Team Number :	apmcm2208815
Problem Chosen :	A

2022 APMCM summary sheet

Feature Extraction of Sequence Images and Modeling Analysis of Mold Flux Melting and Crystallization

By establishing a pixel discrimination model, the problem of identifying temperature data in images is solved. By establishing a feature-based equation of state model, the problem of quantifying image features and establishing the process curve of mold melting and crystallization is solved. By establishing a polynomial regression model, the problems of temperature and time change and the functional relationship between mold flux melting and crystallization process are solved, and the reliability of the results is verified by combining the JMA formula in crystallization kinetics.

For problem one, we establishes a pixel discrimination model, extracts the temperature in the upper left corner of each picture, and makes temperature time curves of 1# and 2# respectively. Firstly, the picture is grayscale processed and the global threshold is selected to binarize the picture, and the image is represented by a 0-1 pixel matrix; Then, each preprocessed picture is cut into 8 parts according to the characteristics of the digital font and position of the picture, and then the recognition rate is improved by filtering and noise reduction processing. Then, 0-9 numbers and °C are selected as templates, and by calculating the correlation coefficient between the image and the template, the template with the largest value is selected as the recognition result, and automatically stored in the table; Finally, the convolutional neural network is used to test the results, and the results are reliable.

For problem two, we establish a feature-based equation of state model, study the dynamic differences in the melting and crystallization processes, and discuss the process curves of melting and crystallization for different characteristics. First, the image is preprocessed to reduce the pixel size to 300*400 size; Then, the "Canny" operator is used to detect the edge, the edge contour image is obtained, and the find function is used to obtain the index vector for the contour edge image to separate the mold from the background. Then, the Tamura texture feature algorithm is used to extract the roughness, contrast, directionality and linearity, observe the change law of the mold in the process of melting and crystallization, and find its recursive formula about time. Finally, the average gray value and roughness are visualized, and the polynomial fitting is realized, and the melting and crystallization process curves after fitting are obtained. The results are tested and analyzed by the five features of roughness, contrast, directionality, linearity and average gray value, which proves that the results are consistent with the experimental phenomena and verifies the reliability of the feature-based equation of state model.

For problem three, a polynomial regression model is developed in this paper to solve the problem of establishing the temperature as a function of time and the melting and crystallisation of the mould melt. Firstly, a polynomial regression is fitted to the temperature at different moments from Problem 1 to establish a segmental function expression. Then, based on the process curves for mould melting and crystallisation found in Problem 2, polynomial regression expressions were created for the amount of melting versus temperature and the amount of crystallisation versus temperature at the same moments. The crystallisation rate is then related to time, and the melting rate is related to time. Finally, in conjunction with the crystallisation kinetics, it was found that the graph of crystallisation rate versus temperature found in this paper was in high agreement with the image of the JMA equation, proving the accuracy and reliability of the results obtained.

Keywords: Convolutional neural networks; Equation of state; Tamura algorithm; Polynomial regression model

Contents

1 Introduction

1.1 Problem Background

The phase distribution of the mould flux in the gap between the mould wall and the strand shell is of great importance for continuous casting. However, high temperatures, transient flows, the complexity of phase changes and chemical reactions and the opacity of the mould wall make it difficult to observe the phase changes of the mould flux directly in this paper. Furthermore the data obtained by the SHTT II tester at melting and crystallisation temperatures needs to be processed by a large number of experimenters before it can guide the design of the mould co-solvent. For this reason, there is an urgent need to develop automatic feature extraction and mathematical modelling techniques for column sequential images.

Melting process: A flux for continuous casting moulds is added to the top of the molten steel in the mould. The solid slag that accumulates on the surface of the molten steel prevents horizontal crusting of the molten steel due to the temperature of the molten steel dropping too quickly. The mould flux is then melted when the temperature of the mould flux reaches its melting point and a sintered layer is formed. The raw material for the flux starts as a low melting point substance and is melted to form a liquid slag, the composition of which changes during the process.

Crystallisation process: During the melting process, a liquid slag layer is produced and covers the surface of the molten steel. The liquid slag may penetrate from the steel surface into the gap between the shell and the copper mould wall, thus forming a slag film. Since the temperature of the liquid slag is positively correlated with the temperature of the steel strand surface, if the mould is forced to cool, the slag film against the copper mould wall will quench and solidify, forming a glassy solid slag film (solidification behaviour of the slag film), while the slag film will, under certain conditions, form a crystalline layer in certain areas (crystallisation behaviour of the slag film), thus constituting a three-layer slag film structure: glassy layer, crystalline layer and liquid slag layer.1.2 Restatement of the Problem

1.2 Problem requirements

(1) Question 1: Using image segmentation and recognition or other techniques, extract the corresponding data for each image and make a temperature time profile. one set of test results in line 1# or 2# is inaccurate and needs to be pointed out and explained.

(2) Question 2: Based on the images in Figure 1, use digital image processing techniques to investigate and quantify the dynamic differences between adjacent sequences of images during mould flow melting and crystallisation. Using this as a basis for time series modelling of the quantified different features, discuss the process profiles of mould melting and crystallisation based on the simulation results obtained from the mathematical model.

(3) Question 3: Based on the results of the studies in questions 1 and 2, develop a mathematical model to discuss the variation of temperature and time as well as the melting and crystallisation processes of the mould melt as a function of temperature. The relationship between the temperature, melting rate and crystallisation rate of the mould melt is then discussed in the light of these results.

2 Problem analysis

Three problems are proposed for the complex phase change and chemical reaction of the mold flux under high temperature heating in the continuous casting process, which makes it difficult to directly observe the phase of the mold flux. First, in order to reduce manpower, it is necessary to develop digital recognition of sequence images and record temperature data; Then, the sequence image is extracted by features, and the model is established through the features to study the process curve of mold melting and crystallization. Finally, the extracted temperature data and the process curve of mold melting and crystallization are deduced to derive the functional relationship expression, and the functional relationship expression of temperature, melting rate and crystallization rate is determined. The three questions, from shallow to deep, are intertwined.

2.1 Analysis of Question One

In the analysis of the images, the numbers to be identified in each of the 562 images are located in the top left corner of the image and the spacing between the data font and the numbers is fixed. The cut will result in a smaller image size and fewer pixel points. It is envisaged that the image will be enlarged by a factor of 10 in preparation for the subsequent operation. Before cutting, this paper envisages grey-scale processing and binarisation of the image, simplifying the image into black and white bicolour i.e. two values of 0,1, splitting out the numbers in the image and then cutting the numbers into 158*93 pixels in size, and setting the size of the numbers to 28*28 pixels by scaling uniformly; this paper notes that the edges of the numbers may not be smooth, and in order to make the edges of the numbers in the image smoother In this paper, it is noted that the edges of the figures may not be smooth, so in order to make the edges of the figures in the image smoother, the image is filtered and noise reduced; this paper envisages cutting out 8 figures per image, but upon observation the cut images are not all figures, this paper envisages identifying the non-numbered °C as the number 10, and then carrying out uniform processing. The correlation coefficient is calculated between the cut image data and the designed standard data. According to the prediction, the larger the correlation coefficient, the higher the similarity between the image data and the standard data. Based on this principle, this paper envisages the establishment of a pixel discrimination model, which will be used to perform a circular traversal to identify the temperature of each image in comparison with the standard sample, exported and input into a table, and thus make a temperature time profile. The paper envisages a comparative analysis of this model and the use of convolutional neural networks to build a digital image recognition model. If the conclusions reached are not significantly different from those of the pixel discrimination model, then the pixel discrimination model is reasonable and the test results are accurate.

2.2 Analysis of Question Two

Problem 2 is a feature extraction type of problem. In this paper, Tamura texture features are used to extract four features: roughness, contrast, directionality and linearity. To make the computation more sensitive, the image is pre-processed with a greyscale process that turns the image into only black and white, and then scales the image to a uniform size. This paper envisages the use of the "Canny" operator to detect edges, but the edge operator has the limitation that it only responds to edges and finds all edges, while non-edges should be discarded. Therefore, in order to make the edge operator results more accurate, this paper envisages a series of image processing: first a Gaussian smoothing filter and convolution of the image to make the image edges smooth; then a non-maximum suppression of the amplitude of the gradient to make the blurred boundary clear; and then a double thresholding technique to further eliminate the noise. The edges are tracked in the image obtained after processing to obtain an edge profile image and to separate the mould from the background. This paper envisages the use of Tamura texture feature extraction followed by observing the changes in the four features during the crystallisation and melting process and finding out their change patterns, listing the feature equations and plotting the results obtained after the calculation into a graph to make the results more intuitive.

2.3 Analysis of Question Three

Problem 3 belongs to the functional relationship derivation class problem. Based on the temperature change data in question 1 and the process curve of mold melting and crystallization in question 2, the functional relationship between them is derived. For the accuracy of the function expression, we chose to use polynomial regression fitting to make the discrete data more relevant to the function image. The function relationship between mold flux melting and crystallization and temperature can be derived by establishing the function relationship between temperature and time and mold flux melting and crystallization and time. The functional relationship is visualized and combined with the crystallization kinetic formula to test the accuracy of the establishment of the functional relationship. If the images are similar, the results are reliable.

3 Model assumptions

(1) Assume that the text in the upper left corner of the image is the top frame.

(2) Assume that the moment of image recording is accurate, and the error is not considered.
(3) Assuming that the mold flux is normal, there is no problem with the use process.

4 Symbol description

Symbol	Description
A_i	A matrix of pixels of 28*28 size for the i number
r_{ij}	The 0-1 pixel matrix of the i digital image has a correlation coefficient with the matrix of the j template
$\max r_{ij}$	The 0-1 pixel matrix of the i digital image is most similar to the matrix of the j template
T_{1t}	Thermocouple No. 1 temperature
T_{2t}	Thermocouple No. 2 temperature
$C_k(x,y)$	The average gray value of the cell
$g(i,j)$	The grayscale value of the pixel at (i,j)
$\Delta G(x,y)$	The gradient vector at each pixel
$N_\theta(i)$	The total number of edge pixels on the corresponding direction angle θ
n_p	The number of histogram peaks
ω_p	Peak quantization range
P_a	The distance point of the $m \times m$ locally oriented symbiotic matrix
b_t	The melting amount of the mold at time t
c_t	The roughness of the mold at time t
I	The number of grains that can be formed per unit volume per unit of time
V^s	The volume of liquid crystallized
V^t	The remaining volume of uncrystallized liquid
a_i	The amount of mold crystallization at the i moment
b_i	The amount of mold melt at the i moment
m_i	Crystallization rate at moment i

n_i	Melting rate at moment i

5 Model building and solutions

5.1 Model buiding and solving Problem One

5.1.1 Image pre-processing

In this paper, the images are first greyed out as well as binarised, and then segmented based on the fact that the data is in the same position.

Step1: The image in Annex 1 was greyed out to give 1316*1792 pixel points, each with a value of 0-255 for radians, 255 for white and 0 for black. Next, the image is binarised. The image is made up of a matrix, each point in the matrix has a different RGB value, which results in a different colour, and the final overall presentation to this paper is a coloured image. The binarisation process changes the range of colours from 256 to 2, thus increasing the speed of computation. A global threshold is then chosen to divide the image into a black or white binary image, with a grey value greater than or equal to the threshold assigned to 1 and vice versa, so that each pixel is represented by 0 or 1, with 0 being black and 1 being white.

Step2: Based on the size of each image, the position of the top left corner data and the fixed font and spacing of the data, this paper sets a fixed cut line to cut the top left corner data of each image. 65*110 pixel points are cut out from the top left corner and the image is enlarged by a factor of 10 in preparation for the subsequent operation.

5.1.2 Pixel discriminant modeling

The model building process in this paper is represented in a flow chart, which is shown below.

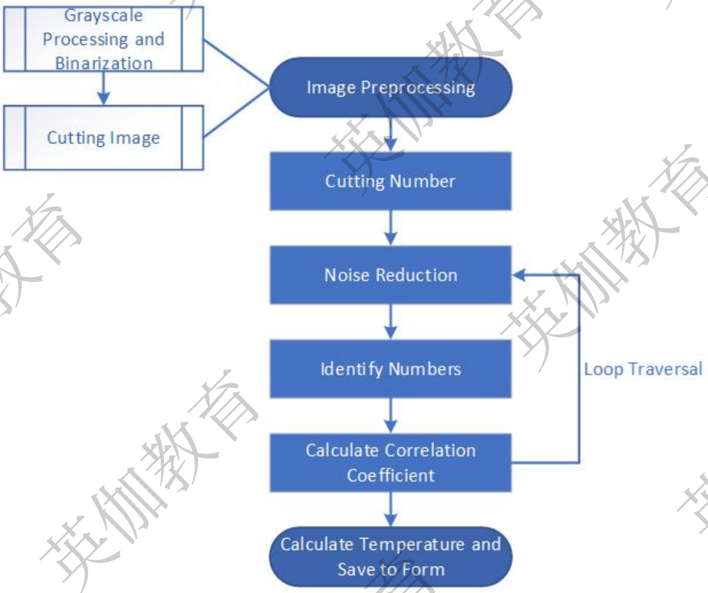
Grayscale Processing and Binarization
Cutting Image
Image Preprocessing
Cutting Number
Noise Reduction
Identify Numbers
Loop Traversal
Calculate Correlation Coefficient
Calculate Temperature and Save to Form

Figure1 Model building flow chart

This article uses a pixel point as a unit and sets the coordinates of the first pixel in the upper left corner to $(1,1)$ to position the number in the image. Since the temperature values of thermocouple 1 and thermocouple 2 are either 3 or 4 digits, the paper is cut at 4 digit size spacing and in pixels, the diagonal coordinates of the size spacing of thermocouple 1 temperature are $(501,280)$ to $(873,366)$ and similarly the diagonal coordinates of thermocouple 2 temperature are $(501,393)$ to $(873,551)$. The difference between the horizontal coordinates of the four figures is 372 pixel points, i.e. each figure is 92 pixel points wide. The vertical coordinate difference is 158 pixel points, i.e. each digit is 158 pixel points high. By this criterion, each digit is cut into an image of 158*93 pixel size, and by scaling, the size of each digit is set to 28*28 pixel points.

Next, the image is filtered for noise reduction, which corrodes the image and then expands it to remove small deviating pixel points. In this paper, the borders of larger objects are smoothed without changing their area, giving the figures in the image smoother edges.

Then, among these digital images, a total of 11 images from 0-9 as well as °C were identified as templates. The correlation coefficient was calculated for each digital image in turn with the 11 templates through a 0-1 matrix of size 28*28, and the correlation coefficient was calculated as follows.

$$r_{ij}=\frac{\sum_{m}\sum_{n}\left(A_i-\bar{A}_i\right)\left(B_j-\bar{B}_j\right)}{\sqrt{\left(\sum_{m}\sum_{n}\left(A_i-\bar{A}_i\right)\right)^2\left(\sum_{m}\sum_{n}\left(B_j-\bar{B}_j\right)\right)^2}} \tag{1}$$

$$\left(i=1,2,\cdots\cdots,8;j=1,2,\cdots\cdots,11\right)$$

Where denotes the 0-1 pixel matrix of size 28*28 for the i th number.When $i=1,2,3,4$，the 4 digits of thermocouple No.1 from left to right are depicted，when $i=5,6,7,8$，Respectively, the No.2 thermocouple is represented by 4 digits from left to right. B_j represents a matrix of 0-1 pixels of 28*28 size of the j template。When $j=1,2,\cdots\cdots,10$，A template representing the number "$j-1$"，For example, B_1 represents a template with the number "0". r_{ij} represents the correlation coefficient between the 0-1 pixel matrix of the i digital image and the matrix of the j template，Then $\max r_{ij}$ represents the 0-1 pixel matrix of the i digital image and the matrix of the j template.The i digit in the image at this moment is recognized as $j-1$, so $N_{t,i}=j-1$ means that the i digit at time t is $j-1$.

Finally the data from each image recognition is processed. Where the individual data are of different orders of magnitude, different orders of magnitude are processed for different digits.

$$
\begin{aligned}
&T_{1t}=\begin{cases} N_{t,1}\times 1000+N_{t,2}\times 100+N_{t,3}\times 10+N_{t,4}\left(N_{t,4}\neq 10\right) \\ N_{t,1}\times 100+N_{t,2}\times 10+N_{t,3}\left(N_{t,4}=10\right) \end{cases} \\
&T_{2t}=\begin{cases} N_{t,5}\times 1000+N_{t,6}\times 100+N_{t,7}\times 10+N_{t,8}\left(N_{t,8}\neq 10\right) \\ N_{t,5}\times 100+N_{t,6}\times 10+N_{t,7}\left(N_{t,8}=10\right) \end{cases} \\
&\left\{N_{t,i}\middle|110\leq t\leq 671,1\leq i\leq 8,t,i\in N\right\}
\end{aligned} \tag{2}
$$

Where T_{1t} is the temperature of thermocouple number 1 and T_{2t} is the temperature of thermocouple number 2. The data output is obtained after this processing.

5.1.3 Pixel discriminant model solution and results

In this paper, the 4,496 sets of processed image data are traversed cyclically using the model, and the output is corrected for changes in order of magnitude and output again to obtain the following table (see appendix and supporting material for details)

Table.1 Temperature table for lines 1# and 2# at various times

NO	Time	1#Temperature	2#Temperature
1	110	900	1142
2	111	904	1146
3	112	910	1146
4	113	914	1144
⋮	⋮	⋮	⋮
560	669	815	1211
561	670	814	1146
562	671	812	1144

The output was visualised as required by the question and the 1# line temperature - 2# temperature time graph is shown in the following figure.

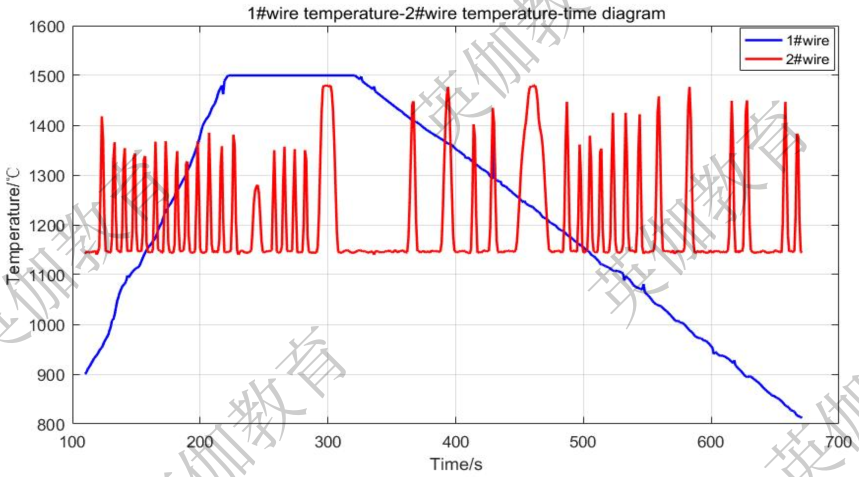

Figure.1 1# line temperature - 2# line temperature time graph

It was observed that the temperature of the 1# line rose smoothly and part of the time at an approximate rate of 5°C per second and 10°C per second, rising to a maximum temperature of 1500°C at 222s. After reaching the maximum temperature the experiment was held for up to 100s then cooled down at a rate of 2°C per second. Observation of the 2# line, the temperature of the 2# line is generally stable and stable for a long time at about 1145°C. At the same time, the temperature of the 2# line fluctuates up and down and some of the temperature changes are large, and there is no obvious pattern.

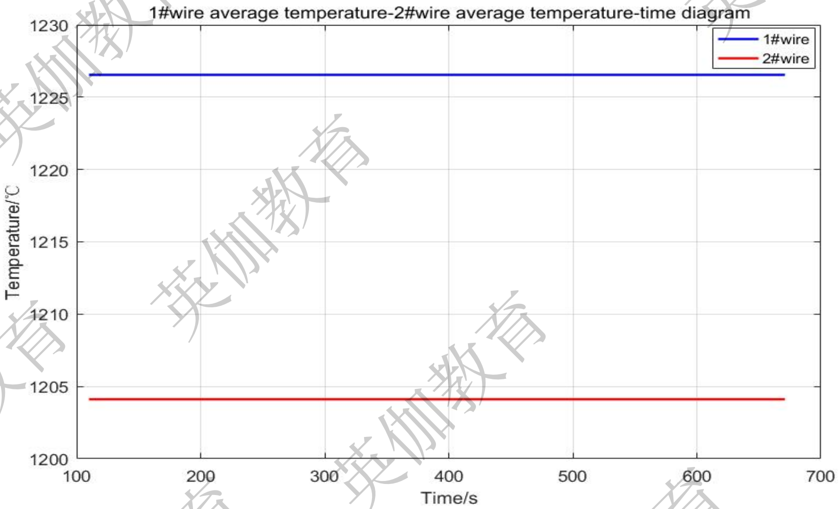

Figure.2 Time diagram of the average temperature of line 1# - average temperature of line 2

From the above graph, it can be seen that the temperature of line 1# increases with time, the temperature first increases, then stays the same and finally decreases; the temperature of line 2# is not very correlated with time and is jagged up and down. In the average temperature image it can be seen that the average temperature of line 1# is higher than the average temperature of line 2#.

The analysis of the 1# line temperature after a linear temperature rise and constant temperature for 100s, followed by a uniform decline, this heating process is consistent with the hot wire method of continuous experiments of temperature control mode. For the 2# line, the temperature always showed irregular jumping lack of reliability. In summary, this paper considers that the temperature data of the 2# line is a problem, the first point may be the 2# line temperature measurement contact poor contact caused by data fluctuations; the second point may be due to the lack of compensation wire and the use of copper wire, then even if there is a temperature gradient in the part will not produce a thermoelectric potential, resulting in errors in the measurement of temperature results.

5.1.4 Test analysis

A convolutional neural network is used to build a digital image recognition model to recognise the image information and compare the output with that of the pixel discrimination model.

5.1.4.1 Introduction to Convolutional Neural Network Principles

Convolutional neural network is a deep feed-forward neural network with features such as local connectivity and shared weights. It is good at processing image, especially image recognition and other related machine learning problems, such as image classification, target detection, image segmentation and other various vision tasks with significant enhancement effects. It is able to classify input information in a translation-invariant manner according to its hierarchical structure, and can perform supervised and unsupervised learning. The shared parameters of the convolutional kernel within its hidden layers and the sparsity of the inter-layer connections enable convolutional neural networks to lattice dotted features with a small amount of computation.

5.1.4.2 How Convolutional Neural Networks Work

A convolutional neural network consists of an input layer, a convolutional layer, a pooling layer, a fully connected layer and an output layer. There are multiple convolutional layers in the network, the purpose of which is to extract image features from the input data, and the sharing of convolutional parameters greatly reduces the number of operations. The presence of multiple convolutional kernels allows for the extraction of image features from multiple perspectives. The convolutional layer is immediately followed by the pooling layer, which acts as a compression of the input features, making the feature map smaller to simplify the complexity of the neural network computation or to re-extract features. [1]

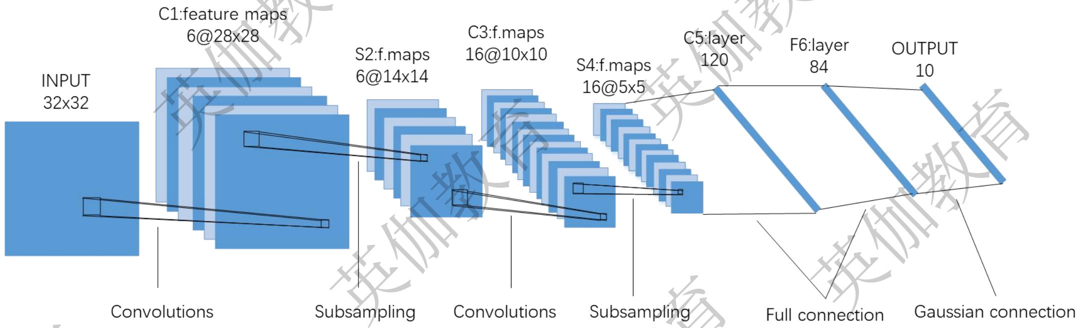

Figure.3 Convolutional neural network working principle diagram

5.1.4.3 Training of Convolutional Neural Networks

Step1: Preparation of the image data to be trained. After the convolutional neural network for handwritten digit recognition has been built, the image data to be trained is input. In this paper, we input eleven sets of image data to be trained, each set contains 23 different handwritten data, a total of 253 handwritten data samples, of which 17 data samples are taken from each set as training data.

Step2: The input layer, convolutional layer and final layer layer parameters are defined. The output layer sets the pixel size template to 28*28 for the sample image data; the convolution layer creates a 2-D convolution layer containing 6 filters of size [5,5]; for the pooling layer this paper uses the maximum pooling function to perform downsampling by dividing the input into rectangular pool regions and then calculating the maximum value for each region. A 2-D convolution layer is then created containing 16 filters of size [5,5]; a pooling kernel of size 2*2 is then created in the pooling layer, and a 2-D convolution layer containing 120 filters of size [5,5] is created for the final feature extraction.

Step3: The convolutional neural network was trained. The prepared training data samples were imported and 50 iterations of training were performed with the following results

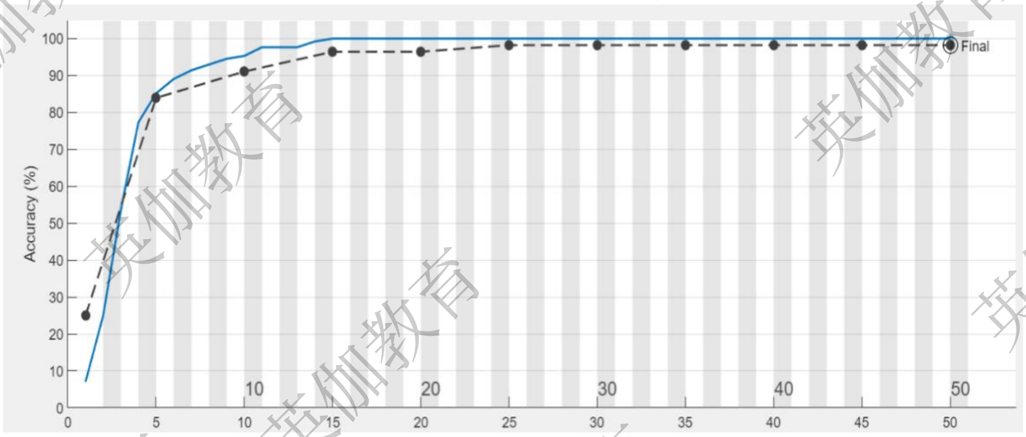

Figure.4 CNN Training Accuracy Image

The actual post-training linearity in the accuracy image is highly consistent with the post-training linearity, with an extremely high training accuracy of 98.21% and valid training results.

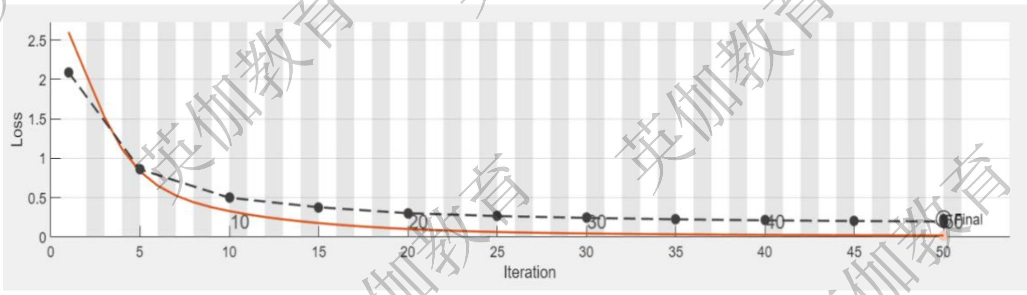

Figure.5 CNN Training Loss Image

Most of the loss curve lines are below 0.5, with less loss, meeting the training requirements. After several training sessions, the training results are all above 90%, and after dozens of training sessions, the training accuracy is up to 98.21%, which basically meets the training requirements. [2]

Step4: The test set was imported for validation. This paper selected eleven experimental groups, respectively 0 to 9 and ℃ (degrees Celsius) in random order for identification, 0 to 9 identification out of normal for the output, for ℃ identification, this paper special treatment output a, and finally the output results of a unified processing, identification results are shown in the following figure.

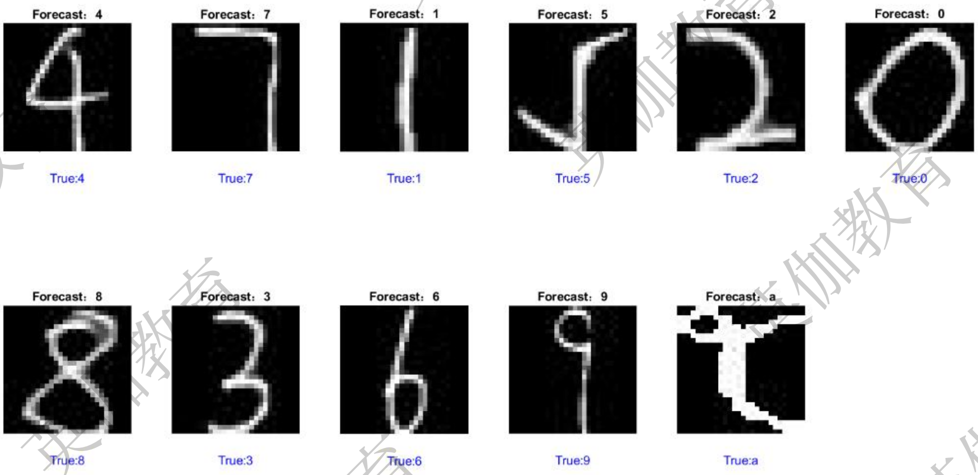

Figure.6 Output of test set validation results

After the image has been analysed, the image is correctly identified for regular digital recognition and output.

5.1.4.4 Use of Convolutional Neural Networks

In this paper, each image will be positioned for cutting as in the preparation above, each image can be separated into eight images data to be analysed after cutting, after all segmentation a total of 4496 images. The images are identified and the results identified are compared with the output of the previous method, the output is shown in the appendix and supporting material, the output is shown in the following figure.

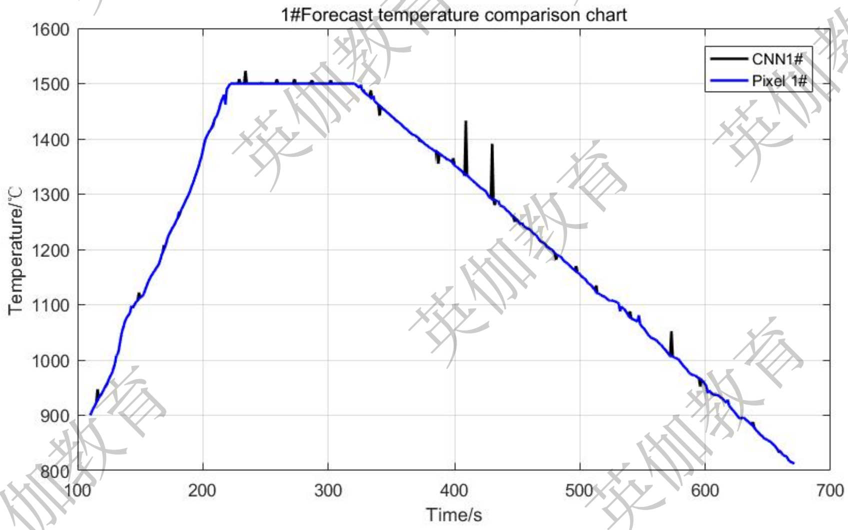

Figure.7 Comparison of CNN and pixel discrimination for 1# line temperature time

After analysis of the image, the black line graph is the result of the convolutional neural network recognition output, the blue line graph is the data obtained from the pixel discrimination method, after comparing the convolutional neural network output results on the image there are a lot of protrusions, part of the data there is a recognition error makes the image jitter, the convolutional neural network used in this paper in the recognition of the same direction and the same picture pixels, the results are not as expected, except for the jitter part of the results and the pixel discrimination model to reach a highly consistent conclusion.

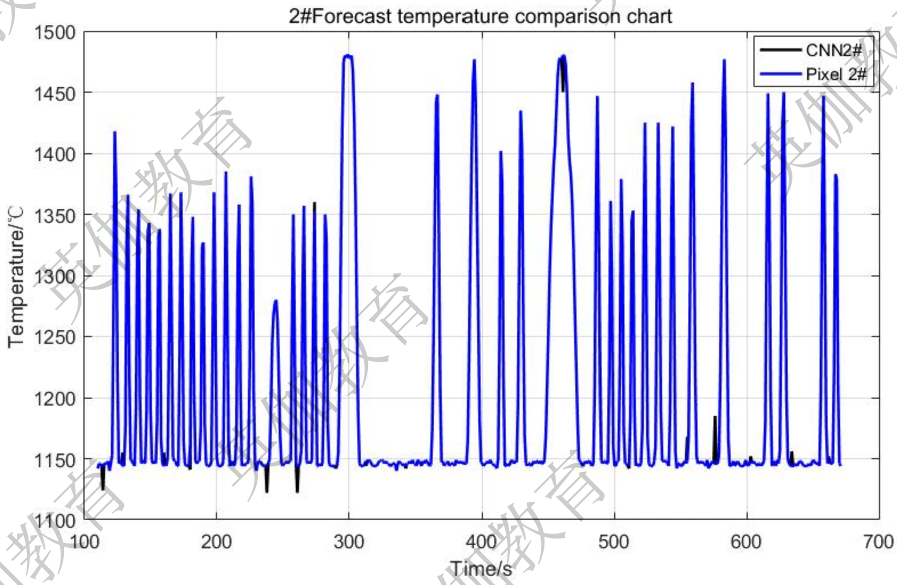

Figure.8 CNN vs Pixel discrimination for 2# line temperature time comparison

Similarly, in the temperature comparison of thermocouple No. 2, the convolutional neural network identified data that was highly consistent with the pixel recognition model except for the dithering part, and thus the pixel discrimination model could pass the test of the convolutional neural network set up in this paper, and thus the pixel discrimination method was chosen appropriately for this paper, so the pixel discrimination model was established.

5.1.5 Short summary

Problem 1 requires the development of a digital image recognition model to automatically extract temperature-related data from images and output them, visualising the output as 1# and 2# lines of time temperature profiles. For this problem, a pixel discrimination model was developed, firstly, the image was pre-processed, the image was greyed out and binarised, the image was turned into a black or white binary image, then based on the location and size of the digits on each image, the data was cut out and enlarged for subsequent preparation; then, each digit on the image was cut into a 158*93 pixel size image, and the size of the digits was uniformly Next, the image is filtered and noise-reduced to make the edges of the numbers smoother; then, the correlation coefficient of each image data to be recognized is calculated with 11 samples, and the two images with the largest correlation coefficients are selected and output numerically, and finally the data is visualized and processed to draw the temperature image; finally, this paper uses the established convolutional neural network-based digital Finally, this paper uses a digital image recognition model based on convolutional neural network to test and analyse the data, which is basically consistent with the conclusion reached by the pixel discrimination model except for the dithering part, so it shows that the model recognition is accurate.

5.2 Problem 2 Model Building and Solving

5.2.1 Preparation for model building

In this paper, 562 adjacent sequential curves of mould melting and crystallisation are greyscaled so that each pixel point has a value corresponding to the depth of the colour used to represent the melting and crystallisation process. The grey scale range is 0-255, where white is 255 and black is 0. The final image is scaled down to a uniform 300*400 pixels for uniform processing later. [3]

5.2.2 Feature-based equation of state modelling

5.2.2.1 Edge detection using the "Canny" operator

The Canny edge detection algorithm locates objects and boundaries in the image and determines the set of sub-regions or contour lines on the image.

Step1：First, an edge-enhanced image is obtained by smoothing a Gaussian filter and convolving the image. Gaussian filtering uses a two-dimensional Gaussian kernel to convolve with the image. This is because the digital image data is in the form of a discrete matrix and the Gaussian kernel is a discrete approximation to a continuous Gaussian function, derived by discrete sampling and normalisation of the Gaussian surface. The edge images obtained by Gaussian convolution also have some points with high gradient values, non-edges, which are a distraction to the true edges and should be removed. In this paper, the magnitude and direction of their gradients are calculated for each image using equation (3).

$$\begin{cases} M(x,y)=\sqrt{G_x(x,y)^2+G_y(x,y)^2} \\ \theta(x,y)=\arctan\left(\dfrac{G_x(x,y)}{G_y(x,y)}\right) \end{cases} \tag{3}$$

Step2： In order to make the blurred boundaries clear, this paper retains the extreme values of the gradient intensity at each pixel point and performs non-extreme suppression of the magnitude of the gradient, suppressing all gradient values except the local maxima to obtain the ideal gradient image. Then, a double-thresholding means is used to further eliminate the noise. In other words, an upper threshold and a lower threshold are set. Pixel points in the image that are larger than the upper threshold are considered to be boundaries, while those that are smaller than the lower threshold are considered to be not boundaries, and those in between are considered to be candidates and need further processing. [4]

Step3： The edges are traced in the processed image and the data contour in the top left corner is replaced with a 0 matrix to obtain the edge contour image (shown in Figure 8).

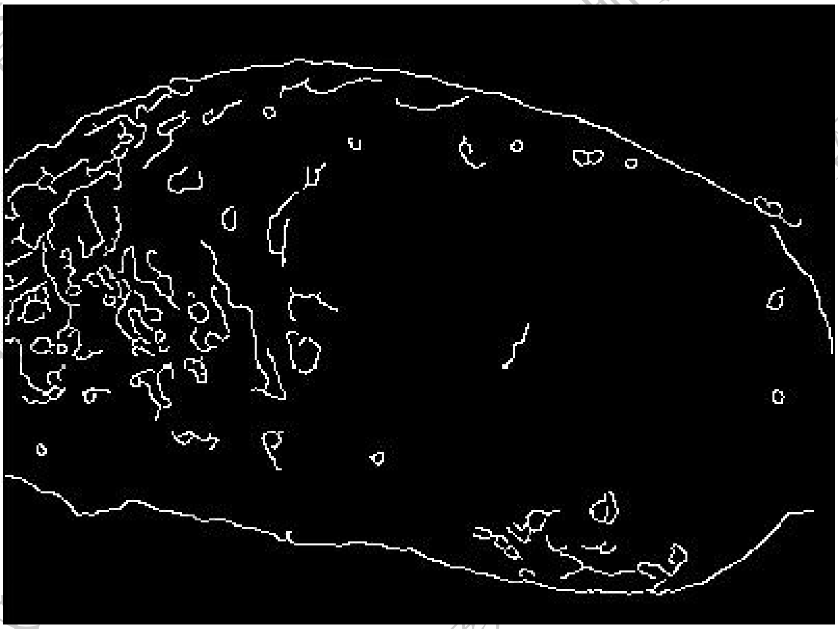

Figure.9 Edge profile at 110s

Step4： For edge contour images, the find function is used to obtain the index vector which separates the mould from the background in the grey-scale map. [5]

5.2.2.2 Feature extraction using Tamura textures

Tamura texture features are widely used to extract information from images, including roughness, contrast, directionality, linearity, regularity and coarseness.

❖ **Extraction of roughness features**

Roughness reflects the amount of granularity in the image texture, with a larger image roughness indicating a more granular image texture pattern and vice versa. The roughness feature extraction method is as follows.

Step1：Find the average gray value of each image element of the image, i.e.

$$C_k\left(x,y\right)=\frac{1}{2^{2i}}\sum_{i=x+2^{l-1}}^{x+2^{l-1}}\sum_{j=y+2^{l-1}}^{y+2^{l-1}}g\left(i,j\right) \tag{4}$$

where $C_k\left(x,y\right)$ is the average gray value of the cell, $g\left(i,j\right)$ is the gray value of the pixel located at $\left(i,j\right)$, and $l=0,1,\cdots,5,$ the active window size is $2^l\times 2^l$ pixels.

Step2： Calculate the average intensity difference between the unoverlapped windows in the horizontal and vertical directions for each pixel point separately, i.e.

$$E_{l,h}\left(x,y\right)=\left|C_l\left(x+2^{l-1},y\right)-C_l\left(x-2^{l-1},y\right)\right| \tag{5}$$

$$E_{l,v}\left(x,y\right)=\left|C_l\left(x,y+2^{l-1}\right)-C_l\left(x,y-2^{l-1}\right)\right| \tag{6}$$

Where,1 is the grey scale value of the pixel point located at that coordinate.

Step3: Calculation of roughness

$$T_{coa} = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} S_{best}\left(i,j\right) \tag{7}$$

where m and n are the length and width of the image, and the value of l is $E\left(x,y\right) = \max$, the optimal size of the setting window is $S_{best}\left(i,j\right) = 2^{l}$.

❖ **Extraction of contrast features**

Contrast is the degree of polarisation of light and darkness in an image and the dynamic range of the grey level, which reflects the depth of the texture grooves and the sharpness of the image. The higher the contrast ratio, the deeper the grooves and the clearer the image; the lower the contrast ratio, the shallower the grooves and the blurrier the image.

Contrast is calculated as

$$T_{con} = \sigma a_4^{-\frac{1}{4}} \tag{8}$$

Where , $a_4 = \frac{\mu_4}{\sigma_4}$, μ_4 is the fourth moment and σ is the variance.

❖ **Extraction of directionality features**

Directivity can reflect how concentrated or divergent an image texture is in a certain direction, and is related to the shape of texture primitives and rules for arrangement. A large degree of directivity indicates that the shape of the image is irregular and the distribution range of light and dark is large.

The directivity feature extraction method is as follows:

Step1： Convolution of the image and the operator of Equation (9) to obtain the gradient vector change ΔH in the horizontal direction and the change in the vertical direction ΔV , which can be expressed as:

$$\Delta H = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \Delta V = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \tag{9}$$

Step2：Calculate the gradient vector for each pixel, ie:

$$\theta = \tan^{-1}\left(\frac{\Delta V}{\Delta H}\right) + \frac{\pi}{2} \tag{10}$$

$$\left|\Delta G\left(x,y\right)\right| = \frac{1}{2}\left(\left|\Delta H\right| + \left|\Delta V\right|\right) \tag{11}$$

where $\Delta G\left(x,y\right)$ is the gradient vector at each pixel.

Step3: Calculate the histogram $H_D\left(l\right)$ of the direction of the construction vector, i.e.:

$$H_D\left(l\right)=\frac{N_\theta\left(l\right)}{\sum_{i=0}^{n-1}N_\theta\left(i\right)} \tag{12}$$

where n is the quantization level of the directional angle, and $N_\theta\left(i\right)$ is the total number of edge pixels on the corresponding directional angle θ ,when $\left|\Delta G\right|\le t$, $\frac{\left(2l-1\right)\pi}{2n}\le\theta\le\frac{\left(2l+1\right)\pi}{2n}$, the number of pixels is $N_\theta\left(l\right)$ .where ΔG is the gradient vector for each pixe， t is the threshold of the gradient vector modulus.For images with an unobvious direction, the value of $H_D\left(l\right)$ is relatively flat, and the image $H_D\left(l\right)$ with a more obvious opponent has a more obvious peak.

Step4：Calculate the directionality, i.e.:

$$T_{dir}=\sum_{p=1}^{n_p}\sum_{\varphi\in\omega_p}\left(\varphi-\varphi_p\right)^2 H_D\left(\varphi\right) \tag{13}$$

Where $\varphi=1,2,\cdots,n-1$, n_p is the number of histogram peaks, $p\left(p>0\right)$ is the peak of histogram $H_D\left(l\right)$, is the ω_p peak quantization range, and φ_p is the quantization value in the maximum histogram in ω_p .

❖ **Extraction of linearity features**

Linearity refers to the degree of deviation in the distance between pixels when the co-occurrence matrix is calculated. The larger the interval between pixels in the image, the greater the linearity and vice versa. The formula for calculating linearity is

$$T_{\text{lim}}=\frac{\sum_{i=1}^{m}\sum_{j=1}^{m}P\left(i,j\right)\cos\left[\left(i-j\right)\frac{2\pi}{n}\right]}{\sum_{i=1}^{m}\sum_{j=1}^{m}P_a\left(i,j\right)} \tag{14}$$

where P_a is the distance point of the $m\times m$ local directional symbiotic matrix.[6]

5.2.3 Solution and results of feature-based equation of state model

5.2.3.1 Crystallization process

Observing the image, it was found that the brightness of the image increased as the crystals grew. Therefore, it is speculated that the change of average gray value is affected by crystal growth, crystal growth, and gray scale decreases, so it can be concluded that gray scale is negatively correlated with crystal area. Assuming a linear relationship between the two and a scale factor of k, the crystallization growth at time t is

$$a_t=-k\Delta h_t+a_{t-1} \tag{15}$$

where $\Delta h_t=h_t-h_{t-1}$, h_t refers to the gray value of the image at time t.

At the same time, the influence of crystal growth on image roughness is also considered. Observing the curved image of roughness over time obtained by using the Tamura texture feature, it is found that the roughness is at a minimum value at 220s, 409s and 537s, and then gradually increases to increase the duration by about 60s. Therefore, it is speculated that the crystallization growth rate is slower at this time. If the coefficient of crystal growth and average gray value change is 0.5k, the relationship between crystal growth and gray value change can be obtained:

$$a_t = -0.5k\Delta h_t + a_{t-1} \tag{16}$$

From this, the recursive formula of the crystallization growth amount at different moments is obtained. Line plots are drawn by images of mean gray difference, and fitted lines are obtained by polynomial fitting, as shown in Figure 9.

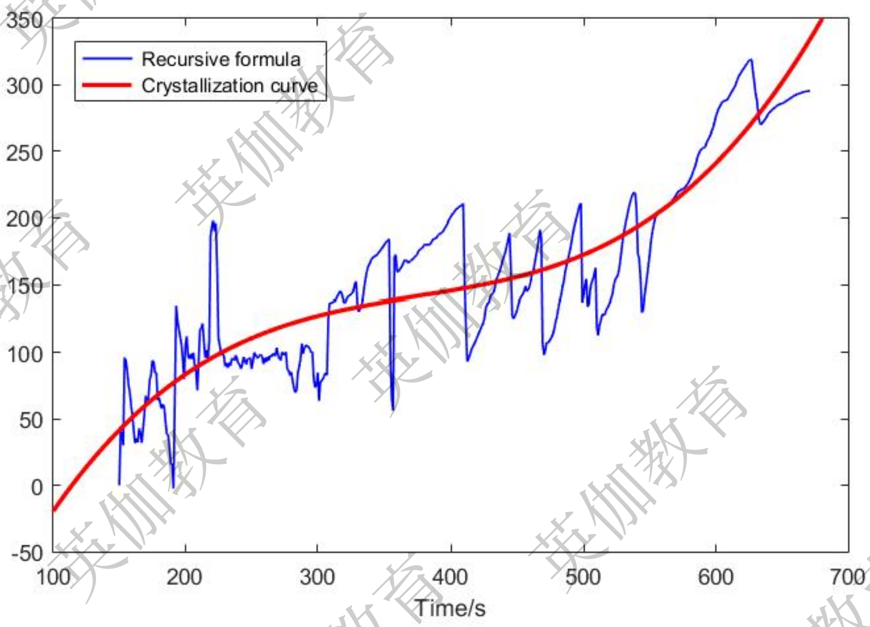

Figure.10 Crystal growth status vs. time plot

As shown in the figure, the blue line is the average gray value, and the red curve is the relationship between crystallization and time after polynomial fitting. It can be seen from the figure that the crystallization rate first increases and then slows down and then increases again with time, and because there is no crystallization result from 110s to 149s in the polynomial fitting image, the image is only the fitting result and has no practical physical significance.

5.2.3.2 Melting process

Looking at the Annex 1 image, as the mold melts, the brightness of the image changes from gray to red, and then from red to gold, and the average gray value also increases. Therefore, it is speculated that the average gray value is positively correlated with the melting amount, and the change of the average gray value increases with the increase of melting amount. At the same time, as the melting process progresses, the surface of the mold becomes smoother and smoother, that is, the roughness decreases. Therefore, it can be deduced that roughness and melting amount are negatively correlated, so assume that the relationship between melting amount and average gray value and roughness is:

$$b_t = \frac{h_t}{c_t} \tag{17}$$

where b_t is the melting amount of the mold at time t, h_t is the average gray value of the image at time t, and c_t is the roughness of the mold at time t. From this, the melting amount at different moments can be obtained, and then compared with the fitting curve, as shown in the figure below.

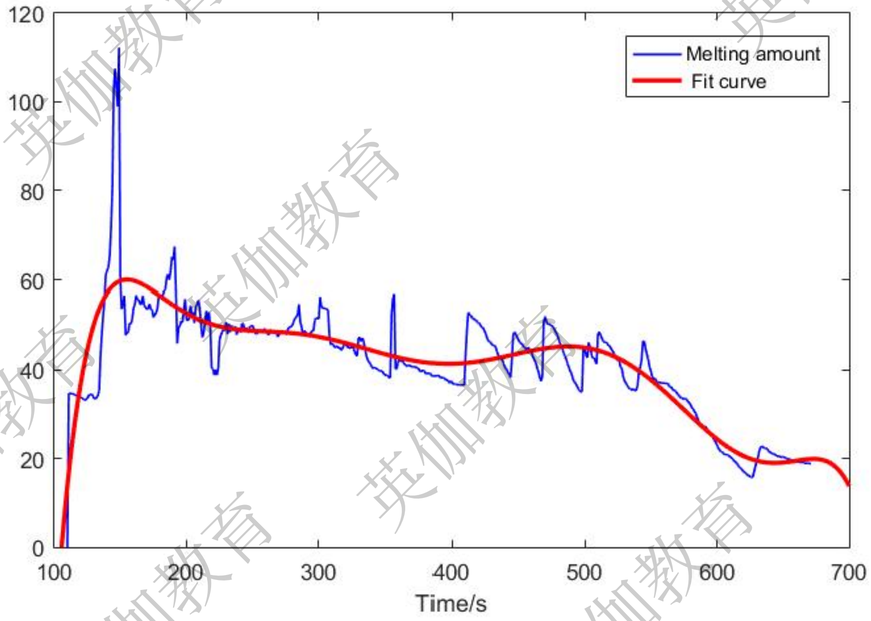

Figure.11 Melting quantity vs. time plot

As shown in the figure, the blue line is the melting amount, that is, the ratio of the average gray value to the roughness, and the relationship diagram with time. In this article, the graph is polynomically fitted, and the fitted line is the red line in the plot. It can be seen from the figure that the melting rate increases with the increase of temperature in the time period from 110s to 149s. [7]

5.2.3.3 Melt crystallization process curve

In this paper, the melting curve and crystallization curve are normalized, and the curve of their relationship with time is combined into one image for better observation. The image is as follows.

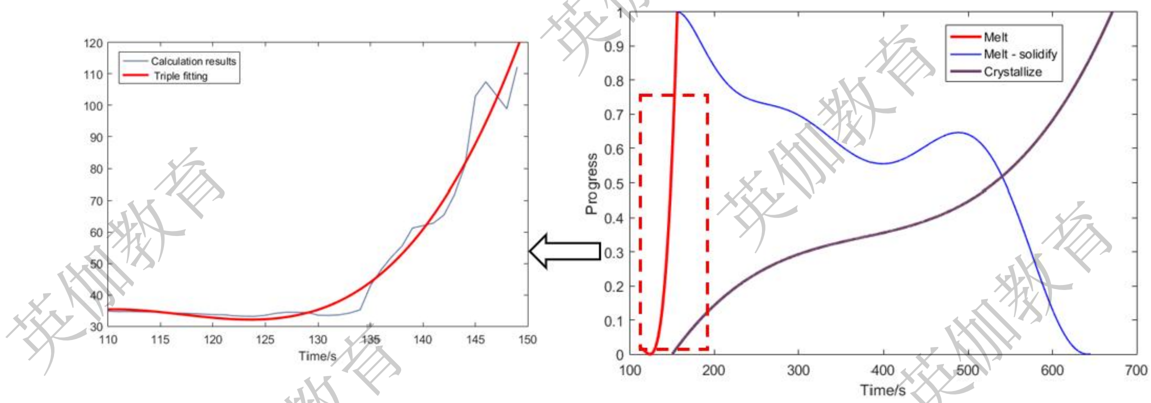

Figure.12 Melt crystallization process curve

As shown in the figure, the red line is the melting curve from 110s to 149s, the blue line is the polynomial fitting result, and the purple line is the result after crystal fitting. Due to the nature of the image conditions and the occlusion of the crystallization process, it is impossible to perform a specific and accurate analysis of the melting process after 150s, and the results of polynomial fitting are used in this paper, and the images before 150s are enlarged and presented for easy reading and grasping the melting characteristics. For the melting process, it is completely melted at 141s, at which point the melting progress reaches its maximum. The melting process has existed since then, but has been reduced relative to the melting progress at 141s. At the same time, the progress of crystallization continued to rise, and at the end of 671s, the maximum progress of crystallization was reached.

5.2.4 Test analysis

In this paper, it is found that roughness, contrast, directionality and linearity reflect the progress characteristics of melting and crystallization to varying degrees. First, this article analyzes the roughness, as shown in the following figure:

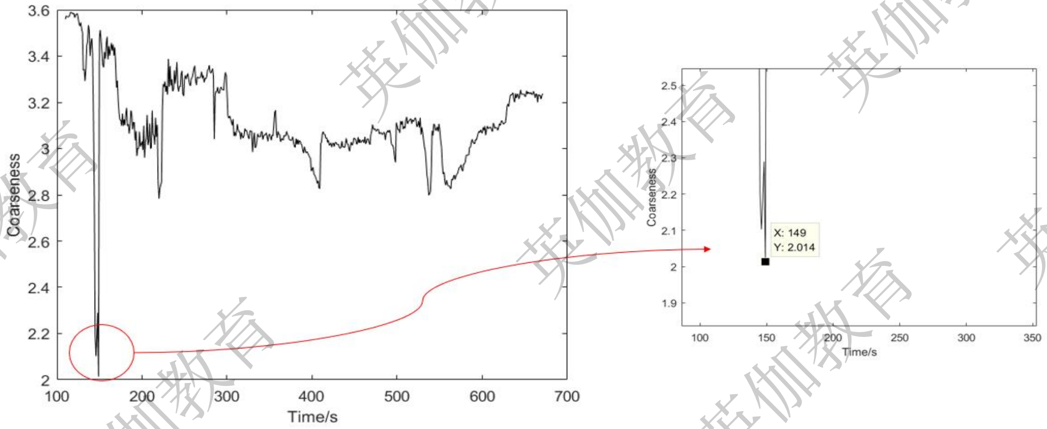

Figure.13 Roughness characteristic analysis diagram

For a detailed analysis of the lowest point of the drastic change in roughness image, it can be seen from the figure that the roughness lowest point time is 149s, when the image is the smoothest, that is, when the color image is the brightest. At this time, the corresponding sample has completely melted, and the steel has also melted, and the degree of melting of the sample is very high.

Then, the melting crystallization process is discussed in this paper with four characteristics: average gray value, linearity, contrast and directionality.

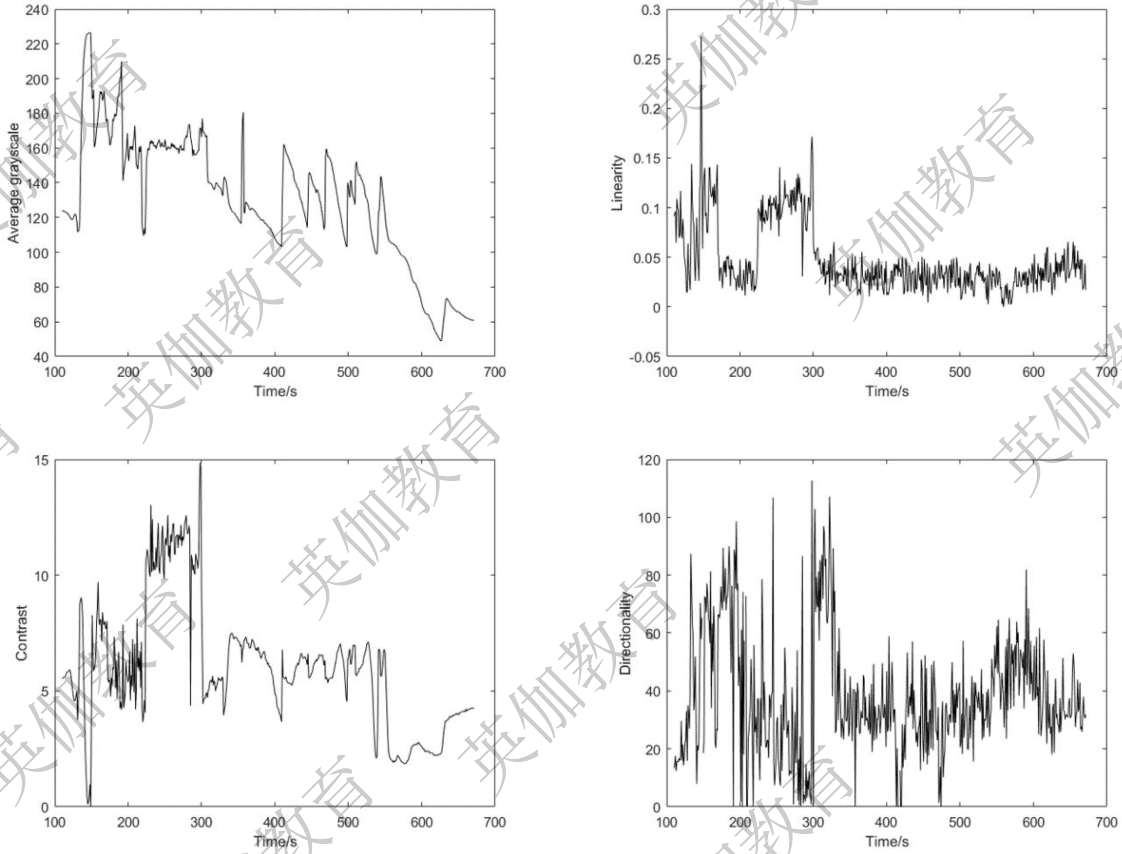

Figure.14 Four feature group diagrams

In the image above, the first row is the average gray value image and the linearity image, and the second row is the contrast image and the direction image, respectively. In this paper, when analyzing the four characteristics, the melting and crystallization are linked, and the image is used to test. In the process of 110s to 149s, due to the higher and higher temperature, the brightness of the picture is increasing, and the average grayscale map as a whole is on the rise. In the crystallization process, that is, after 150s, the average gray value of the image changes when there are crystal nuclei, and then the gray value decreases during the crystal growth process; As the temperature drops, more nuclei appear and the average gray value rises and falls repeatedly. For linearity images, contrast images and directionality images, this paper finds that the images change drastically at the junction of complete melting and crystallization, which shows the process of melting and crystallization, which proves that the conclusions obtained by the feature-based equation of state model are consistent with the experimental phenomenon, and verifies the reliability of the equation of state model of features.

5.2.5 Short summary

The second problem requires studying and quantifying the dynamic differences of images during mold melting and crystallization, establishing models of different image features according to time development, and discussing the change curves of mold melting and crystallization processes. For this question, a feature-based equation of state model is established. First, the picture is preprocessed, and after the grayscale processing of the picture, the image is uniformly reduced to 300*400 pixels according to the proportion. Then, a Gaussian smoothing filter is used to convolve the image, and then the image ladder amplitude is suppressed by a non-maximum value to make the image blurred boundary clearer. Then trace the boundary in the processed image to obtain the outline image, and use the find function on the outline image to separate the mold from the background. Then, the Tamura texture features are used to extract the four image features of roughness, contrast, directionality and linearity, and the change law of the four texture features and the recursive formula about time are found in the process of crystallization and melting, and the calculated data is visualized to obtain a line chart, and finally polynomial fitting is performed to obtain the process curve of melting and crystallization. In this paper, the five characteristics of roughness, contrast, directionality, linearity and average gray value are combined with images and experimental phenomena for examination and analysis, and it is found that the average gray value can obviously reflect the emergence of crystals in the crystallization process. The other four features changed drastically between 141s and 150s, which showed the process of melting and crystallization, which proved that the conclusions obtained by the feature-based equation of state model were consistent with the experimental phenomenon, and verified the reliability of the equation of state model of features.

5.3 Model establishment and solution of problem three

5.3.1 Establishment of polynomial regression model

Observing Figure 1, it can be seen that within the first 222 s, the temperature and time have a linear relationship; Within 222s-321s, the temperature is basically maintained at 1500°C unchanged; After 321s, the temperature has a linear relationship with time and shows a downward trend.

In the first 222 s, it can be seen from observation that the temperature increases linearly with time, so let

$$T = at + b \tag{18}$$

For a given data point $(t_i, T_i)\, i = 110, 111, \cdots, 222$, the primary formula $T = at + b$ is calculated so that the total error is minimized

$$Q = \sum_{i=110}^{222} \left(T - at + b\right)^2 \tag{19}$$

Find $a = 5.3408, b = 307.3435$. So from 110s to 222s, the function of time and temperature is $T = 5.3408t + 307.343$. In the same way, when $t \in (321, 671)$, the equation of time and temperature is $T = -1.9663t + 2135.7$.

In summary, the function of time and temperature is

$$T = \begin{cases} 5.3408t + 307.3435\left(110 \le t \le 222\right) \\ 1500\left(222 \le t \le 321\right) \\ -1.9663t + 2135.7\left(321 \le t \le 671\right) \end{cases} \tag{20}$$

According to the process curve of mold melting and crystallization obtained in question 2, the polynomial regression expression is established in the relationship between melting amount and temperature, and the relationship between crystallization quantity and temperature at the same time

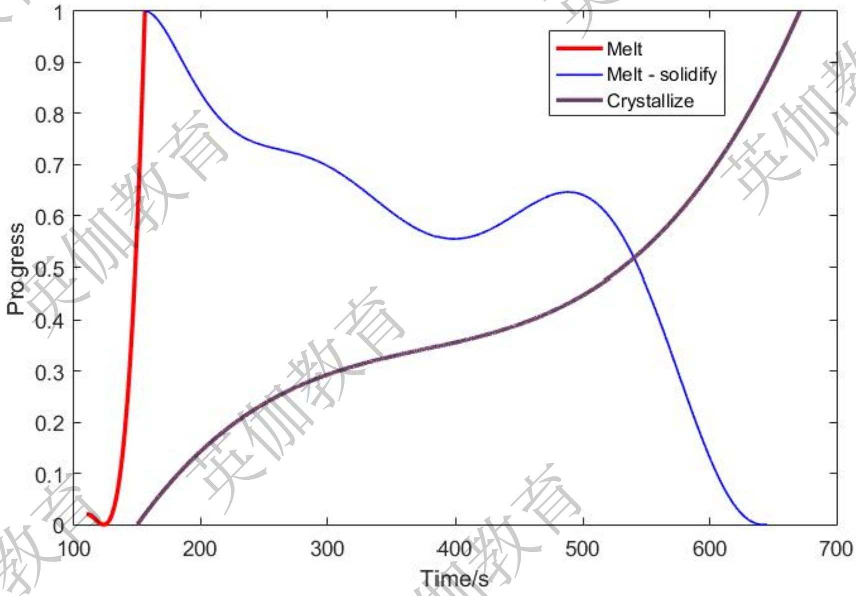

Figure.15 Diagram of melting and crystallization processes

Considering that the fitted results are inaccurate because the data points from 110s to 150s are fewer than the population, the data from 110s to 150s are fitted separately, and the fitting results are obtained as follows:

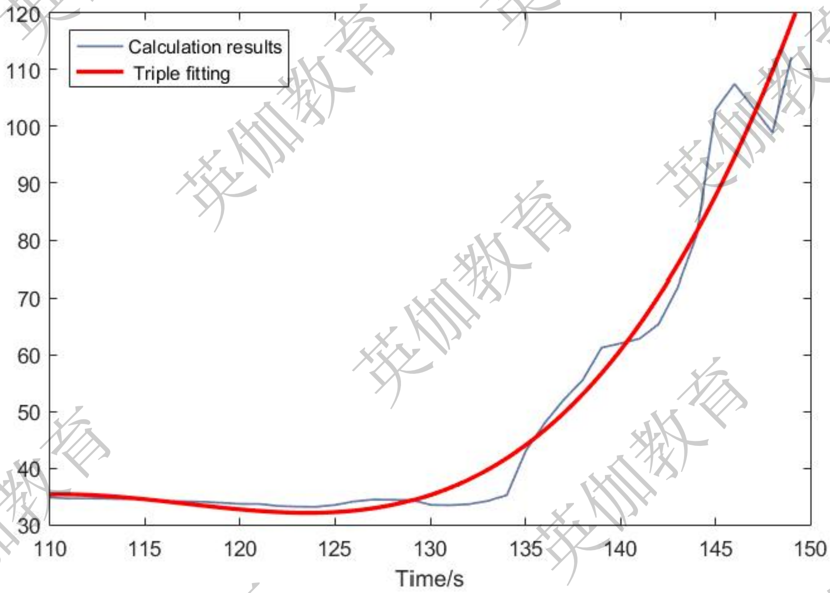

Figure.16 110s-150s melting status fitting diagram

It can be seen from the above figure that during the 110s-150s, the change trend of the melting process is basically the same as the fitting curve, and the difference is not much.

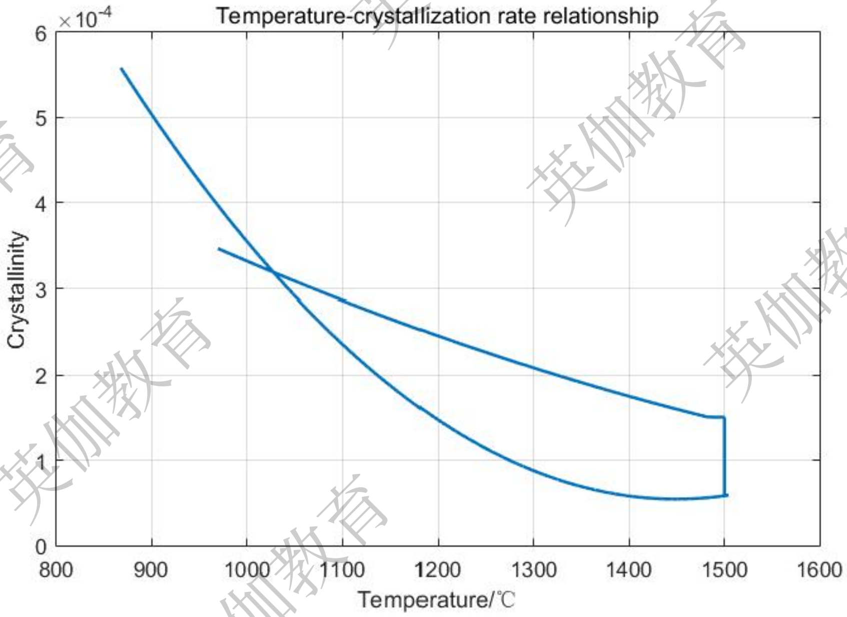

Figure.17 Temperature vs. crystallization plot

From the above figure, it can be found that the crystallization process can be divided into heating and cooling processes, and the two processes are fitted separately

$$\begin{aligned}&\text{Temperature rises：}\ m_i = 2.026\times 10^{-10}T_i^2 - 8.7885\times 10^{-7}T_i + 0.00100\,82\\&\text{Temperature drops：}\ m_i = 1.4897\times 10^{-10}T_i^2 - 4.3176\times 10^{-7}T_i + 0.0031\,828\end{aligned} \tag{22}$$

where T_i is the temperature at i -time and m_i is the crystallization rate at i -time.

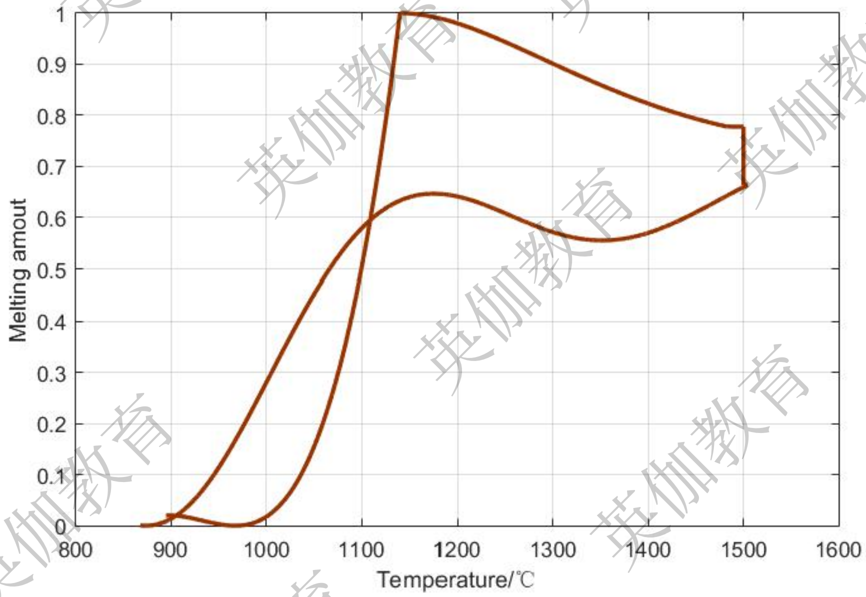

Figure.18 Melt-temperature diagram

As can be seen from the figure above, the relationship between melting amount and temperature reaches a maximum at 1140 °C, and then changes dramatically. Therefore, the two processes before and after 1140 °C are fitted as a function relationship, and the function relationship of the melting rate before 1140 °C is obtained：

$$n_i = 1.9273\times10^{-7}T_i^2 - 3.594\times10^{-4}T_i + 0.16751 \tag{23}$$

Before 1140 °C, the melting rate as a function of ：

$$n_i = -3.9813\times10^{-11}T_i^3 + 1.6791\times10^{-7}T_i^2 - 2.3449\times10^{-4}T_i + 0.10857 \tag{24}$$

Draw the function as an image：

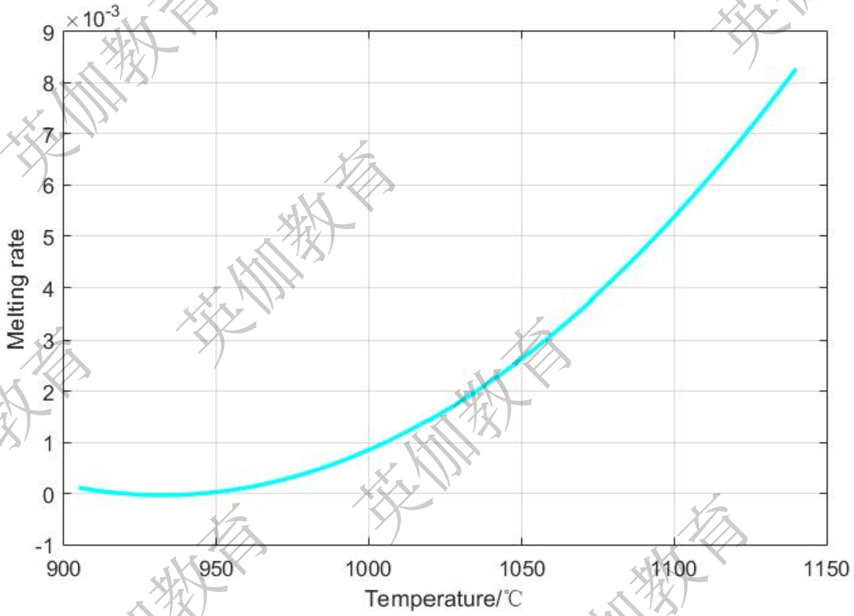

Figure.19 Melting rate-temperature plot below 1140°C

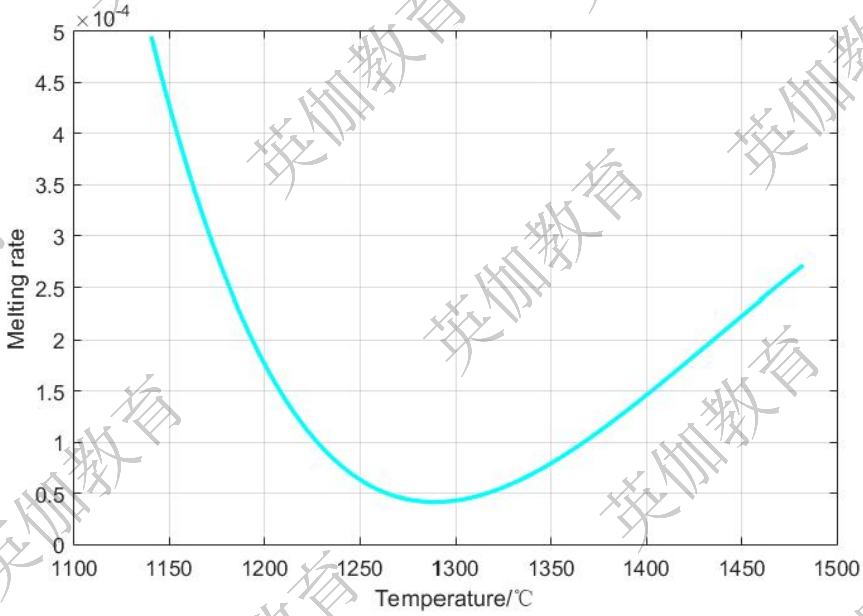

Figure.20 Melting rate-temperature plot above 1140°C

5.3.2 Test analysis

The crystallization rate can be multiplied by the proportion of the crystallization volume that has been crystallized and the total volume of the original liquid by the total crystallization time, assuming that the proportion of the crystallization volume that has been crystallized and the total volume of the original liquid is x, the total crystallization time is t, the crystallized liquid volume is V^s, the remaining uncrystallized liquid volume is V^t, the original total liquid volume is V, and the liquid easily reaches a new temperature, and the holding time at this temperature is τ. The number of crystallized particles in $\mathrm{d}\tau$ time is:

$$N_\tau = IV^t \,\mathrm{d}\tau \tag{25}$$

where I is the amount of crystallization produced per unit volume per unit of time. Let U be the rate at which a single grain interface produces crystals, and assume that the crystals grow at the same rate in all directions and that the grains are spherical. After the τ-time, the volume of the crystal produced in the total time t is

$$V_\tau^s = \frac{4\pi}{3}U^3\left(t-\tau\right)^3 \tag{26}$$

At the beginning of crystallization, $V^\tau \approx V$ is assumed, so at the t moment, the volume of liquid that has been crystallized is the crystal volume produced by τ in $\tau + \mathrm{d}t$ and the period, that is

$$\mathrm{d}V^s = N_\tau V_\tau^s \approx \frac{4\pi}{3}U^3\left(t-\tau\right)^3 \tag{27}$$

From this, it can be deduced that the proportion x of the crystallized crystal volume to the total volume of the original liquid is

$$x = \frac{V^s}{V} = \frac{4\pi}{3}\int_0^t IU^3\left(t-\tau\right)^3 \tag{28}$$

$$\mathrm{d}x = \frac{4\pi}{3}IU^3\left(t-\tau\right)^3 \mathrm{d}t \tag{29}$$

Given the interparticle collision and the correction factor $1-x$ of the reduction of the mother liquor, so

$$\mathrm{d}x = \left(1-x\right)\frac{4\pi}{3}IU^3\left(t-\tau\right)^3 \mathrm{d}t \tag{30}$$

Points are obtained

$$x = \exp\left(-\frac{\pi}{3}IU^3t^4\right) \tag{31}$$

Given the nucleation rate and growth rate over time, equation (24) can be collated：

$$x = 1 - \exp\left(-Kt^{n}\right) \tag{32}$$

where K is the crystallization rate constant and n is the Avrami index. Equation (28) is that the JMA formula[8] substituts the data from Annex II to obtain $K = 0.00038$, $t = t_i - 420, n = 3$, i.e：

$$x = 1 - \exp\left(-0.00038\left(t_i - 420\right)^{3}\right) \tag{33}$$

From the above equation, the relationship between temperature and crystallization rate can be plotted:

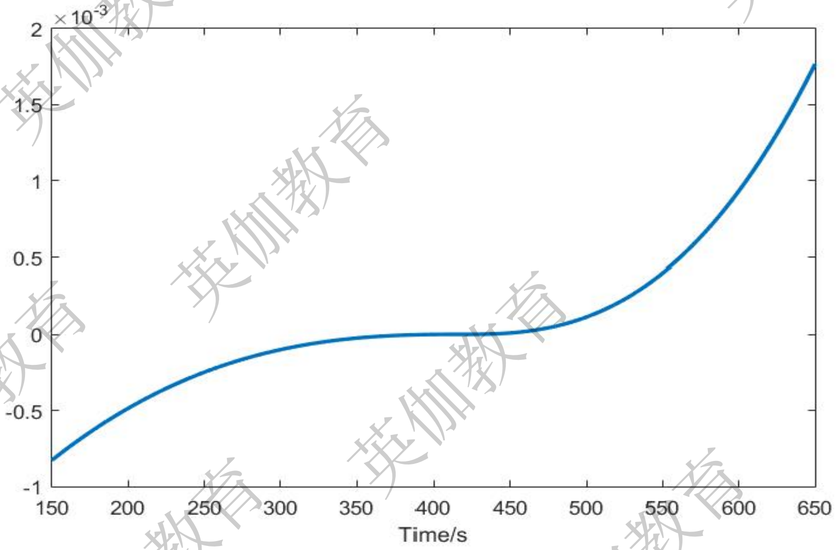

Figure.21 JMA formula crystallinity

Looking at the image, it can be found that Figure 14 is highly consistent with the image obtained in the second question, so it can be proved that the image obtained in the third question is accurate and reliable.

5.3.3 Short summary

Question 3 requires exploring temperature and time changes and the functional relationship between melting and crystallization, and discussing the melting and crystallization kinetics of melt fluxes. For this question, a polynomial regression model is established. Firstly, based on the temperature at different moments found in problem 1, polynomial fitting regression is carried out, so as to establish the time-temperature piecewise function. Then, according to the process curve of mold melting and crystallization obtained in question 2, the relationship between melting amount and temperature and the relationship between crystallization quantity and temperature at the same time are integrated, so as to establish a polynomial regression expression and draw an image. Through the change curve of mold crystallization with respect to time, the crystallization amount is derived to time, so as to determine the relationship between crystallinity and time, and the relationship between melting rate and time can be found in the same way. Finally, combined with the crystallization kinetics, it is found that the relationship between crystallinity and temperature obtained in this paper is highly consistent with the image of the JMA formula, which proves the accuracy and reliability of the obtained results.

6 Evaluation and promotion of models

6.1 Advantages of the model

(1)For image digital recognition, a pixel discrimination model is established, which segments the numbers in the image according to a fixed size, and uses the numbers in the image as the discriminant template to make the results of the model solution more accurate. By calculating the correlation coefficient of the pixel matrix, the template with the largest correlation coefficient is selected as the recognition result, which ensures the reliability of the model.

(2)For the extraction of image features, the Tamura algorithm is used to extract texture features as the basis for model establishment, which ensures the reliability of the model. By analyzing the relationship between the characteristics and mold melting and crystallization, the equation of state model is established to ensure the rationality of the model.

(3)In order to establish a functional relationship between temperature and time discrete data, we choose to use a polynomial regression model to establish it, which ensures the reliability of the functional relationship. Through the visual processing of data, the rationality of the function relationship is guaranteed.

6.2 Disadvantages of the model

(1)Although the pixel discrimination model has high accuracy, it is less flexible and can only recognize fixed images.

(2)Polynomial regression models establish complex functional relationships.

6.3 Promotion of the model

The pixel discrimination model has high accuracy and good recognition effect, and is suitable for identifying images with consistent formats, such as certificate recognition, icon recognition, and so on.

7 References

[1]陈群贤.TensorFlow 下基于 CNN 卷积神经网络的手写数字识别研究[J].信息记录材料,2022,23(09):

[2] Li Ziyang. Single-well lithofacies identification based on logging response and convolutional neural network[J]. Journal of Applied Geophysics,2022,207:68-75.

[3] 刘峰.基于图像边缘检测的绘画机器人轨迹规划研究[D].辽宁：大连理工大学,2021.10.

[4] 赵娅岐.基于 Canny 算子的边缘检测算法研究改进与电路实现[D].湖北：华中

科技大学，2021.
[5] Jadwaa Sana'a Khudayer.X-Ray Lung Image Classification Using a Canny Edge Detector[J].Journal of Electrical and Computer Engineering,2022.
[6] Taguchi Toshio. The post-war rebirth of Yokohama: the planner Akira Tamura's contributions to municipal reform[J]. Planning Perspectives,2022,37(05):1073-1095.
[7] Xiao Zikang. Quantitative Evaluation of Reservoir Heterogeneity in the Ordos Basin Based on Tamura Texture Features[J]. Frontiers in Earth Science.2022.
[8] Liu Hui. Erratum to "Light nuclei production in Au+Au collisions at [formula omitted] = 5–200 GeV from JAM model"[J].Physics Letters,2022,829.

Appendix

Annex I

The code for problem one

```
clc
clear all
I=imread('D:\MATLAB\2022 亚太\Attachment 1\0110.bmp');
I1=rgb2gray(I);%灰度图像处理
I2=im2bw(I1,graythresh(I1));%二值图像处理
I3=I2(1:65,1:110);
I3=imresize(I3,10);
TT=I3(208:366,780:873);
ED_type='disk';
se=strel(ED_type,8);
TT=imopen(TT,se);
TT=imresize(TT,[28,28]);
TT=imcomplement(TT);
%%imwrite(TT,'a.jpg');
fulldate=dir('D:\MATLAB\2022 亚太\Attachment 1\*.bmp');
the_length=length(fulldate);
% for i = 1 : the_length
% image=fulldate(i).name;
% image=fullfile('D:\MATLAB\2022 亚太\Attachment 1',image);
% image=imread(image);
% I1=rgb2gray(image);%灰度图像处理
% I2=im2bw(I1,graythresh(I1));%二值图像处理
% I3=I2(1:65,1:110);
% I3=imresize(I3,10);%对左上角图像分割
% T1=I3(208:366,501:873);%1 号热电偶温度
% T2=I3(393:551,501:873);%2 号热电偶温度
% se=strel(ED_type,8);
% T1=imopen(T1,se);
% T2=imopen(T2,se);
% T1=imresize(T1,3);
% T2=imresize(T2,3);%开运算降噪
% for j = 1:4
% left=1+279*(j-1);
% right=279*j;
% JJ1=T1(36:460,left:right);
% JJ1=imresize(JJ1,[28,28]);
% JJ1=imcomplement(JJ1);
% JJ2=T2(36:460,left:right);
% JJ2=imresize(JJ2,[28,28]);
% JJ2=imcomplement(JJ2);
% imwrite(JJ1,strcat('new1#',num2str(j),fulldate(i).name));
% imwrite(JJ2,strcat('new2#',num2str(j),fulldate(i).name));
% end
% end
%%模板
% N=dir('D:\MATLAB\2022 亚太\数字模板\*.png');
% the_length1=length(N);
% for i = 1 : the_length1
% image=N(i).name;
% image=fullfile('D:\MATLAB\2022 亚太\数字模板',image);
% image=imread(image);
% I1=rgb2gray(image);%灰度图像处理
% I2=im2bw(I1,graythresh(I1));%二值图像处理
% I3=imresize(I2,[28,28]);%对左上角图像分割
% I3=imcomplement(I3);
% j=i-1;
% imwrite(I3,strcat(num2str(j),'.jpg'));
% end
%%方法一模板
image0=fullfile('D:\MATLAB\2022 亚太\数字\new1#20572.bmp');
image0=imread(image0);
image1=fullfile('D:\MATLAB\2022 亚太\数字\new1#10132.bmp');
image1=imread(image1);
image2=fullfile('D:\MATLAB\2022 亚太\数字\new1#20177.bmp');
image2=imread(image2);
image3=fullfile('D:\MATLAB\2022 亚太\数字\new1#20189.bmp');
image3=imread(image3);
image4=fullfile('D:\MATLAB\2022 亚太\数字\new1#20203.bmp');
image4=imread(image4);
image5=fullfile('D:\MATLAB\2022 亚太\数字\new1#20251.bmp');
image5=imread(image5);
image6=fullfile('D:\MATLAB\2022 亚太\数字\new1#20594.bmp');
image6=imread(image6);
image7=fullfile('D:\MATLAB\2022 亚太\数字\new1#20589.bmp');
image7=imread(image7);
image8=fullfile('D:\MATLAB\2022 亚太\数字\new1#10645.bmp');
image8=imread(image8);
image9=fullfile('D:\MATLAB\2022 亚太\数字\new1#10110.bmp');
image9=imread(image9);
% imwrite(image0,'0.jpg');
% imwrite(image1,'1.jpg');
% imwrite(image2,'2.jpg');
% imwrite(image3,'3.jpg');
% imwrite(image4,'4.jpg');
% imwrite(image5,'5.jpg');
% imwrite(image6,'6.jpg');
% imwrite(image7,'7.jpg');
% imwrite(image8,'8.jpg');
% imwrite(image9,'9.jpg');
K{1}=image0;
K{2}=image1;
K{3}=image2;
K{4}=image3;
K{5}=image4;
K{6}=image5;
K{7}=image6;
K{8}=image7;
K{9}=image8;
K{10}=image9;
K{11}=TT;
%%比较相关系数
Z=[];
R=[];
for i = 1 : the_length
for j = 1:4
n=i+109;
image = strcat('D:\MATLAB\2022 亚太\数字
\new1#',num2str(j),'0',num2str(n),'.bmp');
Image = strcat('D:\MATLAB\2022 亚太\数字
\new2#',num2str(j),'0',num2str(n),'.bmp');
image = imread(image);
Image = imread(Image);
for k = 1 : 11
Z(k)=corr2(K{k},image);
Z2(k)=corr2(K{k},Image);
end
if Z(k)<=1
R(i,j)=find(Z==max(Z))-1;
else
R(i,j)=-1;
end
if Z2(k)<=1
R2(i,j)=find(Z2==max(Z2))-1;
else
R2(i,j)=-1;
end
end
end
```

```
%%找出没有图像的图片
wrong=[];
for i = 1 : the_length
if R(i,1)==-1
wrong=[wrong,i+109];
end
end
wrong = wrong';
% for i = 1 : length(wrong)
% image_path = strcat('D:\MATLAB\2022 亚太\Attachment
1\','0',num2str(wrong(i)),'.bmp');
% image = imread(image_path);
% image = rgb2gray(image);
% image=im2bw(image,graythresh(I1));%二值图像处理
% image=image(1:65,1:110);
% image=imresize(image,10);
% TT1=I3(208:366,501:873);%1 号热电偶温度
% TT2=I3(393:551,501:873);%2 号热电偶温度
% se=strel(ED_type,8);
% TT1=imopen(TT1,se);
% TT2=imopen(TT2,se);
% TT1=imresize(TT1,3);
% TT2=imresize(TT2,3);%开运算降噪
% for j = 1:4
% left=1+279*(j-1);
% right=279*j;
% JJ1=TT1(36:460,left:right);
% JJ1=imresize(JJ1,[28,28]);
% JJ1=imcomplement(JJ1);
% JJ2=TT2(36:460,left:right);
% JJ2=imresize(JJ2,[28,28]);
% JJ2=imcomplement(JJ2);
% imwrite(JJ1,strcat('new1#',num2str(j),'0',num2str(wrong(i)),'.bmp'));
% imwrite(JJ2,strcat('new2#',num2str(j),'0',num2str(wrong(i)),'.bmp'));
% end
% end
for i = 1 : length(wrong)
for j = 1 : 4
image_path = strcat('D:\MATLAB\2022 亚太\数字
\','new1#',num2str(j),'0',num2str(wrong(i)),'.bmp');
image = imread(image_path);
for k = 1 : 11
Z(k)=corr2(K{k},image);
end
```

```
R(wrong(i)-109,j)=find(Z==max(Z))-1
end
end
%%热电偶温度变化
for i = 1 : the_length
if R(i,4)~=10
wendu_1(i)=R(i,1)*1000+R(i,2)*100+R(i,3)*10+R(i,4);
else
wendu_1(i)=R(i,1)*100+R(i,2)*10+R(i,3);
end
if R2(i,4)~=10
wendu_2(i)=R2(i,1)*1000+R2(i,2)*100+R2(i,3)*10+R2(i,4);
else
wendu_2(i)=R2(i,1)*1000+R2(i,2)*100+R2(i,3)*10+R2(i,4)-9;
end
end
wendu_1=wendu_1';
wendu_2=wendu_2';
%%校准
for i = 1 : the_length
if wendu_1(i)<300
wendu_1(i)=wendu_1(i)*10+1;
end
if wendu_2(i)<300
wendu_2(i)=wendu_2(i)*10+1
end
end
% for i = 1 : the_length-1
% if abs(wendu_1(i+1)-wendu_1(i))>200
% wendu_1(i+1)=wendu_1(i);
% end
% if abs(wendu_2(i+1)-wendu_2(i))>200
% wendu_2(i+1)=wendu_2(i);
% end
% end
%%数据存入表 2
t=110:the_length+109;
t=t';
% xlswrite('Attachment 2.xlsx',t,'sheet1','B2:B563');
% xlswrite('Attachment 2.xlsx',wendu_1,'sheet1','C2:C563');
% xlswrite('Attachment 2.xlsx',wendu_2,'sheet1','D2:D563');
wendu_1=xlsread('Attachment 2.xlsx','C2:C563');
wendu_2=xlsread('Attachment 2.xlsx','D2:D563');
wendu_1(321,1)=1291;
```

```
%%画图
%%picture1
figure
%plot(t,wendu_1,'b',t,wendu_2,'r');
plot(t,wendu_1,'b','linewidth',1.5);
hold on;
plot(t,wendu_2,'r','linewidth',1.5);
grid on;
xlabel('Time/s');
ylabel('Temperature/℃');
title('1#wire temperature-2#wire temperature-time diagram');
%%picture2
pj_wendu_1=ones(length(wendu_1),1).*sum(wendu_1)/length(wendu_1);
pj_wendu_2=ones(length(wendu_2),1).*sum(wendu_2)/length(wendu_2);
figure
%plot(t,pj_wendu_1,'b',t,pj_wendu_2,'r');
plot(t,pj_wendu_1,'b','linewidth',1.5);
hold on;
plot(t,pj_wendu_2,'r','linewidth',1.5);
grid on;
xlabel('Time/s');
ylabel('Temperature/℃');
title('1#wire average temperature-2#wire average temperature-time
diagram');
%%compare
cnn_1=xlsread('D:\MATLAB\2022 亚太\温度.xls','B1:B562');
cnn_2=xlsread('D:\MATLAB\2022 亚太\温度.xls','C1:C562');
%%1#
figure
%plot(t,wendu_1,'b',t,cnn_1,'r');
plot(t,cnn_1,'k','linewidth',1.5);
hold on;
plot(t,wendu_1,'b','linewidth',1.5);
grid on;
xlabel('Time/s');
ylabel('Temperature/℃');
title('1#Forecast temperature comparison chart');
%%2#
figure
%plot(t,wendu_2,'b',t,cnn_2,'r');
plot(t,cnn_2,'k','linewidth',1.5);
hold on;
plot(t,wendu_2,'b','linewidth',1.5);
grid on;
```

```
xlabel('Time/s');
ylabel('Temperature/℃');
title('2#Forecast temperature comparison chart');
```

CNN

```
clc
clear all
close all


%%import data
digitDatasetPath = fullfile('E:\Matlab\matlab
apply\bin\shouxieshibie\shouxieshibie','/handwrite/');
imds = imageDatastore(digitDatasetPath,...
    'IncludeSubfolders',true,'LabelSource','foldernames');%The folder
name is used as the data tag


%%Number of data set graphs
countEachLabel(imds)


numTrainFiles = 17;%Each number has 22 samples, and 17 samples are taken
as training data
[imdsTrain,imdsValidation] =
splitEachLabel(imds,numTrainFiles,'randomized');
%View the size of the image
img = readimage(imds,1);
size(img)


%%Define the structure of convolutional neural network
layers = [
%Layer of input
imageInputLayer([28 28 1])


%Layer of convolution
convolution2dLayer(5,6,'Padding',2)
batchNormalizationLayer
reluLayer
maxPooling2dLayer(2,'stride',2)


convolution2dLayer(5,16)
batchNormalizationLayer
reluLayer


maxPooling2dLayer(2,'stride',2)


convolution2dLayer(5,120)
batchNormalizationLayer
reluLayer


%Final layer
fullyConnectedLayer(11)
softmaxLayer
classificationLayer];


%%Trained neural network


%%Setting training Parameters
options = trainingOptions('sgdm',...
    'MaxEpochs',50,...
    'ValidationData',imdsValidation,...
    'ValidationFrequency',5,...
    'Verbose',false,...
    'Plots','training-progress');%Shows the progress of the training


%Train the neural network, save the network
net = trainNetwork(imdsTrain,layers,options);
save 'CNNet.mat' net
%load('CNNet.mat')


%%Tagged data
mineSet = imageDatastore('E:\Matlab\matlab apply\bin\手写识别\手写识别
\hw22\hw22', 'IncludeSubfolders',false,'FileExtensions','.jpg' );
mLabels = cell(size(mineSet.Files,1),1);
for i = 1:size(mineSet.Files,1)
[filepath,name,ext] = fileparts(char(mineSet.Files{i}));
```

```
mLabels{i,1} = char(name);
end
mLabels2 = categorical(mLabels);
mineSet.Labels = mLabels2;




%%Use the network to classify and calculate accuracy
%Handwritten data
YPred = classify(net,mineSet);
YValidation = mineSet.Labels;
%Rate of accuracy of calculation
accuracy = sum(YPred == YValidation)/numel(YValidation)
%Plot the forecast results
figure;
nSample = 11;
ind = randperm(size(YPred,1),nSample);
for i = 1:nSample

subplot(2,fix((nSample+1)/2),i)
imshow(char(mineSet.Files(ind(i))))
title(['Forecast: ' char(YPred(ind(i)))])
if char(YPred(ind(i))) ==char(YValidation(ind(i)))
    xlabel(['True:' char(YValidation(ind(i)))],'Color','b')
else
    xlabel(['True:' char(YValidation(ind(i)))],'color','r')
end


end
```

Annex II

The code for question two

```
clc
clear all
close all
I=dir('D:\MATLAB\2022 亚太\Attachment 1\*.bmp');
the_length=length(I);
%%灰度处理
% T=[];
% for i = 1 : the_length
%     image=I(i).name;
%      image=fullfile('D:\MATLAB\2022 亚太\Attachment 1',image);
%      image=imread(image);
%      I1=rgb2gray(image);%灰度图像处理
%      imwrite(I1,strcat(num2str(0),num2str(i+109),'.bmp'));
% end

%%灰度差计算
% for i = 1 : the_length-1
%    image=I(i).name;
%    image1=I(i+1).name;
%    image=fullfile('D:\MATLAB\2022 亚太\灰度图像',image);
%    image1=fullfile('D:\MATLAB\2022 亚太\灰度图像',image1);
%    image=imread(image);
%    image1=imread(image1);
%    II=image1-image;
%    T(i)=sum(II(:));
% end
 t=110:109+the_length;
% x=110.5:108.5+the_length;
% figure
% plot(x,T);
% W=[];
% for i = 1 : length(T)
%  if T(i)>30000000
%      n=i+109;
%      W=[W,n];
%  end
% end
%调用举例：
%image=rgb2gray(imread('example.jpg'));
%f=coarseness(image,5)
for i = 1 : the_length
    image=I(i).name;
    image=fullfile('D:\MATLAB\2022 亚太\灰度图像',image);
    image=imread(image);
    image=imresize(image,[300,400]);
    BW1=edge(image,'canny',0.2);
    Z=zeros(30,30);
    BW1(1:30,1:30)=Z;
    %imwrite(BW1,strcat('lk',num2str(i+109),'.bmp'));
end
imshow(image)
for i = 1 : the_length
    image=I(i).name;
    image=fullfile('D:\MATLAB\2022亚太\灰度图像',image);
    image=imread(image);
    image=imresize(image,[300,400]);
    image=image(105:180,112:225);
    x(i)=mean2(image);
    image=double(image);
    y(i)=std(image(:));
%    Fcon=contrast(image)
%   [Fdir,sita]=directionality(image);
%     Flin=linelikeness(image,sita,4);
     Fcrs = coarseness(image,2 );
     cucao(i)=Fcrs;
%    duibi(i)=Fcon;
%    fangxiang(i)=Fdir;
%    sita_1{i}=sita;
%    xianxing(i)=Flin;
end
figure
%粗糙度
plot(t,cucao,'k')%,t,duibi,'b',t,fangxiang,'r');
xlabel('Time/s')
ylabel('Coarseness');
%对比度
plot(t,duibi,'k')%,t,duibi,'b',t,fangxiang,'r');
xlabel('Time/s')
ylabel('Contrast');
%平均灰度
plot(t,x,'k');
xlabel('Time/s');
ylabel('Average grayscale');
for i = 1:the_length-1
  xx(i)=x(i+1)-x(i);
  yy(i)=y(i+1)-y(i);
end
%平均灰度差
plot(t(1:end-1),xx,'k');
xlabel('Time/s');
ylabel('Average gray difference');
%线性度
plot(t,xianxing,'k');
xlabel('Time/s');
ylabel('Linearity');
%标准差
plot(t,y,'k');
```

```
xlabel('Time/s');
ylabel('standard deviation');
%标准差的差
plot(t(1:end-1),yy,'k');
xlabel('Time/s');
ylabel('');
%方向度
plot(t,fangxiang,'k');
xlabel('Time/s');
ylabel('');

function Fcrs = coarseness( graypic,kmax )%graphic 为待处理的灰度图像，2^kmax
为最大窗口
[h,w]=size(graypic); %获取图片大小
A=zeros(h,w,2^kmax); %平均灰度值矩阵 A
%计算有效可计算范围内每个点的 2^k 邻域内的平均灰度值
for i=2^(kmax-1)+1:h-2^(kmax-1)
    for j=2^(kmax-1)+1:w-2^(kmax-1)
        for k=1:kmax

A(i,j,k)=mean2(graypic(i-2^(k-1):i+2^(k-1)-1,j-2^(k-1):j+2^(k-1)-1));
        end
    end
end
%对每个像素点计算在水平和垂直方向上不重叠窗口之间的 Ak 差
for i=1+2^(kmax-1):h-2^(kmax-1)
    for j=1+2^(kmax-1):w-2^(kmax-1)
        for k=1:kmax
            Eh(i,j,k)=abs(A(i+2^(k-1),j,k)-A(i-2^(k-1),j));
            Ev(i,j,k)=abs(A(i,j+2^(k-1),k)-A(i,j-2^(k-1)));
        end
    end
end
%对每个像素点计算使 E 达到最大值的 k
for i=2^(kmax-1)+1:h-2^(kmax-1)
    for j=2^(kmax-1)+1:w-2^(kmax-1)
        [maxEh,p]=max(Eh(i,j,:));
        [maxEv,q]=max(Ev(i,j,:));
        if maxEh>maxEv
            maxkk=p;
        else
            maxkk=q;
        end
        Sbest(i,j)=2^maxkk; %每个像素点的最优窗口大小为 2^maxkk
    end
end
%所有 Sbest 的均值作为整幅图片的粗糙度
Fcrs=mean2(Sbest);
end


function Fcon=contrast(graypic) %graypic 为待处理的灰度图片
graypic=double(graypic);
x=graypic(:); %二维向量一维化
M4=mean((x-mean(x)).^4); %四阶矩
delta2=var(x,1); %方差
alfa4=M4/(delta2^2); %峰度
delta=std(x,1); %标准差
Fcon=delta/(alfa4^(1/4)); %对比度
end


function [Fdir,sita]=directionality(graypic)
[h w]=size(graypic);
%两个方向的卷积矩阵
GradientH=[-1 0 1;-1 0 1;-1 0 1];
GradientV=[ 1 1 1;0 0 0;-1 -1 -1];
%卷积，取有效结果矩阵
MHconv=conv2(graypic,GradientH);
MH=MHconv(3:h,3:w);
MVconv=conv2(graypic,GradientV);
MV=MVconv(3:h,3:w)
%向量模
MG=(abs(MH)+abs(MV))./2;
%有效矩阵大小
validH=h-2;
validW=w-2
%各像素点的方向
for i=1:validH
    for j=1:validW
        sita(i,j)=atan(MV(i,j)/MH(i,j))+(pi/2);
    end
end
n=16;
t=12;
Nsita=zeros(1,n);
%构造方向的统计直方图
for i=1:validH
    for j=1:validW
```

```
        for k=1:n
            if sita(i,j)>=(2*(k-1)*pi/2/n) &&
sita(i,j)<((2*(k-1)+1)*pi/2/n) && MG(i,j)>=t
                Nsita(k)=Nsita(k)+1;
            end
        end
    end
end
for k=1:n
    HD(k)=Nsita(k)/sum(Nsita(:));
end

[maxvalue,FIp]=max(HD);
Fdir=0;
for k=1:n
    Fdir=Fdir+(k-FIp)^2*HD(k);
end
end


function Flin=linelikeness(graypic,sita,d) %d 为共生矩阵计算时的像素间隔距离
n=16;
[h,w]=size(graypic);
%构造方向共生矩阵
PDd1=zeros(n,n);
PDd2=zeros(n,n);
PDd3=zeros(n,n);
PDd4=zeros(n,n);
PDd5=zeros(n,n);
PDd6=zeros(n,n);
PDd7=zeros(n,n);
PDd8=zeros(n,n);
for i=d+1:h-d-2
    for j=d+1:w-d-2
        for m1=1:n
            for m2=1:n
                %下方向
                if (sita(i,j)>=(2*(m1-1)*pi/2/n) &&
sita(i,j)<((2*(m1-1)+1)*pi/2/n)) && (sita(i+d,j)>=(2*(m2-1)*pi/2/n) &&
sita(i+d,j)<((2*(m2-1)+1)*pi/2/n))
                    PDd1(m1,m2)=PDd1(m1,m2)+1;
                end
                %上方向
                if (sita(i,j)>=(2*(m1-1)*pi/2/n) &&
sita(i,j)<((2*(m1-1)+1)*pi/2/n)) && (sita(i-d,j)>=(2*(m2-1)*pi/2/n) &&
sita(i-d,j)<((2*(m2-1)+1)*pi/2/n))
                    PDd2(m1,m2)=PDd2(m1,m2)+1;
                end
                %右方向
                if (sita(i,j)>=(2*(m1-1)*pi/2/n) &&
sita(i,j)<((2*(m1-1)+1)*pi/2/n)) && (sita(i,j+d)>=(2*(m2-1)*pi/2/n) &&
sita(i,j+d)<((2*(m2-1)+1)*pi/2/n))
                    PDd3(m1,m2)=PDd3(m1,m2)+1;
                end
                %左方向
                if (sita(i,j)>=(2*(m1-1)*pi/2/n) &&
sita(i,j)<((2*(m1-1)+1)*pi/2/n)) && (sita(i,j-d)>=(2*(m2-1)*pi/2/n) &&
sita(i,j-d)<((2*(m2-1)+1)*pi/2/n))
                    PDd4(m1,m2)=PDd4(m1,m2)+1;
                end
                %右下方向
                if (sita(i,j)>=(2*(m1-1)*pi/2/n) &&
sita(i,j)<((2*(m1-1)+1)*pi/2/n)) && (sita(i+d,j+d)>=(2*(m2-1)*pi/2/n) &&
sita(i+d,j+d)<((2*(m2-1)+1)*pi/2/n))
                    PDd5(m1,m2)=PDd5(m1,m2)+1;
                end
                %右上方向
                if (sita(i,j)>=(2*(m1-1)*pi/2/n) &&
sita(i,j)<((2*(m1-1)+1)*pi/2/n)) && (sita(i-d,j+d)>=(2*(m2-1)*pi/2/n) &&
sita(i-d,j+d)<((2*(m2-1)+1)*pi/2/n))
                    PDd6(m1,m2)=PDd6(m1,m2)+1;
                end
                %左下方向
                if (sita(i,j)>=(2*(m1-1)*pi/2/n) &&
sita(i,j)<((2*(m1-1)+1)*pi/2/n)) && (sita(i+d,j-d)>=(2*(m2-1)*pi/2/n) &&
sita(i+d,j-d)<((2*(m2-1)+1)*pi/2/n))
                    PDd7(m1,m2)=PDd7(m1,m2)+1;
                end
                %左上方向
                if (sita(i,j)>=(2*(m1-1)*pi/2/n) &&
sita(i,j)<((2*(m1-1)+1)*pi/2/n)) && (sita(i-d,j-d)>=(2*(m2-1)*pi/2/n) &&
sita(i-d,j-d)<((2*(m2-1)+1)*pi/2/n))
                    PDd8(m1,m2)=PDd8(m1,m2)+1;
                end
            end
        end
    end
```

```
end
f=zeros(1,8);
g=zeros(1,8);
for i=1:n
    for j=1:n
        f(1)=f(1)+PDd1(i,j)*cos((i-j)*2*pi/n);
        g(1)=g(1)+PDd1(i,j);
        f(2)=f(2)+PDd2(i,j)*cos((i-j)*2*pi/n);
        g(2)=g(2)+PDd2(i,j);
        f(3)=f(3)+PDd3(i,j)*cos((i-j)*2*pi/n);
        g(3)=g(3)+PDd3(i,j);
        f(4)=f(4)+PDd4(i,j)*cos((i-j)*2*pi/n);
        g(4)=g(4)+PDd4(i,j);
        f(5)=f(5)+PDd5(i,j)*cos((i-j)*2*pi/n);
        g(5)=g(5)+PDd5(i,j);
        f(6)=f(6)+PDd6(i,j)*cos((i-j)*2*pi/n);
        g(6)=g(6)+PDd6(i,j);
        f(7)=f(7)+PDd7(i,j)*cos((i-j)*2*pi/n);
        g(7)=g(7)+PDd7(i,j);
        f(8)=f(8)+PDd8(i,j)*cos((i-j)*2*pi/n);
        g(8)=g(4)+PDd8(i,j);
    end
end
tempM=f./g;
Flin=max(tempM);%取 8 个方向的线性度最大值作为图片的线性度
End
```

Annex III

Code for question three

```
%%question 3
k=1;a=[];
for i = 1 : 521
a(1)=0;
if i==71||260||388
a(i+1)=-k*xx(i+40)+a(i);
else
a(i+1)=-2*k*xx(i+40)+a(i);
end
end
plot(t(41:end),a,'r');
xlabel('Time/s');

b=[];
```

```
for i = 1 : 562
    b(i)=x(i)/cucao(i);
end
plot(t,b,'r');
xlabel('Time/s');
plot(t(1:40),b(1:40),'r');
t1=150:0.1:671
y1=polyval(fit1.coeff,t1);
plot(t1,y1)
[X,PS] = mapminmax(y1,0,1);

%局部熔化拟合
t3=110:0.1:156;
y3=polyval(fit5.coeff,t3);
plot(t3,y3)
Y2= mapminmax(y3,0,1);
plot(t3,Y2)
%熔化拟合
t2=110:0.1:645
y2=polyval(fit2.coeff,t2);
plot(t2,y2);
[Y,PS] = mapminmax(y2,0,1);
plot(t2,Y)
[m,n]=find(Y(:)==1);
%第二问最终图
figure
n=length(t3);
plot(t3,Y2,'r',t2(n:end),Y(n:end),'b')
hold on
plot(t1,X,'g');
grid on
%温度图
T=xlsread('Attachment 2.xlsx','C2:C563');
plot(t,T)
T=@(m)(5.3408*m+307.3435)*(m<=220&&m>=110)+1500*(m>220&&m<=321)+(-1.966
3*m+2135.7)*(m>321&&m<=671);
m=110:0.1:671;
for i = 1:length(m)
    Tt(i)=T(m(i));
end
% xlswrite('Tt',Tt(:),'sheet1','A1:A5611');
% xlswrite('Tt',t1(:),'sheet1','B1:B5211');
% xlswrite('Tt',X(:),'sheet1','C1:C5211');
% xlswrite('Tt',t2(:),'sheet1','D1:D5351');
% xlswrite('Tt',Y(:),'sheet1','E1:E5351');

%结晶率与温度图
plot(Tt(141:5351),X);
dX=diff(X,1);
plot(Tt(141:5350),dX);
grid on

%%升温结晶率与温度
plot(Tt(141:1102),dX(1:962));
grid on
%%降温结晶率与温度
plot(Tt(2134:5350),dX(1994:5210));
grid on

%%熔化温度图
Y(1:461)=Y2(:);
plot(Tt(1:5351),Y);
grid on

plot(Tt(1:1102),Y3);

dY1=diff(Y,1);
dY2=dY1(461:1101)+0.0005;
plot(Tt(461:1101),dY2);
grid on
dY3=dY1(20:460)+0.0002;
plot(Tt(20:460),dY3);
grid on

tt1=120:0.1:150;
yy1=polyval(fit5.coeff,tt1);
figure
plot(tt1,yy1);
a=0.00038;
b=420;
c=2.6;
yy2=@(tt2)1-exp(-(a*(tt2-b))^c);
tt2=150:650;
yy3=[];
for i = 1 : length(tt2)
    yy3(i)=yy2(tt2(i));
end
%yy3=yy3.*10000;
```

```
%yy3=yy3-yy3(1);
%yy3= mapminmax(yy3,0,1);
plot(tt2,yy3);
```