Asymmetric Feature Fusion for Image Retrieval

Hui Wu[1] Min Wang[2*] Wengang Zhou[1,2*] Zhenbo Lu[2] Houqiang Li[1,2]

[1]CAS Key Laboratory of Technology in GIPAS, University of Science and Technology of China

[2]Institute of Artificial Intelligence, Hefei Comprehensive National Science Center

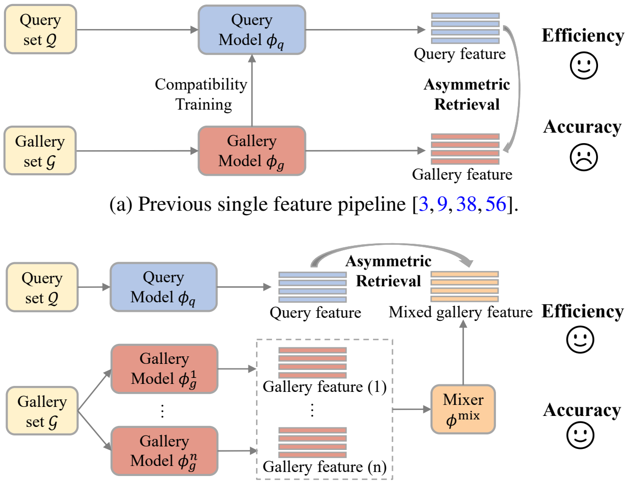
wh241300@mail.ustc.edu.cn, {wangmin,luzhenbo}@iai.ustc.edu.cn, {zhwg,lihq}@ustc.edu.cn

Abstract

In asymmetric retrieval systems, models with different capacities are deployed on platforms with different computational and storage resources. Despite the great progress, existing approaches still suffer from a dilemma between retrieval efficiency and asymmetric accuracy due to the limited capacity of the lightweight query model. In this work, we propose an ***A****symmetric* ***F****eature* ***F****usion (AFF) paradigm, which advances existing asymmetric retrieval systems by considering the complementarity among different features just at the gallery side. Specifically, it first embeds each gallery image into various features, e.g., local features and global features. Then, a dynamic mixer is introduced to aggregate these features into compact embedding for efficient search. On the query side, only a single lightweight model is deployed for feature extraction. The query model and dynamic mixer are jointly trained by sharing a momentum-updated classifier. Notably, the proposed paradigm boosts the accuracy of asymmetric retrieval without introducing any extra overhead to the query side. Exhaustive experiments on various landmark retrieval datasets demonstrate the superiority of our paradigm.*

1. Introduction

Image retrieval [17, 30, 34, 40, 49, 53] has been studied for a long time in the literature. Typically, high-performing image retrieval systems deploy a large powerful model to embed both query and gallery images, which is widely known as **symmetric image retrieval**. However, in some real-world applications, *e.g.*, mobile search, the query side is constrained by resource limitation and thus cannot meet the overhead of deploying a large model. To this end, the paradigm of **asymmetric image retrieval** is first proposed in HVS [9] and AML [3], which has attracted increasing attention from the community [26, 38, 43, 56, 62]. In such a paradigm, deep representation models with different capacities are first trained to be compatible and then deployed on platforms with different resources to strike a balance between retrieval accuracy and efficiency, as shown in Fig. 1a.



(a) Previous single feature pipeline [3, 9, 38, 56].

(b) Our asymmetric feature fusion pipeline.

Figure 1. Illustration of (a) **previous single-feature asymmetric retrieval pipeline** and (b) our **asymmetric feature fusion paradigm**. Due to limited capacity of the lightweight model, existing pipeline achieves efficiency for the query side at the cost of retrieval accuracy degradation. In contrast, our approach enhances existing asymmetric retrieval pipeline from the perspective of *gallery feature fusion*. For efficient retrieval, a dynamic mixer is introduced to aggregate multiple gallery features into a compact embedding. Query model and mixer are jointly trained with compatible constraints. Our method realizes high efficiency without sacrificing retrieval accuracy.

For an asymmetric retrieval system, the most crucial thing is to ensure that the embedding spaces of different models are well aligned. To this end, BCT [38] first proposes a *backward-compatible* learning framework, in which the classifier of the gallery model is inherited to guide the learning of the query model. Recently, various efforts have been devoted to improving cross-model feature compatibility in terms of training objectives [3, 26, 51, 56, 57], model structures [8, 9], *etc*. Despite the great progress, a dilemma

*Corresponding Author: Min Wang and Wengang Zhou.

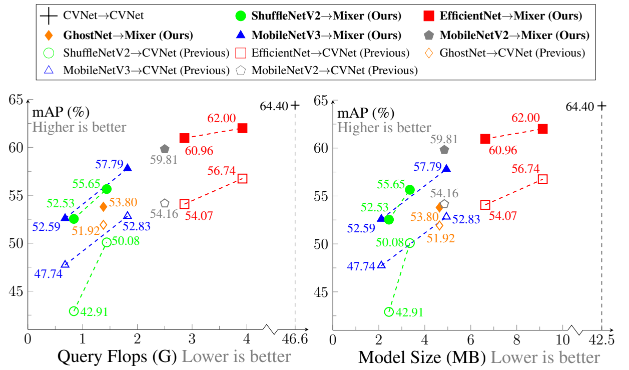


Figure 2. **Average mAP *vs*. FLOPs/Model Size of the query model** for $\mathcal{R}$Oxf + 1M [33] dataset. The notation format "query model → gallery model" in the legend means embedding queries with the query model and retrieving in a gallery set embedded by the gallery model. **A line connecting the dots with one color** represents **a family of lightweight models with different model sizes**. *Previous*: The latest asymmetric retrieval method CSD [56] is adopted to train query model with CVNet [21] deployed as gallery model. *Ours*: our paradigm utilizes CVNet, Token [54], DELG [4] and DOLG [59] to generate aggregated gallery features and trains the mixer and query model jointly.

is still unresolved, *i.e*., the accuracy of asymmetric retrieval is still unsatisfactory compared to that of symmetric retrieval, especially in limited-resource and large-scale scenarios, as shown in Fig. 2 (□, ♢, . . . , △ *vs*. +). We argue that such dilemma is due to the low capacity of lightweight query model, which cannot perfectly achieve feature compatibility with the static powerful gallery model.

To alleviate above issue, we introduce a new paradigm named **A**symmetric **F**eature **F**usion (AFF). It boosts the accuracy of existing asymmetric retrieval systems by considering the complementarity among different features, as shown in Fig. 1b. On the gallery side, it deploys several large powerful models on the cloud to extract diverse features, *e.g*., local features, which are suitable for capturing local matches, or global features that are effective for holistic semantic matching. For efficient retrieval, a dynamic mixer is further proposed to aggregate diverse gallery features into compact embedding, which allows efficient vector search [11, 18] to be exploited. As for the query side, queries are embedded with a single lightweight model. It eliminates time-consuming multiple feature extraction and aggregation processes, realizing a solution suitable for resource-constrained platforms. During training, all the gallery models are fixed, while the mixer and query model are trained jointly by a momentum-updated classifier for achieving feature compatibility.

Compared to previous retrieval approaches, the proposed paradigm has two unique advantages. First, it fuses various features on the gallery side, which notably advances the retrieval accuracy of existing asymmetric retrieval systems. Although the extraction and aggregation processes increase the computational complexity, they are typically performed on resource-rich cloud platforms. In addition, gallery images are embedded offline in advance, whose computational overhead has no influence on the query side. Second, compared with multi-feature fusion methods, our paradigm only deploys a single lightweight model on the query side, which is free of the complex and time-consuming multi-feature extraction and aggregation. Thus, it introduces no extra computational and storage overhead for the query side. Overall, with the proposed asymmetric feature fusion paradigm, our approach achieves high retrieval efficiency and accuracy simultaneously, as shown in Fig. 2 (□ *vs*. ■,. . . ,△ *vs*. ▲). To evaluate our approach, comprehensive experiments are conducted on popular landmark retrieval datasets. The proposed paradigm realizes promising performance improvement for existing asymmetric retrieval systems and leads to the state-of-the-art results across the public benchmarks.

2. Related Work

Feature Representation. In image retrieval, feature representation plays a key role. Hand-crafted local features [2, 24] are widely used in early image retrieval systems [27, 32, 40]. Recently, local features extracted from convolutional neural networks (CNNs) are shown to be more effective [7, 29, 31, 46]. They learn feature detection and representation jointly by attention mechanism [31, 48, 52, 55] or non-maximal suppression [10]. The detected local features are further utilized for geometric verification [32] or aggregated into compact representations by VLAD [16], ASMK [47], *etc*., for efficient retrieval. Recently, global features such as RMAC [49], GeM [34], DELG [4], DOLG [59], Token [54] and CVNet [21], are typically extracted from CNNs by spatial pooling [34, 41, 42], which demonstrate more effectiveness in holistic semantic matching over local features.

Despite the great progress, existing image retrieval systems usually deploy large powerful models for high retrieval accuracy. However, some real-world applications need to deploy query models on resource-constrained platforms, *e.g*., mobile phones, which cannot meet the demand of large models for computational and storage resources. To address this issue, our approach focuses on the setting of asymmetric retrieval, where the query side deploys a lightweight model while the gallery side applies a large one.

Feature Compatibility. The paradigm of *feature compatibility learning* is first proposed by BCT [38]. It enforces the feature of the query model to be close to the corresponding class centroids of the gallery model. Under this paradigm, several efforts [3, 9, 26, 56, 62] have been devoted to improving the feature compatibility across different models. Specifically, AML [3] introduces asymmetric regression loss and contrastive loss to train the query model. CSD [56] takes a step further by constraining the query model to maintain the nearest neighbor structure in the embedding space of the gallery model. Recently, LCE [26] proposes to align the classifiers of different models with a tight boundary loss. HVS [9] further resorts to neural architecture search technique to search for the optimal compatibility-aware model architecture. FCT [36] stores "side information", which is later leveraged to transfer the gallery features for other retrieval tasks. Besides, when solving the model regression problem, methods including PCT [58], REG-NAS [8] and RACT [61], also utilize feature compatibility to alleviate "negative flip".

Differently, to boost existing asymmetric retrieval systems, we introduce a new asymmetric feature fusion paradigm. It enhances the discriminativeness of image features by aggregating diverse features just at the gallery side. Our approach is readily combined with existing methods to achieve better asymmetric retrieval accuracy efficiently.

Lightweight Network. The architecture of deep convolutional neural networks [13, 20] has been evolving for many years. As the application complexity increases [35], model size becomes larger, requiring more computational and storage resources. However, besides accuracy, resource overhead is another important consideration in real-world applications. Real-world tasks usually expect to deploy optimal models on the target resource-constrained platforms. The immediate demand motivates a series of work, *e.g.*, MobileNets [14, 37], ShuffleNets [25, 65], GhostNets [12] and EfficientNets [45], for lightweight model design.

In this work, we focus on asymmetric retrieval in resource-constrained scenarios. Various lightweight models mentioned above are utilized as query models on resource-constrained end platforms.

Feature Fusion. Feature fusion has been widely studied in computer vision, *e.g.*, detection [22, 60], multimedia retrieval [15, 39], *etc*. As for image retrieval, it is broadly divided into three levels. The first is feature level, where features of different modalities [5], scales [42, 59], *etc*., are effectively fused into a single feature. The second is indexing level, where multiple features are jointly indexed [64, 66], or multiple visual vocabularies are fused together [68]. The last is ranking level. Given several ranking lists returned by different retrieval algorithms, graph-based [63], or context-based [67] methods fuse them into the final ranking list.

However, all these methods require to extract multiple features on the query side. It inevitably increases the computational and storage complexity, which is hardly affordable for resource-constrained platforms. In contrast, we introduce a new asymmetric feature fusion paradigm, in which only a single lightweight model is deployed to embed queries and the gallery set is processed offline by various large powerful models on the cloud platforms. The proposed paradigm boosts the accuracy of asymmetric retrieval without adding any extra overhead to the query side.

3. Preliminary on Asymmetric Retrieval

Asymmetric image retrieval aims to deploy models of different sizes on different platforms to realize search efficiency while preserving retrieval accuracy. Given a query set $\mathcal{Q}$ and a gallery set $\mathcal{G}$, query model $\phi_q : x \to \mathbb{R}^D$ and gallery model $\phi_g : x \to \mathbb{R}^D$ are deployed to embed them into L_2-normalized features, respectively. Then, the cosine similarities or Euclidean distances between query and gallery features are calculated to measure the similarities between images. Usually, an asymmetric retrieval system is expected to achieve similar accuracy as that of a symmetric retrieval system, *i.e.*, $\mathcal{M}(\phi_q(\mathcal{Q}), \phi_g(\mathcal{G})) \approx \mathcal{M}(\phi_g(\mathcal{Q}), \phi_g(\mathcal{G}))$, where $\mathcal{M}(\cdot, \cdot)$ denotes the evaluation metric for retrieval, *e.g.*, mAP or Recall@K.

Despite the promising performance achieved by existing asymmetric retrieval methods, we still observe notable retrieval accuracy degradation when compared to deploying large powerful models on both query and gallery sides (□, ◇, . . . , △ *vs.* + in Fig. 2), *i.e.*, $\mathcal{M}(\phi_g(\mathcal{Q}), \phi_g(\mathcal{G})) > \mathcal{M}(\phi_q(\mathcal{Q}), \phi_g(\mathcal{G}))$. This is due to the limited capacity of lightweight models, which cannot perfectly achieve feature compatibility with large powerful models.

In this work, we alleviate the dilemma from the perspective of feature fusion and a new asymmetric feature fusion paradigm is introduced. Specifically, various large powerful models are deployed on the gallery side to extract features, which are further aggregated into compact embedding with a mixer. As for the query side, only a lightweight model is deployed, which is jointly trained with the mixer for feature compatibility. The proposed paradigm improves the accuracy of asymmetric retrieval systems without introducing any overhead to the resource-constrained query side.

4. Asymmetric Feature Fusion

4.1. Overview

As shown in Fig. 3, our AFF consists of multiple global feature models $\{\phi_g^i : x \to \mathbb{R}^{D_i}\}_{i=1}^K$ and local feature models $\{\phi_l^i : x \to \mathbb{R}^{N_i \times d_i}\}_{i=1}^M$ on the gallery side and a lightweight model $\phi_q : x \to \mathbb{R}^d$ on the query side. Let $\mathcal{T}$ denote a training dataset. On the gallery side, each image x in $\mathcal{T}$ is first embedded into multiple global features $\mathbf{G} = \{\boldsymbol{g}^i \in \mathbb{R}^{D_i}\}_{i=1}^K$ and several sets of local features $\mathbf{L} = \{\boldsymbol{l}^i \in \mathbb{R}^{n_i \times d_i}\}_{i=1}^M$, respectively:

$$\boldsymbol{g}^i = \phi_g^i(x) \in \mathbb{R}^{D_i},\ i = 1, 2, \ldots, K, \tag{1}$$

$$\boldsymbol{l}^i = \phi_l^i(x) \in \mathbb{R}^{n_i \times d_i},\ i = 1, 2, \ldots, M. \tag{2}$$

Typically, each local feature is associated with a coordinate tuple and a scale factor, indicating the location and image scale from which it is extracted. Our method ignores these information. All the global and local features

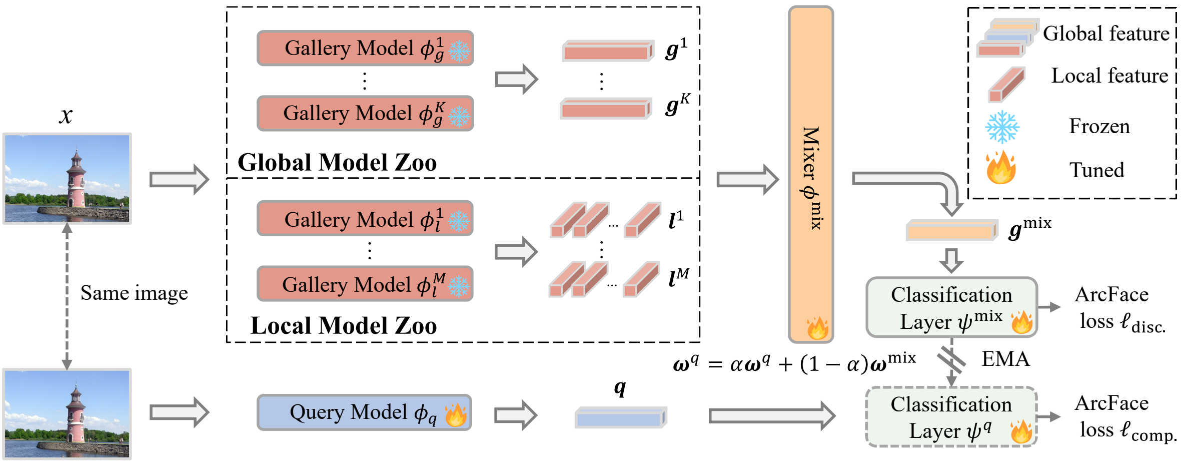


Figure 3. **Overview of the asymmetric feature fusion framework.** Given an image x, several models, *e.g.*, global feature models $\{\phi_g^i : x \rightarrow \mathbb{R}^{D_i}\}_{i=1}^K$ and local feature models $\{\phi_l^i : x \rightarrow \mathbb{R}^{n_i \times d_i}\}_{i=1}^M$, are deployed on the gallery side to embed it into various features $\mathbf{G} = \{\boldsymbol{g}^i \in \mathbb{R}^{D_i}\}_{i=1}^K$ and $\mathbf{L} = \{\boldsymbol{l}^i \in \mathbb{R}^{n_i \times d_i}\}_{i=1}^M$. Then, a dynamic mixer (Sec. 4.2) is introduced to aggregate these features into a compact embedding $\boldsymbol{g}^{\text{mix}} \in \mathbb{R}^d$, which is further fed into the classification layer ψ^{mix} for end-to-end optimization. On the query side, a lightweight model ϕ_q maps the same image x to embedding $\boldsymbol{q} \in \mathbb{R}^d$. After that, $\boldsymbol{q}$ is fed into another classification layer ψ^q, a momentum-updated version of ψ^{mix}, to train the query network for feature compatibility. Classification is adopted as the pretext task in the form of ArcFace [6] loss $\ell_{\text{disc.}}$ and $\ell_{\text{comp.}}$ (Sec. 4.3) to train the mixer and the query network jointly.

are mapped to the same dimension of d by the corresponding fully-connected layers:

$$\boldsymbol{f}_g^i = \boldsymbol{g}^i \boldsymbol{W}_g^i \in \mathbb{R}^d,\ i = 1, 2, \ldots, K, \tag{3}$$

$$\boldsymbol{f}_l^i = \boldsymbol{l}^i \boldsymbol{W}_l^i \in \mathbb{R}^{n_i \times d},\ i = 1, 2, \ldots, M. \tag{4}$$

After that, various gallery features are stacked together to form a feature sequence:

$$\mathbf{F} = [\boldsymbol{f}_g^1; \ldots; \boldsymbol{f}_g^K; \boldsymbol{f}_l^1; \ldots; \boldsymbol{f}_l^M] \in \mathbb{R}^{N \times d}, \tag{5}$$

where $N = K + \sum_{i=1}^M n_i$ is the total number of the gallery features. To reduce the storage overhead of the gallery side and improve search efficiency, a mixer $\phi_{\text{mix}} : \mathbb{R}^{N \times d} \rightarrow \mathbb{R}^d$ (Sec. 4.2) is further introduced to transform $\mathbf{F}$ into compact embedding $\boldsymbol{g}^{\text{mix}} = \phi_{\text{mix}}(\mathbf{F}) \in \mathbb{R}^d$. On the query side, the same training image x is embedded into $\boldsymbol{q}$ by the lightweight query model: $\boldsymbol{q} = \phi_q(x) \in \mathbb{R}^d$.

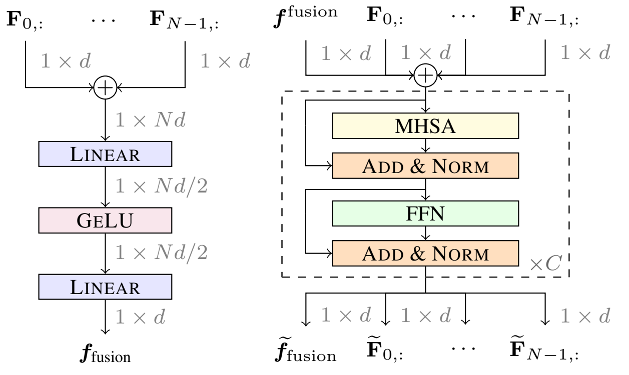
During training, the well-trained gallery models are kept frozen. Only the dynamic mixer ϕ_{mix} and the query model ϕ_q are jointly trained for feature compatibility. The final objective function (Sec. 4.3) consists of two losses:

$$\underset{\phi_{\text{mix}}, \phi_q}{\arg\min} \frac{1}{|\mathcal{T}|} \sum_{x \in \mathcal{T}} (\ell_{\text{disc.}}(\phi_{\text{mix}}, x) + \ell_{\text{comp.}}(\phi_q, x)), \tag{6}$$

where $\ell_{\text{disc.}}(\phi_{\text{mix}}; x)$ ensures the discrimination of the aggregated feature $\boldsymbol{g}^{\text{mix}}$, and $\ell_{\text{comp.}}(\phi_q; x)$ is designed to align query feature $\boldsymbol{q}$ and aggregated feature $\boldsymbol{g}^{\text{mix}}$ in the same latent space so that they are mutually compatible.

4.2. Dynamic Mixer

Given an image encoded by various features $\mathbf{F}$, feature fusion aims to combine those features for better retrieval accuracy. A simple way is to concatenate various features and perform dimension reduction, which is implemented by several fully-connected layers (Fig. 4a). However, it leads to an over-parameterized mixer when the number of features is large, which may cause overfitting.



(a) Simple baseline

(b) Ours

Figure 4. **Different variants of mixer**. (a) Simple baseline: different features are concatenated, followed by dimension reduction with several fully-connected layers. (b) Our mixer: a fusion token and the feature sequence are iteratively processed by a transformer layer [50], where the fusion token dynamically aggregates beneficial features from various gallery features.

In this work, attention mechanism [50] is adopted to aggregate various features (Fig. 4b). A learnable fusion token $\boldsymbol{f}_{\text{fusion}} \in \mathbb{R}^d$ is first added to the top of $\mathbf{F}$ to form the input:

$$\mathbf{F}_{\text{input}} = [\boldsymbol{f}_{\text{fusion}}; \mathbf{F}] \in \mathbb{R}^{(N+1) \times d}. \tag{7}$$

Then, $\mathbf{F}_{\text{input}}$ is iteratively processed C times by a transformer layer, which is formulated as:

$$\begin{aligned}
\bar{\mathbf{Z}}^{i+1} &= \mathrm{LN}(\mathbf{Z}^{i} + \mathrm{MHSA}(\mathbf{Z}^{i})),\\
\mathbf{Z}^{i+1} &= \mathrm{LN}(\bar{\mathbf{Z}}^{i+1} + \mathrm{MLP}(\bar{\mathbf{Z}}^{i+1})),\\
\mathrm{MLP}(\bar{\mathbf{Z}}^{i+1}) &= \mathrm{GELU}(\bar{\mathbf{Z}}^{i+1}\boldsymbol{W}_1)\boldsymbol{W}_2,\\
i &= 0, \dots, C-1,
\end{aligned} \tag{8}$$

where $\mathbf{Z}^0 = \mathbf{F}_{\text{input}}$; MHSA is the Multi-Head Self-Attention [50]; MLP is a two-layer perceptron with parameter matrices $\boldsymbol{W}_1 \in \mathbb{R}^{d\times d_e}$ and $\boldsymbol{W}_2 \in \mathbb{R}^{d_e\times d}$, and an intermediate dimension $d_e = 2 \times d$; LN is the layer normalization [1]. The final output fusion token $\widetilde{\boldsymbol{f}}_{\text{fusion}} = \mathbf{Z}^{C-1}_{0,:}$ is adopted as the aggregated feature $\boldsymbol{g}^{\text{mix}}$ for the gallery side.

4.3. Training Objective Functions

To ensure the superiority of our asymmetric feature fusion paradigm, there are two requirements needed to be guaranteed. First, the aggregated feature $\boldsymbol{g}^{\text{mix}}$ is expected to be more discriminative than any single gallery feature. To this end, following the state-of-the-art metric learning methods [4, 21, 59] in image retrieval, classification is adopted as a pretext task in the form of ArcFace loss [6] to train the mixer. Assuming the classification layer $\psi^{\text{mix}} : \mathbb{R}^d \to \mathbb{R}^N$, with N categories, is parameterized by weights $\boldsymbol{\omega}^{\text{mix}} \in \mathbb{R}^{N\times d}$, the loss is formulated as:

$$\ell_{\text{disc.}}(\phi_{\text{mix}}, x) = -\log \frac{e^{s\cdot\cos(\theta^1_y + m)}}{e^{s\cdot\cos(\theta^1_y + m)} + \sum\limits_{j\neq y} e^{s\cdot\cos(\theta^1_j)}}, \tag{9}$$

where y is the label of the training image x, s is a scale factor, m is the margin, and $\theta^1_y = \arccos(\langle \frac{\boldsymbol{\omega}^{\text{mix}}_{y,:}}{\|\boldsymbol{\omega}^{\text{mix}}_{y,:}\|}, \boldsymbol{g}^{\text{mix}}\rangle)$ is the angle between the y-th L_2-normalized prototype of the classifier ψ^{mix} and the feature $\boldsymbol{g}^{\text{mix}}$.

Second, query feature $\boldsymbol{q}$ and aggregated feature $\boldsymbol{g}^{\text{mix}}$ should be compatible with each other. One may share the same classifier between ϕ^{mix} and ϕ_q, which has shown effectiveness in previous methods [9, 26, 38]. However, our approach expects to train ϕ^{mix} and ϕ_q jointly. Simply sharing classifier couples the training of networks with different capabilities, which may damage the discriminative capability of the aggregated embedding $\boldsymbol{g}^{\text{mix}}$. Besides, the classifier parameters $\boldsymbol{\omega}^{\text{mix}}$ evolve rapidly, which cannot provide a stable target to the query model. To this end, we decouple the training processes of ϕ_{mix} and ϕ_q, while ensuring feature compatibility through a momentum update mechanism. ArcFace loss is still adopted for training query model ϕ_q:

$$\ell_{\text{comp.}}(\phi_q, x) = -\log \frac{e^{s\cdot\cos(\theta^2_y + m)}}{e^{s\cdot\cos(\theta^2_y + m)} + \sum\limits_{j\neq y} e^{s\cdot\cos(\theta^2_j)}}, \tag{10}$$

where $\theta^2_y = \arccos(\langle \frac{\boldsymbol{\omega}^{q}_{y,:}}{\|\boldsymbol{\omega}^{q}_{y,:}\|}, \boldsymbol{q}\rangle)$ is the angle between the y-th L_2-normalized prototype of the classifier ψ^q and the feature $\boldsymbol{q}$. Differently, the parameter $\boldsymbol{\omega}^q$ is not updated through back-propagation, but a moving-averaged version of $\boldsymbol{\omega}^{\text{mix}}$:

$$\boldsymbol{\omega}^q \leftarrow \alpha\boldsymbol{\omega}^q + (1-\alpha)\boldsymbol{\omega}^{\text{mix}}, \tag{11}$$

where $\alpha \in [0, 1)$ is a momentum coefficient. Only the parameters $\boldsymbol{\omega}^{\text{mix}}$ are updated by back-propagation. This momentum update in Eq. (11) decouples the training of ϕ_{mix} and ϕ_q while making $\boldsymbol{\omega}^q$ evolve more smoothly than $\boldsymbol{\omega}^{\text{mix}}$.

5. Experiments

5.1. Experimental Setup

Evaluation Datasets and Metrics. We evaluate the proposed framework on three landmark retrieval datasets, including GLDv2-Test [53], Revisited Oxford ($\mathcal{R}$Oxf), Revisited Paris ($\mathcal{R}$Par) [33]. GLDv2-Test contains $761,757$ gallery images, and 390/750 images as public/private query sets, respectively. The evaluation metric is mAP@100. As for $\mathcal{R}$Oxf and $\mathcal{R}$Par, there are 70 queries for both of them, with $4,993$ and $5,007$ gallery images, respectively. mAP on the Medium and Hard settings are reported. Large-scale results are reported with the $\mathcal{R}$1M [33] dataset added (+ 1M).

Gallery and Query models. Four global features including DELG [4], Token [54], DOLG [59], CVNet [21] and two local features HOW [48], *DELG [4] are adopted as gallery features. As for query model, we only keep the feature extractor of lightweight models, *e.g*., ShuffleNets [25] and MobileNets [37], with a GeM pooling [34] layer and a whitening layer added at the end.

Training Details. GLDv2 [53] is adopted for training, which consists of $1,580,470$ images with $81,311$ classes. All the gallery features are extracted offline for training efficiency. During training, a 512×512-pixel region is cropped from each randomly resized training image, followed by random color jittering. We jointly train the mixer and query model for 20 epochs on four NVIDIA RTX 3090 GPUs with a batch size of 128. SGD is adopted as the optimizer with a weight decay of 0.01 and an initial learning rate of 0.001, which linearly decays to 0 when the desired number of steps is reached. C in Eq. (8) is set to 4. d and d_e in Eq. (8) are both set to $2,048$. Margin m and scale s in Eq. (9) and Eq. (10) are set as 0.3 and 32.0, respectively.

5.2. Ablation Study

Variants of the mixer. In Tab. 1, we compare our proposed mixer against simple baseline (MLP), LAFF [15], and OrthF. [59]. Our transformer-based mixer iteratively extracts useful information from various gallery features via a fusion token, which achieves the best retrieval accuracy.

Impact of different gallery features. In Tab. 2, we investigate how our method responds when different gallery features are added. Results show that each feature is beneficial for the task and features have different effects when

Mixer Type	Query Net ϕ_q	GLDv2-Test		$\mathcal{R}$Oxf + 1M		$\mathcal{R}$Par + 1M	
		Private	Public	Medium	Hard	Medium	Hard
MLP	Mixer	31.13	29.63	74.89	53.79	79.97	62.25
	MobV2	27.63	24.36	65.61	42.84	71.09	52.57
LAFF	Mixer	31.90	30.07	76.30	55.68	81.38	64.31
	MobV2	28.43	26.48	66.53	45.10	73.96	54.37
OrthF.	Mixer	30.27	29.33	73.78	52.00	79.34	61.84
	MobV2	26.73	24.52	63.38	38.86	70.70	51.26
Ours	Mixer	**32.85**	**31.27**	**77.84**	**58.91**	**84.43**	**69.44**
	MobV2	**29.85**	**27.68**	**70.47**	**49.16**	**80.01**	**62.58**

Table 1. Analysis of **different mixer architectures**. Mixer: query images are embedded into various features, which are further aggregated into a compact vector; MobV2: MobileNetV2 is deployed to embed query images. Mixer is also adopted as gallery model. LAFF: [15]; OrthF.: [59].

DELG	Token	DOLG	CVNet	Query Net ϕ_q	GLDv2-Test		$\mathcal{R}$Oxf + 1M		$\mathcal{R}$Par + 1M	
					Private	Public	Medium	Hard	Medium	Hard
✓		✓	✓	Mixer	32.31	29.77	75.49	54.96	83.05	67.21
				MobV2	28.53	25.39	66.49	44.83	77.19	59.08
✓	✓		✓	Mixer	32.09	30.31	76.28	55.86	80.97	62.99
				MobV2	27.45	25.75	66.08	44.51	73.18	53.73
✓	✓	✓		Mixer	31.83	30.07	76.30	55.68	81.38	64.31
				MobV2	26.31	25.34	66.10	44.76	75.11	55.02
	✓	✓	✓	Mixer	32.82	30.99	77.71	**59.28**	82.86	66.95
				MobV2	28.35	26.40	70.20	**49.29**	76.91	57.47
✓	✓	✓	✓	Mixer	**32.85**	**31.27**	**77.84**	58.91	**84.43**	**69.44**
				MobV2	**29.85**	**27.68**	**70.47**	49.16	**80.01**	**62.58**

Table 2. Analysis of **different feature combinations**. Mixer: query images are embedded into various features, which are further aggregated into a compact vector; MobV2: MobileNetV2 is deployed to embed query images. Mixer is adopted as gallery model under all settings.

retrieval is performed in different gallery sets. Our method dynamically aggregates diverse features to enhance the representation of gallery images.

Robustness. When it comes to feature fusion, there is usually no idea whether a specific feature is beneficial for the current task or not in real-world applications. Feature fusion inevitably introduces some noisy features.

In Tab. 3, we investigate the robustness of the proposed framework to noisy features. First, three feature extractors with unsatisfactory performance including RGeM [34], VGeM [34], and RMAC [49], are added to the gallery model zoo. These noisy features cause significant performance degradation for direct feature ensemble, *e.g.*, mAP on $\mathcal{R}$Oxf + 1M drops from 75.30 to 71.52. Our method is almost unaffected. We further consider an extreme case, in which **Gaussian White Noise** is directly added to the gallery feature set. This further leads to a huge performance degradation for feature ensemble, *i.e.*, mAP on $\mathcal{R}$Oxf + 1M drops from 71.52 to 57.78. Our method also suffers a slight performance degradation in this setup. To summarize, our mixer dynamically extracts informative features from diverse gallery features by a fusion token, which demonstrates superior robustness to noise.

Default	RGeM	VGeM	RMAC	Noisy	Method	GLDv2-Test		$\mathcal{R}$Oxf + 1M		$\mathcal{R}$Par + 1M	
						Private	Public	Medium	Hard	Medium	Hard
✓					Ensemble	32.28	29.76	75.30	53.24	83.35	67.05
					Mixer	**32.85**	**31.27**	**77.84**	**58.91**	**84.43**	**69.44**
	✓	✓	✓		Ensemble	17.18	15.71	51.95	26.65	56.19	28.69
					Mixer	**20.84**	**19.11**	**57.85**	**29.55**	**62.92**	**37.54**
✓	✓	✓	✓		Ensemble	29.29	26.90	71.52	45.27	76.74	55.77
					Mixer	**32.87**	**31.00**	**77.28**	**58.03**	**83.56**	**68.05**
✓				✓	Ensemble	29.32	26.88	52.30	25.67	54.44	26.52
					Mixer	**32.60**	**30.94**	**76.40**	**57.13**	**82.65**	**66.52**
✓	✓	✓	✓	✓	Ensemble	30.24	28.47	57.78	29.88	59.88	32.58
					Mixer	**32.56**	**30.61**	**77.65**	**56.96**	**82.68**	**66.49**

Table 3. **Robustness analysis (symmetric retrieval)**. *Default*: four global features including DELG [4], Token [54], DOLG [59] and CVNet [21]. RGeM [34], VGeM [34] and RMAC [34] are three global features with weak retrieval accuracy. Noisy is randomly sampled from a i.i.d D-dimensional Gaussian distribution. Ensemble: feature concatenation; Mixer: various features are aggregated into a compact vector with our proposed mixer.

Method	Query Net ϕ_q	Gallery Net ϕ_g	GLDv2-Test		$\mathcal{R}$Oxf + 1M		$\mathcal{R}$Par + 1M	
			Private	Public	Medium	Hard	Medium	Hard
DOLG [59]	Swin-B	Swin-B	29.69	27.63	75.31	53.67	81.19	64.29
+CSD [56]	MobV2	Swin-B	**26.98**	**24.34**	**66.21**	**43.34**	**70.76**	**50.51**
DOLG [59]	Swin-L	Swin-L	28.42	26.27	72.72	50.55	81.43	62.52
+CSD [56]	MobV2	Swin-L	25.65	23.34	64.26	39.82	69.79	46.53
Ours	Mixer	Mixer	**32.93**	**31.63**	**77.58**	**58.30**	**83.68**	**68.04**
Ours	MobV2	Mixer	**28.57**	**26.57**	**68.17**	**47.27**	**77.90**	**59.01**

Table 4. **Comparison of different gallery models**. Feature dimension is set as 512. Swin-B (88 MB) and Swin-L (197 MB): base and large version of Swin Transformer [23]; Mixer (202 MB): DOLG [59], Token [54], CVNet [21] and DELG [4] are adopted for feature fusion.

Larger gallery models. Considering that feature fusion leads to a significant increase in the number of parameters in the gallery model. This raises the question of whether similar performance can be achieved by using a larger model on the gallery side instead. As shown in Tab. 4, simply using a larger model does not necessarily result in better retrieval accuracy. Moreover, larger models typically entail a significant training overhead. In contrast, our approach only requires training a lightweight fusion network while keeping the existing feature extraction network frozen.

Broad applicability. A common image retrieval practice usually first retrieves candidates via similarity search with global features and then re-ranks with corresponding local features. However, the overhead of extracting local features is much higher than that of global features, which is unaffordable for resource-constrained platforms. Here, we explore the effectiveness of fusing global and local features only on the gallery side.

In Tab. 5a, our method is compared with the traditional ASMK [47]-based local feature aggregation ap-

DELG	*DELG	HOW	Method	Ret.	Ext.	GLDv2-Test		$\mathcal{R}$Oxf + 1M		$\mathcal{R}$Par + 1M	
				(s) ↓	(ms) ↓	Private	Public	Medium	Hard	Medium	Hard
	✓		ASMK*	1.042	388.9	20.21	17.47	62.78	38.59	66.71	40.89
	✓		**Ours**	0.345	404.1	**26.76**	**24.34**	**66.53**	**42.97**	**73.02**	**51.21**
	✓		Ours‡	**0.345**	**16.5**	**23.66**	**21.66**	**60.94**	**41.35**	**69.89**	**46.85**
		✓	ASMK*	0.995	258.1	16.52	14.05	63.66	36.84	58.42	30.73
		✓	**Ours**	0.345	273.2	**23.22**	**22.32**	**68.60**	**41.77**	**67.08**	**46.96**
		✓	Ours‡	**0.345**	**16.5**	**22.08**	**20.16**	**62.36**	**40.07**	**64.92**	**41.64**
	✓	✓	ASMK*	2.324	647.2	20.35	17.28	68.92	40.70	68.06	42.43
	✓	✓	**Ours**	0.345	665.1	**27.24**	**25.72**	**72.40**	**49.86**	**76.78**	**57.03**
	✓	✓	Ours‡	**0.345**	**16.5**	**25.06**	**23.28**	**65.18**	**44.23**	**71.49**	**48.76**
✓	✓	✓	**Ours**	0.345	678.7	**28.13**	**26.95**	**73.71**	**51.00**	**78.60**	**59.25**
✓	✓	✓	Ours‡	**0.345**	**16.5**	**26.38**	**24.22**	**66.48**	**46.96**	**73.55**	**51.89**

(a) Aggregating local features into compact embeddings.

DELG	*DELG	HOW	Method	Ret.	Ext.	GLDv2-Test		$\mathcal{R}$Oxf + 1M		$\mathcal{R}$Par + 1M	
				(s) ↓	(ms) ↓	Private	Public	Medium	Hard	Medium	Hard
✓			GR	0.345	113.7	26.14	24.20	63.71	37.45	70.59	46.94
✓			GR‡	**0.345**	**16.5**	23.33	21.33	60.19	34.14	68.25	44.24
✓	✓		GR + GV†	12.752	425.4	26.81	24.73	68.63	45.33	71.38	47.78
✓	✓		GR + RRT†	1.875	425.4	27.94	26.05	69.01	45.47	72.64	49.93
✓	✓		**Ours**	0.345	441.3	**28.71**	**26.79**	**69.28**	**46.82**	**74.76**	**54.15**
✓	✓		Ours‡	**0.345**	**16.5**	**26.25**	**24.18**	**62.00**	**40.31**	**69.11**	**47.62**
✓		✓	GR + GV†	17.433	371.8	26.01	23.96	65.29	38.49	71.00	48.27
✓		✓	GR + RRT†	2.136	371.8	26.23	24.07	66.90	40.43	71.29	48.15
✓		✓	**Ours**	0.345	387.4	**27.45**	**26.06**	**75.08**	**49.81**	**76.37**	**57.53**
✓		✓	Ours‡	**0.345**	**16.5**	**25.48**	**23.82**	**67.32**	**43.80**	**70.53**	**48.75**

(b) Global retrieval followed by local feature re-ranking.

Table 5. **Applicability analysis**. ‡: *asymmetric retrieval setting*. ↓: lower is better; †: re-implementation; Ret.: average retrieval latency for the $\mathcal{R}$Oxf + 1M dataset. Ext.: average feature extraction latency for the query side. **ASMK*** [47] aggregates local features into binarized embedding. **GR**: global retrieval; **GV**: Geometric Verification [32]; **RRT**: Re-ranking Transformers [44]. Re-ranking is performed on the **top-100** candidates returned by global retrieval. HOW [48] and *DELG [4] are two types of local feature with top performance. Mixer is adopted as gallery model. MobileNetv2 and Mixer are adopted as query model under **asymmetric** and **symmetric** setting, respectively.

proaches [48, 52]. Even under the asymmetric setting, it achieves better accuracy with less overhead for the query side. In Tab. 5b, our method also achieves higher accuracy than re-ranking [32, 44] methods under the asymmetric setting. Note that the re-ranking process introduces significant retrieval latency, while the proposed paradigm only needs efficient vector search. To summarize, our method aggregates diverse features on the gallery side, which greatly reduces the overhead for the query side and online retrieval latency while improving retrieval accuracy.

Analysis of the training strategy. Our AFF adopts the momentum update mechanism to jointly train the mixer and query model. Here, we expect to answer two questions:

(1) *Whether momentum update is necessary or not?* Tab. 6 shows the accuracy of the asymmetric retrieval with different momentum values α in Eq. (11). The accuracy increases gradually as α increases. It performs well when α is in $0.99 \sim 0.999$, showing that a slowly progressing classifier ψ_q is beneficial. The importance of the momentum update is also confirmed by the results in Tab. 8. When our AFF is combined with different approaches, coupling the training process of mixer and query model leads to significant performance degradation, *e.g.*, the mAP on $\mathcal{R}$Oxf + 1M drops from 69.66 to 48.31 for "Mixer + CSD [56]". All these results support our motivation of decoupling the training of the mixer and query model.

(2) *Is joint training more effective than two-stage training?* Two-stage training means that the mixer is first trained, followed by the compatibility training of the query model. It avoids the difficulties arising in joint training but leads to a more complex and time-consuming training process. Besides, due to the limited capacity of the query model, directly aligning it to a static powerful model limits the feature compatible training, which is confirmed by previous methods [19, 28]. Experiment results are shown in Tab. 7. When

Momentum α	GLDv2-Test		$\mathcal{R}$Oxf + 1M		$\mathcal{R}$Par + 1M	
	Private	Public	Medium	Hard	Medium	Hard
0	25.86	24.29	62.75	43.19	71.46	50.63
0.5	27.42	24.71	66.36	45.69	75.18	55.75
0.9	**29.90**	26.74	67.76	46.20	78.55	60.05
0.99	29.85	**27.68**	**70.47**	**49.16**	**80.01**	**62.58**
0.999	29.81	27.17	69.52	48.62	79.04	61.28

Table 6. Analysis of $\boldsymbol{\alpha}$ **in Eq. (11)**. MobileNetV2 and mixer are adopted as query and gallery models, respectively.

Method	Two -stage	GLDv2-Test		$\mathcal{R}$Oxf + 1M		$\mathcal{R}$Par + 1M	
		Private	Public	Medium	Hard	Medium	Hard
CSD [56]	✓	25.51	23.73	64.42	43.90	68.32	47.76
Mixer + CSD	✓	27.23	24.73	67.08	45.50	76.69	57.74
Mixer + CSD	×	**28.73**	**26.28**	**69.66**	**46.82**	**78.41**	**60.15**
LCE [26]	✓	25.15	22.03	63.77	43.37	66.44	46.72
Mixer + LCE	✓	27.29	24.95	66.17	44.44	76.38	56.82
Mixer + LCE	×	**29.34**	**27.35**	**69.48**	**48.95**	**80.28**	**62.70**
Ours	✓	27.90	24.74	66.76	45.20	76.55	58.05
	×	**29.85**	**27.68**	**70.47**	**49.16**	**80.01**	**62.58**

Table 7. Comparison of **two-stage and joint training (asymmetric retrieval)**. Two-satge: mixer is first trained and kept fixed during the training of query model. MobileNetV2 and mixer are adopted as query and gallery models, respectively.

Method	Decoup.	GLDv2-Test		$\mathcal{R}$Oxf + 1M		$\mathcal{R}$Par + 1M	
		Private	Public	Medium	Hard	Medium	Hard
Mixer +	×	12.61	11.74	46.29	26.55	49.57	31.00
REG [3]	✓	**15.87**	**13.40**	**49.91**	**31.07**	**52.25**	**33.64**
Mixer +	×	25.97	23.60	63.30	43.75	73.25	52.95
LCE [26]	✓	**29.34**	**27.35**	**69.48**	**48.95**	**80.28**	**62.70**
Mixer +	×	18.13	16.42	48.31	29.49	47.13	29.94
CSD [56]	✓	**28.73**	**26.28**	**69.66**	**46.82**	**78.41**	**60.15**

Table 8. **Generalization analysis**. Decoup.: training processes of query model and mixer are decoupled with the momentum-updated mechanism. MobileNetV2 and mixer are adopted as query and gallery models, respectively.

	Method	Query Net ϕ_q	Gallery Net ϕ_g	Ext. (*ms*) ↓	Mem. (*GB*) ↓	GLDv2-Test		$\mathcal{R}$Oxf		$\mathcal{R}$Oxf + 1M		$\mathcal{R}$Par		$\mathcal{R}$Par + 1M	
						Private	Public	Medium	Hard	Medium	Hard	Medium	Hard	Medium	Hard
Previous asymmetric image retrieval methods	DELG[‡] [4]	R101	R101	113.7	7.6	26.14	24.20	76.31	55.60	63.71	37.45	86.61	72.38	70.59	46.94
	HVS [9]			16.5	7.6	23.65	20.52	73.85	53.31	58.33	33.81	85.14	71.10	65.44	42.87
	LCE [26]	MobV2	R101	16.8	7.6	23.88	20.45	73.95	52.78	57.76	32.45	84.76	70.78	66.22	43.03
	CSD [56]			16.5	7.6	23.33	21.33	74.72	53.70	60.19	34.14	85.70	71.46	68.25	44.24
	Token[‡] [54]	R101	R101	131.1	3.8	29.20	27.24	82.16	65.75	70.58	47.46	89.40	78.44	77.24	56.81
	HVS[†] [9]			16.3	3.8	25.87	23.51	73.16	53.19	57.68	35.52	84.27	71.43	62.90	41.00
	LCE[†] [26]	MobV2	R101	16.6	3.8	26.06	24.42	74.82	56.40	61.57	39.43	84.09	71.88	65.70	43.86
	CSD[†] [56]			16.3	3.8	**26.58**	**24.10**	75.52	56.83	63.46	39.01	84.50	70.73	65.93	43.77
	CVNet[‡] [21]	R101	R101	113.7	7.6	30.71	28.73	80.01	62.83	74.25	54.56	90.18	79.01	80.82	62.74
	HVS[†] [9]			16.5	7.6	26.50	23.49	74.90	55.17	62.32	42.66	84.92	71.62	67.13	46.80
	LCE[†] [26]	MobV2	R101	16.8	7.6	25.15	22.25	75.95	57.87	63.77	43.37	83.66	69.71	66.44	46.72
	CSD[†] [56]			16.5	7.6	25.51	23.73	**76.44**	**58.41**	**64.42**	**43.90**	85.32	71.61	68.32	47.76
	DOLG[‡] [59]	R101	R101	126.7	1.9	29.46	26.85	82.37	64.94	75.19	53.55	90.97	81.71	82.28	66.45
	HVS[†] [9]			16.1	1.9	24.99	22.03	72.79	54.20	63.29	41.74	85.18	70.72	68.13	48.25
	LCE[†] [26]	MobV2	R101	16.3	1.9	26.04	24.06	72.84	53.70	61.90	40.84	85.77	69.54	67.65	48.53
	CSD[†] [56]			16.1	1.9	25.64	24.04	75.53	56.23	64.02	42.79	**86.34**	**72.84**	**69.29**	**49.47**
Feature Fusion	Ensemble^{5,632}	-	-	485.3	20.9	32.28	29.79	82.86	66.06	75.30	53.24	90.74	81.72	83.35	67.05
	Ours^{512}	Mixer	Mixer	492.7	1.9	**32.93**	**31.63**	85.16	70.35	77.58	58.30	91.38	82.41	83.68	68.04
	Ours^{512}	MobV2	Mixer	16.1	1.9	28.57	26.57	79.46	**64.18**	68.17	47.27	90.44	79.64	77.90	59.01
	mAP gains over the previous asymmetric SOTA					(↑ 1.99)	(↑ 2.47)	(↑ 3.02)	(↑ **5.77**)	(↑ 3.75)	(↑ 3.37)	(↑ 4.10)	(↑ 6.80)	(↑ 8.61)	(↑ 9.54)
	Ours^{2,048}	Mixer	Mixer	493.5	7.6	32.85	31.27	**85.24**	**70.43**	**77.84**	**58.91**	**91.55**	**82.55**	**84.43**	**69.44**
	Ours^{2,048}	MobV2	Mixer	16.5	7.6	**29.85**	**27.68**	**80.19**	64.14	**70.47**	**49.16**	**90.48**	**80.55**	**80.01**	**62.58**
	mAP gains over the previous asymmetric SOTA					(↑ **3.27**)	(↑ **3.58**)	(↑ **3.75**)	(↑ 5.73)	(↑ **6.05**)	(↑ **5.26**)	(↑ **4.14**)	(↑ **7.71**)	(↑ **10.72**)	(↑ **13.11**)

Table 9. **Comparison to the state-of-the-art methods.** ^{d}: feature dimension is d; ↓: lower is better; ‡: re-evaluate official public weights; †: re-implementation; Ext.: latency of feature extraction for query side; Mem.: average memory footprint for $\mathcal{R}$1M dataset; MobV2: MobileNetv2; R101: ResNet101; Mixer: aggregate various features into a compact vector with ϕ^{mix}; Ensemble: feature concatenation.

our AFF is combined with various existing methods, joint training yields a consistent performance boost.

Generalizability analysis. In fact, our AFF is feasible to be combined with various existing methods, which leads to different training loss $\ell_{\text{comp.}}(\phi_q, x)$. For example, when combined with REG [3], we maintain a momentum-updated version $\widetilde{\phi}_{\text{mix}}$ of mixer ϕ_{mix}, whose output feature is denoted as $\widetilde{\boldsymbol{g}}^{\text{mix}}$. Then, $\ell_{\text{comp.}}(\phi_q, x)$ is formulated as $\ell_{\text{comp.}}(\phi_q, x) = -\left\|\boldsymbol{q} - \widetilde{\boldsymbol{g}}^{\text{mix}}\right\|_2^2$. More details about the combinations with existing methods, *e.g.*, CSD [56] and LCE [26], are provided in the supplementary materials. As shown in Tab. 8, our method enhances all existing asymmetric retrieval systems, leading to a promising accuracy improvement without adding any overhead to the query side. All the results demonstrate the generalizability of our method.

5.3. Comparison to the state-of-the-art methods

In Tab. 9, our method is compared with the state-of-the-art asymmetric methods (SOTA). Previous asymmetric retrieval methods, adopting a single feature on the gallery side, still suffer severe performance degradation especially in the large-scale case. Our method performs feature fusion at the gallery side, which significantly improves the accuracy of asymmetric retrieval, *e.g.*, the mAP on $\mathcal{R}$Oxf + 1M increases from 64.42 to 70.47. Besides, compared to direct feature ensemble, our method introduces no extra computation and storage overhead to the query side, which is friendly for various resource-constrained platforms, *e.g.*, the latency of feature extraction on the query side is comparable to the previous asymmetric retrieval method, *e.g.*, 16.5 *ms* for ours *vs*. 16.1 *ms* for previous SOTA.

6. Conclusion

In this work, we introduce a new asymmetric feature fusion paradigm to enhance the accuracy of existing asymmetric systems. It alleviates the dilemma of trade-off between retrieval efficiency and accuracy, caused by the limited capacity of lightweight models. The proposed paradigm deploys various models at the gallery side to extract features, which are further aggregated into compact embedding with a dynamic mixer for efficient retrieval. As for the query side, only a single lightweight model is deployed for feature extraction. The query model and the mixer are jointly trained to achieve feature compatibility. With the proposed framework, the accuracy of asymmetric retrieval is significantly boosted without introducing any additional computational and storage overhead to the query side. Experiments on several datasets demonstrate the wide applicability and excellent effectiveness of our paradigm.

Acknowledgements. This work was supported in part by the National Natural Science Foundation of China under Contract 62102128 and 62021001, and in part by the Fundamental Research Funds for the Central Universities under contract WK3490000007. It was also supported by the GPU cluster built by MCC Lab of Information Science and Technology Institution and the Supercomputing Center of the USTC.

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 5

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded up robust features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 404–417, 2006. 2

[3] Mateusz Budnik and Yannis Avrithis. Asymmetric metric learning for knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8228–8238, 2021. 1, 2, 7, 8

[4] Bingyi Cao, André Araujo, and Jack Sim. Unifying deep local and global features for image search. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 726–743, 2020. 2, 5, 6, 7, 8

[5] Ioana Croitoru, Simion-Vlad Bogolin, Marius Leordeanu, Hailin Jin, Andrew Zisserman, Samuel Albanie, and Yang Liu. Teachtext: Crossmodal generalized distillation for text-video retrieval. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 11583–11593, 2021. 3

[6] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 4, 5

[7] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops (CVPRW)*, pages 224–236, 2018. 2

[8] Rahul Duggal, Hao Zhou, Shuo Yang, Jun Fang, Yuanjun Xiong, and Wei Xia. Towards regression-free neural networks for diverse compute platforms. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 1, 3

[9] Rahul Duggal, Hao Zhou, Shuo Yang, Yuanjun Xiong, Wei Xia, Zhuowen Tu, and Stefano Soatto. Compatibility-aware heterogeneous visual search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10723–10732, 2021. 1, 2, 3, 5, 8

[10] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8092–8101, 2019. 2

[11] Tiezheng Ge, Kaiming He, Qifa Ke, and Jian Sun. Optimized product quantization for approximate nearest neighbor search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2946–2953, 2013. 2

[12] Kai Han, Yunhe Wang, Qi Tian, Jianyuan Guo, Chunjing Xu, and Chang Xu. Ghostnet: More features from cheap operations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1580–1589, 2020. 3

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3

[14] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 3

[15] Fan Hu, Aozhu Chen, Ziyue Wang, Fangming Zhou, Jianfeng Dong, and Xirong Li. Lightweight attentional feature fusion: A new baseline for text-to-video retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 3, 5, 6

[16] Hervé Jégou, Matthijs Douze, Jorge Sánchez, Patrick Pérez, and Cordelia Schmid. Aggregating local image descriptors into compact codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pages 1704–1716, 2011. 2

[17] Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 304–317, 2008. 1

[18] Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pages 117–128, 2011. 2

[19] Xiao Jin, Baoyun Peng, Yichao Wu, Yu Liu, Jiaheng Liu, Ding Liang, Junjie Yan, and Xiaolin Hu. Knowledge distillation via route constrained optimization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1345–1354, 2019. 7

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2012. 3

[21] Seongwon Lee, Hongje Seong, Suhyeon Lee, and Euntai Kim. Correlation verification for image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5374–5384, 2022. 2, 5, 6, 8

[22] Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Yifeng Lu, Denny Zhou, Quoc V. Le, Alan Yuille, and Mingxing Tan. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17182–17191, June 2022. 3

[23] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 10012–10022, 2021. 6

[24] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, pages 91–110, 2004. 2

[25] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. ShuffleNet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 116–131, 2018. 3, 5

[26] Qiang Meng, Chixiang Zhang, Xiaoqiang Xu, and Feng Zhou. Learning compatible embeddings. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 9939–9948, 2021. 1, 2, 3, 5, 7, 8

[27] Andrej Mikulik, Michal Perdoch, Ondřej Chum, and Jiří Matas. Learning vocabularies over a fine quantization. *International Journal of Computer Vision (IJCV)*, pages 163–175, 2013. 2

[28] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 5191–5198, 2020. 7

[29] Anastasiya Mishchuk, Dmytro Mishkin, Filip Radenović, and Jiri Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Conference and Workshop on Neural Information Processing Systems (NeurIPS)*, 2017. 2

[30] David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2161–2168, 2006. 1

[31] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3456–3465, 2017. 2

[32] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007. 2, 7

[33] Filip Radenović, Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondřej Chum. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5706–5715, 2018. 2, 5

[34] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning cnn image retrieval with no human annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pages 1655–1668, 2018. 1, 2, 5, 6

[35] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021. 3

[36] Vivek Ramanujan, Pavan Kumar Anasosalu Vasu, Ali Farhadi, Oncel Tuzel, and Hadi Pouransari. Forward compatible training for large-scale embedding retrieval systems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19386–19395, 2022. 3

[37] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3, 5

[38] Yantao Shen, Yuanjun Xiong, Wei Xia, and Stefano Soatto. Towards backward-compatible representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2, 5

[39] Nina Shvetsova, Brian Chen, Andrew Rouditchenko, Samuel Thomas, Brian Kingsbury, Rogerio S. Feris, David Harwath, James Glass, and Hilde Kuehne. Everything at once - multimodal fusion transformer for video retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20020–20029, 2022. 3

[40] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1470–1470, 2003. 1, 2

[41] Chull Hwan Song, Hye Joo Han, and Yannis Avrithis. All the attention you need: Global-local, spatial-channel attention for image retrieval. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2754–2763, 2022. 2

[42] Yuxin Song, Ruolin Zhu, Min Yang, and Dongliang He. DALG: Deep attentive local and global modeling for image retrieval. *arXiv preprint arXiv:2207.00287*, 2022. 2, 3

[43] Pavel Suma and Giorgos Tolias. Large-to-small image resolution asymmetry in deep metric learning. *arXiv preprint arXiv:2210.05463*, 2022. 1

[44] Fuwen Tan, Jiangbo Yuan, and Vicente Ordonez. Instance-level image retrieval using reranking transformers. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 12105–12115, 2021. 7

[45] Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning (ICML)*, pages 6105–6114, 2019. 3

[46] Yurun Tian, Xin Yu, Bin Fan, Fuchao Wu, Huub Heijnen, and Vassileios Balntas. SOSNet: Second order similarity regularization for local descriptor learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[47] Giorgos Tolias, Yannis Avrithis, and Hervé Jégou. To aggregate or not to aggregate: Selective match kernels for image search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1401–1408, 2013. 2, 6, 7

[48] Giorgos Tolias, Tomas Jenicek, and Ondřej Chum. Learning and aggregating deep local descriptors for instance-level recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 460–477, 2020. 2, 5, 7

[49] Giorgos Tolias, Ronan Sicre, and Hervé Jégou. Particular Object Retrieval With Integral Max-Pooling of CNN Activa-

tions. In *International Conference on Learning Representations (ICLR)*, 2016. 1, 2, 6

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5998–6008, 2017. 4, 5

[51] Timmy S. T. Wan, Jun-Cheng Chen, Tzer-Yi Wu, and Chu-Song Chen. Continual learning for visual search with backward consistent feature embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16702–16711, 2022. 1

[52] Philippe Weinzaepfel, Thomas Lucas, Diane Larlus, and Yannis Kalantidis. Learning super-features for image retrieval. In *International Conference on Learning Representations (ICLR)*, 2022. 2, 7

[53] Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. Google landmarks dataset v2 a large-scale benchmark for instance-level recognition and retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2575–2584, 2020. 1, 5

[54] Hui Wu, Min Wang, Wengang Zhou, Yang Hu, and Houqiang Li. Learning token-based representation for image retrieval. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 2703–2711, 2022. 2, 5, 6, 8

[55] Hui Wu, Min Wang, Wengang Zhou, and Houqiang Li. Learning deep local features with multiple dynamic attentions for large-scale image retrieval. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 11416–11425, 2021. 2

[56] Hui Wu, Min Wang, Wengang Zhou, Houqiang Li, and Qi Tian. Contextual similarity distillation for asymmetric image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9489–9498, 2022. 1, 2, 6, 7, 8

[57] Shengsen Wu, Liang Chen, Yihang Lou, Yan Bai, Tao Bai, Minghua Deng, and Ling-Yu Duan. Neighborhood consensus contrastive learning for backward-compatible representation. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 2722–2730, 2022. 1

[58] Sijie Yan, Yuanjun Xiong, Kaustav Kundu, Shuo Yang, Siqi Deng, Meng Wang, Wei Xia, and Stefano Soatto. Positive-congruent training: Towards regression-free model updates. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14299–14308, 2021. 3

[59] Min Yang, Dongliang He, Miao Fan, Baorong Shi, Xuetong Xue, Fu Li, Errui Ding, and Jizhou Huang. Dolg: Single-stage image retrieval with deep orthogonal fusion of local and global features. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 11772–11781, 2021. 2, 3, 5, 6, 8

[60] Yihan Zeng, Da Zhang, Chunwei Wang, Zhenwei Miao, Ting Liu, Xin Zhan, Dayang Hao, and Chao Ma. Lift: Learning 4d lidar image fusion transformer for 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17172–17181, June 2022. 3

[61] Binjie Zhang, Yixiao Ge, Yantao Shen, Yu Li, Chun Yuan, Xuyuan Xu, Yexin Wang, and Ying Shan. Hot-refresh model upgrades with regression-alleviating compatible training in image retrieval. In *International Conference on Learning Representations (ICLR)*, 2022. 3

[62] Binjie Zhang, Yixiao Ge, Yantao Shen, Shupeng Su, Fanzi Wu, Chun Yuan, Xuyuan Xu, Yexin Wang, and Ying Shan. Towards universal backward-compatible representation learning. In *Proceedings of the International Joint Conference on Artificial Intelligence, IJCAI*, pages 1615–1621, 2022. 1, 2

[63] Shaoting Zhang, Ming Yang, Timothee Cour, Kai Yu, and Dimitris N Metaxas. Query specific fusion for image retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 660–673, 2012. 3

[64] Shiliang Zhang, Ming Yang, Xiaoyu Wang, Yuanqing Lin, and Qi Tian. Semantic-aware co-indexing for image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1673–1680, 2013. 3

[65] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6848–6856, 2018. 3

[66] Liang Zheng, Shengjin Wang, Ziqiong Liu, and Qi Tian. Packing and padding: Coupled multi-index for accurate image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1939–1946, 2014. 3

[67] Liang Zheng, Shengjin Wang, Lu Tian, Fei He, Ziqiong Liu, and Qi Tian. Query-adaptive late fusion for image search and person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1741–1750, 2015. 3

[68] Liang Zheng, Shengjin Wang, Wengang Zhou, and Qi Tian. Bayes merging of multiple vocabularies for scalable image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1955–1962, 2014. 3