# Estimation of Deterioration Levels of Transmission Towers via Deep Learning Maximizing Canonical Correlation between Heterogeneous Features

Keisuke Maeda, *Student Member, IEEE,* Sho Takahashi, *Member, IEEE,* Takahiro Ogawa, *Senior Member, IEEE,* and Miki Haseyama, *Senior Member, IEEE,*

***Abstract*—This paper presents estimation of deterioration levels of transmission towers via deep learning maximizing the canonical correlation between heterogeneous features. In the proposed method, we newly construct a correlation-maximizing deep extreme learning machine based on a local receptive field (CMDELM-LRF). For accurate deterioration level estimation, it is necessary to obtain semantic information that effectively represents deterioration levels. However, since the amount of training data for transmission towers is small, it is difficult to perform feature transformation by using many hidden layers such as general deep learning methods. In CMDELM-LRF, one hidden layer, which maximizes the canonical correlation between visual features and text features obtained from inspection text data, is newly inserted. Specifically, by using projections obtained by maximizing the canonical correlation as weight parameters of the hidden layer, feature transformation for extracting semantic information is realized without designing many hidden layers. This is the main contribution of this paper. Consequently, CMDELM-LRF realizes accurate deterioration level estimation from a small amount of training data.**

***Index Terms*—Deterioration level estimation, deep extreme learning machine, canonical correlation analysis.**

## I. Introduction

ALL countries have critical infrastructures such as power grids, railways, tunnels, bridges and transmission towers. In order to maintain these infrastructures, visual inspection has usually been performed by inspectors. Visual inspection is a labor-intensive and time-consuming process [1]. In order to reduce costs, new techniques for supporting maintenance inspection are required [2]–[6]. Some methods have been proposed for supporting various inspection tasks for infrastructures including distress classification [7], [8], detection of specific distresses [9]–[17], analysis of surface status [18]–[20] and deterioration level estimation [21], [22].

Support for maintenance inspection of transmission towers is important since inspection is performed by inspectors ascending towers, which is a dangerous inspection task [23]. Transmission towers are constructed by using galvanized steel and coated with rust-preventive paint to prevent corrosion [24]. However, recoating is necessary because the rust-preventive paint deteriorates over time [21]. Thus, inspectors have to determine the levels of deterioration of the transmission towers by observing the surfaces of towers and estimating the levels of deterioration based on their experience and knowledge [25], [26]. However, determination of the levels of deterioration might include errors due to ambiguity in inspectors' decision. Therefore, automatic and quantitative analysis of the deterioration levels is necessary by using machine learning technology.

Many researchers have proposed methods for estimating deterioration levels [21], [22] of infrastructures. Although these methods estimate deterioration levels automatically, estimation performance is limited, and we should note the following points.

1) The classifiers used in the above methods are traditional ones, Support Vector Machines (SVMs) [29]. Recently, it has been reported that deep learning methods realize high classification performance, and deep learning-based classification methods for some tasks such as crack detection have been proposed in the civil engineering field [30]. Thus, the development of deep learning-based methods for estimation of deterioration levels is desirable.
2) The above methods only use visual characteristics of the towers. In actual maintenance inspection, inspectors not only take images of deterioration parts but also record text data about the structure under inspection such as the construction date, location and height of the tower. Since these data would contribute to the improvement of classification performance, collaborative use of images and text data is necessary.

In order to overcome these problems, we focus on a deep learning-based estimation method that is realized by using both images and inspection text data. In image recognition fields, several image classification methods with high performance have been proposed such as Convolutional Neural Network (CNN) [30]–[32], which requires a large number of training images. When it is difficult to prepare many training images, some researchers use CNN-fine-tuning instead of CNN trained from scratch. However, it has been reported that the estimation performance of CNN-fine-tuning might not be sufficient when

K. Maeda, T. Ogawa and M. Haseyama are with the Graduate School of Information Science and Technology, Hokkaido University, Sapporo, 060-0814, JAPAN. S. Takahashi is with Faculty of Engineering, Hokkaido University, Sapporo, 060-8628, JAPAN.

This work was partly supported by JSPS KAKENHI Grant Numbers JP18J10373 and JP17H01744 and Global Station for Big Data and Cyber Security, a project of Global Institution for Collaborative Research and Education at Hokkaido University. In this research, we utilized the data that were provided by Tokyo Electric Power Company Research Institute.

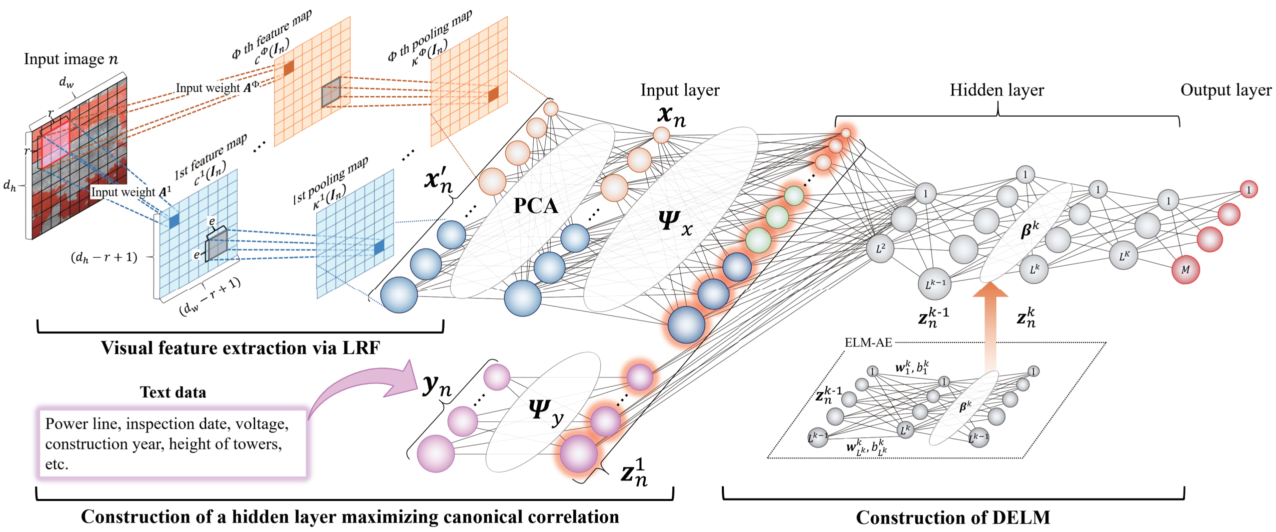


**Fig. 1.** An overview of the proposed method, which consists of three procedures. The first procedure is visual feature extraction using LRF [27]. The second procedure is construction of a hidden layer that maximizes the canonical correlation between visual and text features, which is the main contribution of this paper. The third procedure is construction of the classifier of DELM [28].

it is applied to a task for which it was not designed, that is, in a case in which technical data with a property different from that of the pre-trained data are used [33], [34]. We define technical data as data requiring professional knowledge and experience. In fact, similar tendencies have been observed, as was also confirmed by experiments for which results are shown in this paper. Therefore, a new approach based on deep learning that can effectively handle a small amount of technical data is necessary.

In this paper, we focus on ELM [35] series, which have attracted much attention recently. The number of parameters used in ELM series is small. Thus, it is not necessarily to calculate optimal parameters from a huge amount of training data. Thus, they can be trained by using a small amount of training data. Furthermore, in order to improve the performance from a small number of training images, we newly consider the relationships between heterogeneous features, i.e., images and text data. In most deep learning techniques, many middle layers are necessary for transforming visual information to semantic information. Since the number of parameters to be tuned becomes larger, a large amount of training data is required. On the other hand, we focus on canonical correlation [36] for extracting semantic information with fewer middle layers. In [37], Yeh et al. reported that canonical variates obtained by calculating the canonical correlation between heterogeneous sets of features have better discriminative performance than original features if the heterogeneous sets have semantic relevancy. Thus, based on projection using the canonical correlation, it is expected that visual information can be directly converted to semantic information. Then we can calculate new features that are suitable for representing deterioration levels without preparing a large amount of training data.

In this paper, we present deterioration level estimation via deep learning maximizing the canonical correlation between heterogeneous features. In the proposed method, we use a newly constructed correlation-maximizing deep extreme learning machine based on a local receptive field (CMDELM-LRF) as shown in Fig. 1. CMDELM-LRF is an improved version of DELM-LRF [38], which is our previously reported method. In CMDELM-LRF, we insert a hidden layer that can maximize the canonical correlation between visual features and text features obtained from text data. Specifically, the parameters of the hidden layers correspond to projections obtained by maximizing the canonical correlation between visual features and text features. Thus, by using the obtained projections as weight parameters of the hidden layer of CMDELM-LRF, it becomes feasible to obtain semantic information without designing many hidden layers. The main contribution of the proposed method is the construction of this deep learning framework including the new hidden layer that is capable of feature transformation in consideration of the canonical correlation between heterogeneous features.

This paper is organized as follows. The proposed method is presented in Section II. Experimental results for verifying the effectiveness of the proposed method are shown in Section III. Finally, concluding remarks are presented in Section IV. For smooth explanation of the proposed method, abbreviations used in this paper are shown in Table I.

## II. Correlation-Maximizing Deep Extreme Learning Machine with Local Receptive Field

In this section, we explain the automatic estimation of deterioration levels via CMDELM-LRF. The proposed method consists of three procedures as shown in Fig. 1. First, visual features are automatically extracted from images of the surfaces of the transmission towers based on LRF [27]. Second,

**TABLE I.** Abbreviations used in this paper.

| Abbreviations | Official name |
|---|---|
| CMDELM-LRF | Correlation-Maximizing Deep Extreme Learning Machine-Local Receptive Field |
| SVM | Support Vector Machine |
| CNN | Convolutional Neural Network |
| ELM | Extreme Learning Machine |
| CCA | Canonical Correlation Analysis |
| PCA | Principal Component Analysis |
| LRF | Local Receptive Field |
| DELM | Deep Extreme Learning Machine |
| SVD | Singular Value Decomposition |
| ELM-AE | Extreme Learning Machine-Auto Encoder |
| KELM | Kernel Extreme Learning Machine |
| DELM-LRF | Deep Extreme Learning Machine-Local Receptive Field |
| ELM-LRF | Extreme Learning Machine-Local Receptive Field |

a hidden layer that maximizes the canonical correlation of visual and text features is constructed. Third, a DELM-based classifier [28] is constructed. Then we can obtain deterioration levels based on outputs from the output layer of DELM.

### *A. Feature Extraction Based on LRF*

Given a training image $n$ ($n = 1, 2, ..., N; N$ being the number of training images), we extract visual features from an input matrix $\boldsymbol{I}_n \in \mathbb{R}^{d_h \times d_w}$ corresponding to image $n$. Regarding the color channel, we perform the same processing as in [27]. In order to extract visual features, the proposed method performs two procedures, generation of feature maps and pooling maps, as shown in Fig. 1.

First, we randomly generate an initial weight matrix $\hat{\boldsymbol{A}} \in \mathbb{R}^{r^2 \times \Phi}$. Note that $r \times r$ means the size of the receptive field, and $\Phi$ is the number of feature maps. We orthogonalize the initial weight matrix $\hat{\boldsymbol{A}}$ using singular value decomposition (SVD), and the orthogonal vector $\hat{\boldsymbol{a}}^{\phi} \in \mathbb{R}^{r^2}$ ($\phi = 1, 2, ..., \Phi$) is calculated, where $\hat{\boldsymbol{A}} = [\hat{\boldsymbol{a}}^1, \hat{\boldsymbol{a}}^2, ..., \hat{\boldsymbol{a}}^{\Phi}]$. In the case of $r^2 < \Phi$, we perform the following steps: 1) $(\hat{\boldsymbol{A}})^{\top}$ is orthogonalized via SVD and 2) transposed back, which is the same manner as that in [27]. Thus, an input weight matrix $\boldsymbol{A}^{\phi} \in \mathbb{R}^{r \times r}$, which corresponds to $\hat{\boldsymbol{a}}^{\phi}$ column-wisely, for the $\phi$ th feature map is obtained. By using the obtained input weight matrix $\boldsymbol{A}^{\phi}$, the $\phi$ th feature map $c^{\phi}(\boldsymbol{I}_n)$ is calculated as

$$\begin{aligned} c^{\phi}_{i,j}(\boldsymbol{I}_n) &= \sum_{s=1}^{r} \sum_{u=1}^{r} \boldsymbol{I}_n^{(i+r-s, j+r-u)} \times \boldsymbol{A}^{\phi}_{s,u}, \quad (1) \\ i &= 1,\ 2,\ ...\ , (d_h - r + 1), \\ j &= 1,\ 2,\ ...\ , (d_w - r + 1), \end{aligned}$$

where $c^{\phi}_{i,j}(\boldsymbol{I}_n)$, $\boldsymbol{I}_n^{(i+r-s,j+r-u)}$ and $\boldsymbol{A}^{\phi}_{s,u}$ represent the $(i, j)$ th element of $c^{\phi}(\boldsymbol{I}_n)$, $(i+r-s, j+r-u)$ th element of $\boldsymbol{I}_n$ and $(s, u)$ th element of $\boldsymbol{A}^{\phi}$, respectively. Consequently, the feature map $c^{\phi}(\boldsymbol{I}_n)$ with a size of $(d_h - r + 1) \times (d_w - r + 1)$ is calculated.

Second, we calculate pooling maps by using the obtained feature maps. The pooling size used in the proposed method is $e \times e$, and the size of the pooling map is the same as that of size with the feature map $(d_h - r + 1) \times (d_w - r + 1)$. By using a square/square-root pooling, the $\phi$ th pooling map $\kappa^{\phi}(\boldsymbol{I}_n)$ is calculated as:

$$\begin{aligned} \kappa^{\phi}_{p,q}(\boldsymbol{I}_n) &= \sqrt{\sum_{i=p-e}^{p+e} \sum_{j=q-e}^{q+e} \left\{ c^{\phi}_{i,j}(\boldsymbol{I}_n) \right\}^2}, \quad (2) \\ p &= 1,\ 2,\ ...\ , (d_h - r + 1), \\ q &= 1,\ 2,\ ...\ , (d_w - r + 1), \end{aligned}$$

where $\kappa^{\phi}_{p,q}(\boldsymbol{I}_n)$ represents the $(p, q)$ th element of $\kappa^{\phi}(\boldsymbol{I}_n)$. Note that $c^{\phi}_{i,j}(\boldsymbol{I}_n)$ is the zero-padded feature map. Square/square-root pooling was also used in [39], [40], and its effectiveness has been verified. We can obtain a visual feature vector $\boldsymbol{x}'_n \in \mathbb{R}^{(d_h-r+1)(d_w-r+1)\Phi}$ by aligning each pixel's value of all pooling maps. Finally, since the dimension of the visual feature vector $\boldsymbol{x}'_n$ is much higher than the number of training images, we obtain the feature vector $\boldsymbol{x}_n \in \mathbb{R}^{d_x}$ by performing principal component analysis (PCA) [41].

### *B. Construction of Hidden Layer Maximizing Canonical Correlation*

First, we calculate text features from text data. An example of text data is shown in Table II. Inspectors record transmission lines, type of towers, salt damage, area, inspection date, voltage, construction date, height of towers and coating year as text data. For transmission lines, type of towers, salt damage and area, since inspection records are discrete values, we obtain binary feature vectors whose element corresponding to the inspection record becomes 1. Specifically, the dimension of the binary features is $\sum_{t=1}^{4} D_t$ as shown in Table II. On the other hand, for the other five inspection items, we describe a corresponding inspection record as the element of features. Finally, we obtain text feature vectors $\boldsymbol{y}_n \in \mathbb{R}^{d_y}$ ($d_y = \sum_{t=1}^{4} D_t + 5$) by aligning the above features.

Second, we construct a hidden layer that maximizes the canonical correlation between visual and text features. In order to calculate the hidden layer's weight matrices that can consider the relationship between these features, we obtain projection matrices $\boldsymbol{\Psi}_x$ and $\boldsymbol{\Psi}_y$ by applying canonical correlation analysis (CCA) [36] to the visual feature matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N] \in \mathbb{R}^{d_x \times N}$ and text feature matrix

**TABLE II.** An example of text data. The transmission line represents the name of the line, and the salt damage represents the degree of damage due to salt. The area represents the zone of towers such as a coastal zone.

| Inspection item | Inspection record | Num. of dimension |
|---|---|---|
| Transmission lines | A, B, ... | $D_1$ |
| Type of towers | Angle towers, pipe towers, ... | $D_2$ |
| Salt damage | A', B', C', ... | $D_3$ |
| Area | C, D, ... | $D_4$ |
| Inspection date | 03/10/2015, 03/03/2014, ... | 1 |
| Voltage (kV) | 66, 275, 154, ... | 1 |
| Construction date | 1966, 1977, 1983, ... | 1 |
| Height of towers (m) | 56.4, 73.5, 72, ... | 1 |
| Coating year | 2004, 2001, ... | 1 |
| Sum | - | $d_y = \sum_{t=1}^{4} D_t + 5$ |

$\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_N] \in \mathbb{R}^{d_y \times N}$. Specifically, we seek to maximize the following objective function:

$$(\hat{\boldsymbol{\psi}}_x, \hat{\boldsymbol{\psi}}_y) = \arg\max_{\boldsymbol{\psi}_x, \boldsymbol{\psi}_y} \frac{\boldsymbol{\psi}_x^{\mathrm{T}} \boldsymbol{C}_{XY} \boldsymbol{\psi}_y}{\sqrt{\boldsymbol{\psi}_x^{\mathrm{T}} \boldsymbol{C}_{XX} \boldsymbol{\psi}_x} \sqrt{\boldsymbol{\psi}_y^{\mathrm{T}} \boldsymbol{C}_{YY} \boldsymbol{\psi}_y}}, \tag{3}$$

where $\boldsymbol{\psi}_x$ and $\boldsymbol{\psi}_y$ are projection vectors, and

$$\boldsymbol{C}_{XY} = \boldsymbol{X}\boldsymbol{Y}^{\top}, \tag{4}$$

$$\boldsymbol{C}_{XX} = \boldsymbol{X}\boldsymbol{X}^{\top}, \tag{5}$$

$$\boldsymbol{C}_{YY} = \boldsymbol{Y}\boldsymbol{Y}^{\top}. \tag{6}$$

In order to maximize Eq. (3), we solve the following Lagrange problem:

$$\mathcal{L}(\boldsymbol{\psi}_x, \boldsymbol{\psi}_y) = \boldsymbol{\psi}_x^{\top} C_{XX} \boldsymbol{\psi}_y - \frac{\lambda_x}{2}(\boldsymbol{\psi}_x^{\top} C_{XX} \boldsymbol{\psi}_x - 1) - \frac{\lambda_y}{2}(\boldsymbol{\psi}_y^{\top} C_{YY} \boldsymbol{\psi}_y - 1), \tag{7}$$

where $\lambda_x = \lambda_y$ ($= \lambda$), and $\lambda$ is defined below. Then we solve the following eigenvalue problems:

$$C_{XX}^{-1} C_{XY} C_{YY}^{-1} C_{XY}^{\top} \boldsymbol{\psi}_x = \lambda^2 \boldsymbol{\psi}_x, \tag{8}$$

$$C_{YY}^{-1} C_{XY}^{\top} C_{XX}^{-1} C_{XY} \boldsymbol{\psi}_y = \lambda^2 \boldsymbol{\psi}_y, \tag{9}$$

where $\lambda$ corresponds to the eigenvalue of this problem. Since we can obtain multiple eigenvalues $\lambda$ and their corresponding eigenvectors $\boldsymbol{\psi}_x$ and $\boldsymbol{\psi}_y$ as solutions of the above problem, we obtain the projection matrices as $\boldsymbol{\Psi}_x = [\boldsymbol{\psi}_x^1, \boldsymbol{\psi}_x^2, ..., \boldsymbol{\psi}_x^{d_{cca}}]^{\top} \in \mathbb{R}^{d_{cca} \times d_x}$ and $\boldsymbol{\Psi}_y = [\boldsymbol{\psi}_y^1, \boldsymbol{\psi}_y^2, ..., \boldsymbol{\psi}_y^{d_{cca}}]^{\top} \in \mathbb{R}^{d_{cca} \times d_y}$ by aligning the $d_{cca}$ eigenvectors, respectively. By setting the obtained projection matrices to the hidden layer between LRF and DELM as shown in Fig. 1, we can obtain the projected features $\hat{\boldsymbol{x}}_n \in \mathbb{R}^{d_{cca}}$ and $\hat{\boldsymbol{y}}_n \in \mathbb{R}^{d_{cca}}$, which can consider their relationships, as follows:

$$\hat{\boldsymbol{x}}_n = \boldsymbol{\Psi}_x \boldsymbol{x}_n, \tag{10}$$

$$\hat{\boldsymbol{y}}_n = \boldsymbol{\Psi}_y \boldsymbol{y}_n. \tag{11}$$

In most deep learning methods, transforming visual information to semantic information is realized by using many middle layers. Then since the number of parameters to be tuned is large, we have to prepare a large amount of training data for avoiding over-fitting. On the other hand, directly transforming visual information to semantic information is realized by considering the canonical correlation between visual and text features in our method. Thus, since the number of hidden layers can be set to a smaller number, we can construct networks by using a small amount of training data.

In our method, we use visual and text features. Since the distress images include unnecessary regions such as backgrounds, the visual features contain many noisy information. On the other hand, since the text features are calculated based on actual maintenance inspection recorded by inspectors, they are very high quality features. In order to calculate features with higher representation ability using visual and text features, it is necessary to project these heterogeneous features to a comparable feature space. Therefore, we focus on CCA, which is one of the most general methods which make it possible to project two different features to the comparable feature space. Furthermore, we can perform end-to-end learning of both feature transformation and classification since the projection matrix maximizing the canonical correlation obtained by CCA can be integrated into the neural network. This is the reason why we choose CCA.

### C. Deterioration Level Estimation Based on DELM

Given feature vectors $\hat{\boldsymbol{x}}_n$ and $\hat{\boldsymbol{y}}_n$, $z_n^1 = [\hat{\boldsymbol{x}}_n^{\top}, \hat{\boldsymbol{y}}_n^{\top}]^{\top}$ ($n = 1, 2, ..., N$) is input into DELM. DELM consists of one input layer, $K$ hidden layers and one output layer, that is, the number of layers of DELM is $K + 2$ as shown in Fig. 1. The aim of training DELM is calculation of a weight matrix between the $(k-1)$ th layer and $k$ th layer. Specifically, **(I)** in the case of $k$ being smaller than $K+1$, the weight matrix is calculated by using ELM-Auto Encoder (ELM-AE), which is an unsupervised learning method, and **(II)** in the case of $k = K+1$, the weight matrix is calculated in the same manner as that by ELM [35], which is a supervised learning method.

**(I)** $k = 2, 3, ..., K$

In DELM, the relationship between the $k$ th hidden layer's output matrix $\boldsymbol{Z}^k = [z_1^k, z_2^k, ..., z_N^k]$ and the $(k-1)$ th hidden layer's output matrix $\boldsymbol{Z}^{k-1}$ can be obtained as follows:

$$z_n^k = G(\boldsymbol{\beta}^k z_n^{k-1}), \tag{12}$$

where $\boldsymbol{Z}^k$ is obtained from training data and $\boldsymbol{\beta}^k \in \mathbb{R}^{L^k \times L^{k-1}}$ is a weight matrix between the $k$ th and $(k-1)$ th hidden layers, and $G$ is a sigmoid function as the activation function.

In order to calculate the weight matrix $\boldsymbol{\beta}^k$, we construct ELM-AE layer by layer. The ELM-AE network model, which consists of three layers, an input layer, a hidden layer and an output layer, is shown in the lower right part of Fig. 1. These layers have $L^{k-1}$ input nodes, $L^k$ hidden nodes and $L^{k-1}$ output nodes, respectively. Since ELM-AE is an auto encoder, input features are equal to output features. Given an input vector $\boldsymbol{z}_n^{k-1} \in \mathbb{R}^{L^{k-1}}$, the outputs $\boldsymbol{h}_n^k \in \mathbb{R}^{L^k}$ of hidden layers in ELM-AE can be obtained as

$$\boldsymbol{h}_n^k = G(\boldsymbol{W}^k \boldsymbol{z}_n^{k-1} + \boldsymbol{b}^k), \tag{13}$$

$$\boldsymbol{W}^k (\boldsymbol{W}^k)^\top = \boldsymbol{I}, \tag{14}$$

$$(\boldsymbol{b}^k)^\top \boldsymbol{b}^k = 1. \tag{15}$$

ELM-AE has orthogonal random weight $\boldsymbol{W}^k = [\boldsymbol{w}_1^k, \boldsymbol{w}_2^k, ..., \boldsymbol{w}_{L^k}^k]^\top$ and random bias $\boldsymbol{b}^k = [b_1^k, b_2^k, ..., b_{L^k}^k]^\top$.

Thus, by using an input matrix $\boldsymbol{Z}^{k-1}$ of ELM-AE and an output matrix $\boldsymbol{H}^k = [\boldsymbol{h}_1^k, \boldsymbol{h}_2^k, ..., \boldsymbol{h}_N^k]^\top \in \mathbb{R}^{N \times L^k}$ of the ELM-AE's hidden layer, the output weight $\boldsymbol{\beta}^k$ can be derived as follows:

$$\boldsymbol{\beta}^k = \Big( \frac{\boldsymbol{I}}{C_1} \sum_{l^k=1}^{L^k} \mathrm{KL}(\rho \| \hat{\rho_{l^k}}) + (\boldsymbol{H}^k)^\top \boldsymbol{H}^k \Big)^{-1} (\boldsymbol{H}^k)^\top (\boldsymbol{Z}^{k-1})^\top, \tag{16}$$

where $\mathrm{KL}(\rho \| \hat{\rho}_{l^k}) = \rho \log \frac{\rho}{\hat{\rho}_{l^k}} + (1-\rho) \log \frac{1-\rho}{1-\hat{\rho}_{l^k}}$ is the KL divergence. It has been reported that auto-encoders can obtain features with high representation ability by regularizing the hidden layers' representation to be sparse [42]. KL divergence is often used as a regularization term for the sparse representation of auto-encoders. The parameter $\hat{\rho}_{l^k}$ means the average activation value (averaged over the training set) of the hidden node $l^k$ of the $k$ th hidden layer. We regularize the hidden layer representation to be sparse by a pre-determine small value $\rho$ (sparsity parameter), which is the desired sparseness, in such a way that the KL divergence encourages the above average activation $\hat{\rho}_{l^k}$ to be small. Since KL divergence does not remove the hidden nodes but controls the nodes, the activated nodes according to the input data also changes. Since it is possible to extract important information about which node was activated, the representation ability is improved [42]. Furthermore, $C_1$ is a regularization parameter.

Consequently, we can obtain each layer's output weight $\boldsymbol{\beta}^k$ $(k = 1, 2, ..., K)$ of DELM by using the output weight $\boldsymbol{\beta}^k$ of the ELM-AE. Although general deep learning methods determine parameters of networks by using the back propagation approach, which has high computation costs and requires a large number of training images, the proposed method determines the weight matrix by using ELM-AE, which is layer-by-layer unsupervised learning. In addition, the parameters $\boldsymbol{W}^k$ and $\boldsymbol{b}^k$ are calculated on the basis of random values. Thus, our method does not require a large number of training images and the computation costs are smaller.

**(II)** $k = K + 1$

The output weight matrix $\boldsymbol{\beta}^{K+1}$ between the $K$ th hidden layer and the output layer is calculated by general ELM, which is a supervised learning method. We try to minimize the training error $\boldsymbol{\xi}_n = [\xi_{n,1}, \xi_{n,2}, ..., \xi_{n,M}]^\top$ ($M$ being the number of output nodes in the $M$-class problem) as well as the output weights.

$$\begin{aligned} \min_{\boldsymbol{\beta}^{K+1} \in \mathbb{R}^{M \times L^{K+1}}} R &= \frac{1}{2} \|\boldsymbol{\beta}^{K+1}\|_F^2 + \frac{C_2}{2} \sum_{n=1}^{N} \|\boldsymbol{\xi}_n\|^2, \\ \text{s.t. } \boldsymbol{\beta}^{K+1} \boldsymbol{z}_n^{K+1} &= \boldsymbol{t}_n - \boldsymbol{\xi}_n, \end{aligned} \tag{17}$$

where $\boldsymbol{t}_n = [t_{n,1}, t_{n,2}, ..., t_{n,M}]^\top$ is a vector whose $m$ th element is one, while the other elements become zero if the original true class label is $m$. Furthermore, $C_2$ is a regularization parameter. According to [43], Eq. (17) is equivalent to solving the following optimization problem based on the Karush-Kuhn-Tucker theorem:

$$\begin{aligned} \min_{\boldsymbol{\beta}^{K+1} \in \mathbb{R}^{M \times L^{K+1}}} \tilde{R} = \frac{1}{2} \|\boldsymbol{\beta}^{K+1}\|_F^2 + \frac{C_2}{2} \sum_{n=1}^{N} \|\boldsymbol{\xi}_n\|^2 \\ - \sum_{n=1}^{N} \sum_{m=1}^{M} \alpha_{n,m} \Big\{ (\boldsymbol{\gamma}_m^{K+1})^\top \boldsymbol{z}_n^{K+1} - t_{n,m} + \xi_{n,m} \Big\}, \end{aligned} \tag{18}$$

where $\boldsymbol{\gamma}_m^{K+1} = [\boldsymbol{\beta}_{1,m}^{K+1}, \boldsymbol{\beta}_{2,m}^{K+1}, ..., \boldsymbol{\beta}_{L^{K+1},m}^{K+1}]^\top$ is a vector of the weights linking the hidden layer to the $m$ th output node. By taking derivatives with $\boldsymbol{\gamma}_m^{K+1}$, $\boldsymbol{\xi}_n$ and $\boldsymbol{\alpha}_n$, where $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, ..., \boldsymbol{\alpha}_N]^\top$ and $\boldsymbol{\alpha}_n = [\alpha_{n,1}, \alpha_{n,2}, ..., \alpha_{n,M}]^\top$, the optimal solution of $\boldsymbol{\beta}^{K+1}$ can be obtained as

$$\boldsymbol{\beta}^{K+1} = \boldsymbol{T} (\boldsymbol{Z}^{K+1})^\top \Big( \frac{\boldsymbol{I}}{C_2} + \boldsymbol{Z}^{K+1} (\boldsymbol{Z}^{K+1})^\top \Big)^{-1}, \tag{19}$$

where $\boldsymbol{T} = [\boldsymbol{t}_1, \boldsymbol{t}_2, ..., \boldsymbol{t}_N]$. Consequently, we can obtain the weight matrix $\boldsymbol{\beta}^{K+1}$.

Given new test data $\boldsymbol{z}^{K+1}$, by using the obtained $\boldsymbol{\beta}^{K+1}$, the output value $\boldsymbol{v} = [v_1, \ v_2, \ ..., \ v_M]^\top$ is obtained as

$$\boldsymbol{v} = \boldsymbol{\beta}^{K+1} \boldsymbol{z}^{K+1}. \tag{20}$$

Furthermore, the final result *class* is obtained as

$$class = \arg \max_{m \in \{1, ..., M\}} v_m. \tag{21}$$

Thus, classification based on the proposed method can be completed.

By regarding the classes as deterioration levels, estimation of deterioration levels of transmission towers is realized. Consequently, construction of CMDELM-LRF that can consider the canonical correlation between visual and text features can realize accurate deterioration level estimation.

## III. Experimental Results

In this section, the effectiveness of the proposed method is verified. The experimental conditions are explained in III-A. Evaluation of the performance of the proposed method is explained in III-B. Furthermore, discussions of the novelty of the proposed method are shown in III-C.

### A. Experimental Conditions

In order to verify the effectiveness of the proposed method, we used a dataset that was provided by Tokyo Electric Power

**TABLE III.** Number of images used in this experiment.

| | Num. of images |
|---|---|
| Class A | 1044 |
| Class B | 1351 |
| Class C | 748 |
| Sum | 3107 |

**TABLE IV.** Details of parameters used in the proposed method.

| Details | Parameter | Value |
|---|---|---|
| Input image size | $d_w$ | 50 |
| | $d_h$ | 50 |
| Receptive field size | $r$ | 5 |
| Pooling size | $e$ | 3 |
| Num. of dimensions | $d_x$ | 471 |
| | $d_y$ | 364 |
| Regularization parameters | $C_1$ | $2^{15}$ |
| | $C_2$ | $2^{15}$ |
| Sparsity parameter | $\rho$ | 0.05 |
| Num. of input weights | $\mathbf{\Phi}$ | 35 |
| Num. of hidden layers | $K$ | 4 |
| Num. of hidden nodes | $L^1$ | 394 |
| | $L^2$ | 591 |
| | $L^3$ | 472 |
| | $L^4$ | 377 |
| | $L^5$ | 301 |
| Num. of classes | $M$ | 3 |

Company Research Institute (TEPCO). The dataset has three levels, class A, class B and class C. Class C is the most dangerous level. The details of the number of images are shown in Table III.

In the experiment, the number of layers $K$, the regularization parameters $C_1$ and $C_2$, and the dimension $d_x$, which is the number of dimensions selected by using PCA, are determined in such a way that the proposed method outputs the best estimation performance using the validation dataset. In ELM series, the parameters of the sigmoid function are set to random values. The details of the parameters are shown in Table IV. The verification method was 5-fold cross validation. We evaluated the performance of the proposed method by using Recall, Precision and F-measure, which are defined as follows:

$$\text{Recall} = \frac{\text{Num. of correctly estimated samples}}{\text{Num. of correct samples}}, \tag{22}$$

$$\text{Precision} = \frac{\text{Num. of correctly estimated samples}}{\text{Num. of all samples estimated into each level}}, \tag{23}$$

$$\text{F} - \text{measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}. \tag{24}$$

**TABLE V.** Details of methods used in the experiment.

| | | Feature | | Projected feature | |
|---|---|---|---|---|---|
| Methods | Details | Image | Text | Image | Text |
| Ours | CMDELM-LRF | - | - | ✓ | ✓ |
| Comp. 1 | CMDELM-LRF | - | - | ✓ | - |
| Comp. 2 | CMDELM-LRF | - | - | - | ✓ |
| Comp. 3 | DELM-LRF [38] | ✓ | - | - | - |
| Comp. 4 | DELM-LRF | - | ✓ | - | - |
| Comp. 5 | DELM-LRF | ✓ | ✓ | - | - |
| Comp. 6 | ELM-LRF + CCA | - | - | ✓ | ✓ |
| Comp. 7 | ELM-LRF + CCA | - | - | ✓ | - |
| Comp. 8 | ELM-LRF + CCA | - | - | - | ✓ |
| Comp. 9 | ELM-LRF [27] | ✓ | - | - | - |
| Comp. 10 | DELM [28] | ✓ | - | - | - |
| Comp. 11 | KELM [44] | ✓ | - | - | - |
| Comp. 12 | ELM [35] | ✓ | - | - | - |
| Comp. 13 | SVM [29] | ✓ | - | - | - |
| Comp. 14 | CaffeNet-CNN [45] | ✓ | - | - | - |
| Comp. 15 | VGG16-CNN [46] | ✓ | - | - | - |
| Comp. 16 | Multilayer perceptron | - | - | ✓ | ✓ |

We also used sixteen comparative methods in this experiment as shown in Table V. DELM-LRF is our previously reported method [38]. That method uses original features obtained from images or text data. DELM, KELM, ELM and SVM are constructed on the basis of visual features extracted from Caffe-Net provided by Caffe [45]. In addition, we compared our method with CNN-based methods in order to verify the effectiveness of our method. In transmission towers, since it is difficult to prepare a sufficient number of training images, we used the fine-tuned CNN methods as comparative methods. Specifically, we adopted the Caffe-Net model [45] and the VGG16 model [46]. Especially, the VGG16 model is one of the general and strong deep learning methods. Furthermore, we used a multilayer perceptron-based deep learning, which is one of the simple and benchmarking deep learning methods. Moreover, the number of hidden layers of our method and the above comparative methods is shown in Table VI. From this table, we can confirm that CMDELM-LRF needs less hidden layers than comps. 14 and 15, which are benchmarking CNN-based methods.

### *B. Performance Evaluation*

Recall, Precision and F-measure of the proposed method and the comparative methods are shown in Table VII. The performance of the proposed method is better than that of comp. 5, and the performance of comp. 7 is better than that of comp. 9. Thus, the effectiveness of feature transformation via CCA, which can maximize the canonical correlation between visual and text features, is verified. In addition, since the proposed method improves comps. 1 and 2, the use of projected multimodal features is effective for deterioration level estimation.

Since comps. 6–8 have five hidden layers, these methods are also deep learning methods. The performance of the proposed method and that of comp. 6 are better than the performance of the other methods including comps. 9 and 11–13. Thus, the effectiveness of deep learning-based methods is verified.

**TABLE VI.** Network configurations used in the experiment are shown, and "conv", "pool", "pca", "cca", "norm", "fc-elm" and "fc" represent convolution layer, pooling layer, pca procedure, cca procedure, local response normalization layer, ELM-based fully connected layer and general fully connected layer, respectively.

| | Ours and comps. 1 and 2 | Comps. 3-5 | Comps. 6-8 | Comp. 9 | Comp. 14 | Comp. 15 | Comp 16 |
|---|---|---|---|---|---|---|---|
| | conv | conv | conv | conv | conv | conv×2 | conv |
| | pool | pool | pool | pool | pool | pool | pool |
| | pca | pca | pca | pca | norm | conv×2 | pca |
| | cca | fc-elm×4 | cca | fc-elm×1 | conv | pool | fc×4 |
| | fc-elm×4 | - | fc-elm×1 | - | pool | conv×3 | - |
| | - | - | - | - | norm | pool | - |
| | - | - | - | - | conv ×3 | conv×3 | - |
| | - | - | - | - | pool | pool | - |
| | - | - | - | - | fc×3 | conv×3 | - |
| | - | - | - | - | - | pool | - |
| | - | - | - | - | - | fc×3 | - |
| Num. of hidden layers | 8 | 7 | 5 | 4 | 13 | 21 | 7 |

**TABLE VII.** Recall, Precision and F-measure of all methods.

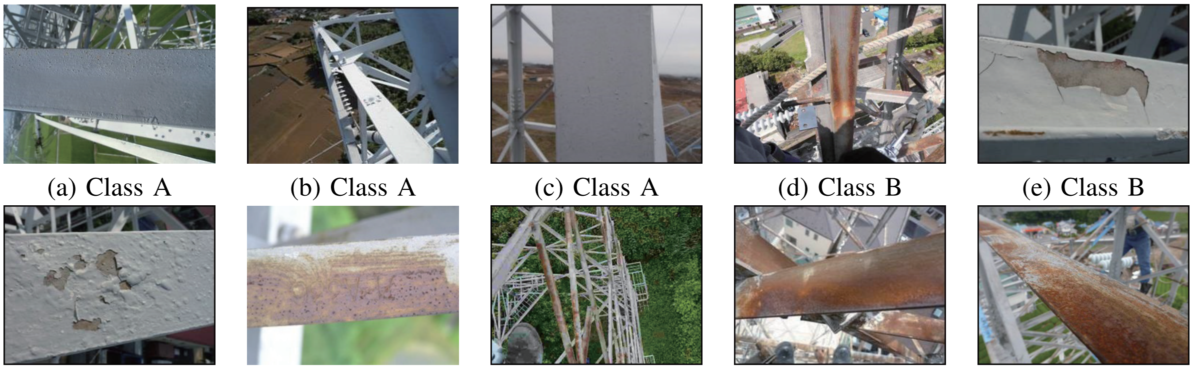
| | Class A | | | Class B | | | Class C | | | Average | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method | R | P | F | R | P | F | R | P | F | R | P | F |
| Ours | 0.985 | 0.902 | **0.939** | 0.799 | **0.956** | **0.863** | **0.922** | **0.819** | **0.863** | **0.902** | **0.892** | **0.888** |
| Comp. 1 | 0.978 | 0.881 | 0.923 | 0.751 | 0.937 | 0.826 | 0.900 | 0.780 | 0.834 | 0.877 | 0.866 | 0.861 |
| Comp. 2 | 0.496 | 0.511 | 0.498 | 0.424 | 0.581 | 0.488 | 0.595 | 0.403 | 0.479 | 0.505 | 0.499 | 0.488 |
| Comp. 3 | **0.987** | 0.890 | 0.933 | 0.593 | 0.952 | 0.721 | 0.912 | 0.621 | 0.733 | 0.831 | 0.821 | 0.796 |
| Comp. 4 | 0.765 | 0.594 | 0.668 | 0.512 | 0.792 | 0.620 | 0.703 | 0.591 | 0.637 | 0.660 | 0.659 | 0.642 |
| Comp. 5 | 0.851 | 0.647 | 0.735 | 0.609 | 0.854 | 0.708 | 0.763 | 0.734 | 0.742 | 0.741 | 0.745 | 0.728 |
| Comp. 6 | 0.970 | **0.913** | 0.937 | 0.783 | 0.931 | 0.840 | 0.896 | 0.793 | 0.830 | 0.883 | 0.879 | 0.869 |
| Comp. 7 | 0.959 | **0.913** | 0.931 | 0.810 | 0.920 | 0.847 | 0.870 | 0.805 | 0.823 | 0.880 | 0.879 | 0.867 |
| Comp. 8 | 0.561 | 0.700 | 0.619 | 0.620 | 0.714 | 0.662 | 0.770 | 0.519 | 0.620 | 0.651 | 0.644 | 0.634 |
| Comp. 9 | 0.880 | 0.856 | 0.862 | 0.688 | 0.857 | 0.745 | 0.814 | 0.651 | 0.714 | 0.794 | 0.788 | 0.774 |
| Comp. 10 | 0.498 | 0.427 | 0.460 | 0.331 | 0.522 | 0.405 | 0.511 | 0.362 | 0.424 | 0.447 | 0.437 | 0.430 |
| Comp. 11 | 0.464 | 0.401 | 0.430 | 0.347 | 0.463 | 0.397 | 0.430 | 0.354 | 0.388 | 0.414 | 0.406 | 0.405 |
| Comp. 12 | 0.485 | 0.414 | 0.446 | 0.321 | 0.476 | 0.383 | 0.448 | 0.337 | 0.384 | 0.418 | 0.409 | 0.405 |
| Comp. 13 | 0.127 | 0.546 | 0.204 | **0.909** | 0.432 | 0.585 | 0.034 | 0.288 | 0.060 | 0.357 | 0.422 | 0.283 |
| Comp. 14 | 0.517 | 0.362 | 0.426 | 0.363 | 0.464 | 0.407 | 0.163 | 0.207 | 0.182 | 0.347 | 0.344 | 0.338 |
| Comp. 15 | 0.835 | 0.707 | 0.765 | 0.611 | 0.763 | 0.679 | 0.762 | 0.695 | 0.727 | 0.736 | 0.722 | 0.724 |
| Comp. 16 | 0.944 | 0.818 | 0.872 | 0.592 | 0.896 | 0.708 | 0.920 | 0.693 | 0.788 | 0.819 | 0.802 | 0.789 |

Since the proposed method improves comp. 6, it is shown that adding hidden layers contributes to improvement of the performance. It should be noted that the contribution of feature transformation via CCA is greater than that of the hidden layers of DELM.

Furthermore, by comparing CMDELM-LRF with comps. 14-16, it is confirmed that our method is superior to other deep learning methods including very deep networks. Especially, although comp. 15 has a lot of hidden layers compared to our method as shown in Table VI, the estimation performance of our method is higher than that of the comparative method. Therefore, the effectiveness of the CCA-based feature transformation is verified.

From Table VII, it can be seen that comps. 6 and 7 have better performance on class A. Comps. 6 and 7 are methods, which combine ELM-LRF and CCA. In this experiment, although images belonging to class A have no deterioration, images belonging to classes B and C have deterioration. Thus, in order to classify the adjacent deterioration levels such as classes B and C, it is necessary to calculate features with higher representation ability which can discriminate their small difference. However, since comps. 6 and 7 are shallow neural networks, their representation ability is lower compared to deep neural networks such as CMDELM-LRF. This leads to a decline of estimation performance of classes B and C. Furthermore, in this experiment, since hidden layers' parameters are determined in such a way that the average of F-measure is high, it is considered that parameters of comps. 6 and 7 are tuned so as to distinguish class A from the other classes due to the difficulties in classification of classes B and C. Therefore, comps. 6 and 7 achieve high performance on class A.

In the proposed method, since we use random value-based $\boldsymbol{\beta}^k$, we evaluate the correctness of $\boldsymbol{\beta}^k$ indirectly by comparing the estimation performance. Specifically, we compared



(a) Class A (b) Class A (c) Class A (d) Class B (e) Class B

(f) Class B (g) Class C (h) Class C (i) Class C (j) Class C

**Fig. 2.** Examples of images [47] that were correctly estimated by the proposed method. It should be noted that the centering $50 \times 50$ $(= d_w \times d_h)$ pixel images were used for the classification in this experiment.

CMDELM-LRF with a multilayer perceptron-based deep neural network (comp. 16), which was constructed via the back propagation approach. Although it is known that the back propagation approach generally provides an optimized parameter $\boldsymbol{\beta}^k$, a large amount of training data is required in order to obtain the optimal solution. Thus, the estimation performance is limited since the number of training images used in this experiment is small. As shown in Table VII, it is confirmed that the estimation performance of CMDELM-LRF is higher than that of comp. 16. This means that ELM-AE is more effective than the back propagation approach for the calculation of $\boldsymbol{\beta}^k$ when there is only a small amount of training data.

Examples of images that were correctly estimated by the proposed method are shown in Fig. 2. From Fig. 2, it can be seen that images of transmission towers have many variations. Figures. 2 (b), (d) and (h) are distant-view images, and the others are near-view images. The angles of subjects are different as shown in Figs. 2 (a) and (c). Thus, the proposed method can estimate various kinds of images correctly. In general estimation methods such as [21], in order to cope with such variations, the targets in images are clipped manually. Although these are semi-automatic methods, our method is a fully automatic estimation method since we can input the original images into our method directly. Therefore, the effectiveness of the proposed method is verified in terms of practical application.

Furthermore, we compare computational complexity of the proposed method with that of comps. 14 and 15. The construction of hidden layers is shown in Table VI, and the computational complexity of each procedure is shown in Table VIII. Since a back propagation approach is generally used for training of comps. 14 and 15, Table VIII also includes the complexity of the back propagation. In Table VIII, $k$ is the index of a layer, and $K$ is the number of layers. $\Phi^k$ is the number of filters in the $k$ th layer. $\Phi^{k-1}$ is the number of input channels of the $(k-1)$ th layer. $d_h^k$ and $d_w^k$ are the size of the output feature map. $r^k$ and $e^k$ are the size of the convolutional filter and that of pooling filter, respectively. $d_{in}$ and $d_{out}$ are the dimensions of input features and output features, respectively. Furthermore, $L^k$ is the number of nodes in the $k$ th layer. $N$ means the number of images, and $E_p$ is the number of epochs for training of networks.

From this table, in the training step, the computational complexity of comps. 14 and 15 is $O(\sum_{k_c}^{K_c} \Phi^{k_c-1} d_h^{k_c} d_w^{k_c} \Phi^{k_c} {r^{k_c}}^2 + E_p N \sum_k^K L^k L^{k-1})$ since these methods adopt the back propagation approach, which requires a lot of computational costs for training. Note that $K_c$ is the number of convolution layers, and $k_c$ is the index of a convolution layer. We can ignore the computational complexity of the other procedures since the other procedures have much lower complexity than "convolution" and "back propagation" as shown in Table VIII. In order to train effective deep networks, comps. 14 and 15 need a lot of epochs $E_p$. Furthermore, since the number of hidden layers $K$ including $K_c$ of these methods are comparably large as shown in Table VI, they have high computational complexity. On the other hand, the complexity of the proposed method is $O(\sum_{k_c}^{K_c} \Phi^{k_c-1} d_h^{k_c} d_w^{k_c} \Phi^{k_c} {r^{k_c}}^2)$, and $K_c = 1$ due to the construction of only one convolution layer. In the proposed method, since the size $d_h^{k_c} \times d_w^{k_c}$ of input images and filter size $r^{k_c}$ are low. From the above, it is confirmed that the proposed method has extremely lower complexity than comps. 14 and 15 in the training step. Actually, the measured computational time of the proposed method is much lower than that of recent deep learning techniques such as comps. 14 and 15 as shown in Table IX. Specifically, the cost of the proposed method, CaffeNet and VG16 are $5.08 \times 10^1$ sec, $1.44 \times 10^4$ sec and $2.16 \times 10^3$ sec. Details of the computation costs of training procedures are shown in Table IX. The proposed method was trained by using a personal computer (CPU) with Intel (R) Core (TM) CPU i7-3770 @3.40 GHz with 16 Gbytes RAM. CaffeNet-CNN was trained by using a personal computer (GPU) with Intel (R) Xeon (R) CPU E5-2699 v3@2.30 GHz with 512 Gbytes RAM and GPU Tesla K80. From this table, the cost of the training of DELM is much lower even if it is trained by using CPU. This is because ELM-AE, which is a layer-by-layer unsupervised learning method, is used for the construction of DELM as mentioned in II-C.

Moreover, in the test step, since the complexity of all of the proposed method and comps. 14 and 15 depend on the number of convolution layers, a difference of computational

**TABLE VIII.** Computational complexity of each procedure.

| Procedure | Computational complexity | CMDELM-LRF | Comp. 14 | Comp. 15 |
|---|---|---|---|---|
| Convolution | $O(\Phi^{k-1} d_h^k d_w^k \Phi^k r^{k^2})$ | ✓ | ✓ | ✓ |
| Pooling | $O(\Phi^{k-1} d_h^k d_w^k e^{k^2})$ | ✓ | ✓ | ✓ |
| PCA | $O(d_{in}^2 N)$ | ✓ | - | - |
| CCA | $O(d_{in}^2 N)$ | ✓ | - | - |
| Norm | $O(\Phi^{k-1} d_h^k d_w^k)$ | - | ✓ | - |
| Fc-elm | $O(L^{k^2} L^{k-1})$ | ✓ | - | - |
| Fc | $O(d_{in} d_{out})$ | - | ✓ | ✓ |
| Back propagation | $O(E_p N \sum_k^K L^{k-1} L^k)$ | - | ✓ | ✓ |

**TABLE IX.** Computation cost (sec) of the training procedures of ours, Caffe-Net and VGG16.

| Procedure | Ours | Caffe-Net | VGG16 |
|---|---|---|---|
| Feature extraction | $5.01 \times 10^1$ | - | - |
| Maximization of canonical correlation | $2.62 \times 10^{-1}$ | - | - |
| Construction of classifier | $4.49 \times 10^{-1}$ | $1.44 \times 10^4$ | $2.16 \times 10^3$ |
| Sum | $5.08 \times 10^1$ | $1.44 \times 10^4$ | $2.16 \times 10^3$ |

**TABLE X.** Computation cost of test procedures of ours, Caffe-Net and VGG16.

| Method | Computation cost (sec) |
|---|---|
| Ours | $\mathbf{7.35 \times 10^{-1}}$ |
| Caffe-Net | 1.31 |
| VGG16 | 2.06 |

order of these methods may be slight. However, the proposed method consists of one convolution layer, but comps. 14 and 15 consist of a lot of convolution layers. Therefore, since the measured computational time of the proposed method is lower as shown in Table X, we realize high speed computation. From the above discussion, CMDELM-LRF is effective for actual deterioration level estimation.

### *C. Discussion*

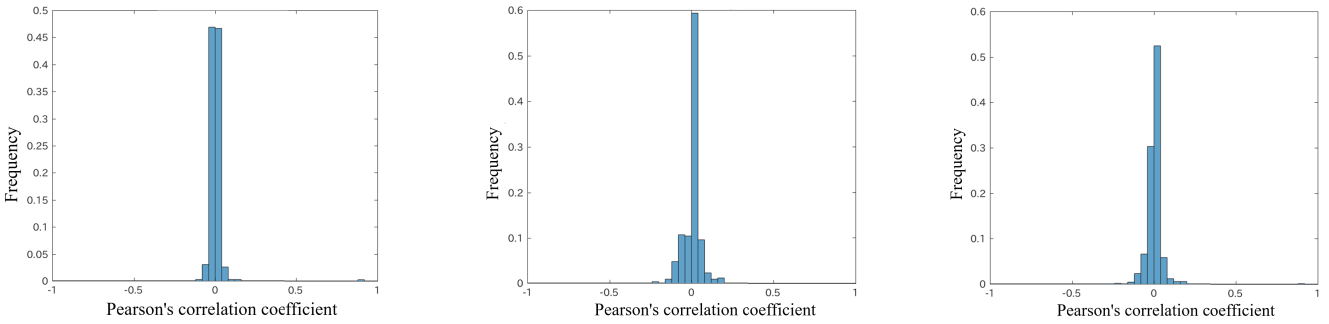
In this subsection, we discuss the effectiveness of the use of CCA, which is the main contribution of the proposed method. Specifically, we discuss the reason why the projected features obtained by CCA contribute to the improvement of the estimation performance. In general classification tasks, it is known that the stronger correlation between labels and features is, the more discriminative features are [48]. Thus, in order to evaluate the relationships between the estimation performance and the use of the projected features or the original features, we calculated the Pearson's correlation coefficients between features and labels in the training data. Specifically, we calculated the correlation coefficients between class labels and each dimension of features and constructed their histograms as shown in Figs. 3 and 4. Note that a bin value of each histogram is normalized by the total number of dimensions of a target feature. For example, in case of Fig. 3 (a), since the dimension $d_x$ of the original visual features is 471, frequency of 471 coefficients is displayed. Figure 3 shows the histograms of the correlation coefficients between the original features and labels. Figures 3 (a), (b) and (c) correspond to the results of comps. 3, 4 and 5, respectively. Similarly, Fig. 4 shows the histograms of the correlation coefficients between the CCA-based projected features and labels. Figures 4 (a), (b) and (c) correspond to the results of comps. 1 and 2, and our method, respectively.

These figures mean that the larger the number of values close to ±1 in the horizontal axis, the higher the correlation with labels is. Furthermore, in order to quantitatively compare the results in these figures, we calculated the variance of the correlation coefficients (Var) and the average of the sum of the absolute coefficient values (AveA). The larger these values are, the higher the correlation between features and labels is. Their discussions are shown below.

- Comparison between Figs. 3 (a) and 4 (a)
  As shown in these figures and the values of "Var" and "AveA", we can confirm that the correlation between the projected visual features and labels is higher. As is clear from the experimental results of comps. 1 and 3, the projected visual features are more effective than the original visual features.
- Comparison between Figs. 3 (b) and 4 (b)
  On the other hand, as shown in Figs. 3 (b) and 4 (b), it is confirmed that the correlation between the projected text features and labels is lower. Furthermore, as similar to the above relationship, the performance of comp. 2 using the projected text features is also lower.

From the above two points, it is verified that there actually exists the relationship between the correlation and the estimation performance.

- Comparison between Figs. 3 (c) and 4 (c)
  Figure 3 (c) has a histogram obtained by naturally integrating Figs. 3 (a) and 3 (b). Similarly, Fig. 4 (c) has a histogram obtained by integrating Figs. 4 (a) and 4 (b).

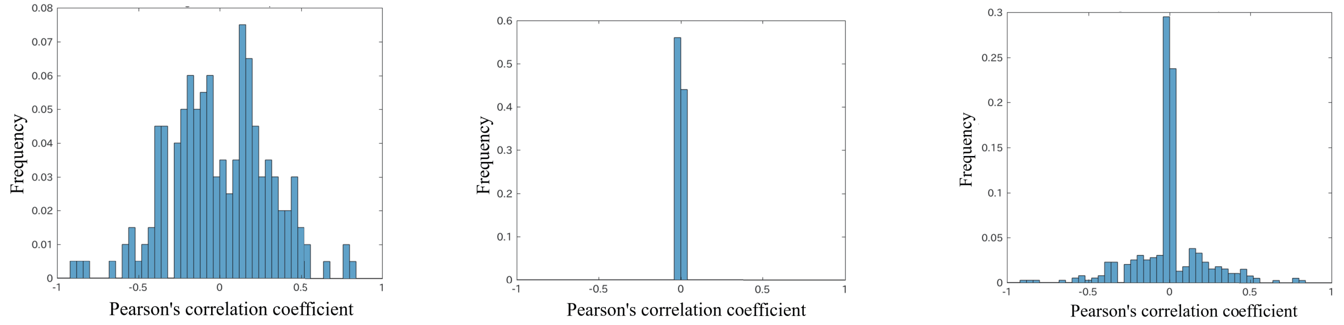


(a) Correlation between visual features $\boldsymbol{x}_n$ and class labels. Var : $2.25 \times 10^{-3}$, AveA : $1.73 \times 10^{-2}$

(b) Correlation between text features $\boldsymbol{y}_n$ and class labels. Var : $2.22 \times 10^{-3}$, AveA : $2.67 \times 10^{-2}$

(c) Correlation between visual and text features $[\boldsymbol{x}_n{}^\top, \boldsymbol{y}_n{}^\top]^\top$ and class labels. Var : $2.24 \times 10^{-3}$, AveA : $2.16 \times 10^{-2}$

**Fig. 3.** Pearson's correlation coefficients between the original features and class labels. In addition, "Var" means the variance of the coefficients, and "AveA" means the average of the sum of the absolute coefficient values. Figures 3 (a), (b) and (c) correspond to the results of comps. 3, 4 and 5, respectively.



(a) Correlation between projected visual features $\hat{\boldsymbol{x}}_n$ and class labels. Var : $9.40 \times 10^{-2}$, AveA : $2.49 \times 10^{-1}$

(b) Correlation between projected text features $\hat{\boldsymbol{y}}_n$ and class labels. Var : $5.27 \times 10^{-5}$, AveA : $7.27 \times 10^{-3}$

(c) Correlation between projected visual and text features $[\hat{\boldsymbol{x}}_n^\top, \hat{\boldsymbol{y}}_n^\top]^\top$ and class labels. Var : $4.70 \times 10^{-2}$, AveA : $1.28 \times 10^{-1}$

**Fig. 4.** Pearson's correlation coefficients between the CCA-based projected features and class labels. In addition, "Var" means the variance of the coefficients, and "AveA" means the average of the sum of the absolute coefficient values. Figures 4 (a), (b) and (c) correspond to the results of comps. 1 and 2, and our method, respectively.

It is confirmed that the aligning features of both the projected visual and text features have higher correlation with labels as shown in Figs. 3 (c) and 4 (c). Furthermore, the values of "Var" and "AveA" are also higher than those of the original features. In addition, our method achieves higher performance than comp. 5. CCA is a method calculating the projection in such a way that the correlation between two kinds of features is maximized, and it does not include a process to make the correlation between features and labels high. Nevertheless, since the projected features $[\hat{\boldsymbol{x}}_n{}^\top, \hat{\boldsymbol{y}}_n{}^\top]^\top$ obtained via CCA are strongly correlated with labels, the projection obtained via CCA can realize not only the maximization of the correlation between visual and text features but also the calculation of the discriminative features. Consequently, the effectiveness of the novelty of the proposed method is verified.

From the above discussions, we can newly confirm that "Var" and "AveA" can become evaluation indices for selection of the CCA-based projected features and the original features. In other words, by using these indices, we may select effective features which provide further improvement of the estimation performance. This will be addressed in our future work.

## IV. Conclusions

We have proposed deterioration level estimation via deep learning that maximizes the canonical correlation between heterogeneous features. Our CMDELM-LRF can transform visual and text information to more semantic information through the hidden layers. In CMDELM-LRF, by inserting one hidden layer, which can maximize the canonical correlation between visual and text features, feature transformation is realized without designing many hidden layers. Therefore, CMDELM-LRF could be trained by using a small amount of training data. This is the main contribution of this paper. The effectiveness of the proposed method was verified from experimental results.

We can provide the big contribution to the signal processing field in the following points. As shown in the introduction, since LRF calculates visual features based on random values, and there are a few parameters to be optimally determined, the combination use of ELM and LRF is effective for a small amount of training data. In fact, it has been reported that performance improvement was realized by applying the ELM-LRF-based method [49]. On the other hand, it has also been reported that heterogeneous features provided higher estimation performance than only single visual features [50], [51]. Therefore, even if it is difficult to prepare a large amount of training data, it is convinced that the performance will further increase by integration of both ELM-LRF-based methods and the use of heterogeneous features. However, in the recent studies of ELM-LRF series, the methods, which can be applied to single modality such as visual information, have only been proposed. In other words, ELM-LRF-based methods have not been extended for multimodal data. Therefore, we have proposed CMDELM-LRF, which can effectively use multimodal features, while retaining the advantages of ELM and LRF. Consequently, although our method is constructed by using the existing CCA, since it can integrate both ELM-LRF-based methods and the use of heterogeneous features, we can contribute considerably to the field of signal processing.

We used all text features in the construction of CMDELM-LRF. However, since text data have various kinds of inspection items, selection of text data to be used is required for extracting more effective text features. Here, Wang et al. verified that performing $L_{2,1}$-norm on the projection matrices, which can transform multimodal features to a common feature space, is effective for the feature selection [52]. The feature selection can provide relevant and discriminative features from coupled feature spaces simultaneously. Thus, we will calculate the projection including the feature selection approach by introducing the $L_{2,1}$-norm to the CCA's objective function as our future work. This will lead to further improvement of the estimation performance.

## References

[1] E. Schnebele, B. Tanyu, G. Cervone, and N. Waters, "Review of remote sensing methodologies for pavement management and assessment," *European Transport Research Review*, vol. 7, no. 2, p. 7, 2015.

[2] B. Bergquist and P. Söderholm, "Data analysis for condition-based railway infrastructure maintenance," *Quality and Reliability Engineering International*, vol. 31, no. 5, pp. 773–781, 2015.

[3] Y. C. Lai, D. C. Fan, and K. L. Huang, "Optimizing rolling stock assignment and maintenance plan for passenger railway operations," *Computers & Industrial Engineering*, vol. 85, pp. 284–295, 2015.

[4] R. Montero, J. Victores, S. Martínez, A. Jardón, and C. Balaguer, "Past, present and future of robotic tunnel inspection," *Automation in Construction*, vol. 59, pp. 99–112, 2015.

[5] P. Xu, Q. Sun, R. Liu, R. R. Souleyrette, and F. Wang, "Optimizing the alignment of inspection data from track geometry cars," *Computer-Aided Civil and Infrastructure Engineering*, vol. 30, no. 1, pp. 19–35, 2015.

[6] N. Pouliot, P. L. Richard, and S. Montambault, "Linescout technology opens the way to robotic inspection and maintenance of high-voltage power lines," *IEEE Trans. Power and Energy Technology Systems Journal*, vol. 2, no. 1, pp. 1–11, 2015.

[7] M. Minagawa, S. Satoh, and T. Kamitani, "Prototype diagnosis expert system with knowledge refinement function," *Journal of Computing in Civil Engineering*, vol. 15, no. 2, pp. 112–117, 2001.

[8] K. Maeda, S. Takahashi, T. Ogawa, and M. Haseyama, "Distress classification of road structures via adaptive Bayesian network model selection," *Journal of Computing in Civil Engineering*, vol. 31, no. 5, p. 04017044, 2017.

[9] A. G. YO, N. A. Okine, G. Garateguy, R. Carrillo, and G. R. Arce, "Multiresolution information mining for pavement crack image analysis," *Journal of Computing in Civil Engineering*, vol. 26, no. 6, pp. 741–749, 2011.

[10] Q. Zou, Y. Cao, Q. Li, Q. Mao, and S. Wang, "Cracktree: Automatic crack detection from pavement images," *Pattern Recognition Letters*, vol. 33, no. 3, pp. 227–238, 2012.

[11] E. Zalama, G. B. J. Gómez, R. Medina, and J. Llamas, "Road crack detection using visual features extracted by gabor filters," *Computer-Aided Civil and Infrastructure Engineering*, vol. 29, no. 5, pp. 342–358, 2014.

[12] H. N. Nguyen, T. Y. Kam, and P. Y. Cheng, "An automatic approach for accurate edge detection of concrete crack utilizing 2d geometric features of crack," *Journal of Signal Processing Systems*, vol. 77, no. 3, pp. 221–240, 2014.

[13] T. C. Hutchinson and Z. Chen, "Improved image analysis for evaluating concrete damage," *Journal of Computing in Civil Engineering*, vol. 20, no. 3, pp. 210–216, 2006.

[14] L. Sun, M. Kamaliardakani, and Y. Zhang, "Weighted neighborhood pixels segmentation method for automated detection of cracks on pavement surface images," *Journal of Computing in Civil Engineering*, vol. 30, no. 2, p. 04015021, 2015.

[15] K. Aydin and O. Kisi, "Applicability of a fuzzy genetic system for crack diagnosis in timoshenko beams," *Journal of Computing in Civil Engineering*, vol. 29, no. 5, p. 04014073, 2014.

[16] M. R. Jahanshahi and S. F. Masri, "Parametric performance evaluation of wavelet-based corrosion detection algorithms for condition assessment of civil infrastructure systems," *Journal of Computing in Civil Engineering*, vol. 27, no. 4, pp. 345–357, 2012.

[17] R. Adhikari, O. Moselhi, A. Bagchi, and A. Rahmatian, "Tracking of defects in reinforced concrete bridges using digital images," *Journal of Computing in Civil Engineering*, p. 04016004, 2016.

[18] M. O ' Byrne, F. Schoefs, B. Ghosh, and V. Pakrashi, "Texture analysis based damage detection of ageing infrastructural elements," *Computer-Aided Civil and Infrastructure Engineering*, vol. 28, no. 3, pp. 162–177, 2013.

[19] S. Varadharajan, S. Jose, K. Sharma, L. Wander, and C. Mertz, "Vision for road inspection," in *Proc. IEEE Winter Conference on Applications of Computer Vision*, 2014, pp. 115–122.

[20] P. Jonsson, J. Casselgren, and B. Thornberg, "Road surface status classification using spectral analysis of nir camera images," *IEEE Trans. Sensors Journal*, vol. 15, no. 3, pp. 1641–1656, 2015.

[21] F. Tsutsumi, H. Murata, T. Onoda, O. Oguri, and H. Tanaka, "Automatic corrosion estimation using galvanized steel images on power transmission towers," in *Proc. IEEE Transmission & Distribution Conference & Exposition: Asia and Pacific*, 2009, pp. 1–4.

[22] B. Yan, S. Goto, A. Miyamoto, and H. Zhao, "Imaging-based rating for corrosion states of weathering steel using wavelet transform and psosvm techniques," *Journal of Computing in Civil Engineering*, vol. 28, no. 3, p. 04014008, 2013.

[23] F. Zhang, W. Wang, Y. Zhao, P. Li, Q. Lin, and L. Jiang, "Automatic diagnosis system of transmission line abnormalities and defects based on uav," in *Proc. IEEE International Conference on Applied Robotics for the Power Industry*, 2016, pp. 1–5.

[24] N. Fuse, A. Naganuma, T. Fukuchi, Y. Hori, M. Mizuno, and K. Fukunaga, "Underfilm corrosion of transmission tower cross-arms service-used in a pacific coast area," *Corrosion*, vol. 71, no. 11, pp. 1387–1397, 2015.

[25] S. Woo, I. Chu, B. Youn, and K. Kim, "Development of the corrosion deterioration inspection tool for transmission tower members," *KEPCO Journal on Electric Power and Energy*, vol. 2, no. 2, pp. 293–298, 2016.

[26] TEPCO Press Release PC Watch [Smart O & M], http://pc.watch.impress.co.jp/docs/news/1024585.html.

[27] G. B. Huang, Z. Bai, L. L. C. Kasun, and C. M. Vong, "Local receptive fields based extreme learning machine," *IEEE Computational Intelligence Magazine*, vol. 10, no. 2, pp. 18–29, 2015.

[28] E. Cambria, G. B. Huang, L. L. C. Kasun, H. Zhou, C. M. Vong, J. Lin, J. Yin, Z. Cai, Q. Liu, and K. L. et al., "Extreme learning machines [trends & controversies]," *IEEE Intelligent Systems*, vol. 28, no. 6, pp. 30–59, 2013.

[29] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[30] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, “Road crack detection using deep convolutional neural network,” in *Proc. IEEE International Conference on Image Processing*, 2016, pp. 3708–3712.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Proc. Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[32] Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen, “Medical image classification with convolutional neural network,” in *Proc. IEEE International Conference on Control Automation Robotics & Vision*, 2014, pp. 844–848.

[33] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, “Decaf: A deep convolutional activation feature for generic visual recognition.” in *Proc. IEEE International Conference on Machine Learning*, vol. 32, 2014, pp. 647–655.

[34] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, “Learning deep features for scene recognition using places database,” in *Proc. Advances in Neural Information Processing Systems*, 2014, pp. 487–495.

[35] G. B. Huang, Q. Y. Zhu, and C. K. Siew, “Extreme learning machine: a new learning scheme of feedforward neural networks,” in *Proc. IEEE International Joint Conference on Neural Networks*, vol. 2, 2004, pp. 985–990.

[36] H. Hotelling, “Relations between two sets of variates,” *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.

[37] Y. R. Yeh, C. H. Huang, and Y. C. F. Wang, “Heterogeneous domain adaptation and classification by exploiting the correlation subspace,” *IEEE Trans. Image Processing*, vol. 23, no. 5, pp. 2009–2018, 2014.

[38] K. Maeda, S. Takahashi, T. Ogawa, and M. Haseyama, “Automatic estimation of deterioration level on transmission towers via deep extreme learning machine based on local receptive field,” in *Proc. IEEE International Conference on Image Processing*, 2017, pp. 2379–2383.

[39] S. Andrew, K. P. W, C. Zhenghao, B. Maneesh, S. Bipin, and N. A. Y, “On random weights and unsupervised feature learning,” in *Proc. International Conference on Machine Learning*, 2011, pp. 1089–1096.

[40] A. Coates, A. Ng, and H. Lee, “An analysis of single-layer networks in unsupervised feature learning,” in *Proc. International Conference on Artificial Intelligence and Statistics*, 2011, pp. 215–223.

[41] S. Wold, K. Esbensen, and P. Geladi, “Principal component analysis,” *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[42] J. Xie, L. Xu, and E. Chen, “Image denoising and inpainting with deep neural networks,” in *Proc. Advances in Neural Information Processing Systems*, 2012, pp. 341–349.

[43] G. B. Huang, H. Zhou, X. Ding, and R. Zhang, “Extreme learning machine for regression and multiclass classification,” *IEEE Trans. Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 2, pp. 513–529, 2012.

[44] G. B. Huang, Q. Y. Zhu, and C. K. Siew, “Extreme learning machine: theory and applications,” *Neurocomputing*, vol. 70, no. 1, pp. 489–501, 2006.

[45] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” in *Proc. ACM International Conference on Multimedia*. ACM, 2014, pp. 675–678.

[46] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[47] H. Hamada *et al.*, Tokyo Electronic Power Company Holdings, Inc. (*private communication*).

[48] S.-B. Chen, Y. Zhang, C. H. Ding, Z.-L. Zhou, and B. Luo, “A discriminative multi-class feature selection method via weighted l2, 1-norm and extended elastic net,” *Neurocomputing*, vol. 275, pp. 1140–1149, 2018.

[49] Y. Shen, J. Chen, and L. Xiao, “Supervised classification of hyperspectral images using local-receptive-fields-based kernel extreme learning machine,” in *Proc. IEEE International Conference on Image Processing*, 2017, pp. 3120–3124.

[50] X. Zhang, X. Zhang, X. Li, Z. Li, and S. Wang, “Classify social image by integrating multi-modal content,” *Multimedia Tools and Applications*, vol. 77, no. 6, pp. 7469–7485, 2018.

[51] K. Tang, M. Paluri, L. Fei-Fei, R. Fergus, and L. Bourdev, “Improving image classification with location context,” in *Proc. IEEE International Conference on Computer Vision*, 2015, pp. 1008–1016.

[52] K. Wang, R. He, W. Wang, L. Wang, and T. Tan, “Learning coupled feature spaces for cross-modal matching,” in *Proc. IEEE International Conference on Computer Vision*, 2013, pp. 2088–2095.

**Keisuke Maeda** received his B.S., M.S. degrees in Electronics and Information Engineering from Hokkaido University, Japan in 2015 and 2017, respectively. He is currently pursuing an Ph.D. degree at the Graduate School of Information Science and Technology, Hokkaido University. He is currently a Research Fellow of the Japan Society for the Promotion of Science. His research interests are multimodal processing and its applications. He is a student member of the IEEE and IEICE.

**Sho Takahashi** received his B.S., M.S. and Ph.D. degrees in Electronics and Information Engineering from Hokkaido University, Japan in 2008, 2010 and 2013, respectively. He joined the Graduate School of Information Science and Technology, Hokkaido University, as an Assistant Professor in 2013. He was an Associate Professor of the Education and Research Center for Mathematical and Data Science, Hokkaido University from 2017 to 2018. He is currently an Associate Professor in the Faculty of Engineering, Hokkaido University. His research interests include semantic analysis and visualization in various data. He is a member of the IEEE, IEICE, Institute of Image Information and Television Engineers (ITE), Japan Society of Civil Engineering (JSCE) and Society of Automotive Engineering of Japan (JSAE).

**Takahiro Ogawa** received his B.S., M.S. and Ph.D. degrees in Electronics and Information Engineering from Hokkaido University, Japan in 2003, 2005 and 2007, respectively. He is currently an assistant professor in the Graduate School of Information Science and Technology, Hokkaido University. His research interests are multimedia signal processing and its applications. He has been an Associate Editor of ITE Transactions on Media Technology and Applications. He is a member of the IEEE, EURASIP, IEICE, and Institute of Image Information and Television Engineers (ITE).

**Miki Haseyama** received her B.S., M.S. and Ph.D. degrees in Electronics from Hokkaido University, Japan in 1986, 1988 and 1993, respectively. She joined the Graduate School of Information Science and Technology, Hokkaido University as an associate professor in 1994. She was a visiting associate professor of Washington University, USA from 1995 to 1996. She is currently a professor in the Graduate School of Information Science and Technology, Hokkaido University. Her research interests include image and video processing and its development into semantic analysis. She has been a Vice-President of the Institute of Image Information and Television Engineers, Japan (ITE), an Editor-in-Chief of ITE Transactions on Media Technology and Applications, a Director, International Coordination and Publicity of The Institute of Electronics, Information and Communication Engineers (IEICE). She is a member of the IEEE, IEICE, Institute of Image Information and Television Engineers (ITE) and Acoustical Society of Japan (ASJ).