CLASS SIZE VARIANCE MINIMIZATION TO METRIC LEARNING FOR DISH IDENTIFICATION

SHILONG FENG[1], HAOQIN XIE[1], HONGBO YIN[1], XIAOPENG CHEN[2], DESHUN YANG[2], PATRICK P. K. CHAN[1]

[1]School of Computer Science and Engineering, South China University of Technology, Guangzhou, China
[2]Guangzhou Paike Pushi Information Technology Co., Ltd., Guangzhou, China
E-MAIL: 1824101490@qq.com, xavier.xhq@qq.com, yinhbo@outlook.com, 783820092@qq.com, 1063411703@qq.com, patrickchan@scut.edu.cn

Abstract:

The objective of metric learning is to search a suitable metric for measuring distance or similarity between samples. Usually, it aims to minimize the distance between samples of same class and maximizes the distance between samples of different classes. However, most metric learning methods do not consider the sizes of classes, which may cause negative impact on the performance in classification since the size of a cluster is usually ignored in the distance comparison. In this work, we propose a triplet loss with variance constraint. Our method focuses not only on the distances between samples but also on the sizes of classes. The size difference between classes is also minimized in our objective function. The experimental results confirm that our method outperforms the one without the class size variance.

Keywords:

Metric learning; Triplet loss

1 Introduction

Metric learning [1] aims to represents the similarity of objects by a distance function. For a classification problem, the distance between the objects in the same class is expected to be small; otherwise, the distance should be large. Different from the traditional classification which aims to find a separating plane, metric learning focuses on adjusting the metric to separate the objects of different classes. Due to its flexibility and adaptability, metric learning has been successfully applied to applications, *e.g.* face verification system [2] and person re-identification system [3].

One of research directions of the metric learning focuses on the objective function in training [4]. For example, triplet loss [5] is most commonly used objective function in the metric learning. It ensures that an object (*i.e.*anchor) is closer to other objects in the same class (*i.e.*the positive samples) than to the others in different classes (*i.e.*the negative samples). However, the most of the existing methods does not consider the similarity of clusters' sizes. As in classification, an unseen object may be ranked according to the distance between the object and the center. The sizes of clusters are ignored in ranking. The unseen sample may be classified as the small cluster although it is located inside a large cluster if it is closer to the center of the small cluster.

This work aims to demonstrate the importance of similarity of clusters' size when applying metric learning to a classification problem. The variance of the average distance between the center and samples for all classes is considered in the objective measure. Our method encourages the similar sized clusters by minimizing their variances. The variation of the cluster size is small for classes trained by our proposed method comparing with the traditional method. The better performance should be achieved by our proposed method since the factor of radius of a cluster is removed.

The rest of the paper is arranged as follows. Section 2 introduces the related concepts of this study including the metric learning and its objective functions. The proposed method is discussed in detail in Section 3, and evaluated experimentally in Section 4. Finally a conclusion is given in Section 5.

2 Related work

The common objective functions used in metric learning are introduced and discussed in this section.

2.1 Distance based method

Distance based method is one of the common methods in metric learning. The intra-class and inter-class distances are generally considered. The aim of metric learning is to minimize the intra-class distance and also maximize the inter-class distance in order to separate different classes clearly.

Some methods [6–8] consider the absolute value of the distance. Usually, the intra-class or inter-class distances are bounded by a pre-defined value. For example, MLSI [1] minimizes the intra-class distance with the constraint which inter-class distance should be smaller than a threshold. However, the determination on the absolute distance value is difficult. The bound on the distances is ignored in the relative distance methods. These methods focus on the difference between the intra-class and inter-class distance. For example, the difference between intra-class and inter-class distances is maximized [9].

2.1.1 Triplet loss

Triplet loss [5] is one of most famous and commonly used objective functions in metric learning. It ensures that an object (*i.e.*anchor) is closer to other objects in the same class (*i.e.*the positive samples) than to the others in different classes (*i.e.*the negative samples), which is defined as:

$$\begin{aligned} &\|f(x_i^a) - f(x_i^p)\|_2^2 + \alpha < \|f(x_i^a) - f(x_i^n)\|_2^2, \\ &\forall (f(x_i^a), f(x_i^p), f(x_i^n)) \in \Gamma \end{aligned} \tag{1}$$

where x_i^a is the i^{th} anchor, x_i^p and x_i^n are the positive and negative of x_i^a. $f(\cdot)$ is a function mapping a sample to a trained space. α is a margin that is enforced between positive and negative pairs, and Γ is the set of all possible triplets in the training set.

The triplet loss is defined as:

$$L_{tri} = \sum_{i}^{N} \left[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right]_+ \tag{2}$$

where N is the number of all triplets. $[\cdot]_+$ represents the hinge function defined as: $L(m_i) = \max(0, 1 - m_i(w))$. By minimizing (2), samples of same class are closer together while samples in different classes will be moved further away.

A study [5] points out that some triplets easily satisfy the triplet constraint (*i.e.*the anchor is closer to a positive sample than to a negative one). These triplets would not contribute to the learning but slow down the training procedure. Hard triplet [5] is proposed to train a model only using the triplets that violate the constraint. A number of studies, for example, a multi-stage training strategy [3], and MMD-NCA [10] revise triplet loss function in order to learn better separation on different kinds of objects.

3 Proposed method

After the metric learning, the difference between the size of classes is usually ignored in classification. Figure 1 shows an example when the size of two clusters are different. Assume the unseen sample denoted by the red triangle belongs to the class 1. Although it locates inside the cluster of the class 1, it is classified as the class 2 since it is closer to the class 2. This example shows that the radius is usually ignored in the classification process. As a result, this paper proposes a novel objective function which minimizes the variance of clusters' size based on triplet loss.

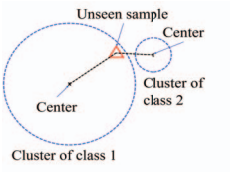


FIGURE 1. An example of the clusters with different sizes. The distance between an unseen sample and a cluster center of a class is used in classification. Although the unseen sample is located inside the cluster of the class 1, it will be classified as the class 2.

3.1 Triplet loss with variance

The proposed loss function defined as (3) contains two parts: the traditional triplet loss in (2) and the variance constraint, which discourage the variance of the clusters' sizes of classes.

$$Loss = (1 - \beta) * L_{tri} + \beta * L_{var} \tag{3}$$

where β is a hyperparameter to control the importance of two terms. We first define the size of a class ($s^{(i)}$) as the average distance of a sample and the center of its class. Assume $\mathbf{x}^i$ is a sample in the class i in (4).

$$s^{(i)} = \frac{1}{N^{(i)}} \sum_{\mathbf{x} \in C^{(i)}} d(\mathbf{x}, \bar{\mathbf{x}}^{(i)}) \tag{4}$$

where $C^{(i)}$ denotes a set contains all samples of the class i, $N^{(i)} = |C^{(i)}|$ denotes the number of samples in the class i, and $\bar{\mathbf{x}}^{(i)}$ is the average of all samples of the class i, *i.e.*$E_{\mathbf{x} \in C^{(i)}}(\mathbf{x})$.

d is a distance function. L_{var} is then defined as the variance of $s^{(i)}$ for all classes:

$$L_{var} = var_{i=1,2,...,c}(s^{(i)}) \tag{5}$$

where c represents the number of classes.

4 Experiments

The experimental evaluation on the proposed and the traditional triplet loss function is discussed in this section. The dataset is introduced including the collection process and data description in Section 4.1. The experimental setting is described in Section 4.2. The experimental results and discussion are included in Section 4.3.

4.1 Datasets

Two different datasets of dish identification (*i.e.D1* and *D2*) are collected from a restaurant in summer and winter. An automatic identification machine (shown in Figure 2) is set up in a restaurant. A photo is taken on a tray after a customer chooses the dishes, shown in Figure 3. Each dish is then segmented from the image by edge detection [11] and feature points detection and alignment [12], and resized to 128×128 pixels. Examples of a segmented dish from *D1* and *D2* are illustrated in Figure 4. Finally, 2,695 images of 77 classes, and 1,260 images 36 classes are collected in *D1* and *D2*.

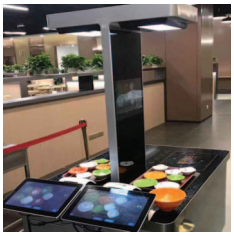


FIGURE 2. The automatic identification machine in a restaurant.

FIGURE 3. A photo taken by the automatic identification machine.

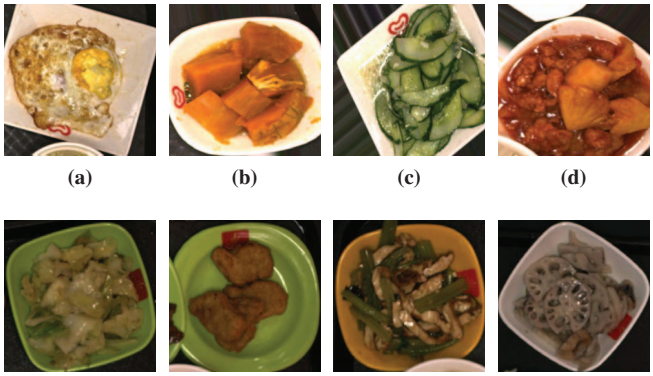


FIGURE 4. Examples of dish images in two collected datasets. Images in the first and second row are from *D1* and *D2*.

4.2 Experimental setting

For each experiment, a dataset is divided randomly into training and test sets. Training and test sets contain 5 and 30 images for each class in order to avoid the imbalance problem. Data augmentation [13–17] is applied to the datasets. The data is then augmented by rotation, translation, cropping, scaling, flipping and brightness adjustment. We set a specific range for each parameters, and a parameter set is randomly chosen for each augmentation. Two additional samples are generated for each sample in *D1*, while additional one sample is generated for *D2*. The model used for feature extraction in our method is ResNet-50 [18] due to its satisfying performance in many applications [19–22]. All samples in the augmented training set will be used as an anchor once in each epoch. For each anchor, a positive and negative sample will be randomly selected from the augmented training set to form a triplet. 50% chance to use a hard triplet otherwise just triplet each time in our method. β is set from *0.1* to *1.0* with the interval 0.1. The variance of classes' size is calculated in each batch which contains 128 samples. The loss defined as (3) is minimized by the back-propagation.

The centers of classes are calculated by using augmented training samples in the feature space mapped by the trained model. The distance between an unseen sample and these centers is calculated in the feature space in recognition. Three commonly used metrics including accuracy, F1-score and Normalized Mutual Information (NMI) are applied to evaluate the performance of the model. Each experiment is carried out for three independent runs.

4.3 Experimental results and analysis

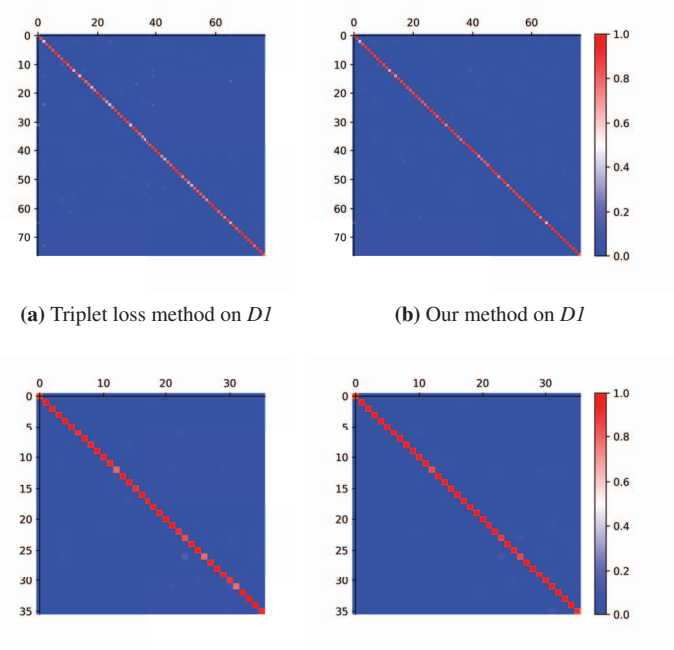
Figure 5 shows the results of our method with different β and the triplet loss method in terms of accuracy, F1-score and NMI on *D1* and *D2*. As with different β is independent to the triplet loss method, the values of the triplet loss method is constant. When β tends to 0, our method is closer to the triplet loss method; while β tends to 1, the triplet loss is ignored and the newly added term is dominant.

The performance of all methods is consistent by all evaluation methods. When β is relatively small, our method achieves slightly better performance than the triplet loss method. This indicates that considering the variance of the clusters' size is useful in learning. When β reaches to a threshold, which is around 0.4 - 0.6, the performance of our method drops significantly. As the variance term aims to complement the learning, the performance of the model is bad when the variance dominates the loss function. As a result, the value of β should be between 0.1 - 0.4 in order to avoid the domination of the variance term.

The errors made by our method with $\beta = 0.3$ and triplet loss method is illustrated by the confusion metric, shown in Figure 6. The color closer to blue (red) indicates the smaller (larger) value (*i.e.*the percentage of samples of a class are assigned to another class). Y-axis is the true class and x-axis is the predicted class. The figures illustrate that after considering the additional variance term, the number of the mis-classified samples is reduced significantly. The color of our method is more trends to blue than the one of the triplet loss method. It suggests that the variance of cluster is a critical factor in metric learning.

The change of average inter-class distance, and variance of clusters' size of the results of out method with different β are shown in Figure 7. All values are normalized linearly into 0 to 1 for better illustration. Our method becomes the triplet loss method when $\beta = 0$. Generally, the values of average inter-class distance (denoted by the blue lines) and the variance of clusters' size (denoted by the green lines) are inversely proportional to β, which means minimizing the variance term not only reduces the variance of clusters' size but also sacrifices the average inter-class distance. As closer classes are more difficult to be distinguishable, using larger β may downgrade the performance.

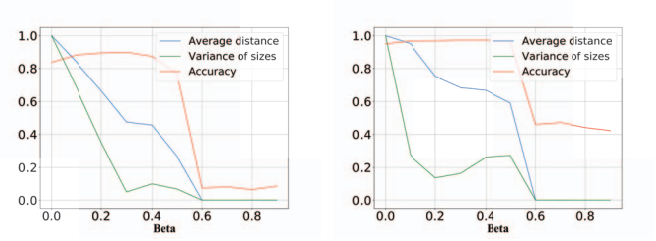
The correlation between the average inter-class distance and predict probability of our method and the triplet loss method is given in Table 1. The relation in our method is stronger (*i.e.*less than -0.86) than the one in the triplet loss function (*i.e.*less than -0.62). It explains that the variance of clusters' size affects the decision. After reducing the influence of the variance, the relation between the average inter-class distance and predict prob-



(a) Triplet loss method on *D1* **(b)** Our method on *D1*

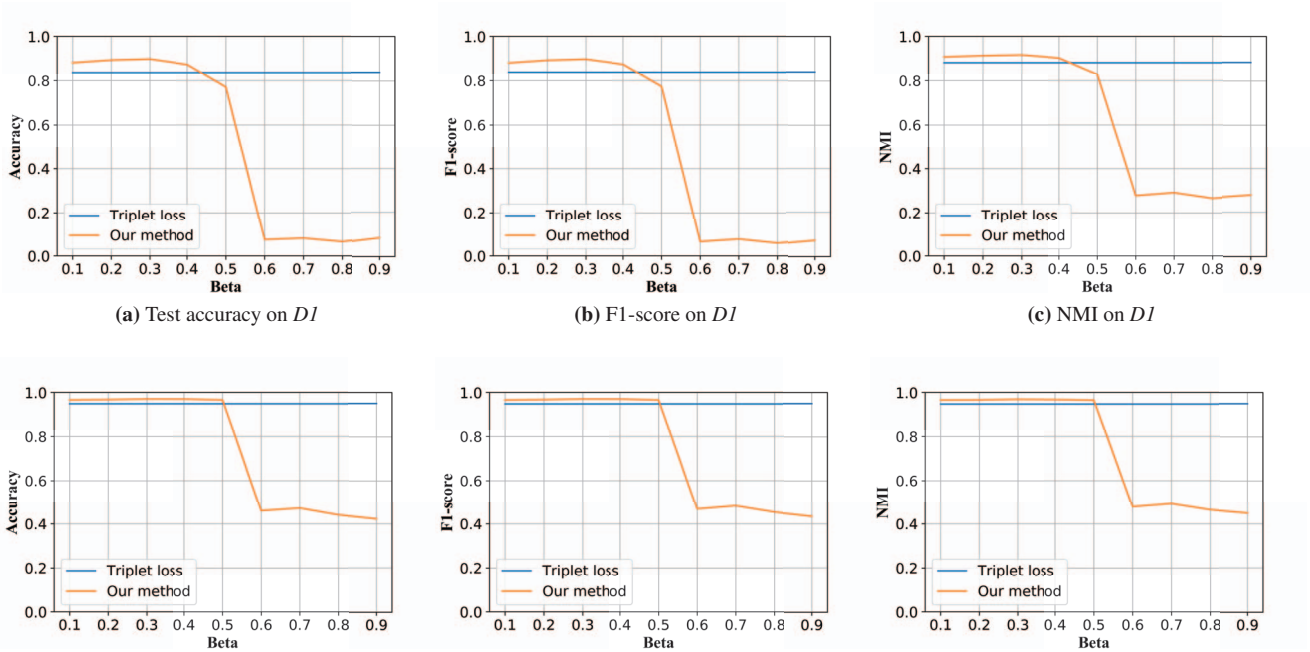
(c) Triplet loss method on *D2* **(d)** Our method on *D2*

FIGURE 6. Confusion matrices of our method with $\beta = 0.3$ and the triplet loss method on *D1* and *D2*.



(a) *D1* **(b)** *D2*

FIGURE 7. Average inter-class distance, variance and accuracy of the results of our proposed method in *D1* and *D2*.



(a) Test accuracy on *D1* **(b)** F1-score on *D1* **(c)** NMI on *D1*

(d) Test accuracy on *D2* **(e)** F1-score on *D2* **(f)** NMI on *D2*

FIGURE 5. Accuracy, F1-score and NMI of our method and the triplet loss method on *D1* and *D2*

ability is stronger.

TABLE 1. Correlation between the average inter-class distance and predict probability.

Dataset	Triplet loss	Our method
D1	-0.628	-0.865
D2	-0.733	-0.872

5 Conclusion

This work proposes a new objective function considers the triplet loss and the variance of the clusters' size of classes. Our method considers not only the distances between samples but also the size difference between classes. The hyper parameter β is applied to balance the importance of the triplet loss and the variance. The experimental results show that the proposed method is more accurate than the traditional one. This study shows that the size difference of classes influences the classification accuracy. Reducing the variance is useful for accurate prediction.

Acknowledgements

This work is supported by Natural Science Foundation of Guangdong Province, China (No.2018A030313203), the Fundamental Research Funds for the Central Universities South China University of Technology (No.2018ZD32).

References

[1] E. P. Xing, M. I. Jordan, S. J. Russell, and A. Y. Ng, "Distance metric learning with application to clustering with side-information," in *Advances in neural information processing systems*, 2003, pp. 521–528.

[2] S. Chopra, R. Hadsell, Y. LeCun *et al.*, "Learning a similarity metric discriminatively, with application to face verification," in *CVPR (1)*, 2005, pp. 539–546.

[3] Y. Zhang, Q. Zhong, L. Ma, D. Xie, and S. Pu, "Learning incremental triplet margin for person re-identification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 9243–9250.

[4] D. Li and Y. Tian, "Survey and experimental study on metric learning methods," *Neural networks*, vol. 105, pp. 447–462, 2018.

[5] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[6] Y. Ying and P. Li, "Distance metric learning with eigenvalue optimization," *Journal of machine Learning research*, vol. 13, no. Jan, pp. 1–26, 2012.

[7] P. Jain, B. Kulis, J. V. Davis, and I. S. Dhillon, "Metric and kernel learning using a linear transformation," *Journal of machine Learning research*, vol. 13, no. Mar, pp. 519–547, 2012.

[8] A. Mignon and F. Jurie, "Pcca: A new approach for distance learning from sparse pairwise constraints," in *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012, pp. 2666–2672.

[9] J. Hu, J. Lu, and Y.-P. Tan, "Deep metric learning for visual tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 11, pp. 2056–2068, 2015.

[10] H. Coskun, D. Joseph Tan, S. Conjeti, N. Navab, and F. Tombari, "Human motion analysis with deep metric learning," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 667–683.

[11] X. Ma, B. Li, Y. Zhang, and M. Yan, "The canny edge detection and its improvement," 10 2012, pp. 50–58.

[12] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov 2004. [Online]. Available: https://doi.org/10.1023/B:VISI.0000029664.99615.94

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[15] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *CoRR*, vol. abs/1712.04621, 2017. [Online]. Available: http://arxiv.org/abs/1712.04621

[16] S. C. Wong, A. Gatt, V. Stamatescu, and M. D. McDonnell, "Understanding data augmentation for classification: when to warp?" *CoRR*, vol. abs/1609.08764, 2016. [Online]. Available: http://arxiv.org/abs/1609.08764

[17] S. A. E. V. Fábio Perez, Cristina Vasconcelos, "Data augmentation for skin lesion analysis," *CoRR*, vol. abs/1809.01442, 2018. [Online]. Available: http://arxiv.org/abs/1809.01442

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[19] C. Arsene, "Complex Deep Learning Models for Denoising of Human Heart ECG signals," *arXiv e-prints*, p. arXiv:1908.10417, Aug 2019.

[20] R. Sibechi, O. Booij, N. Baka, and P. Bloem, "Exploiting Temporality for Semi-Supervised Video Segmentation," *arXiv e-prints*, p. arXiv:1908.11309, Aug 2019.

[21] N. L. Baisa, "Robust Online Multi-target Visual Tracking using a HISP Filter with Discriminative Deep Appearance Learning," *arXiv e-prints*, p. arXiv:1908.03945, Aug 2019.

[22] T. H. Park, S. Sharma, and S. D'Amico, "Towards Robust Learning-Based Pose Estimation of Noncooperative Spacecraft," *arXiv e-prints*, p. arXiv:1909.00392, Sep 2019.