Dual-Tuning: Joint Prototype Transfer and Structure Regularization for Compatible Feature Learning

Yan Bai 🇨🇳, Jile Jiao 🇨🇳, Yihang Lou 🇨🇳, Shengsen Wu, Jun Liu 🇨🇳, Member, IEEE, Xuetao Feng, and Ling-Yu Duan 🇨🇳

Abstract—Visual retrieval system faces frequent model update and deployment. It is a heavy workload to re-extract features of the whole database every time. Feature compatibility enables the learned new visual features to be directly compared with the old features stored in the database. In this way, when updating the deployed model, we can bypass the inflexible and time-consuming feature re-extraction process. However, the old feature space that needs to be compatible is not ideal and faces outlier samples. Besides, the new and old models may be supervised by different losses, which will further cause distribution discrepancy problem between these two feature spaces. In this article, we propose a global optimization Dual-Tuning method to obtain feature compatibility against different networks and losses. A feature-level prototype loss is proposed to explicitly align two types of embedding features, by transferring global prototype information. Furthermore, we design a component-level mutual structural regularization to implicitly optimize the feature intrinsic structure. Experiments are conducted on six datasets, including person ReID datasets, face recognition datasets, and million-scale ImageNet and Place365. Experimental results demonstrate that our Dual-Tuning is able to obtain feature compatibility without sacrificing performance.

Index Terms—Compatible feature learning, prototype transfer, structure regularization.

I. INTRODUCTION

Visual matching and retrieval systems are widely utilized in many scenarios, such as image retrieval [1], [2], person re-identification (ReID) [3], [4], [5], [6], vehicle re-identification [7], [8], [9], and face recognition [10], [11]. Most visual systems use deep learning models to map images into an embedding space. In this space, features of similar samples are close to each other, and tend to form a cluster when they share the same class. Commonly, the embedding features of a large-scale image collection are extracted in advance, and referred to as the gallery set or database. Then, the retrieval is accomplished by ranking all gallery features according to their distances to the input query feature.

To achieve higher performance, the models deployed in practical industrial systems often need to be updated from time to time, due to extended training data, improved network architectures or loss functions. Once the model is updated, to ensure feature consistency, the entire gallery set needs to be re-extracted. It is a time and memory consuming process because millions or even billions of gallery features have to be re-generated for a large-scale retrieval system. Besides, the computing power is very limited for real-time surveillance analysis. For example, in a practical ReID system, all the GPU computing power needs to support visual analysis on massive real-time video streams. There is even no spare computing power for such occasional large-scale history feature re-extraction. Moreover, the original gallery images may not be available due to privacy issue. Therefore, in this paper, we focus on feature compatible learning, which enables the feature extracted by the newly learned model to be directly compared to that of an old model, without sacrificing performance.

Then the old gallery feature set can still work smoothly with the new queries, bypassing the repetitive and painful feature re-extraction process, as shown in Fig. 1.

To achieve feature compatibility, a straightforward solution is to constrain the representations produced by the new and old...
models to be identical or similar. Budnik et al. [12] introduced an asymmetric metric learning method for knowledge transfer task. It projects the features of the new model into the old embedding space, and uses metric loss, such as triplet loss [13], to optimize the distances between new and old features. Besides the feature-level optimization, Shen et al. proposed BCT [14], a backward compatible training framework for feature compatible learning, which uses the classifier component of the old model to supervise the training process of the new model.

For an ideal compatible model, its feature should be compatible with the existing old features and meanwhile get better discrimination ability from updated data or network designs. The compatible learning faces three challenges and the existing methods can not cope with them well. 1) The old model’s embedding space is usually not ideal and need to be updated by the new model. It usually exists some sensitive limitations, such as outlier and noise. Therefore, the behavior of imitating the instance-level “bad” old features [12] has a potential risk of damaging the new model’s ability to achieve stronger discrimination power. 2) There may exist a severe distribution discrepancy between new and old feature spaces, which is caused by different classifier types or supervision losses between new and old models. Therefore, only using the old classifier to achieve compatibility [14] faces severe performance drops. And the instance-level optimization [12] that depends on few samples is also insufficient and cannot cover the whole distribution structure. 3) In the database, there exist the continuously added new model’s features and the existing old model’s features. The new and old features of different classes should not overlap in the embedding space. However, all the existing methods ignore the optimization for the mixed gallery scenario. To solve these three challenges, a better optimization scheme that can achieve global optimization between the new and old embedding spaces is necessary.

Such global information can be obtained from both the embedding space and classifier’s hypothesis space [15]. For dual tuning in these two spaces, the feature-level optimization can explicitly achieve compatibility by embedding feature alignment, and the classifier’s supervision can also implicitly constrain the feature intrinsic structure to obtain compatible features. Therefore, it is necessary and effective to use these two types of information simultaneously, which has also been demonstrated in other fields [15], [16], such as the widely used triplet loss + softmax loss optimization scheme in ReID.

In this paper, we propose a compatible feature learning method called Dual-Tuning that optimizes both embedding space and classifier space. It includes a feature-level compatible prototype loss for embedding space and a component-level mutual structural regularization for classifier space. In the embedding space, the features belonging to the same class tend to be clustered together [16], [17]. This inspires us to represent each class as a prototype. Using a prototype to describe each class can alleviate the limitations of outlier and distribution discrepancy in each class. Then the whole manifold structure can be described by all prototypes. We transfer such prototype knowledge from old embedding space to the new one. Thereby the new features could closely surround the old prototypes to achieve compatibility while also obtaining a more compact and discriminative space. Besides, to simulate the mixed gallery scenario where both new and old features exist, we further introduce a memory bank to calculate the prototypes in the new embedding space. By utilizing both the new and old prototypes, the feature compatibility and discrimination power can be further boosted.

Besides the feature-level optimization, a component-level mutual structural regularization for classifier space is also proposed. Intuitively, the classifier provides the “rules” of feature intrinsic structure. If features derived from two models are compatible and can mutually match, the features from one model can also satisfy the other model’s rules. Therefore, a component interoperation scheme is designed for mutual structural regularization. We recombine the embedding backbone of one model and the classifier from the other model. In this way, the structural information in old classifier can be used to supervise new embedding learning. Simultaneously, the old embedding module can also assist in formulating the structural rules of the new classifier.

Overall, our contributions can be summarized as follows:

- We propose a Dual-Tuning method, which explicitly exploits the global prototype to align features in embedding space and implicitly utilizes the intrinsic structure information encoded in classifier. In this way, the feature compatibility can be globally well achieved from two complementary spaces.
- We propose a prototype-based compatible loss that enforces the learned new feature to be distributed around the old prototypes. Such prototypes describe the distribution of the whole embedding space, which can bridge and align the new and old spaces for optimization under a global view.
- A novel mutual structural regularization is further proposed to use both the old embedding module and old classifier head to supervise the new model training. The discrimination “rules” encoded in the new and old classifiers are utilized to give an additional implicit view of feature distribution in optimization.

Dual-Tuning is robust against the changing factors in new model, such as network architectures, loss functions, feature dimensions, and training data. The cross-test performance between new and old features can even surpass only using the old one, without sacrificing its new feature self-test performance on million-scale datasets.

II. RELATED WORK

A. Feature Compatible Learning

Before feature compatible learning, previous works [18], [19] aimed to learn a separate mapping model to translate features from a source model to a target model. However, an additional feature re-representation process is also needed before a feature can be compared with the stored gallery features. Due to the increasing size of gallery sets and the heavy workload of re-extracting gallery features, feature compatible learning without the need of such a redundant step was proposed.

Recently, Shen et al. [14] proposed BCT, a backward-compatible training framework with a regularization function
by using the old classifier in training the new embedding model, without additional feature re-generation process. Moreover, CMP-NAS [20] expanded beyond BCT to neural architecture search [21], which can automatically obtain a smaller new model and achieve compatibility with the original old model. However, BCT faces several limitations. When the classifiers/loss functions of new and old models are different, BCT faces severe performance drop. If the old model is without a classifier (e.g., supervised by triplet), BCT even can not work. Besides, BCT is sensitive to non-ideal old model, and it sacrifices new model’s performance to achieve compatibility in their experiments [14]. Differently, our Dual-Tuning can explicitly align embedding features by transferring the prototype information, and implicitly constrain the feature intrinsic structure in a bi-directional manner. In this way, we can better achieve feature compatibility even the old embedding space is not ideal and has distribution discrepancy with the new space.

B. Domain Adaptation

Domain adaptation [22], [23] aims to learn a model for the unlabeled target domain by leveraging knowledge from a labeled source domain. A group of methods uses various metrics to mitigate the distribution difference caused by domain gap, such as maximum mean discrepancy (MMD) [24] and its variants [25], [26]. Another typical line aims to design a feature space such that confusion between the source and target distributions in that space is maximal [27], [28], [29]. These methods can be used to align the (marginal) distribution of the new and old classes. The problem we focus on differs in that we aim to achieve feature-level compatibility between any feature pair of new and old models.

C. Knowledge Distillation

Knowledge distillation aims to guide the learning of a small network using the knowledge from a large network. Hinton et al. [30] proposed using the probabilistic outputs of a larger network as soft targets to supervise the smaller network, and KL Divergence loss is adopted. Besides, some variants were proposed to mimic the intermediate feature maps [31], attention maps [32], or second-order statistics (Gram matrix) [33] from the teacher network. However, knowledge distillation is not applicable in compatible learning. Since the existing teacher model (old model) is unsatisfactory, mimicking a “weak” teacher will affect new models to get better discrimination capability.

D. Incremental Learning

Incremental learning [34] aims to obtain the ability to continuously learn and adapt the models to new tasks without losing the already acquired knowledge. The directions include replaying samples [35], [36], adding extra regularization term to consolidate previous knowledge [37], [38], and introducing different model parameters to each task [39], [40]. Unlike incremental learning trying to maintain the performance on old classes after introducing new classes, feature compatible learning focuses on enabling the feature matching between new and old models.

III. PROPOSED METHOD

In this work, we focus on feature compatible learning, which requires the learned features of a new (updated) model to be directly compared with features of its old version. Considering it is effective to use embedding and classification spaces simultaneously [15], [16], we hope to achieve dual-tuning in these two spaces. A novel compatible prototype loss is designed to explicitly constrain feature distance in embedding space. And a mutual structural regularization is proposed to further achieve compatibility implicitly in classifier space.

A. Problem Formulation

There are two models in feature compatible learning, an old model with fixed parameters, and a new model whose feature is compatible with the old feature. Typically, a network model can be split into two components, an embedding backbone \( \phi(\cdot; \omega_\phi) \) and a task head \( h(\cdot; \omega_h) \), where \( \{\omega_\phi, \omega_h\} \) are the network parameters. For simplicity, those two components are also denoted as \( \phi(\cdot) \) and \( h(\cdot) \). In visual retrieval, the classification network is widely used to obtain the embedding feature [16]. The embedding module \( \phi: \mathcal{X} \rightarrow \mathcal{F} \) maps an input \( x \in \mathcal{X} \) to an embedding space \( \mathcal{F} \), and the head module \( h: \mathcal{F} \rightarrow \mathcal{P} \) further projects the features to the classifier’s hypothesis space \( \mathcal{P} \). Then the category probabilities can be obtained by \( p = h(\phi(x; \omega_\phi); \omega_h) \), as shown in Fig. 2(a). In feature compatible learning, the old model can be represented as \( \phi_O \) and \( h_O \) trained on the old training set \( \mathcal{T}_O \). As model structure evolves or data increases, a new model with \( \phi_N \) and \( h_N \) will be generated based on the new training set \( \mathcal{T}_N \).

After model updating, some of the gallery features stored in the database are still derived from the old model, while the queries are extracted from the new model. The compatible feature learning makes it possible to conduct retrieval between the new query features and old gallery features. The expected compatible features derived from the new model should satisfy two criteria. First, it should be compatible with the old features. Second, compatible learning should not affect the new model optimization to get better performance from updated data or network designs.

The distance relationship in the compatible embedding space can be described as,

\[
d(\phi_{N}(a), \phi_{o}(p)) < d(\phi_{N}(a), \phi_{o}(n)), \phi_{o} \in \{\phi_{O}, \phi_{O}'\}, \\
\forall (a, p, n) \in \{(a, p, n) | y_a = y_p \text{ and } y_a \neq y_n\},
\]

(1)

where \( o \) denotes the query sample, and \( p, n \) denote positive and negative galleries. \( y \) is the class label, and \( d(\cdot) \) is the Euclidean distance. This formulation can be described as, no matter if the gallery features are from the old or new embedding space, the distance between the positive pair \((a, p)\) should be consistently smaller than the negative pair \((a, n)\). Note that, (1) optimizes the mixed gallery situation where both new and old features exist in the gallery, which has been ignored in [12], [14].
B. Compatible Prototype Loss

1) Center-Based Prototype Loss: To enable feature compatibility, we design a center-based prototype loss to explicitly constrain feature distance in the embedding space, as shown in Fig. 2(b). To achieve a global optimization, we use prototypes to represent the manifold of the old embedding space. By transferring such manifold knowledge, the new embedding space can be embedded into the old one to achieve feature alignment explicitly. Note that, given any sample, we can extract its old model’s embedding feature by \( \phi_O \), even the given sample is not in the old training set. Therefore, for each sample \( x \) in the new training set \( \mathcal{T}_N \), we can calculate its fixed old embedding feature \( \phi_O(x) \). Then the prototype for each class can be represented by its old feature centers as,

\[
\phi_O(x) = \frac{1}{|\mathcal{P}(c)|} \sum_{x \in \mathcal{P}(c)} \phi_O(x),
\]

(2)

where \( \mathcal{P}(c) \) is the sample set of class \( c \), and \( |\mathcal{P}(c)| \) is the sample number of \( \mathcal{P}(c) \). In this way, given a newly added class not contained in the old training set, we can also calculate its old feature center by \( \phi_O \). For \( C \) classes in new training set \( \mathcal{T}_N \), we can obtain \( C \) centers \( \mathcal{M}_O = \{\phi_O(x)\}_{c=1}^C \).

These centers are robust for describing the whole embedding space. We did a toy experiment to validate it. A ReID model on Market1501 has only 61.49% mAP, yet replacing query as its class center can get 84.52% mAP. Such significant improvement indicates that by constraining new features surrounding its old center, even the old database/gallery does not update, impressive results can be obtained.

After obtaining the old prototypes, a straightforward solution is to constrain the distance between one old prototype and one new feature in each optimization. However, such pair-wise local optimization is inefficient and cannot cover the whole structure of the embedding space. Considering the old model features are fixed and will not change during the new model’s training, we aim to use all of the old prototypes to support a global optimization.

To achieve this, we design a prototype-based prediction strategy that optimizes the distances between a new feature and all old prototypes. Inspired by the concept of prototype [41], we design a similarity-based prototype prediction loss for compatible learning. Such prediction loss will not incorporate an extra classifier and can achieve an explicit distance optimization in embedding space. Concretely, we select a query \( \phi_N(x) \) with new feature and all prototypes \( \mathcal{M}_O \) from old features, then compute the cosine similarities between them. Based on the similarity scores, we aim to predict which prototype/class the query belongs to. Thus, the constraint of prototype loss \( L_{\text{proto}} \) can be formulated as,

\[
L_{\text{proto}}(\phi_N; \mathcal{M}_O, \tau_N) = -\log \frac{\exp(\langle \phi_N(x)^c, m_O^c \rangle)}{\sum_{c'=1}^{\mathcal{C}} \exp(\langle \phi_N(x), m_O^{c'} \rangle)},
\]

(3)

where \( \langle \cdot, \cdot \rangle \) denotes the cosine similarity. \( x \in \tau_N \) belongs to the same class \( c \) as \( m_O^c \). Prototype loss is able to maximize the similarity between \( \phi_N(x) \) and \( m_O^c \), meanwhile minimize the similarities to all other old prototypes (i.e., towards 0). It is worth mentioning that the proposed \( L_{\text{proto}} \) shows the following major advantages.

a) Different from Softtriple loss [42], [43] or softmax loss, the prototype is neither a learnable weight nor from a mini-batch. It is a fixed feature center calculated by all the old features, which can bridge the new and old embedding space, and well support the global embedding optimization.
and old embedding backbone can be \( L \) and \( C, \omega \), \( \tau \) (7) \( \{ M \} \). i.e. in (1). How-

\( h < d (M - \text{exp}m M) \) and \( d L \) denotes the samples from class \( M \) and \( \text{center}

b) The old embedding space may have outliers. Using a proto-
type to represent a class can bypass the instance level noise
limitations. Besides, the new and old spaces face distribution
discrepancy caused by different supervision losses. For exam-
ple, metric loss tends to produce a lumpy distribution, while
classification loss usually has a ray distribution. Using the
prototype rather than an instance-level feature or a classifier can
alleviate this limitation. As shown in Fig. 3, if the old model
and new model are supervised by classification loss and triplet
loss separately, the new features will closely surround the old
prototypes to achieve compatibility while also pursuing a better
representation.

c) The compatible prototype loss is not limited to the assump-
tion in previous work [14] that requires the old and new training
data to have overlap. It can work well with only the new dataset
\( \tau N \) and old embedding backbone \( \phi O \).

2) Memory-Based Prototype Loss: Center-based prototype
loss simulates the scene where the new query matches the old
gallery, i.e., \( d(\phi N(a), \phi O(p)) < d(\phi O(a), \phi O(n)) \) in (1).
However, besides the existing old features, the new features will
also be continuously added into the database, after obtaining
the new model. Therefore, \( d(\phi N(a), \phi N(p)) < d(\phi O(a), \phi O(n)) \)
and \( d(\phi N(a), \phi O(p)) < d(\phi N(n), \phi O(n)) \) in (1) should also be
satisfied.

To achieve this, we further design a memory-based prototype
loss in the new embedding space. Since the new model is con-
tinuously updated during training, we cannot compute the whole
dataset features to obtain the centers in each iteration. There-
fore, we use a dynamic memory bank [44], [45], [46] to store
the embedding features. The memory bank is implemented by a
queue, in which the current mini-batch enqueued and the oldest
mini-batch dequeued. The queue can decouple the calculated
embedding number from the mini-batch size. Then we design a
discrete dictionary to store the class ID and corresponding pro-
totypes. For the \( C \) classes in the current queue, the memory \( M N \)
has \( C \) key-value pairs to store the class ID \( c \) and corresponding
prototypes \( m c \). At each iteration, we calculate the prototypes
with the features in the current queue as,

\[
m c = \frac{1}{|Q(c)|} \sum_{x \in Q(c)} \phi N(x), \quad (4)
\]

where \( Q(c) \) denotes the samples from class \( c \) in current queue. By considering both the new and old prototypes, we can simulate
the mixed gallery situations. Each class has a new and an old
prototype, we randomly select one from them, i.e., \( m \) or \( m c \), and \( M = \{ M O, M N \} \). Then, updating Eq. 3, the final
compatible prototype loss can be represented as,

\[
\mathcal{L}_{\text{proto}}(\phi N; M, \tau N) = -\log \frac{\exp(\langle \phi N(x c), m c \rangle)}{\sum_{c=1}^{C} \exp(\langle \phi N(x c), m c \rangle)}. \quad (5)
\]

C. Mutual Structural Regularization

Besides the feature-level optimization, we also design a
component-level mutual structural regularization to further im-
prove the compatibility, as shown in Fig. 2(c). Given a network,
it can be split into two components, i.e., an embedding module
for feature extraction and a head for target task. Such a head
can be treated as the “rules,” which contains the intrinsic structure of
embedding space. Take a classifier head as an example, classifier
head formulates classification rule, and according to this rule,
the embedding in feature space \( F \) can be mapped to category
probabilities in classifier’s hypothesis space \( P, h w, : F \rightarrow P \).

Therefore, if features from two models can conduct mutual
matching, the features derived from one model can also pro-
duce good predictions when passing through the other model’s
classifier. This inspires us to design a component interopera-
tion for mutual structural regularization. Specifically, the distri-
bution structure of the new features should satisfy the “rules” of
the old classifier, and correspondingly, the new classifier should
also predict correctly based on the old features.

For a classification task, the widely used cross-entropy loss
can be expressed as,

\[
\mathcal{L}_{\text{CE}}(\phi, h; \tau) = \sum_{(x, y) \in \tau} -y \log h(\phi(x; \omega \phi); \omega h), \quad (6)
\]

where \( y \) is the one hot vector label of sample \( x \). For training
a single new or old model, the optimization targets are
\( \mathcal{L}_{\text{CE}}(\phi O, h N; \tau N) \) and \( \mathcal{L}_{\text{CE}}(\phi O, h O; \tau O) \), correspondingly.

The component interoperation means that, for the new and old
models, we can recombine the embedding module of one model
and classifier head of the other model while still maintaining
good prediction performance. Thus, the optimization targets of
mutual structural regularization \( \mathcal{L}_{\text{str}}(\phi N, h N; \tau N, \tau N) \) can be expressed as,

\[
\arg \min_{\omega \phi N, \omega h N} \left( \mathcal{L}_{\text{CE}}(\phi O, h O; \tau N) + \mathcal{L}_{\text{CE}}(\phi O, h N; \tau N) \right). \quad (7)
\]

In \( \mathcal{L}_{\text{str}} \), the modules of old model \( (\phi O, h O) \) are fixed,
and only new model is updated \( (\phi N) \). Different from
BCT [14] which only uses the old classifier head to supervise
the new embedding module, our component interoperation can
provide dual supervision. The old “rules” will guide the training
of the new embedding module. At the same time, the old fea-
tures will also assist in formulating the embedding rules of the
new classifier.
D. Overall Objective

For learning ideal embedding features, works [15], [16] in other fields demonstrate that it is effective to optimize embedding and classification spaces simultaneously (e.g., metric-based triplet loss + classifier-based softmax loss in ReID). Our two schemes are complementary and also perform in these two spaces. The prototype loss performs in the embedding space and can achieve compatibility in any scenario. Besides, if the old head is available, our mutual structural regularization can further provide knowledge in classifier to get more enhanced results. Finally, to simultaneously exploit the prototype knowledge in the embedding space and conduct mutual structural regularization by old components, the final optimization objective for Dual-Tuning is,

$$L_{all} = L_{proto}(\phi; M) + L_{stra}(\phi, h; \tau) + L_{CE}(\phi, h; \tau),$$

where $L_{CE}$ is the supervision loss for the target task, which is not limited to cross-entropy loss and does not need to be the same form as the old model. $L_{proto}$ and $L_{stra}$ are illustrated in Eq. 5 and Eq. 7, respectively.

IV. EXPERIMENTS

A. Experiment Setting

Dataset: To comprehensively evaluate the compatible learning approaches, our experiments are conducted on six widely used dataset, including the person ReID datasets Market1501 [48] and MSMT17 [49], face recognition datasets IMDB-Face [50], IJB-C [51], and million-scale classification datasets ImageNet [52] and Place365 [53].

- Person ReID is a typical retrieval task which can be used to evaluate model compatibility. For Market1501 and MSMT17, the standard person ReID training and testing process is performed [49]. The mean Average Precision (mAP) and Top-K accuracy are used as evaluation metrics.

- For face recognition, IMDB-Face is adopted as training dataset and IJB-C is works as evaluation set. We adopt the two standard testing protocols for face recognition: 1:1 verification and 1:N search. The evaluation metric is TAR (%)@FAR=10^{-4} for verification task, and TNIR(%)@FPR=10^{-2} for retrieval task.

- The comparison method BCT [14] provides its results on ImageNet and Place365 at its official Github. Therefore, we also follow their experiments for a fair comparison. The model is trained on the training set. And during testing, each image in validation set is considered as a query image, and all other images are considered as gallery images, the same setting as in [14]. Top-K accuracy is adopted as protocol.

The dataset information is shown in Table II.

Implementation details: For compatible model learning, we adopt two widely used architectures ResNet18 and ResNet50 [54] as our backbones, whose feature dimensions are 512 and 2048 respectively. For Market1501 and MSMT17, we employ the bag-of-tricks scheme [16] for ReID model training, and the input size is $256 \times 128$. For IMDB-Face [50], IJB-C [51], the images are aligned to $112 \times 112$. For ImageNet and Place365, we adopt the training scheme in [14], and the image input size is $224 \times 224$. For the memory bank, the queue size is set to 4096. The supervision loss ($L_{CE}$ in (8)) for target task is cross-entropy loss. We optimize the model with Adam optimizer and use 1 GPU for network training. The model is trained for 120 epochs with a start learning rate of $3.5 \times 10^{-4}$ and performs learning rate decay $1/10$ in the $40th$ and $70th$ epochs.

Experiment setting: We conduct “Cross-test” and “Self-test” experiments for compatible model evaluation:

- Cross-test: In retrieval, the query and gallery features are extracted from the new and old model respectively, and the Euclidean distances are directly computed between the new and old features. If the dimensions of these two types features are different, zero padding will be used for dimension alignment [14].

- Self-test: In retrieval, the query and gallery features are both extracted from the new (learned) model.

To emulate the practical case that a new model is generated when a new larger training set is available, half (50%) of the classes are used for training an old model, and all (100%) classes are used for training a new model. In our table presentation, the top part shows the performances of the independent models. The bottom part provides the performances of feature compatibility learning.

B. Compatibility Analysis

The results of Market1501, MSMT17, IJB-C, Place365, and ImageNet are shown in Table I, III IV, and V. The “Res18 v.s. Res18” or “Res50 v.s. Res18” in setting (new v.s. old) column means the new model is with ResNet18 / ResNet50 backbone, and the old model is ResNet18. Several comparison methods are adopted, including a naive $L_2$ loss [47], knowledge distillation method KL Divergence [30], metric-based Asymmetric triplet loss [12], and compatible learning method BCT [14].

Independently trained new and old model: We first conduct a simple experiment to test the cross-test performance between two independently trained models, i.e., Independent (new) and Independent (old) in Table I. The cross-test performance between these two independent models is almost 0% accuracy in most experiments, demonstrating its bad compatibility. In addition, the cross-test between Res18-50% and Res18-100% models on Market1501 is 13.55% mAP, since they use the same initialization parameters, which is helpful for feature compatibility.

A naive $L_2$ loss: Given a sample, the $L_2$ loss aims to minimize the Euclidean distance between its new and old features [47]. However, such a straightforward approach is too local and strict for new feature learning. It not only fails to achieve feature compatibility, but also interferes with the training of new models, as shown in Table I and V.

Knowledge distillation - KL Divergence: We also compare with the classic KL Divergence [30] for knowledge distillation, which constrains the probabilistic outputs of the new model by those of the old model. However, the new model (student) should
TABLE I

| Settings (new v.s. old) | Method | Cross-test mAP | Self-test mAP |
|------------------------|--------|----------------|---------------|
|                        | Top-1  | Top-5          | Top-1         |
|                        |        |                |               |
| Res18-50%              | Independent (old) | 28.32 | 55.97 | 70.78 |
| Res18-100%             | Independent (new) | 4.15  | 10.38 | 25.64 |
|                        |        | 42.01          | 68.91         |
|                        |        | 81.89          |               |
|                        |        | 13.55          | 21.35         |
|                        |        | 42.19          | 80.43         |
|                        |        | 92.31          | 97.54         |
| Market1001             |        |                |               |
|                        |        |                |               |
| L2 Loss [47]           | 9.87   | 20.32          | 37.93         |
|                        |        | 36.61          | 63.28         |
|                        |        | 79.43          |               |
| KL Divergence [30]     | 29.34  | 56.31          | 72.47         |
|                        |        | 37.17          | 64.62         |
|                        |        | 78.87          |               |
| Asymmetric Triplet [12] | 26.60 | 54.29          | 69.92         |
|                        |        | 40.74          | 68.32         |
|                        |        | 81.04          |               |
| Self-Tuning (Ours)     | 31.11  | 58.85          | 75.99         |
|                        |        | 42.58          | 69.59         |
|                        |        | 82.76          |               |
| Dual-Tuning (Ours)     | 32.08  | 59.70          | 76.09         |
|                        |        | 42.17          | 70.06         |
|                        |        | 82.83          |               |
|                        |        | 69.53          | 86.67         |
|                        |        | 95.04          | 98.05         |
|                        |        | 70.92          | 97.33         |

TABLE II

| Datasets Information |
|-----------------------|
| Datasets  | Classes | Images |
| Makel501 | 1,501 | 32,886 |
| MSMT17   | 4,101 | 126,411 |
| IMDB-Face | 59K | 1.7 million |
| IJB-C    | 3,513 | 130K |
| ImageNet | 1,000 | 1.2 million |
| Place365 | 365  | 1.8 million |

TABLE III

| Performance Comparison on the IMDB-Face and IJB-C Datasets |
|-----------------------------------------------------------|
| Settings (new v.s. old) | Method | Verifi. | Self-test |
|-------------------------|--------|--------|-----------|
|                        | Top-1  | Top-5  | Top-1     |
|                        |        |        |           |
| Res101-50%             | Independent (old) | -      | -        | 77.86 |
| Res101-100%            | Independent (new) | -      | -        | 86.96 |
|                        |        |        | 76.88     |
| Res18-50%              | KL Divergence [30] | 80.34 | 69.02 | 84.95 |
|                        | BCT [14] | 27.3  | 37.8  | 61.2  |
|                        |        | 62.0   |       |
| Res18-50%              | Asymmetric Triplet [12] | 29.4  | 58.2  | 38.1  |
|                        | BCT [14] | 27.3  | 37.8  | 61.2  |
|                        |        | 62.0   |       |
| Res18-50%              | Dual-Tuning (Ours) | 82.13 | 72.05 | 86.93 |
|                        |        |        | 76.71     |

TABLE IV

| Performance Comparison on the Place365 Dataset |
|------------------------------------------------|
| Settings (new v.s. old) | Method | Cross-test mAP | Self-test mAP |
|-------------------------|--------|----------------|---------------|
|                        | Top-1  | Top-5          | Top-1         |
|                        |        |                |               |
| Res18-50%              | Independent (old) | -      | -        | 39.5  |
| Res18-100%             | Independent (new) | 0.0    | 0.2      | 3.30  |
|                        |        | 62.0            |               |
| Res18-50%              | Asymmetric Triplet [12] | 29.5  | 58.7  | 31.0  |
|                        | BCT [14] | 27.5  | 37.8  | 62.0  |
|                        |        | 64.0            |               |
| Res18-50%              | Dual-Tuning (Ours) | 28.5  | 58.0  | 35.2  |
|                        |        | 64.0            |               |

TABLE V

| Performance Comparison on the ImageNet Dataset |
|------------------------------------------------|
| Settings (new v.s. old) | Method | Cross-test mAP | Self-test mAP |
|-------------------------|--------|----------------|---------------|
|                        | Top-1  | Top-5          | Top-1         |
|                        |        |                |               |
| Res18-50%              | Independent (old) | -      | -        | 39.5  |
| Res18-100%             | Independent (new) | 0.1    | 0.5      | 62.5  |
|                        |        | 81.5            |               |
| Res50-100%             | Asymmetric Triplet [12] | 13.0  | 32.8  | 43.8  |
|                        | BCT [14] | 42.2  | 65.5  | 55.6  |
|                        |        | 76.6            |               |
| Res18-50%              | Dual-Tuning (Ours) | 46.0  | 68.8  | 62.2  |
|                        |        | 81.5            |               |
In Table I, Part-(a) and part-(b) use ResNet18 as old model, but use ResNet18 and ResNet50 as new model, respectively. Part-(b) is more challenging since new and old models have different architectures and feature dimensions. We show more advantage than BCT [14] in more challenging Part-(b). As shown in Table V, Dual-tuning beats BCT by a remarkable margin on the million-scale ImageNet dataset (46.0% v.s. 42.2% for cross-test and 62.2% v.s. 55.6% for self-test). For Place365 and IJB-C, our Dual-tuning also achieves superior performances, as shown in Table III and Table IV.

### C. Analysis on Dual-Tuning

**Ablation study:** Our method contains two parts: a compatible prototype loss ($L_{\text{proto}}$) and a mutual structural regularization ($L_{\text{stru}}$). In Table VI, the Compatible Prototype indicates compatible prototype loss and the Mutual-Reg indicates mutual structural regularization. Only using the compatible prototype loss can get 80.72% mAP on the self-test, and even surpasses the independently trained new model 80.43% mAP (see Table I). This can be attributed to that prototype loss can exploit the prototype-based manifold information in old embedding space, and meanwhile minimize the impact of noise and disordered distribution derived from the old space. Moreover, with additional mutual structural regularization (Mutual-Reg), Dual-Tuning can further boost the cross-test performance, and also obtain good performance on the self-test setting.

For the ablation study of compatible prototype loss, incorporating new prototypes for a mixed retrieval can achieve better representation than only using the old prototypes (Compatible Prototype v.s. Center-based Prototype). Besides, for the mutual structural regularization (Mutual-Reg), Dual-Tuning can further boost the cross-test performance, and also obtain good performance on the self-test setting.

For the ablation study of compatible prototype loss, incorporating new prototypes for a mixed retrieval can achieve better representation than only using the old prototypes (Compatible Prototype v.s. Center-based Prototype). Besides, for the mutual structural regularization (Mutual-Reg), Dual-Tuning can further boost the cross-test performance, and also obtain good performance on the self-test setting.

Influence of different queue sizes for prototype loss: Since the features in a queue are calculated from different mini-batches, the queue size will influence features’ consistency. We further analyze the influence of different queue sizes for memory-based prototypes. We select 5 different queue sizes for the Market1501 dataset trained with ResNet50 backbone. With the queue size of 1024, 2048, 4096, 8192 and 16384, the independently trained model achieve 81.42%, 81.83%, 82.06%, 81.87%, and 80.29% mAP respectively. The queue size does not have a great impact on performance. One possible reason is that the network is updated with a momentum scheme, this enables a large and consistent queue for learning prototypes [45].

**Center-based asymmetric loss:** We further construct different metric-based loss to demonstrate the effectiveness of compatible prototype loss. After obtaining the old centers, we can also conduct an asymmetric center loss, whose triplet unit is $(\phi_O(x_c), m_c, m_c')$. The positive representation is the old prototype $m_c$ with the same class as $x_c$, and the negative is an old prototype of other classes $m_c'$. As shown in Table VII, the center-based asymmetric loss can achieve better performance than triplet loss, while worse than our compatible prototype loss. The results further demonstrate the necessity of using the whole structure information in embedding space.

### D. Discussion

In this section, we first explore the mixed gallery scenario, then analyze the feature compatibility under different supervision losses and model architectures. Without additional explanation, both the new and old models use ResNet18 backbone, and are supervised by softmax (cross-entropy) loss.

**Performances on the mixed gallery:** In the gallery, there exist the old features and continuously added new features. To simulate this scenario, a hybrid gallery retrieval experiment is performed on Market1501 dataset. As shown in Fig. 4, we change the ratio of the old and new features in the gallery and report the results of different methods. The performances of asymmetric...
triplet loss and contrastive loss [12] extremely degrade in the mixed gallery settings. Under the setting of 80% old data and 20% new data in the gallery, asymmetric triplet can only achieve 50.96% mAP, which is much inferior to our 70.02% mAP. As asymmetric loss lacks global optimization, in mixed setting, its new and old features of different classes overlap in embedding space. The space becomes chaotic, then result drops. Differently, the proposed compatible prototype loss has global optimization ability. Furthermore, it mixes new and old prototypes to simulate the mixed gallery scenario during training stage. Such great performance advantages strongly demonstrate the robustness of our method.

Results of different supervision losses: Different supervision losses will generate different feature distributions. We test using softmax and triplet loss [13] for the old model training, and adopting softmax, softmax+triplet [13], and circle loss [55] for the new model. As shown in Table VIII, BCT [14] can not work when the old model is supervised by triplet loss without a classifier, while our prototype loss in Dual-Tuning can achieve compatibility. Additionally, BCT suffers huge performance drops in circle v.s. softmax setting, since the circle loss has a different classification mechanism from softmax. Differently, our Dual-Tuning only suffers a slightly performance drops for self-test, and achieve impressive result for cross-test, which beats BCT by a remarkable margin. The superior results indicate the designed compatible prototype loss can mitigate the affection by the distribution discrepancy caused by different supervision losses.

Results of different model architectures: We also conduct experiments that the new model with different architectures and feature dimensions from the old model. First, we test the new model with ResNet50 of 2048 feature dimension, and old model with ResNet18 of 512 feature dimension. As shown in Table I, IV and V, during retrieval with old gallery features, using new ResNet50 feature as query can achieve better performance than the old ResNet18 feature.

More results of different architectures are shown in Table IX. We test 5 different architectures, including the extremely lightweight networks Osnet384 and Osnet512 [56], the latest split-attention network ResNeSt50 [57], and the widely used networks ResNet18 and ResNet50. The feature dimensions for Osnet384, Osnet512, and ResNeSt50 are 384, 512, and 2048, respectively. We use zero padding [14] to align features with different dimensions. For different architectures, compatibility will be difficult to obtain. The cross-test between new and old models can achieve impressive performances for any configuration. And in some configuration, the self-test performance faces a slight drop. For example, in the Osnet512 v.s. Osnet384 settings, the self-test performance drops 0.46% mAP (85.15% v.s. 85.61%).

Towards multi-model and sequential compatibility: We simulate the scene with three model versions on Market1501 dataset. These three models are sequentially updated using 25%, 50%, and 100% training data, respectively. The version 2 model $\phi_2$ is compatible to version 1 $\phi_1$. The version 3 model $\phi_3$ only conducts supervision from $\phi_2$ to achieve compatibility. The sequential compatible results are shown in Table X. We can find that, $\phi_3$ can achieve feature compatibility with $\phi_1$, even $\phi_1$ is not directly involved in training $\phi_3$. Compared with BCT, we can better solve the sequential updated models, e.g., 60.51% v.s. 53.64% mAP on cross-test between $\phi_3$ and $\phi_1$.

Domain gap and no overlapping classes between new and old datasets: If the new and old datasets have a domain gap, the old model will perform worse in the new scenario. Actually, if the old model performs extremely poorly in the new scenario, it will not be deployed. Domain adaptation is usually used to obtain an acceptable model before the model is deployed. To analyze the performance in such extreme scenarios, we still build the domain gap experiments.

We use Dukemtmc-ReID [58] and MSMT17 to train the new and old models respectively, and test on Dukemtmc-ReID. They are collected in different countries with severe distribution shifts.
As shown in Table XI, the old model only has 41.79% mAP on Dukemtmc test set, yet replacing query as its class center can get 71.52% mAP, since the center is very robust against noises. Thus, our Dual-Tuning can get good cross-test results by making a new feature to surround its old center (62.61% v.s. 41.79%).

Moreover, since there are no overlapping classes between new and old datasets, BCT can not work. Differently, our compatible prototype loss and O2N-Reg (\(LCE(\phi_N, h_O; \tau_N')\) in (7)) can work well.

**Better old model:** We have simulated the scenarios where the old model needs to be updated to obtain better performance by a new model in other experiments. In Table. XII, we test the scenario that the old model has a larger network with better performance than the new model. We find that if the old model has better performance, the compatible model trained by our Dual-Tuning can obtain more knowledge from the old model and get better self-test results (82.59% v.s. 80.43%).

**Different initializations analysis:** For the same architecture, we analyze the results of different initialization schemes for training compatible models. All the old models are initialized by ImageNet pre-trained models. The compatible new models are initialized by random initialization, ImageNet pre-trained model, and old model. In Table XIII, initializing the new and old models with identical initial weights (i.e., ImageNet model) can facilitate compatibility than random initialization. Besides, using the old model as initialization can further boost the performance.

**Different feature dimension alignment scheme:** Besides the zero-padding used in matching features of different dimensions, we further explore using extra transformation to achieve feature dimension alignment. For example, using ResNet18 (512-dim) and ResNet50 (2048-dim) as the old and new model, respectively, additional transformation (a fully connected layer) is added for ResNet50 to project the 2048-dim feature into 512-dim. In Table. XIV, the extra transformation does not get better performance than zero-padding scheme. Besides, in the self-test, we find that using the first 512 elements of the 2048-dim feature can already achieve a similar performance as using the original 2048-dim feature. Thus, the zero-padding scheme can work well on the cross-test experiments.

**Training and inference efficiency:** Our compatible learning framework is very efficient. Most of the training cost comes from the forward propagation and back propagation, and the time cost of loss calculation can be considered negligible. Therefore, our computation cost is almost the same as the other methods, such as BCT [14], Asymmetric Triplet [12], etc. Compared to the independently trained new model, our Dual-Tuning only bring an additional 4 minutes training cost on the Market1501 dataset, as shown in Table XV. The new and old models are ResNet18. We test the complexity on a PC with i7 CPU and 1 RTX 2080 Ti.

As for the inference time, only the new model is used to extract features. Therefore, when adopting the same model architecture for the new model, the inference time of all the compatible methods is the same. For example, for a resnet18 model, the inference time is \(\sim 0.01\) s for a batch (64 images, input size is 224 × 224).

**V. Conclusion**

In this paper, we propose a Dual-Tuning method to achieve a global optimization for feature compatible learning. Dual-Tuning contains a compatible prototype loss for an explicit embedding feature alignment and a mutual structural regularization by component interoperation. We conduct experiments on four datasets, including the person ReID datasets and million-scale image classification datasets. Compared with other methods, we can achieve superior compatible performance in mixed gallery.
scenarios, and can better deal with the distribution discrepancy between new and old embedding spaces.

There remain several future directions for compatible feature learning. 1) The real-world datasets involving large-scale classes or identities should be further investigated. 2) Neural architecture search, model compression, and model quantization methods can obtain more compact models. The combination of these methods with feature compatible learning needs to be further explored. 3) To obtain better compatibility, advanced loss functions and optimization strategies are also worth to be deeply explored.

REFERENCES

[1] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, “Deep image retrieval: Learning global representations for image search,” in Proc. Eur. Conf. Comput. Vis., 2016, pp. 241–257.

[2] O. Russakovsky et al., “ImageNet large scale visual recognition challenge,” Int. J. Comput. Vis., vol. 115, no. 3, pp. 211–252, 2015.

[3] Z. Gao et al., “DCR: A unified framework for holistic/partial pair ReID,” IEEE Trans. Multimedia, vol. 23, pp. 3332–3345, 2021.

[4] C. Yan et al., “Beyond triplet loss: Person Re-identification with fine-grained difference-aware pairwise loss,” IEEE Trans. Multimedia, vol. 24, pp. 1665–1677, 2022.

[5] Z. Gao et al., “STNReID: Deep convolutional networks with pairwise spatial transformer networks for partial person re-identification,” IEEE Trans. Multimedia, vol. 22, no. 11, pp. 2905–2913, Nov. 2020.

[6] Y. Fu et al., “Self-similarity grouping: A simple unsupervised cross domain adaptation approach for person re-identification,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 6112–6121.

[7] Y. Bai et al., “Group-sensitive triplet embedding for vehicle reidentification,” in Proc. IEEE Trans. Multimedia, vol. 20, pp. 2385–2399, 2018.

[8] Z. Tang et al., “PAMTRI: Pose-aware multi-task learning for vehicle re-identification using highly randomized synthetic data,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 211–220.

[9] Y. Lou, Y. Bai, J. Liu, S. Wang, and L. Du., “Veri-wild: A large dataset and a new method for vehicle re-identification in the wild,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 3235–3243.

[10] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, “ArcFace: Additive angular margin loss for deep face recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 4690–4699.

[11] H. Wang et al., “CosFace: Large margin cosine loss for deep face recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 5265–5274.

[12] M. Badnik and Y. Arvithis, “Asymmetric metric learning for knowledge transfer,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 8228–8238.

[13] R. Hadsell, S. Chopra, and Y. LeCun, “Dimensionality reduction by learning an invariant mapping,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2006, pp. 1735–1742.

[14] Y. Shen, Y. Xiong, W. Xia, and S. Souto, “Towards backward-compatible representation learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 6368–6377.

[15] J. Zhong, X. Wang, Z. Kou, J. Wang, and M. Long, “Bi-tuning of pre-trained representations,” 2020, arXiv:2011.06182.

[16] H. Luo et al., “A strong baseline and batch normalization neck for deep person re-identification,” IEEE Trans. Multimedia, vol. 22, pp. 2597–2609, 2020.

[17] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, “A discriminative feature learning approach for deep face recognition,” in Proc. Eur. Conf. Comput. Vis., 2016, pp. 499–515.

[18] C.-Y. Wang, Y.-L. Chang, S.-T. Yang, D. Chen, and S.-H. Lai, “Unified representation learning for cross model compatibility,” 2020, arXiv:2008.04821.

[19] H. Hu et al., “Towards visual feature translation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 3004–3013.

[20] R. Duggal et al., “Compatibility-aware heterogeneous visual search,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 10723–10732.
[48] L. Zheng et al., “Scalable person re-identification: A benchmark,” in Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 1116–1124.

[49] L. Wei, S. Zhang, W. Gao, and Q. Tian, “Person transfer GAN to bridge domain gap for person re-identification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 79–88.

[50] F. Wang et al., “The devil of face recognition is in the noise,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 765–780.

[51] B. Maze et al., “IARPA janus benchmark-C: Face dataset and protocol,” in Proc. Int. Conf. Biometrics, 2018, pp. 158–165.

[52] J. Deng et al., “Imagenet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2009, pp. 248–255.

[53] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, “Places: A 10 million image database for scene recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 6, pp. 1452–1464, Jun. 2018.

[54] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.

[55] Y. Sun et al., “Circle loss: A unified perspective of pair similarity optimization,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 6398–6407.

[56] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang, “Omniscale feature learning for person re-identification,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 3702–3712.

[57] H. Zhang et al., “ResNeSt: Split-attention networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops, 2022, pp. 2736–2746.

[58] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, “Performance measures and a data set for multi-target, multi-camera tracking,” in Proc. Eur. Conf. Comput. Vis., 2016, pp. 17–35.

Jile Jiao received the B.S. degree from Central South University, Changsha, China, in 2009 and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2014. She is currently an Algorithm Engineer with Intime Department, Alibaba Group, Beijing, China. Her research interests include computer vision, image processing, and person reid and their application in retail industry.

Xuetao Feng received the B.E. degree from the Singapore University of Technology and Design, Singapore, and the Singapore University of Technology and Design, Singapore, in 2019. He is currently an Assistant Professor with the Singapore University of Technology and Design, Singapore. His research interests include computer vision and machine learning.

Ling-Yu Duan received the Ph.D. degree in information technology from The University of Newcastle, Callaghan, NSW, Australia, in 2008. He is currently a Full Professor with the National Engineering Laboratory of Video Technology, School of Electronics Engineering and Computer Science, Peking University (PKU), Beijing, China. Since 2012, he has been the Associate Director of the Rapid-Rich Object Search Laboratory, a Joint Lab between Nanyang Technological University, Singapore, and PKU. Since 2019, he has also been with Peng Cheng Laboratory, Shenzhen, China. He has authored or coauthored about 200 research papers. His research interests include multimedia indexing, search, and retrieval, mobile visual search, visual feature coding, and video analytics. He was the recipient of the IEEE ICME Best Paper Award in 2019/2020, IEEE VCIP Best Paper Award in 2019, EURASIP Journal on Image and Video Processing Best Paper Award in 2015, Ministry of Education Technology Invention Award (First Prize) in 2016, National Technology Invention Award (Second Prize) in 2017, China Patent Award for Excellence (2017), National Information Technology Standardization Technical Committee Standardization Work Outstanding Person Award in 2015. He was a Co-Editor of MPEG Compact Descriptor for Visual Search Standard (ISO/IEC 15938-13) and MPEG Compact Descriptor for Video Analytics (CDVA) Standard (ISO/IEC 15938-15). He is also an Associate Editor for IEEE TRANSACTIONS ON MULTIMEDIA, ACM TRANSACTIONS ON INTELLIGENT SYSTEMS AND TECHNOLOGY and ACM Transactions on Multimedia Computing, Communications, and Applications, and the Area Chair of ACM MM and IEEE ICME. He is the member of the MSA Technical Committee in IEEE-CAS Society.