Supplemental Material for
A Machine Learning Surrogate Modeling Benchmark for Temperature Field Reconstruction of Heat-Source Systems
Xiaoqian Chen†, Zhiqiang Gong†*, Xiaoyu Zhao†, Weien Zhou† & Wen Yao†

1Defense Innovation Institute, Chinese Academy of Military Science, Beijing 100000, China

Appendix A Overview
This document includes supplementary material to “A Machine Learning Surrogate Modeling Benchmark for Temperature Field Reconstruction of Heat-Source Systems”. Included are detailed versions of the set of machine learning surrogate modeling methods, the TFRD dataset and other more results.

Appendix B Baseline Methods
Appendix B.1 Point-based Methods
Temperature field reconstruction using point-based methods focuses on one instance of heat source system. These methods attempt to learn the mapping function from coordinates of a certain point to the corresponding temperature value. For each instance, the domain of the system is divided into Pols and monitoring points where monitoring points are used as training samples and Pols as testing samples. In our tests, we choose three classes of commonly used methods, i.e. interpolation methods, general machine learning methods, and the neural networks.

Appendix B.1.1 Interpolation Methods
- k-nearest neighbor nonlinear interpolation (KInterpolation): Due to the non-uniformly distributed monitoring points in the system, nonlinear interpolation, i.e. the RBF interpolation, is selected as the interpolation method. Each of PoI is supposed to be affected by the k-nearest monitoring points. This work uses the Euclidean distance as the correlation metric between PoIs and monitoring points. Therefore, the reconstructed temperature at \((x_0, y_0)\) can be calculated as

\[
T(x_0, y_0) = \sum_{(x_{s_i}, y_{s_i}) \in S_k(x_0, y_0)} \frac{e^{-\|(x_0-x_{s_i})^2+(y_0-y_{s_i})^2\|_2}}{\sum_{i=1}^{m} e^{-\|(x_0-x_{s_i})^2+(y_0-y_{s_i})^2\|_2}} f(x_{s_i}, y_{s_i}),
\]

where \(S_k(x_0, y_0)\) describes the k-nearest monitoring points of \((x_0, y_0)\). \(T(x_0, y_0)\) stands for the predicted temperature values of \((x_0, y_0)\) and \(f(x_{s_i}, y_{s_i})\) denotes the monitoring temperature values.

- Global gaussian interpolation (GInterpolation): For GInterpolation, each of PoI is related to all the monitoring points. The reconstructed temperature at \((x_0, y_0)\) is related to all the monitoring points, and it can be formulated as

\[
T(x_0, y_0) = \sum_{i=1}^{m} \frac{e^{-\|(x_0-x_{s_i})^2+(y_0-y_{s_i})^2\|_2}}{\sum_{j=1}^{m} e^{-\|(x_0-x_{s_j})^2+(y_0-y_{s_j})^2\|_2}} f(x_{s_i}, y_{s_i}),
\]

where \(m\) describes the number of monitoring points.

Appendix B.1.2 General Machine Learning Methods
In this work, we evaluated 4 commonly used machine learning methods for TFR-HSS task, i.e., polynomial regression [28], random forest regression [29], Gaussian process regression [31], and support vector regression [32].

Appendix B.1.3 Neural Networks
- Multi-layer perception for point-based modeling (MLPP) [8]: Fig. B1 shows the network structure of the MLPP for TFR-HSS task. As the figure shows, for point-based methods, we construct the mapping from coordinates of the point to the temperature value.
- Restricted Boltzmann machine (RBM) [35]: RBM is one-layer version of DBN as Fig. B2 shows.
- Deep belief networks (DBNs) [34]: Fig. B2 shows the network architecture of DBNs for the reconstruction task.

* Corresponding author (email: gongzhiqiang13@nudt.edu.cn)
† Xiaoqian Chen and Zhiqiang Gong contributed equally to this work.
Appendix B.2 Vector-based Methods

Even though point-based methods are easy to implement, it performs one instance one task and one has to resolve the optimization for other instances. This would sharply increase the cost time in reconstruction process. Therefore, this work proposes other computational modelings for learning of one class one task. Among these modelings, vector-based methods are the simplest. This class of methods learns the mapping between the temperature vector of monitoring points to that of PoIs. This work selects the multi-layer perception (MLP) [8], Conditional Neural Processes (CNP) [6], and the Transformer [21] as representatives.

- **Multi-Layer Perception for vector-based modeling (MLPV) [8]**: Just as Fig. B3 shows, the vector of temperature information of monitoring points is used as the input and the temperature information of PoIs is obtained through the MLPV.
- **Conditional Neural Processes (CNP) [6]**: Fig. B4 shows the network structure of the CNP for the task.
- **Transformer [21]**: Fig. B5 shows the flowchart of Transformer method for TFR-HSS task.

Appendix B.3 Image-based Methods

As the simplest way to achieve one class one task, vector-based methods usually ignore the physical and spatial correlation between PoIs and monitoring points. Through domain discretization, we computationally model the TFR-HSS task as an image-to-image regression problem. It learns the mapping between the temperature matrix of monitoring points to the overall temperature field of the domain. Generally, the deep regression models are used as such image-based methods. This work adapts commonly used FCN, FPN, UNet and SegNet as baselines for image-based methods. As table B1 shows, each baseline framework supports several different backbone networks.

- **Fully convolutional networks (FCN) [24]**: Fig. B6 shows the network structure of FCN with AlexNet backbone for TFR-HSS task.
- **UNet [25]**: Fig. B7 shows the network structure of UNet for the TFR-HSS task.
- **SegNet [26]**: Fig. B8 shows the network architecture of SegNet for TFR-HSS task. As table 1 shows, AlexNet is chosen as the backbone network of SegNet.
Network structure of MLPV for TFR-HSS task.

**Table B1** The deep surrogate models used in image-based methods for TFR-HSS task.

| Backbone   | FCN | FPN | UNet | SegNet |
|------------|-----|-----|------|--------|
| AlexNet    | FCN-1 | ×   | ×    | SegNet |
| VGG16      | FCN-2 | ×   | UNet | ×      |
| ResNet18   | FCN-3 | FPN | ×    | ×      |

- Feature pyramid networks (FPN) [23]: Fig. B9 shows the structure of FPN and ResNet18 is used as the backbone of the FPN for current task.

**Appendix B.4 Graph-based Methods**

This work mainly tests the performance of graph convolutional network (GCN) [9].
- Graph convolutional network (GCN) [9]: Fig. B10 shows the network structure of GCN for TFR-HSS task.

**Appendix C TFRD**

**Appendix C.1 Task Definition**

**Appendix C.1.1 Boundary Conditions**

Based on the heat sink and sine-function boundary conditions, this work constructs the three typical boundary conditions used in TFRD as Fig. C1 shows.

**Appendix C.1.2 Components (or Heat Sources)**

As Fig. C2 shows, the TFRD mainly considers components with three different shapes, namely the rectangle-like, capsule-like and the circle-like shape.

Besides, the TFRD also considers the power distributions of different components. Two typical distributions in engineering, namely the uniformly distributed and non-uniformly distributed heat sources, are used in the construction of TFRD.

Considering the shapes as well as the power distributions of the heat sources, TFRD includes three types of heat-source layout information on the domain, namely the type A (see table C1 for details), type B (see table C2), type C (see table C3). In the table, ‘U’, ‘N’ represents the ‘uniformly’ and ‘non-uniformly’ distributed heat sources, respectively. ‘r’ denotes the ‘rectangle’, ‘p’ stands for the ‘capsule’ and ‘c’ represents the ‘circle’. All the heat sources are put horizontally, and the length and the width in these tables means the side length in horizontal and vertical direction, respectively.
**Table C1** The layout information and characteristics of heat source components of Type A in TFRD. The location means the center of the component in the 200 × 200 grid.

| No. | Type | Length(m) | Width(m) | Location          |
|-----|------|-----------|----------|-------------------|
| 1   | Ur   | 0.012     | 0.012    | (0.019, 0.0915)   |
| 2   | Ur   | 0.016     | 0.03     | (0.0875, 0.079)   |
| 3   | Ur   | 0.015     | 0.015    | (0.045, 0.0145)   |
| 4   | Ur   | 0.03      | 0.03     | (0.08, 0.025)     |
| 5   | Ur   | 0.02      | 0.02     | (0.0685, 0.0885)  |
| 6   | Up   | 0.03      | 0.015    | (0.036, 0.0335)   |
| 7   | Up   | 0.02      | 0.04     | (0.021, 0.0655)   |
| 8   | Up   | 0.015     | 0.03     | (0.0425, 0.0795)  |
| 9   | Up   | 0.02      | 0.03     | (0.06, 0.055)     |
| 10  | Up   | 0.03      | 0.02     | (0.022, 0.014)    |

**Table C2** The layout information and characteristics of heat source components of Type B in TFRD.

| No. | Type | Length(m) | Width(m) | Location          |
|-----|------|-----------|----------|-------------------|
| 1   | Ur   | 0.015     | 0.015    | (0.016, 0.0915)   |
| 2   | Ur   | 0.01      | 0.02     | (0.0925, 0.079)   |
| 3   | Ur   | 0.02      | 0.03     | (0.0825, 0.025)   |
| 4   | Up   | 0.015     | 0.02     | (0.0725, 0.0835)  |
| 5   | Up   | 0.015     | 0.03     | (0.036, 0.0335)   |
| 6   | Up   | 0.03      | 0.015    | (0.021, 0.0655)   |
| 7   | Nc   | 0.02      | 0.02     | (0.0465, 0.0795)  |
| 8   | Nc   | 0.028     | 0.028    | (0.06, 0.055)     |
| 9   | Nc   | 0.02      | 0.02     | (0.017, 0.014)    |
| 10  | Nc   | 0.024     | 0.024    | (0.055, 0.014)    |

**Table C3** The layout information and characteristics of heat source components of Type C in TFRD.

| No. | Type | Length(m) | Width(m) | Location          |
|-----|------|-----------|----------|-------------------|
| 1   | Nr   | 0.016     | 0.012    | (0.019, 0.0915)   |
| 2   | Nr   | 0.012     | 0.015    | (0.0875, 0.079)   |
| 3   | Nr   | 0.024     | 0.024    | (0.045, 0.0145)   |
| 4   | Nr   | 0.012     | 0.024    | (0.08, 0.025)     |
| 5   | Nr   | 0.015     | 0.012    | (0.0685, 0.0885)  |
| 6   | Nr   | 0.012     | 0.024    | (0.036, 0.04)     |
| 7   | Nr   | 0.018     | 0.018    | (0.015, 0.0655)   |
| 8   | Nr   | 0.024     | 0.012    | (0.0425, 0.0795)  |
| 9   | Nr   | 0.012     | 0.012    | (0.06, 0.055)     |
| 10  | Nr   | 0.018     | 0.018    | (0.017, 0.014)    |
| 11  | Nr   | 0.018     | 0.012    | (0.036, 0.061)    |
| 12  | Nr   | 0.018     | 0.009    | (0.061, 0.04)     |
Appendix C.1.3 Monitoring Points
For different sub-data in TFRD, the number of monitoring points is listed in table C4.

Appendix C.1.4 Representative Cases
Based on different boundary conditions, heat sources, and monitoring points, this work mainly considers three representative cases to formulate the TFRD. Corresponding to these cases, this work constructs the three sub-data in TFRD, namely the Heat Sink (HSink) sub-task, the All Dirichlet (ADlet) sub-task, and the Dirichlet by Sine function distribution (DSine) sub-task.

Case 1: HSink sub-task. HSink denotes a heat source system with heat sink for heat dissipation. The width of the heat sink $\delta$ is set to 0.01m with a constant temperature valued $T_0 = 298K$ (Dirichlet BC). All the other boundaries are adiabatic (Neumann BC) except the heat sink. The internal heat source uses the configuration of type A.

Case 2: ADlet sub-task. ADlet denotes a heat source system with all different Dirichlet boundary conditions for heat dissipation where one boundary is set to sine-wave distribution and the others are set to constant temperature valued $T_0 = 298K$. Besides, the internal heat source uses the configuration of type B.

Case 3: DSine sub-task. DSine denotes a heat source system with one sine-wave distributed boundaries for dissipation. All the other three boundaries are adiabatic (Neumann BC). The internal heat source uses the configuration of type C.

These three cases are used as representatives to construct the TFRD to advance the state-of-the-art of TFR-HSS task. Besides, for TFRD, the power intensity (or the maximum power intensity for gaussian power distribution) of each heat source in the system is

Table C4 Number of monitoring points for TFRD. OB, BC and OC represent on boundary, between components and on components, respectively.

| Positions | OB   | BC   | OC   | Total |
|-----------|------|------|------|-------|
| HSink     | $3 \times 4$ | 10   | $1 \times 10$ | 32    |
| ADlet     | $3 \times 4$ | 10   | $1 \times 10$ | 32    |
| DSine     | $3 \times 4$ | 10   | $1 \times 12$ | 34    |
Appendix C.2 Special Samples

Here, we construct special samples where 1/4, 1/2, 3/4, all but one of all the heat sources are with zero-power intensity. Overall, we construct five special test sets for each sub-task, namely

- **Test 1**: Samples where all the heat sources are with the same intensity. Fig. C3(b) shows examples of samples in Test 1.
- **Test 2**: Samples where 1/4 of the heat sources are with zero-power intensity and the remainder are with random selected power intensity. Fig. C3(c) shows examples of samples in Test 2.
- **Test 3**: Samples where half of the heat sources are with zero-power intensity and half are with random selected power intensity. Fig. C3(d) shows examples of samples in Test 3.
- **Test 4**: Samples where 3/4 of the heat sources are with zero-power intensity and the remainder are with random selected power intensity. Fig. C3(e) shows examples of samples in Test 4.
- **Test 5**: Samples where only one heat source is with random selected intensity and the remainder are with zero-power intensity. Fig. C3(f) shows examples of samples in Test 5.

Appendix C.3 Data Generator

The steady-state temperature field corresponding to a specific sample is calculated via FEniCS \(^1\) as the ground-truth temperature field to evaluate the performance of reconstruction methods. FEniCS is a popular open-source computing platform for solving partial differential equations (PDEs). It enables users to quickly translate scientific models into efficient finite element code. The data generator code in this work is developed based on FEniCS and released at https://github.com/shendu-sw/recon-data-generator.

The generator supports the design of heat sources, such as the shape and layout angle, as well as the design of the boundary conditions, including the heat sink and the sine-wave distributed boundary.

TFRD is generated under this developed data generator. We have provided the configuration files of the TFRD and one can generate more samples if needed. Furthermore, other researchers can generate more interesting samples of other cases to advance the state-of-the-art methods for TFR-HSS task.

\(^1\) https://fenicsproject.org/

---

**Figure B6** Network structure of FCN-1 for TFR-HSS task [7].

**Figure B7** Network structure of UNet for TFR-HSS task [7].

ranging from 0 to 30000 W/m\(^2\).
Appendix C.4 Temperature Field Reconstruction Dataset (TFRD)

To advance the state-of-the-art methods in TFR-HSS task, this work constructs the TFRD dataset 2), a new diversity, large-scale temperature field reconstruction dataset, using the proposed data generator and special sample generating strategies.

The TFRD consists of three sub-data, namely the HSink data, the ADlet data, and the DSine data, corresponding to the three sub-tasks in Subsection Appendix C.1. Examples of TFRD are shown in Fig. C4. Table C5 lists the details of training and testing samples in TFRD.

The developed TFRD considers both the diversity of sampling strategies and the completeness of the generated samples. This brings more challenges for the TFR-HSS task and the test dataset would be more suitable for promoting the state-of-the-art methods in TFR-HSS task.

Table C5 Number of training and testing samples in TFRD.

| Data  | TRAIN | TEST |
|-------|-------|------|
|       | Train | Validation | Total | 0 | 1 | 2 | 3 | 4 | 5 | Total |
| HSink | 8000  | 2000     | 10000 | 10000 | 2000 | 2000 | 2000 | 2000 | 2000 | 20000 |
| ADlet | 8000  | 2000     | 10000 | 10000 | 2000 | 2000 | 2000 | 2000 | 2000 | 20000 |
| DSine | 8000  | 2000     | 10000 | 10000 | 2000 | 2000 | 2000 | 2000 | 2000 | 20000 |

Appendix D More Results

In this section, we evaluate all these kinds of temperature field reconstruction methods mentioned before: point-based methods, vector-based methods, image-based methods and graph-based methods. For each type, we choose some representative ones as baseline for evaluation: KInterpolation, GInterpolation, polynomial regression (PR), random forest regression (RFR), Gaussian process regression (GPR), support vector regression (SVR), MLPP, RBM, and DBNs for point-based methods, MLPV, CNP, and Transformer for vector-based methods, four typical deep regression models (i.e. FCN, FPN, UNet, SegNet) combined with three deep backbone models (i.e. AlexNet, VGG16, ResNet18) for image-based methods, and the representative graph-based methods (i.e. GCN) are adopted.

2) The TFRD is downloadable at https://pan.baidu.com/s/14BipTer1SkHlRqQNhKtQ, Password: tfrd
Figure B10  Network structure of GCN for TFR-HSS task.

Figure C1  Typical boundary conditions used in TFRD. (a) Heat sink; (b) All Dirichlet; (c) Sine-wave distribution.

Appendix D.1 Experimental Setups

In our experiments, we firstly test nine kinds of point-based methods for reconstruction as before mentioned. For KInterpolation, we set the number of neighbors $k$ to 3. Gaussian kernel is used as the correlation metric, so as to GInterpolation. As for polynomial regression, the degree of polynomial features is set to 5. For random forest regression, the number of trees in the forest is set to 500. For MLPP, the structure of the network is set to ‘2-100-50-1’. For RBM, the number of hidden nodes is set to 800. For DBNs, the structure of the network is set to ‘2-250-50-10-1’.

As for vector-based methods, the structure of the network is set to ‘Input-512-512-512-Output’ for MLPV where ‘Input’ and ‘Output’ are temperature vectors with dimensions of $1 \times m$ and $n \times 1$ and $m$ is the number of monitoring points and $n$ is the number of PoIs. While for CNP, the structure of encoder network is set to ‘Input1-(2+1)-128-128-128-256’ and the structure of decoder network is set to ‘Input2-(256+2)-256-256-128-128-Output’. The ‘Input1’ describes temperature vectors combined with the position information of monitoring points and the dimension is $m \times 3$. ‘Input2’ is the output of encoder combined with the position information of PoIs and the dimension is $n \times (256 + 2)$.

For image-based methods, we just use the commonly used deep regression models with some slightly adaptive adjustment. Since the TFR-HSS task is a typical regression problem, these deep models are changed to the regression ones with $L_1$ loss for training. It should be noted that all the deep learning is implemented under the pytorch-lightning [22] deep learning framework.

For graph-based methods, we use the graph convolutional networks as [36] shows. In the experiments, the number of neighbors for each points is set to 8. Besides, ‘dense’ block is used for experiments and the number of basic blocks is set to 3.

It should be noted that for vector-based methods and graph-based methods, only ‘$50 \times 50$’ grids of points are used as PoIs in single model and we use 16 parallel models for the reconstruction of the overall temperature field.

Appendix D.2 Evaluation Metrics

To thoroughly evaluate the reconstruction performance for different methods quantitatively, this work uses the following three metrics based on the temperature field information we mainly concern about in engineering, namely the mean absolute error (MAE), the maximum of absolute error (MaxAE), the component-constrained mean absolute error (CMAE), the maximum of component-constrained absolute error (M-CAE), and the boundary-constrained mean absolute error (BMAE).

For convenience, $\Omega$, $\Omega_c$, $\Omega_b$ represent the whole heat-source domain, the area on component, and the area on boundary, respectively.
Figure C2  Heat sources with different shapes and power distributions in TFRD. (a) Uniformly distributed rectangle-like heat sources; (b) Uniformly distributed capsule-like heat sources; (c) Non-uniformly distributed circle-like heat sources.

Mean absolute error (MAE) measures the mean value of absolute error of the predicted temperature field. It can be formulated as

\[
E_{MAE} = \frac{1}{|\Omega|} \sum_{(x_i, y_j) \in \Omega} |T(x_i, y_j) - \hat{T}(x_i, y_j)|, \tag{D1}
\]

where \( \hat{T} \) describes the real temperature field obtained by numerical analysis (i.e. FEniCS) and is used as the true label for evaluation.

Maximum of absolute error (MaxAE) measures the maximum of absolute error of the predicted temperature field, and it can be calculated as

\[
E_{MaxAE} = \max_{(x_i, y_j) \in \Omega} |T(x_i, y_j) - \hat{T}(x_i, y_j)|, \tag{D2}
\]

where \( \hat{T} \) is the same as MAE.

Component-constrained mean absolute error (CMAE) computes the mean value of the absolute error over the heat-source component. Generally, it can be formulated as

\[
E_{CMAE} = \frac{1}{|\Omega_c|} \sum_{(x_i, y_j) \in \Omega_c} |T(x_i, y_j) - \hat{T}(x_i, y_j)|. \tag{D3}
\]

In the experiments, the \( \Omega_c \) can be measured by the layout matrix of the heat source system.

Maximum of component-constrained absolute error (M-CAE) describes the maximum error of the predicted temperature field over the heat-source components. It can be formulated as

\[
E_{M-CAE} = \max_{(x_i, y_j) \in \Omega_c} |T(x_i, y_j) - \hat{T}(x_i, y_j)|. \tag{D4}
\]

Boundary-constrained mean absolute error (BMAE) computes the mean value of the absolute error near the boundaries of the heat-source systems. It can be written as

\[
E_{BMAE} = \frac{1}{|\Omega_b|} \sum_{(x_i, y_j) \in \Omega_b} |T(x_i, y_j) - \hat{T}(x_i, y_j)|. \tag{D5}
\]

In the following, all the reconstruction performance of baseline methods will be evaluated under the five metrics.

Appendix D.3 Experimental Results

In this subsection, we evaluate different baseline methods on our TFRD dataset and give the corresponding results and analysis. It should be noted that all the results are the mean value from the different test sets.

Appendix D.3.1 Results with Point-based Methods

In this set of experiments, we test the performance of the point-based methods for TFR-HSS task. Tables D1-D5 illustrates the MAE, MaxAE, CMAE, M-CAE, BMAE using these point-based methods over our TFRD dataset. For all sub-data in TFRD, the results are the reconstruction performance over 10000 testing samples. For MAE, the RBM performs the best over ADlet and DSine data and RFR performs the best over the HSink data. Besides, for BMAE, we can find that RBM performs the best over ADlet and DSine data and k-nearest the best over HSink data. This means that RBM can provide good performance among these point-based methods. However, under MaxAE, the MLPP can provide better performance, this means that MLP can reduce the maximum error for TFR-HSS task. Furthermore, compared these methods under CMAE and M-CAE, RBM and PR can provide a relative better performance. This means that RBM and PR can better reconstruct the temperature field over the area on component. It should also be noted that theoretically, DBNs can provide better performance than RBM. This work only test the performance of DBN with a fixed structure. Other researchers are encouraged to try DBNs other better structures. Overall, these methods provide a different performance under different metrics. One can choose proper methods based on different requirements.
Figure C3  Examples of samples in different test sets for HSink.

Figure C4  Examples of TFRD.
| Data  | KInterpolation | GInterpolation | PR  | RFR  | GPR  | SVR  | MLPP | RBM  | DBNs |
|-------|----------------|----------------|-----|------|------|------|------|------|------|
| **HSink** |                |                |     |      |      |      |      |      |      |
| test0 | 2.1153         | 2.1141         | 2.7274 | **2.0129** | 4.0537 | 4.7118 | 2.0746 | 3.3856 | 3.5165 |
| test1 | 2.0605         | 2.0593         | 2.7043 | **1.7685** | 3.9935 | 5.0673 | 1.9285 | 3.3602 | 3.4855 |
| test2 | 1.5193         | 1.5184         | 1.9180 | **1.5072** | 2.8405 | 2.8184 | 1.7435 | 2.3765 | 2.4740 |
| test3 | 1.1177         | **1.1171**     | 1.3918 | 1.3120 | 2.0795 | 1.7597 | 1.3760 | 1.7180 | 1.7960 |
| test4 | 0.7259         | **0.7255**     | 0.8667 | 0.7335 | 1.3380 | 0.9470 | 0.9300 | 1.0635 | 1.1190 |
| test5 | 0.2796         | 0.2795         | 0.3054 | **0.2715** | 0.4841 | 0.2765 | 0.4020 | 0.3690 | 0.3910 |
| **ADlet** |                |                |     |      |      |      |      |      |      |
| test0 | 1.1229         | 1.0545         | 0.3640 | 1.3285 | 1.0178 | 1.3652 | 0.6133 | **0.2865** | 0.4103 |
| test1 | 1.0558         | 1.0555         | 0.2778 | 1.3160 | 1.0497 | 1.3880 | 0.6690 | **0.2188** | 0.4050 |
| test2 | 0.9698         | 0.9695         | 0.3404 | 1.2930 | 0.9452 | 1.2295 | 0.7075 | **0.2932** | 0.4585 |
| test3 | 0.9120         | 0.9120         | 0.3019 | 1.2515 | 0.9084 | 1.1450 | 0.7925 | **0.2676** | 0.4917 |
| test4 | 0.8545         | 0.8545         | 0.2436 | 1.2036 | 0.8704 | 1.0715 | 0.8755 | **0.2241** | 0.5217 |
| test5 | 0.7956         | 0.7955         | **0.1371** | 1.1376 | 0.8117 | 1.0075 | 0.9345 | 0.1506 | 0.5522 |
| **DSine** |                |                |     |      |      |      |      |      |      |
| test0 | 1.0256         | 1.0249         | 0.3427 | 0.7971 | 0.4634 | 1.2536 | 0.6434 | **0.3343** | 0.7345 |
| test1 | 1.0340         | 1.0378         | 0.3237 | 0.8054 | 0.5910 | 1.6408 | 0.6520 | **0.3086** | 0.7250 |
| test2 | 0.7516         | 0.7510         | 0.3250 | 0.6572 | 0.5109 | 0.8265 | 0.6010 | **0.3196** | 0.7410 |
| test3 | 0.6349         | 0.6344         | 0.3095 | 0.5947 | 0.5483 | 0.6801 | 0.5815 | **0.3070** | 0.7405 |
| test4 | 0.4608         | 0.4603         | 0.2803 | 0.5277 | 0.6176 | 0.5012 | 0.5725 | **0.2742** | 0.7385 |
| test5 | 0.3650         | 0.3646         | 0.2556 | 0.5218 | 0.6773 | 0.4252 | 0.6220 | **0.2478** | 0.7315 |
Table D2  Maximum of absolute error (K) of different point-based methods on our TFRD dataset.

| Data  | KInterpolation | GInterpolation | PR  | RFR  | GPR  | SVR  | MLPP | RBM  | DBNs  |
|-------|----------------|----------------|-----|------|------|------|------|------|-------|
| HSink |                |                |     |      |      |      |      |      |       |
| test0 | 34.375         | 34.375         | 34.996 | 38.732 | 49.737 | 60.969 | **18.963** | 38.476 | 26.475 |
| test1 | 34.222         | 34.222         | 34.833 | 39.737 | 49.227 | 61.172 | **19.234** | 38.308 | 26.313 |
| test2 | 23.983         | 23.983         | 24.419 | 26.220 | 34.614 | 41.126 | **15.259** | 26.848 | 18.515 |
| test3 | 17.240         | 17.240         | 17.582 | 18.445 | 24.923 | 28.565 | **12.123** | 19.327 | 13.327 |
| test4 | 10.563         | 10.563         | 10.787 | 10.925 | 15.294 | 16.722 | 8.7175 | 11.845 | **8.1835** |
| test5 | 3.5640         | 3.5640         | 3.6404 | 3.4555 | 4.9921 | 5.0053 | 3.8780 | 3.979 | **2.7535** |
| ADlet |                |                |     |      |      |      |      |      |       |
| test0 | 8.0353         | 7.4310         | 7.0840 | 7.6611 | 4.6250 | 8.2426 | **3.4669** | 4.2221 | 3.9329 |
| test1 | 7.5360         | 7.5235         | 4.5461 | 7.8026 | 4.9187 | 8.2100 | 3.7165 | **1.9739** | 3.7570 |
| test2 | 7.4446         | 7.4445         | 7.3387 | 8.2208 | 4.9622 | 8.8370 | **4.1060** | 4.7014 | 4.6015 |
| test3 | 7.4558         | 7.4555         | 6.8339 | 8.6888 | 5.2955 | 9.2500 | 4.6065 | **4.5034** | 4.9785 |
| test4 | 7.4660         | 7.4660         | 5.6723 | 9.2095 | 5.5571 | 9.6495 | 5.0705 | **4.0086** | 5.1756 |
| test5 | 7.4759         | 7.4760         | 3.1105 | 9.7938 | 5.5856 | 10.048 | 5.3975 | **2.8542** | 5.3065 |
| DSine |                |                |     |      |      |      |      |      |       |
| test0 | 13.047         | 13.039         | **2.3598** | 7.7505 | 3.2693 | 17.551 | 4.3344 | 3.0781 | 4.4122 |
| test1 | 13.039         | 13.031         | **2.2609** | 8.1904 | 3.8230 | 17.952 | 4.2045 | 3.0547 | 4.4065 |
| test2 | 10.968         | 10.961         | **2.5763** | 8.0775 | 3.5988 | 14.187 | 4.0480 | 3.2993 | 4.3805 |
| test3 | 9.9865         | 9.9795         | **2.6056** | 8.4518 | 3.8802 | 12.701 | 3.8940 | 3.4029 | 4.3905 |
| test4 | 8.4198         | 8.4133         | **2.6697** | 9.5488 | 4.5209 | 10.287 | 3.6375 | 3.5033 | 4.3335 |
| test5 | 7.3991         | 7.3931         | **2.7276** | 10.106 | 5.2583 | 8.6001 | 3.4545 | 3.5875 | 4.3135 |
| Data   | KInterpolation | GInterpolation | PR | RFR | GPR | SVR | MLPP | RBM | DBNs |
|--------|----------------|----------------|----|-----|-----|-----|------|-----|------|
| HSink  |                |                |    |     |     |     |      |     |      |
| test0  | 2.0705         | 2.0699         | 2.4482 | **2.0557** | 3.7212 | 4.7291 | 2.0717 | 3.1413 | 3.4240 |
| test1  | 2.0278         | 2.0272         | 2.4286 | **1.8497** | 3.6384 | 5.1010 | 1.8850 | 3.1185 | 3.3885 |
| test2  | 1.4887         | **1.4883**     | 1.7436 | 1.6675 | 2.8809 | 2.8198 | 1.8150 | 2.2080 | 2.4945 |
| test3  | 1.1085         | **1.1083**     | 1.2416 | 1.3748 | 2.3920 | 1.8587 | 1.5540 | 1.5935 | 1.8980 |
| test4  | 0.7438         | **0.7436**     | 0.7652 | 1.0370 | 1.9492 | 1.2739 | 1.2965 | 0.9805 | 1.2995 |
| test5  | 0.3676         | 0.3675         | **0.2904** | 0.5905 | 1.4343 | 0.9465 | 1.1025 | 0.3515 | 0.7370 |
| ADlet  |                |                |    |     |     |     |      |     |      |
| test0  | 0.8552         | 0.8184         | **0.2044** | 1.1565 | 0.9210 | 1.0906 | 0.5161 | 0.2102 | 0.3860 |
| test1  | 0.8194         | 0.8195         | **0.1565** | 1.1460 | 0.9128 | 1.0995 | 0.5635 | 0.1652 | 0.3602 |
| test2  | 0.7684         | 0.7685         | **0.2029** | 1.1738 | 0.8849 | 1.1055 | 0.5635 | 0.2286 | 0.4595 |
| test3  | 0.7356         | 0.7355         | **0.1924** | 1.1697 | 0.8763 | 1.1180 | 0.6155 | 0.2232 | 0.5220 |
| test4  | 0.7093         | 0.7095         | **0.1724** | 1.1775 | 0.8676 | 1.1765 | 0.6695 | 0.2134 | 0.5885 |
| test5  | 0.7191         | 0.7190         | **0.1533** | 1.2654 | 0.9257 | 1.3610 | 0.7465 | 0.2036 | 0.6442 |
| DSine  |                |                |    |     |     |     |      |     |      |
| test0  | 0.6882         | 0.6882         | **0.1980** | 0.6780 | 0.3741 | 1.1282 | 0.5520 | 0.2528 | 0.6108 |
| test1  | 0.6940         | 0.6940         | **0.1751** | 0.6596 | 0.4949 | 1.4734 | 0.5320 | 0.2180 | 0.5905 |
| test2  | 0.5099         | 0.5099         | **0.1857** | 0.5733 | 0.4539 | 0.8174 | 0.5515 | 0.2439 | 0.6229 |
| test3  | 0.4339         | 0.4339         | **0.1802** | 0.5304 | 0.5158 | 0.7059 | 0.5460 | 0.2418 | 0.6276 |
| test4  | 0.3390         | 0.3390         | **0.1643** | 0.5083 | 0.6718 | 0.5906 | 0.5255 | 0.2327 | 0.6270 |
| test5  | 0.3084         | 0.3084         | **0.1444** | 0.5965 | 0.8569 | 0.5746 | 0.5630 | 0.2253 | 0.5890 |
Table D4  Maximum of component-constrained absolute error (K) of different point-based methods on our TFRD dataset.

| Data    | KInterpolation | Interpolation | PR | RFR | GPR | SVR | MLPP | RBM | DBNs |
|---------|----------------|---------------|----|-----|-----|-----|------|-----|------|
| test0   | 30.594         | 30.594        | 28.364 | 26.518 | 28.229 | 29.965 | 26.443 | 26.051 | 13.671 | 31.409 | 24.126 |
| test1   | 12.959         | 12.959        | 10.348 | 10.386 | 12.277 | 6.0095 | 5.0932 | 5.4980 | 4.9010 | 5.134 | 4.3285 |
| test2   | 6.5416         | 6.5416        | 4.7508 | 4.2340 | 5.1534 | 5.4803 | 5.0932 | 5.4980 | 4.9010 | 5.134 | 4.3285 |
| test3   | 1.8369         | 1.8369        | 1.3894 | 1.2070 | 1.8982 | 1.4795 | 1.3885 | 1.3120 | 1.3905 | 1.3470 | 1.3120 |
| test4   | 4.3970         | 4.3970        | 3.9806 | 3.0106 | 5.1896 | 2.3850 | 1.4795 | 1.3470 | 1.3120 | 1.3905 | 1.3470 |
| test5   | 4.7087         | 4.7087        | 4.7087 | 4.7087 | 4.7087 | 4.7087 | 4.7087 | 4.7087 | 4.7087 | 4.7087 | 4.7087 |
| test0   | 2.0807         | 2.0807        | 1.0583 | 1.0583 | 1.0583 | 1.0583 | 1.0583 | 1.0583 | 1.0583 | 1.0583 | 1.0583 |
| test1   | 2.0807         | 2.0807        | 1.0583 | 1.0583 | 1.0583 | 1.0583 | 1.0583 | 1.0583 | 1.0583 | 1.0583 | 1.0583 |
| test2   | 3.6241         | 3.6241        | 3.3300 | 3.3300 | 3.3300 | 3.3300 | 3.3300 | 3.3300 | 3.3300 | 3.3300 | 3.3300 |
| test3   | 2.9428         | 2.9428        | 2.4928 | 2.4928 | 2.4928 | 2.4928 | 2.4928 | 2.4928 | 2.4928 | 2.4928 | 2.4928 |
| test4   | 1.9195         | 1.9195        | 1.3804 | 1.3804 | 1.3804 | 1.3804 | 1.3804 | 1.3804 | 1.3804 | 1.3804 | 1.3804 |
| test5   | 1.0106         | 1.0106        | 0.5606 | 0.5606 | 0.5606 | 0.5606 | 0.5606 | 0.5606 | 0.5606 | 0.5606 | 0.5606 |
| Data   | KInterpolation | GInterpolation | PR | RFR | GPR | SVR | MLPP | RBM | DBNs |
|--------|----------------|----------------|----|-----|-----|-----|------|-----|------|
|        |                |                |    |     |     |     |      |     |      |
| test0  | 1.8488         | 1.8489         | 7.0221 | 3.0301 | 6.2012 | 7.1983 | 3.2152 | 8.0296 | 7.4393 |
| test1  | 1.7835         | 1.7836         | 6.9528 | 2.9621 | 6.1921 | 7.5743 | 3.0210 | 7.9810 | 7.3960 |
| test2  | 1.3292         | 1.3293         | 4.9465 | 2.1555 | 4.2991 | 4.4373 | 2.5665 | 5.6235 | 5.2000 |
| test3  | 0.9787         | 0.9787         | 3.5933 | 1.5810 | 3.1067 | 2.8436 | 1.9520 | 4.0565 | 3.7445 |
| test4  | 0.6294         | 0.6294         | 2.2380 | 0.9875 | 1.9656 | 1.5577 | 1.2610 | 2.5020 | 2.2965 |
| test5  | 0.2371         | 0.2371         | 0.7816 | 0.3385 | 0.7026 | 0.4576 | 0.5115 | 0.8605 | 0.7700 |
|        |                |                |    |     |     |     |      |     |      |
|        |                |                |    |     |     |     |      |     |      |
| test0  | 1.2393         | 1.2084         | 1.0016 | 2.4119 | 1.6395 | 2.7287 | 1.2858 | 0.6157 | 0.7415 |
| test1  | 1.2126         | 1.2175         | 0.6483 | 2.4154 | 1.7367 | 2.8010 | 1.3635 | 0.3487 | 0.7320 |
| test2  | 1.1148         | 1.1145         | 1.0024 | 2.4186 | 1.5819 | 2.4560 | 1.4360 | 0.6641 | 0.8495 |
| test3  | 1.0544         | 1.0545         | 0.9186 | 2.4091 | 1.5780 | 2.3005 | 1.5790 | 0.6310 | 0.9280 |
| test4  | 0.9935         | 0.9935         | 0.7613 | 2.3895 | 1.5800 | 2.1595 | 1.7210 | 0.5503 | 1.0033 |
| test5  | 0.9344         | 0.9345         | 0.4345 | 2.3423 | 1.5503 | 2.0420 | 1.8175 | 0.3912 | 1.0835 |
|        |                |                |    |     |     |     |      |     |      |
| test0  | 1.5262         | 1.5258         | 0.6950 | 1.4123 | 0.8132 | 3.1592 | 1.3649 | 0.5555 | 1.3532 |
| test1  | 1.5403         | 1.5399         | 0.6556 | 1.5140 | 0.9894 | 3.7323 | 1.3655 | 0.5205 | 1.3480 |
| test2  | 1.2529         | 1.2256         | 0.7227 | 1.3577 | 0.8823 | 2.1433 | 1.2875 | 0.5679 | 1.3765 |
| test3  | 1.0937         | 1.0933         | 0.7149 | 1.3884 | 0.9403 | 1.808 | 1.2540 | 0.5667 | 1.3840 |
| test4  | 0.8874         | 0.8871         | 0.7033 | 1.5675 | 1.0214 | 1.3869 | 1.1950 | 0.5393 | 1.3970 |
| test5  | 0.7575         | 0.7572         | 0.6897 | 1.6691 | 1.0915 | 1.2009 | 1.2020 | 0.5167 | 1.4020 |
Appendix D.3.2  Results with Vector-based Methods

The comparison results of the vector-based methods over our TFRD dataset are shown in Tables D6-D7. Under the specific configurations in this work, MLPV and transformer can provide better performance than CNP. On general test samples, the vector-based methods outperform the former point-based methods while on special test samples, the point-based methods can provide better performance than vector-based methods. These vector-based methods have great potentials to improve the reconstruction performance and other researchers can design other MLPVs, CNPs as well as transformers to obtain better performance.
Table D6 MAE, MaxAE, and CMAE (K) of different vector-based methods on our TFRD dataset.

| Data | MAE   | MaxAE | CMAE |
|------|-------|-------|------|
|      | MLPV  | CNP   | Transformer | MLPV  | CNP   | Transformer | MLPV  | CNP   | Transformer |
| HSink |       |       |              |       |       |              |       |       |              |
| test0 | 0.2550 | 0.4670 | 0.3909 | 16.286 | 16.410 | 16.920 | 0.2508 | 0.4850 | 0.4007 |
| test1 | 1.0527 | 0.6810 | 3.1004 | 17.609 | 18.069 | 22.834 | 1.0481 | 0.6543 | 3.0415 |
| test2 | 0.1916 | 0.5109 | 0.4114 | 11.366 | 11.675 | 11.809 | 0.1883 | 0.5540 | 0.4434 |
| test3 | 0.2467 | 0.6169 | 0.7545 | 8.2424 | 9.1955 | 10.283 | 0.2443 | 0.6680 | 0.7814 |
| test4 | 1.1254 | 0.9903 | 3.8421 | 6.4342 | 7.3091 | 17.865 | 1.1242 | 1.0120 | 3.7647 |
| test5 | 7.9676 | 2.1179 | 18.484 | 17.065 | 9.8000 | 50.511 | 7.9727 | 2.0468 | 18.116 |
| ADlet |       |       |              |       |       |              |       |       |              |
| test0 | 0.1165 | 0.2142 | 0.1281 | 0.8102 | 1.2010 | 0.8808 | 0.1085 | 0.2705 | 0.1228 |
| test1 | 0.1171 | 0.1940 | 0.1388 | 0.8429 | 1.1013 | 0.8871 | 0.1090 | 0.2165 | 0.1364 |
| test2 | 0.1061 | 0.2299 | 0.1258 | 0.8550 | 1.2899 | 0.9334 | 0.0987 | 0.2977 | 0.1249 |
| test3 | 0.0993 | 0.2343 | 0.1331 | 0.8942 | 1.3562 | 0.9631 | 0.0922 | 0.2943 | 0.1373 |
| test4 | 0.0926 | 0.2439 | 0.1590 | 0.9323 | 1.4944 | 0.9879 | 0.0857 | 0.2878 | 0.1734 |
| test5 | 0.0862 | 0.2679 | 0.2256 | 0.9706 | 1.8677 | 1.0280 | 0.0796 | 0.2895 | 0.2607 |
| DSine |       |       |              |       |       |              |       |       |              |
| test0 | 0.1131 | 0.2625 | 0.1409 | 2.4086 | 2.7869 | 2.4317 | 0.1116 | 0.3331 | 0.1501 |
| test1 | 0.1216 | 0.2514 | 0.6039 | 2.4050 | 2.8502 | 3.5485 | 0.1191 | 0.2641 | 0.6585 |
| test2 | 0.0823 | 0.2912 | 0.1526 | 2.2313 | 2.7694 | 2.3372 | 0.0811 | 0.3734 | 0.1809 |
| test3 | 0.0711 | 0.3084 | 0.2488 | 2.1460 | 2.7521 | 2.5493 | 0.0695 | 0.3744 | 0.2917 |
| test4 | 0.0584 | 0.4053 | 1.0722 | 2.0037 | 2.8005 | 5.3000 | 0.0552 | 0.3967 | 1.1769 |
| test5 | 0.0563 | 0.5807 | 2.5157 | 1.9055 | 3.3483 | 11.017 | 0.0514 | 0.5119 | 2.7286 |
### Table D7: M-CAE and BMAE (K) of different vector-based methods on our TFRD dataset.

| Data | M-CAE | BMAE |
|------|-------|------|
|      | MLPV  | CNP  | Transformer | MLPV  | CNP  | Transformer |
| HSink |       |      |             |       |      |             |
| test0 | 5.3560 | 9.2710 | 5.4634 | **0.3125** | 0.6195 | 0.4517 |
| test1 | 6.8864 | 10.191 | 12.649 | 1.0901 | **1.0295** | 3.4909 |
| test2 | 3.7454 | 6.8442 | 4.3016 | **0.2325** | 0.6015 | 0.4135 |
| test3 | 2.8640 | 5.4441 | 4.6010 | **0.2734** | 0.7189 | 0.8079 |
| test4 | 3.7129 | 5.0322 | 13.723 | **1.1089** | 1.2413 | 4.3876 |
| test5 | 16.735 | **8.7155** | 47.974 | 7.7166 | **2.7955** | 20.024 |
| ADlet |       |      |             |       |      |             |
| test0 | **0.7110** | 1.1439 | 0.8182 | **0.1608** | 0.2150 | 0.1768 |
| test1 | 0.7341 | 1.0298 | 0.7870 | **0.1614** | 0.2563 | 0.1807 |
| test2 | **0.7181** | 1.2247 | 0.8595 | **0.1462** | 0.2212 | 0.1648 |
| test3 | 0.7395 | 1.2714 | 0.8843 | **0.1368** | 0.2381 | 0.1588 |
| test4 | 0.7654 | 1.3534 | 0.9141 | **0.1274** | 0.2739 | 0.1541 |
| test5 | **0.7931** | 1.5182 | 1.0010 | **0.1180** | 0.3444 | 0.1516 |
| Dsine |       |      |             |       |      |             |
| test0 | 1.4288 | 1.5097 | **1.3901** | 0.1515 | 0.3045 | 0.1801 |
| test1 | 1.4357 | 1.8228 | 2.6152 | **0.1631** | 0.3684 | 0.6637 |
| test2 | 1.2384 | 1.3957 | 1.2684 | **0.1272** | 0.3511 | 0.1884 |
| test3 | 1.1536 | 1.4005 | 1.5104 | **0.1186** | 0.3985 | 0.2878 |
| test4 | 1.0275 | 1.6602 | 4.7415 | **0.1101** | 0.5909 | 1.1629 |
| test5 | **0.9531** | 2.1893 | 10.164 | **0.1105** | 0.8421 | 2.6646 |

**Appendix D.3.3  Results with Image-based Methods**

In this subsection, comparisons of image-based methods over our TFRD are displayed in detail. Tables D8-D12 illustrate the reconstruction performance under the five metrics, respectively. From these results, we can find that these image-based methods can obtain a better performance under these mature deep models. However, for different data, different deep models provide different performance. For MAE, FCN-AlexNet can provide relative better performance over HSink. While over Dsine and ADlet, FCN-VGG16, FCN-ResNet18, and UNet outperform other deep models. For MaxAE, FCN-VGG16 and FCN-ResNet18 obtain the better performance than other deep methods. While for CMAE, we can find FCN-VGG16 and UNet can better reconstruct the temperature field over the area with component laid on. Using the M-CAE, it can be also noted that FCN-VGG16 can provide the smallest errors. By BMAE, FCN-VGG16 and FCN-ResNet18 can better reconstruct the temperature values of boundary areas.
| Data   | FCN-AlexNet | FCN-VGG16 | FCN-ResNet18 | UNet | FPN-ResNet18 | SegNet-AlexNet |
|--------|-------------|-----------|--------------|------|--------------|----------------|
| **HSink** |             |           |              |      |              |                |
| test0  | 0.0770      | 0.0313    | 0.1852       | 0.0424 | 1.1241    | 0.4938         |
| test1  | 1.0609      | 1.6509    | 2.1991       | 2.5572 | 4.6388     | 4.4109         |
| test2  | 0.1238      | 0.0826    | 0.1830       | 0.1022 | 0.9378     | 0.5170         |
| test3  | 0.3141      | 0.5528    | 0.9043       | 0.5358 | 1.9366     | 1.4354         |
| test4  | 0.3805      | 3.3480    | 5.0925       | 2.5424 | 6.9208     | 5.9153         |
| test5  | 5.9066      | 12.5481   | 17.0572      | 11.7915 | 24.2758   | 19.7565        |
| **ADlet** |             |           |              |      |              |                |
| test0  | 0.0197      | 0.0045    | 0.0055       | 0.0056 | 0.0323     | 0.0478         |
| test1  | 0.0209      | 0.0097    | 0.0154       | 0.0197 | 0.0474     | 0.0780         |
| test2  | 0.0265      | 0.0050    | 0.0066       | 0.0087 | 0.0351     | 0.0524         |
| test3  | 0.0323      | 0.0084    | 0.0127       | 0.0145 | 0.0489     | 0.0667         |
| test4  | 0.0430      | 0.0194    | 0.0341       | 0.0265 | 0.0915     | 0.1050         |
| test5  | 0.0655      | 0.0469    | 0.0885       | 0.0491 | 0.1923     | 0.1848         |
| **DSine** |             |           |              |      |              |                |
| test0  | 0.0405      | 0.0149    | 0.0156       | 0.0769 | 0.1174     | 0.1105         |
| test1  | 0.1043      | 0.1741    | 0.1978       | 0.2117 | 0.3738     | 0.8992         |
| test2  | 0.0651      | 0.0434    | 0.0271       | 0.1124 | 0.2016     | 0.1934         |
| test3  | 0.1000      | 0.1073    | 0.0852       | 0.1416 | 0.3727     | 0.4372         |
| test4  | 0.2644      | 0.4347    | 0.6304       | 0.2180 | 1.1573     | 1.8453         |
| test5  | 0.4865      | 0.8252    | 1.6043       | 0.2916 | 2.3816     | 3.6252         |
| Data | FCN-AlexNet | FCN-VGG16 | FCN-ResNet18 | UNet | FPN-ResNet18 | SegNet-AlexNet |
|------|-------------|-----------|--------------|------|--------------|---------------|
| HSink |
| test0 | 5.5919  | 6.0139  | 5.8690  | 1.2622  | 7.8960  | 67.501  |
| test1 | 9.2356  | 7.4870  | 11.035  | 19.432  | 17.587  | 71.362  |
| test2 | 4.5057  | 4.3918  | 4.6344  | 4.2848  | 6.2121  | 44.575  |
| test3 | 6.0788  | 3.8987  | 6.3878  | 14.421  | 8.1194  | 31.015  |
| test4 | 13.670  | 6.7744  | 15.582  | 22.949  | 25.268  |
| test5 | 22.571  | 17.719  | 33.926  | 44.575  |
| ADlet |
| test0 | 0.2561  | 0.0478  | 0.0529  | 0.2738  | 0.1880  | 13.806  |
| test1 | 0.2150  | 0.0661  | 0.1004  | 0.8137  | 0.2317  | 13.806  |
| test2 | 0.3182  | 0.0501  | 0.0620  | 0.5258  | 0.2057  | 13.806  |
| test3 | 0.3485  | 0.0654  | 0.0982  | 0.9286  | 0.2387  | 13.806  |
| test4 | 0.3986  | 0.1084  | 0.2007  | 1.5380  | 0.3306  | 13.806  |
| test5 | 0.4778  | 0.1829  | 0.3913  | 2.3261  | 0.5342  | 13.806  |
| DSine |
| test0 | 0.6174  | 0.3051  | 0.3073  | 1.0286  | 0.6721  | 18.310  |
| test1 | 1.6360  | 1.9426  | 0.8309  | 2.6648  | 1.9192  | 18.824  |
| test2 | 1.1140  | 0.6847  | 0.4012  | 1.5354  | 1.0070  | 12.736  |
| test3 | 1.8929  | 1.5803  | 0.7029  | 2.1973  | 1.5875  | 10.410  |
| test4 | 4.8252  | 5.8363  | 2.2948  | 4.4178  | 4.8246  | 7.9009  |
| test5 | 7.8327  | 11.036  | 4.1608  | 7.6280  | 10.407  | 8.1821  |
Table D10  CMAE (K) of different image-based methods on our TFRD dataset.

| Data | FCN-AlexNet | FCN-VGG16 | FCN-ResNet18 | UNet | FPN-ResNet18 | SegNet-AlexNet |
|------|-------------|-----------|--------------|------|--------------|----------------|
| **HSink** | | | | | | |
| test0 | 0.0618 | **0.0266** | 0.1791 | 0.0399 | 1.1042 | 0.0977 |
| test1 | **0.9862** | 1.6242 | 2.1129 | 2.4668 | 4.4075 | 3.9648 |
| test2 | 0.1038 | **0.0819** | 0.1750 | 0.0924 | 0.9315 | 0.2561 |
| test3 | **0.2681** | 0.5523 | 0.8329 | 0.4962 | 1.9237 | 1.2489 |
| test4 | **1.2109** | 3.3246 | 4.7873 | 2.4139 | 6.7131 | 5.8228 |
| test5 | **5.4804** | 12.475 | 16.542 | 11.169 | 23.152 | 19.887 |
| **ADlet** | | | | | | |
| test0 | 0.0200 | 0.0064 | 0.0070 | 0.0059 | 0.0285 | **0.0052** |
| test1 | 0.0218 | **0.0124** | 0.0196 | 0.0212 | 0.0484 | 0.0402 |
| test2 | 0.0272 | **0.0066** | 0.0081 | 0.0090 | 0.0333 | 0.0106 |
| test3 | 0.0336 | **0.0105** | 0.0161 | 0.0124 | 0.0527 | 0.0275 |
| test4 | 0.0458 | **0.0245** | 0.0445 | 0.0256 | 0.1068 | 0.0741 |
| test5 | 0.0713 | 0.0596 | 0.1143 | **0.0453** | 0.2307 | 0.1719 |
| **DSine** | | | | | | |
| test0 | 0.0369 | **0.0156** | 0.0165 | 0.0732 | 0.1209 | 0.0489 |
| test1 | **0.0932** | 0.1573 | 0.2047 | 0.2267 | 0.3999 | 0.8813 |
| test2 | 0.0603 | 0.0447 | **0.0284** | 0.1109 | 0.2176 | 0.1503 |
| test3 | 0.0926 | 0.1073 | **0.0879** | 0.1389 | 0.4024 | 0.4172 |
| test4 | 0.2450 | 0.3984 | 0.6657 | **0.2155** | 1.2082 | 1.9082 |
| test5 | 0.4483 | 0.7232 | 1.7327 | **0.2901** | 2.4318 | 3.8291 |
Table D11  M-CAE (K) of different image-based methods on our TFRD dataset.

| Data       | FCN-AlexNet | FCN-VGG16 | FCN-ResNet18 | UNet   | FPN-ResNet18 | SegNet-AlexNet |
|------------|-------------|-----------|--------------|--------|--------------|----------------|
| **HSink**  |             |           |              |        |              |                |
| test0      | 0.7208      | 0.5701    | **0.4912**   | 0.7091 | 1.7274       | 1.1822         |
| test1      | 4.6535      | **3.3769**| 4.2642       | 16.753 | 11.285       | 20.428         |
| test2      | 1.5439      | **0.5612**| 0.6252       | 1.7534 | 1.8217       | 2.0255         |
| test3      | 4.6925      | **1.5450**| 2.2954       | 7.2596 | 4.5517       | 5.6378         |
| test4      | 13.378      | **5.9205**| 9.8341       | 27.668 | 19.425       | 15.134         |
| test5      | 22.122      | **17.303**| 27.095       | 124.2  | 67.755       | 33.698         |
| **ADlet**  |             |           |              |        |              |                |
| test0      | 0.2165      | 0.0472    | 0.481        | 0.1777 | 0.1096       | **0.0412**     |
| test1      | 0.1848      | **0.0539**| 0.0942       | 0.3780 | 0.1561       | 0.2949         |
| test2      | 0.2696      | **0.0481**| 0.0556       | 0.2527 | 0.1277       | 0.0775         |
| test3      | 0.2883      | **0.0593**| 0.0910       | 0.2892 | 0.1795       | 0.1671         |
| test4      | 0.3095      | **0.0992**| 0.1952       | 0.5442 | 0.3061       | 0.3466         |
| test5      | 0.3502      | **0.1795**| 0.3897       | 0.8198 | 0.5106       | 0.6758         |
| **DSine**  |             |           |              |        |              |                |
| test0      | 0.2932      | 0.0822    | **0.0746**   | 0.6875 | 0.2686       | 0.3653         |
| test1      | 0.9852      | 0.6003    | **0.5613**   | 1.8759 | 1.1367       | 2.6726         |
| test2      | 0.5241      | 0.2471    | **0.1427**   | 1.1257 | 0.4927       | 0.7772         |
| test3      | 0.9592      | 0.5229    | **0.3652**   | 1.6367 | 0.9265       | 1.4179         |
| test4      | 2.7451      | **1.6201**| 1.8473       | 3.1210 | 3.1601       | 3.8681         |
| test5      | 4.5241      | **3.0286**| 3.9680       | 4.4668 | 7.0322       | 6.4485         |
| Data   | FCN-AlexNet | FCN-VGG16 | FCN-ResNet18 | UNet | FPN-ResNet18 | SegNet-AlexNet |
|--------|-------------|-----------|--------------|------|--------------|----------------|
| HSink  |             |           |              |      |              |                |
| test0  | 0.1747      | 0.0870    | 0.2643       | 0.0757 | 1.2749       | 10.150         |
| test1  | 1.4127      | 1.7106    | 2.5818       | 4.3360 | 5.9682       | 14.007         |
| test2  | 0.2360      | 0.1395    | 0.3011       | 0.2830 | 0.9430       | 6.8046         |
| test3  | 0.5357      | 0.6419    | 1.3455       | 1.3765 | 2.0534       | 5.5172         |
| test4  | 2.0229      | 3.4179    | 6.2118       | 5.3066 | 8.7634       | 7.3198         |
| test5  | 7.2527      | 12.2060   | 18.031       | 25.808 | 32.158       | 17.290         |
| ADlet  |             |           |              |      |              |                |
| test0  | 0.0061      | 0.0033    | 0.0046       | 0.0092 | 0.0562       | 1.0941         |
| test1  | 0.0128      | 0.0072    | 0.0067       | 0.0227 | 0.0515       | 1.1013         |
| test2  | 0.0082      | 0.0040    | 0.0049       | 0.0145 | 0.0519       | 1.0961         |
| test3  | 0.0115      | 0.0062    | 0.0060       | 0.0215 | 0.0467       | 1.0997         |
| test4  | 0.0196      | 0.0109    | 0.0094       | 0.0331 | 0.0420       | 1.1053         |
| test5  | 0.0382      | 0.0193    | 0.0175       | 0.0484 | 0.0419       | 1.1134         |
| DSine  |             |           |              |      |              |                |
| test0  | 0.0474      | 0.0274    | 0.0237       | 0.0826 | 0.1281       | 1.7665         |
| test1  | 0.1600      | 0.3075    | 0.2290       | 0.3741 | 0.4603       | 2.3723         |
| test2  | 0.0823      | 0.0901    | 0.0456       | 0.1349 | 0.1999       | 1.4794         |
| test3  | 0.1358      | 0.2323    | 0.1319       | 0.1906 | 0.3814       | 1.5146         |
| test4  | 0.3952      | 0.9098    | 0.7312       | 0.4392 | 1.3996       | 2.2798         |
| test5  | 0.7407      | 1.6608    | 1.6473       | 0.8412 | 3.0093       | 3.3245         |
Finally, in this set of experiments, we test the graph-based methods over our TFRD. Table D13 illustrates the reconstruction performance over graph convolutional networks under the five metrics. As former introduces, the graph-based methods can not only be used in two-dimensional heat-source systems, but also in three-dimensional systems. In the experiments, eight neighbors are used to formulate the graph correlation. As the table shows, the method can provide a performance of MAE with 0.6826K for HSink, 0.1027K for ADlet, and 0.2873K for DSine. These graph-based methods would be more flexible with high potentials for temperature field reconstruction.

**Table D13** MAE, MaxAE, CMAE, M-CAE and BMAE (K) of graph convolutional networks on our TFRD dataset.

| Data | MAE  | MaxAE | CMAE | M-CAE | BMAE |
|------|------|-------|------|-------|------|
| HSink |      |       |      |       |      |
| test0 | 0.6826 | 3.8414 | 0.6852 | 1.8717 | 0.7466 |
| test1 | 4.9126 | 10.030 | 4.8890 | 6.9999 | 4.8519 |
| test2 | 1.0231 | 6.6219 | 1.0258 | 2.6458 | 1.1582 |
| test3 | 2.2914 | 10.145 | 2.2890 | 4.3277 | 2.4307 |
| test4 | 6.7385 | 16.628 | 6.7140 | 9.8772 | 6.8340 |
| test5 | 16.934 | 28.441 | 16.857 | 22.018 | 16.869 |
| ADlet |      |       |      |       |      |
| test0 | 0.1027 | 0.3921 | 0.0975 | 0.3729 | 0.1209 |
| test1 | 0.1730 | 0.5345 | 0.1850 | 0.5062 | 0.1387 |
| test2 | 0.1075 | 0.4024 | 0.1051 | 0.3794 | 0.1324 |
| test3 | 0.1350 | 0.4330 | 0.1430 | 0.4111 | 0.1460 |
| test4 | 0.2208 | 0.6111 | 0.2565 | 0.6031 | 0.1658 |
| test5 | 0.4610 | 1.2672 | 0.5654 | 1.2663 | 0.2056 |
| DSine |      |       |      |       |      |
| test0 | 0.2873 | 13.039 | 0.2250 | 0.7892 | 0.4090 |
| test1 | 0.9811 | 11.156 | 0.9897 | 1.6190 | 0.9883 |
| test2 | 0.4736 | 13.820 | 0.4194 | 1.0718 | 0.5764 |
| test3 | 0.7446 | 13.766 | 0.6997 | 1.4152 | 0.8069 |
| test4 | 2.0323 | 15.159 | 2.0613 | 3.0374 | 1.9459 |
| test5 | 3.7082 | 17.774 | 3.8575 | 5.3327 | 3.4545 |

**Appendix D.3.5 Comparisons between Different Metrics**

In this subsection, we make deep comparisons between different metrics of baseline methods. Due to page limitation, this work mainly lists the comparisons over HSink. We compare these metrics under four classes of baselines, namely M-CAE and MaxAE, CMAE, BMAE and MAE, MaxAE and MAE, M-CAE and CMAE.

**Comparisons between M-CAE and MaxAE.** Fig. D1 shows the comparison results of different methods. From the figure, we can find that the error by M-CAE is far lower than that by MaxAE. This means the area on component can be better reconstructed by these methods. However, it should also be noted that over test 5, the reconstruction methods cannot work well on the whole system and the error under M-CAE is approach to that under MAE.

**Comparisons among CMAE, BMAE and MAE.** Fig. D2 shows the comparison results of representative methods. For point-based methods, the values of BMAE is larger than that of MAE and the value of MAE is larger than that of CMAE. This means that point-based methods can work better on areas with components laid on and in contrast cannot work well on the boundary. For vector and graph-based methods, the methods present similar performance on the three metrics. While for image-based methods, we can obtain
similar conclusions except the SegNet-AlexNet. The SegNet-AlexNet seems cannot work well on the boundary area and provide poor temperature field on the boundary.

Comparisons between MaxAE and MAE, M-CAE and CMAE. Fig. D3 and D4 presents the comparison results, respectively. From Fig. D3, we can find that the errors of predicted temperature values of different points in the system present large variance. Generally, most of these methods can provide an accurate average temperature prediction. However, the largest predicted error in the system can be more than ten times than the average error. Since the area on component is usually what we care most, we present the comparisons of the component area in Fig. D4. The prediction divergence is alleviated on the component area especially with the image-based methods where the M-CAE value is almost equal to the CMAE value.

References

1. Badrinarayanan, Vijay and Kendall, Alex and Cipolla, Roberto. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence, 2017, 39(12): 2481-2495
2. Z. Gong, P. Zhong, Y. Yu, et al. A CNN with multiscale convolution and diversified metric for hyperspectral image classification. IEEE Trans. Geosci. Remote Sens., 2019, 57(6): 3599-3618
3. Z. Gong, P. Zhong, W. Hu. Statistical loss and analysis for deep learning in hyperspectral image classification. IEEE Trans. Neural Netw. Learn. Syst., 2021, 32(1): 322-333
4. Z. Gong, W. Hu, X. Du, et al. Deep manifold embedding for hyperspectral image classification. IEEE Trans. Cybern., 2021
5. M. Grujicic, C. L. Zhao, E. C. Dusel. The effect of thermal contact resistance on heat management in the electronic packaging. Applied Surface Science, 2005, 246: 290-302
6. M. Garnelo, D. Rosenbaum, C. Maddison, et al. Conditional Neural Processes. In ICML, 2018: 1704-1713
7. Z. Gong, W. Zhou, W. Peng, et al. Physics-Informed Deep Reversible Regression Model for Temperature Field Reconstruction of Heat-Source Systems. arXiv preprint arXiv:2106.11929, 2021
8. S. Haykin. Neural Networks and Learning Machines. New York: Prentice Hall, 2009
9. Z. Wu, S. Pan, F. Chen, et al. A comprehensive survey on graph neural networks. IEEE Trans. Neural Netw. Learn. Syst., 2020
10. Z. Li, N. Kovachki, K. Azizzadenesheli, et al. Fourier neural operator for parametric partial differential equations. In: ICLR, 2021: 1-16
11. E. Laloya, O. Lucia, H. Sarnago, et al. Heat management in power converters: From state of the art to future ultrahigh efficiency systems. IEEE TPEL, 2015, 31(11): 7896-7908
12. R. G. Zhou, W. Hu, P. Fan, et al. Quantum realization of the bilinear interpolation method for neqr. Scientific Reports, 2017, 7(1): 1-17
13. D. Gonzalez Cuadrado, A. Marconnet, G.R. G. Paniagua. Inverse conduction heat transfer and kriging interpolation applied to temperature sensor location in microchips. JEP, 2018, 140(1): 010905
14. K. V. L. Narayana, V. N.D. Kumar. Development of an intelligent temperature transducer. IEEE Sensors Journal, 2016, 16(12): 4696-4703
15. I. Ferain, C. A. Colinge, J. P. Colinge. Multigate transistors as the future of classical metal-oxide-semiconductor field-effect transistors. Nature, 2011, 479(7373): 310-316
16. A. L. Moore, L. Shi. Emerging challenges and materials for thermal management of electronics. Materials Today, 2014, 17(4): 163-174
17. J. P. Holman. Heat transfer. New York: McGrawHill, 2002
18. H. Saleh, I. Hashim. Conjugate natural convection in a porous enclosure with non-uniform heat generation. TPM, 2012, 94(3):
Figure D4  Comparisons between M-CAE and CMAE.