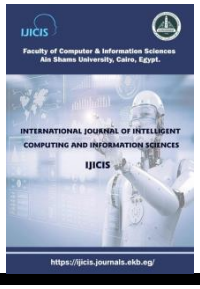
IJICIS, Vol.25, No.3, 30-40
DOI: 10.21608/ijicis.2025.401650.1409

International Journal of Intelligent Computing and Information Sciences

https://ijicis.journals.ekb.eg/



DEEP METRIC LEARNING FOR FEW-SHOT PLANT DISEASES IMAGE CLASSIFICATION

Hosam S. EL-Assiouti*

Scientific Computing,
Faculty of Computer and Information Sciences, Ain Shams University,
Cairo, Egypt
hossamsherif@cis.asu.edu.eg

Hadeer El-Saadawy

Scientific Computing,
Faculty of Computer and Information Sciences, Ain Shams University,
Cairo, Egypt
Hadeer_ibrahim@cis.asu.edu.eg

Maryam N. Al-Berry

Scientific Computing,
Faculty of Computer and Information Sciences, Ain Shams University,
Cairo, Egypt
maryam_nabil@cis.asu.edu.eg

Mahmoud El Gamal

Scientific Computing,
Faculty of Computer and Information Sciences, Ain Shams University,
Cairo, Egypt
mahmoud.elgamal@cis.asu.edu.eg

Received 2025-07-07; Revised 2025-09-01; Accepted 2025-09-02

Abstract: *Image classification is a powerful and widely used technique for distinguishing objects across various benchmarks. However, it suffers from several limitations. First, it fails to recognize or adapt to images of unseen categories, making it unsuitable for real-world applications where new categories frequently emerge during testing. Additionally, traditional classification models assume that the training and testing data are drawn from the same distribution, as is the case with most benchmarks. However, in real-word scenarios, even images from the same categories can be captured under different environmental conditions and challenging settings, making a well-trained classification model ineffective when handling out-of-distribution (OOD) data. Few-shot learning comes into play, where few-shot learning models can adapt to unseen categories and generalize better to OOD data using only a small labeled support set during test. In this paper, we present a resource-efficient deep metric learning network for plant leaf disease recognition in few-shot scenarios, addressing real-world challenges, where new diseases may emerge and field conditions can vary significantly. Specifically, we introduce a lightweight triplet network that leverages efficient embedding backbones. We employ MobileNetV2 and MobileViT-S as our network embedding backbones and optimize the network using the triplet loss. Experiments are conducted on the PlantVillage dataset, where the model is trained on 28 categories and evaluated on 10 unseen categories. Using MobileViT-S as the embedding backbone, our approach achieves a top-1 few-shot classification accuracy of 87.18% on the unseen categories.*

Keywords: *Deep learning, Deep metric learning, Few-shot learning, Lightweight networks, Triplet loss.*

***Corresponding Author**: Hosam S. EL-Assiouti

Scientific Computing Department, Faculty of Computer and Information Science, Ain Shams University, Cairo, Egypt

Email address: hossamsherif@cis.asu.edu.eg

1. Introduction

Plants and crop diseases are one of the key concerns for food security, as crops are considered the main source of food for the world population [1], [2]. Identifying plant diseases at an early stage is essential as it can reduce its harmful effects. Monitoring plant diseases manually can be time-consuming, difficult and error-prune. Deep learning and computer vision have been widely useful in automatically detecting plant diseases, especially in its early stages [3], [4].

Image classification models have demonstrated powerful performance in distinguishing between different categories. However, Image classification comes with various limitations. Image classification models are typically trained with Softmax output layer where the neurons in the output layer correspond to a fixed set of categories. During inference, the model assigns the input image to the category corresponding to the output neuron with the highest probability. This limits the classification model's ability to recognize or adapt to images of unseen categories, making it unsuitable for tasks where new categories frequently emerge during testing, such as person re-identification, retail product recognition and signature verification. Secondly, in most datasets, the training and testing sets are drawn from the same distribution, so a well-trained classification model can achieve very high accuracy on the test set but fails to generalize on out-of-distribution (OOD) data, which is a very common challenge in real-world scenarios.

Motivated by this, Few-Shot learning (FSL) [5] has been developed to address classification tasks with only few labeled examples. It typically aims to mimic the human ability to recognize new objects from just one or a few labeled instances. Various approaches have been proposed to address few-shot recognition tasks, most notably deep metric learning [6], [7], [8] and meta-learning [9], [10].

In plant disease recognition, most available datasets consist of images collected under controlled environments and conditions. For example, PlantVillage dataset [3] is one of the most widely used benchmarks for plant disease classification contains images captured in consistent settings. As a result, training and testing samples are drawn from the same distribution, allowing classification models to achieve high accuracy on the test set but fail to generalize effectively to real-world out-of-distribution agricultural scenarios, where new diseases may emerge and field conditions can vary significantly, and thus motivating the adoption of few-shot learning approaches in our work.

In this work, we introduce a resource-efficient and lightweight deep metric learning approach for few-shot recognition of plant leaf diseases. Specifically, we employ a triplet network that utilizes a lightweight backbone as the embedding model and optimize it using the triplet loss function [6]. We evaluate our approach on the PlantVillage dataset by splitting it into disjoint training and testing sets, with 28 classes used for training and the remaining 10 for testing. Our network employs two lightweight backbone architectures: MobileNetV2 (CNN-based) [11] and MobileViT-S (hybrid) [12]. The performance is assessed under various N-way K-shot settings, with classification accuracy reported using both 1-NN and mean-based strategies, as detailed in the Experiments section. Our approach achieves its best few-shot classification accuracies of 65.02%, 84.01% and 87.18% for the 10-way 1-shot, 10-way 5-shot and 10-way 10-shots settings, respectively, when using MobileViT-S as the embedding backbone along with the 1-NN classification strategy.

In the following sections, we present the structure of the paper. Section 2 provides recent advances in deep metric learning and efficient lightweight backbones and how they are integrated in our work. The proposed approach is discussed in detail in Section 3. Section 4 discussed the implementation details, the utilized datasets as well as the experiments conducted in our research. Lastly, Section 5 provides a conclusion and outlines the future directions.

2. Related Work

2.1. Deep Metric Learning

Deep metric learning aims to encode similarity between instances by learning a representation function that maps data into a non-linear embedding space, where instances of the same category are mapped close together, while instances of different categories are mapped far apart. Deep metric learning approaches take advantage of deep neural network backbones to construct these representative embeddings from input images. These approaches have shown significant success across a variety of real-world applications including face verification [6], [13], [14], [15], person re-identification [16], [17], [18], fashion item retrieval [19], [20] and image retrieval [7], [21], [22]. A variety of loss functions have been proposed to optimize deep metric learning models [23], [6], [24], [25], [18], [26].

2.2. Efficient Lightweight Backbones

Building resource-efficient and lightweight deep neural networks is essential for practical real-world applications, particularly for deployment on embedded and edge devices. Lightweight backbones have been employed in different deep learning tasks including image classification, object detection, image segmentation, and few-shot learning. Efficient convolution neural networks (CNNs) have been focusing not only on achieving high performance on benchmark datasets, but also on developing reliable lightweight models with reduced computational complexity [11], [27], [28], [29], [30], [31]. Meanwhile, Vision Transformers (ViTs) have been effective at capturing long-range dependencies and acquiring global contextual representation compared to CNNs. However, they typically require higher computational cost due to the quadratic complexity associated with the self-attention mechanism. Thus, various work focused on developing efficient lightweight ViTs [32], [33]. Another research direction focused on developing powerful lightweight hybrid models that comprises the benefits of both CNNs and ViTs while being resource efficient [12], [34], [35], [36], [37].

In our work, we strive to optimally combine a powerful deep metric learning approach with an efficient light-weight backbone as an embedding model. Thus, we employ triplet loss as our deep metric learning objective, with different resource-efficient backbones including MobileNetV2 and MobileViT-S.

3. Methodology

In this section, we introduce the overall methodology of our proposed approach. We begin first by giving a brief illustration of the used lightweight backbones, we then revisit the triplet loss formulation, which serves as the foundation of our deep metric learning framework. Then, we describe our proposed lightweight triplet network.

3.1. Lightweight Backbones

3.1.1. MobileNetV2 Backbone

MobileNets [11], [29], [30] are lightweight convolutional neural networks that are designed for resource-constrained and edge devices. They are built primarily based on depthwise separable convolution, which notably reduce computational complexity compared to standard convolutions. The MobileNetV2 variant further introduces inverted residual blocks with linear bottlenecks, making it one of the most widely adopted lightweight CNN backbones across various downstream applications such as image classification, image segmentation, object detection and few-shot learning.

3.1.2. MobileViT Backbone

MobileViT, proposed by Mehta et al. [12], aims to effectively combine the advantages of convolutional neural networks (CNNs) and Vision Transformers (ViTs) to design a lightweight and resource-efficient backbone for mobile and edge vision tasks. Since CNNs are powerful due to their spatial built-in inductive biases. However, they are spatially local and lack the global processing capabilities present in ViTs due to their limited receptive field. On the other hand, ViTs are powerful in learning global representation, but are typically considered heavyweight compared to CNNs. From this pursuit, MobileViT introduces a novel MobileViT block that integrates convolution with transformers to effectively model local and global context. MobileViT models the local information using the MobileNetV2 (MV2) block and models the global information using the MobileViT block which substitutes local processing in convolutions with global processing using the multi-headed self-attention (MHSA) operation [38]. Although MHSA operations generally have quadratic complexity and are considered a key source of computational bottleneck in ViTs. MobileViT applies MHSA only to low-resolution feature maps, significantly reducing computational costs and maintaining lightweight design.

3.2. Triplet Loss

The concept of triplet loss was originally presented by [6]. It is one of the most employed loss functions in deep metric learning. Triplet loss constructs triplets consisting of three instances, an anchor $(\boldsymbol{x})$, a positive $(\boldsymbol{x^+})$ and a negative $(\boldsymbol{x^-})$. The anchor and positive are sampled from the same class (i.e., share the same label) while the negative is sampled from a different class. Each element in the triplet is fed to a deep neural network to be transformed from the input space to the embedding space, forming $\boldsymbol{f(x), f(x^+)\ and\ f(x^-)}$ corresponding to the anchor, positive and negative images, respectively. The triplet loss objective is to decrease the distance between the anchor and positive samples while increasing the distance between the anchor and negative samples. Specifically, it enforces the distance between the feature embeddings of the anchor and positive to be smaller than the distance between the feature embeddings of the anchor and negative with a fixed margin $\boldsymbol{\alpha}$. The triplet loss is formulated as follows:

$$\mathcal{L} = \sum_{x, x^+, x^- \in T} max(\|f(x) - f(x^+)\|_2^2 - \|f(x) - f(x^-)\|_2^2 + \alpha, 0) \qquad (1)$$

$\boldsymbol{\alpha}$ is a fixed violate margin that determines the minimum distance to be enforced between the anchor-positive and anchor-negative pairs. $\boldsymbol{T}$ denotes a mini-batch of triplets composing $\{\ \boldsymbol{x}, \boldsymbol{x^+}, \boldsymbol{x^-}\}$. The term $\|\ .\ \|_2^2$ refers to the squared Euclidian distance in the embedding space.

3.3. Proposed Approach

Our proposed approach employs the triplet network [6]. The triplet network processes three images as input, an anchor image, a positive image, and a negative image. The anchor and positive images are taken randomly from the same category while the negative is taken from a different category within the training set. Consequently, the three images are fed to a shared embedding network which projects the given images from the input image space to a low-dimensional embedding space. In this space, images from the same category should have embeddings that are close together, whereas images belonging to different categories should be positioned far apart. We adopt the triplet loss function for optimizing our network, which is clearly discussed in the previous section. Prior to computing the triplet loss, the three output embedding vectors are L2-normalized to have unit norm. We utilize lightweight and efficient backbone architectures pre-trained on the ImageNet dataset [39] for building a resource-efficient triplet network. Our proposed lightweight triplet network is given in Figure 1.

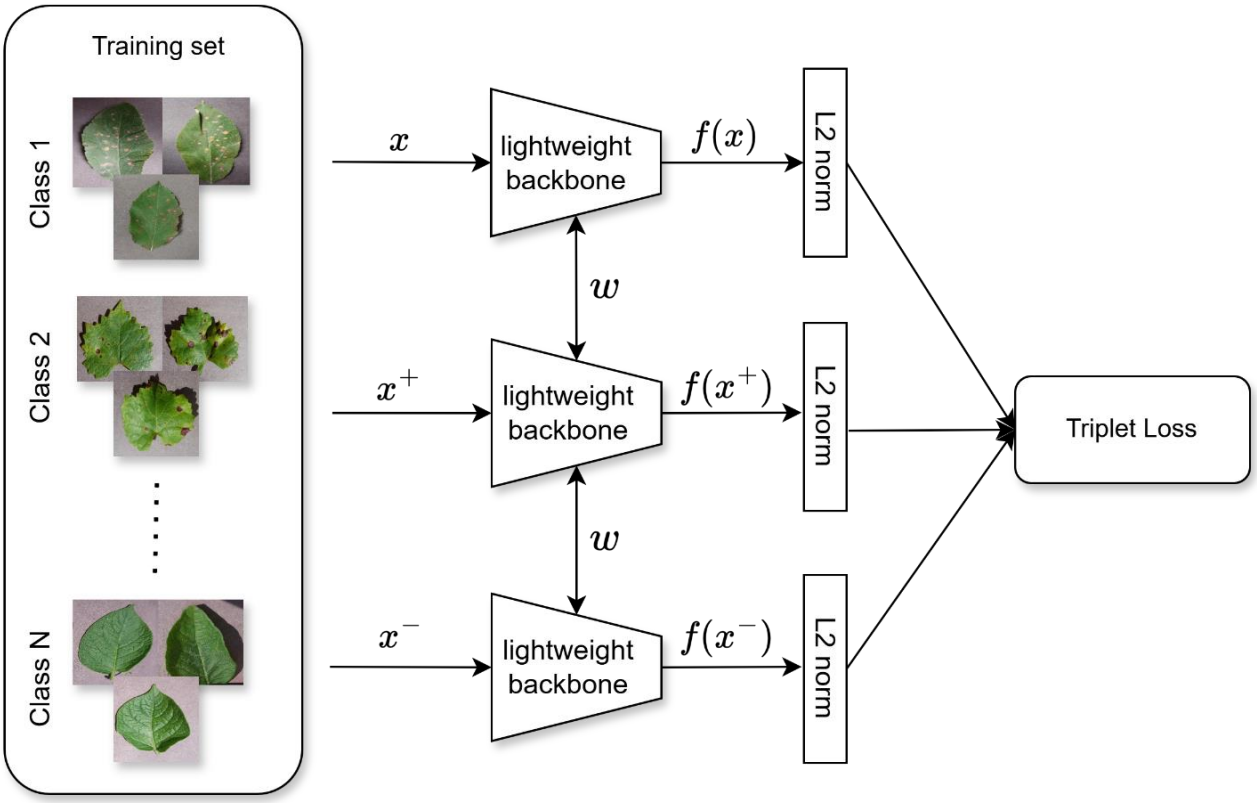


Figure. 1: The architecture of our proposed lightweight triplet network. The weights of the lightweight backbone are shared across all three branches, as depicted in the figure. During the training phase, triplets are formed only from a subset of classes (denoted as N in the figure), while the remaining M classes are left for testing purposes to assess the model's ability to generalize to unseen categories.

4. Experiments and Results

4.1 Implementation Details

Our triplet networks are trained on input images resized to 224×224 resolution over 25 epochs, with a batch size of 32 on the Plant Village dataset. Training is performed with PyTorch on an NVIDIA P100 GPU. We use the Adam optimizer [40] with an initial learning rate of 1e-5 and a StepLR scheduler

which decreases the learning rate by a factor of 0.1 every 8 epochs. In Eq 1., the margin $\boldsymbol{\alpha}$ is set to 1 in our experiments.

4.2. Dataset

The Plant Village dataset [3] is one of the largest and most widely employed datasets for plant disease classification. It comprises 54,303 labeled images of both healthy and diseased plant leaves, belonging to 38 distinct classes, representing diseases across 14 different crop species. We split the dataset into a source domain and a target domain, ensuring that there is no overlap in categories between them. Specifically, the embedding models are trained on 28 categories, while the remaining 10 categories are reserved for testing purposes to assess the network's generalization ability on unseen classes, as provided in Table 1 and Table 2. To maintain a balanced formulation of triplets across training categories, we generate the training triplets by randomly sampling 500 anchor-positive pairs from each class. For each of these pairs, a negative sample is randomly selected from a different category, resulting in a total number of 14000 training triplets. A sample of the constructed triplets from the training categories is depicted in Figure 2.

Figure. 2: Example of the triplets constructed from the training categories

Table 1: The list of categories used for training the proposed triplet network

	Plant Type	Category
1	Apple	Healthy
2	Apple	Cedar apple rust
3	Apple	Black rot
4	Grape	Healthy
5	Grape	Leaf blight
6	Grape	Esca (black measles)
7	Grape	Black rot
8	Corn	Healthy
9	Corn	Cerocospora leaf spot
10	Corn	Northen leaf blight
11	Corn	Common rust
12	Peach	Healthy
13	Peach	Bacterial Spot
14	Strawberry	Healthy
15	Strawberry	Leaf scorch
16	Cherry	Healthy
17	Cherry	Powdery mildew
18	Potato	Healthy
19	Potato	Early blight
20	Potato	Late blight
21	Tomato	Healthy
22	Tomato	Yellow leaf curl virus
23	Tomato	Early blight
24	Tomato	Late blight
25	Tomato	Mosaic virus
26	Tomato	Spider mites
27	Tomato	Target spot
28	Tomato	Leaf mold

Table 2: The List of categories used for evaluating the proposed methodology in our experiments.

	Plant Type	Category
1	Apple	Healthy
2	Blueberry	Healthy
3	Orange	Citrus greening
4	Raspberry	Healthy
5	Soybean	Healthy
6	Squash	Powdery mildew
7	Pepper	Healthy
8	Pepper	Bacterial spot
9	Tomato	Bacterial spot
10	Tomato	Septoria leaf spot

4.3. Experimental Results

As stated in the dataset section, we split our data into training and testing sets, denoted as $\boldsymbol{D_{train}}$ and $\boldsymbol{D_{test}}$, respectively. $\boldsymbol{D_{train}}$ contains categories $\boldsymbol{C_{train}}$, while $\boldsymbol{D_{test}}$ comprises unseen categories $\boldsymbol{C_{test}}$. Where $\boldsymbol{D_{train}}$ and $\boldsymbol{D_{test}}$ are non-overlapping in terms of object categories, such that $\boldsymbol{C_{train}} \cap \boldsymbol{C_{test}} = \emptyset$. The $\boldsymbol{D_{train}}$ contains 28 categories while the $\boldsymbol{D_{test}}$ includes the remaining 10 categories which are stated in Table 1 and Table 2. For few-shot learning evaluation, our approach is assessed under standard N-way K-shot settings. Specifically, we conduct experiments using 10-way 1-shot, 10-way 5-shot, and 10-way 10-shot configurations. In our experiments, we fixed the parameter N to 10 which represents the unseen categories in $\boldsymbol{D_{test}}$. Following common few-shot protocols, we randomly sample K support images from each of the N categories in $\boldsymbol{D_{test}}$, where the remaining images in $\boldsymbol{D_{test}}$ serving as query samples. Consequently, the total support set size becomes 10, 50, and 100 for K values of 1, 5, and 10, respectively. During testing, the model classifies each query image into one of the N novel classes based on feature embeddings learned by the network, by comparing its embedding to those of the support set for all N categories. In Table 3, we report the computational complexity of lightweight backbones employed in our triplet network, in terms of parameters and floating-point operations (FLOPs).

Table 3: A comparison between MobileNetV2 and MobileViT-S in terms of Params and FLOPs. The Params and FLOPs are recomputed on 224×224 input images, excluding the models' fully connected layers.

Backbone network	# Params	FLOPs
MobileNetV2	3.0M	0.3G
MobileViT-S	5.4M	1.4G

We evaluate and compare the performance of our network using two different lightweight embedding backbones: MobileNetV2 and MobileViT-S. These backbones are evaluated under various N-way K-shot configurations, as previously described. For calculating the test classification accuracy, we use two classification strategies based on the support and query embeddings. First, we apply the 1-nearest neighbor (1-NN) approach, where each query image is assigned to the class of its nearest support embedding among the N test classes. Second, we compute the distance between the query embedding and all K support embeddings for every test class, and the query is then assigned to the class whose support set (comprising K samples) yields the lowest mean distance to the query embedding among all N classes. As observed in Table 4, adopting MobileViT-S as the embedding backbone along with the 1-NN as the classification strategy yields to favorable performance across all evaluation settings. The detailed few-shot classification accuracies for both backbones under various N-way K-shot settings and evaluation strategies (1-NN and mean-based) are reported in Table 4.

Table 4: Few-shot classification performance on the PlantVillage dataset. 10-way here represents the 10 unseen classes within the test set.

Backbone	Classification strategy	10-way 1-shot	10-way 5-shot	10-way 10-shot
MobileNetV2	Mean	55.18%	70.88%	73.34%
MobileNetV2	1-NN	55.18%	74.10%	80.92%
MobileViT-S	Mean	65.02%	82.85%	83.96%
MobileViT-S	1-NN	65.02%	84.01%	87.18%

5. Conclusion

In this work, we introduce a computationally efficient deep metric learning network for few-shot recognition, which leverages a lightweight backbone and is optimized using the triplet loss. we experiment our triplet network with both a lightweight CNN-based backbone (MobileNetV2) and a hybrid backbone (MobileViT-S). One of our experimental observations is that using a hybrid embedding backbone outperforms its CNN-based counterpart in deep metric learning approaches, owing to its global context modeling and input-adaptive weighting mechanisms. The proposed approach achieves a top-1 few-shot classification accuracy of 87.18% on the PlantVillage dataset, demonstrating strong generalization performance on unseen test categories. In future, we plan to investigate alternative efficient embedding backbones, explore various hard-mining strategies to enhance the effectiveness of the triplet loss, and even experiment with different loss functions to further improve model optimization and generalization.

References

[1] R. N. Strange and P. R. Scott, “Plant disease: a threat to global food security,” Annu Rev Phytopathol, vol. 43, pp. 83–116, 2005.
[2] J. B. Ristaino et al., “The persistent threat of emerging plant disease pandemics to global food security,” Jun. 2021.
[3] S. P. Mohanty, D. P. Hughes, and M. Salathé, “Using Deep Learning for Image-Based Plant Disease Detection,” Front. Plant Sci., vol. 7, Sep. 2016.
[4] L. Li, S. Zhang, and B. Wang, “Plant Disease Detection and Classification by Deep Learning—A Review,” IEEE Access, vol. 9, pp. 56683–56698, 2021.
[5] J. Snell, K. Swersky, and R. Zemel, “Prototypical Networks for Few-shot Learning,” in Advances in Neural Information Processing Systems, 2017.
[6] F. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: A unified embedding for face recognition and clustering,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2015.
[7] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese, “Deep Metric Learning via Lifted Structured Feature Embedding,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2016.
[8] X. Li, X. Yang, Z. Ma, and J.-H. Xue, “Deep metric learning for few-shot image classification: A Review of recent developments,” Pattern Recognition, vol. 138, p. 109381, Jun. 2023.
[9] C. Finn, P. Abbeel, and S. Levine, “Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks,” in Proceedings of the 34th International Conference on Machine Learning, PMLR, Jul. 2017.
[10] Y. Chen, Z. Liu, H. Xu, T. Darrell, and X. Wang, “Meta-Baseline: Exploring Simple Meta-Learning for Few-Shot Learning,” in 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Oct. 2021.
[11] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Jun. 2018.
[12] S. Mehta and M. Rastegari, “MobileViT: Light-weight, General-purpose, and Mobile-friendly Vision Transformer,” presented at the International Conference on Learning Representations, Oct. 2021.
[13] J. Hu, J. Lu, and Y.-P. Tan, “Discriminative Deep Metric Learning for Face Verification in the Wild,” in 2014 IEEE Conference on Computer Vision and Pattern Recognition, Jun. 2014.

[14] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.
[15] H. Wang et al., "CosFace: Large Margin Cosine Loss for Deep Face Recognition," in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Jun. 2018.
[16] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Deep Metric Learning for Person Re-identification," in 2014 22nd International Conference on Pattern Recognition, Aug. 2014.
[17] F. Wang, W. Zuo, L. Lin, D. Zhang, and L. Zhang, "Joint Learning of Single-Image and Cross-Image Representations for Person Re-identification," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2016.
[18] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond Triplet Loss: A Deep Quadruplet Network for Person Re-identification," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jul. 2017.
[19] S. Ibrahimi, N. Van Noord, Z. Geradts, and M. Worring, "Deep Metric Learning for Cross-Domain Fashion Instance Retrieval," in 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Oct. 2019.
[20] X. Zhao, H. Qi, R. Luo, and L. Davis, "A Weakly Supervised Adaptive Triplet Loss for Deep Metric Learning," in 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Oct. 2019.
[21] M. Opitz, G. Waltner, H. Possegger, and H. Bischof, "BIER — Boosting Independent Embeddings Robustly," in 2017 IEEE International Conference on Computer Vision (ICCV), Oct. 2017.
[22] M. T. Law, R. Urtasun, and R. S. Zemel, "Deep Spectral Clustering Learning," in Proceedings of the 34th International Conference on Machine Learning, PMLR, Jul. 2017.
[23] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality Reduction by Learning an Invariant Mapping," in 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), Jun. 2006.
[24] W. Ge, W. Huang, D. Dong, and M. R. Scott, "Deep Metric Learning with Hierarchical Triplet Loss," in Computer Vision – ECCV 2018, vol. 11210, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., in Lecture Notes in Computer Science, vol. 11210. , Cham: Springer International Publishing, 2018.
[25] K. Sohn, "Improved Deep Metric Learning with Multi-class N-pair Loss Objective," in Advances in Neural Information Processing Systems, Curran Associates, Inc., 2016.
[26] Y. Sun et al., "Circle Loss: A Unified Perspective of Pair Similarity Optimization," in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2020.
[27] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Jun. 2018.
[28] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design," in Computer Vision – ECCV 2018, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., Cham: Springer International Publishing, 2018.
[29] A. G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv.org. Accessed: Apr. 01, 2024. [Online]. Available: https://arxiv.org/abs/1704.04861v1
[30] A. Howard et al., "Searching for MobileNetV3," in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Oct. 2019.

[31] M. Tan and Q. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” in Proceedings of the 36th International Conference on Machine Learning, PMLR, May 2019.
[32] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou, “Training data-efficient image transformers & distillation through attention,” in Proceedings of the 38th International Conference on Machine Learning, PMLR, Jul. 2021.
[33] K. Wu et al., “TinyViT: Fast Pretraining Distillation for Small Vision Transformers,” in Computer Vision – ECCV 2022, vol. 13681, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds., in Lecture Notes in Computer Science, vol. 13681. , Cham: Springer Nature Switzerland, 2022.
[34] Y. Chen et al., “Mobile-Former: Bridging MobileNet and Transformer,” in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2022.
[35] S. Mehta and M. Rastegari, “Separable Self-attention for Mobile Vision Transformers,” Transactions on Machine Learning Research, Sep. 2022.
[36] Y. Li et al., “EfficientFormer: Vision Transformers at MobileNet Speed,” Advances in Neural Information Processing Systems, Dec. 2022.
[37] A. Hatamizadeh et al., “FASTERVIT: FAST VISION TRANSFORMERS WITH HIERARCHICAL ATTENTION,” 2024.
[38] A. Vaswani et al., “Attention is All you Need,” in Advances in Neural Information Processing Systems, 2017.
[39] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in 2009 IEEE Conference on Computer Vision and Pattern Recognition, Jun. 2009.
[40] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” Jan. 30, 2017, arXiv: arXiv:1412.6980.