Training Convolutional Autoencoders with Metric Learning

Yousuke Onitsuka*, Wataru Ohyama† and Seiichi Uchida*
* *Kyushu University*
744, Motooka, Nishi-ku, Fukuoka-shi, 819-0395 JAPAN
† *Saitama Institute of Technology*
1690 Fusaiji, Fukaya-shi, Saitama 369-0293 JAPAN
Email: yosuke.onitsuka@human.ait.kyushu-u.ac.jp, ohyama@sit.ac.jp, uchida@ait.kyushu-u.ac.jp

***Abstract*—We propose a new Training method that enables an autoencoder to extract more useful features for retrieval or classification tasks with limited-size datasets. Some targets in document analysis and recognition (DAR) including signature verification, historical document analysis, and scene text recognition, involve a common problem in which the size of the dataset available for training is small against the intra-class variety of the target appearance. Recently, several approaches, such as variational autoencoders and deep metric learning, have been proposed to obtain a feature representation that is suitable for the tasks. However, these methods sometimes cause an overfitting problem in which the accuracy of the test data is relatively low, while the performance for the training dataset is quite high. Our proposed method obtains feature representations for such tasks in DAR using convolutional autoencoders with metric learning. The accuracy is evaluated on an image-based retrieval of ancient Japanese signatures.**

***Keywords*-convolutional autoencoder; metric learning; feature extraction; historical document analysis**

I. Introduction

Analyzing small size datasets with considerable intra-class variation is still a common problem in many pattern recognition tasks. In document analysis and recognition (DAR), we can find several tasks suffering from this problem, for instance, signature verification [1] [2] [3] [4], historical document analysis [5], and scene text recognition [6] [7]. While researches handling large-scale datasets attract mainstream attention in DAR community, it is still important to develop a proper method to handle small datasets.

For the small dataset problem, dimensionality reduction is a common approach. When using small datasets, we sometimes encounter the situation where a high dimensional feature space is not adequate to achieve the required performance, due to so-called curse-of-dimensionality. To prevent performance degradation by the high dimensional feature space, several dimensionality reduction methods, for instance, principal component analysis (PCA), linear discriminant analysis (LDA), and autoencoders (AEs) have been proposed. The feature mapping model for dimensionality reduction is adaptively selected for a target task. A convolutional neural network (CNN) or a convolutional autoencoder (CAE) are widely used as a feature extractor for image recognition or retrieval tasks.

On the other hand, when we have a dataset with large intra-class variation and small inter-class variation, we can employ metric learning (ML) based methods. The basic concept of the ML is that feature vectors are mapped into a metric space where the intra-class variation is suppressed and, at the same time, the inter-class difference is enlarged. If we have a dataset which contains a large enough number of samples in each class, we can expect that the ML obtains an adequate mapping to separate the overlapped classes.

Both the dimensionality reduction and ML can be represented as feature mapping functions. High-degree nonlinear mapping (HDNM) functions, such as multiple-layer neural networks, have a higher representation ability for feature mapping. For instance, HDNM functions can estimate a more complex sample distribution for the dimensionality reduction. For ML, HDNM functions can represent complicated mappings to obtain separable feature distributions. However, the performance and stability of HDNM function heavily depend on the quality and quantity of a training dataset. The higher representation ability of such mapping functions can cause overfitting to a small dataset. Correct estimation of mapping function parameters for a small class is quite difficult even for HDNM functions.

In this research, the authors propose a new training method for a multi-layer CAE. The basic concept of the proposed method, as shown in Figure 1, is to control the complexity of the dimensionality reduction by introducing a metric learning. In the proposed method, the CAE and the ML are trained simultaneously. By this training, the CAE can extract low dimensional features of which the intra-class variation is restricted. The effectiveness of the proposed method is evaluated using an ancient Japanese signature dataset, which is a typical example of a small dataset with considerable intra-class variation.

Our main contributions are summarized as follows:

1) A new training method for the CAE is proposed. This method can learn not only the feature represented by the CAE but also the distance metric that restricts the intra-class variation of the feature. Note that these two training tasks are solved simultaneously.
2) The effectiveness of the proposed method is confirmed using a typical example of the small size dataset with

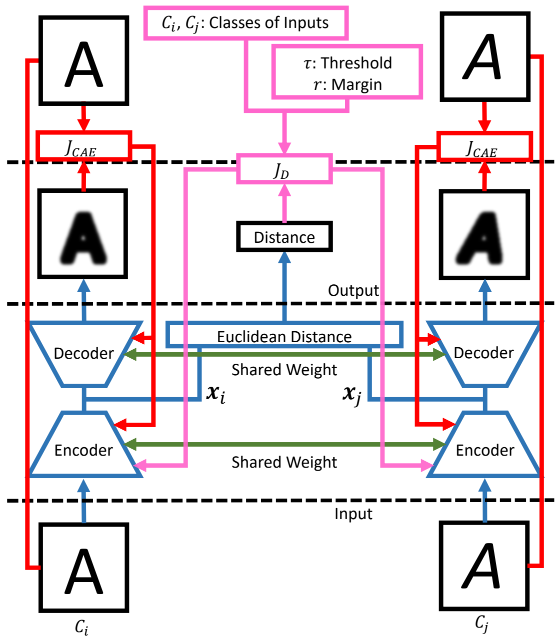


Figure 1. Flowchart of the proposed CAE with metric learning. A given pair of character images of which classes are c_1 and c_2 is input into the two CAEs which share the weights and biases. The CAEs extract a pair of feature vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ as the outputs of each encoder layer. The Euclidean distance between $\mathbf{x}_1$ and $\mathbf{x}_2$ is calculated as a dissimilarity measure for c_1 and c_2. The CAEs are trained to minimize the CAE loss function defined by mean squared error. Simultaneously, encoder layers in CAE are trained to minimize the ML loss function defined by Eq.(6).

the considerable intra-class variation.

3) The proposed method can handle such the difficult data more properly than standard CAEs and ML methods.

II. Related Work

Handling a small dataset with a considerable intra-class variation is a common problem in DAR. In this research, the authors consider a topic from historical document analysis as a typical example of this problem.

While some large-scale datasets are available in the literature of historical document analysis, for instance, Kuzushiji-MNIST [8], collecting a large enough size of an annotated dataset for particular targets is a tedious and time-consuming task. To support annotation by experts, image-based retrieval systems have been proposed. When a comprehensive dataset is available, a deep neural network can be adapted successfully [9]. However, in many cases in historical document analysis, handcrafted features and traditional similarity or dissimilarity calculations are employed for retrieval tasks [10], [11].

When the size of a dataset is small, overfitting of classifiers can happen. The overfitting may degrade the performance of classification or retrieval of test data. To prevent overfitting and to extract efficient feature components, dimensionality reduction or feature extraction is a conventional technique. Traditionally, the linear mapping of feature vectors into a lower dimensional subspace, for instance, PCA and LDA [12], are widely used. Neural networks are also adapted for feature extraction to handle more complex structures in a feature space. A multi-layer autoencoder (AE) [13] and a convolutional autoencoder (CAE) [14] have been proposed and successfully adapted for obtaining efficient subspaces by end-to-end training. Recently, some trials were conducted to introduce genetic models into an AE architecture [15] to compensate for the data size.

For datasets which include considerable intra-class variation, metric learning (ML)-based methods can be an efficient tool. The ML-based methods map an original feature sample into a metric space where the intra-class variation is restricted. To improve the separability between distributions of multiple classes, discriminative metric learning [16] and its extension using neural networks [17], [18] have been proposed.

III. The Proposed Method

The proposed method is motivated by the idea of deep discriminative metric learning (DDML) [17], which is an ML by a neural network-based learning scheme. In this section, we first briefly review the DDML, and then present the details of the proposed CAE training with metric learning.

A. Discriminative deep metric learning

In the conventional ML-based methods, a distance metric $d_f^2(\mathbf{x}_i, \mathbf{x}_j)$ between two given d-dimensional feature samples $\mathbf{x}_i \in \mathbb{R}^d$ and $\mathbf{x}_j \in \mathbb{R}^d$ is defined by the following equation,

$$d_f^2(\mathbf{x}_i, \mathbf{x}_j) = \|f(\mathbf{x}_i) - f(\mathbf{x}_j)\|_2^2, \tag{1}$$

where $f : \mathbb{R}^d \mapsto \mathbb{R}^k$ is a function which maps the given a sample into a k-dimensional metric space. The common objective of the ML-based methods is to obtain a good distance metric which evaluates the distance between a pair of samples smaller (or larger) if the samples belong to the same (or different) classes. By the definition of the distance metric in Eq.(1), the objective of ML-based method is equivalent to optimize the mapping function $f(\mathbf{x})$.

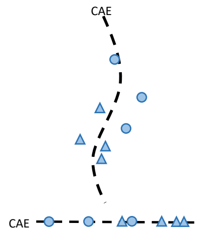
In DDML, the mapping function $f(\mathbf{x})$ is defined as multiple layers of nonlinear transformations,

$$f(\mathbf{x}) = \mathbf{h}^{(M)}, \tag{2}$$

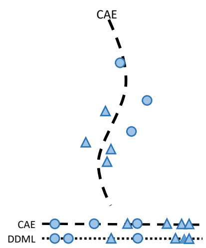
$$\mathbf{h}^{(m)} = s\left(\mathbf{W}^{(m)}\mathbf{h}^{(m-1)} + \mathbf{b}^{(m)}\right) \in \mathbb{R}^{p^{(m)}}, \tag{3}$$

$$\mathbf{h}^{(1)} = s\left(\mathbf{W}^{(1)}\mathbf{x} + \mathbf{b}^{(1)}\right) \in \mathbb{R}^{p^{(1)}}, \tag{4}$$

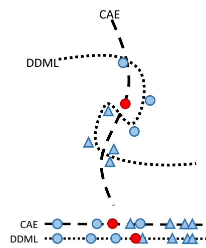
Here, $\mathbf{W}^{(m)} \in \mathbb{R}^{p^{(m)} \times p^{(m-1)}}$ is a projection matrix and $\mathbf{b}^{(m)}$ is a bias vector in the m-th layer. They are parameters to be optimized through learning. The function $s : \mathbb{R} \mapsto \mathbb{R}$ is a nonlinear activation function.



(a) The original 2D feature space and the 1D subspace by CAE.



(b) DDML with the CAE subspace of (a).



(c) Simultaneous optimization of DDML and CAE.

Figure 2. Three scenarios of mapping samples from two-dimensional original space into a one-dimensional subspace.

The parameters of DDML is obtained by minimizing the following loss function:

$$\begin{aligned} J_{\mathrm{ML}} &= J_{\mathrm{D}} + J_{\mathrm{N}} \\ &= \frac{1}{2}\sum_{i,j} g\left(r - l_{ij}(\tau - d_f^2(\mathbf{x}_i, \mathbf{x}_j))\right) \\ &\quad + \frac{\lambda}{2}\sum_{m=1}^{M}\left(\left\|\mathbf{W}^{(m)}\right\|_F^2 + \left\|\mathbf{b}^{(m)}\right\|_2^2\right), \end{aligned} \tag{5}$$

where $g(z) = \frac{1}{\beta}\log\left(1+\exp(\beta z)\right)$is the generalized logistic loss function [18]. τ, r, λ and β, respectively are parameters corresponding to a threshold, a margin from the threshold, a regularization parameter and a sharpness parameter. $\|\mathbf{W}\|_F$ denotes the Frobenius norm of the matrix $\mathbf{W}$. $l_{ij} = 1$ and $l_{ij} = -1$ if $\mathbf{x}_i$ and $\mathbf{x}_j$ belong to the same and the different classes, respectively. The details of the optimizing algorithm are provided in the paper by Hu et al [17].

B. Convolutional autoencoder and deep metric learning

In the the original work of DDML [17], it is applied to some hand-crafted features. However, DDML is also applicable to features extracted by a CNN or a CAE. This fact raises a new idea: is it possible to learn CAE and DDML simultaneously (i.e., in an end-to-end manner) so that feature representation and its distance metric are optimized for each other? If possible, is there any merit?

Figure 2 illustrates this merit in a two-class classification problem in a two-dimensional feature space. The training samples are depicted by a circle or a triangle. In this figure, we compare three combinations of feature extraction and metric learning in a lower-dimensional sub-space. (The straight line below the two-dimensional space suggests the one-dimensional subspace trained by CAE and/or DDML.)

As shown in Figure 2(a), CAE is the method to obtain a low-dimensional subspace which optimally approximates the distribution of training samples. The dotted curve shows the subspace. Since the CAE is trained without any class information, the training samples are mapped to its subspace while loosing their class separability, as shown in the straight line in the bottom of (a).

We can employ DDML to improve the separability of two class samples. When we apply DDML to the fixed CAE features, only a distance metric in the subspace is optimized (Figure 2(b)). We can expect some improvement of separability and classification accuracy between the two classes. However, DDML with fixed features has a limitation where the overlap of the two classes is not resolved.

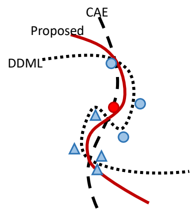
Contrast to the above two combinations, when DDML with CAE is trained in an end-to-end manner, it optimizes not only the distance metric but the feature representation. Thanks to the high representation ability of DDML, CAE can obtain a complex shape subspace which optimizes the distance metric between two classes. However, this high representation ability easily causes overfitting to the training dataset. For instance, let we have a test sample from the circle class (shown by the red circle in Figure 2(c)). The test sample may be correctly classified into the circle class by DDML with fixed CAE features because of the distance from the circle class is smaller than that from the triangle class in the subspace. However, the test sample may be misclassified by DDML with CAE because of the smaller distance from the triangle class due to overfitting of the subspace.

C. CAE training with metric learning

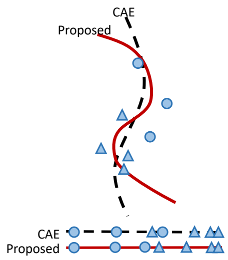
We propose a CAE training with metric learning. The proposed method controls the representation ability of the CAE model by introducing a distance metric learning to the output of encoder layer.

Figure 3 illustrates the basic concept of the proposed method. The main objective of this method is to obtain a better optimized subspace while suppressing overfitting. The proposed method intends to obtain a subspace such as the red thick curve in the subfigures. While the subspace has a higher complexity to represent the feature distribution, the representation ability is restricted to prevent overfitting the small dataset.

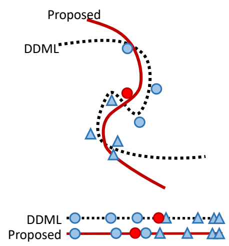
Figure 1 shows the flowchart of the proposed training method for CAE with metric learning. The proposed method consists of two AEs which share weights and biases. The dissimilarity between the two input images is calculated by the Euclidean distance between the extracted feature vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ as the output of the encoder layer. In this model, training of the AE and distance metric are performed simultaneously.



(a) The proposed method



(b) CAE vs the proposed method



(c) DDML vs the proposed method

Figure 3. The basic concept of the proposed method. The red solid line shows the low dimensional subspace obtained by the proposed method. While the proposed method has a higher complexity to represent the distribution of training data, the problem of overfitting can be suppressed.

The loss function in the proposed method is defined by the following equation.

$$J_{\text{CAEMC}} = (1-\alpha)J_{\text{CAE}} + \alpha J_{\text{D}}, \tag{6}$$

where J_{CAE} is the loss function for CAE, which is calculated by the binary cross entropy function. J_{D} is the loss function for the distance metric, which is defined in Eq.(5) with $d^2(\mathbf{x}_i, \mathbf{x}_j)$ equals Euclidean distance between the output of encoder layer. α is a parameter to balance the weight between AE and metric such that J_{CAE} and J_{D} have a similar range.

IV. Experiments

We conducted experiments to evaluate the effectiveness of the proposed method for a small-size dataset with considerable intra-class variation. We used a dataset of ancient Japanese signature images as a typical example of the problem. In the following subsections, first, we explain the dataset in detail. Second, we describe the concrete setting of the experiments, and finally, discuss the evaluation method.

A. Dataset

The dataset used for the evaluation experiments is a collection of Japanese ancient signatures, as known as “Kaou” images. Kaou is a special symbol written in ancient letters in eastern Asia to identify the writer. Figure 4 shows some image examples from the dataset. We observe that the signatures consist of hand-written brush strokes. The dataset contains the following typical properties;

1) The dataset has a large intra-class (inner-person) variety. The shape of signatures changes drastically due to aging or changing social position of the writer.

(a) writer A (AD.1376)

(b) writer A (AD.1382)

(c) writer B

Figure 4. Examples of ancient Japanese signature, “Kaou”. (a) by writer A at AD. 1376, (b) by writer A at AD. 1382, and (c) by writer B. Signatures contains a large intra-class variation even by a same writer due to aging or social promotion. Although (b) and (c) are signatures of different writers, the shape of them is very similar.

Figures 4 (a) and (b) are the signatures of the same person. While the time gap between them is 6 years, his signature had changed due to the promotion of his social position.

2) The shape of some signature writers is quite similar. This means that the dataset contains some classes in which inter-class (inter-person) difference is significantly small. For instance, Figures 4 (b) and (c) are signatures from different writers; however, the shape of these signatures is similar. This frequently happens due to historical custom.
3) The size of each writer class is basically small and imbalanced. The ancient signature images are collected from historical documents. Therefore, the total dataset size is restricted. Almost all the classes have less than 5 signature image samples.

The dataset consists of 3,266 binary images from 1,582 writer classes. There is a wide variety in the number of images par writer class (the smallest and the largest are 1 and 33, respectively). An input image contains one cropped signature in the size of 256×256 pixels.

B. Structure of CAE

The CAE consists of 22 layers in total, with 11 layers each for the decoder and encoder. In the decoder, 5 pairs 5×5 of convolution filters with padding and 2×2 max-pooling kernels are located. In the encoder, the same number of pairs of convolution filters and up-sampling kernels are used. The numbers of filters of the convolution layers are 16, 16, 8, 8, 4, 4, 8, 8, 16, and 16 in order from the input side. Since the input and output images are binary images, binary cross entropy is used as the loss function. The ReLU activation function is used, except in the output layer which uses sigmoid functions.

C. Setting of parameters

The proposed method contains some parameters to determined in advance. We determined these parameters by the following steps;

1) The α in Eq.6 is determined by the following steps. First, we calculated a converged J_{CAE} and J_{D} using two independent networks trained using only J_{CAE} and J_{D}, respectively. Next, the α defined by the solution of the equation $(1-\alpha)J_{\text{d}} - \alpha J_{\text{d}} = 0$.

2) The threshold τ is set to equal the median between $d_{\max}$ and $d_{\min}$, defined by $(d_{\max} + d_{\min})/2$. Where $d_{\max}$ and $d_{\min}$ denote the maximum and minimum distance values calculated between two samples selected from the same class.
3) The margin r is defined by the difference between $d_{\max}$ and the threshold τ.
4) The regularization parameter λ is set to 0.01, which is the default value in Keras library.
5) The sharpness parameter β controls the converged value of loss function. We selected a beta value from some predefined candidates, which maximizes the converged value of loss function.

D. Experimental protocol

To evaluate the effectiveness of the proposed CAE training with the metric constraint for feature extraction, we conducted an image retrieval experiment. In the experiment, the dissimilarity between two signature images is calculated by a distance metric between the extracted features. A retrieving system outputs a single or multiple candidates corresponding to a given query. The output candidates are usually sorted by dissimilarity measure in ascending order. In this experiment, each signature image in the test set is input as a query, top-n candidates are selected from a gallery set.

The dataset is divided into three subsets, i.e., training, validation, and test sets. Since the size of the entire dataset is relatively small, we split the randomly selected subset as evaluation data. The CAE is trained using remaining training and validation sets. The training set is also used as the gallery. The number of samples contained in training, validation, and test sets are 2835, 316, and 175, respectively. We repeat this random splitting 10 times to obtain statistical results for performance evaluation.

We evaluate the retrieval performance using the cumulative accuracy profile (CAP) against the number of top candidates n for each query. When at least one sample among the candidates belongs to the same writer class as that of the query, the query is counted as correct. The accuracy score is calculated by dividing the number of corrects by the number of queries.

We compare the following three distance metrics for retrieval performance.

1) Euclidean distance between features extracted by the pre-trained CAE (CAE+Euclid) to confirm a baseline performance.
2) Distance metric obtained by DDML applied to the pre-trained CAE (CAE+DDML).
3) Euclidean distance between features extracted by the proposed CAE with metric learning (proposed),

V. Results

Figure 5 shows the CAP curves obtained by each distance metric. In the figure, the baseline performance is corresponding to the CAP by the Euclidean distance with the pre-trained CAE. While the CAP is increased by introducing the proposed learning method with the metric constraint, the CAP is significantly decreased by applying DDML to the pre-trained CAE.

Figure 6 shows some examples of retrieving results by the three distance metrics. In each row, the left-most image indicated by the blue frame is a query. The 10 candidates corresponding to the query in the same row are shown in ascending order from left to right. The red frames on candidate images denote that the candidate belongs the to same writer class as that of the query. The scatter plots in the right-hand size to visualize the retrieval results in two dimensional feature space. In these figures, the extracted features are projected to 2D subspace using PCA. Query images are inverted to be identified easily. The color of a signature image frame indicates the writer class of the signature.

We observe the proposed method and improve the retrieval performance compared to the baseline results. Candidate images having the same writer class as query are given higher ranking by the proposed method. In contrast to this, the degradation of retrieval performance by DDML is also observed in the examples. Especially, for the two queries (A and C), it is intuitively difficult to understand the reason of the candidate selection.

As mentioned in III-B, our concern is that these phenomena are caused by the overfitting by DDML for the small dataset. To address this concern, we visualize the feature space obtained by the three methods. In the feature space obtained by DDML(Figure 6(b)), we can see that the samples create more concentrated clusters than that in CAE (Figure 6(a)) and the proposed method (Figure 6(c)). However, the query C (indicated by the arrow C) is out of the cluster although it belongs to the same writer class as the query B (the cluster indicated by arrow B). Actually, the query C contains irregular deformation which is not seen in the cluster B, where the right-bottom tail of signature is cut off. Once the overfitting occurs, it is quite difficult to predict the behavior of mapping against unseen variations of input images. Contrast to DDML(Figure 6(b)), the proposed generates more ideal distribution for these examples thanks to the learning CAE with metric learning.

These results suggest that the proposed training method is suitable for a small dataset with considerable intra-class variations.

VI. Conclusion

We proposed a new training method that enables a CAE to extract more useful features for retrieval or classification tasks with limited size datasets. The purpose was to extract the features in which class information was reflected as distance. In achieve this objective, we proposed a new training method combining CAE and metric learning. The

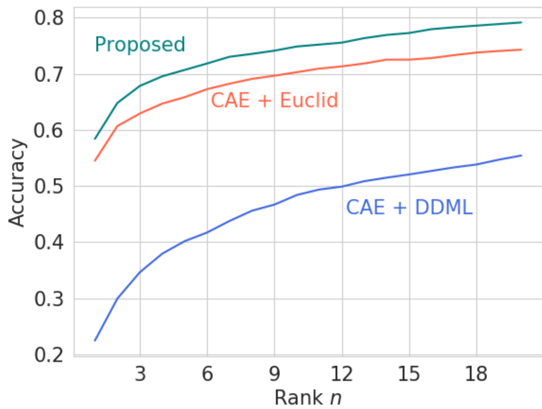
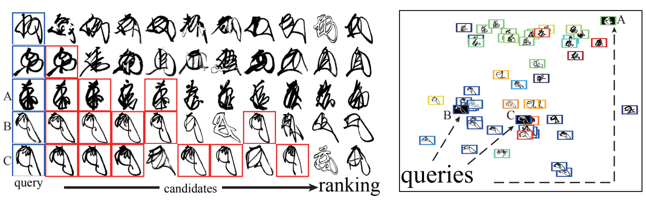
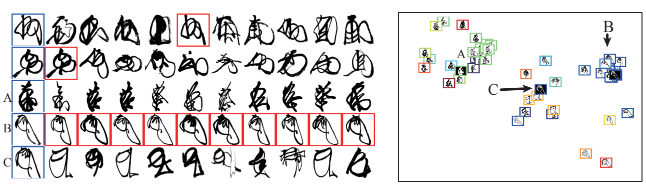


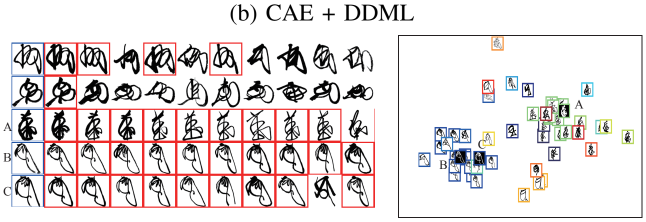
Figure 5. The cumulative accuracy profile (CAP) curve by three distance metrics. The CAP by the proposed method shows the better performance compared to both the baseline and the DDML.



(a) CAE + Euclid (baseline results)



(b) CAE + DDML



(c) The proposed method

Figure 6. Examples of retrieving results by three distance metrics (left) and visualization of the results in 2-D feature space projected by PCA (right). (a) CAE+Euclid (baseline results), (b) CAE+DDML, and (C) the proposed CAE learning with metric learning. The left-most image with the blue frame is a query. The 10 candidates corresponding to the query in the same row are shown in ascending order from left to right. The red frames denote that the candidate belongs the same writer class as that of the query.

effectiveness of the proposed method was evaluated using the ancient Japanese signature dataset which is a typical example of the target program. The results suggest that this method is suitable for extracting features reflecting class information in addition to the features obtained by CAE. Future research topics could include further investigation of the effectiveness of the proposed method using other datasets and adaptation for standard autoencoders.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Numbers JP17H00921 and JP17H06100.

References

[1] K. Cpałka, M. Zalasiński, and L. Rutkowski, "A new algorithm for identity verification based on the analysis of a handwritten dynamic signature," *Applied Soft Computing*, 2016.

[2] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Learning features for offline handwritten signature verification using deep convolutional neural networks," *Pattern Recognition*, 2017.

[3] M. Diaz, A. Fischer, M. A. Ferrer, and R. Plamondon, "Dynamic signature verification system based on one real signature," *IEEE Transactions on Cybernetics*, 2018.

[4] M. Zalasiński and K. Cpałka, "A new method for signature verification based on selection of the most important partitions of the dynamic signature," *Neurocomputing*, 2018.

[5] B. K. Barakat and J. El-Sana, "Binarization free layout analysis for arabic historical documents using fully convolutional networks," in *ASAR*, 2018.

[6] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *TPAMI*, 2015.

[7] M. Liao, B. Shi, and X. Bai, "Textboxes++: A single-shot oriented scene text detector," *IEEE Transactions on Image Processing*, 2018.

[8] T. Clanuwat, M. Bober-Irizar, A. Kitamoto, A. Lamb, K. Yamamoto, and D. Ha, "Deep learning for classical japanese literature," *arXiv preprint arXiv:1812.01718*, 2018.

[9] C. Sirirattanapol, Y. Matsui, S. Satoh, K. Matsuda, and K. Yamamoto, "Deep image retrieval applied on kotenseki ancient japanese literature," in *ISM*, 2017.

[10] K. Kameyama, S.-N. Kim, M. Suzuki, K. Toraichi, and T. Yamamoto, "Content-based image retrieval of kaou images by relaxation matching of region features," *IJUFKS*, 2006.

[11] A. Kitadai, M. Inoue, Y. Takata, G. Fang, H. Baba, A. Watanabe, and S. Inoue, "Technologies and improvements of image search service for handwritten character patterns on japanese historical documents," in *ICDAR*, 2017.

[12] A. M. Martínez and A. C. Kak, "Pca versus lda," *TPAMI*, 2001.

[13] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, 2006.

[14] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *ICANN*, 2011.

[15] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[16] H. Wang, L. Feng, J. Zhang, and Y. Liu, "Semantic discriminative metric learning for image similarity measurement," *IEEE Transactions on Multimedia*, 2016.

[17] J. Hu, J. Lu, and Y.-P. Tan, "Discriminative deep metric learning for face verification in the wild," in *CVPR*, 2014.

[18] A. Mignon and F. Jurie, "Pcca: A new approach for distance learning from sparse pairwise constraints," in *CVPR*, 2012.