PROCEEDINGS OF SCIENCE

An investigation about pretrainings for the multi-modal sensor data

Tomohiro Mashita,[a] **Kenshiro Tamata,**[b,*] **Ryota Ioka,**[c] **Ryo Itoh,**[c] **Hiroki Matsuzaki**[c] **and Toshihide Miyake**[c]

[a]*Faculty of Information and Communication Engineering, Osaka Electro-Communication University,*
18-8 Hatsucho, Neyagawa-shi, Osaka 572-8530, Japan

[b]*Graduate School of Information Science and Technology, Osaka University,*
Yamadaoka 1-5, Suita city, Osaka 565-0871, Japan

[c]*Hitachi Zosen Coorporation,*
2-11, Funamachi 2-chome, Taisho-ku, Osaka 551-0022, Japan

E-mail: mashita@osakac.ac.jp

This paper investigates the effect of pretraining and fine-tuning for a multi-modal dataset. The detaset used in this study is accumulated in a garbage disposal facility for the facility control and consists of 25000 sequential images and corresponding sensor values. The main task for this dataset is to classify the state of garbage incineration from an input image for the combustion state control. In this kind of task, pretraining with an unsupervised dataset and fine-tuning with a small supervised dataset is a typical and effective approach to reducing the costs of making supervised data. To find effective pretraining, we investigated and compared some pretraining with the sensor values and an autoencoder. Moreover, we compared some sensor selection methods for pretraining with sensors. The results show the performance and discussion about fine-tuned models with frozen and unfrozen pretraining parameters and the sensor selection.

International Symposium on Grids & Clouds (ISGC) 2024
24-29 March 2024
Academia Sinica, Taipei, Taiwan

[*] Speaker

1. Introduction

A deep neural network model generally requires a large amount of supervised data for training, and making a training dataset is costly. The typical approaches to save the cost for dataset making are pretraining or domain adaptation with unsupervised datasets obtained from the same domain or another dataset for a similar task.

One of the typical approaches for pretraining is self-supervised learning. Autoencoder is one of the general architectures for pretraining, and it is trained to encode an input to a feature vector and then trained to regenerate input data. In the case of a model for image processing, there is another approach based on contrastive learning without supervised learning [1].

In the case of a multi-modal dataset consisting of images and corresponding sensor values, the relationship between sensor values and images has a possibility for good pretraining. In this study, we focused on the pretraining with a multi-modal dataset that contains images and corresponding sensor values. We evaluated and investigated pretraining methods for an image classification task.

2. Related works

The data accumulation and annotation cost is a significant issue in the actual machine learning tasks. Especially after the invention of deep learning, those issues became more significant. An autoencoder architecture [2] has been used to pretrain deep neural networks from the early days of deep learning, and it is one of the typical approaches for pretraining deep neural networks. In the case of a model based on a convolutional neural network (CNN) for image inputs, a conv-deconv architecture is famous for encoding input images to feature values. The conv-deconv architecture is also employed in the semantic segmentation task, which outputs an image of a dense pixel-wise segmentation map. There are some widely used models based on conv-deconv architecture [3–5]

Transformer [6] and its derivative models [7], including Vision Transormer (ViT) [8], achieved good performance in various tasks. Naturally, Transformer models are also used as encoder-decoder models. In the case of image encoding, some models based on Transformer are proposed [7, 9]. To decide the model used in the experiment, we compare the performances of CNN based architecture and ViT in the classification tasks with the supervised dataset.

3. Pretraining setting

3.1 Dataset

The dataset we used in this study consists of sequential images and sensor values obtained in a garbage disposal facility. The image sequence is 25000 frames, and 84 sensor values correspond to one frame. The main task is to classify the state of garbage incineration from an image for facility control. In actuality, this task is a 3-label image classification. For the evaluation and fine-tuning, we prepared a supervised dataset consisting of three labels C_0, C_1, and C_2, and the number of labeled images is 978, 1528, and 2593, respectively. Fig. 1 shows examples of input images.

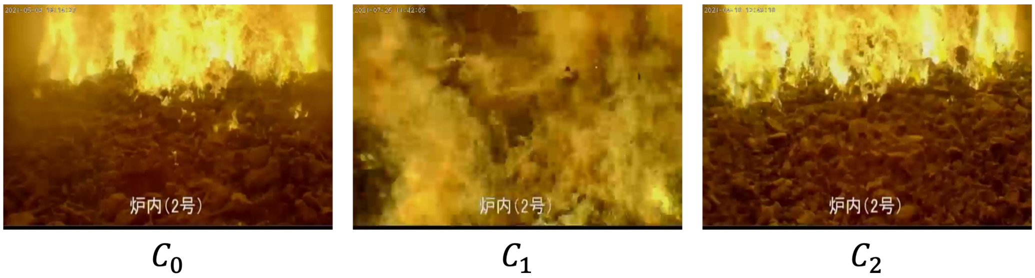


Figure 1: Sample images.

3.2 Backbone model

To choose the backbone model used in this study, we evaluated ResNet50 [10] and ViT. Those models are trained with our supervised dataset without pretraining. The evaluation result in Table 2 indicates that the ViT performed better than ResNet. Therefore, we employ ViT as the backbone model for this study.

Figs. 2 and 3 show the architecture consisting of Encoder and Prediction blocks. As shown in 2, an Encoder block consists of multi-layer ViT blocks and encodes feature values in the pretraining phase. Then, as shown in Fig. 3, the multi-layer-perceptron (MLP) layer predicts sensor values. The red blocks in the figures have trainable parameters and are trained in the pretraining and fine-tuning phase. The brown blocks are removed after the pretraining. The green blocks do not have trainable parameters. H and W mean the input image's height and width. d is the number of dimensions in the hidden layer of the ViT blocks. LN blocks refer to layer normalization. In this study, the actual parameters of the pretraining model are $H = 256$, $W = 256$, $d = 64$, the number of heads in the attention layer is 4, patch-size is 32×32, and ViT blocks are 4. The settings for the pretraining are: 64 batch size, AdamW optimizer, 0.001 learning rate, 0.0001 weight decay, and 2000 epoch training. The number of images and corresponding sensor values for the training and validation are 10000 and 2899, respectively.

Table 1: Dataset for this study.

	train	validation	test
C_0	128	425	425
C_1	128	700	700
C_2	128	1232	1233

Table 2: Backbone model.

backbone model	accuracy
ResNet	0.7273
ViT	0.8032

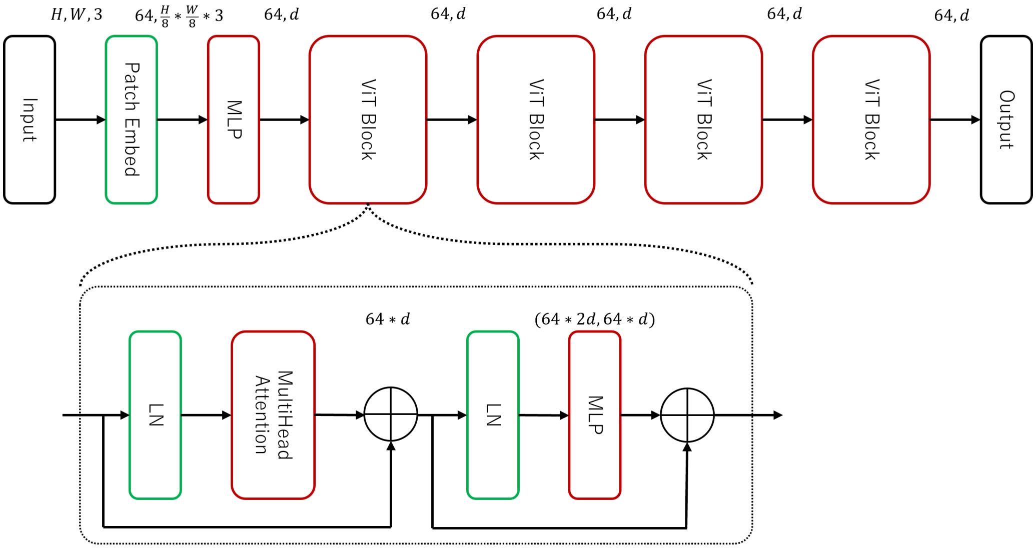


Figure 2: Pretraining encoder block.

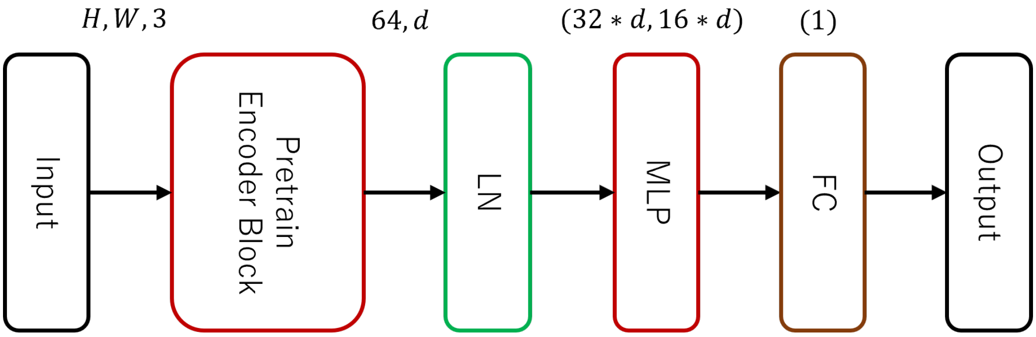


Figure 3: Pretraining model.

3.3 Sensor selection

The dataset used in this study has 84 sensor values for each frame. Since the garbage disposal facility, the dataset's source, has some control systems, the sensor values include the parameters for the control. Some of them are used for the control system and have dependency and time delay. Thus, some are unsuitable for the pretraining, and we must select suitable sensor values.

We employed interpretable regression models, Lasso [11] and Random Forest (RF) [12], to select sensor values. We applied those regression models to classify the state of garbage incineration by all sensor values without images. We used a supervised dataset for this sensor selection. Tables 3 and 4 show the top 5 contributive sensors and their sum of absolute coefficients in Lasso and RF, respectively. According to those results, we employed the sensor values shown in Table 3 and 4. Moreover, we employed the sensor values, S_7, S_8, S_9, and S_{10}, which the specialist in the garbage incineration facility selects.

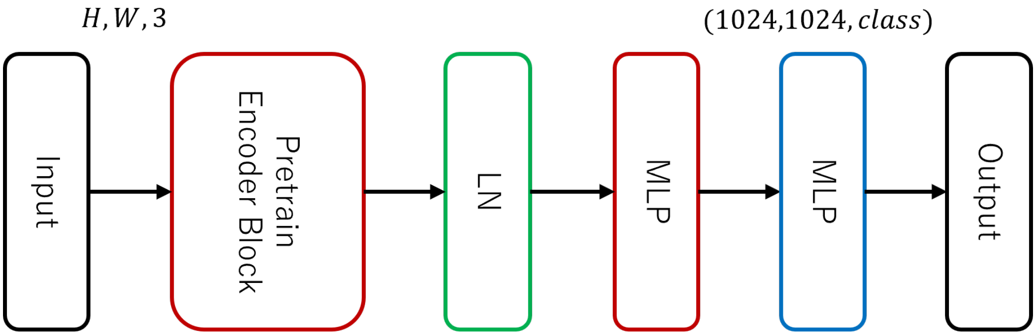


Figure 4: Fine-tuning model.

4. Experiment

4.1 Fine-tuning setting

Fig. 4 shows the architecture for the fine-tuning. The red blocks have pretrained parameters, and the blue blocks have randomly initialized parameters. The training parameters in fine-tuning are the same as in the pretraining phase. There are three classes for the classification task in the fine-tuning.

4.2 Performance comparison by fine tuning

To investigate the effect of the pretraining, we fine-tune the pretrained models and compare the performances. The models in this comparison are pretrained to infer the corresponding sensor value and then fine-tuned with the supervised dataset with three classes and 128 images for each class. We also evaluated two models pretrained with AE and all sensors. As a baseline, we trained a model without pretraining.

Table 3: Coefficients of Lasso.

sensor	Class 0	Class 1	Class 2	ABS SUM
sensor 40	0	-1.800	2.035	3.835
sensor 38	-0.4855	1.674	-0.3846	2.544
sensor 70	0	-1.169	1.170	2.339
sensor 46	-0.2825	0.7681	-0.7158	1.766
sensor 41	-0.7593	-0.3225	0.1587	1.241

Table 4: Coefficients of Random Forest.

sensor	coefficient
sensor 40	0.1007
sensor 38	0.06911
sensor 41	0.04068
sensor 39	0.03350
sensor 5	0.02885

In the fine-tuning, the numbers of images for training, validation, and testing are shown in Table 1. The training parameters are batch-size 64, learning rate 0.001, weight decay 0.0001, and 2000 epochs. In this comparison, the model parameters including pretrained parameters are trainable. Table 5 shows the inference performance of the models. In this comparison, the model pretrained with the sensor 39 achieved best performance but it is close to the other models' performances. Moreover, the difference between the models with pretrainings and without pretraining are small, we can consider that the number of supervised dataset is enough to train the models.

4.3 Analysis of pretraining

To confirm the effect of pretraining clearly, we conducted fine-tuning with frozen models that prohibit updating pretrained parameters. In this evaluation, the parameters in the red blocks in Fig.4 are frozen, and the blue blocks are trained in fine-tuning. The number of images for the fine-tuning and training parameters is the same as the training shown in Sec.4.2.

Table 6 shows the inference performances of the models. This result clearly shows the performance difference depending on the sensors used in the pretrainings. AE achieved the best performance, and this result means that whereas the pretrainings predicting a sensor value encode the information regarding the sensor value, AE encodes all information included in the images used for the pretraining. Therefore, the feature value encoded by AE has rich but verbose information. The feature values encoded by the pretrainings with the sensor values have limited information, and their performances depend on the relevance between the domains of the pretraining and the main task. Fig. 5 shows the difference in the fine-tuning after the pretraining with AE and sensor 39. The model pretrained with sensor 39 is trained quickly, but the model pretrained with AE is trained slowly. This result supports the above-mentioned difference between the pretraings with AE and sensors.

Table 5: Comparison.

Pretraining	Accuracy	Precision	Recall	F1
sensor 5	0.8109	0.7884	0.7970	0.7923
sensor 7	0.7884	0.7668	0.7691	0.7678
sensor 8	0.8020	0.7791	0.8043	0.7889
sensor 9	0.8176	0.7946	0.8262	0.8072
sensor 10	0.8024	0.7797	0.7830	0.7813
sensor 38	0.8126	0.7868	0.8196	0.7997
sensor 39	**0.8299**	**0.8144**	0.8273	**0.8187**
sensor 40	0.8147	0.7917	0.8048	0.7975
sensor 41	0.8117	0.7871	**0.8285**	0.8021
sensor 46	0.8053	0.7808	0.8251	0.7954
sensor 70	0.7994	0.7757	0.8055	0.7873
all sensors	0.8168	0.7933	0.8250	0.8061
AE	0.8142	0.7991	0.8057	0.8023
no pretraining(base)	0.8032	0.7873	0.7934	0.7899

As for the sensor selection, the performances of the models pretrained with the sensors selected by Lasso and RF are close to the best performance achieved by the model pretrained with AE and are better than the models pretrained with the hand-selected sensors. However, some of the sensors selected by Lasso and RF performed worse than the hand-selected sensors. Thus, interpretable machine models are adequate for selecting sensors but only partially reliable.

5. Conclusions

We investigated the effectiveness of pretraining with the corresponding sensor values for the image classification task in the garbage disposal facility. In this study, we employed and evaluated several methods for selecting sensor values, including Lasso, RF, and hand selection by a specialist who has extensive experience in the garbage disposal facility. As a result of the sensor selection, some sensors selected by machine learning, Lasso and RF, achieved better performance than the hand selection, but the others included sensors with worse performance. Moreover, AE achieved good performance in both frozen and unfrozen fine-tuning, but there is a difference in the learning speed of the fine-tuning from the model pretrained with a good sensor value. Thus, we concluded that the difference in the range of feature encoding causes this difference. Future work includes a more detailed investigation of the difference between pretraining with AE and sensor value prediction. We will focus on training stability in the case of a smaller dataset for fine-tuning.

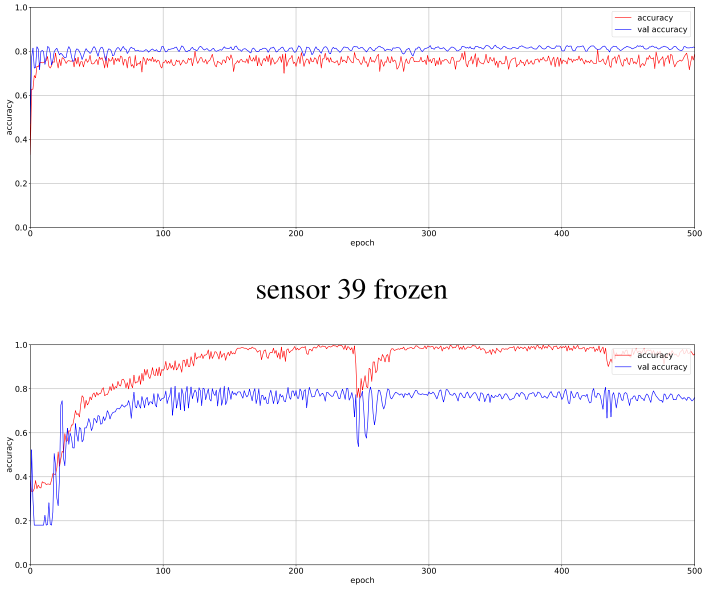
References

[1] X. Chen and K. He, "Exploring simple siamese representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 15 750–15 758.

[2] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

Table 6: Comparison with frozen parameters

model	Accuracy	Precision	Recall	F1
sensor 5	0.6480	0.6656	0.5051	0.4988
sensor 7	0.6739	0.6560	0.5970	0.6159
sensor 8	0.6764	0.6782	0.6863	0.6689
sensor 9	0.7354	0.7244	0.6931	0.7057
sensor 10	0.7587	0.7523	0.6907	0.7117
sensor 38	0.7897	0.7680	**0.7929**	0.7776
sensor 39	0.8003	0.7737	0.7859	0.7794
sensor 40	0.7973	0.7709	0.7831	0.7764
sensor 41	0.7672	0.7401	0.7315	0.7353
sensor 46	0.7388	0.7206	0.7548	0.7288
sensor 70	0.6166	0.6329	0.4709	0.4729
all sensors	0.7769	0.7490	0.7678	0.7573
AE	**0.8109**	**0.7897**	0.7909	**0.7903**



sensor 39 frozen

AE frozen

Figure 5: Learning curves of the fine-tunings after the pretraining with AE and sensor 39.

[3] V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[4] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.

[5] H. Noh, S. Hong, and B. Han, “Learning deconvolution network for semantic segmentation,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1520–1528.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.

[7] R. Ranftl, A. Bochkovskiy, and V. Koltun, “Vision transformers for dense prediction,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12 179–12 188.

[8] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929*, 2020.

[9] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 16 000–16 009.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[11] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 58, no. 1, pp. 267–288, 1996.

[12] T. K. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1. IEEE, 1995, pp. 278–282.