$ //Sharpen edges to one pixel

2: Background Subtraction

Binarise I_{rgb} according to Otsu's method $\rightarrow I_{bw}$ [46]

Morphological denoising $I_{bw} \rightarrow I_{mor}$

3: $E_{section} = I_{mor} \odot I_{edge}$

Usually, the camera is calibrated in visual measurement to obtain the intrinsic (focal length, optical center and skew coefficient) and extrinsic (rotation and translation) parameters of the camera. However, ordinary calibration plates cannot be displayed in the camera due to the use of an infrared thermal imaging camera for shooting. Special calibration plates composed of different materials must be used to display the images required for calibration in the thermal imaging camera. At the same time, due to the complex production environment of the factory, it is impossible to directly place the calibration plate on the production line, which has also become the reason for restricting the use of camera calibration.

To cope with the problem, we extensively explored the data. We found the following two attributes of the videos useful:

• As shown in figure 3.16, the physical distance w between the conveyor barriers (the width of the conveyor along X) is known, which helps to find the physical correspondence of one pixel.

• There is only one vanishing point (the intersection of the two green segments) in figure 3.16, and the objects captured on the conveyor have the foreshortening effect.

According to the space perspective projection as shown in figure 3.17, we can see that the objects of the same physical size seem to be smaller when the distances between the objects and the camera increase. Therefore, though the physical width of the conveyor remains constant, the width of the conveyor in pixels decreases as y increases. The physical size represented by one pixel increases as well. According to the photography triangulation (see figure 3.18), we have the ratio r_i between the physical size w and the pixel number w_{pi} as in

$$r_i = w/w_{pi} \tag{3.2}$$

with w the physical width of the conveyor as shown in figure 3.18, which corresponds to the digital width w_{p0} in the image space with $y = 0$. It can also be regarded as the physical length

represented by one pixel at $y = 0$.

Algorithm 6: Conveyor Boundaries Extraction

Input: I_{eb}

Output: $f_1(x), f_2(x)$

1: Mask to Select Region of Interest

The mask is created by selecting the points around boundaries $\rightarrow I_{mask}$

The Mask is applied to the edge image $\rightarrow I_{ROIedge} = I_{edge} \cdot I_{mask}$

2: Straight Line Fitting

Line fitting in $I_{ROIedge} \rightarrow f_1(x), f_2(x)$

Figure 3.16: Line fitting for conveyor boundaries

The physical width of the conveyor is known. In order to calculate r_i, we need to extract the conveyor boundaries correspond to the barriers, as shown in figure 3.16 to get w_{pi}. The two boundaries extracted are represented by $f_1(x)$ and $f_2(x)$, respectively. For a given $y = i$, the width w_{pi} of the conveyor in pixels can be determined.

Then we use equation (3.3) to calculate the physical size p_s of any steel sections.

$$p_s = r_i \cdot w_{si} \tag{3.3}$$

where w_{si} is the width of the section in pixel at $y = i$.

3.3.2 Dimension Measuring Algorithm

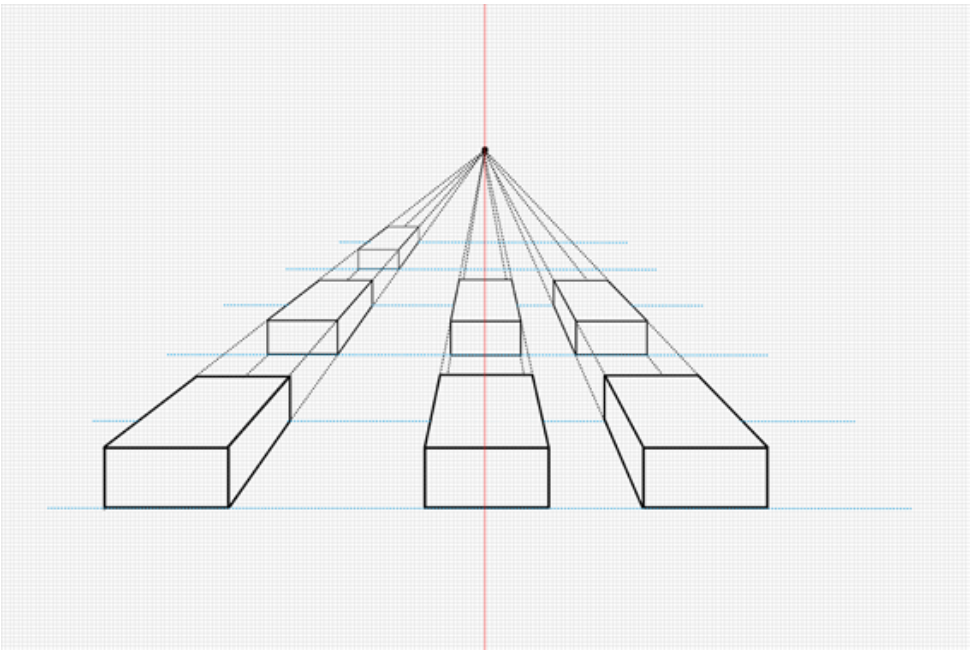


Figure 3.17: Space perspective projection

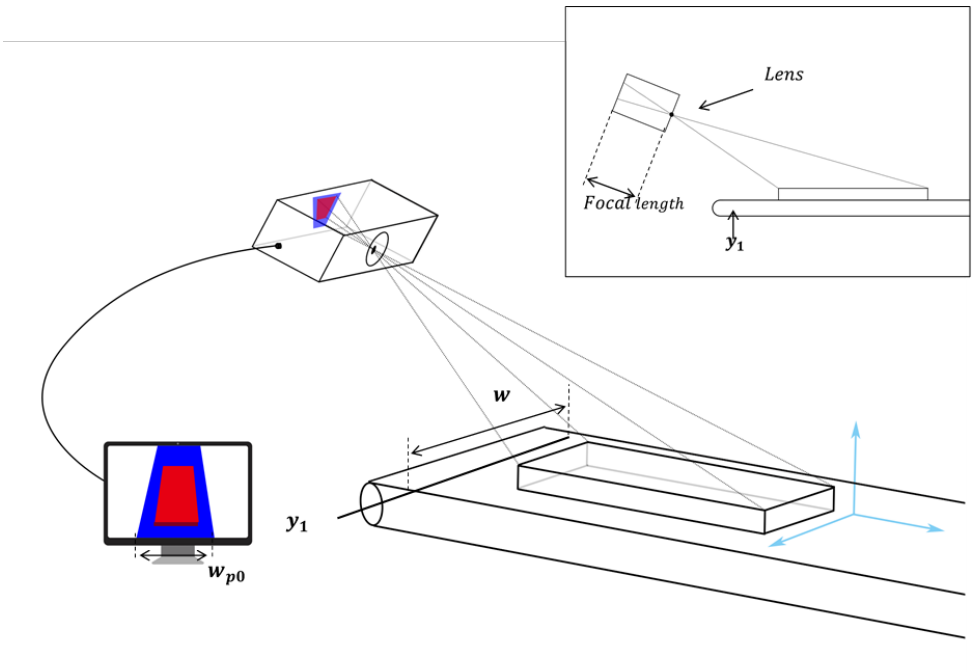


Figure 3.18: Schematic diagram of the visual sizing

(a)

(b)

Figure 3.19: Section recognition and edge extraction: (a) The original image; (b) Section extracted with edges

Algorithm 7: Boundaries Recognition and Sizing

Input: I_{eb}

Output: Number of pixels between two boundaries w_{pi}

1: Moore-Neighbor Tracing Algorithm

Boundaries extraction $\rightarrow B_1, B_2, \cdots, B_n$

Boundaries selection $\rightarrow B_{max}$

Calculate the number w_{pi} of pixels between two boundaries

2: Boundaries Extraction with Line Fitting

Initialize local area selector $I_{w\times h}$, step s

Moore-Neighbor tracing algorithm to extract local boundaries $\rightarrow B_l$

Line fitting in $B_l \rightarrow L_1$ and L_2

Calculate the number w_{pi} of pixels between L_1 and L_2

To convert pixel numbers in the image space to the physical dimensions, we need to recognise the section edges and count the pixel numbers between the two edges of interest. In this dissertation, with the purpose of comparison, we use two algorithms as given in algorithm 7 to recognise the edges and then convert them into physical sizes.

Sub-algorithm 1 uses the Moore-neighbor tracing algorithm directly to get the edge information from the binarised results produced by [13]. The edges of the section are recognised and the diameter of the section in the image space is calculated by counting the number of the pixels between the two edges, as shown in figure 3.19. Sub-algorithm 2 introduces a local area selector $I_{w\times h}$ to constrain the boundary extraction area. $I_{w\times h}$ moves in the image matrix I_{eb} resulted from algorithm 5, with a stride of s both vertically and horizontally to get a local area $L_{w\times h}$. The boundaries in $L_{w\times h}$ are then extracted by Moore-neighbor tracing algorithm as B_l and further fit into line segments L_1 and L_2 by the first order polynomial regression. The diameter of the section in the image space is calculated by averaging the number of pixels between the two line segments.

With algorithm 7, we now have the number w_{pi} of pixels in the image space that corresponds to the steel section diameter, which is converted to physical dimensions by equation (3.3).

3.3.3 Homographic Extension

The above two subsections provide the solution to measure the steel sections when there is no camera calibration performed. However, the accuracy of the second method does not reach the expected error tolerance interval. Also, the Root Mean Square Error (RMSE) of both methods are quite significant. Thus, we involve homography in the solution to further improve

Figure 3.20: Points for calculating homography matrix

the accuracy and narrow down the RMSE.

To derive the homography matrix, several sets (each set with four points) of points that locate along the conveyor barriers are selected. Figure 3.20 shows one set of the points A, B, C and D, the coordinates of which are denoted as

$$\begin{bmatrix} x_A & x_B & x_C & x_D \\ y_A & y_B & y_C & y_D \end{bmatrix}^T \tag{3.4}$$

The corresponding coordinates after applying homographic transformation are denoted as

$$\begin{bmatrix} \tilde{x}_A & \tilde{x}_B & \tilde{x}_C & \tilde{x}_D \\ \tilde{y}_A & \tilde{y}_B & \tilde{y}_C & \tilde{y}_D \end{bmatrix}^T \tag{3.5}$$

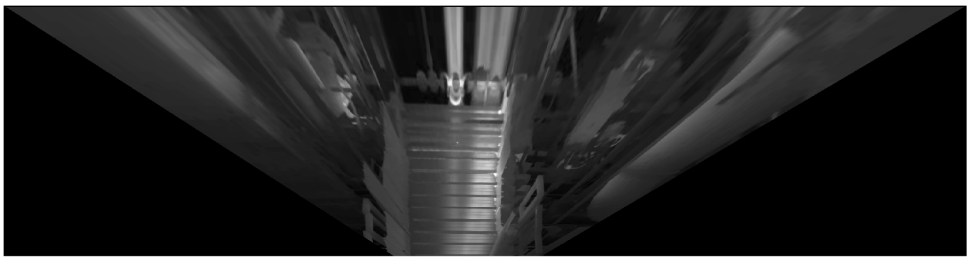
where x_A, x_B, x_C, x_D and y_A, y_B, y_C, y_D are the x and y coordinates of points A, B, C and D, respectively. And those with tildes in equation (3.5) are the corresponding coordinates after homographic transformation. These coordinates satisfy $\tilde{x}_A = x_A$, $\tilde{x}_B = x_B$, $\tilde{x}_C = x_A$, $\tilde{x}_D = x_B$, $\tilde{y}_A = y_A$, $\tilde{y}_B = y_B$, $\tilde{y}_C = y_C$, and $\tilde{y}_D = y_C$.

To eliminate the errors caused by the points selection procedure, five sets of points are chosen to calculate the homography matrix and a final one given in equation (3.6) is obtained

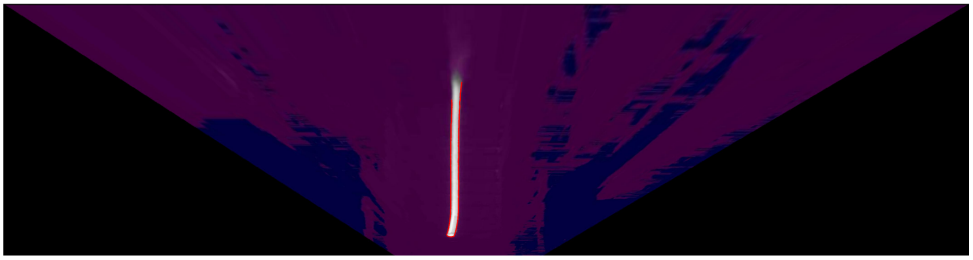
by averaging the five matrices resulted from each set of points.

$$H = \begin{bmatrix} -0.0025277 & 0 & 0 \\ -0.0021371 & -0.0032857 & -0.0000043 \\ 0.9424616 & 0.3342812 & -0.0006065 \end{bmatrix} \tag{3.6}$$

By applying the homographic tranformation to the original images, we can get the top-view images as in figure 3.21(a). Then, the top-view images are further processed by algorithm 5 and 7 to get the physical dimensions of the steel sections.



(a)



(b)

Figure 3.21: Homography transformation: (a) The image of conveyor after homography transformation; (b) Section recognition on transformed image

3.4 Experiments ans Analyses

Four sets of experiments are conducted to demonstrate the effectiveness of the methods proposed. The configuration of each set are as follows.

- **Experiment 1:**
  - **Data:** Original images;
  - **Pixel Counting:** Algorithm 7, sub-algorithm 1.
- **Experiment 2:**
  - **Data:** Homographic images;
  - **Pixel Counting:** Algorithm 7, sub-algorithm 1.
- **Experiment 3:**

- **Data:** Original images;
- **Pixel Counting:** Algorithm 7, sub-algorithm 2.

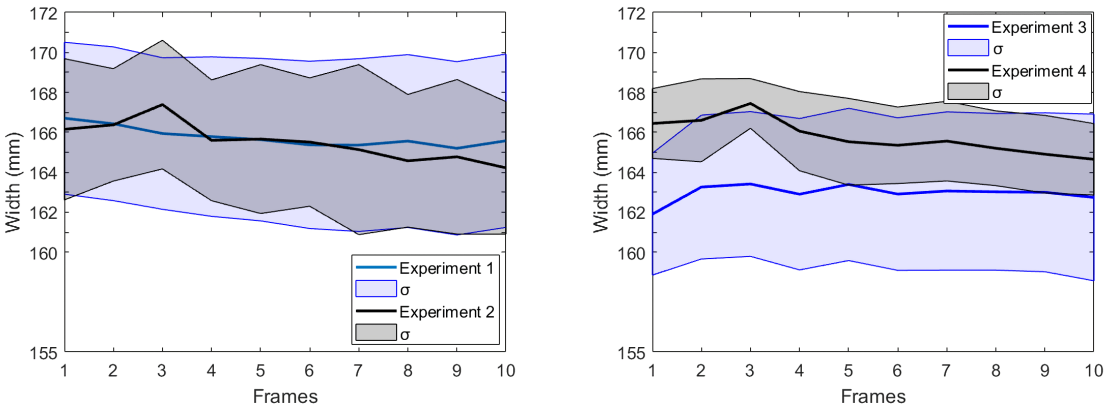
- **Experiment 4:**
  - **Data:** Homographic images;
  - **Pixel Counting:** Algorithm 7, sub-algorithm 2.

In each set of the experiments, 10 frames of a video filmed by a statically-mounted, uncalibrated thermal camera are processed. The steel section to be measured is a cylindrical one with ground truth diameter $165mm$. The diameter and the corresponding RMSE are calculated respectively as follows

$$l = \sum_{i=1}^{M} \bar{l}_i / M, \quad \text{with} \quad \bar{l}_i = \sum_{j=1}^{M_{ij}} l_{ij} / M_{ij} \tag{3.7}$$

$$RMSE = \sqrt{\sum_{i=1}^{M} \left(\bar{l}_i - l\right)^2 / M} \tag{3.8}$$

where $i = 1, \cdots M$ is the index of the frames, l_{ij} with $j = 1, \cdots, M_{ij}$ indicates the section diameter corresponds to the j-th y coordinate, $\bar{l}_i$ is the averaged physical diameter from frame i, and l is the mean from the M frames. As we processed 10 frames, so M is set to 10 in the dissertation.



(a) (b)

Figure 3.22: Experiment results: (a) Experiment 1-2; (b) Experiment 3-4

Figure 3.22(a) shows the measured results and standard deviations σ of **Experiment 1** and **Experiment 2**. We can see that by processing the original images directly, we can get fairly accurate results. However, after homographic transformation, both the diameter estimation and the RMSE are improved. The reason lies in that by the the homographic transformation, the distortion caused by pixels further away from the camera are corrected to some extent. Figure

Table 3.1: RMSE and $\bar{\sigma}$

	Experiment 1	Experiment 2	Experiment 3	Experiment 4
RMSE	1.1147	1.0312	2.0866	0.8738
$\bar{\sigma}$	4.1175	3.4562	3.7745	1.8771

3.22(b) shows the results of **Experiment 3** and **Experiment 4**. We can see that both the diameter estimation and the RMSE are large while processing the original images. However, after the homographic transformation, the diameter estimation accuracy is significantly improved with quite small RMSE. The reason why sub-algorithm 2 given in algorithm 7 reports poor results while processing the original images is the accuracy of the slope and intersection of the extracted lines in the selected area could be affected by the distortion of the steel sections easily. After the homographic transformation, the two edges are almost parallel, which makes the extracted line parameters more stable. Table **??** shows the RMSE and the average standard deviation $\bar{\sigma}$ of the 4 experiments.

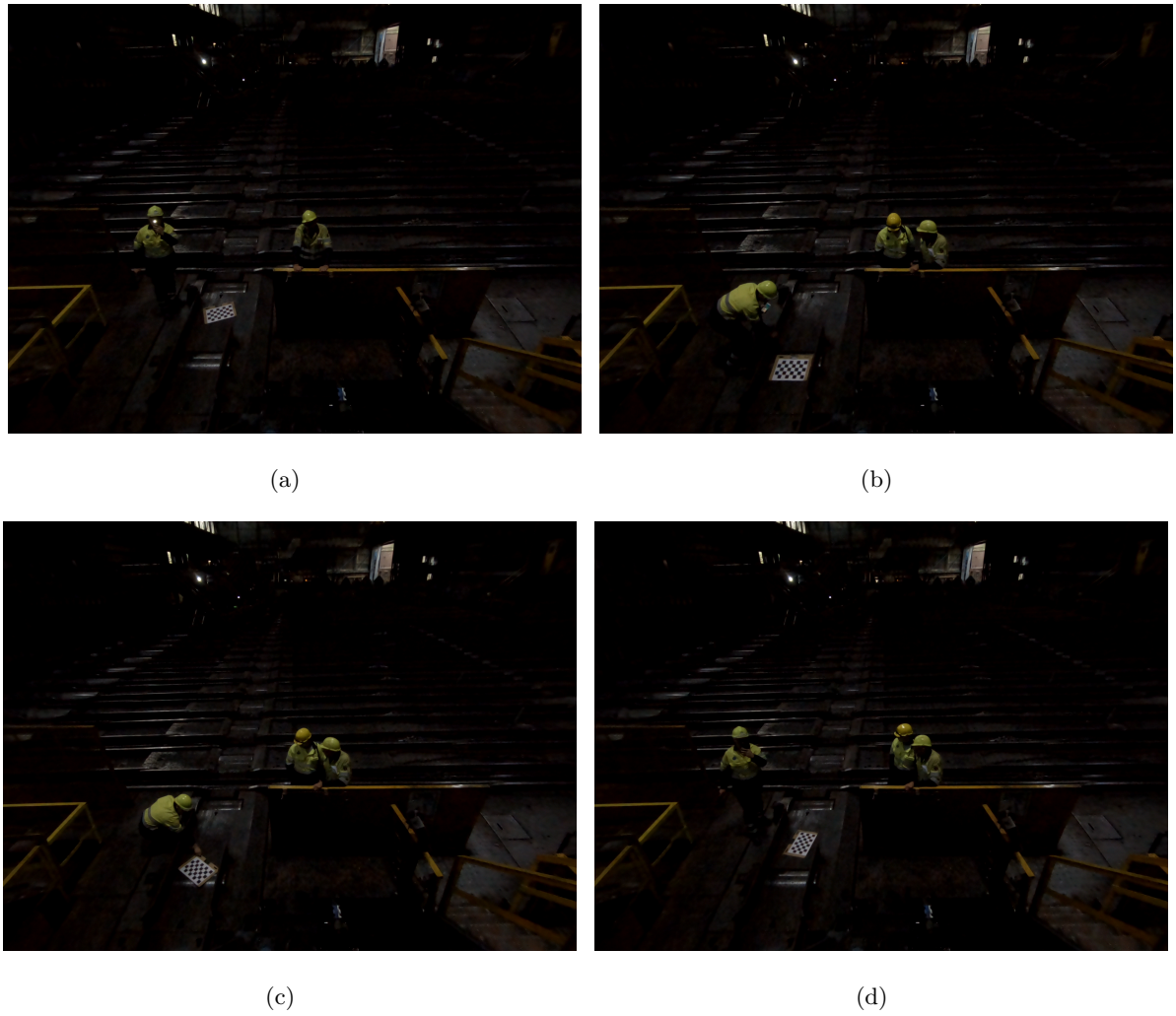
Upon examining all four results, it is evident that **Experiment 4** yields the most favorable outcomes, as indicated by its lowest Mean and RMSE values. This can be attributed to two main factors. Firstly, the utilization of homographic transformation aids in rectifying distortions, thus enhancing the robustness of sub-algorithm 2. Secondly, the line fitting method implicitly filters out system noises inherent within the selected area ($w \times h = 20 \times 85$ with stride $s = 5$ manually in experiments). This additional filtering step contributes to further improvements in the measurement accuracy of the steel section.

From the results above, it is possible to measure the dimension of high-temperature steel using an uncalibrated thermal imaging camera. The distribution of measurement results is relatively average and is of a reference value. However, during the experiment, it is found that there is a fixed error of 5-10mm between the measurement result and the actual size, and the error is not due to the device. Therefore, an adjustable fixed compensation is directly added to the final measurement result. This fixed error is then found and resolved in the subsequent measurement using a dual camera.

3.5 Edge Detection and Random Regression

Thermal imaging cameras have lower image resolution than optical cameras and are somewhat inadequate for accuracy when measuring steel at a medium distance. Moreover, because the calibration using a thermal imaging camera requires a special calibration plate, the calibration process becomes more cumbersome.

At the same time, not every production line is straight and has two side guardrails parallel to it. It is also possible that the spacing of the guardrails will skew over time. So the new measurement algorithm uses images taken by an optical camera and a print calibration board to calibrate the camera. Optical cameras have higher resolution and can effectively improve measurement accuracy. In addition, the calibration process makes the measurement independent of the environmental reference.



(a) (b)

(c) (d)

Figure 3.23: Checkerboards for Calibration

Figure 3.23 shows that the calibration plate is placed on the side of the production line to calibrate the production line plane and the camera. With the camera position unchanged, the calibration process is completed by changing the position of the calibration plate on the side of the production line.

Figure 3.24 shows the image of the steel section on the producing line taken by the camera. It can be seen that the steel is bright orange as the result of photographing high-temperature steel with an optical camera. To some extent, the light from the steel also illuminates part of the production line.

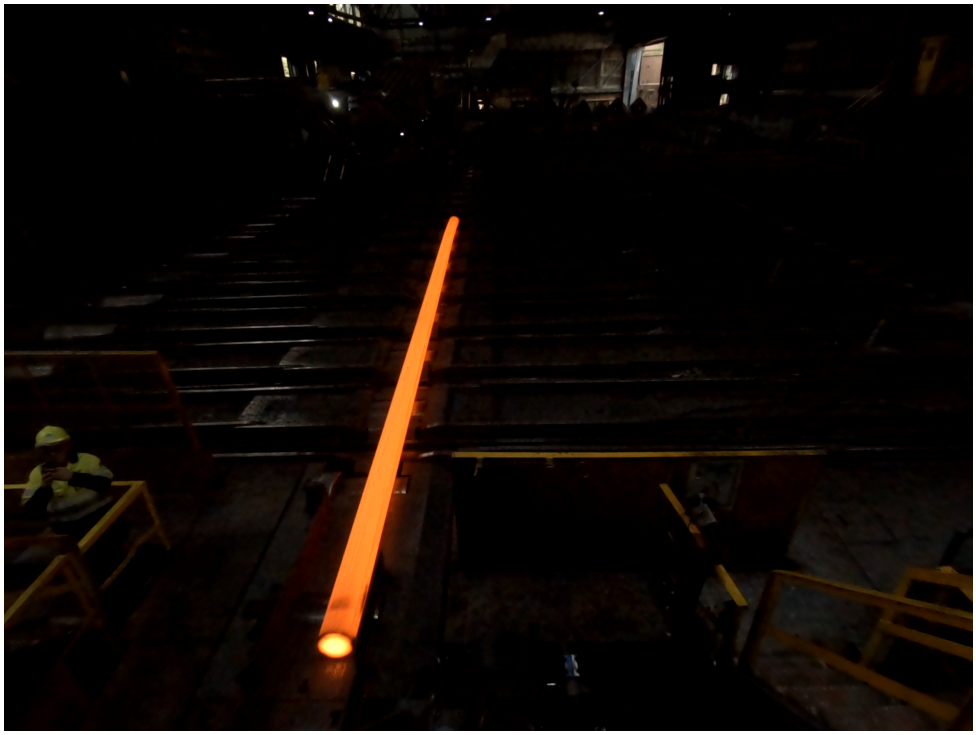


Figure 3.24: Steel Section Filmed by Optical Camera

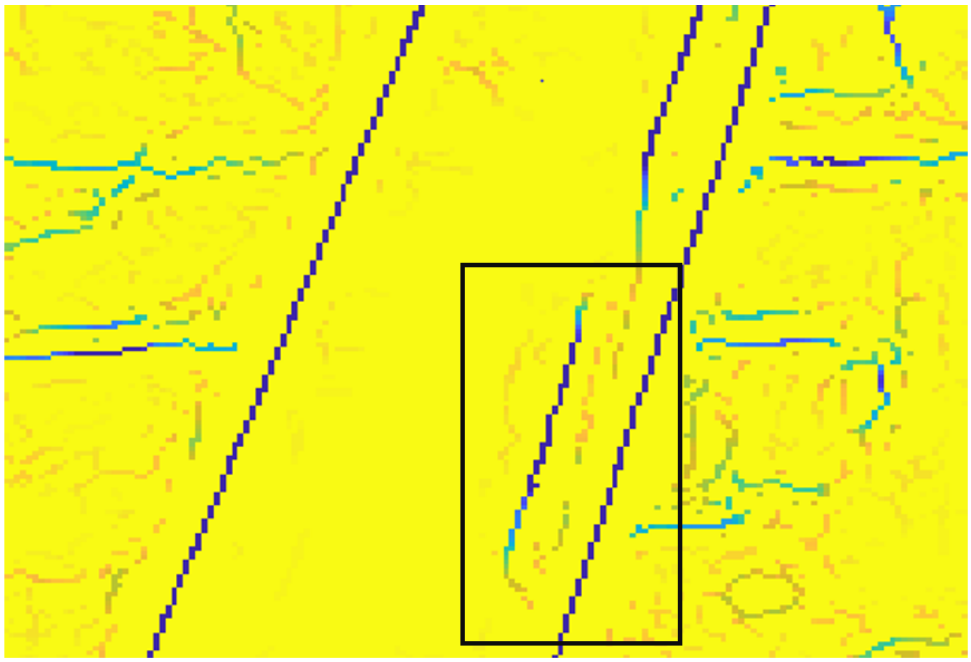


Figure 3.25: Edges detected by the structural random forests. The black rectangle shows that more than one edges are detected, where only one is expected. Prominent edges are marked in dark blue, while other weak edges are marked in light colours.

3.5.1 Edge Detection

In spite of the existence of both traditional edge detection methods [45] and the current deep learning based state-of-the-art edge detection methods [48], we adopt the structural random forests algorithm [13] to extract edges in the video frames. Compared with traditional edge detection methods whose performance relies on setting up good thresholds, the structural random forests algorithm can provide relatively stable and adaptive results without setting up parameters. In addition, the structural forests can deal with Red, Green, Blue (RGB) images directly, while traditional edge detection methods such as Canny require to convert RGB images to gray scale images, which would limit the efficiency. Compared with deep learning based edge detection methods, the structural random forests algorithm is easier to train, lighter to be deployed, and less dependent on expensive hardware like GPUs.

Edges detected by the structural random forests provide inputs to our proposed measuring framework. Both HRB edges and environment edges are detected at this stage. Preliminary results with the structural random forests algorithm [13] and with thermal images are reported in our preceding paper [60]. The main idea is to detect edges in images via constructing a structured forest. One of the disadvantages of the structured forest algorithm, as discussed in [13], is the occurrence of diffused edges, which causes accuracy degradation. We implement the algorithm by enabling the Non-Maximum Suppression (NMS) [13] to sharpen the extracted edges. Figure 3.25 shows sharpened edges from the structural random forests algorithm. Three types of edges are detected: *1*) Edges of the HRB, which are the two longest edges marked in dark blue. *2*) Strong non-HRB edges, e.g., the dark blue edge apart from the HRB edge within the rectangle. *3*) Weak non-HRB edges, e.g., all edges apart from the two strong ones.

The hot steel bars, however, have only two prominent edges and we need well pronounced edges in order to calculate the diameter of a cylindrical bar. As it can be seen within the rectangle shown in Figure 3.25, there are more than one detected edges, although we are expecting just one. The reasons for this ambiguity are twofold. From the HRB aspect, while rolling along the conveyor, the HRB is cooling down unevenly. This results in intensity changes in the images and hence leads to extra 'edge' detection by the algorithm. From the structural random forests algorithm aspect, Dollar et al. [13] explain that the edges can be diffused due to the fact that edge estimation can shift a few pixels from the true location. The underlying cause is that the voting mechanism used cannot ensure the noisy edge predictions to be well aligned. This also causes weak edges (marked in light colours) as shown in Figure 3.25.

To mitigate the effects of those extra edges and to achieve high accuracy in the measurements, a sliding window random regression algorithm is further applied to process the edges detected by the structural random forests.

3.5.2 Sliding Window Random Regression

To find the edges of interest, which are edges of a HRB, in a given frame I_{RGB}, we propose Algorithm 8 to subtract the background and get boundaries of interest of the HRB edges. In order to do that, I_{RGB} is binarised according to [46] based on histogram information, resulting in I_{BW}. It is then processed with opening morphological methods followed by a dilation [55] to remove imperfections, caused by temperature diffusion and background noises. The resulted I_{MOR} is applied to mask the edges I_{EDGE} detected by the structural random forests, resulting in I_{GEDGE} with mainly edges of the HRB.

To remove weak edges, pixels in I_{GEDGE} with intensities less than a threshold β are suppressed. Edges in I_{GEDGE} are further binarised and denoised with morphological methods resulting in I_{MEDGE}. Figures 3.26 (a) and (b) show examples of edges detected by Algorithm 8. We can see that edges of HRBs become prominent and most weak edges are removed. However, there are still environmental edges that mix up with the HRB edges. They are caused by the cooling process. Their intensities are usually weaker than those of the HRB edges, while still stronger than those weak edges. If we set up the parameters of Algorithm 8 to be with high values, there is a risk that the HRB edges are removed as well. We, therefore, set up moderate values of the parameters in Algorithm 8 to remove weak edges.

Next, we use the Moore-Neighbor tracing algorithm [41] to extract the boundaries of interest I_{BOI} of I_{MEDGE} and to make sure that HRB edges are enclosed within the boundaries of interest. Figures 3.26 (c) and (d) show in green the extracted boundaries of interest. Obviously, the presence of environmental 'edges' is inevitable. This, however, makes the remote sizing even more challenging and increases the necessity of uncertainty quantification of the measurements.

Algorithm 8: Boundaries Of Interest Extraction

Input: I_{RGB}

Output: Detected HRB edge boundaries I_{BOI}

1: Binarise I_{RGB} according to Otsu's method $\rightarrow I_{BW}$ [46]

2: Morphological denoising $I_{BW} \rightarrow I_{MOR}$

3: Structural Random Forests with NMS enabled $\rightarrow I_{EDGE}$

4: Mask I_{EDGE} with I_{MOR} and convert the result to grey-scale $I_{GEDGE} = I_{MOR} \odot I_{EDGE}$

5: Strength edges in I_{GEDGE}

6: Morphological denoising $I_{CEDGE} \rightarrow I_{MEDGE}$

7: Moore-Neighbor Tracing Algorithm to extract boundaries of $I_{MEDGE} \rightarrow I_{BOI}$.

With boundaries of interest obtained from Algorithm 8, we now give details of the sliding window random regression algorithm in Algorithm 9. We define a binary sliding window $I_{H \times W}$ (shown in white in Figures 3.26 (c) and (d)) with height H and width W to select a region

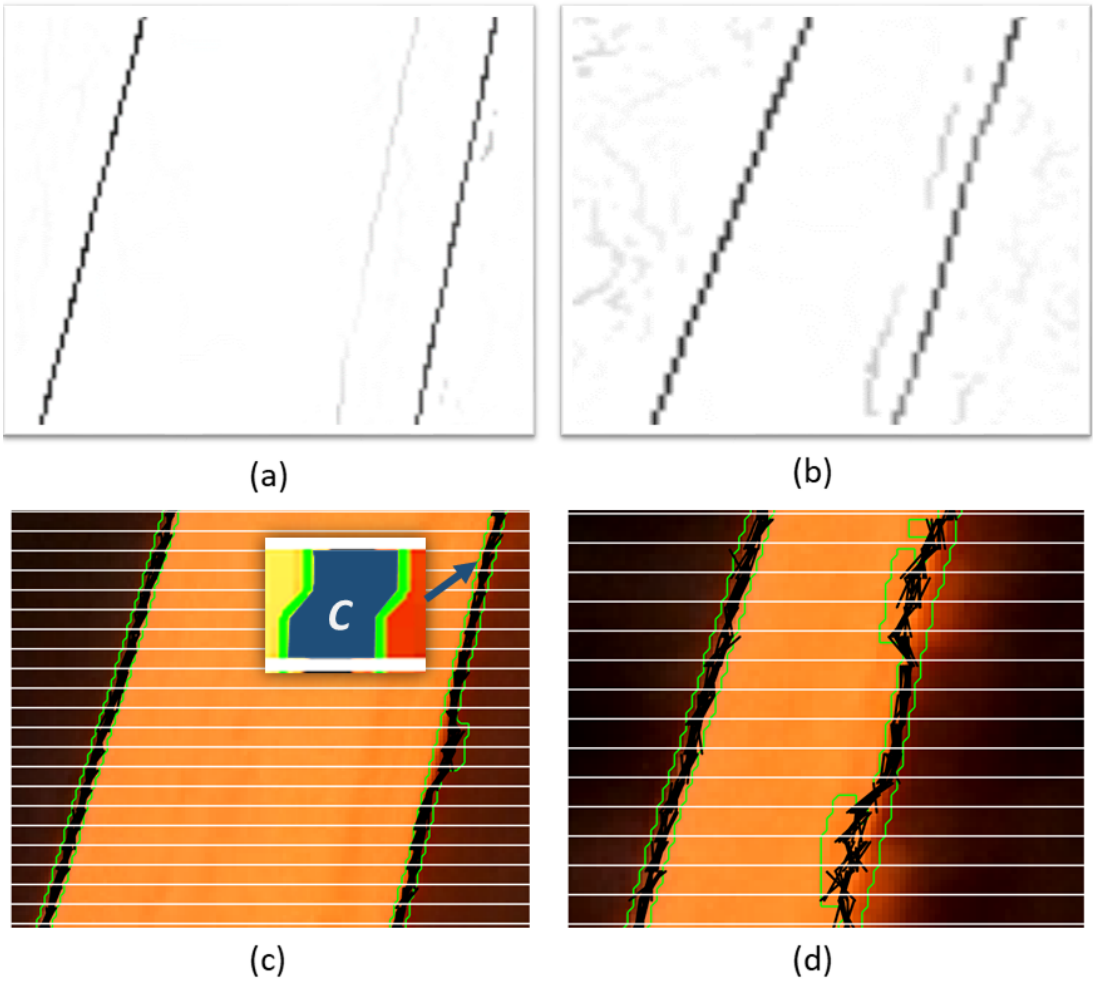


Figure 3.26: Results of the sliding window random regression algorithm. (a) and (b) are the edges detected by Algorithm 8. (c) and (d) show the sliding window random regression results, where the green curves are boundaries of interest, white lines indicate the sliding windows, and black line segments are from sliding window random regression algorithm. (c) also shows a Current Boundary Of Interest (CBOI) with its area marked in dark blue, and the area is denoted by C.

of interest for the sliding window random regression algorithm. Considering the background is subtracted by Algorithm 8 and the HRB moves vertically, W can be set to the width of the image. In this dissertation, we set up the sliding stride to be equal to H without loss of generality, and assume that the sliding window $I_{H\times W}$ can move m steps in total.

For the i-th sliding window $I_{H\times W}$, with $i \in \{1, \cdots, m\}$, we suppose that there are n_i HRB edges within it. For the j-th HRB edge, with $j \in \{1, \cdots, n_i\}$, there is a corresponding CBOI that encloses the HRB edge. Figure 3.26 (c) shows an example of CBOI in the center of the figure. It is enclosed by the current sliding window edges (on the top and bottom of the dark blue area) and boundaries of interest (on the left and right of the dark blue area). We denote the left and right edges of the CBOI as B_j. The coordinates belong to B_j are then concatenated as $\hat{\mathbf{x}}_{ij}$ and $\hat{\mathbf{y}}_{ij}$, which are further classified by the K-means algorithm [30] into several clusters, depending on the number of HRB edges.

As it can be seen from Figures 3.26 (c) and (d), the HRBs are not segmented well due to the unevenly cooling process and other factors. This directly leads to corrupted CBOIs, or expansion of CBOIs. It would ultimately degrade the accuracy of the calculated diameters of the steel sections. Thus, we randomly sample S point pairs from $\hat{\mathbf{x}}_{ij}$ and $\hat{\mathbf{y}}_{ij}$ and next apply a polynomial model to fit the HRB edges from the samples.

Without any loss of generality, the following polynomial regression model

$$f(x) = c_0 + c_1 x^1 + c_2 x^2 + \cdots + c_k x^k \tag{3.9}$$

is applied. In our case, we aim to measure the diameter of a cylindrical HRB. Therefore, a first order polynomial regression model is sufficient. By assigning $c_1 \triangleq a_{ij}$ and $c_0 \triangleq b_{ij}$, we have a linear model

$$f_{ij}(x) = a_{ij}x + b_{ij}. \tag{3.10}$$

After the HRB edges are fitted, we use (3.10) to generate a new set of points to represent the corresponding HRB edges. In our case, we set

$$\mathbf{x}_{ij} = [(i-1) * H + 1, \cdots, i * H], \tag{3.11}$$

and $\mathbf{y}_{ij}$ is then produced by

$$\mathbf{y}_{ij} = a_{ij}\mathbf{x}_{ij} + b_{ij}. \tag{3.12}$$

The black line segments in Figures 3.26 (c) and (d) show the results of Algorithm 9. The results shown in these figures are obtained after $T = 10$ times sampling from each CBOI. We can see that, if the CBOIs are not severely corrupted, the line segments from the sliding window random regression algorithm define well the HRB edges. However, when CBOIs are corrupted, the line segments from the sliding window random regression algorithm no longer represent the

Algorithm 9: Sliding window Random Regression of Hot Rolled Bar Edges

Input: I_{BOI}, a binary sliding window $I_{H\times W}$

Output: $\mathbf{x}_{ij}$, $\mathbf{y}_{ij}$, $i = 1, \cdots, m$, and $j = 1, \cdots, n_i$.

1: **for** $i = 1, \cdots, m$ **do**

2: Determine the HRB edges number n_i within $I_{H\times W}$

3: **for** $j = 1, \cdots, n_i$ **do**

4: Concatenate the coordinates of B_j as $\hat{\mathbf{x}}_{ij}$ and $\hat{\mathbf{y}}_{ij}$.

5: Using k-means to classify $\hat{\mathbf{x}}_{ij}$ and $\hat{\mathbf{y}}_{ij}$ into two clusters, each corresponds to a HRB edge.

6: Linear regression

$$f_{ij}(x) = a_{ij}x + b_{ij}$$

7: Re-calculate y coordinates from row $(i-1) * H + 1$ to row $i * H$ within the sliding window

$$\mathbf{x}_{ij} = [(i-1) * H + 1, \cdots, i * H]$$

$$\mathbf{y}_{ij} = a_{ij}\mathbf{x}_{ij} + b_{ij}$$

8: **end for**

9: **end for**

HRB edges accurately. Here, we first provide the transformation from the image plane to the physical plane. The treatment of the measurement uncertainty will be given in the next section.

In our case, two HRB edges are expected within the sliding window $I_{H\times W}$. Therefore, we set up all n_i to be equal to $n = 2$. The $\mathbf{x}_{ij}$ and $\mathbf{y}_{ij}$ coordinates from the image plane are converted to coordinates in the physical plane through the transformation

$$\begin{bmatrix} x_{ij}^w \\ y_{ij}^w \\ z_{ij}^w \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \mathbf{K} \begin{bmatrix} x_{hj}^I \\ y_{hj}^I \\ 1 \end{bmatrix}, \tag{3.13}$$

where $\mathbf{R}$ and $\mathbf{T}$ are respectively the rotation and translation matrices, and $\mathbf{K}$ is the intrinsic matrix of the camera parameters. These matrices are obtained via the calibration process. The coordinates $x_{ij}^I \in \mathbf{x}_{ij}$ and $y_{ij}^I \in \mathbf{y}_{ij}$ are from the image plane and $[x_{ij}^w, y_{ij}^w, z_{ij}^w, 1]^T$ is the vector of corresponding coordinates in the physical plane. Given the vectors $I_{i1} = [x_{i1}^I, y_{i1}^I]^T$ and $I_{i2} = [x_{i2}^I, y_{i2}^I]^T$ on two HRB edges with $x_{i1}^I = x_{i2}^I$, the diameter l of the HRB is then calculated through

$$l = \|P_1 - P_2\|_2, \tag{3.14}$$

with $P_1 = [x_{i1}^w, y_{i1}^w]^T$ and $P_2 = [x_{i2}^w, y_{i2}^w]^T$, which are physical plane correspondences to I_{i1} and

I_{i2}. Here $\|.\|_2$ denotes the Euclidean norm.

3.6 Weighted Variance for Uncertainty Quantification

Similarly to the measurement accuracy, the level of trust in the measurements is also essential to this task as it supports downstream decision-making. In this dissertation, we quantify the measurement uncertainties on the final results with the weighted variance as part of the Algorithm 10. In our case, the trust level reflects how severely the measurements are affected by corrupted CBOIs along with other noises. Figures 3.27(a) and (b) show well determined boundaries of interest, while Figures 3.27(c) and (d) show boundaries of interest with corrupted CBOIs. When the measurements are from corrupted CBOIs, the variance value will go high.

Given the i-th sliding window, there are n_i CBOIs in it, and each corresponds to a B_j with $j \in \{1, \cdots, n_i\}$. For each B_j, we sample T times, each time with S point pairs from $\hat{\mathbf{x}}_{ij}$ and $\hat{\mathbf{y}}_{ij}$. From each point pair, we use (3.13) and (3.14) to calculate S diameters, which are further averaged to find the representative diameter value l_{it}, with $t = 1, \cdots, T$. Then the measurement and variance in the current sliding window are calculated respectively from

$$l_i = \sum_{t=1}^{T} l_{it}/T, \ \sigma_i = \sqrt{\sum_{i=1}^{T} (l_{it} - l_i)^2/T}. \tag{3.15}$$

With the sampling strategy, impacts of the corrupted CBOIs on the measurements are mitigated, especially when the area of a CBOI is small. However, when the area of a corrupted CBOI is big, it becomes difficult to use the fitted line segments to represent the HRB edges. The reason lies in that compared with samples constrained in a smaller CBOI, samples from a bigger CBOI are more dispersed. Hence, we consider this CBOI area for quantifying the measurement uncertainty.

In this dissertation, we have taken the CBOI areas into consideration by using CBOI areas as weights, and each σ_i from (3.15) is then adjusted by the weight. To achieve that, for the i-th sliding window, we first calculate the area of each CBOI within it and get the respective areas $A_{i1}, \cdots, A_{in_i}$. A similarity ratio $\mathcal{R}_i$ is next calculated via

$$\mathcal{R}_i = \Big(\sum_{j=1}^{n_i} A_{ij}/n_i\Big)/C, \tag{3.16}$$

where C is the average area of uncorrupted CBOIs. We use it to normalise the similarity ratio. Figure 3.26 (c) shows an uncorrupted CBOI and C is the area of the region marked in dark blue. The values of the similarity ratio $\mathcal{R}_i$ could fall into the following three cases: *1)* $\mathcal{R}_i < 1$, *2)* $\mathcal{R}_i = 1$, and *3)* $\mathcal{R}_i > 1$. We would prefer the first two cases because it means $A_{i1}, \cdots, A_{in_i}$ are on average smaller or equal to C. This indicates that samples from the corresponding CBOIs are constrained in small areas, and line segments fitted from these samples are more likely to align

Algorithm 10: Uncertainty Quantification

Input: $\hat{\mathbf{x}}_{ij}$ and $\hat{\mathbf{y}}_{ij}$ after clustering, a binary sliding window $I_{H\times W}$, and sampling times T.

Output: diameter l_i and weighted variance Σ_i

1: **for** $i = 1, \cdots, m$ **do**

2: CBOI areas in current window, $A_{i1}, \cdots, A_{in_i}$, with average $A_i = \sum_{j=1}^{n_i} A_{ij}/n_i$, and similarity ratio $\mathcal{R}_i = A_i/C$.

3: Calculate weight $\mathcal{L}_i$ from (3.17).

4: Weighted variance calculation from (3.18).

5: **end for**

with the HRB edges. On the contrary, when the third case happens, we would know that the samples are from CBOIs with areas greater than C, which will increase the chance that fitted line segments from these samples are unaligned with HRB edges. To convey the information, we, therefore, define the variance weight as

$$\mathcal{L}_i = \mathcal{R}_i^2 * \exp(\mathcal{R}_i - 1),\ i = 1, \cdots, m. \tag{3.17}$$

We use $\mathcal{R}_i^2$ to indicate that the weight is proportional to the CBOI area. It is obvious that $\mathcal{R}_i$ values greater than 1 would lead to greater $\mathcal{L}_i$, when compared with $\mathcal{R}_i$ values equal to or are smaller than 1.

We now give the trust level in the measurements as

$$\Sigma_i = \mathcal{L}_i * \sigma_i,\ i = 1, \cdots, m. \tag{3.18}$$

Note that a big value of Σ_i means a reduced trust in the measurement, compared with a measurement with a small Σ_i value. The bigger the Σ_i value is, the more scattered the measurement is, in a wide area.

3.7 Performance Validation and Evaluation

To demonstrate the effectiveness of the framework, we processed video frames captured by a GoPro® Hero 7 Black camera with Matlab 2018a programs. The PC configuration includes an Intel(R) Core(TM) i7-7800X CPU and 16.0GB RAM. The 2.7K camera mode is used and the shutter speed is set up to 1/480 s, to restrain the distortion and overexposure. The camera is calibrated on the scene with a checkerboard of 8×5 squares of size 50×50 mm. Although the whole HRB is visible in the images, we only focus on the region where the checkerboard was placed to exclude errors caused by calibration parameters. The ground-truth diameter of the HRB is 265.5 mm. The parameters α and β are set to 0.95 and 240, respectively. Here the height H is equal to 5 pixels and the width W of the image is equal to 2704 pixels. Table I

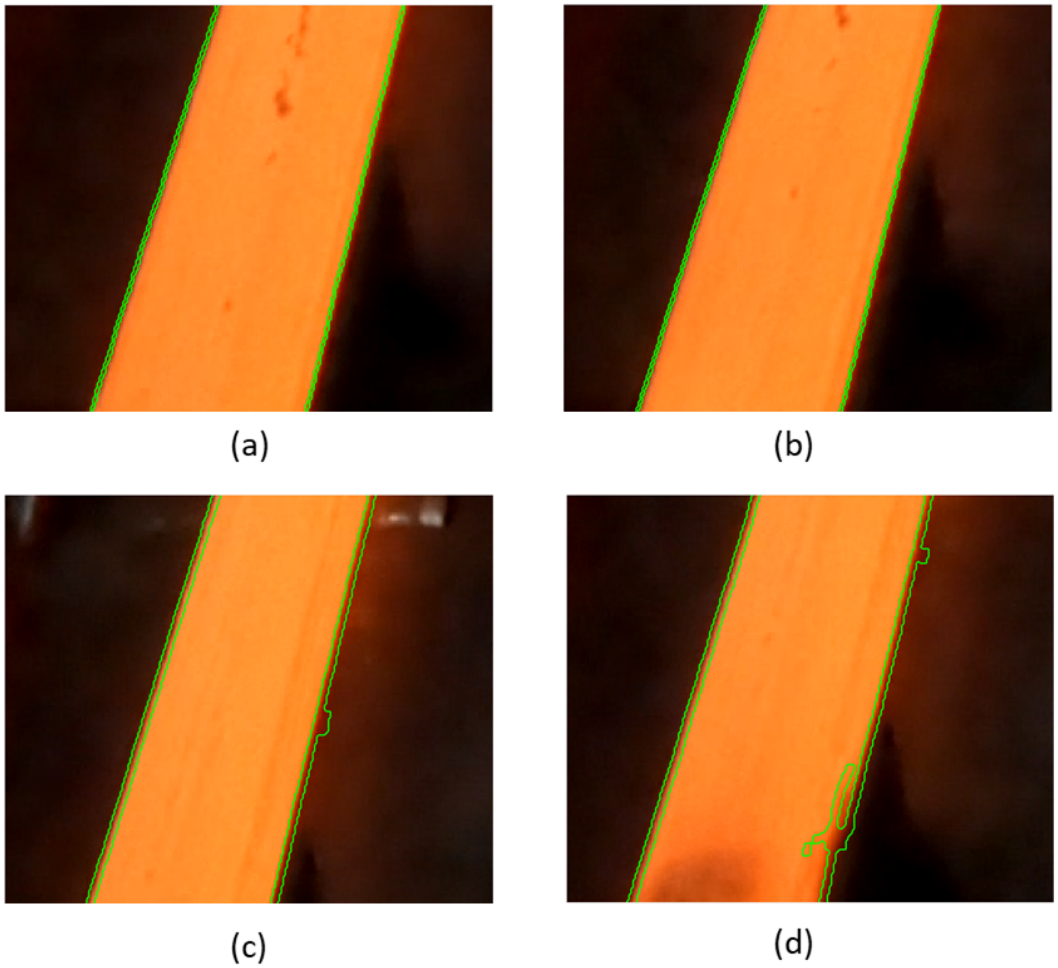


Figure 3.27: Boundaries of interest from four randomly selected images

Table 3.2: Efficiency evaluation of the algorithms

	Algorithm 8	Algorithm 9	Algorithm 10
Time(ms)	407	38	52

shows the average computational costs of each algorithm. The computational time is suitable for real time applications since the HRBs move slowly (even stop for sawing the ends) on the conveyor.

Figure 3.28 shows results from the proposed framework. For a compact representation of the results, we have aggregated the HRB diameter ground-truth (265.5 mm) and the tolerance zone ($[265.5-3.0\ mm, 265.5+3.0\ mm]$), the measurements from our framework, and the weighted variance for uncertainty quantification into a single figure. In each sub-figure, the x-axis indicates the number of sliding windows. The left y-axis shows the ground-truth, measurements, and the tolerance zone. The right y-axis shows only the one Σ interval, to show the trust level in the diameter measurement. Compared with the narrower Σ intervals, a wider one Σ interval (or even a peak interval) means a reduced level of trust in the measurement, as it means to us that the measurement scatters in a big area.

The measurement is not necessarily enclosed by the one Σ interval due to different scales of the two y-axes. However, the trends of the one Σ interval and the corresponding measurement should be kept consistent. Especially, the one Σ interval should convey the information when the measurement goes over the tolerance zone.

For instance, we can see in Figures 3.28(a) and 3.28(b) that, the measurements fluctuate around the ground-truth and the one Σ intervals change in accordance. There are no dramatic changes of the measurements or the one Σ intervals observed in these two figures. All the measurements are bounded by the tolerance zone. This indicates that these measurements are trust-able. In fact, Figures 3.28(a) and 3.28(b) show typical results from images that are similar to Figures 3.27(a) and 3.27(b), where the CBOIs are not corrupted.

In comparison, when CBOIs are corrupted as shown in Figures 3.27(c) and 3.27(d), we can observe drastic changes of the measurements and the one Σ intervals. Typical results are shown in Figures 3.28(c) and 3.28(d). There are one prominent measurement peak in Figure 3.28(c) and two in Figure 3.28(d). We can see measurements corresponding to the peaks go over the tolerance zone, and thus should not be trusted. With our proposed uncertainty quantification algorithm, this information is encoded in the corresponding one Σ interval. As can be seen from Figures 3.28(c) and 3.28(d), wide one Σ intervals emerge in accordance with the measurement peaks. Therefore, the one Σ interval would provide us with a trust level in the measurements.

With the proposed framework, when the one Σ interval changes drastically, we would

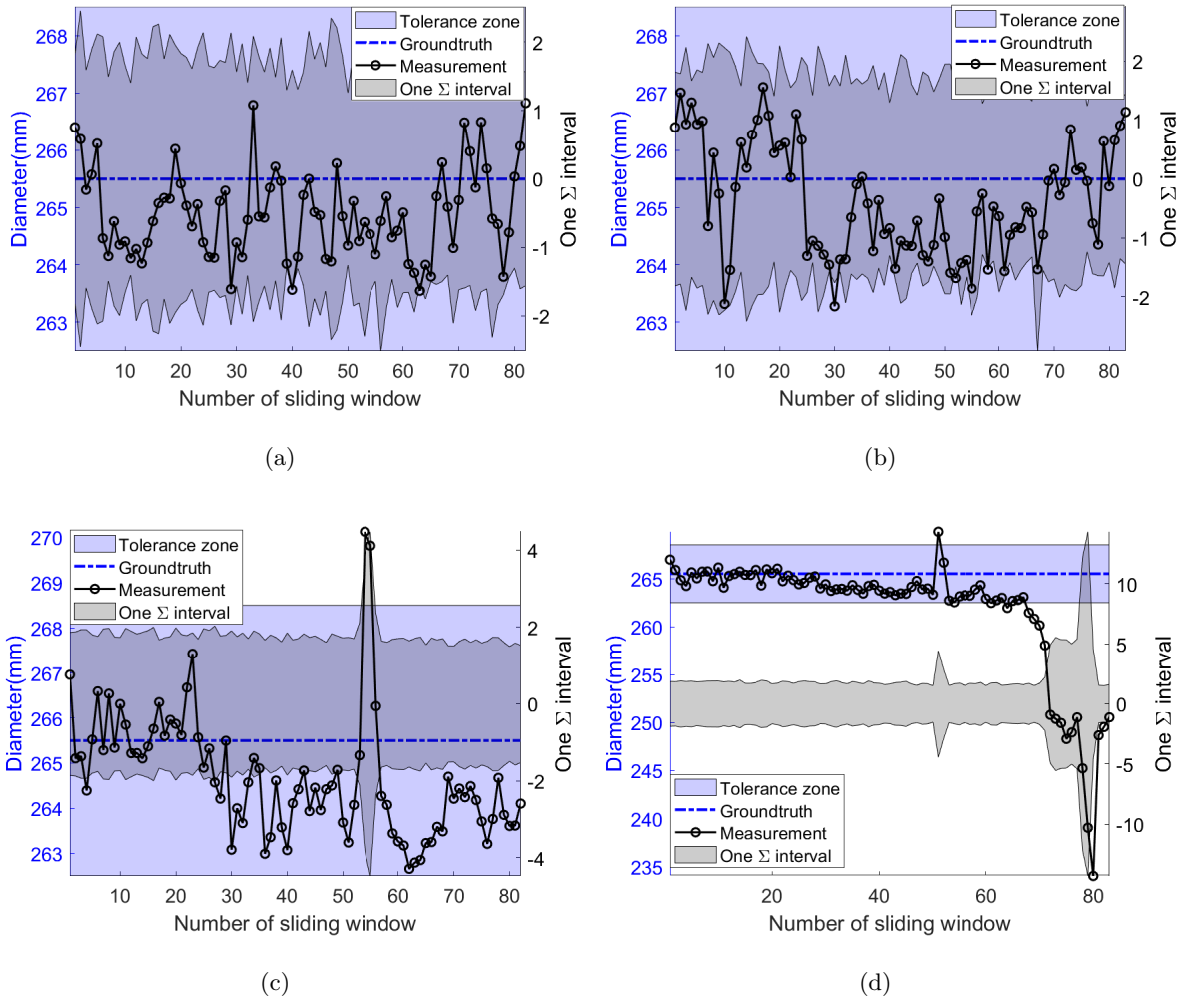


Figure 3.28: Diameter measurements and trust level from four randomly-selected frames: (a) and (b), measurements with high trust level, (c) measurements correspond to one corrupted CBOI, (d) measurements correspond to two corrupted CBOIs.

discard (not trust-able) the corresponding measurements. While we have shown measurements within sliding windows in Figure 3.28, we now give measurements from a whole frame by integrating all the trust-able measurements from the sliding windows. For frame-wise measurements, we use the following relations to calculate the measurement $\bar{l}_i$ and variance σ.

$$\bar{l}_i = \sum_{i=1}^{m_f} l_i / m_f, \ \sigma = \sqrt{\sum_{i=1}^{m_f} \left(l_i - \bar{l}_i\right)^2 / m_f}, \tag{3.19}$$

where m_f is the number of the processed frames.

Table 3.3: RMSE of Fig.3.28

	a	b	c	d
RMSE	0.9684	1.0401	1.5649	7.5128

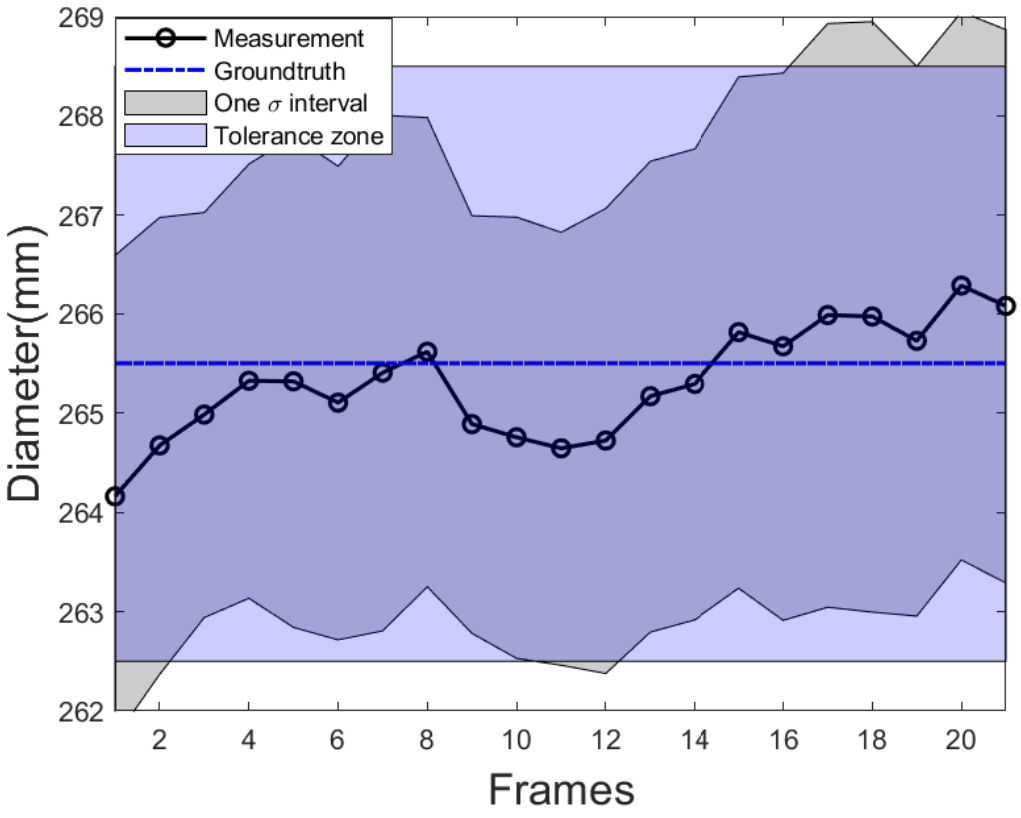


Figure 3.29: Diameter measurements from frames

The frame-wise measurements are shown in Figure 3.29. Note that the x-axis now indicates the frame numbers. Since measurements from corrupted CBOIs are discarded, we now use (3.19) to represent the one σ interval. Be aware that the weighted variance Σ is used to quantify measurement uncertainty, while σ here is just a normal variance describing how spread out the frame-wise measurements are. We can see that the difference between each measurement and ground-truth pair is within $[-2.5\ mm, 2.5\ mm]$ and the RMSE is 0.5798, which is in line with the rolling standards $[-3.0\ mm, 3.0\ mm]$.

3.8 Summary

This chapter provides an overview of various edge recognition and extraction algorithms, ranging from traditional differentiator-based methods to contemporary edge detection techniques. Of particular note is the application of the structural random forest edge detection algorithm, utilized in practical scenarios to discern the edge of steel sections during actual production, thereby facilitating the measurement of these sections. Within this chapter, two distinct algorithms for quantifying steel dimensions are delineated.

The first algorithm employs infrared thermal imaging images as the primary source of visual data, utilizing production line parapets as markers to gauge steel dimensions. It should be

noted that while thermal imaging is employed in this case, optical imaging could be harnessed to achieve comparable outcomes. The algorithm commences with the application of structural random forest edge detection to identify the conveyor belt image within the background, subsequently extracting the edges of the conveyor parapets. A noise reduction algorithm is then applied to filter these extracted edges. The actual physical size represented by each pixel in the image is calculated by correlating the widths of the extracted conveyor parapets with their actual intervals. Concurrently, the image's homography matrix can be deduced from the vertical edges of the conveyor belt, enabling the original image to undergo a homography transformation. This ensures that all pixels within the image denote the same actual size. After establishing the pixel size via the background conveyor, edge detection and recognition algorithms are deployed to identify and extract the steel section within the image. The dimensions of the steel section are subsequently calculated by converting the pixel size into real-world measurements.

This approach employs only a monocular camera and obviates the need for prior hardware calibration. Instead, environmental references serve as calibration indicators. Consequently, the debugging and calibration tasks following the camera installation can be accomplished remotely, eliminating the need for additional field adjustments. As long as the camera's angle of capture encompasses the object being measured, it can be installed without necessitating the inconvenience of regular on-site maintenance and calibration.

The second method employs an optical image, captured by a GoPro camera, as the input. The optical image has a higher pixel count and less input delay than the infrared thermal imaging image. Simultaneously, the cost and maintenance complexity of optical camera equipment is significantly lower than that of thermal imaging cameras. In this method, rather than searching for a reference object in the background for size reference, a calibration plate is utilized post-camera setup to determine the camera's spatial parameters. Calibrating the camera beforehand enables the system to be independent of any specific object in the background and to accurately obtain the reference size, even in complex environments where the conveyor belt is not perpendicular to the camera. Subsequently, the structured random forest edge detection and recognition algorithm is still used to identify the position and edge of the steel section. After pinpointing the edges of the steel section, a sliding window random regression algorithm is employed to filter the edges and estimate the error. Finally, the internal and external parameters of the camera, obtained through calibration, can be directly used to map the size of the steel section in the image plane to the real-world plane.

Both methods utilize a monocular camera as the system input to measure the dimensions of the target steel section in real-time, with a background reference or calibration serving as the dimension reference. However, both methods necessitate a fixed size compensation when measuring steel sections with varying diameters. For instance, when measuring 190mm diameter

steel, the system compensates for the 190mm diameter, thereby ensuring precise measurement accuracy for steel around 190mm in diameter. However, if the production line switches to producing 150mm diameter steel at this juncture, the measurement system would exhibit a substantial fixed error. This measurement error arises because the dimension reference plane does not align with the measurement plane. Whether conveyor belts or calibration plates, they generate a reference plane consistent with the site ground, i.e., the conveyor belts. However, the steel itself has thickness, and the diameter represented by the pixels used for measurement is actually at the height of the steel's radius off the ground, which fluctuates with the steel's size. Since a monocular camera does not provide depth information for the image, there is no way to identify such errors during processing. Consequently, a dual-camera system is employed in subsequent experiments to address this issue.

Chapter 4

Image Registration

4.0.1 Practical Industrial Task

When hot steel sections are moving onto the mills, the sections with different sizes are situated at a different level from the ground. This creates challenges to computer vision systems. The approach proposed here is able to cope with such challenges and adapt to the measurement plane changes. Unfortunately, a measurement system based on monocular data is not feasible due to the lack of depth information. Although monocular camera data can predict the depth of field to a certain extent through the deep learning method, it needs to provide a lot of data from the corresponding environment to train the system, and the accuracy of the final results is not enough for remote sizing tasks [39].

So, the binocular system is chosen since it provides depth information through the disparity generated by the two cameras. After acquiring the images from two cameras, each image pair should be registered first.

Due to the light changing problems in the factory, using the images' intrinsic feature points becomes unreliable. Furthermore, the large baseline of the two cameras leads to a significant difference in the subject's position in the images, which further increases the image registration difficulty. Due to the diversity and various types of registration images, a general method suitable for all registration tasks cannot be designed. Each method should consider the assumed geometric deformation types between images and the radiation deformation and noise damage, the accuracy of registration, and the characteristics of application data. To solve this problem, extrinsic feature points are used by the system. A checkerboard is used in the camera calibration process, and it also provides external features points for image registration.

4.0.2 Image Registration

As shown in Figure 4.0.2, the above Figure 4.1(a) is the calibration plate image taken by the left camera, and the following figure .4.1(b) is the image taken by the right camera at the same time.

(a)

(b)

Figure 4.1: The Checkerboard Captured by Two Cameras: (a) Left Cam; (b) Right Cam.

For image registration, the coordinates of the corners of the calibration plates in the two images are extracted first, and then the transformation matrix is calculated using the coordinates of the corners. The right camera image can be successfully registered with the left camera image by converting the right camera image using the computed transformation matrix. The whole process of the algorithm is shown in Alg.11

Figure 4.0.2 shows how the checkerboard image taken by the right camera matches the image taken by the left camera through the image registration process. Figure 4.2(a) shows the initial image of the checkerboard taken by the right camera. Through the calculated transformation matrix, the initial image is transformed into the image shown in Figure 4.2(b). Finally, Figure 4.2(c) shows the registration results by superimposing the left camera image and the

Algorithm 11: Image Registration

Input: I_L, I_R

Output: The registered image I_{Rr}

1: Extract the corner points C_L, C_R of checkerboards in I_L, I_R

2: Calculate the geometric transformation T_{RL} ,which transform $C_R \rightarrow C_L$

3: Apply T_{RL} to I_R :$T_{RL}(I_R) = I_{Rr}$

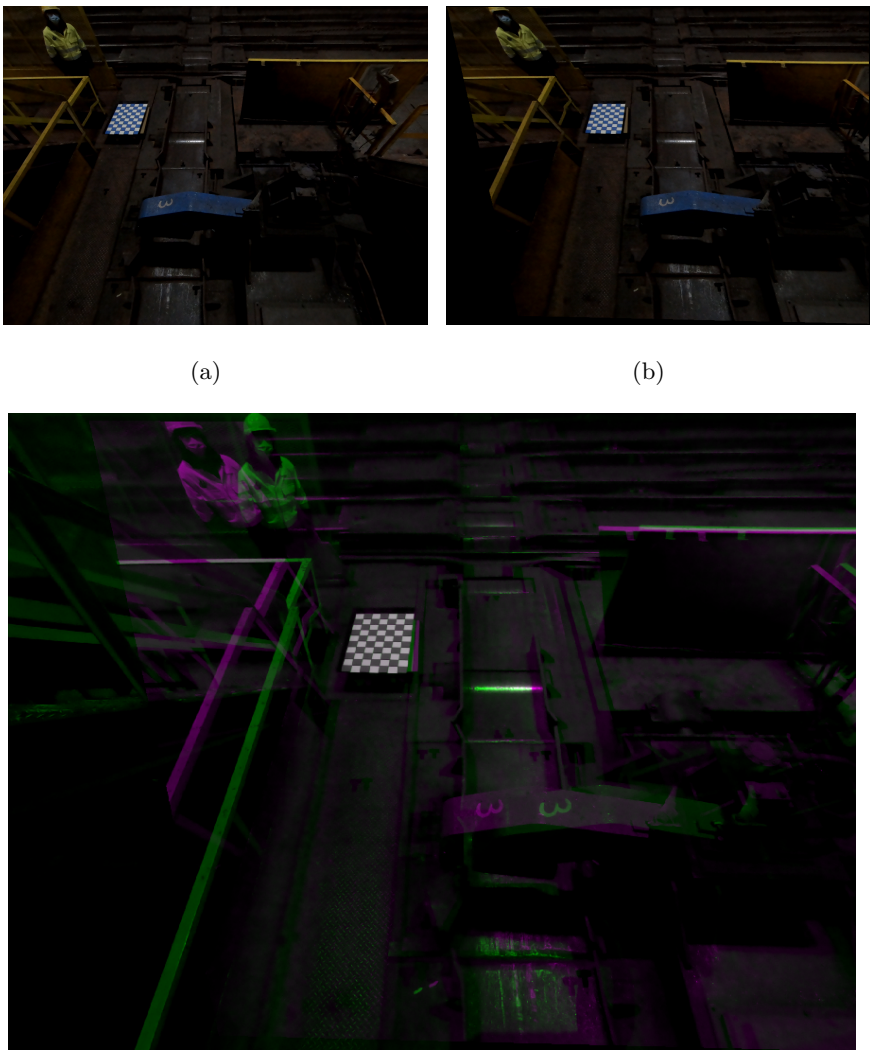
transformed right camera. It can be seen from the image that the checkerboards in the two camera images are perfectly matched. At this moment, according to the projection principle, objects in the same plane as the calibration plate are perfectly mapped in the image. Objects that are not in the same plane as the checkerboard will generate parallax to varying degrees according to the distance from the calibration plane. The farther the object is from the plane, the greater the parallax. It can be seen that the guardrail on the left side, the baffle on the right side and the cutting machine marked with "3" in the middle of the image all generate considerable parallax due to a considerable distance from the calibration plane. In Figure 4.2(c), purple represents the right camera image after registration, and the green represents the left camera image.

Figure 4.2(c) uses the transformation matrix calculated by the checkerboard to register the steel section images. It can be seen that although the checkerboards have perfectly matched, the steel section is higher than the calibration plane, resulting in noticeable parallax of the steel section part in the two images. Obviously, using the data of the calibration plate at this time to measure the steel will result in a significant error.

However, the quality of the image registration could be improved further. The target of measurement is the steel section, and the diameter of the steel section has many specifications, which means that the plane of measurement height is frequently changing. Moreover, the plane where the checkerboard is located is not at the same height as the measurement height. The plane difference between the two images will lead to the fact that although the two images' checkerboards are perfectly overlapped, the fitting degree of the steel parts of interest is limited. A method is needed that is able to assess the fitting quality of steel sections from two images and optimize the registration process.

4.0.3 Registration Evaluation

For the whole measurement system, the accuracy of image registration is measured by the matching degree of steel section parts in the images. So steel parts and their edges need to be identified, which is similar to the use of a monocular camera to measure the size in the previous chapter.



(a) (b)

(c)

Figure 4.2: Checkerboard Image from Right Cam: (a) Original Image; (b) Registered Image (after Tranformation); (c) Show Together with Image from Left Cam.

After obtaining the steel section edges, we select the section area close to the checkerboard for evaluation. Next, positive and negative P_{re} pixels above and below the middle point of checkerboards are selected in order to create the testing area R_{test}. The polygons $P_{l,r}$ are composed of the upper and lower bounds of the detection area and the steel section's edges in the detection area. After $P_{l,r}$ are created, calculate the percentage of the overlapped area between two polygons, which is the quality Q_R of the registration. The higher Q_R is, the better the registration quality is.

Figure 4.4(a) shows the image pairs of the original image taken from the left camera and the right camera's registered images. The green part is from the left camera, and the magenta part is from the right camera. The regions with grayscale colour are overlapped areas with the same intensity.

Figure 4.4(b) are the detected polygons for steel sections from left and registered right

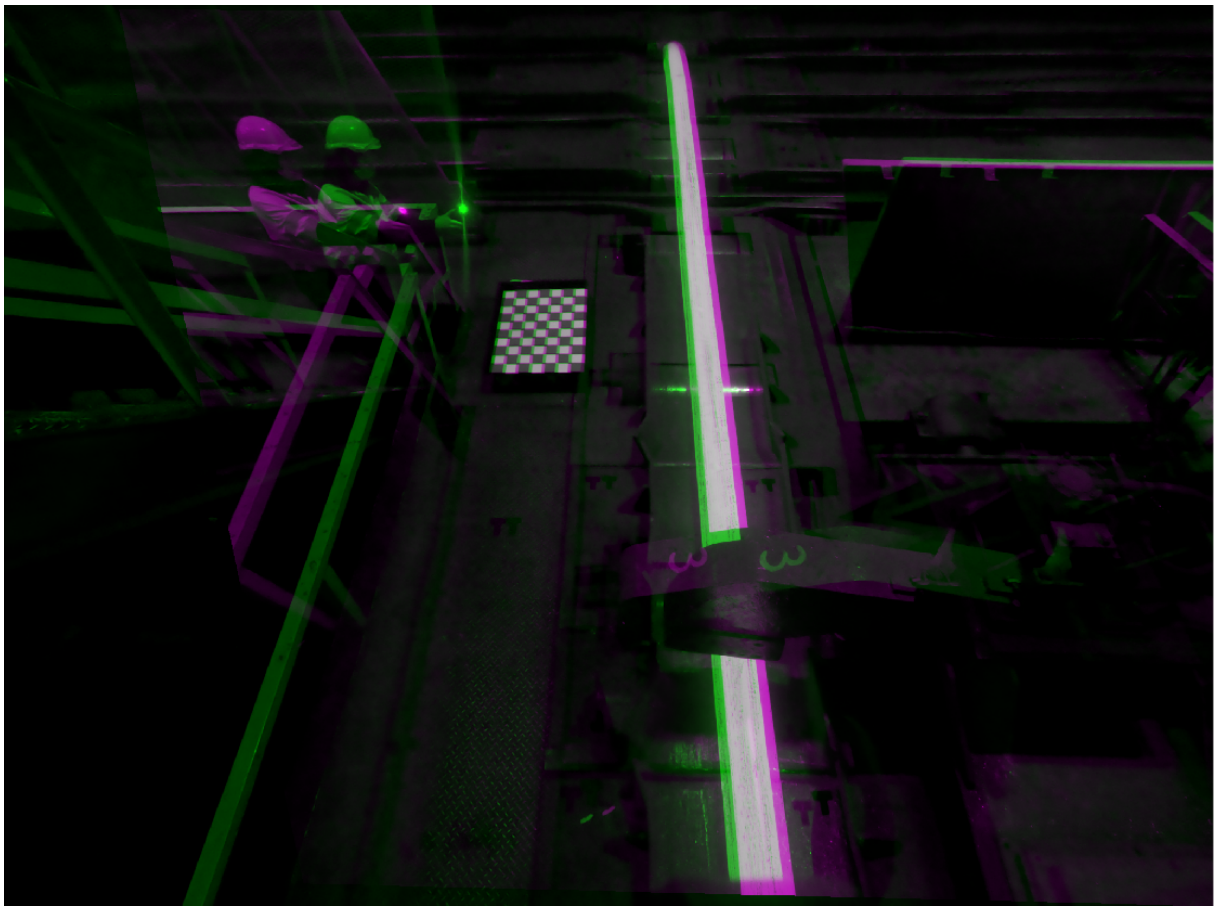


Figure 4.3: Steel Section Images after Registration

Algorithm 12: Registration Evaluation

Input: I_{BOI},P_{re}

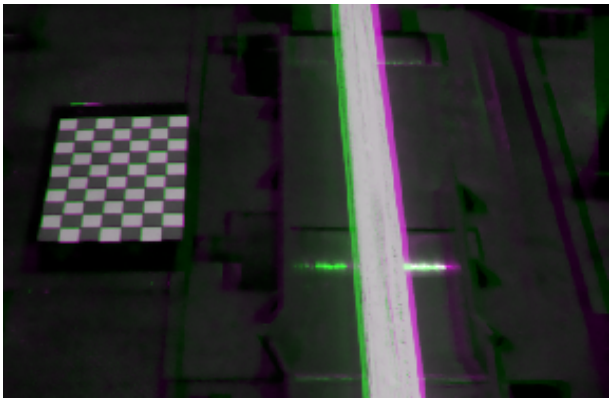
Output: The Quality of Registration Q_R

1: Find the middle point of checkerboard $[x_{board} \quad y_{board}]$

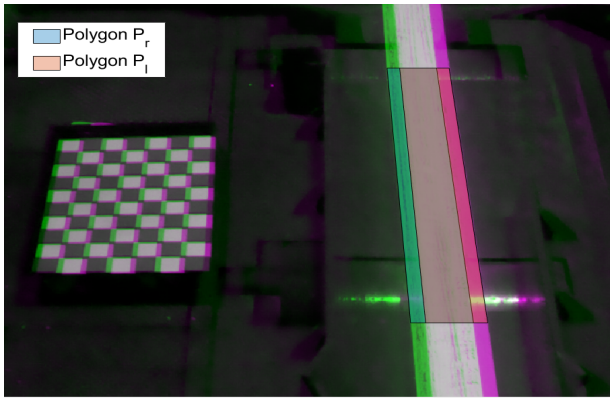
2: Create the testing area R_{test} with bounds $y_{board \pm P_{re}}$

3: Create polygons $P_{r,l}$ with Edges in R_{test}

4: Calculate the percentage Q_R of overlapped area between P_r and P_l



(a)



(b)

Figure 4.4: Registered Steel Sections: (a) Registered Images; (b) Polygons P_r and P_l.

images. The overlapped area Q_R between two polygons over the polygon for left image P_l form the quality of registration.

$$Q_R = \frac{P_r \cap P_l}{P_l}. \tag{4.1}$$

In order to give an direction to this quality of registration, we consider Q_R to be positive when the centroid of P_l is on the left of the centroid of P_r and vice versa.

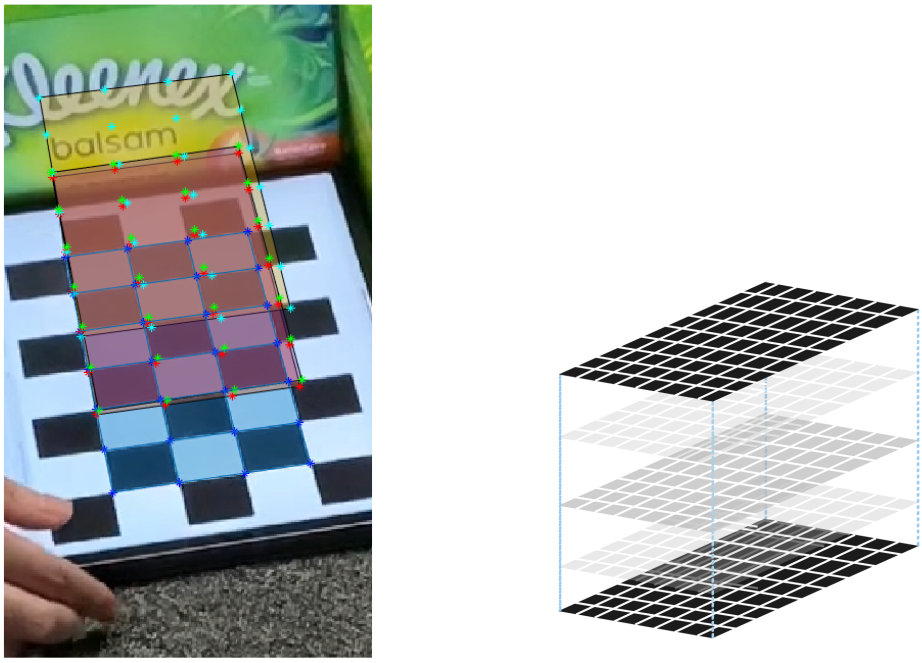
4.0.4 Height Information for Adjustment of Registration Result

By changing the height of the checkerboard from the ground, the calibration plane can be closer to the height of the steel section. The closer the distance between the two, the better the image registration quality is. The most direct method is to take many checkerboard images at different heights during calibration, to obtain the data of the calibration plane in a specific height range. It is time-consuming and laborious to collect the data of several groups of a checkerboard with different heights, hence damaging the reproducibility of the method. So we only collected the checkerboard data at a lower and higher position, and the data within and out the range are generated by interpolation.

When a camera takes an image, it essentially project the real three-dimensional object onto the two-dimensional image plane, and the projection process is linear. The definite ratio point of division between checkerboards with different heights in the real-world is also a linear projection in the image plane Eq. (4.2).

$$\mathbb{P}\left(\frac{(x_1, y_1) + \lambda\,(x_2, y_2)}{1+\lambda}\right) = \frac{\mathbb{P}\,(x_1, y_1) + \mathbb{P}\lambda\,(x_2, y_2)}{1+\lambda}, \tag{4.2}$$

where $\mathbb{P}$ is the projection transformation and $\lambda \in \mathbb{N}^+$.



(a) (b)

Figure 4.5: Virtual Checkerboards witch Different Heights

Therefore, an interpolation process can be realized by directly inserting data points between the checkerboard's corresponding points at different heights. An interpolation process is applied to improve the registration process results and it is described as Algorithm 13. After the interpolation, combined with the previous registration quality Q_R, we can automatically update the registration result. Figure 4.7 shows some virtual calibration boards generated by calibration boards. Theoretically, as long as having two images of calibration boards with different heights, it can generate virtual calibration boards of any height.

According to the value of $100 - |Q_R|$, when Q_R is positive, the algorithm will choose the virtual checkerboard with higher height. When W_R is negative, a lower virtual checkerboard is chosen, which make the virtual checkerboard at the same height as the height of the steel section.

The whole sizing process is shown in the flow chart in Figure 4.6. At the beginning of the

Algorithm 13: Adjust Registration

Input: $C_{LL}, C_{LH}, C_{RL}, C_{RH}$

Output: The adjust registered image I_{Rr}

1: Interpolate n set of corner points of checkerboards C_{Li},C_{Ri}

2: **for** $k = 1, \cdots, n$ **do**

3: Calculate $Q_R i$ for C_{Li},C_{Ri}

4: **end for**

5: Find the I_{Rr} with minimum $100 - |Q_R|$

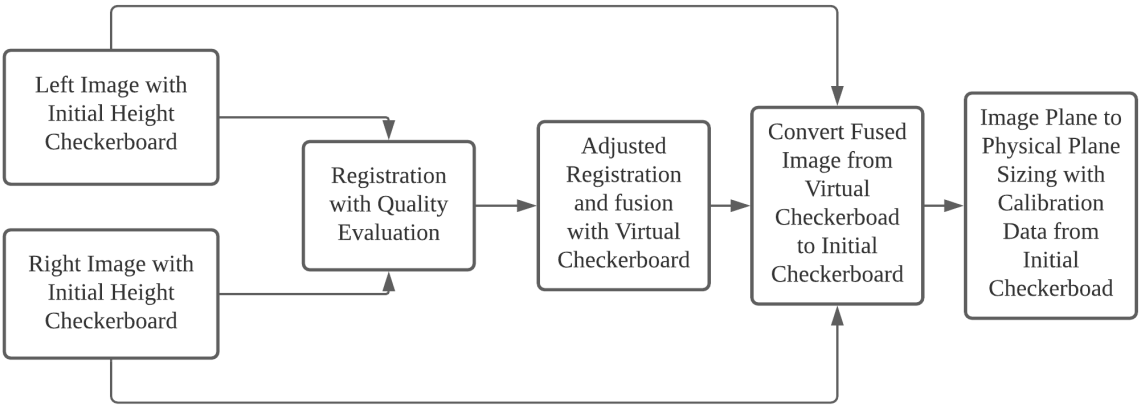


Figure 4.6: Flow Chart of Sizing Process

Figure 4.7: Virtual Checkerboard for Image Registration: The Leftmost Image Shows Image Captured when the Checkerboard is on the Floor; The Middle Image Shows the Checkerboard on Another Height; The Rightmost Image Shows the Virtual Checkerboard at the Desired Height.

process, we had images of the steel section taken by two cameras in the left and right directions simultaneously. At this time, the images have the initial checkerboard, which is used to calibrate the dual cameras. However, we do not know whether the height of these calibration plates is consistent with the actual measurement plane of the steel section and how much difference there is. Then the two images are registered, and the virtual checkerboard corresponding to the measurement plane of the steel section is generated by assessing the registration quality in Section 4.0.3. After that, the whole image is transformed by calculating the geometric transformation between the virtual checkerboard and the initial checkerboard. In this way, we can make the measurement plane coincides with the initial checkerboard's height.

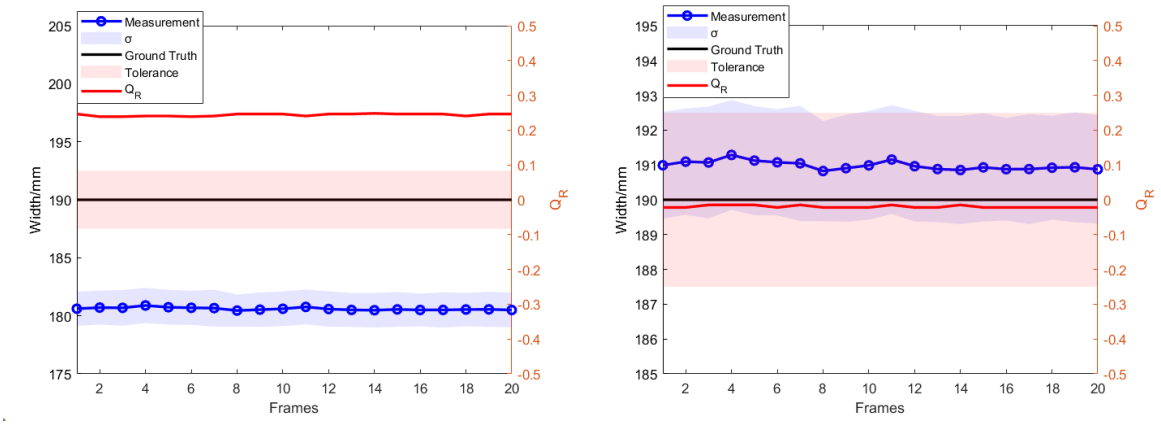
As Figure 4.7 shows, when using the checkerboard is set on the ground, the cameras use the calibration data on a different plane from the measurement plane. Therefore, the measured results will be larger than the actual results in this case. In order to correct this measurement error, the position of the checkerboard should be raised to the height consistent with the steel radius to make the height of the measurement plane compatible with the steel radius. Through the virtual calibration plate, the calibration plate's height can be freely moved as shown in in Figure 4.7 (c). Through the measured value of Algorithm 13, the height of the virtual calibration plate can also be ensured, as shown in Figure 4.7 (b).

When the registration process is completed, with the camera internal and external parameters obtained from the calibration process and the previous work [60], the steel section size can be calculated and converted from the image plane to the physical plane.

In the considered steel production case study, two hot rolling bar (HRB) edges are expected within the sliding window $I_{H\times W}$, with height H and width W. Therefore, we set up all n_i to be equal to $n = 2$. The $\mathbf{x}_{ij}$ and $\mathbf{y}_{ij}$ coordinates from the image plane are converted to coordinates in the physical plane through the transformation

$$\begin{bmatrix} x_{ij}^w \\ y_{ij}^w \\ z_{ij}^w \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \mathbf{K} \begin{bmatrix} x_{hj}^I \\ y_{hj}^I \\ 1 \end{bmatrix}, \tag{4.3}$$

where $\mathbf{R}$ and $\mathbf{T}$ are respectively the rotation and translation matrices, and $\mathbf{K}$ is the intrinsic matrix of the camera parameters. These matrices are obtained via the calibration process. The coordinates $x_{ij}^I \in \mathbf{x}_{ij}$ and $y_{ij}^I \in \mathbf{y}_{ij}$ are from the image plane and $[x_{ij}^w, y_{ij}^w, z_{ij}^w, 1]^T$ is the vector of corresponding coordinates in the physical plane. Given the vectors $I_{i1} = [x_{i1}^I, y_{i1}^I]^T$ and $I_{i2} = [x_{i2}^I, y_{i2}^I]^T$ on two HRB edges with $x_{i1}^I = x_{i2}^I$, the diameter l of the HRB is then calculated through



(a) (b)

Figure 4.8: Sizing Results: (a) Q_R is positive and $|Q_R|$ is large; (b) Q_R is negative and $|Q_R|$ is small.

Table 4.1: RMSE of Fig.4.8(a),4.8(b)

	a	b
RMSE	9.3741	0.9872

$$l = \|P_1 - P_2\|_2 \,, \tag{4.4}$$

where $P_1 = [x_{i1}^w, y_{i1}^w]^T$ and $P_2 = [x_{i2}^w, y_{i2}^w]^T$ are the physical plane correspondences to I_{i1} and I_{i2}. Here $\|.\|_2$ denotes the Euclidean norm.

Figure 4.9 shows the measurement results of seven images in a video sequence. The seven frames are separated by 100 frames, showing the measurement process of 700 frames. 190 mm is the target diameter of steel rolling. The blue data points are measurement results with virtual checkerboards and red data points are direct estimate results without virtual checkerboard. Figure 4.8(a) and Figure 4.8(b) show the sizing results with different checkerboard parameters and how $|Q_R|$ evaluates the sizing quality. In Figure 4.8(a), the sizing results are underestimated. The Q_R values are large and positive, showing that the true size should be much larger than the estimation. In Figure 4.8(b), a small negative Q_R shows the true size is slightly smaller than the estimated value.

4.1 Summary

This chapter presents various feature extraction and matching algorithms for image registration. These types of algorithms can autonomously extract feature information within images, and complete image registration by pairing features in different images and calculating the trans-

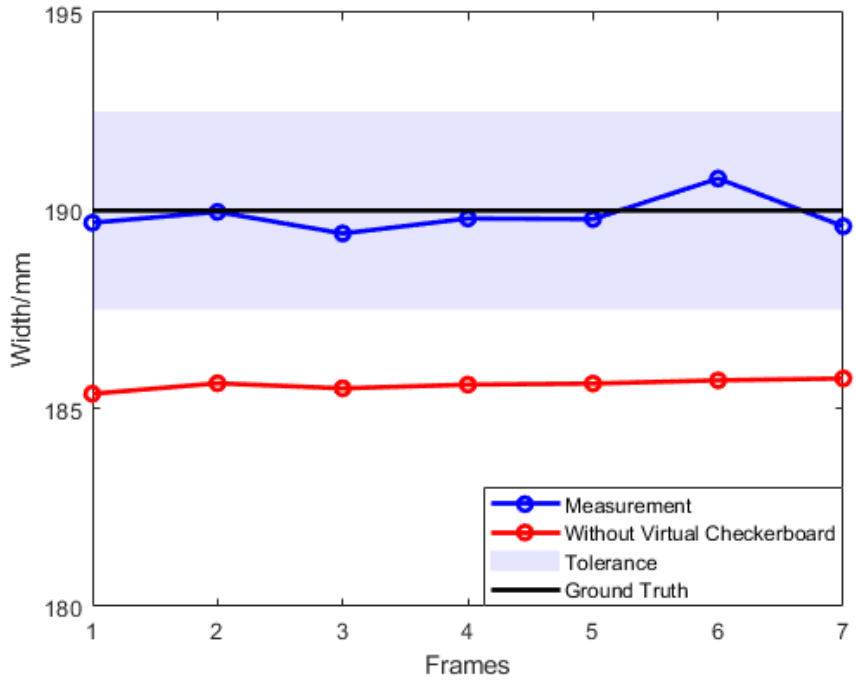


Figure 4.9: Sizing Results for Seven Different Frames

formation matrix via these feature coordinates. Moreover, mature feature extraction algorithms can exhibit rotation and scale invariance, thereby maximizing the consistency of high-quality feature points between images.

In practical factory measurement applications, the monocular measurement method detailed in Chapter 3 cannot fulfill the accurate measurement of steel of varying sizes due to the dimensional conversion error brought about by the misalignment between the measurement plane and the calibrated reference plane. To identify and correct the distance between the two planes, a new measurement system was implemented, using a dual-camera system with two GoPro cameras positioned side by side.

The accuracy of binocular measurement over medium and long distances is largely determined by the camera resolution and the distance between the two cameras. Higher camera resolution results in each pixel representing a smaller actual size, thereby directly improving measurement accuracy. The distance between the cameras determines the disparity between the images captured by the dual cameras. The greater the distance, the larger the disparity, enabling objects far from the camera to still produce recognizable parallax.

Due to the complexity of the actual factory environment and in order to enhance the measurement capability and accuracy over medium and long distances, the distance between the cameras in the dual-camera system is set relatively large. This substantial separation results in significantly different views from the two cameras and generates large parallax. In such complex situations, the quality of feature points obtained using automated feature recognition algorithms

tends to be low. The number of feature points that can match each other in different camera images needs to be increased. Therefore, a checkerboard is still used as a calibration reference, providing the feature points for image registration during the actual measurement process.

Image registration using the corners of the checkerboard as a reference can initially pair the left and right cameras. However, further adjustments are necessary for high-precision visual measurements. Since the error of the monocular camera system is caused by the misalignment between the measurement plane and the calibration plane, the solution proposed by the binocular system is to align the calibration plane with the measurement plane. The measurement plane is determined by the size of the steel and the height of the transport zone, so it is at a fixed height and is not a variable that can be altered. This implies that the measurement system can only opt to adjust the calibration plane to coincide with the measurement plane. The position of the checkerboard determines the calibration plane's position. By continuously changing the checkerboard's position, any calibration plane data can be acquired. However, physically moving the checkerboard directly and continuously to match different sizes of steel is evidently time-consuming and impossible on some production lines. Therefore, a virtual checkerboard is proposed to adjust the calibration plane's position. By obtaining two images of the checkerboard at different heights, a checkerboard at any height can be theoretically interpolated and extrapolated.

After obtaining a virtual checkerboard that can be at any height, the next question is how to select the appropriate virtual checkerboard to complete the image registration. This chapter proposes an algorithm for calculating the accuracy of steel image registration. The degree of overlap of the steel section parts near the target area in the image after registration serves as a measure of registration accuracy. A higher degree of overlap in the steel section part indicates greater registration accuracy. Simultaneously, the registration accuracy will provide a direction based on the location of the non-coincident portions of the left and right camera steel section images. With this direction and accuracy measurement, the system can determine whether it needs to retrieve a virtual checkerboard at a higher or lower height. In this way, by combining registration accuracy with the virtual checkerboard, the virtual checkerboard with the highest accuracy is chosen as the calibration plane and used as the reference for the dimension conversion process. The calibration plane created by the selected virtual checkerboard can align with the measured plane, thus solving the measurement error caused by plane mismatch.

The system's performance is evaluated using various real data and different metrics. The results show that a high precision sizing performance with a tolerance range less than $2mm$ is achieved, contributing to the quality assurance of manufacturing tasks. The achieved remote sizing accuracy exceeds 95%, thanks to the efficient registration approach that combines extrinsic image features with accurate image registration algorithms.

Chapter 5

Image Fusion

5.1 Practical Industrial Task

In the preceding chapter, the combination of the virtual checkerboard and the image registration evaluation algorithm has resulted in highly accurate image registration between the two cameras. Consequently, the positions of the steel section parts captured by the left and right cameras in the image are now perfectly aligned. While utilizing a single image after registration and a virtual checkerboard for visual measurement can yield high-precision results, the potential benefits of the dual camera setup have yet to be fully exploited. By fusing the images from the dual cameras, the details of the steel section parts can be enhanced, edges can be sharpened, and missing edges can be filled. This image fusion process facilitates more precise edge detection algorithms, ultimately improving the robustness of the sizing process.

5.1.1 FFT Fusion

The FFT is an important image processing method, which applies the discrete Fourier transform to the image and changes the image information from the spatial domain to the frequency domain. The left and right images are fused by combining the phase and magnitude maps of two images in the frequency domain. The two dimensional (2D) discrete Fourier transformation for a image $f[m,n]$ of size $m \times n$ is defined as:

$$F_{k,l} = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f[m,n] e^{-j2\pi\left(\frac{k}{M}m + \frac{l}{N}n\right)} \tag{5.1}$$

where $F[k,l]$ can be decomposed into its amplitude $\|F[k,l]\|$ and phase $\angle F[k,l]$. To fuse two images I1 and I2 in the frequency domain, the fusion rule is given as Algorithm 14. The discrete Fourier transform is implemented as a FFT.

According to algorithm.14, the two images are first Fourier transformed to extract the amplitude and phase after Fourier transformation. Then, compare the amplitude of the two images at each pixel location, reserving the amplitude and phase of the image with a larger

Algorithm 14: FFT Fusion

Input: $F_{I1}[k,l], F_{I2}[k,l]$

Output: The fused image I_F

Calculate the magnitude $\|F[k,l]\|$ and phase $\angle F[k,l]$
for $k = 1, \cdots, M$ **do**
  for $l = 1, \cdots, N$ **do**
    if $\|F_{I1}[k,l]\| > \|F_{I2}[k,l]\|$ **then**
      $\|F[k,l]\| = \|F_{I1}[k,l]\|$
      $\angle F[k,l] = \angle F_{I1}[k,l]$
    else
      $\|F[k,l]\| = \|F_{I2}[k,l]\|$
      $\angle F[k,l] = \angle F_{I2}[k,l]$
    end if
  end for
end for
Inverse FFT $F[k,l] \rightarrow I_F$

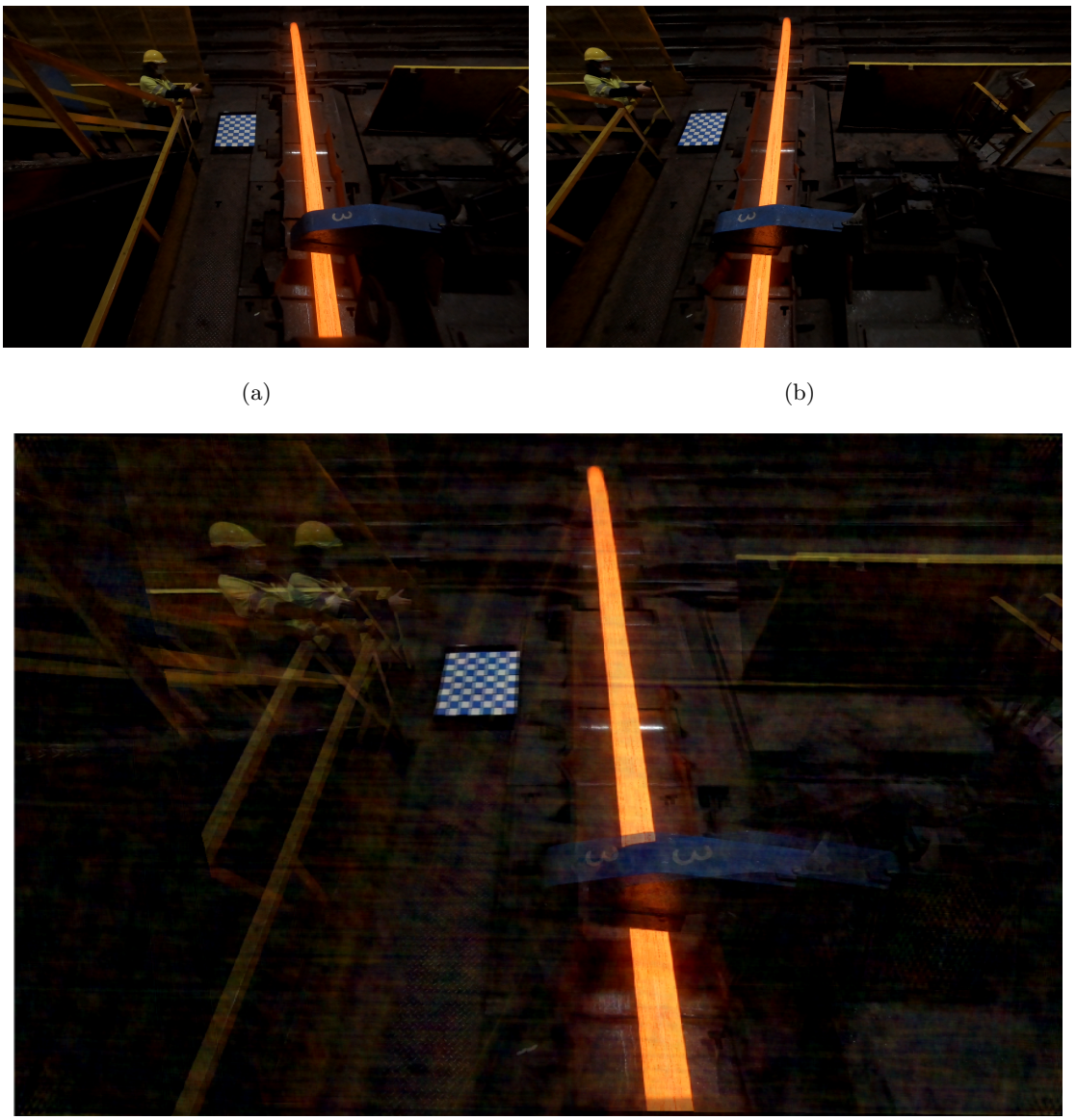
amplitude as the amplitude and phase of the fused image. Finally, the image is restored using the inverse Fourier transform.

Figure 5.1 shows the result of Fourier image fusion. Figures.5.1(a) and 5.1(b) are the original images taken by the left and right cameras, respectively. Figure 5.1(c) is the result of Fourier image fusion. As the image shows, due to the previous image registration work, there is no left-right duplication of the steel part, as opposed to the other parts of the picture, which are impossible to overlap. As a result, the sharpness of the steel section parts and the edges are much better than the rest of the image.

With image fusion technology, the blind spot of a single camera perspective can be compensated for by an image taken by another camera. As shown in Figure 5.2, the area circled by a red circle in the picture is the part that a single camera has not captured. Some missing steel was restored by fusion technology. It can be seen that if the size of the obscure object is relatively small or if the obscured object plane is farther from the measured plane (with a large parallax), the covering of the obscured object on the steel can be directly ignored by the image fusion.

5.1.2 Discrete Wavelet Transform Fusion

Image fusion based on a discrete wavelet transform (DWT) [47] is also a common and popular method in image fusion. In essence, a two-dimensional DWT applies several one-dimensional discrete wavelet transforms based on the selected wavelet type in the horizontal and vertical directions of the image. The choice of wavelets affects the image fusion results.



(a) (b)

(c)

Figure 5.1: Input Steel Section Images and FFT Fusion Results: (a) Left Camera Image; (b) Right Camera Image;(c) FFT Fusion Results.

The process of two-dimensional discrete wavelet transformation of the image is shown in Figure 5.3. First, the image is transformed with a horizontal one-dimensional wavelet to obtain low frequency information L and high frequency information H. Then the image is longitudinally decomposed to obtain four frequency bands LL_1, LH_1, HL_1 and HH_1. LL represents horizontal low frequency, vertical low frequency, LH represents horizontal low frequency, vertical high frequency, HL represents horizontal high frequency, vertical low frequency, HH represents horizontal high frequency, vertical high frequency. In the next layer, the decomposition is another horizontal and vertical DWT of LL's dual low frequency information.

Discrete wavelet transform fusion is to first perform discrete wavelet transform on the image and fuse the discrete wavelet coefficients of the two transformed images through the set

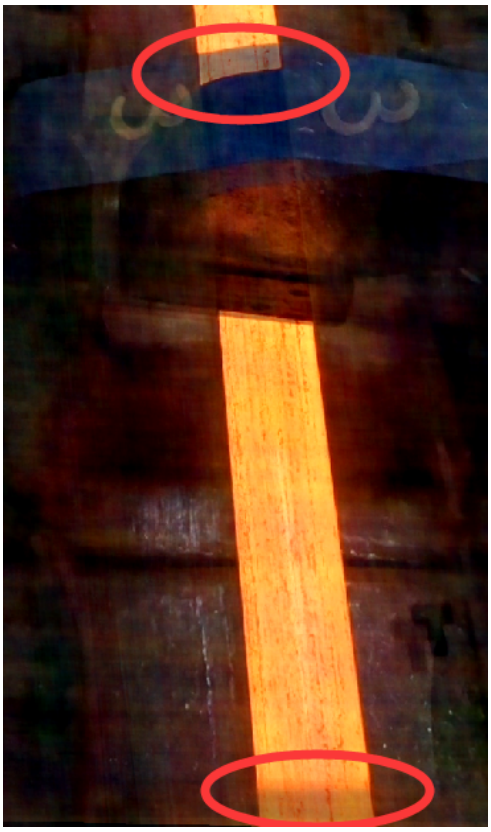


Figure 5.2: Fusion for Inpainting

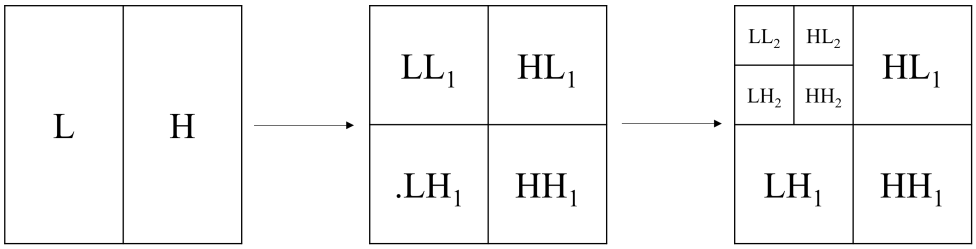


Figure 5.3: Discrete Wavelet Transform

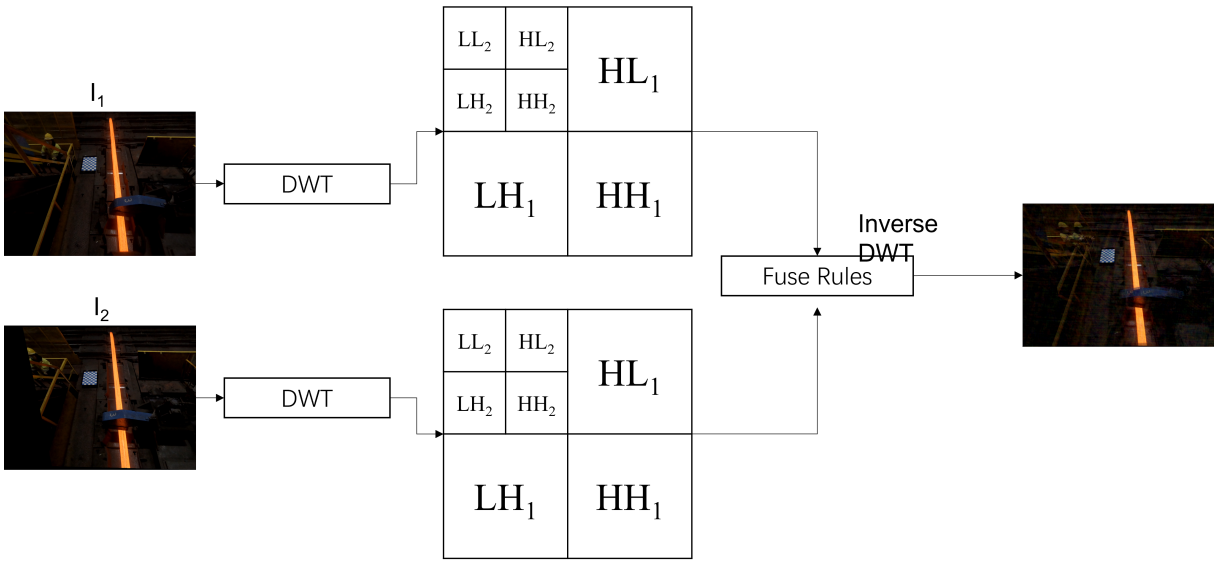


Figure 5.4: Discrete Wavelet Transform Fusion

fusion rules. For example, the absolute value method, the information with larger coefficient after transformation is retained as the result of fusion; The weighted average method is used to weighted average the transformed coefficients. After the new wavelet transform coefficients are obtained, the fused image is obtained by inverse transform.

DWT-Daubechies Wavelets

In this dissertation, the DWT is implemented with Daubechies wavelets [11,23]. In the evaluation and validation process, a different number of wavelet coefficients are retained. The DWT method results are presented with 2, 4,8 and 16 coefficients and this is denoted as DB2, DB4,DB8 and DB16, respectively.

DWT-Fejér-Korovkin wavelets

In addition to DB wavelets, Fejér-Korovkin wavelets (FK) wavelets transform are applied in DWT fusion. The FK wavelets are denoted as FK4, FK6, FK8 and FK18, respectively, according to different filter coefficients. The FK wavelets have shown better high-frequency performance than other waveforms [44].

5.1.3 Performance Validation and Evaluation

In this case, since there is no ground truth image available for evaluating the fusion performance, evaluation metrics that do not require reference images are utilized. The evaluation is conducted from three different aspects, namely information content, image contrast, and the dissimilarity between the fused image and the original image.

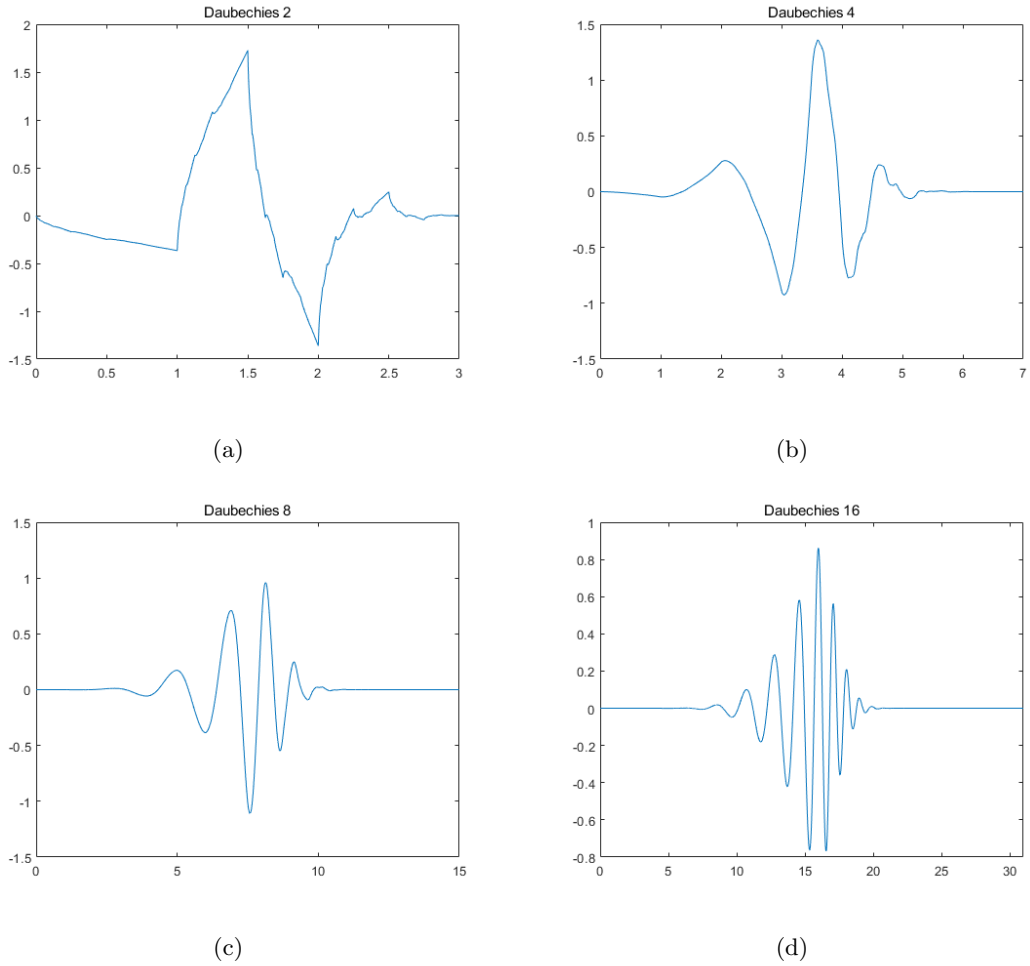


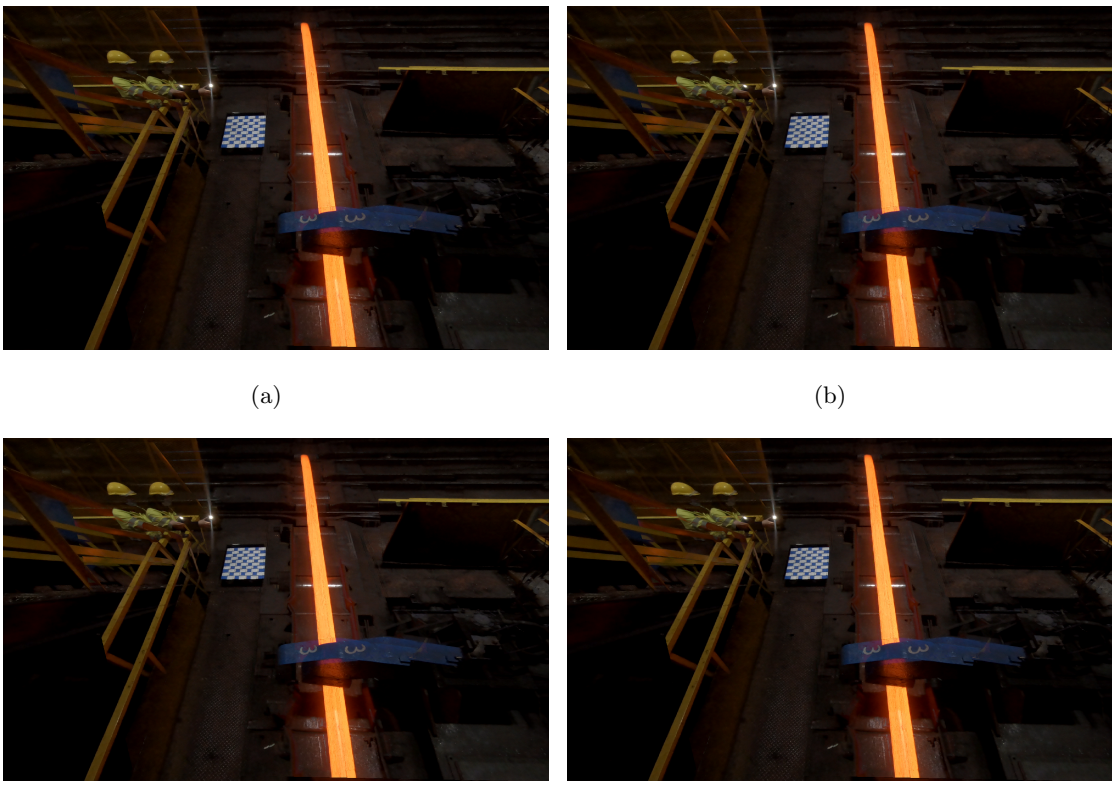
Figure 5.5: Daubechies Wavelets: (a) DB2; (b) DB4;(c) DB8;(d) DB16.

For information content evaluation, metrics such as entropy and mutual information are employed to measure the amount of information present in the fused image. Higher values indicate a greater amount of information captured in the fused image.

Image contrast evaluation assesses the ability of the fusion algorithm to enhance the visual contrast of the fused image. Metrics such as standard deviation and spatial frequency are utilized to quantify the contrast improvement achieved by the fusion process.

To evaluate the difference between the fused image and the original image, metrics such as structural similarity and sum of the correlation of differences are utilized. These metrics measure the similarity between the fused image and the original image in terms of structural patterns and pixel intensity differences.

By analyzing the performance of the fusion algorithm using these evaluation metrics, we can assess its effectiveness in terms of information preservation, contrast enhancement, and similarity to the original image. [23, 26] The metrics we used are:



(a) (b)

(c) (d)

Figure 5.6: Daubechies Wavelets Fused Results: (a) DB2; (b) DB4;(c) DB8;(d) DB16.

Information Entropy H

$$H(X) = -\sum P \log_2 P, \tag{5.2}$$

where P_i is the normalized histogram and X here denotes the considered image. The amount of information contained in the fused image can be measured by information entropy. Images with more information have higher information entropy. The unit of entropy is $bit/pixel$.

Standard Deviation SD

$$SD = \sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(f(i,j) - \bar{\mu}\right)^2 / MN} \tag{5.3}$$

where f is the intensity of pixel $[0, 255]$, $\bar{\mu}$ is the average intensity of pixels, M and N are the width and height of image. The image standard deviation reflects the discrete degree of the image pixel brightness and mean value. The larger the standard deviation is, the more pronounced the contrast between light and dark is.

Spatial Frequency [14] SF

$$SF = \sqrt{(RF)^2 + (CF)^2} \tag{5.4}$$

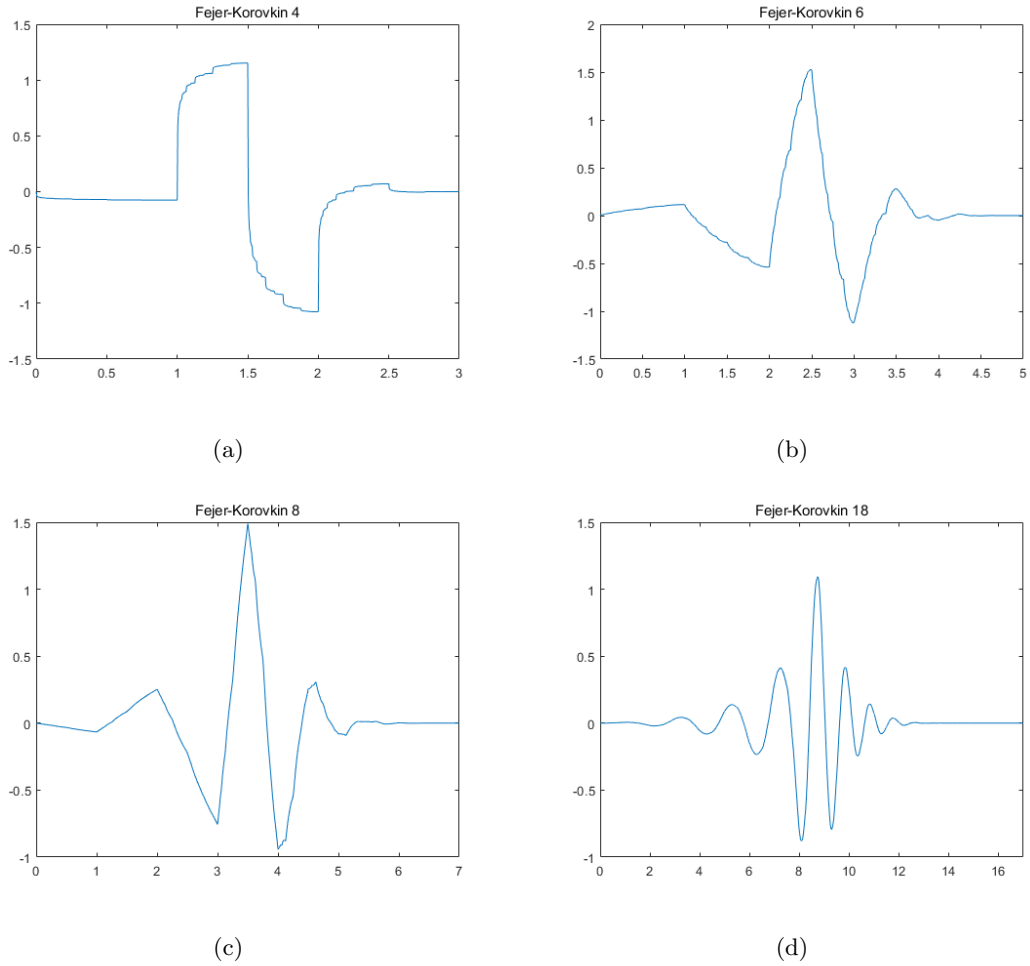


Figure 5.7: Fejér-Korovkin Wavelets: (a) FK4; (b) FK6;(c) FK8;(d) FK18.

where RF is the row frequency and CF is the column frequency. Spatial frequency is the frequency of calculating the image row and column respectively. Generally speaking, higher frequency means better image quality.

Average Gradient AG

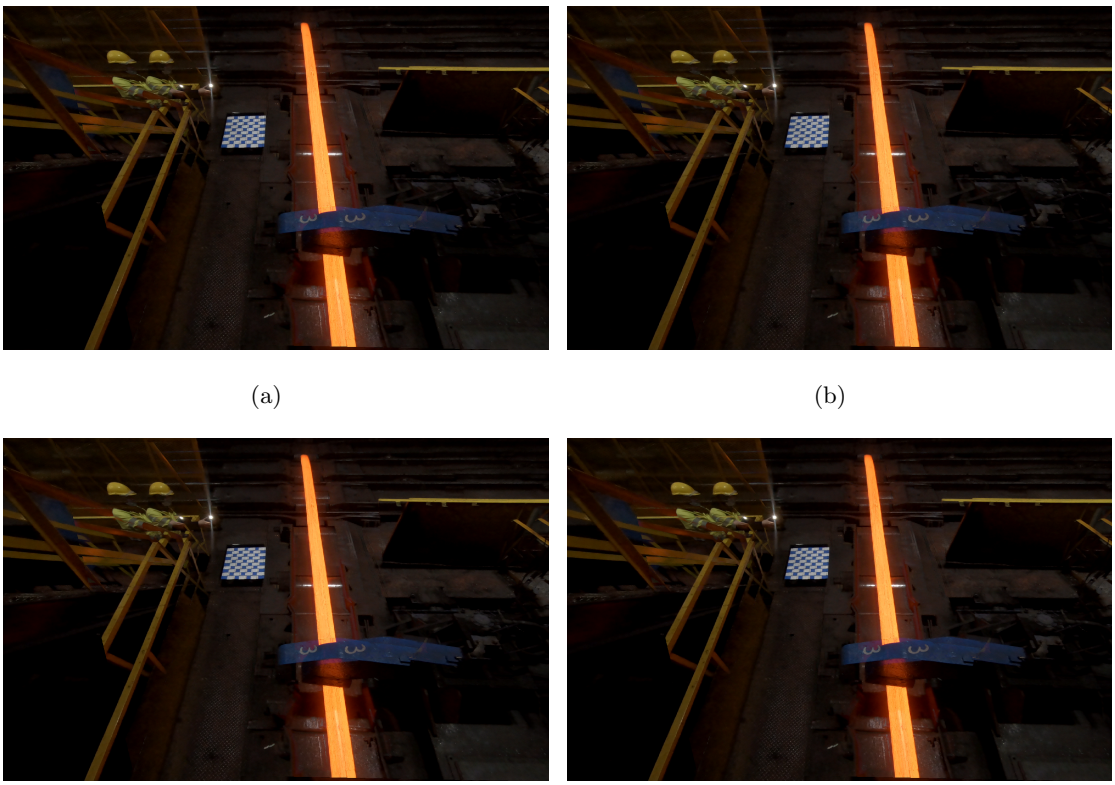
$$AG = \frac{1}{(H-1)(W-1)} \sum_{x} \sum_{y} \frac{G(x,y)}{\sqrt{2}} \tag{5.5}$$

where H and W are height and width of image, G is the gradient magnitude of image. The first-order difference between the pixel value of a pixel and its adjacent pixels reflects the edge information of the pixel. The magnitude of this rate of change can be used to represent the clarity of the image.

Feature Mutual Information [19] FMI

$$FMI = I_{FA} + I_{FB} \tag{5.6}$$

where I_{FA} and I_{FB} are the mutual information between image A,B and fused image F. The FMI evaluates the dependency between input images and fused image. A larger FMI usually



(a) (b)

(c) (d)

Figure 5.8: Fejér-Korovkin Wavelets Fused Results: (a) FK4; (b) FK6;(c) FK8;(d) FK18.

means a better fusion quality.

Sum of the Correlation of Differences [4] SCD

$$SCD = r(D_1, S_1) + r(D_2, S_2) \tag{5.7}$$

where D_i is the difference between the input image S_i and the fused image, $r\,(.)$ denotes the correlation function.

Edge-Based Structural Similarity [9] $ESSIM$

$$ESSIM = function(l\,(I_1, I_2)\,, c\,(I_1, I_2)\,, e\,(I_1, I2)\,) \tag{5.8}$$

$$N_{essim}(I_f) = \left(1 - \frac{ESSIM(I_1, I_f) + ESSIM(I_2, I_f)}{2}\right) \times 1000 \tag{5.9}$$

where $l(I_1, I_2)$ is a function characterising the luminance difference between images I_1 and I_2, $c(I_1, I_2)$ is a function for the contrast comparison and $e(I_1, I_2)$ is function for the edge comparison between the two considered images. Compared with the original SSIM, $ESSIM$ uses edge comparison to replace the original structural comparison. This makes the metrics more sensitive to the edge information, which is more critical for the sizing algorithm proposed

in this dissertation. Here, we consider the $ESSIM$ between fused image I_f and original images I_1, I_2 as shown in equation 5.9. The lower the value of N_{essim} is, the better the fusion quality is.

Table 1 presents average results over 4 videos, each containing 100 frames. As can be seen from Table 1, the fused images from both cameras by different methods are evaluated under different metrics. For metrics reflecting information contained, the FFT and DWT with high order coefficients approach have better results. These fusion results contain more details and textures. FFT gives best results under SD, FMI, SCD and $ESSIM$. DWT fusion with 16 coefficients DB wavelet has larget entropy H. For metrics related to contrast and the clarity of edges, DWT with 4 coefficients FK wavelet lead better fusion results. On the whole, FFT fusion contains more information from original images, DWT fusion with FK 4 wavelet gives results with high contrast, which benefit the edge extraction process.

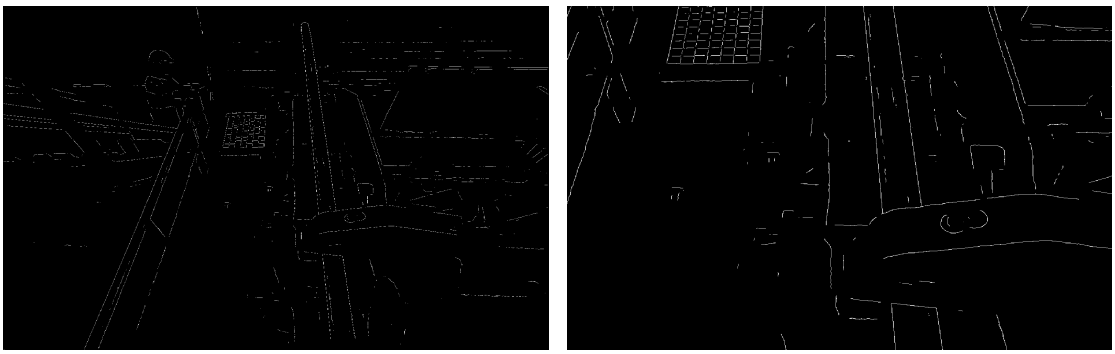
Table 5.1: Fusion Performance Evaluation Results

Metrics	H	SD	SF	AG	FMI	SCD	ESSIM
Left Image	1.4805	53.9503	6.5669	4.3598	-	-	-
Right Image	1.5052	54.3409	6.3998	4.0942	-	-	-
FFT	1.5056	**56.4264**	6.8742	4.8237	**0.9663**	**1.3364**	**0.1458**
DWT-DB Wavelets 2	1.5334	53.3971	7.0793	5.4118	0.9610	0.7495	0.1792
DWT-DB Wavelets 4	1.5492	53.1009	7.0279	5.3707	0.9622	0.7502	0.1791
DWT-DB Wavelets 8	1.5579	52.9425	6.9745	5.3455	0.9609	0.7370	0.1794
DWT-DB Wavelets 16	**1.5738**	52.6410	6.9257	5.3080	0.9593	0.7375	0.1799
DWT-FK Wavelets 4	1.5185	53.5126	**7.1620**	**5.4216**	0.9612	0.7606	0.1712
DWT-FK Wavelets 6	1.5392	53.2564	6.9780	5.3429	0.9626	0.7485	0.1795
DWT-FK Wavelets 8	1.5455	53.1555	7.0109	5.3871	0.9624	0.7548	0.1803
DWT-FK Wavelets 18	1.5628	52.9686	6.9345	5.3159	0.9606	0.7486	0.1780

Image fusion enhances the original image, so the computer can better process the image content. The above image fusion quality indicators can reflect the quality of the fused image to a certain extent in view of the information contained. However, whether the fused image benefits edge detection and vision measurement require further testing.

Figure 5.9 depicts the edge of the steel section image captured by the left camera using the structural random forest edge detection algorithm, as discussed in Section 3.1. The original image has a higher resolution, while Figure 5.9(b) displays a locally enlarged view. For edge binarization, a threshold value of 30% has been selected.

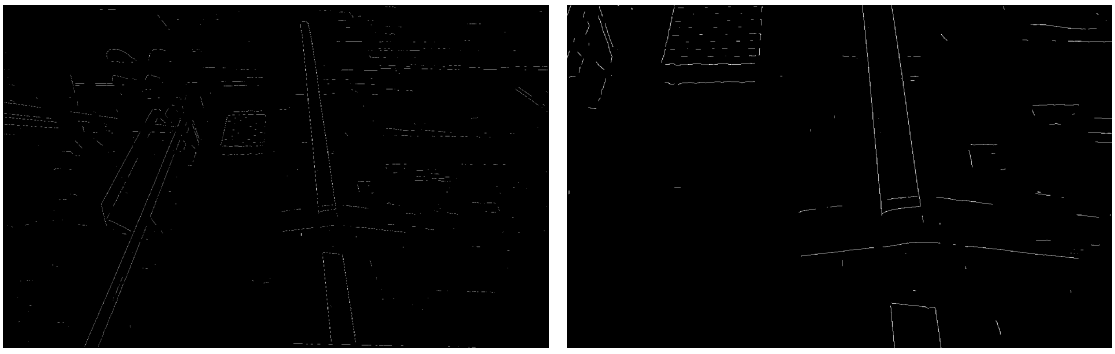
From the figure, it is evident that the algorithm effectively recognizes the edges of the steel



(a) (b)

Figure 5.9: Original Image Edge Detection: (a) Image Edge from Left Cam; (b) Zoomed Image Edge from Left Cam.

section. However, it is important to note that some misidentified edges are present due to the presence of texture surrounding and on the steel. These misidentifications can be attributed to the complexity of the steel's surface and the challenges associated with distinguishing between genuine edges and texture-related variations.

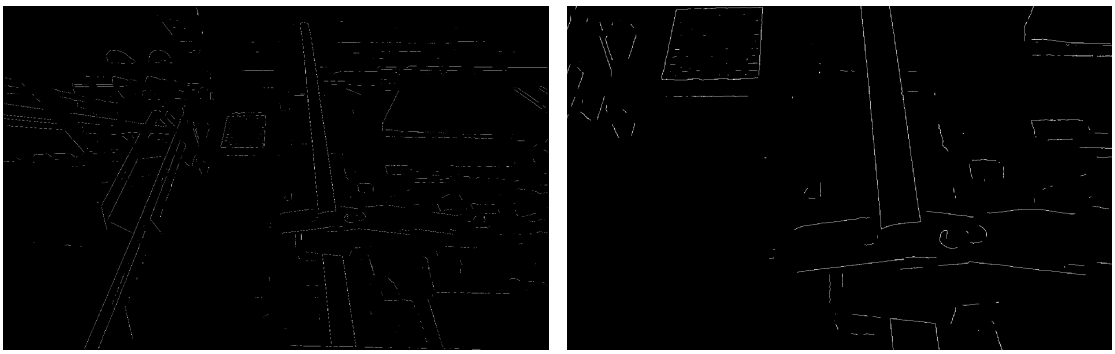


(a) (b)

Figure 5.10: FFT Fused Image Edge Detection: (a) FFT Fused Image Edge; (b) Zoomed FFT Fused Image Edge.

Figure 5.10 showcases the edges obtained by applying the edge detection algorithm to the FFT fused image. The parameters used for the edge detection algorithm are consistent with those in Figure 5.9 mentioned earlier. In Figure 5.10(b), it can be observed that the presence of noise edges near the steel section has significantly reduced, and the incorrect edges caused by textures on the steel are no longer visible. Simultaneously, the edges of the steel section necessary for visual measurement have been preserved entirely.

This demonstrates that employing FFT image fusion prior to edge extraction yields better results compared to directly extracting edges from the original image. The FFT image fusion effectively reduces noise and improves the accuracy of the detected edges, ensuring that the steel section edges crucial for visual measurement are accurately represented.



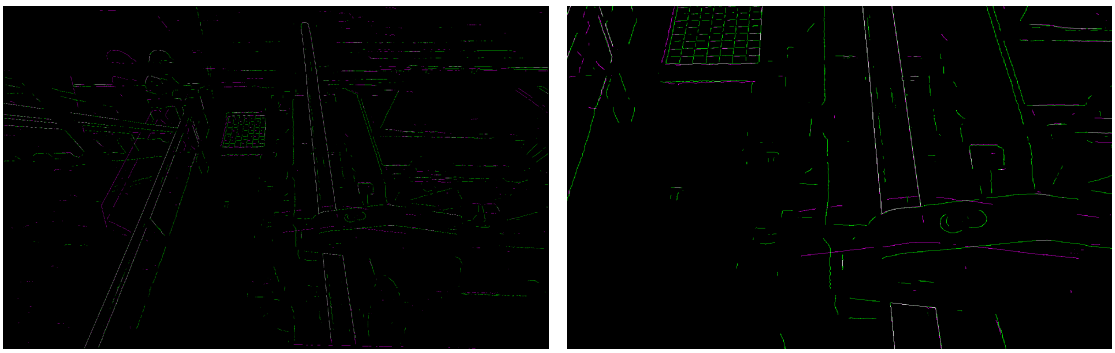
(a) (b)

Figure 5.11: DWT Fused Image Edge Detection: (a) DWT Fused Image Edge; (b) Zoomed DWT Fused Image Edge.

Figure 5.11 displays the result of edge extraction from the image fused using FK4 discrete wavelet transform. The obtained edges exhibit similar characteristics to those generated by FFT fusion and demonstrate superior performance compared to edge extraction from the original image.

The application of FK4 discrete wavelet transform fusion enhances the quality of the edge detection process, resulting in more accurate and well-defined edges. The fused image retains the essential edges of the steel section while effectively reducing noise and improving edge detection performance.

The subsequent analysis will provide further comparison and evaluation of the differences between FFT and DWT results.



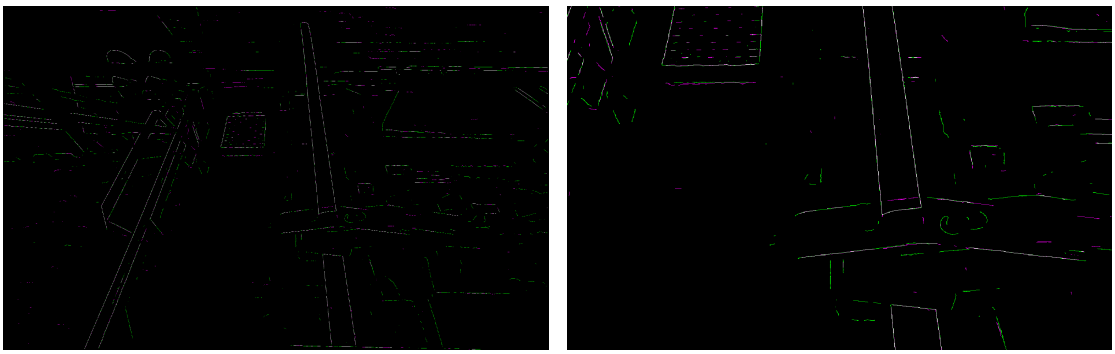
(a) (b)

Figure 5.12: Left Cam Image Edge Compared with FFT Fused Edge: (a) Left Image, FFT Fused Image; (b) Zoomed Left Image FFT Fused Image.

Figure 5.12 provides a visual comparison between direct edge extraction from the left camera image and edge extraction after fusion using the FFT image. The green line represents the result of edge extraction from the left camera image, the magenta line represents the edge extraction result after FFT image fusion, and the white line represents the overlap of the two.

It is evident that both methods successfully extract the edges of the steel section, as indicated by the overlapping white line. However, direct edge extraction from the camera source image tends to identify numerous unnecessary edges, particularly the green edges present near and on the steel in the image. On the other hand, the edges obtained after FFT image fusion exhibit significantly improved performance in this regard.

The FFT image fusion effectively reduces noise and enhances the clarity of the extracted edges. Unwanted edges caused by texture and other factors are minimized, resulting in a cleaner and more accurate representation of the steel section edges. This demonstrates the superiority of FFT image fusion in improving the quality and reliability of edge extraction in comparison to direct extraction from the camera source image.



(a) (b)

Figure 5.13: DWT Fused Edge Compared with FFT Fused Edge: (a) DWT FFT Fused Image Edges; (b) Zoomed DWT FFT Fused Image Edges.

Figure 5.13 provides a comparison between edges detected from the FFT fused image and the DWT fused image. The magenta line represents the edges obtained from the FFT image fusion, while the green line represents the edges obtained from the DWT image fusion.

It can be observed that the DWT image fusion produces more detailed edges compared to the FFT fusion. As a result, additional edges are detected in the background items, which are not related to the steel sections. However, these additional edges do not have any significant impact on the performance of the edge extraction algorithm for steel edges.

The DWT fusion method exhibits better sensitivity to small-scale details in the image, resulting in the detection of more fine edges. While these extra edges may not be relevant to the steel sections, they do not interfere with the accuracy and reliability of the steel edge extraction process. Thus, the DWT fusion approach demonstrates its capability in capturing intricate details in the image, without compromising the effectiveness of the steel edge extraction algorithm.

5.2 Summary

After image registration, image fusion is necessary to combine the image information captured by the dual cameras, thereby enhancing the image quality.

In practical industrial tasks, Fourier image fusion and wavelet transformation fusion are employed to merge the registered images. Following image fusion, various image fusion analysis metrics are utilized to assess the quality of the fused image. The edge extraction algorithm is used to extract the edges of the fused image, and comparisons are made between the edges extracted by different fusion methods and those without image fusion.

The Fourier image fusion method used in this chapter involves conducting a direct two-dimensional Fourier transformation on the source image. The phase angle and amplitude post-Fourier transformation are combined according to the fusion rules detailed in algorithm.14. Finally, the inverse Fourier transform is used to restore the fused image. However, due to the characteristics of Fourier transform, the overlay of triangular waves can't perfectly restore the signal when the distribution is uneven. While short-time Fourier transform can address some issues by segmenting the signal, the selection of window size for interception also presents a challenge. Consequently, direct Fourier fusion leads to some ghosting in the texture of the fused result.

Apart from Fourier transform fusion, discrete wavelet transform image fusion is also examined. Wavelet and Fourier sinusoidal waves differ in that wavelet oscillations are concentrated around a single point. By utilizing scaling and translation operations, multi-scale fine analysis of functions or signals can be performed, thereby resolving many complex issues that Fourier Transform is unable to tackle. In this dissertation, DB and FK wavelets of different orders are employed as wavelet bases to transform the image, with image fusion being achieved through the fusion of coefficients post-wavelet transformation.

Information Entropy, Standard Deviation, Spatial Frequency, Average Gradient, Feature Mutual Information, Sum of the Correlation of Differences, and Edge-Based Structural Similarity are used as quality criteria for fused images. Based on these metrics, the results of FFT image fusion are superior. However, due to the nature of FFT, a large amount of the data is redundant information, and the fusion results also visually exhibit texture shadowing. Therefore, when considering the quality of the fused image, image fusion using the FK4 wavelet yields the best results.

In the edge extraction experiment for fusion results, both FFT and DWT image fusion methods provide better results than using single-camera images directly. In edge extraction, the texture ghosting produced by FFT has an unexpected positive effect. The texture becomes blurred except for the steel part, resulting in fewer unwanted edges being identified.

Chapter 6

Pattern Recognition

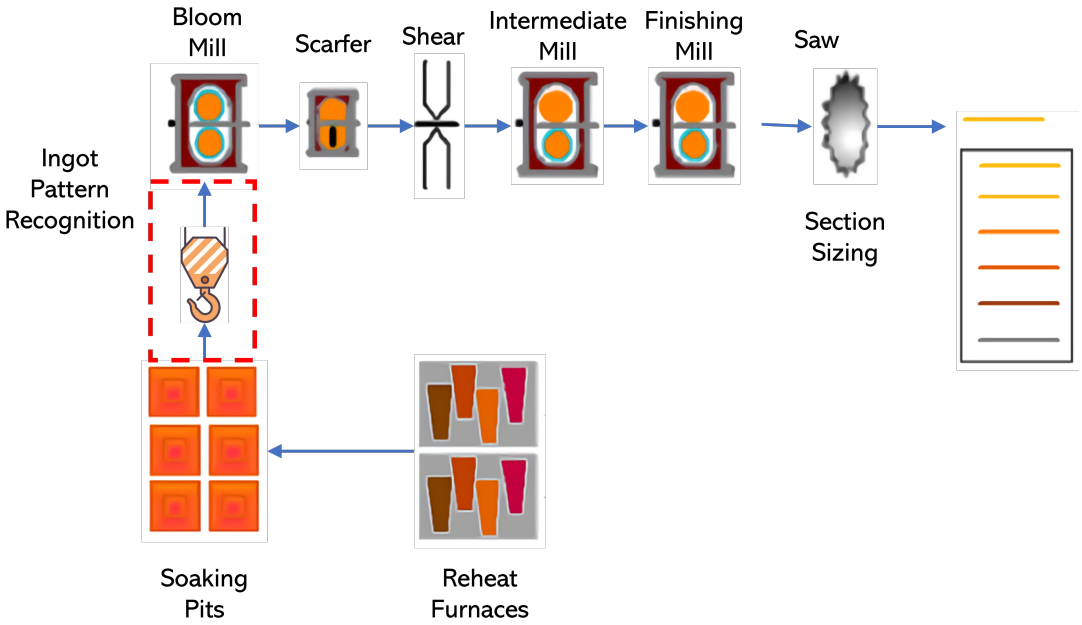


Figure 6.1: Steel Rolling Workflow

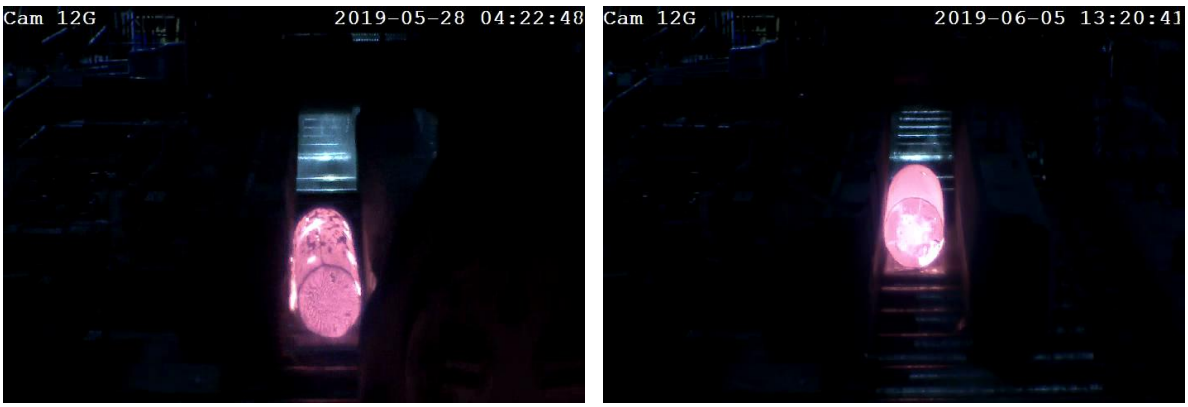
As shown in Figure 6.1, the steel ingots are heated at the beginning of the steel production process, and the ingots are immersed in the soaking pit. The heated ingots are then clipped out of the soaking pit by a crane and placed on the rolling line. The entire process of ingots transport is manual, so the direction in which the ingot is placed on the production line is uncertain. The composition of the bottom and top of the ingot varies due to the internal precipitation of the ingot during shaping. Different ingot compositions can lead to changes in steel performance indicators. To avoid affecting the final steel qualification rate, steel mills usually remove parts with more impurities. If the ingots are placed in the wrong direction on the production line, the steel plant will remove the wrong part and eventually scrap the entire steel.

Figure 6.2 shows the process of placing an ingot on a rolling line with a grab arm taken by a surveillance camera. It can be seen that the location of the camera installation is facing

(a) (b) (c)

Figure 6.2: Placing a Ingot onto the Mill

the conveyor belt where the ingot is placed, which makes the experiment directly use the video data taken by the camera as the data source for identification.



(a) (b)

Figure 6.3: Ingots On Mill: (a) Top Side to Camera; (b) Bottom Side to Camera.

The top and bottom of the ingot are different in appearance. Figure 6.3(a) shows an ingot with the top facing the camera, and Figure 6.3(b) is an ingot with the bottom facing the camera. It can be seen from the figure that the oxide on the top surface of the ingot has a particular pattern, which is called the "brain pattern" by the factory. On the other hand, the bottom surface of the ingot looks smoother without special oxidation marks. Therefore, due to the characteristics of the ingot, the direction of the ingot on the conveyor belt can be distinguished by using the camera to take pictures and identify this special oxidation pattern. With the aid of the classification algorithm, factory workers can get the information of ingot placement direction at the first time and adjust the ingot direction to prevent steel loss.

6.1 Ingot Detection

The ingot should be extracted from the backgound in the image to classify the pattern of the ingot ends. First, the image is clipped. When the ingot is placed on the conveyor belt, the ingot is moved to the centre of the picture. Therefore, as shown in Figure 6.4, the position of the red

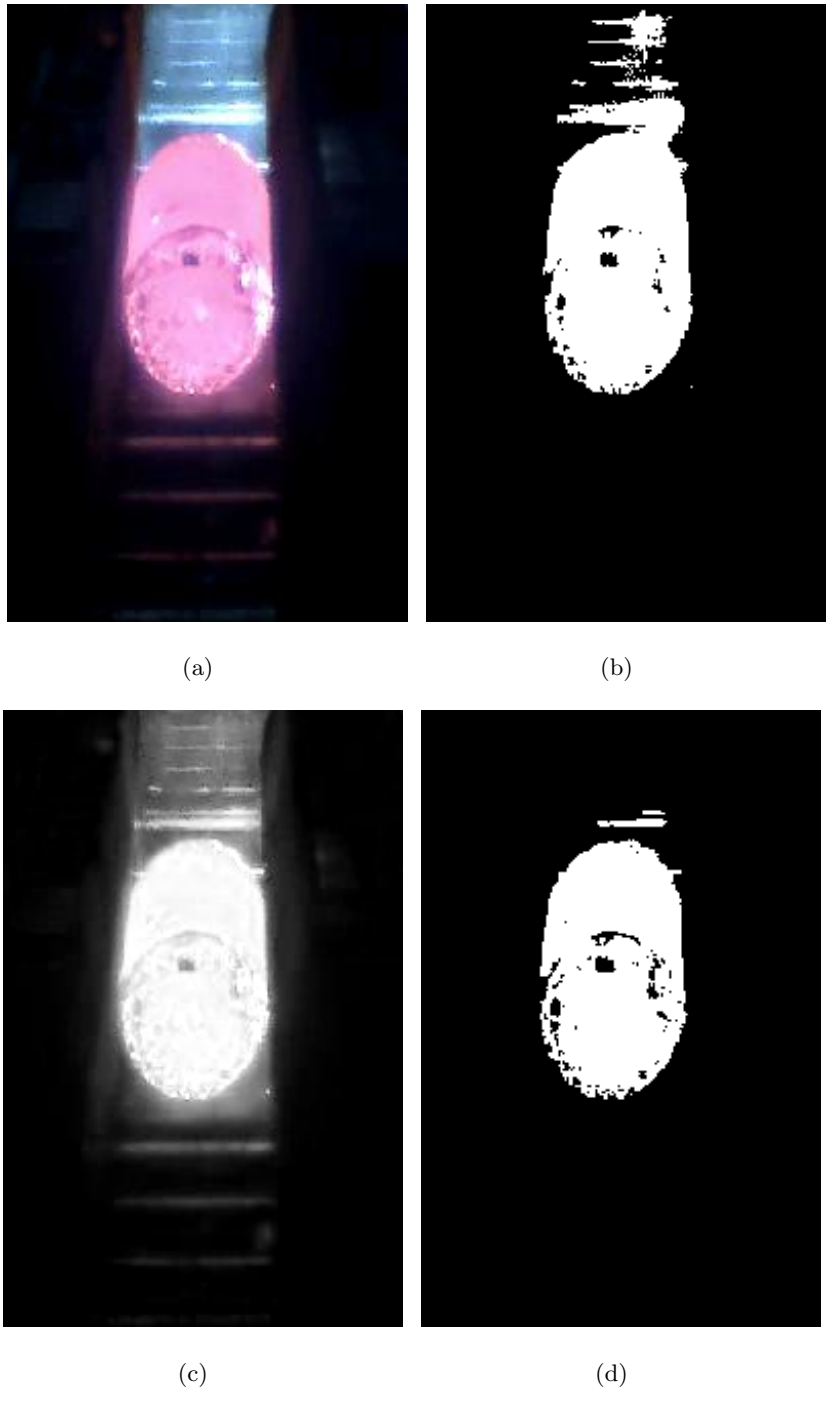
box in the image is preserved as the region of interest. There are two main advantages of setting the region of interest in this way, one is to reduce the computational load of the algorithm, and the other is to eliminate the cases where some ingot is not placed in the detection area (as shown in Figure 6.5). The ingot appears in the picture when the boom grabs it and not yet placed on the conveyor belt. Therefore, the end of the ingot is not visible at this time, and the ingot should not be identified.

Figure 6.4: Region of Interest

Figure 6.5: Ingot not Placed on Conveyor yet

Because there is a big difference in brightness between a high-temperature ingot and its surroundings, extracting the ingot part in the image is practical by directly binarizing it.

However, direct light illumination on the conveyor belt in steel rolling equipment results in a high gloss area on the conveyor belt, which interferes with the selection of ingots to some extent. However, the white light and the orange-red light emitted by the high-temperature ingot are different, so only binarizing the red channel of the RGB image can avoid some interference.



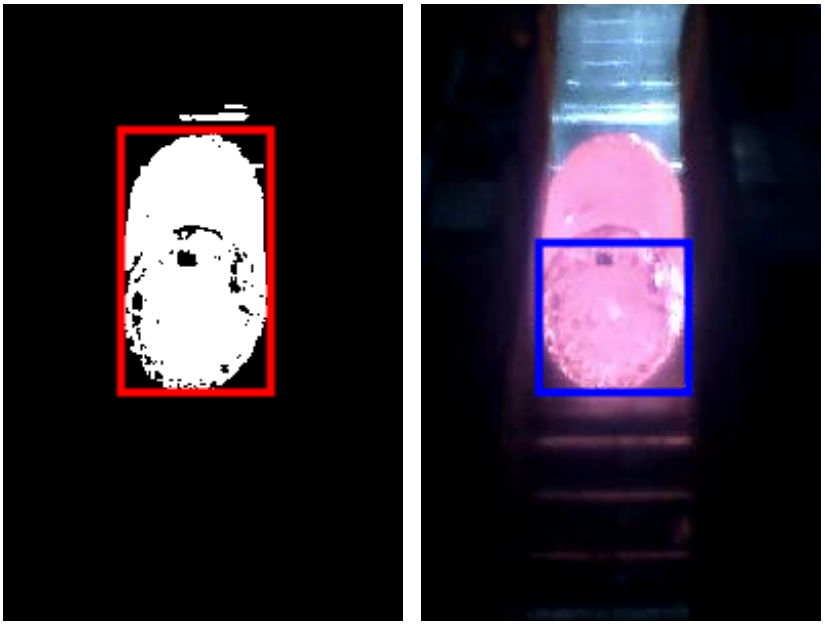
(a) (b)

(c) (d)

Figure 6.6: Binarization of Cropped Image: (a) Cropped Image; (b) Binarization of RGB Channel; (c) Cropped Image Red Channel; (d) Binarization of Red Channel.

Figure 6.6shows the result of the binarization of the clipped image. Figure 6.12(a) is the image after clipping. Figure 6.12(b) is the result of the binarization of the whole RGB channel. Figure 6.12(c) is the visualization of the red channel of the clipped image, and Figure 6.12(d)

is the result of the binarization of the red channel. From Figure 6.12(b), it can be seen that the ingot and the highlighted part of the conveyor belt are connected. Even if the threshold of binarization is increased, the ingot part will be missing, resulting in a significant error in the extraction of the ingot area. In Figure 6.12(d), only the red channel is binarized to preserve the region of the ingot to the maximum extent. Although the highlighted part of the conveyor belt may be identified, it is not connected to the ingot part and can be excluded in subsequent screening.



(a) (b)

Figure 6.7: Ingot End Extraction: (a) Ingot Extraction; (b) End Extraction.

After image binarization, 8-pixel connectivity is detected for image elements. Find the connected component in the horizontal, vertical, and two diagonal adjacent pixels to find the completed objects in the image. The object with the largest area is reserved as an alternative, and the alternative region is triple-filtered. The region is restricted by area, rotation, and position, respectively, and is identified as an ingot region through these three restrictions. Figure 6.7(a) shows the ingot region detected. As Figure 6.7(b) shown, once the ingot region is confirmed, the bottom square area of the ingot is intercepted as a picture of the ingot end.

Algorithm 15 presents the process of detecting the ingot ends. The algorithm operates on a video sequence, iterating through each frame until the end is reached.

In the first step, the red channel of the Region of Interest (ROI) is binarized to obtain a binary image. This binarization process distinguishes the ingot end patterns from the background.

Next, the connectivity of pixels in the binarized image is evaluated to detect the objects present in the region. This step helps identify the distinct ingot end patterns based on their

Algorithm 15: Segment the End side of Ingots

1: **for** i=1:end of video **do**

2: Binarised the ROI red channel

3: Detect the objects in the region by evaluating the connectives of pixels [21]

4: Filter the detected objects by constrains

$Area$: $5000 < Area < 14000$

$Orientation$:$|Orientation| < 15$

$Position$:The centoid of object should in ROI

5: Extract the bottom square area of the ingot region

6: **end for**

connected pixel regions.

After object detection, triple filters are applied to refine the results. These filters impose constraints on the properties of the detected objects, such as the area, orientation, and position. Only objects that satisfy these constraints are retained as potential ingot end patterns.

Finally, the bottom square area of the ingot region is selected, encompassing the detected ingot ends. This extraction step ensures that the relevant portion of the ingot end patterns is isolated for further analysis.

Figure 6.8 illustrates the ingot end images that have been successfully extracted from the video sequence using the described algorithm. These images showcase the identified ingot end patterns, which are essential for subsequent processing and analysis tasks.

Figure 6.8: Extracted Ingot Ends

In most cases, the algorithm can capture the complete ends images. However, even if the image's green and blue channels are filtered, the light on the conveyor belt will still be connected to the ingot area. Moreover, because the monitoring camera is prone to overexposure, some areas in the image have extremely high brightness, leading to significant errors in ingot position recognition.

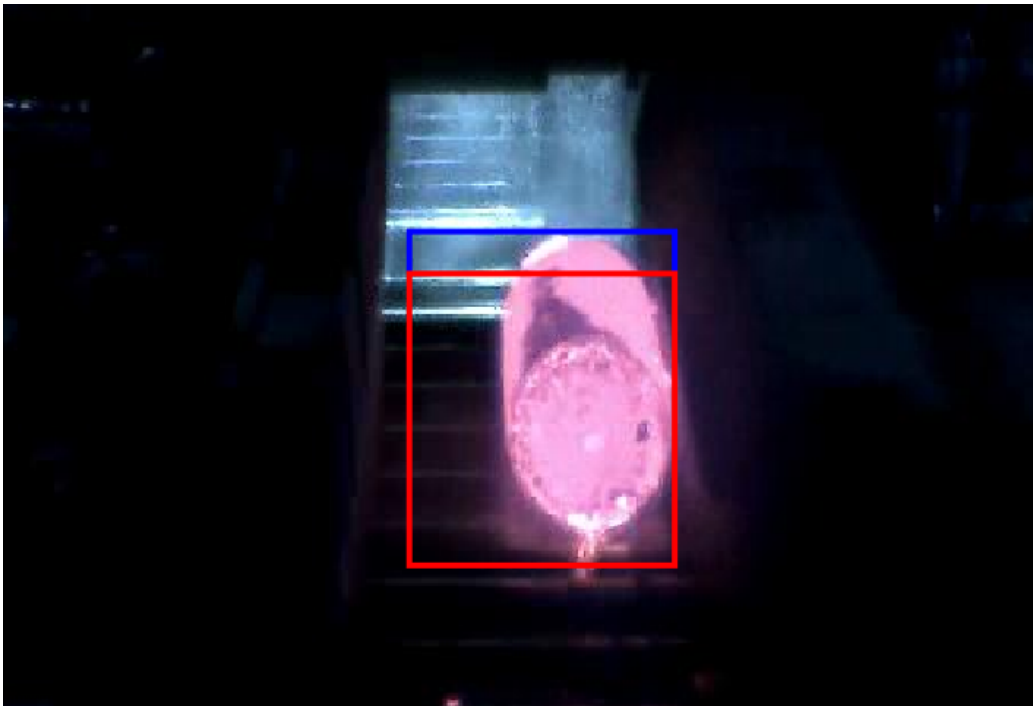


Figure 6.9: Failure of Ingot Extraction

As shown in Figure 6.9, because the illumination of the rolled steel position is connected to the front of the ingot, the algorithm of pixel connectivity detection considers the white light on the conveyor belt as the ingot region. This makes the ingot region very large and the cropped square end area position is incorrect. In this case, Hough circle detection is used to further extract the round end of the ingot in the image.

Algorithm.16 shows the Hough circle detection algorithm. Each edge point is extended in its gradient direction through edge recognition and gradient detection of the image to find the overlapping circle centre. After confirming the coordinate of the circle centre, confirm the radius of the circle reversely. Finally, a circular mask is obtained to extract the circular ends ingots. Figures.6.10 and 6.11 show the ends intercepted by the algorithm.

6.2 Pattern Recognition

After the algorithm can automatically extract the image of the end of the steel ingot, what needs to be done is to identify the captured end and judge whether there are representative 'brain patterns' in the image.

The ingot pattern recognition task can be efficiently solved by comparing ends images. The closeness between images is determined by comparing the cropped end images (top side and bot side to camera) with appropriate similarity measures. If two images have a high degree of similarity, they are likely to be both the top side or bottom side. On the contrary, if the similarity between the two images is very low, the two images may come from ingots with

Algorithm 16: Hough Circle Detection Method

Input: Cropped Ingot end side I_{end}

Output: Circle Centres C,radius R and circular cropped image I_{circle}.

Detect the edge feature points E_k and there gradient direction θ_k

Initialize the centre voting space V_c.

for $m = 1, \cdots, k$ **do**

    Extend forward along the gradient direction θ_m of the point E_m

    Every time meet a point :$V_c(i, j) += 1$

end for

Sort the voting space $V_c(i, j)$ to get circle centre C

Initialize Circle radius Voter V_r

for $n = 1, \cdots, k$ **do**

    $r_n = |CE_n|$

    $V_r(r) += 1$

end for

Sort V_r to find the radius R

Use C and R to crop image and get I_{circle}

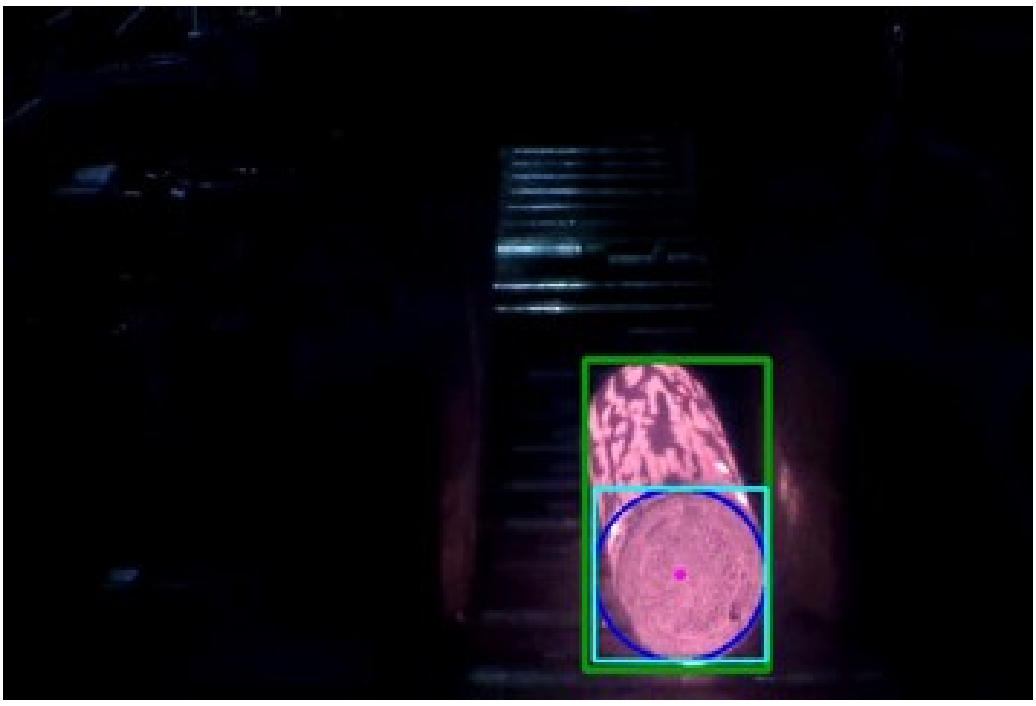


Figure 6.10: Circular Ingot End Extraction

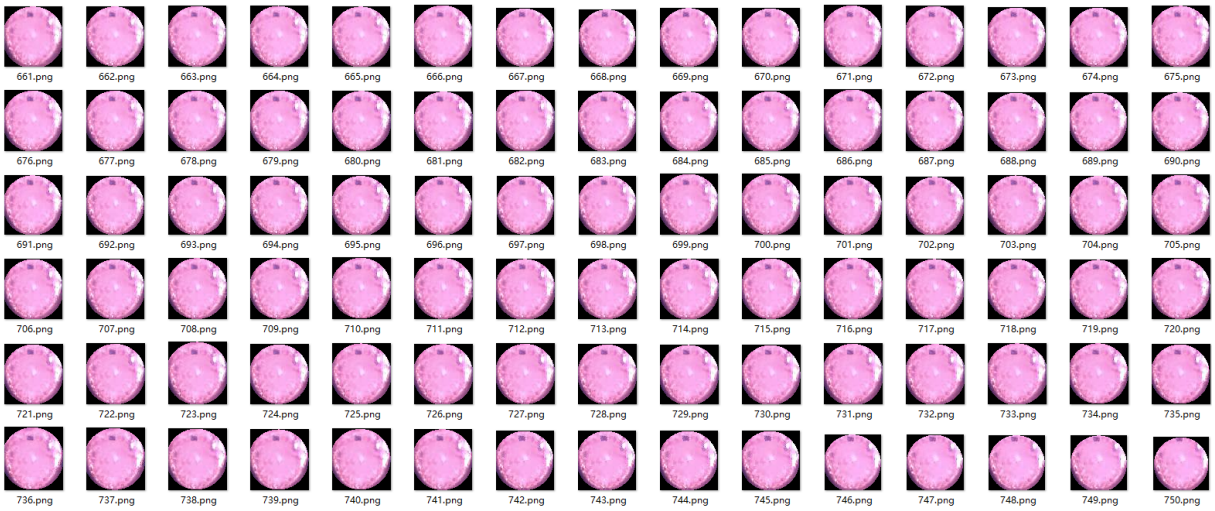


Figure 6.11: Extracted Circular Ingot Ends

different directions. Our experience in solving related tasks such as target tracking shows that one of the best similarity measures index, is the so-called structural similarity measure [38, 62].

There are many image similarity measures, the most commonly ones of which are the Minimum Mean Square Error, Mean Absolute Error or Peak-Signal to Noise Ratio. Most of these measures do not have good invariance and cannot capture the perceived similarity of images when brightness, contrast, compression, or noise change [62]. Hence, in our work we adopt a measure which in its principle is close to the way the human visual system functions.

6.2.1 The Structural Similarity Measure and Its Advantages in Pattern Recognition

Based on the premise that the human visual system view the world, a image metric has been proposed in [62], called the Structural Similarity Measure (SSIM) Index. The SSIM index, S, between two images, $\boldsymbol{a}$ and $\boldsymbol{b}$ is defined as follows:

$$S(\boldsymbol{a}, \boldsymbol{b}) = \left(\frac{2\mu_a \mu_b}{\mu_a^2 + \mu_b^2} \right)^{\alpha} \left(\frac{2\sigma_a \sigma_b}{\sigma_a^2 + \sigma_b^2} \right)^{\beta} \left(\frac{\sigma_{ab}}{\sigma_a \sigma_b} \right)^{\gamma}, \tag{6.1}$$

where μ, σ stand for the sample mean and sample standard deviation, respectively, and σ_{ab} corresponds to the sample covariance between an intensity value from image $\boldsymbol{a}$ and intensity value from image $\boldsymbol{b}$. The three components of S, reading from the left, measure how close the luminance, contrast and structural similarity of the two images are. The product in (6.1) can be seen as a fusion of three independent fusion cues. The assumption of the independence is based on the fact that a moderate variation in the luminance or contrast is not affecting the structures of the image objects. During the manufacturing process there are many disturbances, including illumination changes. However, the SSIM is less sensitive to illumination changes compared with other similarity measures.

The exponents $\alpha, \beta, \gamma \geq 0,\ \alpha+\beta+\gamma > 0$ are used to adjust the impact of each measurement on the final value of S. The reader is referred to [38, 62] for full details.

The measure defined in (6.1) is symmetric and has a unique upper bound: $S(\boldsymbol{a}, \boldsymbol{b}) \leq 1,\ S(\boldsymbol{a}, \boldsymbol{b}) = 1$ iff $\boldsymbol{a} = \boldsymbol{b}$.

6.2.2 Image Dissimilarity

Let's consider the similarity between two grayscale images, represented here as vectors formed from the image regions. One of the ways to convert the similarity $S(\boldsymbol{a}, \boldsymbol{b})$ into normalised dissimilarity $D(\boldsymbol{a}, \boldsymbol{b})$ is as follows [63]:

$$D(\boldsymbol{a}, \boldsymbol{b}) = \frac{c_0 - S(\boldsymbol{a}, \boldsymbol{b})}{c_1},$$

where c_0 and c_1 are chosen to map a distance into the interval $[0, 1]$. An alternative way [63],

$$D(\boldsymbol{a}, \boldsymbol{b}) = \frac{c_0}{S(\boldsymbol{a}, \boldsymbol{b})} - 1. \tag{6.2}$$

is preferred however, as it only requires knowledge of maximal value of S and is more sensitive to very dissimilar vectors. The dissimilarity between images used in the method proposed is obtained by substituting (6.1) into (6.2) (however often $c_0 = 1$):

$$D(\boldsymbol{a}, \boldsymbol{b}) = \left(\frac{2\mu_a\mu_b}{\mu_a^2 + \mu_b^2}\right)^{-\alpha} \left(\frac{2\sigma_a\sigma_b}{\sigma_a^2 + \sigma_b^2}\right)^{-\beta} \left(\frac{\sigma_{ab}}{\sigma_a\sigma_b}\right)^{-\gamma} - 1. \tag{6.3}$$

The next section presents the classification results obtained from the real data with the steel ingots.

6.3 Results from the First Ingot Data Set

The first group of video data sets provided contains four videos named '40 ingoing bottom-1', '40 ingoing bottom-2', '40 ingoing top-1' and '40 ingoing top-2'. Two video data show bottom-side ingots towards the camera, and the other two are top-side ingots towards the camera. The images of the ends of the ingots in the four video data are extracted.

One of the end image has been chosen to be the reference. All of the other end images are compared to that image with the SSIM index, which is a index to evaluate the similarity between two images. The end images which have higher similarity can be seen as the same side with the reference image.The end images which have low SSIM indexes will be recognised as the opposite side of reference.

As shown in Figure 6.12, the SSIM indexes of top end videos and bottom end videos are visualized. The top side SSIM indexes is relatively high compared to the bottom side. The drop of SSIM index for the top side in the last part is due to the color change when the ingot was reheated.

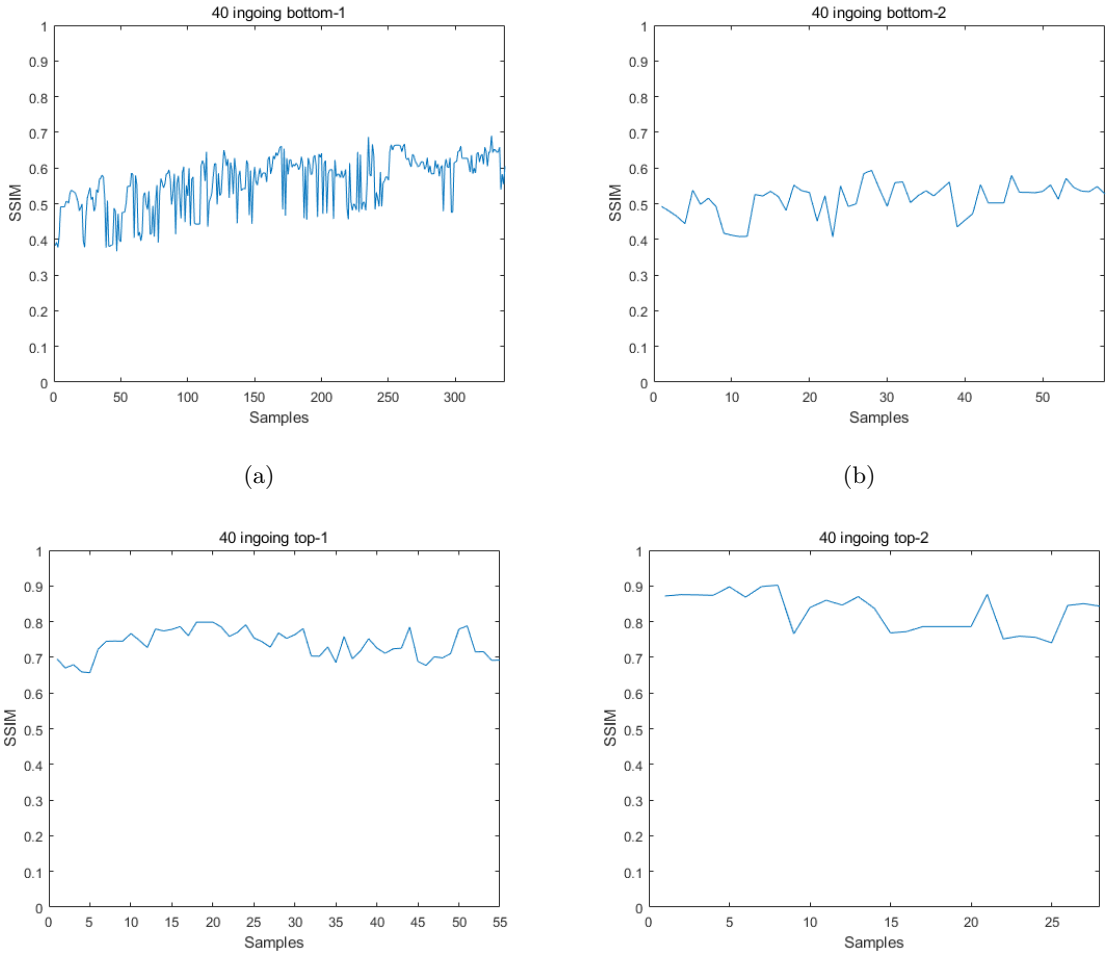


Figure 6.12: SSIM results: (a) 40 ingoing bottom-1; (b) 40 ingoing bottom-2; (c) 40 ingoing top-1; (d) 40 ingoing top-2.

Table 6.1: SSIM Index

SSIM	Max	Min	Mean
40 ingoing bottom-1	0.6905	0.3667	0.5562
40 ingoing bottom-2	0.5934	0.4036	0.4976
40 ingoing top-1	0.7985	0.5517	0.7139
40 ingoing top-2	0.9021	0.6802	0.7901

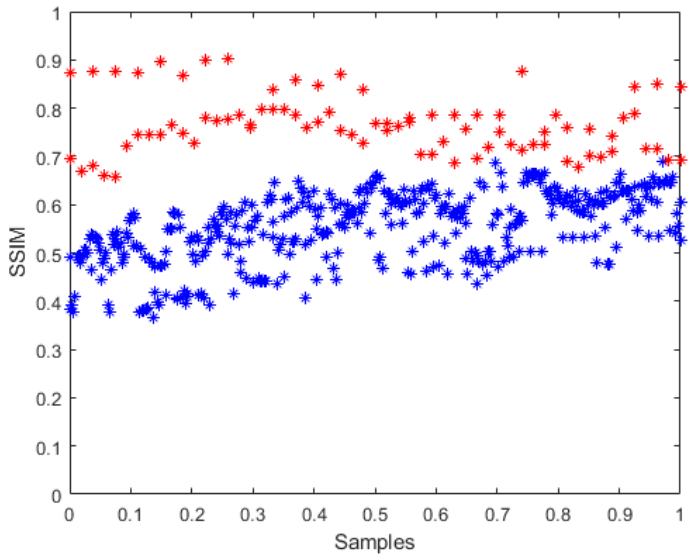


Figure 6.13: SSIM Index of Top and Bottom Ends

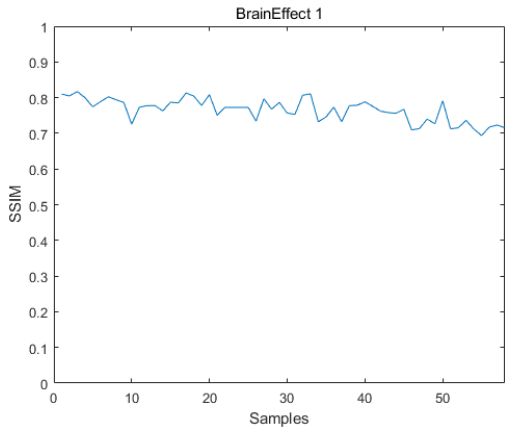
Figure 6.13 shows the SSIM index of the ingot ends captured from four videos. The blue data points in the figure are from the top ends, and the red data points are from the bottom ends. It can be seen that although the two data are slightly coincident, it is theoretically possible to identify the top by setting a threshold. After obtaining 12 video data for the second round, the same method was applied and tested to verify the reliability and feasibility of the algorithm.

6.4 Results from the Second Ingot Data Set

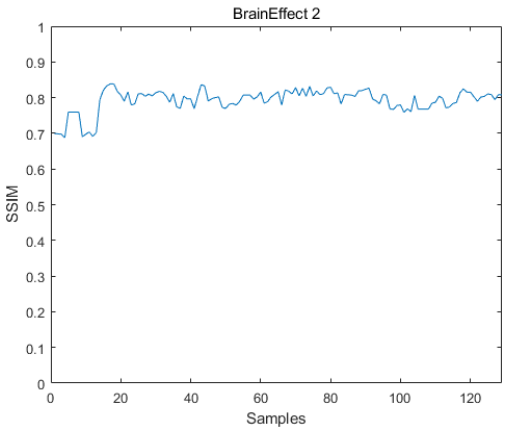
Besides the first four video data, a second ingot data set is provided. There are 12 videos in the second ingot data set, 6 for the 'top ends to the camera' with the brain patterns and 6 for the 'bottom ends to the camera' with smooth ends. The names of data are 'BrainEffect-$1, \ldots, 6$' and 'Smooth-$1, \ldots, 6$' respectively. Compared to the first ingot data set, some of the data are hard to be classified due to the low resolution and the overexposed effect.

Figure 6.19 shows the results of SSIM values obtained from 6 groups of data with the top of the ingot facing the camera. The similarity of most data is higher than 0.7 and maintained at about 0.8. Figure 6.20 shows six groups of data from the bottom of the ingot towards the camera. The SSIM values of most of the data are slightly higher than 0.7 and lower than 0.7. It can be seen that the SSIM value of the image at the end of the ingot can distinguish the direction of the ingot to a certain extent.

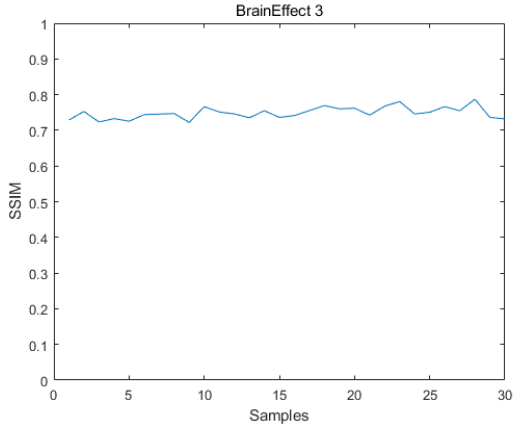
Figure 6.16 shows the integrated SSIM values. The red data points are all from $BrainEffect\ X$. The blue data points are from $Smooth\ X$. The two kinds of data can be classified to a certain extent by setting a threshold value of 0.75. Figure 6.17 shows the confusion matrix of the results produced by the previous algorithm. It can be seen that when 0.75 is used as the threshold, the



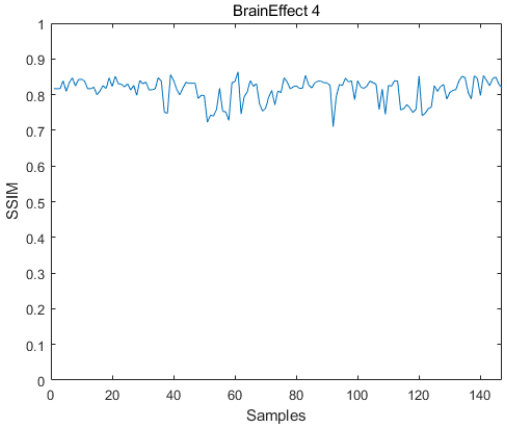
(a)



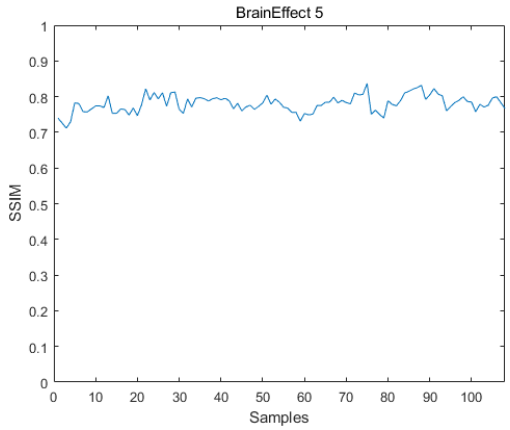
(b)



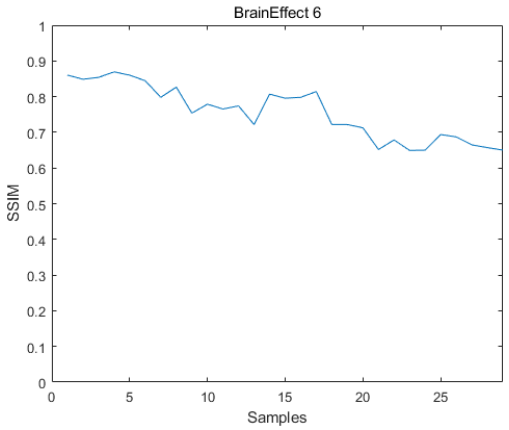
(c)



(d)

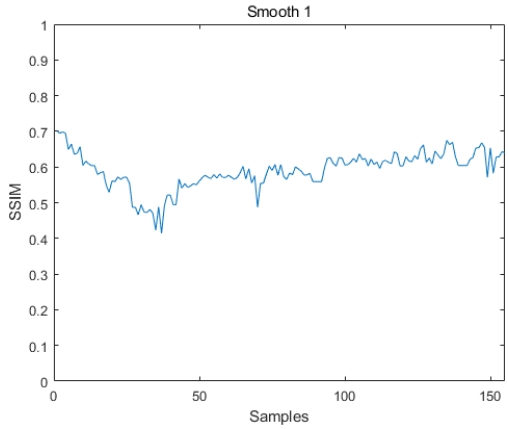


(e)

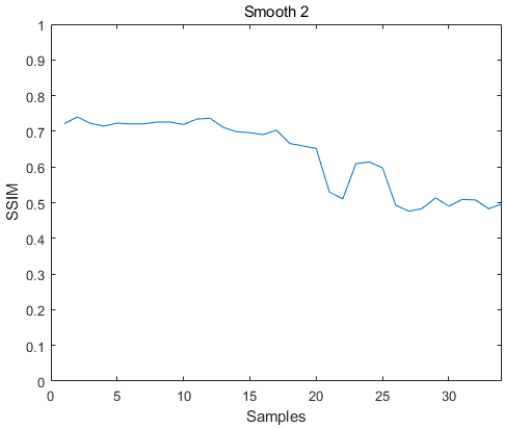


(f)

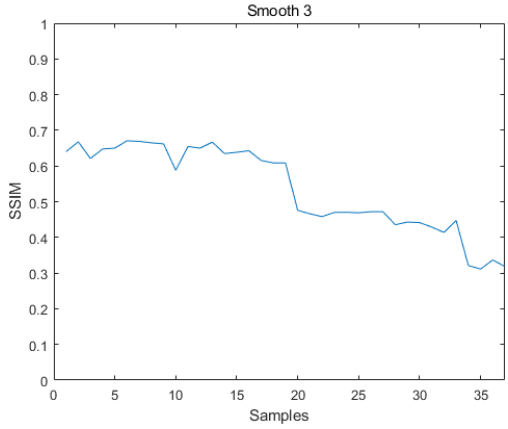
Figure 6.14: SSIM results: (a) BrainEffect 1; (b) BrainEffect 2; (c) BrainEffect 3; (d) BrainEffect 4; (e) BrainEffect 5;(f) BrainEffect 6.



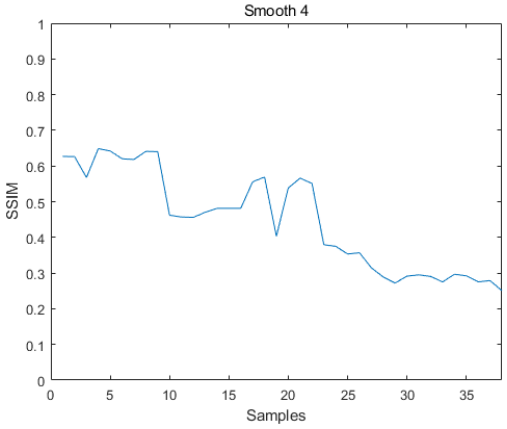
(a)



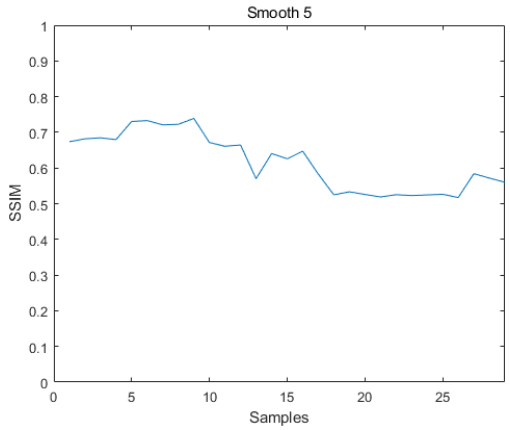
(b)



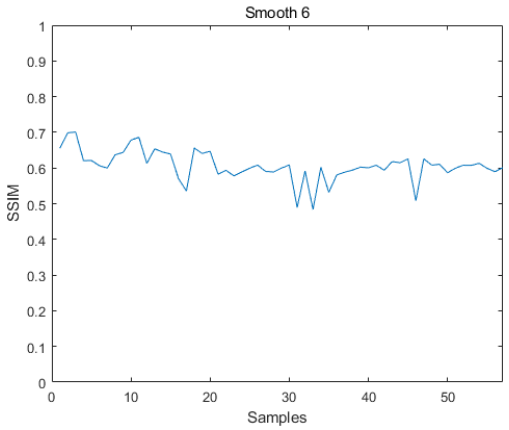
(c)



(d)



(e)



(f)

Figure 6.15: SSIM results: (a) Smooth 1; (b) Smooth 2; (c) Smooth 3; (d) Smooth 4; (e) Smooth 5; (f) Smooth 6.

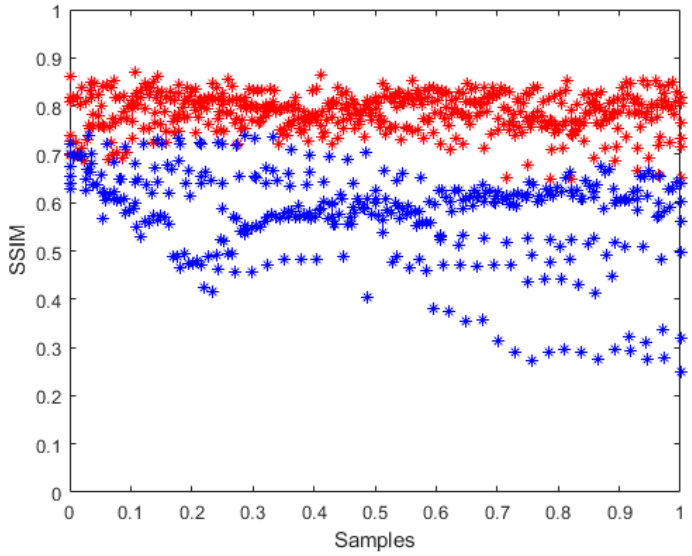


Figure 6.16: SSID Indexs of the Second Data Set

top ends precision is 1, the recall is 0.8134, and the accuracy is 0.9180. Although the recall rate is not high, the top of the ingot is successfully identified in eight sets of image sequences with the top facing the camera. This is due to a misjudgement that the brightness changes when the temperature of the ingot rises near the rolling area.

Precision, Recall and Accuracy are used to assess the performance of the developed approach. These are defined as follows [49]:

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}, \tag{6.4}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}, \tag{6.5}$$

$$Accuracy = \frac{True\ Positive + True\ Negative}{All}. \tag{6.6}$$

The recall can be increased by lowering the threshold, but the relative precision will be lower. In order to further improve the performance of classification, dual SSIM values are used for recognition in the next Section 6.5. Table 6.2 shows the performance of the classification algorithm.

Table 6.2: Classification Evaluation

	Precision	Recall	Accuracy
Top Ends	1	0.8134	0.9180
Bottom Ends	0.8724	1	

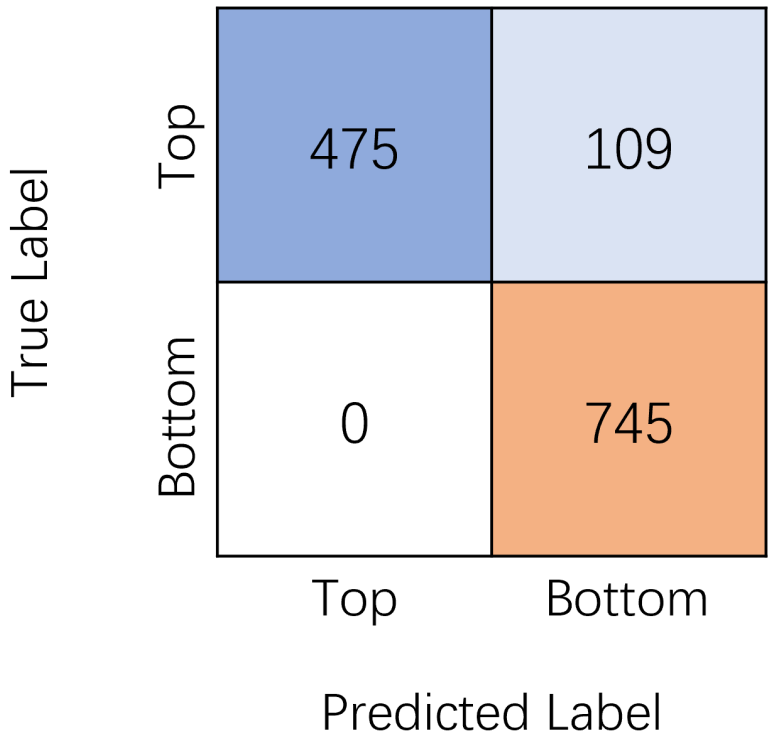
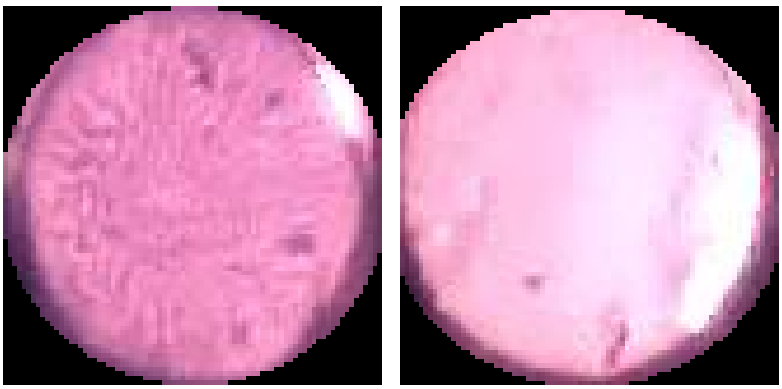


Figure 6.17: Confusion Matrix of SSIM Classification

6.5 Dual SSIM Classifier

The previous approach was to use only the SSIM values generated by referencing a TOP image and use them as classifiers based on thresholds. Such a classifier can effectively identify the direction of the ingot in most cases, but the SSIM value will be affected in some cases with different brightness. So double SSIM value recognition compares the truncated endpoint map with the top and bottom references at the same time to calculate two SSIM values. The direction of the ingot is distinguished by comparing the SSIM values.



(a) (b)

Figure 6.18: Reference Images: (a)Top End;(b)Bottom End

Figure 6.18 shows the reference images of top end and botoom end. Through the above

section extraction technology, a total of 1329 end-section images of ingots are extracted from 8 bottom videos and 8 top videos.

Algorithm 17: Dual SSIM Direction Recognition

Input: $I_{top}, I_{bot}, I_{section}$

Output: The direction of section $I_{section}$

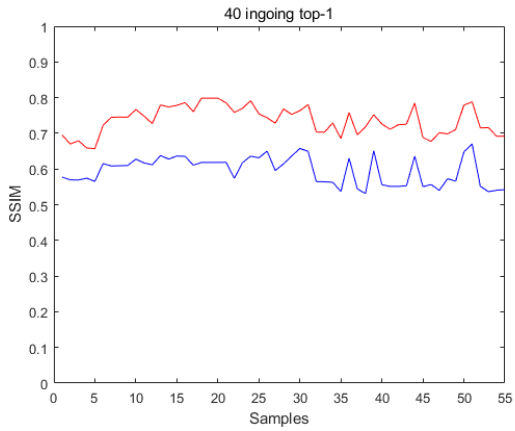
1: Calculate the SSIMs between $I_{section}$ and I_{top}, I_{bot}
2: $SSIM_{top} = \boldsymbol{SSIM}(I_{top}, I_{section})$
3: $SSIM_{bot} = \boldsymbol{SSIM}(I_{bot}, I_{section})$
4: **if** $SSIM_{top} >= SSIM_{bot}$ **then**
5: The direction of $I_{section}$ is TOP
6: **else**
7: The direction of $I_{section}$ is BOT
8: **end if**

Figures.6.19 and 6.20 show the classification results of 16 video data. The red data in the figures are the SSIM index calculated by comparing with the top reference, and the blue data is the SSIM index calculated with the bottom reference. As shown in the figures, almost all video data images with the top ends of ingots facing the camera have red lines above the blue lines, which means $SSIM_{top}$ are larger than $SSIM_{bot}$. Moreover, the blue line exceeds the red for all data with the bottom ends of the ingots facing the camera.

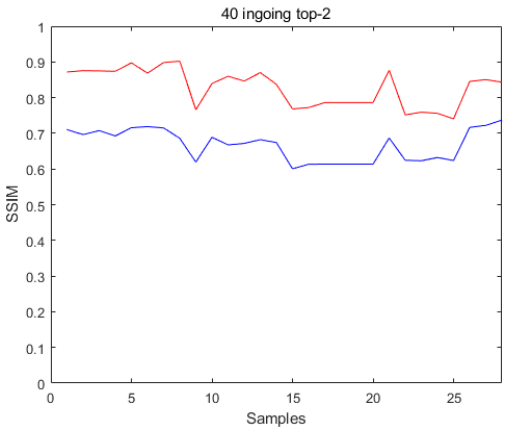
Table 6.3: Dual SSIM Classification Evaluation

	Precision	Recall	Accuracy
Top Ends	1	0.9589	0.9819
Bottom Ends	0.9688	1	

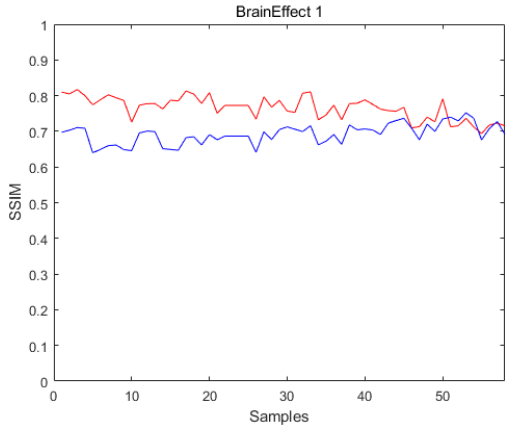
Figures.6.21 and Table 6.5 are confusion matrices and evaluations for the classification results of the dual SSIM index classifier, respectively. Compared with a single SSIM index classifier, the classifier's performance is greatly improved. For the top ends data, the precision is 1, the recall rate is 0.9589, and the overall classification accuracy is 0.9819. 24 frames are misclassified, 5 came from 'BrainEffect 1', 8 from 'BrainEffect 5', and the remaining 11 came from 'BrainEffect 6'. The misclassified images identified from 'BrainEffect 1' and 'BrainEffect 6' were all filmed when the ingot was near the rolling area. In that area, the ingot temperature increases and the brightness changes, which interferes with identifying the SSIM index. The factory requires to alert the situation when top ends ingots facing to the camera, so even though there were misclassifications in the area before the rolling, the top situation was already indicated by the classifier before that time and will not result in a misdetection. In 'BrainEffect 5', the misclassification occurred just after the ingot end was detected. As the ingot moves further, the


40 ingoing top-1
SSIM
Samples

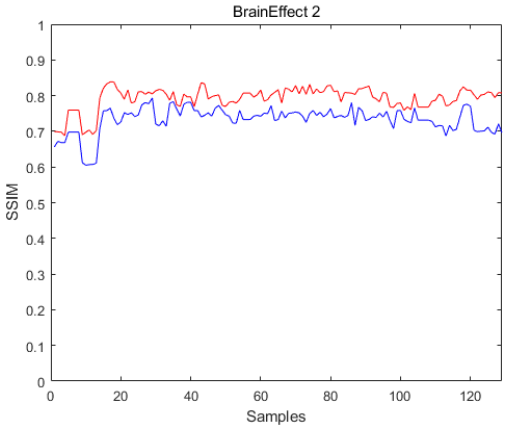
(a)


40 ingoing top-2
SSIM
Samples

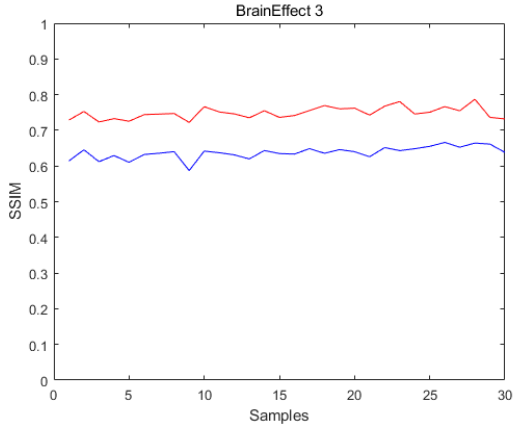
(b)


BrainEffect 1
SSIM
Samples

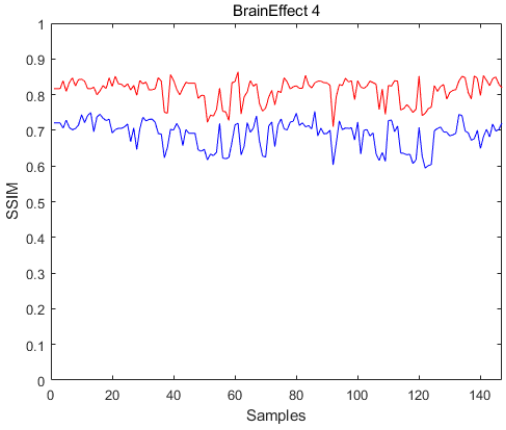
(c)


BrainEffect 2
SSIM
Samples

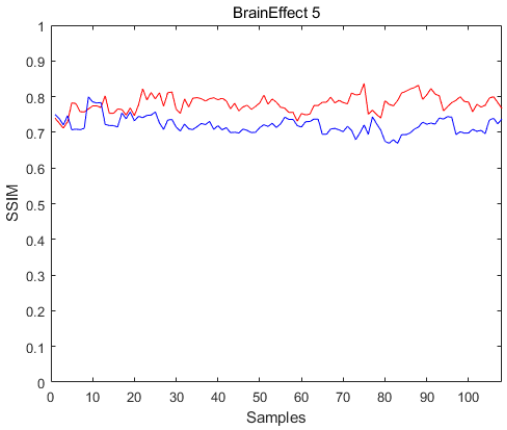
(d)


BrainEffect 3
SSIM
Samples

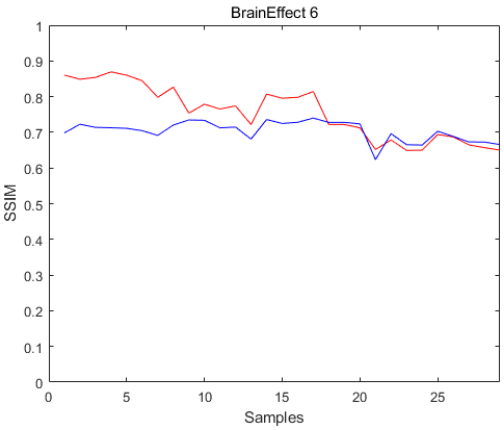
(e)


BrainEffect 4
SSIM
Samples

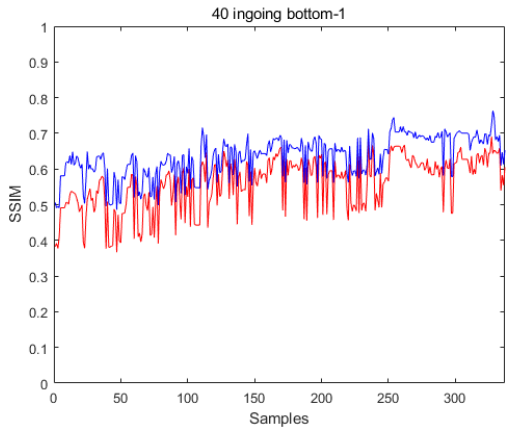
(f)


BrainEffect 5
SSIM
Samples

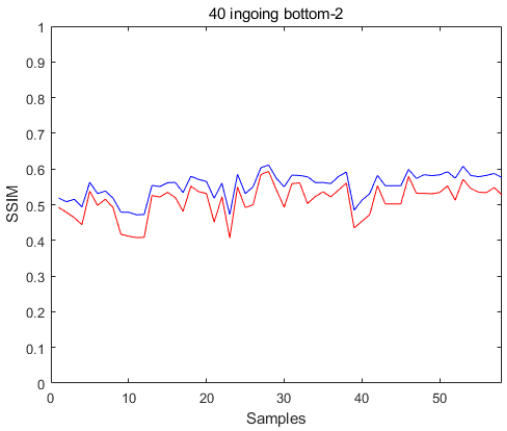
(g)


BrainEffect 6
SSIM
Samples

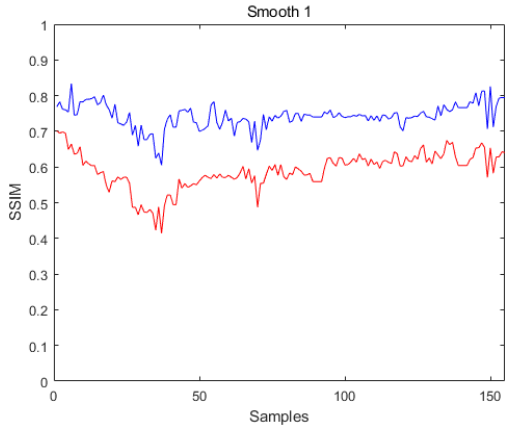
(h)


40 ingoing bottom-1
SSIM
Samples

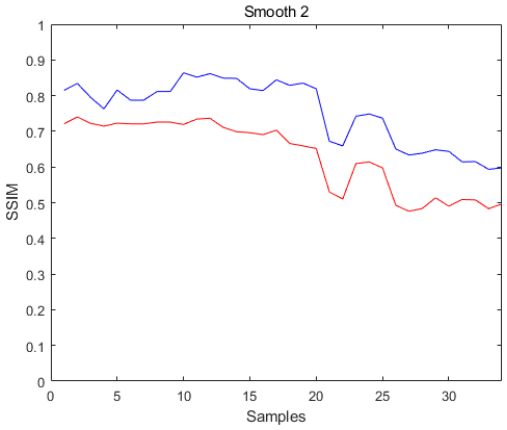
(a)


40 ingoing bottom-2
SSIM
Samples

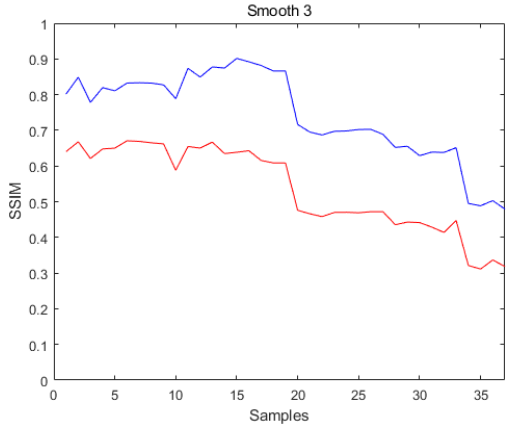
(b)


Smooth 1
SSIM
Samples

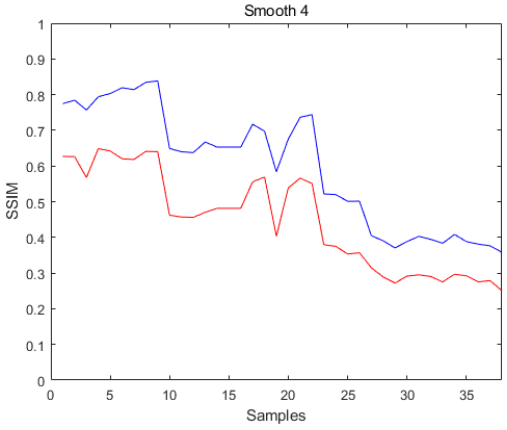
(c)


Smooth 2
SSIM
Samples

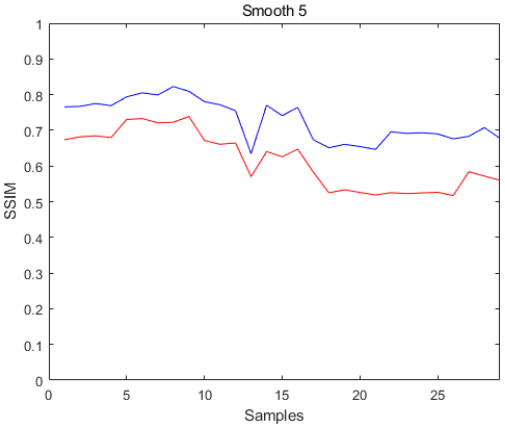
(d)


Smooth 3
SSIM
Samples

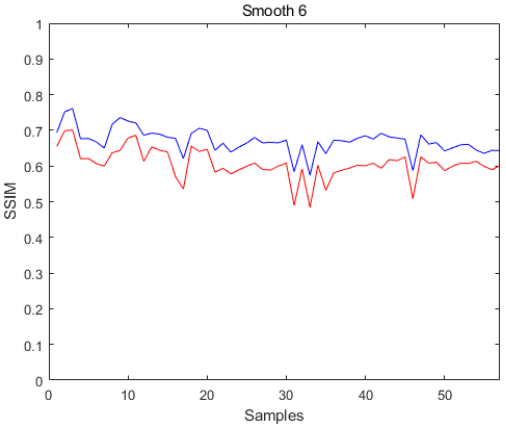
(e)


Smooth 4
SSIM
Samples

(f)


Smooth 5
SSIM
Samples

(g)


Smooth 6
SSIM
Samples

(h)

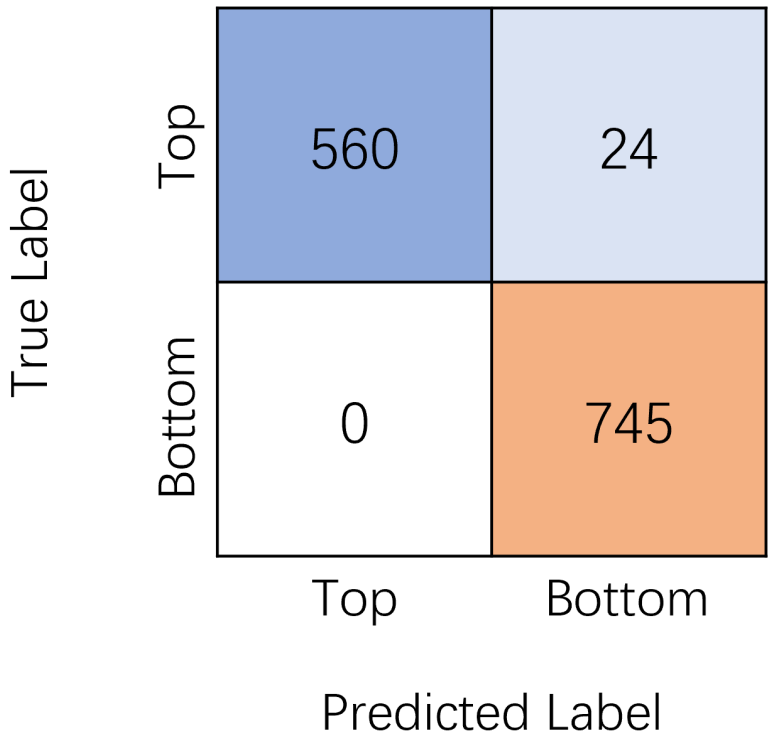


Figure 6.21: Confusion Matrix of Dual SSIM Classifier

classifier is able to detect the top ends in time and issue an alert.

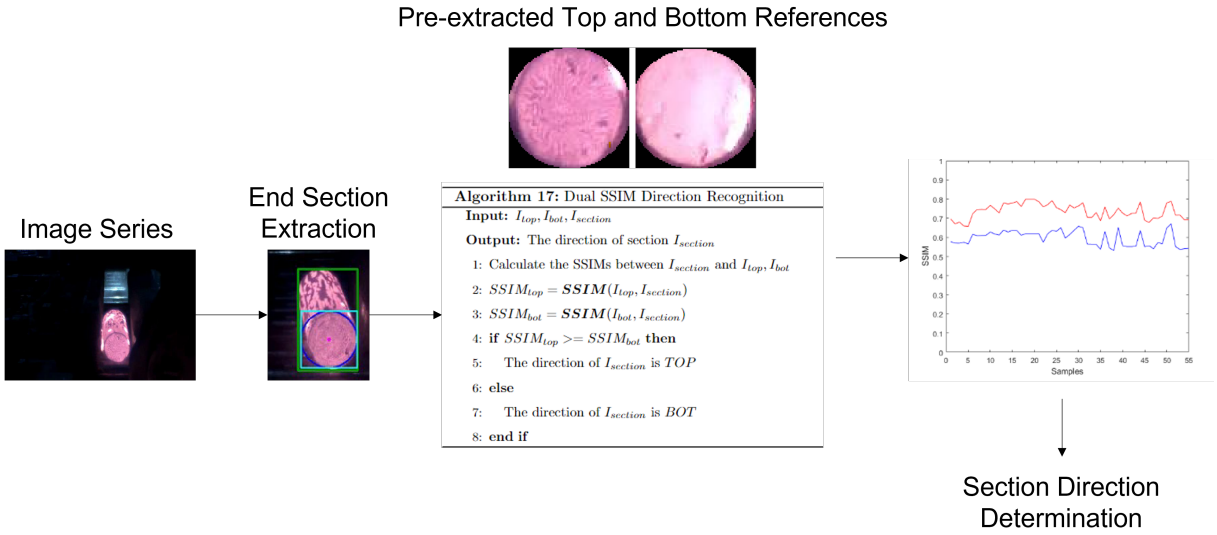


Figure 6.22: Workflow of Pattern Recognition

Figure 6.22 depicts the workflow of the pattern recognition algorithm used in this study. The algorithm follows a step-by-step process to determine the direction of the ingot's end section.

The first step involves extracting the end section of the ingot from the series of images. This extraction process isolates the specific region of interest containing the end section for further analysis.

Next, the extracted end section is compared with the reference images of the top and

bottom sections using the structural similarity index ($SSIM$). The $SSIM$ index measures the similarity between two images based on their structural patterns and pixel intensities. By calculating the $SSIM$ index for both the top and bottom references, the algorithm determines which direction exhibits a higher $SSIM$ index.

Finally, based on the comparison results, the direction with the higher $SSIM$ index is determined as the direction of the ingot's section. This information can be used to alert the staff and ensure the correct orientation of the ingot during production processes.

Overall, the pattern recognition algorithm employs image analysis techniques and the $SSIM$ index to accurately determine the direction of the ingot's end section, contributing to quality control and production efficiency in the steel plant.

6.6 Summary

This chapter introduces a classification case study in an actual steel plant. The factory needs to determine the direction of the ingots placed on the production line, and if the top of the ingot (the one with the 'Brain' pattern) is facing the camera, it needs to alert the staff to change the ingot direction in time. Incorrect ingot direction can affect the performance of the finished steel and may lead to scrapping the steel.

An algorithm for automatically detecting and intercepting the round surface of the ingot end is provided. Firstly, the high-brightness part of the image is obtained by binarizing the image, and then the blocks of objects are obtained by checking the pixel connectivity. Next, the ingot region is found by checking the area, the orientation, and the position among the blocks. Finally, the Hough circle detection algorithm is applied to the ingot region to extract the round end of the ingot.

Upon extracting the images of the ingot ends, we developed a classifier based on a similarity measure. The classifier utilizes the Structural Similarity Index Measure (SSIM) to calculate the SSIM values between the image of the ingot end and a selected reference image of the "brain" pattern. Using an SSIM threshold of 0.75, the precision for the top endpoint classification is 1, recall is 0.8134, and the overall classification accuracy is 0.9180. In most scenarios, a singular SSIM classifier can complete the classification task, but the choice of threshold makes the system's robustness somewhat lacking.

To enhance the performance of the classifier, we proposed a dual SSIM index classifier. This novel classifier calculates two SSIM indices by comparing the similarity between the ingot end images and the reference images of both the top and bottom simultaneously. The two calculated SSIM indices are referred to as $SSIM_{top}$ and $SSIM_{bot}$. The orientation of the ingot is determined by comparing these two SSIM values. If $SSIM_{top}$ is larger than $SSIM_{bot}$, the

ingot is identified as having its top end facing the camera, which should trigger an alert. This new dual SSIM index classifier significantly improves the classification performance. For the top end data, precision is 1, recall is raised to 0.9589, and the overall accuracy is 0.9819. Upon analyzing the misclassified data, it is found that these misclassified instances do not impact the overall direction recognition system and will not cause misjudgments.

Chapter 7

Conclusions and Future Work

7.1 Thesis Achievements

In Chapter 2, the main types of algorithms used in vision measurement are introduced, which mainly cover edge extraction algorithm, image registration algorithm and image fusion algorithm.

In Chapter 3, two vision measurement algorithms are proposed and applied to measure the dimension of high-temperature steel on the production line of a steel plant.

The first uses infrared thermal camera images as visual data input and production line width as a reference to measure steel dimensions. The algorithm uses a combination of structural random forest edge detection and noise reduction algorithm to identify the background conveyor belt image to extract the edges on both sides of the conveyor belt. Next, the actual world size represented by each pixel in the image is calculated using the extracted conveyor belt edge and the actual conveyor belt size. At the same time, the image's homography matrix can be calculated using the vertical conveyor belt edge, and the original image can be projected homography so that all the pixels in the whole image represent the same actual size. After using the background conveyor to determine the pixel size, edge detection and recognition algorithms are used to identify and extract the steel parts in the image and then calculate the steel dimensions.

The second method uses optical images taken by the GoPro camera as input. Optical images have higher pixels and lower input delay than infrared thermal images. In the second method, instead of looking for a reference object in the background environment for size reference, the checkerboard is used after setting the camera to determine the camera's spatial parameters. Pre-calibration of the camera removes the system from relying on specific objects in the background and provides a good reference size when the environment is complex and the conveyor belt is not perpendicular to the camera. After that, the structural random forest edge detection and recognition algorithm is used to locate the position and edge of the steel section.

After identifying the edge of the section, a sliding window random regression algorithm is used to filter the edge and estimate the error. Finally, the internal and external parameters of the camera obtained by calibration can be used directly to map the steel section dimensions in the image plane to the real plane.

In order to correct the inconsistency between the measurement plane and the calibration plane, a dual-camera system with two GoPro cameras placed separately is used in Chapter.4.

Due to the complexity of the factory environment, in order to improve the ability and accuracy of mid-distance measurement, the camera interval of the dual-camera system is set relatively large, and a checkerboard is used as the feature point of image registration. Image registration using the corner of the calibration board as a reference can initially pair the images from the left and right cameras. Further adjustments are needed for high-precision vision measurements. Since the error of the monocular camera is caused by the mismatch between the measurement plane and the calibration plane, the idea of using a dual-camera system to solve this problem is to change the calibration plane to the measurement plane. The measured plane is determined by the steel's size and the conveyor's height, so the measured plane is at an objective height and is not a variable that can be changed. That is, the measurement system can only choose to change the calibration plane to make it coincide with the measurement plane. The position of the checkerboard determines the position of the calibration plane. Any calibration plane data can be obtained by continuously changing the checkerboard's position. However, physically moving the checkerboard directly and continuously to match different sizes of steel is time-consuming and impossible on some production lines. Therefore, a virtual checkerboard is proposed to change the position of the calibration plane. The checkerboard of any height can be theoretically interpolated and extrapolated by obtaining two images of the checkerboard of different heights.

After obtaining a virtual checkerboard at any height, the chapter provides an algorithm for calculating the accuracy of steel image registration. A measure of registration accuracy is given by identifying the degree of overlap of steel parts near the target area in the image after registration. Higher coincidence for the steel part results in higher registration accuracy. At the same time, registration accuracy will provide a direction based on the location of the non-coincident portions of the left and right camera steel images. With this direction and accuracy, the system knows whether to retrieve a virtual checkerboard at a higher height or lower. In this way, by combining the registration accuracy with the virtual checkerboard, the virtual checkerboard with the highest accuracy is selected as the calibration plane and used as the reference system for dimension conversion. The calibration plane produced by the virtual calibration plate selected in this way can be consistent with the measured plane height of the steel, which solves the measurement error caused by the plane mismatch.

In Chapter 5, after image registration, image fusion is required in order to combine the image information taken by dual cameras to improve the image quality.

Fourier image fusion and wavelet transformation fusion are used in a factory case to fuse the registered image. After image fusion, various image fusion analysis indexes are used to analyze the image quality after fusion. The edge extraction algorithm is used to extract the edges of the fused image, and the edges extracted by different fusion methods and without image fusion are compared.

The Fourier image fusion used in Chapter 5 is a direct two-dimensional Fourier transform of the source image combined with the fusion rule. In addition to Fourier transform fusion, discrete wavelet transform image fusion is also tested. Wavelet and Fourier chord waves are different. Wavelet oscillations are concentrated near one point. Through scaling and translation operations, multi-scale thinning analysis of functions or signals solves many complex problems that Fourier Transform cannot solve. In the experiment, DB and FK wavelets of different series are used as wavelet bases to transform the image. The coefficients after the wavelet transform are fused to complete the operation of image fusion.

The fused images are evaluated with multiple metrics. From these data, FFT image fusion results are better. However, due to the nature of FFT, a large amount of data is redundant information, and the fusion results can also visually see the texture shadow. Therefore, from the quality of fused image, image fusion using FK4 wavelet brings the best results.

In the experiment of edge extraction for fusion results, both FFT and DWT image fusion provide better results than using single-camera images directly. In edge extraction, the texture ghosting produced by FFT has a positive effect instead. Texture becomes blurred except for the steel part, resulting in fewer unwanted edges being identified.

In Chapter.6, a classification case in an actual steel plant is introduced. The factory needs to determine the direction of the ingots placed on the production line, and if the top of the ingot (the one with the 'Brain' pattern) is facing the camera, it needs to alert the staff to change the ingot direction in time. Wrong ingot direction can affect the performance of finished steel and lead to a scrap of steel.

An algorithm for automatically detecting and intercepting the round surface of the ingot end is provided. First, the high-brightness part of the image is obtained by binarizing the image, and then the blocks of the objects are obtained by checking the pixel connectivity. Next, the ingot region is found by checking the area, the orientation and the position among the blocks. Finally, the Hough circle detection algorithm is applied to the ingot region to extract the round end of the ingot.

After extracting the ends of the ingots, a classifier is developed using a similarity measure. The classifier uses the structural similarity measure to calculate the SSIM values between the

end image and the selected reference brain pattern image. In most cases, a single SSIM classifier can complete the classification task, but the threshold selection makes the system less robust.

To improve the performance of the classifier, a dual SSIM index classifier is presented. The new classifier calculates two SSIM indexes by simultaneously comparing the similarity between the ingot ends images and the top and bottom reference images. The two calculated SSIM index is named $SSIM_{top}$ and $SSIM_{bot}$. The direction of the ingot is determined by comparing the two SSIM values. If $SSIM_{top}$ is larger than $SSIM_{bot}$, the ingot is identifyied as the top end to the camera, which should be warned. The new dual SSIM index classifier improves classification performance a lot. For top ends data, the precision is 1, the recall is raised to 0.9589, and the overall accuracy is 0.9819. From analysing the misclassified data, it can be found that these misclassified cases will not affect the whole direction recognition system and will not cause misjudgments.

7.2 Direction of Future Research

In terms of edge extraction and steel recognition, the future research direction can be to use deep learning methods to complete a one-step recognition and extraction model. This way of image recognition should improve the overall robustness of the method. Then, the selection of thresholds will be more adaptive.

In image registration, the existing algorithm for calculating the virtual checkerboard and the method for selecting the virtual checkerboard are direct interpolation and comparison. In the following work, we can optimize the selection process of the checkerboard and use the closed-loop feedback system to automatically fit the parameters of the virtual checkerboard to improve the efficiency of the entire selection process.

At the current stage, the measuring plane of the steel section remote measuring approach, proposed in this dissertation, is parallel to the plane of the actual calibration plate. Therefore, directly translating the virtual checkerboard can obtain the calibration data of various parallel planes. In future research, the virtual calibration board can be rotated and turned to obtain the data of each angle plane in 3D space. In this way, the same algorithm can still be used even if the object is placed on the plane intersecting the calibration plate plane.

The measurement algorithm in this dissertation is to measure the size of a single steel section. Suppose multiple objects in the image are identified through the target detection algorithm, and the image registration technology of the virtual checkerboard is applied to multiple objects. In that case, it is possible to measure different objects in the image at the same time. Then, pixel-level image fusion can be performed on different objects in the same image to enhance the image quality.

In the current pattern recognition algorithm, the standard image is an image extracted from the existing data. An image selected in this way does not summarize all the features of a classification well. In the next process, an algorithm needs to be developed to update the standard image continuously with new features got from other samples. This standard reference image will be more efficient in comparing more data and will have better results in the classification.

Future work will also focus on exploring the capabilities of deep learning methods that can adapt to different changes in the environment. Convolutional Neural Networks (CNNs), such as region-based CNNs and Bayesian networks, show great potential in solving complex recognition tasks and can be considered for further improvement of steel detection and measurement systems. These deep learning approaches can enhance the accuracy and robustness of the system by automatically learning and adapting to various steel surface conditions and defects.

Additionally, an important area of future research is the integration of the results from the recognition tasks into the control algorithm for the steel rolling process. By embedding the information obtained from the steel detection and measurement system, real-time adjustments can be made to optimize the rolling parameters and improve overall productivity and quality. This integration will require the development of advanced algorithms and models that can effectively utilize the detected information for closed-loop control.

Furthermore, future research could also develop advanced sensor fusion approaches to combine data from multiple sources, such as vision-based systems, laser measurements and other non-contact measurement technologies. This integration can provide a more comprehensive understanding of the steel production process, enabling enhanced detection and measurement capabilities.

In summary, future research efforts will focus on leveraging deep learning methods, integrating recognition results into control algorithms, and exploring sensor fusion techniques to further enhance the efficiency, accuracy, and adaptability of steel detection and measurement systems in the steelmaking process. These advancements will contribute to improving operational efficiency, reducing waste, and ensuring high-quality steel production.

Bibliography

[1] Ikram E Abdou and William K Pratt. Quantitative design and evaluation of enhancement/thresholding edge detectors. *Proceedings of the IEEE*, 67(5):753–763, 1979.

[2] Md Zahangir Alom, Tarek M Taha, Christopher Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Brian C Van Esesn, Abdul A S Awwal, and Vijayan K Asari. The history began from alexnet: A comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164*, 2018.

[3] Veysel Aslantas and Rifat Kurban. Fusion of multi-focus images using differential evolution algorithm. *Expert Systems with Applications*, 37(12):8861–8870, 2010.

[4] Veysel Aslantaş and Emre Bendes. A new image quality metric for image fusion: The sum of the correlations of differences. *AEU-International Journal of Electronics and Communications*, 69(12):1890–1896, 2015.

[5] Chao Bi, Jianguo Fang, Di Li, and Xinghua Qu. Study on application of color filters in vision system of hot forgings. In *Optical Measurement Technology and Instrumentation*, volume 10155, pages 1015522–1–1015522–9. International Society for Optics and Photonics, 2016.

[6] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: binary robust independent elementary features. In *Proceedings of the European Conference on Computer Vision*, 2010.

[7] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):679–698, 1986.

[8] JK Che and Mani Maran Ratnam. Real-time monitoring of workpiece diameter during turning by vision method. *Measurement*, 126:369–377, 2018.

[9] Guan-Hao Chen, Chun-Ling Yang, Lai-Man Po, and Sheng-Li Xie. Edge-based structural similarity for image quality assessment. In *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, volume 2, pages II–II. IEEE, 2006.

[10] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. Ieee, 2005.

[11] Ingrid Daubechies. Where do wavelets come from? a personal point of view. *Proceedings of the IEEE*, 84(4):510–513, 1996.

[12] Thomas G Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, 40(2):139–157, 2000.

[13] Piotr Dollár and C Lawrence Zitnick. Fast edge detection using structured forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(8):1558–1570, 2014.

[14] A.M. Eskicioglu and P.S. Fisher. Image quality measures and their performance. *IEEE Transactions on Communications*, 43:2959–2965, 1995.

[15] Hassen Fourati. *Multisensor Data Fusion: From Algorithms and Architectural Design to Applications (Book)*. Series: Devices, Circuits, and Systems, CRC Press, Taylor & Francis Group LLC, August 2015.

[16] Yoav Freund and Llew Mason. The alternating decision tree learning algorithm. In *Proceedings of ICML*, volume 99, pages 124–133, 1999.

[17] Xianbin Fu, Bin Liu, and Yucun Zhang. An optical non-contact measurement method for hot-state size of cylindrical shell forging. *Measurement*, 45(6):1343–1349, 2012.

[18] Silvio Giancola, Matteo Valenti, and Remo Sala. *A Survey on 3D Cameras: Metrological Comparison of Time-of-Flight, Structured-Light and Active Stereoscopy Technologies*. Springer, 2018.

[19] Mohammad Bagher Akbari Haghighat, Ali Aghagolzadeh, and Hadi Seyedarabi. A non-reference image fusion metric based on mutual information of image features. *Computers & Electrical Engineering*, 37(5):744–756, 2011.

[20] David L. Hall and James Llinas. *Multisensor Data Fusion*. Electrical Engineering & Applied Signal Processing Series. CRC Press, 2001.

[21] Robert M Haralick and Linda G Shapiro. *Computer and robot vision*, volume 1. Addison-wesley Reading, 1992.

[22] Ying He, Bin Liang, Yu Zou, Jin He, and Jun Yang. Depth errors analysis and correction for Time-of-Flight (ToF) cameras. *Sensors*, 17(1):92, 2017.

[23] Haithem Hermessi, Olfa Mourali, and Ezzeddine Zagrouba. Multimodal medical image fusion review: Theoretical background and recent advances. *Signal Processing*, 183:108036, 2021.

[24] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 278–282. IEEE, 1995.

[25] Carl G Hoyos and BM Zimolong. *Occupational safety and accident prevention: behavioral strategies and methods*. Elsevier, 2014.

[26] Alex P. James and Belur V. Dasarathy. Medical image fusion: A survey of the state of the art. *Information Fusion*, 19:4–19, 2014.

[27] Xin Jin, Qian Jiang, Shaowen Yao, Dongming Zhou, Rencan Nie, Jinjin Hai, and Kangjian He. A survey of infrared and visual image fusion methods. *Infrared Physics Technology*, 85:478–501, 2017.

[28] Xiangxiong Kong and Jian Li. Vision-based fatigue crack detection of steel structures using video feature tracking. *Computer-Aided Civil and Infrastructure Engineering*, 33(9):783–799, 2018.

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[30] Krishna Kummamuru and Narasimha Murty M. Genetic K-means algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 29(3):433–439, 1999.

[31] Jiewu Leng, Weinan Sha, Baicun Wang, Pai Zheng, Cunbo Zhuang, Qiang Liu, Thorsten Wuest, Dimitris Mourtzis, and Lihui Wang. Industry 5.0: Prospect and retrospect. *Journal of Manufacturing Systems*, 65:279–295, 2022.

[32] Shutao Li, Xudong Kang, Leyuan Fang, Jianwen Hu, and Haitao Yin. Pixel-level image fusion: A survey of the state of the art. *Information Fusion*, 33:100–112, 2017.

[33] Shutao Li and Bin Yang. Multifocus image fusion using region segmentation and spatial frequency. *Image and Vision Computing*, 26(7):971–979, 2008.

[34] Shutao Li, Bin Yang, and Jianwen Hu. Performance comparison of different multi-resolution transforms for image fusion. *Information Fusion*, 12(2):74–84, 2011.

[35] Wei Liu, Zhenyuan Jia, Fuji Wang, Xin Ma, Wenqiang Wang, Xinghua Jia, and Di Song. An improved online dimensional measurement method of large hot cylindrical forging. *Measurement*, 45(8):2041–2051, 2012.

[36] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157. IEEE, 1999.

[37] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[38] Artur Loza, Lyudmila Mihaylova, David Bull, and Nishan Canagarajah. Structural similarity-based object tracking in multimodality surveillance videos. *Machine Vision and Applications*, 20(2):71–83, February 2009.

[39] Yue Luo, Jimmy Ren, Mude Lin, Jiahao Pang, Wenxiu Sun, Hongsheng Li, and Liang Lin. Single view stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 155–163, 2018.

[40] Shanshan Lv, Mingshun Jiang, Chenhui Su, Lei Zhang, Faye Zhang, Qingmei Sui, and Lei Jia. Phase difference-3D coordinate mapping model of structural light imaging system based on extreme learning machine network. *IEEE Access*, 8:68974–68981, 2020.

[41] Thulfiqar H Mandeel, Muhammad Imran Ahmad, Mohd Nazrin Md Isa, Said Amirul Anwar, and Ruzelita Ngadiran. Palmprint region of interest cropping based on moore-neighbor tracing algorithm. *Sensing and Imaging*, 19(1):15, 2018.

[42] David Marr. Vision: A computational investigation into the human representation and processing of visual information. 1982.

[43] Yasir M Mustafah, Rahizall Noor, Hasbullah Hasbi, and Amelia Wong Azma. Stereo vision images processing for real-time object distance and size measurements. In *Proceedings of the 2012 International Conference on Computer and Communication Engineering (ICCCE)*, pages 659–663. IEEE, 2012.

[44] Sofia C. Olhede and Andrew T. Walden. The Hilbert spectrum via wavelet projections. *Proceedings of The Royal Society A: Mathematical, Physical and Engineering Sciences*, 460:955–975, 2004.

[45] Mohammadreza Asghari Oskoei and Huosheng Hu. A survey on edge detection methods. *University of Essex, UK*, 33, 2010.

[46] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, 1979.

[47] Gonzalo Pajares and Jesús Manuel de la Cruz. A wavelet-based image fusion tutorial. *Pattern Recognition*, 37:1855–1872, 2004.

[48] Xavier Soria Poma, Edgar Riba, and Angel Sappa. Dense extreme inception network: Towards a robust cnn model for edge detection. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 1923–1932, 2020.

[49] David MW Powers. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*, 2020.

[50] Paul L. Rosin. Measuring corner properties. *Computer Vision and Image Understanding*, 73:291–307, 1999.

[51] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *Proceedings of the European Conference on Computer Vision*, 2006.

[52] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: An efficient alternative to SIFT or SURF. In *Proceedings of International Conference on Computer Vision*, 2011.

[53] Claude Elwood Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423, 1948.

[54] S Sharma, Ananya Kalyanam, and Sameera Shaik. Technical analysis of CNN-Based face recognition system—a study. In *Proceedings of the Computational Intelligence in Data Mining*, pages 545–556. Springer, 2019.

[55] Pierre Soille. *Morphological image analysis: principles and applications*. Springer Science & Business Media, 2013.

[56] G. Sreeja and O. Saraniya. Chapter 3 - Image Fusion Through Deep Convolutional Neural Network. In Arun Kumar Sangaiah, editor, *Deep Learning and Parallel Computing Environment for Bioengineering Systems*, pages 37–52. Academic Press, 2019.

[57] Xiaohong Sun, Jinan Gu, Shixi Tang, and Jing Li. Research progress of visual inspection technology of steel products—a review. *Applied Sciences*, 8(11):2195, 2018.

[58] Zhisong Tian, Feng Gao, Zhenlin Jin, and Xianchao Zhao. Dimension measurement of hot large forgings with a novel time-of-flight system. *The International Journal of Advanced Manufacturing Technology*, 44(1-2):125–132, 2009.

[59] Bangguo Wang, Wei Liu, Zhenyuan Jia, X Lu, and Y Sun. Dimensional measurement of hot, large forgings with stereo vision structured light system. *Journal of Engineering Manufacture*, 225(6):901–908, 2011.

[60] Peng Wang, Yueda Lin, Ree Muroiwa, Simon Pike, and Lyudmila Mihaylova. Computer vision methods for automating high temperature steel section sizing in thermal images. In *Proceedings of the Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–6. IEEE, 2019.

[61] Zhiyuan Wang, Renwei Liu, Todd Sparks, Heng Liu, and Frank Liou. Stereo vision based hybrid manufacturing process for precision metal parts. *Precision Engineering*, 42:1–5, 2015.

[62] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[63] Andrew Webb. *Statistical Pattern Recognition*. John Wiley and Sons, Ltd., 2011.

[64] James M Wilson and Alan McKinlay. Rethinking the assembly line: Organisation, performance and productivity in Ford Motor Company, c. 1908-27. *Business History*, 52(5):760–778, 2010.

[65] Bin Wu, Fang Zhang, and Ting Xue. Monocular-vision-based method for online measurement of pose parameters of weld stud. *Measurement*, 61:263–269, 2015.

[66] Bin Yang and Shutao Li. Multifocus image fusion and restoration with sparse representation. *IEEE Transactions on Instrumentation and Measurement*, 59(4):884–892, 2009.

[67] Jinghao Yang, Wei Liu, Renwei Zhang, Zhenyuan Jia, Fuji Wang, and Shijie Li. A method for measuring the thermal geometric parameters of large hot rectangular forgings based on projection feature lines. *Machine Vision and Applications*, 29(3):467–476, 2018.

[68] Lu Yang, Baoqing Wang, Ronghui Zhang, Haibo Zhou, and Rongben Wang. Analysis on location accuracy for the binocular stereo vision system. *IEEE Photonics Journal*, 10(1):1–16, 2017.

[69] Jae-Chern Yoo and Tae Hee Han. Fast normalized cross-correlation. *Circuits, Systems and Signal processing*, 28:819–843, 2009.

[70] Aneta Zatočilová, David Paloušek, and Jan Brandejs. Image-based measurement of the dimensions and of the axis straightness of hot forgings. *Measurement*, 94:254–264, 2016.

[71] Lixia Zhang, Guangping Zeng, and Zhaocheng Xuan. A survey of fusion methods for multisource image. *Computer Engineering and Science*, 44(02):321–334, 2022.

[72] Song Zhang. High-speed 3D shape measurement with structured light methods: A review. *Optics and Lasers in Engineering*, 106:119–131, 2018.

[73] Yijun Zhou, Yongchao Wu, and Chen Luo. A fast dimensional measurement method for large hot forgings based on line reconstruction. *The International Journal of Advanced Manufacturing Technology*, 99(5-8):1713–1724, 2018.

[74] Jingguo Zhu, Menglin Li, Yan Jiang, Tianpeng Xie, Feng Li, Chenghao Jiang, Ruqing Liu, and Zhe Meng. Research on online 3D laser scanner dimensional measurement system for heavy high-temperature forgings. In *Proceedings of the AOPC 2017: 3D Measurement Technology for Intelligent Manufacturing Conference*, volume 10458, pages 104581Q–1–104581Q–9. International Society for Optics and Photonics, 2017.

[75] Barbara Zitová and Jan Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21:977–1000, 2003.