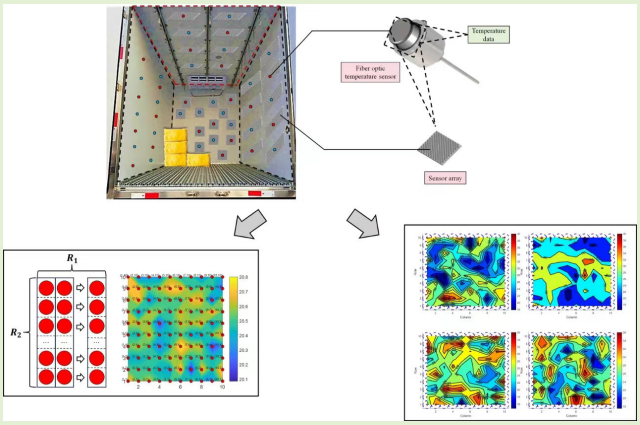
A Temperature Monitoring Method for Sensor Arrays Based on Temperature Mapping and Improved Mask R-CNN

Zuoxun Wang, Chuanyu Cui, Jinxue Sui, Yong Zhang, and Changkun Guo

***Abstract*—Temperature monitoring of sensor arrays is indispensable for ensuring the stable operation of the entire sensor system. This article presents a novel method (TISM) for sensor array temperature monitoring based on temperature mapping and an enhanced Mask region-convolutional neural network (R-CNN) framework. Initially, the method establishes a robust mapping correlation between sensor temperature data and spatial coordinates, thereby facilitating precise data acquisition and strategic rule formulation through a temperature qualification protocol. Subsequently, employing subarray analysis, the temperature data are structured into a matrix and transformed into a temperature heat map. The thermal image is further refined using interpolation techniques to enhance the accuracy and stability of the monitoring system. Additionally, an improved Mask R-CNN model is proposed, enabling effective target recognition and feature extraction from the temperature thermogram, thereby facilitating the extraction of temperature state information. Ultimately, sensor temperature states are determined based on color discrepancy and temperature mapping, thus achieving the objective of sensor array temperature monitoring. The method was compared with artificial neural network temperature prediction (ANNTM), phase-shifted grating, and photoelectric oscillation temperature monitoring (MPTM). Comparison indicators include comprehensive temperature prediction effect, accuracy, stability, and monitoring range. Notably, the proposed method attains a prediction accuracy of 97.13%, showcasing substantial improvements over ANNTM in terms of mean deviation and standard deviation by 25.89% and 1.91%, respectively. Furthermore, compared to MPTM's limited monitoring range of 490 °C-495 °C, the proposed method offers a significantly broader monitoring scope. Moreover, in terms of integrated temperature prediction for the sensor array, the proposed approach exhibits superior performance with smaller prediction errors, closely aligning with actual temperature values. Experimental validation corroborates the effectiveness of the proposed method, thereby underscoring its promising potential for real-time temperature monitoring of sensor arrays in practical applications.**



***Index Terms*— Color discrepancy, improved Mask region-convolutional neural network (R-CNN), smoothing, subarrays, temperature heatmap, temperature mapping.**

I. Introduction

SENSOR arrays play a pivotal role in various domains of modern technological advancements [1], [2], [3], [4], [5], encompassing key sectors such as industrial automation, intelligent robotics, healthcare, environmental monitoring, and the Internet of Things (IoT). These arrays are structured to furnish comprehensive information support for decision-making and system optimization by deploying multiple sensors across diverse locations, thus offering efficient solutions for smart development. Notably, temperature monitoring within sensor arrays stands out as a fundamental technology within sensor applications. It serves as the cornerstone for ensuring the stable operation of sensor arrays, enhancing system safety, and advancing equipment intelligence. In recent years, both domestic and international scholars have proposed numerous innovative methods and models to advance the field of sensor array temperature monitoring.

Manuscript received 27 March 2024; accepted 10 May 2024. Date of publication 12 June 2024; date of current version 1 August 2024. The work was funded by the project of Yantai City Science and Technology Innovation to help the transformation of old and new kinetic energy. Fund NO: KXDNY2023-25. The associate editor coordinating the review of this article and approving it for publication was Dr. Muhammad A. Jamshed. *(Corresponding author: Zuoxun Wang.)*

The authors are with the School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai, Shandong 264005, China (e-mail: wangzuoxun@126.com; cuichuanyu1999@163.com; suijx@sdtbu.edu.cn; zzxd@sdtbu.edu.cn; youdianerbaqi@163.com).

Digital Object Identifier 10.1109/JSEN.2024.3401123

Wei et al. [6] proposed a temperature monitoring technology based on artificial neural networks (ANNs). It utilizes average sea surface temperature (SST) and anomalous data SST to train convolutional neural networks and integrates deep neural networks to reduce temperature-monitoring errors. However, this method requires a large amount of labeled data and exhibits poor monitoring robustness in the presence of temperature perturbations and high temperature bursts, as well as limited adaptation to dynamically changing temperature environments. In contrast, the method presented in this article establishes a mapping relationship between temperature data and location information to provide accurate data and rule strategies, thus circumventing the need for extensive labeled training data and enhancing monitoring robustness and sensor fault judgment accuracy. Tu et al. [7] introduced a temperature monitoring system with phase-shifting grating and photoelectric oscillation. It has the characteristics of high sensitivity, noncontact, and real-time performance. However, it is restricted to high-temperature monitoring, with a limited range between 490 °C and 495 °C. In contrast, the method proposed in this article offers a wider temperature monitoring range and broader applicability. Verma et al. [8] and Thiyagarajan et al. [9] introduced feasible methods to monitor human body temperature through sensors. Although the flexible, wearable skin temperature-monitoring sensor proposed in the article offers advantages such as single-use and simple preparation, it suffers from long response times and risks accuracy degradation in high-temperature environments. Moreover, its applicability is limited to human body temperature monitoring and cannot be extended to sensor array temperature monitoring. Lee et al. [10] proposed a wafer temperature monitoring system based on a surface acoustic wave (SAW) temperature sensor array. However, it fails to consider sensor position information and target identification. In contrast, this article proposes a monitoring method that integrates location information, achieving higher prediction accuracy and a wider monitoring range through model improvements. Peng et al. [11] proposed a technology using fiber optic sensors for temperature monitoring of lithium-ion batteries. The technology has high sensitivity and accurate temperature monitoring. Nevertheless, issues such as restricted installation locations, poor stability, and high production costs are noted. In contrast, the method proposed in this article realizes more accurate monitoring with a wider monitoring range and places greater emphasis on long-term stability, enhancing practicality. Chen et al. [12], Atallah et al. [13], Badar et al. [14], and Caputo et al. [15] also gave different temperature monitoring methods.

Most of the above methods use sensors or sensor arrays to implement temperature-monitoring applications and apply them in a certain field. The temperature monitoring of the sensor or sensor array itself is not involved, so it is easy to cause misjudgment. At the same time, their monitoring effects will also be restricted by the different placement positions of the sensor arrays. A single location layout cannot balance the relationship between location and temperature, affecting monitoring accuracy. Robustness and monitoring range still need to be improved.

To overcome the above shortcomings, in this work, a sensor array temperature monitoring method based on temperature mapping and improved Mask region-convolutional neural network (R-CNN) is proposed. This method starts from the sensor array itself and focuses on its own temperature monitoring. Overcome the limitations of a single layout with multipoint monitoring. Taking the temperature monitoring inside the automatic loading and unloading truck as the application scenario, the technology upgrade is completed because the temperature inside the carriage is restricted by the temperature required by the cargo. If the goods are fresh meat or dairy products, the temperature of the carriage should generally be controlled below 0 °C. Therefore, real-time monitoring of the temperature inside the vehicle compartment is crucial. However, due to the complexity of the vehicle compartment structure, the monitoring effect of the sensor array is severely restricted by its location. Therefore, traditional monitoring methods are not suitable for applications in this field. This method overcomes the above shortcomings and achieves accurate monitoring of the temperature of the vehicle compartment through the layout of the optical fiber temperature sensor array on the inner wall of the vehicle compartment. It not only solves location constraints but also improves monitoring effects.

Fiber optic temperature sensors use optical principles to determine the ambient temperature by measuring changes in optical signals in optical fibers at different temperatures. Because it is soft and bendable and the optical signal transmission speed is very fast, it can adapt to the complex structure inside the car, and the data can be quickly transmitted to the central processing unit for analysis. Therefore, choosing an optical fiber temperature sensor is an important guarantee for providing real-time temperature monitoring. The fiber type of the sensor is single-core multimode, and the fiber core is 62.5 μm. The optical fiber coating is 125 μm, and the bending radius is greater than 60 mm. The pulling force under dynamic conditions reaches 200 N, and the pulling force under static conditions is 100 N. The fiber diameter is 3 mm. The prediction accuracy can reach $\pm$0.5 °C of full scale. Its temperature monitoring range can reach between $-$400 °C and 1500 °C. When the temperature is lower than $-$400 °C, the optical fiber system will turn on the light-emitting diode (LED) to make the fluorescent temperature measurement system work; therefore, its temperature measurement range is wider. When multiple optical fiber temperature sensors are arranged in an array on the inner wall of a truck compartment, they can simultaneously monitor the temperature at different locations in the compartment and transmit the data to the central processing unit. Through analysis and integration of these data, the interior of the compartment can be monitored and controlled in real-time temperature status.

The actual monitoring application diagram is shown in Fig. 1.

The main contributions of this work are as follows.

1) A mapping relationship between temperature data and location information is established. Combine temperature data with sensor location to provide accurate data information and rule policies.

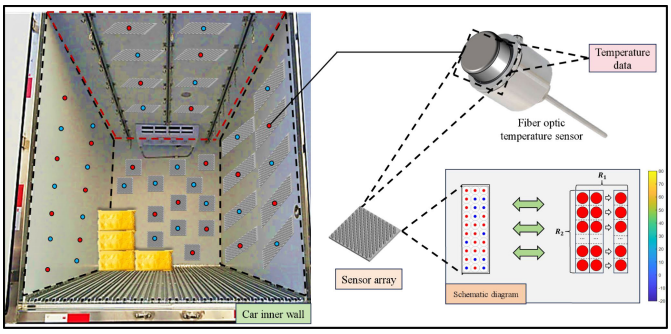


Fig. 1. Actual monitoring application diagram.

2) A monitoring method that comprehensively considers sensor location information is proposed. Through sensor subarray expansion and temperature heat map analysis, the balance of position and temperature is completed with multipoint monitoring.

3) An improved Mask R-CNN model is designed. Target recognition and feature extraction of specific areas are achieved to form a differential temperature image. This method of temperature mapping combined with target recognition brings new ideas and solutions to the field of sensor array temperature monitoring.

This article is organized as follows.

Step 1: Establishment of a sensor array, acquisition of temperature data, and position information of the sensor array. Subsequently, a mapping relationship between the two and a temperature-mapping rule are established.

Step 2: Creation of sensor subarrays. The temperature data is matrixed and transformed into a temperature heat map through matrix analysis and library function conversion. Smoothing of the temperature heat map is achieved via interpolation.

Step 3: Expansion of the sensor subarray to cover the entire sensor array. The same operations as Step 2 are performed to obtain a smoothed temperature thermogram.

Step 4: Improved Mask R-CNN model (DLM Mask R-CNN). The smoothed temperature thermogram serves as the input layer for target recognition and feature extraction in specific regions, resulting in a differential temperature image.

Step 5: Calculation of the color difference degree of the differential temperature image using the Euclidean distance in the Commission Internationale d'Eclairage (CIE) Lab* color space.

Step 6: Filtering of the color difference degree according to the established temperature mapping rules. Sensor determination of normal and abnormal temperatures in the sensor array is accomplished based on the mapping relationship.

Step 7: Validation of the effectiveness of the proposed method through experimental analysis, comparing it with traditional temperature monitoring methods.

These steps collectively form a comprehensive approach for temperature monitoring of sensor arrays, ensuring the stable operation of the sensor array and the overall system.

II. Related Work

A. Information Settings and Temperature Mapping Rules

A sensor array is an ensemble and system of multiple sensors working together. It overcomes the drawbacks of data limitations of individual sensors and increases the dimensions of observation applications. Ultimately, it realizes high-precision and high-resolution environmental sensing and data acquisition. This article takes the scale 10×10 sensor plane array as the research object. In-depth technological innovation to complete the technological upgrading to realize the temperature-monitoring task of the sensor array. Set up the sensor array matrix, which is used to represent each sensor element. Sensor $= [s_{1,1}, \ldots, \ldots; s_{1,10}; \ldots s_{10,1}, \ldots s_{1,10}]$ defines the position information of each sensor element. The position matrix is denoted as $P_{\text{sensor}} = [p_{1,1}, \ldots, p_{1,10}; \cdots; p_{10,1}, \ldots, p_{10,10}]$. Setting the initial temperature data matrix of the sensor array: $T_{\text{sensor}} = [T_{1,1}, \ldots, T_{1,10}; \cdots; T_{10,1}, \ldots, T_{10,10}]$. Where sensor, P_{sensor} T_{sensor} are all of size 10×10. If the temperature of the sensor array changes, then there is $T_{1,1} \Rightarrow T_{1,1} \pm \Delta T_{1,1}, T_{1,2} \Rightarrow T_{1,2} \pm \Delta T_{1,2}, \ldots\ldots, T_{10,10} \Rightarrow T_{10,10} \pm \Delta T_{10,10}$. The update rule is as follows:

$$Q_s^b \Rightarrow \begin{cases} T_{1,1} = T_{1,1} \pm \Delta T_{1,1} \\ T_{1,2} = T_{1,2} \pm \Delta T_{1,2} \\ \ldots\ldots \\ T_{10,10} = T_{10,10} \pm \Delta T_{10,10} \end{cases} \tag{1}$$

where Q_s^b is the temperature data matrix after a temperature change occurs in the sensor array. $T_{1,1}, T_{1,2}, \ldots, T_{10,10}$ is the temperature of each sensor element before the temperature change. $\Delta T_{1,1}, \Delta T_{1,2}, \ldots, \Delta T_{10,10}$ is the amount of change corresponding to each sensor element. From (1), the updated temperature data are expressed as follows: $Q_s^b = [T_{1,1}, \ldots, T_{1,10}; \cdots; T_{10,1}, \ldots, T_{10,10}]$, where b represents the number of changes, and $\Delta T_{1,1}, \Delta T_{1,2}, \ldots, \Delta T_{10,10}$ indicates the amount of change each time. Therefore, their values are not fixed. Now, let us establish the mapping relationship between the sensor array position matrix P_{sensor} and the temperature matrix T_{sensor}. This relationship is expressed as a bijection relationship. According to the correspondence law f, there is a one-to-one correspondence between sensor array position and temperature, denoted as $f : \{P_{\text{sensor}}(\text{rand}, \text{rand}) \Leftrightarrow T_{\text{sensor}}(\text{rand}, \text{rand})\}$, where rand is an integer satisfying $1 \leq \text{rand} \leq 10$. Typical sensors operate in the temperature range of [−40 °C, 85 °C]. Some high-temperature-resistant sensors can reach a higher temperature tolerance range, such as improved fiber Bragg grating temperature sensor (PS-FBG) temperature sensors constituting the temperature monitoring system, which achieves a temperature monitoring range between 490 °C and 495 °C. We establish the temperature setting range of the sensor array, denoted as $[k_1, k_2]$. The amount of each temperature change is T_{change}, and the maximum threshold of temperature change is $|T_{\max}|$. Therefore, the normal temperature change of the sensor array should satisfy the following two conditions.

1) The magnitude after each temperature change should always be within the range of $[k_1, k_2]$. In other words, $k_1 \leq T_{\text{rand,rand}} \leq k_2$.

2) The magnitude of each temperature change should be less than the maximum threshold for temperature change. That is, $T_{\text{change}} < |T_{\max}|$.

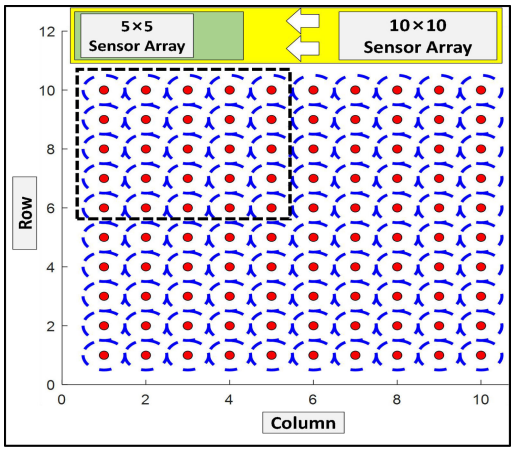


Fig. 2. 5 × 5 sensor subarrays.

Then the remaining cases that do not satisfy both conditions 1) and 2) are recognized as temperature anomaly states. Based on the above information, we set up the position information and temperature data of the sensor array. The mapping relationship between the two is established. The temperature mapping rules are defined.

B. Analysis of Sensor Subarrays

A subset of the original sensor array is selected, consisting of a 5 × 5 array of sensors. Rapid prototyping and testing are conducted on this smaller subset before implementing the model on the entire sensor array. This approach not only reduces computational burden but also ensures correct functionality and expected outcomes. Subsequently, the methodology is scaled up to larger sensor arrays. The design of the subarray is illustrated in Fig. 2. The initial temperature of this subarray is set. Since the primary focus of this article is on the temperature monitoring method rather than temperature acquisition, the temperature dataset used in this study is artificially set. Six sets of temperature data are collected from the subarray.

Now let $Q_s^0 = T_{\text{sensor}}$. Q_s^0 is equivalent to the initial temperature data matrix T_{sensor}. The remaining five subtemperature matrices are denoted as Q_s^1, Q_s^2, Q_s^3, Q_s^4, Q_s^5. Defined

$$E_H^0 = \begin{bmatrix} e_{1\cdot} \\ \vdots \\ e_{5\cdot} \end{bmatrix} \Rightarrow \begin{bmatrix} \sum_{i=1}^{5} e_{1i} - \min(e_{1\cdot}) \\ \vdots \\ \sum_{i=1}^{5} e_{5i} - \min(e_{5\cdot}) \end{bmatrix} \tag{2}$$

where E_H^0 is the primary pressure drop matrix. $e_{1\cdot}, \ldots, e_{5\cdot}$ are all row vectors with one row and five columns. i is a constant. e_{1i} is the ith element in $e_{1\cdot}$, and $\min(e_{1\cdot})$ is the smallest element in $e_{1\cdot}$. Define the secondary pressure drop matrix

$$E_H^1 = \begin{bmatrix} e_{1\cdot} \\ \vdots \\ e_{5\cdot} \end{bmatrix} \Rightarrow \begin{bmatrix} \sum_{i=1}^{5} e_{1i} \cdot (\max(e_{1\cdot}) - \min(e_{1\cdot})) \\ \vdots \\ \sum_{i=1}^{5} e_{5i} \cdot (\max(e_{5\cdot}) - \min(e_{5\cdot})) \end{bmatrix} \tag{3}$$

where E_H^1 is the secondary voltage drop matrix. $\max(e_{1\cdot})$ is the largest element in e_1.

After the primary and secondary buckling process, at this point the buckling matrices E_H^0 and E_H^1 are both of size 5 × 1. They follow the following rules:

$$H = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_5 \end{bmatrix} \Rightarrow \begin{bmatrix} E_H^0(1,1)/E_H^1(1,1) \\ \vdots \\ E_H^0(5,1)/E_H^1(5,1) \end{bmatrix} \tag{4}$$

where H has size 5 × 1. $\alpha_1, \ldots, \alpha_5$ are all row vectors with one row and five columns. $E_H^0(1,1)$ is the element in row 1 and column 1 of E_H^0, that is, $E_H^0(1,1) = e_{1\cdot}$. $E_H^0(1,1)/E_H^1(1,1)$ is the ratio of $e_{1\cdot}$ in E_H^0 to the corresponding element of $e_{1\cdot}$ in E_H^1. Now define $E_H^0 = [e_{1\cdot}, \ldots, e_{5\cdot}]^T = [Q_s^0(1\cdot), \ldots, Q_s^0(5\cdot)]^T$. $E_H^1 = [e_{1\cdot}, \ldots, e_{5\cdot}]^T = [Q_s^0(1\cdot), \ldots, Q_s^0(5\cdot)]^T$. They become the new E_H^0 and E_H^1 after the buckling processes of (3) and (4). The final matrix formed by Q_s^0 after the buckling process is denoted as H_0

$$H_0 = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_5 \end{bmatrix} = \begin{bmatrix} E_H^0(1,1)/E_H^1(1,1) \\ \vdots \\ E_H^0(5,1)/E_H^1(5,1) \end{bmatrix} \tag{5}$$

where H_0 is a matrix of size 5 × 1. Similarly, Q_s^1, Q_s^2, Q_s^3, Q_s^4, and Q_s^5 buckled to obtain H_1, H_2, H_3, H_4, and H_5, respectively. Define $C_0 = \begin{bmatrix} H_0 \cdot \xi_c \\ \xi_c \end{bmatrix}$, $C_1 = \begin{bmatrix} H_1 \cdot \xi_c \\ \xi_c \end{bmatrix}$, $C_2 = \begin{bmatrix} H_2 \cdot \xi_c \\ \xi_c \end{bmatrix}$, $C_3 = \begin{bmatrix} H_3 \cdot \xi_c \\ \xi_c \end{bmatrix}$, and $C_5 = \begin{bmatrix} H_5 \cdot \xi_c \\ \xi_c \end{bmatrix}$. Modeling normalized constraints

$$\begin{cases} \beta_0 \cdot C_0 + \beta_1 \cdot C_1 + \cdots + \beta_5 \cdot C_5 = 0 \\ Q_n^0 = diag(\xi_c, \xi_c, \ldots, \xi_c) \cdot \left(\beta_0 \cdot Q_s^0 + \beta_1 \cdot Q_s^0 \right. \\ \qquad \left. + \cdots + \beta_5 \cdot Q_s^0\right) \\ Q_n^1 = diag(\xi_c, \xi_c, \ldots, \xi_c) \cdot \left(\beta_0 \cdot Q_s^1 + \beta_1 \cdot Q_s^1 \right. \\ \qquad \left. + \cdots + \beta_5 \cdot Q_s^1\right) \\ \cdots\cdots \\ Q_n^5 = diag(\xi_c, \xi_c, \ldots, \xi_c) \cdot \left(\beta_0 \cdot Q_s^5 + \beta_1 \cdot Q_s^5 \right. \\ \qquad \left. + \cdots + \beta_5 \cdot Q_s^5\right) \end{cases} \tag{6}$$

where C_0, C_1, C_2, C_3, C_4, and C_5 are the expansion matrix. $\beta_0, \beta_1, \beta_2, \beta_3, \beta_4$, and β_5 are the constraint factors of the expansion matrix. Q_n^0, Q_n^1, Q_n^2, Q_n^3, Q_n^4, and Q_n^5 are the temperature matrix after constraints. $diag$ is a diagonal matrix. The ξ_c is the equilibrium branching parameter

$$\begin{aligned} &\beta_0 \cdot C_0 + \beta_1 \cdot C_1 + \cdots + \beta_5 \cdot C_5 \\ &\quad = \beta_0 \begin{bmatrix} H_0 \cdot \xi_c \\ \xi_c \end{bmatrix} + \beta_1 \begin{bmatrix} H_1 \cdot \xi_c \\ \xi_c \end{bmatrix} + \cdots + \\ &\beta_5 \begin{bmatrix} H_5 \cdot \xi_c \\ \xi_c \end{bmatrix} = \begin{bmatrix} H_0 \cdot \xi_c \cdot \beta_0 \\ \xi_c \cdot \beta_0 \end{bmatrix} + \begin{bmatrix} H_1 \cdot \xi_c \cdot \beta_1 \\ \xi_c \cdot \beta_1 \end{bmatrix} \\ &\quad + \cdots + \begin{bmatrix} H_5 \cdot \xi_c \cdot \beta_5 \\ \xi_c \cdot \beta_5 \end{bmatrix} \end{aligned}$$

where

$$\beta_0 = [\, E_H^0 \ \cdots \ E_H^1 \,]^T \begin{bmatrix} \left|Q_s^0 - Q_s^1\right| & 0 \\ 0 & \left|Q_s^0 - Q_s^5\right| \end{bmatrix} \begin{bmatrix} E_H^0 \\ \vdots \\ E_H^1 \end{bmatrix}.$$

$$\beta_1 = \left[\, E_H^0 \ \cdots \ E_H^1 \,\right]^T \begin{bmatrix} \left|Q_s^1 - Q_s^0\right| & 0 \\ 0 & \left|Q_s^1 - Q_s^2\right| \end{bmatrix} \begin{bmatrix} E_H^0 \\ \vdots \\ E_H^1 \end{bmatrix}.$$

The rest are analogous. The normalized constraint model ensures that the temperature matrix is branch limited in compression. Establish a normalized compression system and construct an auxiliary matrix

$$K_\gamma^0 = \begin{bmatrix} \frac{Q_n^0(1,1)}{\max(Q_n^0)} & \cdots & \frac{Q_n^0(1,5)}{\max(Q_n^0)} \\ \cdots & \cdots & \cdots \\ \frac{Q_n^0(5,1)}{\max(Q_n^0)} & \cdots & \frac{Q_n^0(5,5)}{\max(Q_n^0)} \end{bmatrix}, K_\gamma^1 = \begin{bmatrix} \frac{Q_n^0(1,1)}{\max(Q_n^1)} & \cdots & \frac{Q_n^1(1,5)}{\max(Q_n^1)} \\ \cdots & \cdots & \cdots \\ \frac{Q_n^1(5,1)}{\max(Q_n^1)} & \cdots & \frac{Q_n^1(5,5)}{\max(Q_n^1)} \end{bmatrix}$$

$$K_\gamma^2 = \begin{bmatrix} \frac{Q_n^2(1,1)}{\max(Q_n^2)} & \cdots & \frac{Q_n^2(1,5)}{\max(Q_n^2)} \\ \cdots & \cdots & \cdots \\ \frac{Q_n^2(5,1)}{\max(Q_n^2)} & \cdots & \frac{Q_n^2(5,5)}{\max(Q_n^2)} \end{bmatrix}, K_\gamma^3 = \begin{bmatrix} \frac{Q_n^3(1,1)}{\max(Q_n^3)} & \cdots & \frac{Q_n^3(1,5)}{\max(Q_n^3)} \\ \cdots & \cdots & \cdots \\ \frac{Q_n^3(5,1)}{\max(Q_n^3)} & \cdots & \frac{Q_n^3(5,5)}{\max(Q_n^3)} \end{bmatrix}$$

$$K_\gamma^4 = \begin{bmatrix} \frac{Q_n^4(1,1)}{\max(Q_n^4)} & \cdots & \frac{Q_n^4(1,5)}{\max(Q_n^4)} \\ \cdots & \cdots & \cdots \\ \frac{Q_n^4(5,1)}{\max(Q_n^4)} & \cdots & \frac{Q_n^4(5,5)}{\max(Q_n^4)} \end{bmatrix}, K_\gamma^5 = \begin{bmatrix} \frac{Q_n^5(1,1)}{\max(Q_n^5)} & \cdots & \frac{Q_n^5(1,5)}{\max(Q_n^5)} \\ \cdots & \cdots & \cdots \\ \frac{Q_n^5(5,1)}{\max(Q_n^5)} & \cdots & \frac{Q_n^5(5,5)}{\max(Q_n^5)} \end{bmatrix}$$

where K_γ^0, K_γ^1, K_γ^2, K_γ^3, K_γ^4, and K_γ^5 are an auxiliary matrix. $Q_n^0(1,1)$ is the element in row 1 and column 1 of Q_n^0. $\max(Q_n^0)$ represents the largest element in Q_n^0

$$H_{l_1} = \sum_{m=1}^{5} \sum_{n=1}^{5} K_\gamma^{l_1}(m,n) \tag{7}$$

where H_{l_1} is a constant. l_1 is an integer in $[0, 5]$. $K_\gamma^{l_1}(m,n)$ is the element of the mth row and nth column in $K_\gamma^{l_1}$. Both m and n are constants. Then the normalized temperature matrices are denoted as Q_p^0, Q_p^1, Q_p^2, Q_p^3, Q_p^4, and Q_p^5. They satisfy

$$Q_p^{l_1}(m,n) = \frac{K_\gamma^{l_1}(m,n)}{H_{l_1}} \ (0 \le l_1 \le 5, 1 \le m \le 5, 1 \le n \le 5) \tag{8}$$

where $Q_p^{l_1}(m,n)$ is the element of the mth row and nth column in $Q_p^{l_1}$. Then Q_p^0, Q_p^1, Q_p^2, Q_p^3, Q_p^4, and Q_p^5 at this point are the temperature matrix compressed to between 0 and 1. Normalization of the temperature matrix ensures comparability between temperature readings from different sensors and eliminates errors caused by numerical differences. Second, normalized temperature values in the range 0–1 facilitate visual interpretation and integration with other metrics. This is the basis for accomplishing digital temperature image conversion through library functions. It improves the overall effectiveness and usability of the monitoring system.

Based on the above information, the temperature data matrix is converted into a temperature heat map. The temperature heat map is shown in Fig. 3. The converted temperature thermogram also needs to be smoothed. There are many commonly used interpolation methods, such as bilinear interpolation, cubic spline interpolation, and Kriging interpolation. In this method, we use Kriging interpolation.

This is due to the remarkable smoothness of Kriging interpolation when dealing with temperature matrices. It is an interpolation model based on statistical principles. The interpolation method takes into account the properties of spatial temperature variability. It effectively balances the effect of neighboring points by semivariance function and weight calculation. It makes the interpolation results present a continuous and smooth character in the whole matrix region. This smoothness helps to estimate the temperature of the target point more accurately. Especially when spatial variability exists, it provides a more reliable depiction of the temperature field. It is known that Q_p^0, Q_p^1, Q_p^2, Q_p^3, Q_p^4, and Q_p^5. Set the temperature traversal function, which is used to characterize the surrounding sensor locations of the target sensor

$$Z(p_D, p_\tau) = (p_o - p_D)(p_o - p_\tau) \cdot p_o^+ + p_D \cdot (p_o - p_\tau) \cdot p_o^- + p_\tau \cdot (p_o - p_D) \cdot p_o^\uparrow + p_D \cdot p_o^\downarrow \cdot p_\tau \tag{9}$$

where $Z(p_D, p_\tau)$ is the temperature traversal function. p_D and p_τ are the positions of any two sensors other than the target sensor. p_o is the target sensor position. p_o^+, p_o^-, $p_o^\uparrow$, and $p_o^\downarrow$ are the immediate neighboring sensors above and below the left and right of the target sensor, respectively. If the immediate neighboring sensors do not exist, make them zero. Create the exponential semi-variance function in Kriging interpolation

$$C(Z(p_D, p_\tau)) = \sigma^2 \cdot \exp\left(-\left(\frac{Z(p_D, p_\tau)}{\varpi}\right)^\phi\right) \tag{10}$$

where $C(Z(p_D, p_\tau))$ is the semivariance function, which is used for spatial variability between sensor arrays. σ^2 denotes the variance. ϖ is the interpolation range parameter. ϕ is the shape parameter. The adjustment of these parameters increases the smoothness of the temperature thermogram and makes it easier for feature extraction. In the interpolation smoothing process, a new coordinate grid is generated in order to generate a set of uniformly distributed coordinate points in the target region for interpolation. This ensures that the interpolation results have reasonable estimates throughout the region.

The position matrix of the sensor array is known: $P_{\text{sensor}} = [p_{1,1}, \ldots, p_{1,10}; \cdots; p_{10,1}, \ldots, p_{10,10}]$. The position matrix of the selected sensor subarray can be expressed as $P_{\text{sensor}} = [p_{1,1}, \ldots, p_{1,5}; \cdots; p_{5,1}, \ldots, p_{5,5}]$. Define the coordinate network of the interpolated region to form a network region of size $R \times C$. The specific generation method is as follows:

$$x_{\text{new},i} = x_{\min} + i \cdot \frac{x_{\max} - x_{\min}}{R - 1} \tag{11}$$

$$y_{\text{new},j} = y_{\min} + j \cdot \frac{y_{\max} - y_{\min}}{C - 1} \tag{12}$$

where $x_{\max}$, $x_{\min}$, $y_{\max}$, and $y_{\min}$ are the maximum and minimum coordinates of the original position, respectively. i and j are constants, respectively. $x_{\text{new},i}$ and $y_{\text{new},j}$ denote the coordinates of the newly generated position. R is the length of the network area and C is the width of the network area.

$x_{\max}$, $x_{\min}$, $y_{\max}$, and $y_{\min}$ are determined by spatial variability in temperature. This is done as follows. The variability in

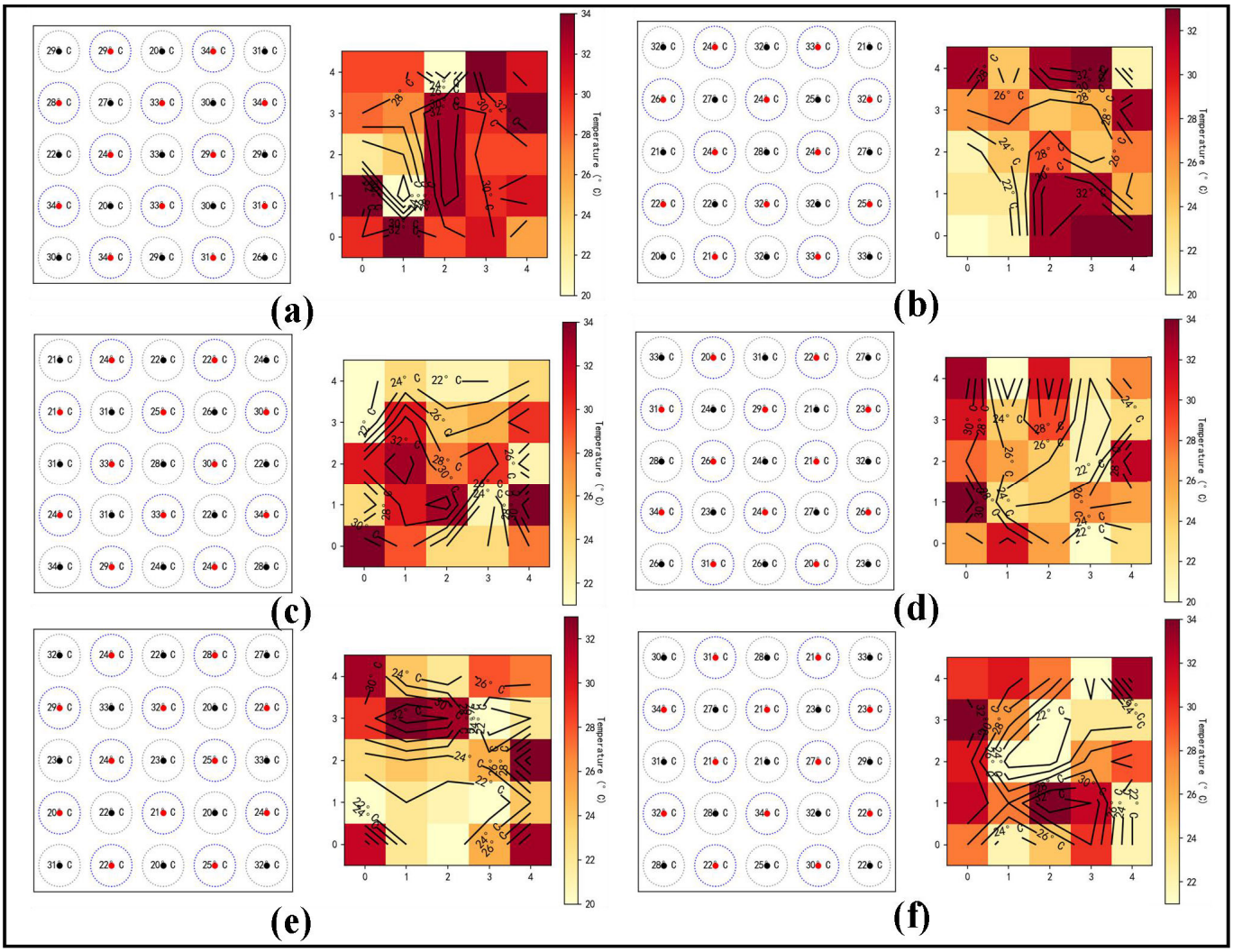


Fig. 3. Temperature heat map. (a)–(f) Six different temperature heat maps.

temperature space is analyzed with the help of $C(Z(p_D, p_\tau))$. Any two points in the network region are selected and the quantization process is as follows:

$$C\left(Z\left(p_D, p_\tau\right)\right) = \sigma^2 \cdot \left(1 - \exp\left(-\left(\frac{Z\left(p_D, p_\tau\right) \cdot Tk_d}{\varpi}\right)^{\phi}\right)\right) \tag{13}$$

where Tk_d is the temperature difference between any two points in the network area. The corresponding $C(Z(p_D, p_\tau))$ is obtained from the temperature difference between different points. $C(Z(p_D, p_\tau))$ trend is analyzed and the appropriate difference range parameter (ϖ) is determined to ensure that the interpolation results have accuracy and continuity throughout the monitoring region. Set the area threshold to $\gamma_{\text{threshold}}$. If the semivariance function of two points is greater than this threshold, then the temperature difference between the two points is considered to be a neighboring point. Determine other neighboring points in the same way.

1) Select a neighboring point and calculate the temperature difference between it and the other points. The points with the maximum and minimum temperature difference are noted as datum points (p_{sd1} and p_{sd2}).
2) For each neighboring point, determine the set of semivariable functions (k_{vary}) for the neighboring points around it. These sets contain all functions less than $\gamma_{\text{threshold}}$. Semivariable functions that are common between sets and sets and are smaller than $\gamma_{\text{threshold}}$ are defined as new sets ($k_{\text{vary}}^{\text{new}}$).
3) $x_{\max} = x_{\text{rand}}$, where x_{rad} is the average horizontal coordinate of the point corresponding to the largest value after p_{sd1} is added to $k_{\text{vary}}^{\text{new}}$ to form a new set.
4) $x_{\min} = x_{\text{ran}}$, where x_{ran} is the average level coordinate of the point corresponding to the smallest value after p_{sd2} is added to $k_{\text{vary}}^{\text{new}}$ to form the new set.
5) $y_{\max} = y_{\text{rad}}$, where y_{rad} is the average vertical coordinate of the point corresponding to the largest value after p_{sd1} is added to $k_{\text{vary}}^{\text{new}}$ to form a new set.
6) $y_{\min} = y_{\text{ran}}$, where y_{ran} is the average vertical coordinate of the point corresponding to the smallest value after p_{sd2} is added to $k_{\text{vary}}^{\text{new}}$ to form a new set. Based on the above information, the exact values of $x_{\max}$, $x_{\min}$, $y_{\max}$, and $y_{\min}$ are determined.

The newly generated position is added to the coordinate network of the interpolated region and is continuously updated

in this way. The method provides uniformly distributed target points for the interpolation region. This allows the interpolation function to be estimated accurately at these points. The calculation of the interpolation is satisfied

$$Z_{\text{inter}}\,(0 \le l_1 \le 5) = Q_p^{l_1} \cdot C\,(Z\,(p_D,\, p_\tau)) \sum_{i=1}^{R} \sum_{j=1}^{C} \left(x_{\text{new},i} - H_{l_1}\right)\left(y_{\text{new},\,j} - H_{l_1}\right) \tag{14}$$

where Z_{inter} is the interpolation calculation function. During the interpolation smoothing process, we focus on adjusting the key parameters of the Kriging interpolation model. These parameters include the variance parameter σ^2, the interpolation range parameter ϖ, and the shape parameter ϕ. The variance parameter influences the overall smoothness of the interpolation, while the interpolation range parameter controls the extent to which neighboring points affect the target point. The shape parameter affects the sensitivity of the interpolation function to neighboring points. Adjusting these parameters enables the model to more accurately adapt to the global and local characteristics of the actual data, resulting in optimized interpolation of the temperature matrix. This leads to a more precise and smoother estimation of the temperature distribution. After completing the temperature matrix smoothing, the subarray temperature thermogram is depicted in Fig. 4 as $P_1 - P_6$. Subsequently, the sensor subarray of size 5×5 is extended to the entire sensor array, as shown in Fig. 5. The smoothed temperature thermogram is obtained using the same procedure. In the sensor array of size 10×10, 24 different temperature thermograms are generated, denoted as $P_1 - P_{24}$. These thermograms illustrate the temperature variations, as shown in Fig. 6.

C. Construction of DLM Mask R-CNN Model

Mask R-CNN is an instance segmentation model that builds upon the Faster R-CNN architecture. It enhances target recognition and achieves accurate pixel segmentation for each instance by incorporating additional branching networks. The improved Mask R-CNN model (referred to as DLM Mask R-CNN) proposed in this article extends the original Mask R-CNN model with an adaptation layer, a data enhancement module, and a color difference degree loss function (referred to as C-Loss). The architecture of the DLM Mask R-CNN model is illustrated in Fig. 7. The working principle of DLM Mask R-CNN is as follows.

The smoothed and processed temperature thermogram of the sensor array is used as the feature input to the system. First is the adaptation layer. The functions of the adaptation layer include preprocessing, normalizing, and resizing the input images to match the input requirements of the model. Thus, it mainly ensures that the model can effectively learn and process the temperature information in subsequent layers for more accurate target detection and segmentation tasks. The final output is the adaptation result. The inputs Resnet101 backbone network and feature pyramid network (FPN) are used for primary feature map extraction immediately afterward. The extracted primary feature map F^1 completes the secondary feature map extraction through the data enhancement module, the attention mechanism module, and the open pose module, respectively. Among them, the data enhancement module improves the robustness of the model by introducing diverse temperature scene transformations. It enables the model to learn and extract temperature features more efficiently to cope with the noise in temperature variations. The open pose module generates the temperature key point heat map convolutional block attention module (CBAM) attention mechanism for maximum pooling CBAM consists of two parts. They are temperature channel attention module (T-CAM) and temperature spatial attention module (T-SAM). The above part performs the convolution of the feature map with maximum pooling and average pooling. The important features of the temperature heat map are extracted. Subsequently, these feature maps are fed into the region of interest (ROI) of the DLM Mask R-CNN network model. Finally, classification and regression processes are performed using multilayer convolution, class box, and mask branching. Significant temperature difference image is generated. CLoss is the color difference degree loss function. It consists of CLoss_1, CLoss_2, and CLoss_3. These are the stage loss functions of color variability loss function in processing. The color difference degree loss function is used to measure the difference between the output features of the model and the input features. The DLM Mask R-CNN is now analyzed in detail.

1) Basic Mask R-CNN Network: The basic Mask R-CNN network comprises four components: feature extraction, candidate region network, ROI alignment layer, and task branching. Feature extraction typically employs a pretrained backbone network to extract rich semantic features from the input image. The candidate region network is responsible for generating potential target candidate frames. The ROI alignment layer performs precise ROI pooling operations to map candidate frames of different sizes onto a fixed-size feature map. Finally, the task branch encompasses the target detection branch and the segmentation branch, responsible for predicting the bounding box of the target and generating an accurate mask of the target, respectively. This structure enables Mask R-CNN to conduct target detection and instance segmentation simultaneously, providing comprehensive semantic and precise spatial information for each detected target. The loss function of Mask R-CNN mainly consists of three parts: Loss_{cls}, Loss_{box}, and $\text{Loss}_{\text{mask}}$.

2) Adaptation Layer: After inputting the features, the first thing that comes in is the adaptation layer. Let X_{in} be a number of temperature thermograms for the input. In this article, 24 sheets are used as an example. The feature is separated. $X_{\text{in}} = P_1 + \cdots + P_{24}$. X_{out} is the output of adaptation and the adaptation result is output. The output adaptation result is denoted as $X_{\text{in1}}, X_{\text{in2}}, \ldots, X_{\text{in24}}$. Ada is the adaptation function. Then the adaptation process can be expressed as

$$X_{\text{out}} = Ada\,(X_{\text{in}}) \tag{15}$$

where X_{out} is the adapted output. Ada is the adaptation function. X_{in} is the input of several temperature heat maps. The output adaptation results are expressed as $X_{\text{in1}}, X_{\text{in2}}, \ldots, X_{\text{in24}}$. $X_{\text{in}} = P_1 + \cdots + P_{24}$.

The purpose of the adaptation layer in the DLM Mask R-CNN is to standardize temperature thermograms of varying

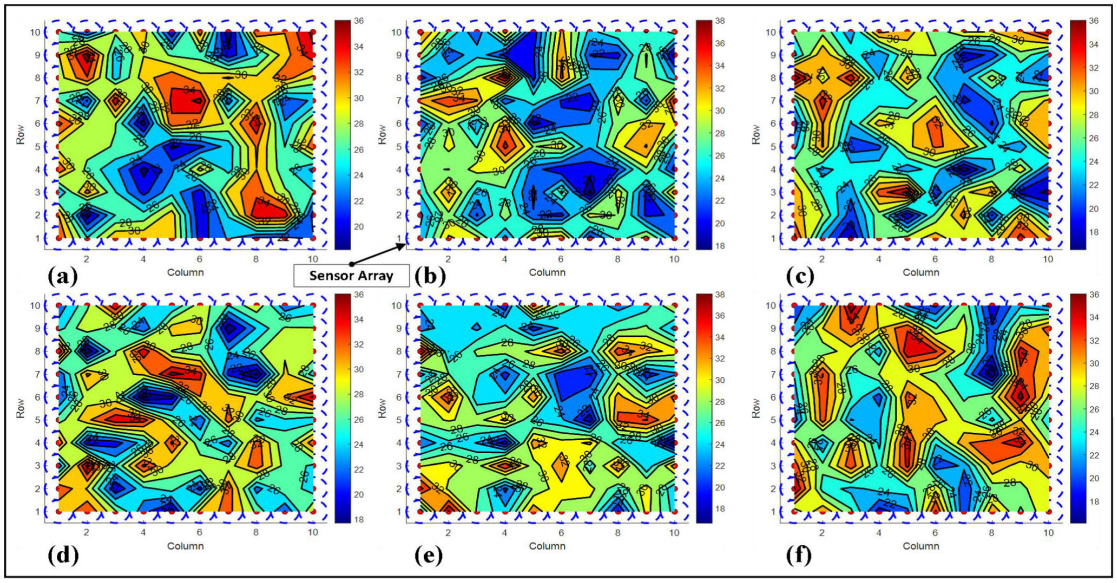


Fig. 4. Thermogram of temperature after smoothing treatment. (a)–(f) Six different temperature heat maps after smoothing.

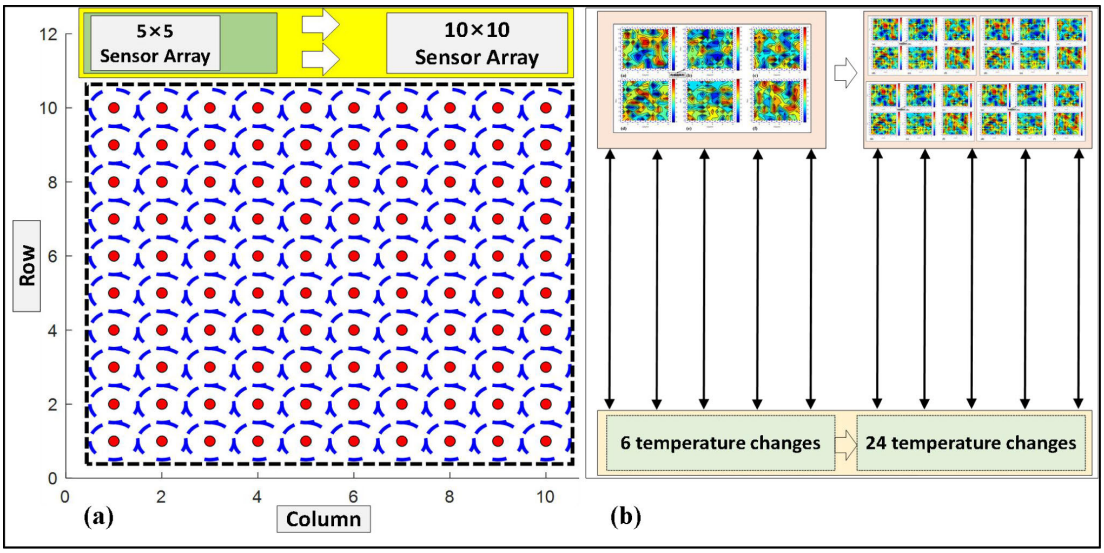


Fig. 5. 10 × 10 sensor array. (a) Planar distribution of the sensor array. (b) Expansion of six temperature changes to 24 temperature changes.

sizes, resolutions, or feature dimensions into the format required by the model. This ensures that the network can accurately process different types of data, essentially preprocessing the initial image.

3) Data Augmentation Module: In DLM Mask R-CNN, the data augmentation module plays a crucial role by randomly transforming and expanding the primary feature map, thereby increasing data diversity. This augmentation enhances the model's generalization capability and mitigates overfitting, ultimately improving the performance of temperature modeling across various scenarios. The module operates through three phases, namely, the affine phase, the nonlinear distortion phase, and the complex noise injection phase. The exploded view of the data enhancement module is shown in Fig. 8. The three stages are expressed as

$$\begin{cases} X_{\text{aug1}} = AT\left(X_{\text{ina}-24}, M_1\right) \\ X_{\text{aug2}} = ND\left(X_{\text{ina}-24}, \alpha_N, \beta_N\right) \\ X_{\text{aug3}} = \text{ICN}\left(X_{\text{ina}-24}, \alpha_I, \beta_I\right) \end{cases} \tag{16}$$

where AT, ND, and ICN represent the affine function, nonlinear distortion function, and complex noise injection function, respectively. X_{aug1}, X_{aug2}, and X_{aug3} represent the image data after processing in their respective stages. $X_{\text{ina1}-24}$ refers to any one of the image features output by F^1. M_1 represents the affine transformation matrix. α_N and β_N are parameters of the nonlinear aberration. α_I and β_I represent the standard deviation and intensity of the noise measure, respectively, enabling simulation of various noise sources in temperature images. After the stage transformation, classification and training of the features are completed, and the results are output.

4) Color Variability Loss Function (CLoss): The color difference degree loss function is introduced in DLM Mask R-CNN with three components: CLoss_1, CLoss_2, and CLoss_3. The principle is to guide the model to capture the color information of the target more accurately by comparing the color features of the mask generated by the model with the actual target. CLoss_1 focuses on the average value of the colors. CLoss_2 focuses on the contrast of the colors. CLoss_3 focuses on

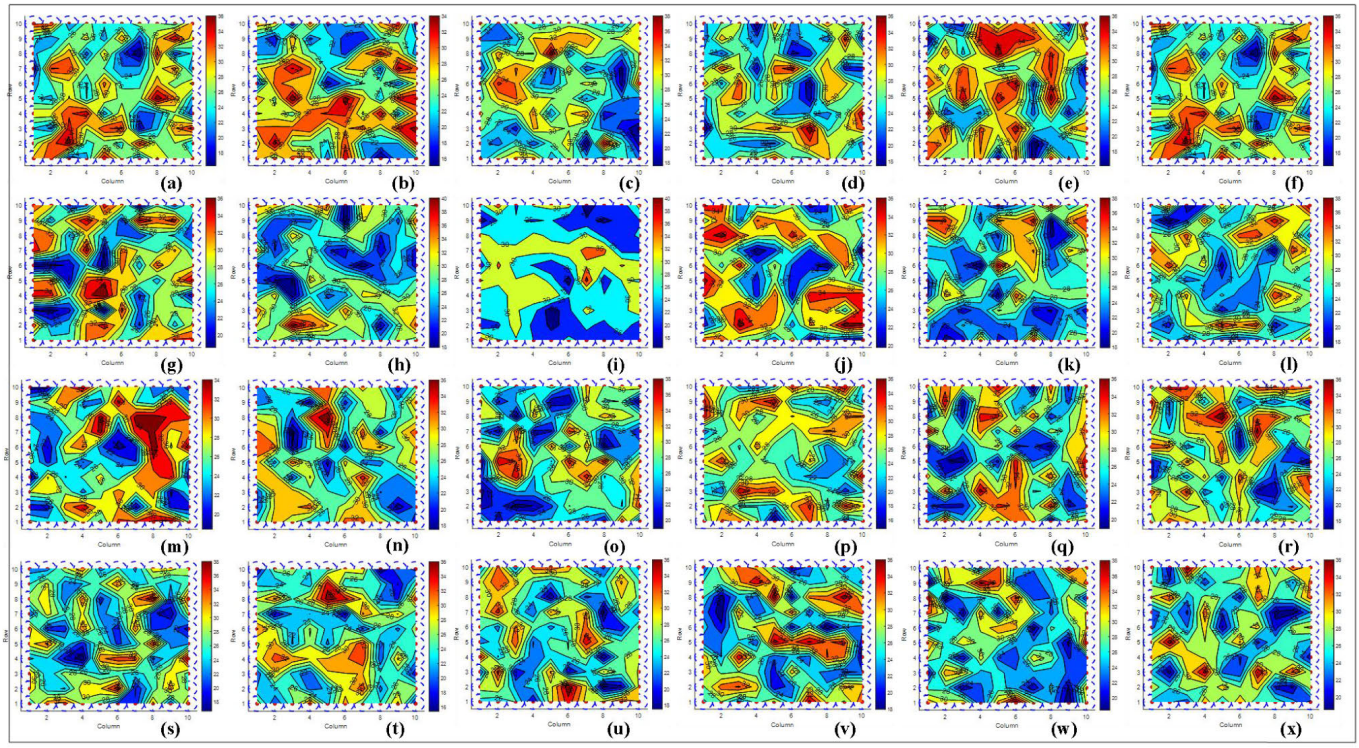


Fig. 6. (a)–(x) Thermogram of 24 different temperatures of the sensor array.

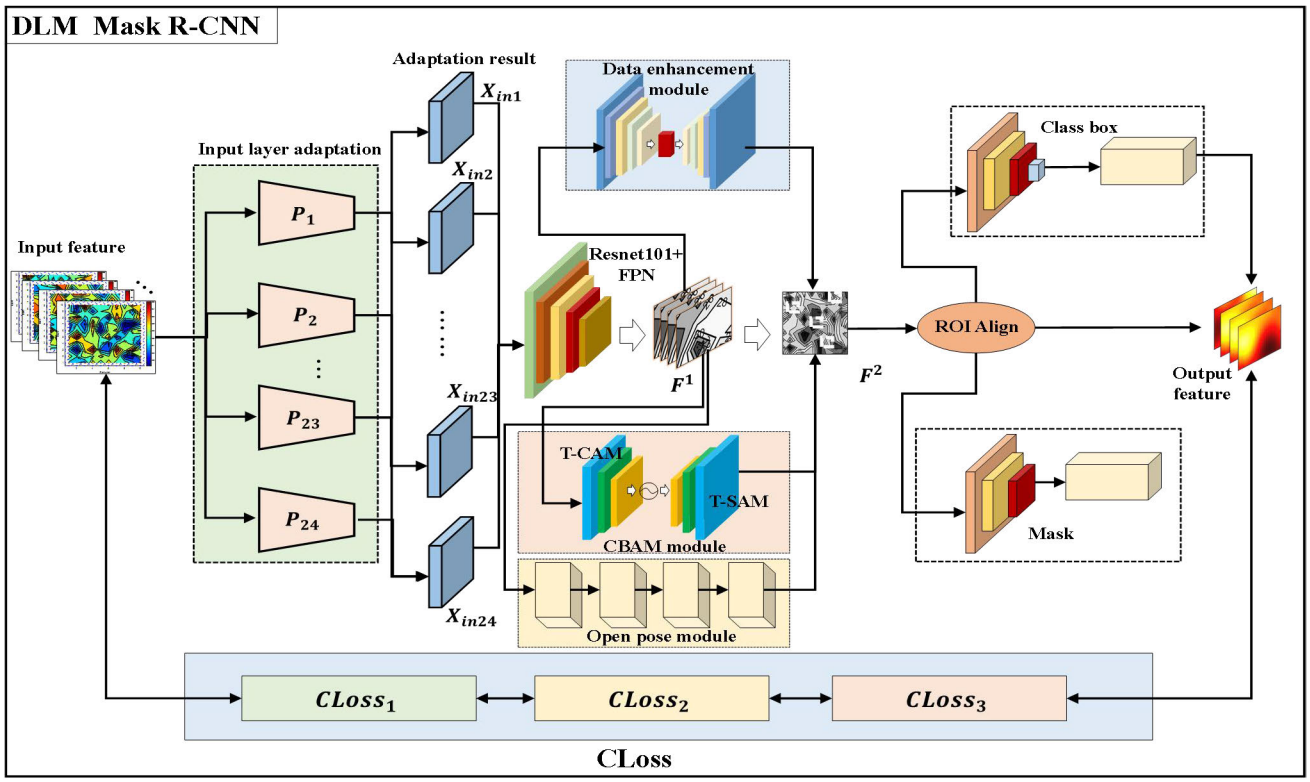


Fig. 7. DLM Mask R-CNN model architecture diagram.

the uniformity of the color. This combination of overall characteristics of color, contrast, and uniformity improves the model's ability to perceive the color of the target. The model is depicted as follows:

$$\mathrm{CLoss} = \lambda_{c1} \cdot \mathrm{CLoss}_1 + \lambda_{c2} \cdot \mathrm{CLoss}_2 + \lambda_{c3} \cdot \mathrm{CLoss}_3 \quad (17)$$

where CLoss is the color difference loss function. λ_{c1}, λ_{c2}, and λ_{c3} are all weighting factors. They control how much each component contributes to the overall loss. CLoss_1, CLoss_2, and CLoss_3 are the color difference loss functions of each part.

5) Improved CBAM Module: The CBAM introduced in DLM Mask R-CNN is improved to incorporate the features of

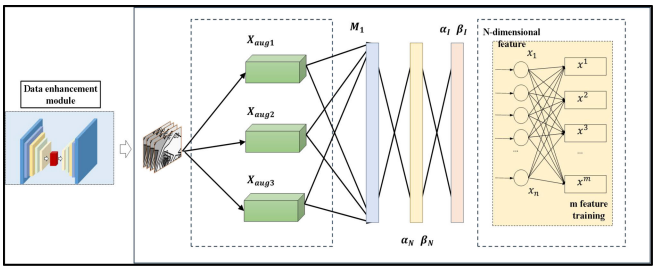


Fig. 8. Data enhancement module.

temperature monitoring. It includes T-CAM and T-SAM. T-CAM and T-SAM are described specifically in the following.

T-CAM uses temperature information to weight different channels of the feature map to highlight temperature-related features. It works by calculating the average temperature weights for each channel and then applying these weights to the feature maps of the corresponding channels. This helps the network to focus more on features related to temperature changes. The process can be modeled by the following equation:

$$\begin{aligned} T_{\mathrm{CAM}} &= \left(T_{\mathrm{pict}}\left(x, y\right), \alpha_{\mathrm{CAM}}\right) \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F_{\mathrm{temp}} \cdot F_{\mathrm{channel}}\left(T_{\mathrm{pict}}\left(x, y\right), \alpha_{\mathrm{CAM}}\right) dtds \end{aligned} \tag{18}$$

where $T_{\mathrm{CAM}} = (T_{\mathrm{pict}}(x, y), \alpha_{\mathrm{CAM}})$ denotes the weighted result of the α_{CAM}th channel at image $T_{\mathrm{pict}}(x, y)$. The binary integration represents the binary integration of the entire temperature distribution. It covers both spatial and temporal scales. F_{temp} represents a function of the temperature distribution. It describes the distribution of temperature weights at different points in time. $F_{\mathrm{channel}}(T_{\mathrm{pict}}(x, y), \alpha_{\mathrm{CAM}})$ represents the response value of the feature map at position $T_{\mathrm{pict}}(x, y)$, the α_{cam}th channel. The procedure is to compute the weighted result $T_{\mathrm{CAM}}(T_{\mathrm{pict}}(x, y), \alpha_{\mathrm{CAM}})$ for the α_{cam}th channel at position $T_{\mathrm{pict}}(x, y)$ by double integrating the temperature distribution F_{temp} and the feature map channel response $F_{\mathrm{channel}}(T_{\mathrm{pict}}(x, y), \alpha_{\mathrm{CAM}})$.

T-SAM by considering temperature information at different locations. The feature maps are spatially weighted to highlight areas associated with the temperature distribution. It works by calculating the temperature weights for each spatial location and then applying these weights to the feature maps at the corresponding locations. The process can be modeled by the following equation:

$$\begin{aligned} T_{\mathrm{SAM}} &= \left(T_{\mathrm{pict}}\left(x, y\right), \alpha_{\mathrm{CAM}}\right) \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F_{\mathrm{temp}} \cdot F_{\mathrm{spatial}}\left(T_{\mathrm{pict}}\left(x, y\right)\right) dtds \end{aligned} \tag{19}$$

where $T_{\mathrm{SAM}}(T_{\mathrm{pict}}(x, y), \alpha_{\mathrm{CAM}})$ denotes the weighting result for the α_{cam}th channel at image $T_{\mathrm{pict}}(x, y)$. F_{spatial} denotes the spatial distribution function of the feature map at position $T_{\mathrm{pict}}(x, y)$. The procedure is performed by double integration of the temperature distribution F_{temp} and the spatial distribution of the feature map F_{spatial}. The weighted result at position F_{spatial} is calculated for the α_{cam}th channel.

These modules leverage both average and maximum pooling operations effectively, enhancing CBAM's ability to distinguish between temperature feature channels. This refinement enables CBAM to prioritize important features, leading to overall performance improvement.

Based on the above enhancements, we construct the DLM Mask R-CNN. This model can recognize targets and extract features from the smoothed processed temperature thermogram to generate a significant temperature difference image. DLM Mask R-CNN offers notable advantages over traditional Mask R-CNN for temperature monitoring tasks. It facilitates target detection and segmentation sensitive to temperature changes. This temperature-aware design enhances the model's robustness in complex environments and elevates the accuracy and stability of temperature monitoring.

D. Color Discrepancy and Mapping Determination

Obtaining the degree of color difference of images with significant temperature differences is based on the Euclidean distance in the CIE Lab* color space ΔE. In the CIE Lab* color space, each color consists of three channels. L* (luminance), a* (green to red color range), and b* (blue to yellow color range), respectively. Thus, a color can be represented by three values. When calculating the degree of color difference, the difference between two colors is usually compared, so two sets of Lab* values are involved. The *RGB* to CIE *XYZ* color conversion matrix is mainly used to process the color information of images with significant temperature differences. The processing is as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124564 & 0.3575761 & 0.1804375 \\ 0.2126729 & 0.7151522 & 0.0721750 \\ 0.0193339 & 0.1191920 & 0.9503041 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{20}$$

where X, Y, and Z are the color attribute locations of the significant temperature difference image. R, G, and B are the colors of the three channels of red, green, and blue.

Transformation from *XYZ* color space to CIE Lab color space, where Lab denotes a transformation that takes into account luminance nonlinearity. This process is performed using the following equation:

$$\begin{cases} L^* = 116\left(\left(Y/Y_n\right)^{1/3}\right) - 16 \\ a^* = 500\left(\left(X/X_n\right)^{1/3} - \left(Y/Y_n\right)^{1/3}\right) \\ b^* = 200\left(\left(Y/Y_n\right)^{1/3} - \left(Z/Z_n\right)^{1/3}\right) \end{cases} \tag{21}$$

where X_n, Y_n, and Z_n are all standard values in *XYZ* space for a D65 light source. $L^*$is brightness. a^* is the color range from green to red. b^* is the color range from blue to yellow. Since the planar sensor array is considered in this article; therefore, Z can be set to 0. Then the Euclidean distance formula between two color attribute points in Lab $*$ color space is

$$D_{\mathrm{colour}}\left(X, Y, Z = 0\right)$$

$$= \sqrt{(L(X) - L(Y))^2 + (a(X) - a(Y))^2 + (b(X) - b(Y))^2} \tag{22}$$

where D_{color} is the Euclidean distance of the color attribute point. $L(X), L(Y), a(X), a(Y), b(X)$, and $b(Y)$ are the channel parameters regarding the position of the color attributes.

Based on the above information, we obtain the color dissimilarity of significant temperature difference images with the help of Euclidean distance in CIE Lab∗ color space. ΔE is the color dissimilarity set. It is denoted as $\Delta E = \{\Delta E_1, \Delta E_1, \cdots\cdots, \Delta E_u\}$. u is the number of color distinctiveness. The content of this set is determined by $D_{\text{color}}(X, Y, Z = 0)$., and the actual data of the changed temperature corresponding to ΔE is denoted as $T_{cs} = \{tm_1, tm_2, \cdots\cdots, tm_u\}$. that is, $\Delta E \Leftrightarrow T_{cs} = \{\Delta E_1 \Leftrightarrow tm_1, \Delta E_2 \Leftrightarrow tm_2, \cdots\cdots, \Delta E_u \Leftrightarrow tm_u\}$. The above set is filtered according to the temperature mapping rule established earlier. The rules are as follows.

1) The size after each temperature change should always be within the range of $[k_1, k_2]$. That is, $k_1 \leq T_{rand,rand} \leq k_2$.

2) The size of each temperature change should be less than the maximum threshold for temperature change, that is, $T_{\text{change}} < |T_{\max}|$.

$T_{\text{change}} < |T_{\max}|$. ΔE_{ranf} and tm_{ranf} are any elements in the sets ΔE and T_{cs}, respectively. The principle of determination is as follows: 1) determine whether ΔE_{ranf} satisfies rule 2). If $\Delta E_{ranf} \geq |T_{\max}|$, then directly determine ΔE_{ranf} at this time as a temperature anomaly state. If $\Delta E_{ranf} < |T_{\max}|$, then continue to determine tm_{ranf}. If $k_1 \leq tm_{ranf} \leq k_2$, then determine ΔE_{ranf} as temperature normal state. If $tm_{ranf} > k_2$ or $tm_{ranf} < k_1$, then judge ΔE_{ranf} as temperature abnormal state. If tm_{ranf} is judged first, then operate in the same way. The temperature state judgment rule is shown in Fig. 9. Two datasets are obtained after the judgment, that is, normal temperature collection and abnormal temperature collection. They are denoted as $T_{\text{normal}} = \{nor_1, nor_2, \cdots\cdots, nor_{\text{mid}}\}$ and $T_{\text{abnormal}} = \{abn_1, abn_2, \cdots\cdots, abn_{u-mid}\}$ respectively. The update rule of sensor array temperature is known according to (1). It is known that the mapping relationship between sensor array position information and temperature information is denoted as $f : \{P_{\text{sensor}}(\text{rand,rand}) \Leftrightarrow T_{\text{sensor}}(\text{rand, rand})\}$. Then we can judge the position of the sensors by the mapping relationship with the help of the normal temperature set and the abnormal temperature set. The state information of the sensor array is finally obtained. The size of the planar sensor array discussed in this article is 10×10. The total number of sensors is 100, but here $u \neq 100$. The reason is that the temperature difference image covers the entire sensor array. The coverage area is not only limited to the sensors themselves but also includes a lot of interstitial space; therefore, it is crucial to obtain the sensing range of each sensor. This is the reason why the sensor sensing range is designed in Fig. 1. If the temperature information is not in the sensing range of any sensor, then the proximity principle is used to determine the sensor location.

The method proposed in this article primarily aims to offer a novel solution for temperature monitoring of sensor arrays. It effectively establishes the correlation between temperature status and sensor location, ultimately facilitating the classification and identification of normal and abnormal temperatures within the sensor array. As for the impact of this method on temperature prediction, we provide detailed insights through Section II-E.

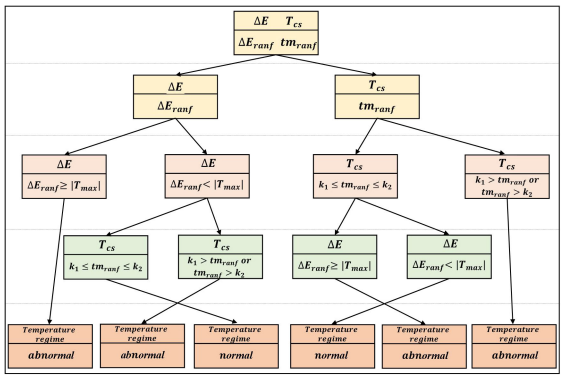


Fig. 9. Temperature status determination rules.

E. Experimental Analysis

1) Integrated Temperature Projections: The principle governing the combined temperature of the sensor array is as follows: initially, there is a 10×10 sensor array with only one sensor, which records 24 different temperatures corresponding to various moments of $t_1, t_2, \cdots\cdots, t_{24}$. Subsequently, sensors are incrementally added to the array, with each new sensor also recording 24 different temperatures corresponding to different moments of $t_1, t_2, \cdots\cdots, t_{24}$. This process continues until the number of sensors reaches 100.

1) Sensor array with one sensor, the combined temperature of this sensor array is syn_1, and the corresponding time is t_{n1}.

2) Sensor array with two sensors, the combined temperature of this sensor array is syn_2, corresponding time is t_{n2}.

3) The sensor array has 100 sensors, the combined temperature of this sensor array is syn_{100}, and the corresponding time is t_{n100}.

The above process is satisfied

$$syn_i = \frac{1}{r} \cdot \sum_{i=1}^{r} T\text{sensor}_i \tag{23}$$

where syn_i is the comprehensive temperature of the sensor array where the ith sensor is located. Tsensor_i represents the average temperature of 24 different temperatures of the ith sensor. Through the above process, we obtained 100 different integrated temperature data of the sensor array. These temperature data are saved as raw data and are predicted by different methods. These methods include the traditional ANN temperature prediction (ANNTM), phase shifted grating, and photoelectric oscillation temperature monitoring (MPTM), and the method of this article (TISM). The prediction results are tested as shown in Fig. 10.

Fig. 10 displays the raw, predicted, and smoothed predicted temperatures for the integrated temperature of the sensor array, along with the prediction error of each method. By comparing simulations, it becomes evident that the temperature

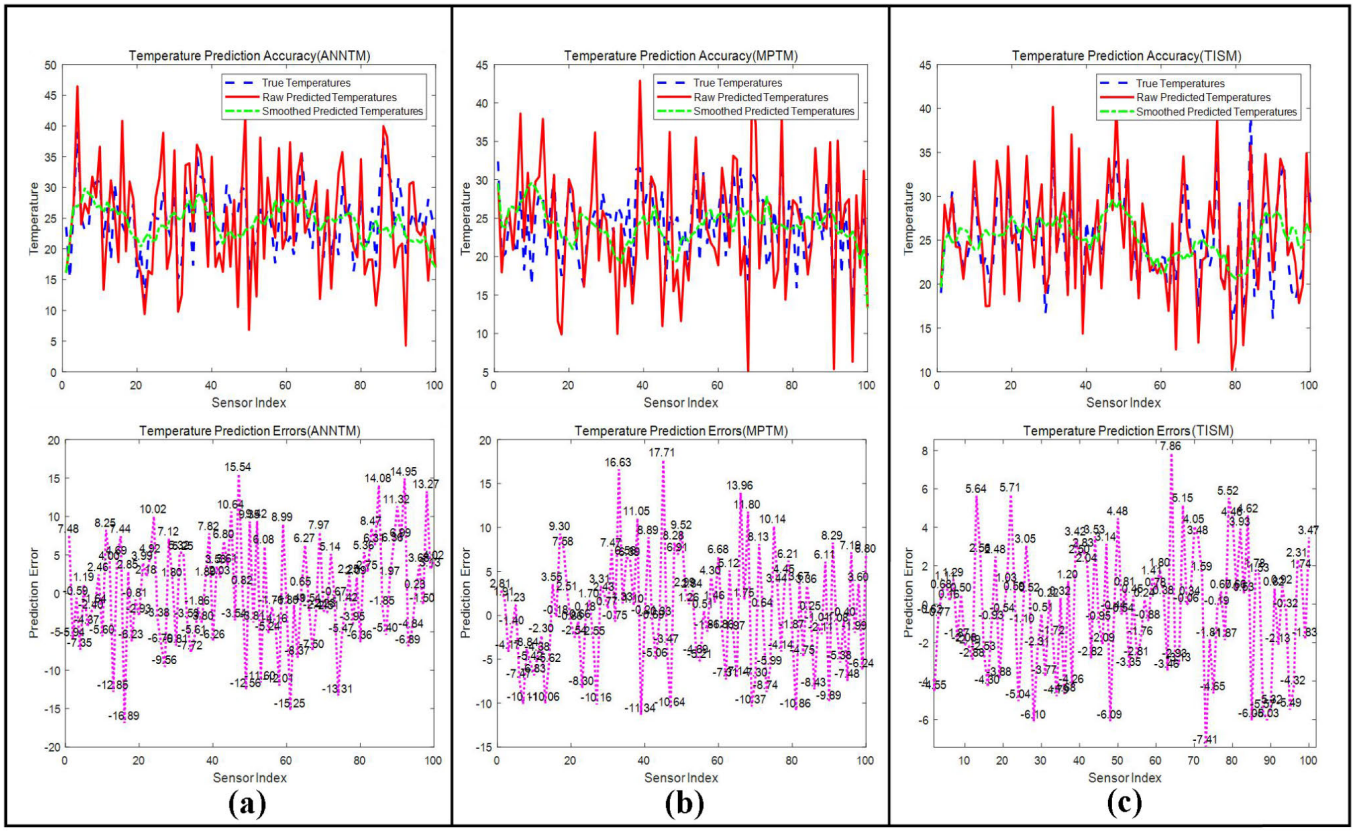


Fig. 10. Effectiveness of different methods for temperature prediction. (a) Temperature prediction effect and prediction error of ANNTM. (b) Temperature prediction effect and prediction error of MPTM. (c) Temperature prediction effect and prediction error of TISM.

prediction effect presented in this article exhibits a superior fitting degree and smaller prediction error. Consequently, it effectively fulfills the temperature-monitoring task of the sensor array.

2) Prediction of Accuracy and Stability: ANNTM involves decomposing raw temperature data into distinct datasets and constructing two neural network models. Temperature prediction is achieved by fusing the output of these models. In testing, ANNTM achieved a temperature prediction accuracy of 71.24% for mean deviation and 95.22% for standard deviation. Similarly, MPTM also exhibits high prediction accuracy. Now, based on 24 different temperatures of the sensor array, these two methods are compared with the approach presented in this article for prediction accuracy. The comparison results are illustrated in Fig. 11.

Fig. 11(a)–(c) provides a visual representation of temperature prediction accuracy using different methods, illustrating the error in predicted temperatures. The length of the error bars reflects the average temperature error for each sensor, with shorter bars indicating more accurate predictions and less uncertainty. Conversely, longer bars signify greater uncertainty and variability in the estimated temperature values. Fig. 11(c) demonstrates superior predictive stability and higher accuracy. The numbers above the error bars denote the center value of each error bar, representing the error center for each temperature change in the sensor array. The overall mean error of the method proposed in this article is −0.0285, with a total

TABLE I
MEAN AND VARIANCE OF ERROR CENTERS

Error center index	ANNTM	MPTM	TISM
Mean value	0.52083	0.52292	0.59833
variance	0.13744	0.11446	0.08538

error standard deviation of 0.9657 and calculated using the accuracy formula, the prediction accuracy reaches 97.13%. Compared to ANNTM, the temperature prediction accuracy for mean deviation and standard deviation is improved by 25.89% and 1.91%, respectively. Moreover, it exhibits higher prediction accuracy compared to MPTM

$$\text{Accuracy} = \left(1 - \frac{|\text{PME}|}{\text{TESD}}\right) \times 100\% \tag{24}$$

where Accuracy is the prediction accuracy, PME is the overall mean error, and TESD is the total error standard deviation. Fig. 11(d)–(f) shows the trend of the change in the center values of the error bars of the three methods. By observing the changes in the center values of the error bars of the three methods, their mean and variance can be calculated with the help of MATLAB. As shown in Table I.

Table I reveals that the mean value of the method proposed in this article is higher compared to ANNTM and MPTM. This suggests that the article's method generally yields higher mean predictive values. Additionally, the variance of this article's method is lower compared to the other two. Variance,

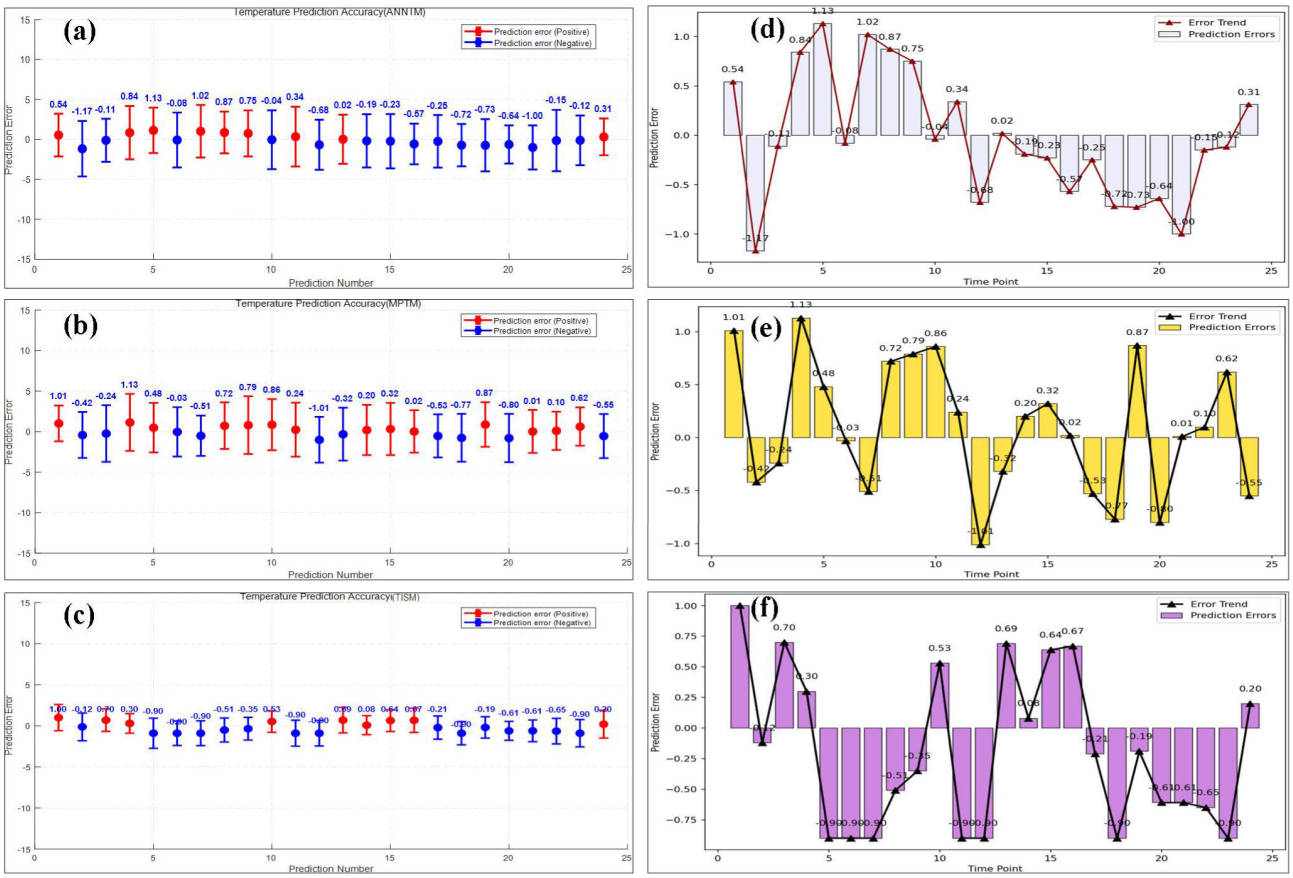


Fig. 11. Temperature prediction accuracy comparison. (a)–(c) Error effects of ANNTM, MPTM, and TISM in predicting temperature, respectively. (d)–(f) Changing trends of the error center values of these three methods.

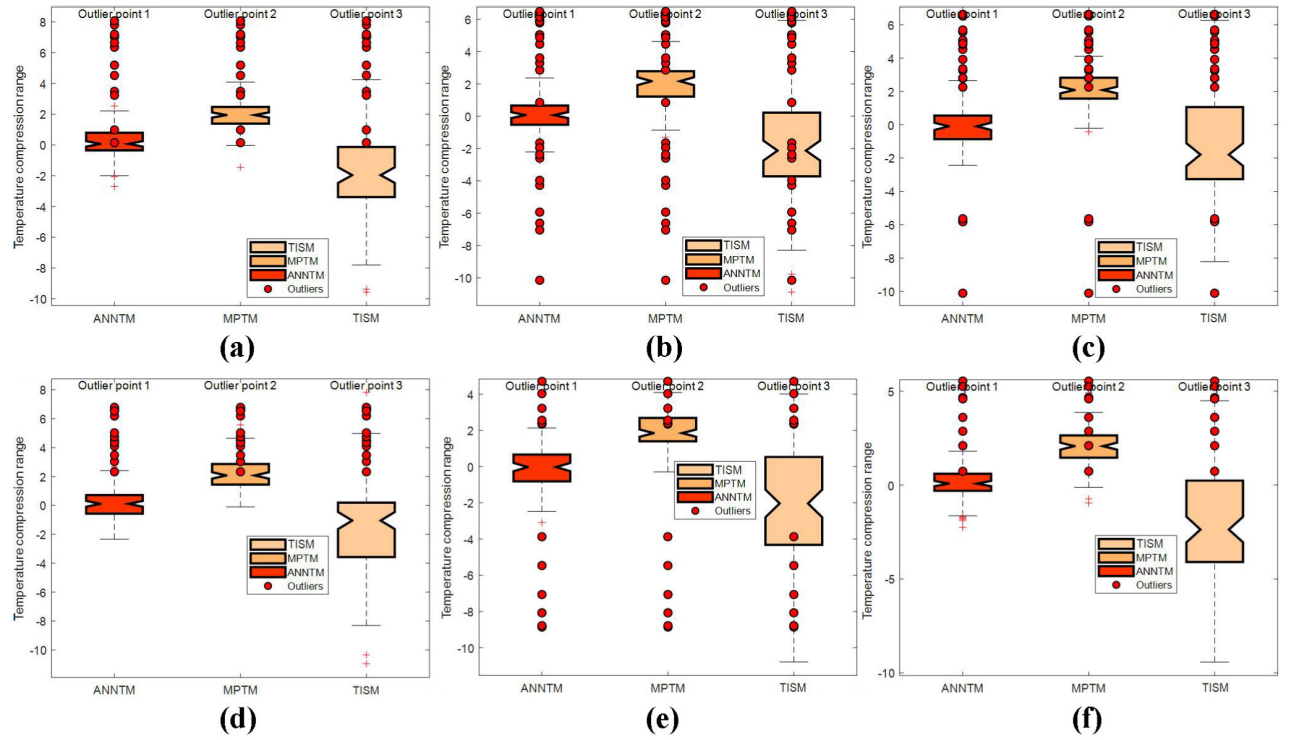


Fig. 12. Temperature-monitoring range of different methods. (a)–(f) Comparison of monitoring range between ANNTM, MPTM, and TISM.

as a measure of data dispersion, indicates how spread out the data points are. Therefore, lower variance suggests that the temperature prediction results of this article's method are more consistent, and the sensor array temperature monitoring

data exhibit less fluctuation. On the contrary, the temperature prediction results of the other two methods are more dispersed and exhibit greater fluctuations.

3) Temperature Monitoring Range: The comparison of temperature-monitoring ranges among the three methods is conducted. While MPTM excels at accurately monitoring temperature changes from 490 °C to 495 °C in real time, its monitoring range is limited. To better visualize the monitoring ranges and mitigate the significant differences among them, a temperature compression technique is employed in the simulation. The results of the simulation comparison experiment are depicted in Fig. 12.

In Fig. 12, each box corresponds to a temperature monitoring method: ANNTM, MPTM, and the method proposed in this article. The upper and lower boundaries of the boxes denote the first quartile ($Q1$) and the third quartile ($Q3$), respectively. The median (Median) is represented by the middle line within the box. The length of the box indicates the range of temperature data distribution, with a longer length suggesting a wider distribution of temperature values. The fill color of the box illustrates the distribution of temperature values for each method. Red dots indicate outliers, which are temperature data points that deviate significantly from the rest of the data. Analyzing the boxplot, we observe that the method proposed in this article exhibits relatively longer box lengths, indicating a broader distribution of temperature values. The longer box length also reflects a wider interquartile range of the data. In contrast, ANNTM and MPTM display relatively shorter boxes, suggesting that their temperature values are more concentrated within a narrower range. For instance, MPTM's temperature monitoring range is limited to 490 °C–495 °C. All three methods exhibit outliers, but the method proposed in this article shows a broader distribution of outlier points compared to the box boundaries. Since outliers represent extreme cases in the data, this suggest a wider range of temperature monitoring capabilities for the method proposed in this article.

III. Conclusion

In this work, a sensor array temperature monitoring method based on temperature mapping and improved Mask R-CNN is proposed. The method combines technologies such as sensor data processing, digital image conversion, image recognition, and color space analysis. It provides a new solution for temperature monitoring of car compartments and other complex environments through an array of optical fiber temperature sensors. The specific content is summarized in the following aspects.

A. Accurate Data Information and Rule Strategies

Establishing a mapping relationship between sensor temperature data, position information, and temperature limit rules ensures accurate data information and rule strategy for sensor array temperature monitoring.

B. Sensor Subarray Construction

Matrixing temperature data and utilizing matrix analysis and library function conversion to generate temperature heat maps. Interpolation technology enables smooth processing of temperature and heat maps, while expanding the subarray enhances overall monitoring system accuracy and stability. Through multipoint monitoring, it effectively overcomes the influence of sensor location deployment on temperature monitoring in complex environments.

C. Improved Mask R-CNN Model

Enhancements including an adaptive layer, data enhancement module, and color difference loss function are integrated into the model. The model combines channel and temperature information in CBAM, enabling target recognition, feature extraction, and differential temperature image generation. Utilizing the Euclidean distance in the CIE Lab$*$ color space effectively extracts temperature state information. Finally, sensor temperature determination is conducted according to temperature mapping rules, completing the sensor array temperature-monitoring task.

Furthermore, the proposed method is compared with two traditional methods based on sensor array integrated temperature prediction, accuracy, stability, and temperature-monitoring range.

The method achieves a prediction accuracy of 97.13%, outperforming ANNTM and MPTM in terms of average deviation, standard deviation, and monitoring range. The smaller prediction error and wider monitoring range demonstrate the method's potential for real-time accurate temperature monitoring of sensor arrays, making it a promising solution for practical applications.

Although the sensor array temperature-monitoring method proposed in this article has enhanced monitoring accuracy, stability, and monitoring range, it still faces certain limitations. One major limitation is its high dependence on sensor array data, which restrict monitoring results based on the quality and quantity of sensor data available. Additionally, the incorporation of various modules in the DLM Mask R-CNN model increases the complexity of the model, impacting its practical applicability.

Our future research directions aim to address these limitations by optimizing data quality, increasing data diversity, and simplifying algorithms. Furthermore, we plan to optimize and upgrade the DLM Mask R-CNN model by completing the compression and quantization processes. These efforts are aimed at further improving the performance and applicability of the sensor array temperature-monitoring method.

Finally, the research method of this article focuses on the monitoring of the internal temperature of the vehicle compartment in the loading system. Especially for semispace type vans. In the future, we will also further expand application scenarios and apply it to temperature monitoring of full-space vans and other complex environments.

Acknowledgment

The authors would like to thank the editors and reviewers for their suggestions and comments that helped to improve this article.

REFERENCES

[1] C. Zuliani et al., "Flow compensated gas sensing array for improved performances in breath-analysis applications," *IEEE Sensors Lett.*, vol. 4, no. 3, pp. 1–4, Mar. 2020, doi: 10.1109/LSENS.2020.2974200.

[2] A. Pouryazdan et al., "Design and characterisation of a non-contact flexible sensor array for electric potential imaging applications," *IEEE Sensors J.*, vol. 21, no. 23, pp. 26328–26336, Dec. 2021, doi: 10.1109/JSEN.2021.3064276.

[3] X. Tian et al., "Pressure sensor array with low-power near-sensor CMOS chip for human gait monitoring," *IEEE Sensors Lett.*, vol. 5, no. 2, pp. 1–4, Feb. 2021, doi: 10.1109/LSENS.2021.3053963.

[4] Y. Xu et al., "In-ear integrated sensor array for the continuous monitoring of brain activity and of lactate in sweat," *Nature Biomed. Eng.*, vol. 7, no. 10, pp. 1307–1320, Sep. 2023, doi: 10.1038/s41551-023-01095-1.

[5] Q.-B. Zhu et al., "A flexible ultrasensitive optoelectronic sensor array for neuromorphic vision systems," *Nature Commun.*, vol. 12, no. 1, Mar. 2021, Art. no. 1798, doi: 10.1038/s41467-021-22047-w.

[6] L. Wei, L. Guan, and L. Qu, "Prediction of sea surface temperature in the South China Sea by artificial neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 4, pp. 558–562, Apr. 2020, doi: 10.1109/LGRS.2019.2926992.

[7] K. Tu, Z. Xie, W. Zeng, D. Han, and H. Yan, "High temperature accurate monitoring based on phase-shifting grating and photoelectric oscillation," *IEEE Photon. Technol. Lett.*, vol. 33, no. 21, pp. 1169–1172, Nov. 1, 2021, doi: 10.1109/LPT.2021.3112569.

[8] N. Verma et al., "A novel wearable device for continuous temperature monitoring & fever detection," *IEEE J. Transl. Eng. Health Med.*, vol. 9, pp. 1–7, 2021, doi: 10.1109/JTEHM.2021.3098127.

[9] K. Thiyagarajan, G. K. Rajini, and D. Maji, "Cost-effective, disposable, flexible, and printable MWCNT-based wearable sensor for human body temperature monitoring," *IEEE Sensors J.*, vol. 22, no. 17, pp. 16756–16763, Sep. 2022, doi: 10.1109/JSEN.2021.3088466.

[10] H. Lee et al., "Development of highly sensitive and stable SAW-based temperature sensor array and its interface electronics for realtime monitoring of wafer surface temperature in plasma chamber," *J. Elect. Eng. Technol.*, vol. 19, no. 4, pp. 2491–2499, 2024.

[11] J. Peng, S. Jia, H. Yu, X. Kang, S. Yang, and S. Xu, "Design and experiment of FBG sensors for temperature monitoring on external electrode of lithium-ion batteries," *IEEE Sensors J.*, vol. 21, no. 4, pp. 4628–4634, Feb. 2021, doi: 10.1109/JSEN.2020.3034257.

[12] H. Chen et al., "Integrated lateral SBD temperature sensor of a 4H-SiC VDMOS for real-time temperature monitoring," *IEEE Trans. Electron Devices*, vol. 70, no. 7, pp. 3813–3819, Jul. 2023, doi: 10.1109/TED.2023.3278624.

[13] L. Atallah, E. Bongers, B. Lamichhane, and S. Bambang-Oetomo, "Unobtrusive monitoring of neonatal brain temperature using a zero-heat-flux sensor matrix," *IEEE J. Biomed. Health Informat.*, vol. 20, no. 1, pp. 100–107, Jan. 2016, doi: 10.1109/JBHI.2014.2385103.

[14] M. Badar, P. Lu, Q. Wang, T. Boyer, K. P. Chen, and P. R. Ohodnicki, "Real-time optical fiber-based distributed temperature monitoring of insulation oil-immersed commercial distribution power transformer," *IEEE Sensors J.*, vol. 21, no. 3, pp. 3013–3019, Feb. 2021, doi: 10.1109/JSEN.2020.3024943.

[15] D. Caputo, G. de Cesare, M. Nardini, A. Nascetti, and R. Scipinotti, "Monitoring of temperature distribution in a thin film heater by an array of a-Si:H temperature sensors," *IEEE Sensors J.*, vol. 12, no. 5, pp. 1209–1213, May 2012, doi: 10.1109/JSEN.2011.2167506.

Zuoxun Wang received the B.E. degree in motor and control from Shandong University of Technology, Zibo, China, in 1997, the M.E. degree in computer applications technology from Northeast Electric Power University, Jilin City, China, in 2005, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2015.

He is currently a Professor at the School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai, China. He has authored/coauthored over 40 journal articles and conference papers. His research interests include intelligent robot and control.

Chuanyu Cui was born in 1999. He is currently pursuing the master's degree in electronic information with Shandong Institute of Commerce and Industry, Jinan, China.

He has published some patents. His research interests include robot behavior planning and automated loading and unloading truck robots. He is also interested in the direction of image recognition, machine learning, microcontroller technology, and so on.

Mr. Cui has participated in mathematical modeling, physics competitions, mathematics competitions, and other competitions and won relevant awards.

Jinxue Sui received the B.E. degree in automation from Shandong University of Technology, Zibo, China, in 2001, the M.E. degree in control theory and control engineering from Northeast Electric Power University, Jilin City, China, in 2005, and the Ph.D. degree in navigation, directing and control from Naval Aviation University, Yantai, China, in 2009.

He is currently a Professor at the School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai. He has authored/coauthored over 30 journal papers and conference papers. His research interests include intelligent robot and control.

Yong Zhang was born in June 1979. He received the Ph.D. degree in engineering from Shandong Agricultural University, Tai'an, China, in 2019.

He is a Lecturer and a Teacher at the School of Information and Electronic Engineering, Shandong University of Business and Economics, Yantai, China. He has participated in the development and application of "A Leakage Monitoring System for Rigid Landfills in Qinghai," "A GPRS Based Environmental Monitoring System for Coastal Wetlands in the Yellow River Delta," "An Intelligent Fully Automatic Vegetable Transplanting Machine," "A Tea Withering and Drying system," and "A Fully Automatic Tea Processing and Production Line." He has published more than ten SCI, EI, and other papers, authorized two utility model patents, and one software copyright, accepted one invention patent.

Changkun Guo received the B.Eng. degree in computer science and technology from Ludong University, Yantai, China, in 2021. He is currently pursuing the master's degree in electronic and information engineering with the School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai.

His research interests include robots, robot control, and multiagent systems.