Pattern Recognition 93 (2019) 365–379

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

CycleMatch: A cycle-consistent embedding network for image-text matching

Yu Liu[a], Yanming Guo[b], Li Liu[b,c], Erwin M. Bakker[a], Michael S. Lew[a,*]

[a] Department of Computer Science, Leiden University, Leiden, 2333 CA, The Netherlands
[b] College of System Engineering, National University of Defense Technology, Changsha, Hunan 410073, China
[c] Center for Machine Vision and Signal Analysis, University of Oulu, Oulu 8000, Finland

ARTICLE INFO

Article history:
Received 20 September 2018
Revised 15 March 2019
Accepted 1 May 2019
Available online 4 May 2019

Keywords:
Image-text matching
Embedding
Deep neural networks
Late-fusion inference

ABSTRACT

In numerous multimedia and multi-modal tasks from image and video retrieval to zero-shot recognition to multimedia question and answering, bridging image and text representations plays an important and in some cases an indispensable role. To narrow the modality gap between vision and language, prior approaches attempt to discover their correlated semantics in a common feature space. However, these approaches omit the intra-modal semantic consistency when learning the inter-modal correlations. To address this problem, we propose cycle-consistent embeddings in a deep neural network for matching visual and textual representations. Our approach named as CycleMatch can maintain both inter-modal correlations and intra-modal consistency by cascading dual mappings and reconstructed mappings in a cyclic fashion. Moreover, in order to achieve a robust inference, we propose to employ two late-fusion approaches: average fusion and adaptive fusion. Both of them can effectively integrate the matching scores of different embedding features, without increasing the network complexity and training time. In the experiments on cross-modal retrieval, we demonstrate comprehensive results to verify the effectiveness of the proposed approach. Our approach achieves state-of-the-art performance on two well-known multi-modal datasets, Flickr30K and MSCOCO.

1. Introduction

Nowadays, the explosive growth of multimedia data in social networks (e.g. image, video, text and audio) has triggered a massive amount of research activities in multi-modal understanding and reasoning. For instance, we can recognize a picture of a panda after hearing the description “black and white bears” without ever having seen one. This example demonstrates the cross-modal interaction between vision and language. These heterogeneous data offers us the opportunity to understand the world from diverse perspectives, while giving rise to the challenges of bridging different modalities. In this paper, we focus on the task of image-text matching, which aims to incorporate heterogeneous representations from visual and textual modalities. In practice, this task plays an essential role for a wide variety of tasks in the multimedia research, for examples, cross-modal retrieval [1,2], visual question answering [3], zero-shot recognition [4] and video captioning [5].

The core issue with image-text matching is searching for an appropriate embedding space where related images and texts can be matched correctly. Driven by the great strides made by deep learning [6–8], recent research has been dedicated to exploring deep neural networks for learning powerful embedding features, in order to narrow the modality gap between visual and textual domains. These networks are typically composed of two branches for generating visual and textual embedding features in a common latent space, respectively [9–13]. Then, a similarity-based ranking loss is used to measure the latent embedding features. Latent embeddings can distill common semantic information about both the visual content and textual description. To directly match the similarities between vision and language, researchers further exploit dual embeddings by translating an input feature in the source space to be the feature in the target space [14–17]. Both the latent and dual embeddings can capture inter-modal semantic correlations, however, they are limited in preserving intra-modal semantic consistency. Our motivation for this work is that: *A robust embedding method should be able to learn representations of both the source and target modalities.*

Inspired by the idea of cycle-consistent learning [18,19], we propose cycle-consistent embeddings in an image-text matching network, which can incorporate both *inter-modal correlations* and *intra-modal consistency* for learning robust visual and textual em-

* Corresponding author.
E-mail address: m.s.k.lew@liacs.leidenuniv.nl (M.S. Lew).

https://doi.org/10.1016/j.patcog.2019.05.008
0031-3203/

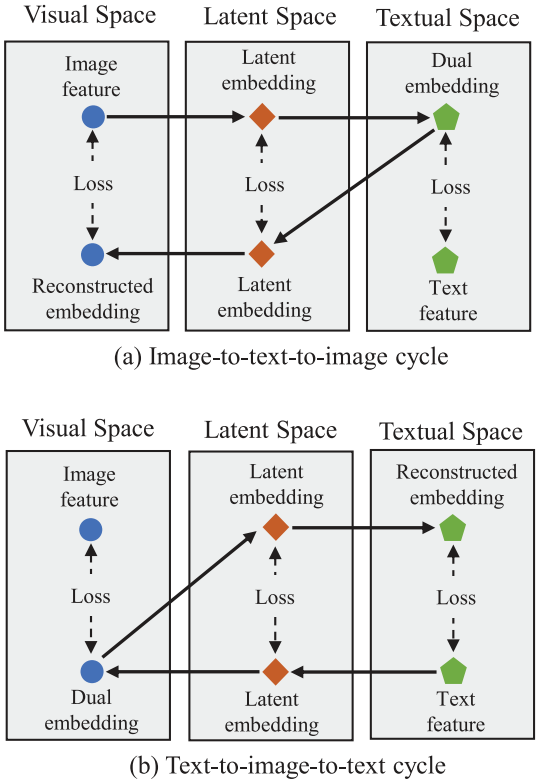


Fig. 1. Schematic pipeline of the proposed cycle-consistent embedding method. It is composed of two cycle branches: (a) image-to-text-to-image cycle and (b) text-to-image-to-text cycle. We first perform a dual mapping by transforming the input feature into the target feature space. Then, a reconstructed mapping is used to generate a reconstructed embedding in the source feature space. Moreover, we construct a latent space to correlate latent embeddings of the two mappings. The two branches share the mapping functions for transformations between three feature spaces, and can be trained jointly by optimizing the matching losses in the three feature spaces.

beddings. Fig. 1 illustrates our embedding method by integrating three feature embeddings, including dual, reconstructed and latent embeddings. Specifically, it has two cycle branches, one starting from an image feature in the visual space and the other from a text feature in the textual space. For each branch, it first accomplishes a dual mapping by translating an input feature in the source space to be a dual embedding in the target space. Inverse to the dual mapping, we then exploit a reconstructed mapping, with the aim of translating the dual embedding back to the source space. Moreover, we learn a latent space during the dual and reconstructed mappings and correlate the latent embeddings. In the three feature spaces, we compute their ranking losses to jointly optimize the whole embedding learning. Consequently, our visual-textual embedding method can learn not only *inter-modal mappings* (i.e. image-to-text and text-to-image), but also *intra-modal mappings* (i.e. image-to-image and text-to-text).

The contributions of this work are summarized as follows:

- We propose a novel deep cycle-consistent embedding network for image-text matching. Our approach called CycleMatch can cascade dual and reconstructed mappings together to maintain inter-modal correlations and intra-modal consistency. To our best knowledge, this is the first work to explore the usage of cycle consistency for solving the task of image-text matching.
- To improve the inference at the test stage, we present two late-fusion approaches to efficiently integrate the matching scores of multiple embedding features without increasing the training complexity.
- In the experiments, our cycle-consistency embedding outperforms traditional embeddings with considerable improvements for cross-modal retrieval on two multi-modal datasets, i.e. Flickr30K and MSCOCO. In addition, our results are competitive with the state-of-the-art performance on both datasets.

The rest of this paper is structured as follows. Related works are introduced in Section 2. Section 3 presents the details regarding the proposed CycleMatch. The late-fusion inference approaches are shown in Section 4. The experimental results are reported in Section 6. Finally, Section 7 summarizes the conclusions and discusses the future work.

2. Related work

Our work is related to image-text matching, deep visual-textual embedding and cycle-consistent learning.

2.1. Image-text matching

The problem of image-text matching has been studied by the multimedia community for decades. One typical solution is to unify heterogeneous representations into a latent embedding space, and then measure their similarity to ensure related pairs are more similar than unrelated ones. To be specific, Canonical Correlation Analysis (CCA) [20] is a classical and important embedding method, which can learn linear transformations to project two modalities into a latent space where their correlation is maximized. In addition, many variants [21–24] are proposed to leverage the effectiveness of CCA. For example, kernel CCA [21] extended the classical linear CCA by learning non-linear transformations. Moreover, Gong et al. [25] integrated a third view with the two-view CCA using high-level image semantics, in order to gain a better separation for multi-modal data. Ranjan et al. [26] proposed a multi-label CCA approach by introducing multi-label information while learning the cross-modal subspaces. In practice, the integration of images and texts is a core issue for a variety of multi-modal applications [3,4,27,28]. For example, Karaoglu et al. [29] proposed to detect words from images and then to combine the textual cues with the visual ones. Their method showed promising performance improvements for both place classification and logo retrieval. Similarly, Bai et al. [30] developed a unified and end-to-end trainable network, where the attention mechanism was further incorporated to better match the extracted textual and visual cues, to address the difficulties in fine-grained image classification.

2.2. Deep visual-textual embedding

With the increasing progress of deep learning, research efforts have been made to CCA into deep neural networks [26,31–33]. However, most deep CCA models rely on expensive decorrelation computations, which limit their generalization abilities at large-scale data. Alternatively, a number of recent approaches [12,13,34–37] address the task by designing two-branch networks to embed visual and textual features into a common latent space, and then learn latent embeddings by optimizing a ranking loss between matched and unmatched image-text pairs. For instance, Wang et al. [9] built a simple and efficient matching network to preserve the structure relations between images and texts in the latent space. To associate image regions with words, the attention mechanism was integrated into visual-textual embedding models [10,11]. In addition to the pairwise ranking loss, recent approaches [38,39] leveraged extra loss functions to enhance the discrimination of the learned embedding features.

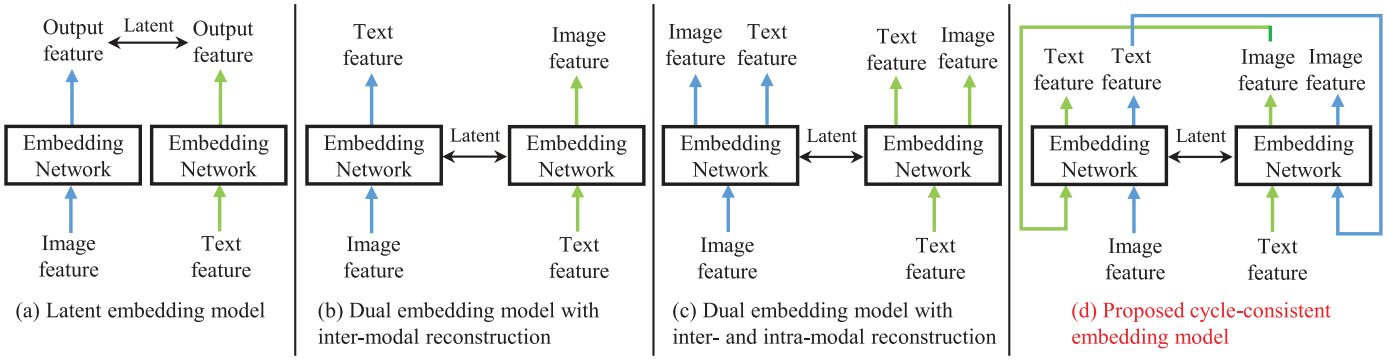


Fig. 2. Conceptual illustration of variants of image-text matching models. (a) Latent embedding model. (b) Dual embedding model with inter-modal reconstruction. (c) Dual embedding model with inter-modal and intra-modal reconstruction. Note that each embedding network consists of two branches to output the image feature and text feature, separately. (d) Our cycle-consistent embedding model. The models in (b), (c), (d) also impose latent embeddings on hidden layers. Our model cascades the two embedding networks in a cyclic fashion, which can enhance interactions between two embedding networks.

Another line of research [14,15,40–42] focuses on learning dual embeddings between two modalities, e.g. projecting visual features into the textual feature space and vice versa. Essentially, the dual embedding models are motivated by autoencoders. For instance, Feng et al. [14] proposed a correspondence cross-modal autoencoder model. 2WayNet [16] built the projections between two modalities and regularized them with Euclidean loss. Recently, the work of Gu et al. [17] utilized two generative models to synthesize grounded visual and textual representations. Also, Huang et al. [43] jointly modeled image-sentence matching and sentence generation. Note that, latent embeddings can be additionally used in the dual embedding models to enhance cross-modal relations.

In contrast to the above studies, our approach builds a reconstructed mapping upon the dual mapping, and generates cycle-consistent embeddings that are beneficial to the process of matching visual-textual representations. In Fig. 2, we show the differences of our model from previous works.

2.3. Cycle-consistent learning

There are a few papers exploring cycle consistency for diverse applications [18,19,44–46]. They are mainly motivated by the fact that, cycle-consistent learning is encouraged to produce additional feedback signals to improve the bi-directional translations. Specifically, He et al. [18] proposed a dual-learning mechanism based on deep reinforcement learning, where one agent was used to learn the primal task, e.g. English-to-French translation, and the other agent for the dual task, e.g. French-to-English translation. More recently, Zhu et al. [19] exploited cycle-consistent adversarial networks (CycleGAN), which combined a cycle-consistency loss with an adversarial loss [47] to perform unpaired image-to-image translations between two different visual domains. A similar idea was also presented in [48,49]. Inspired by CycleGAN, several recent works have transferred the cycle-consistency loss to many supervised tasks [50–52].

Although prior works have shown the effectiveness of using cycle-consistent constraints for intra-modal domain mappings, yet in the context of cross-modal representation learning, its effectiveness has not been well investigated. In contrast to prior approaches that utilize cycle-consistent constraints within one modality (e.g. neural machine translation and image-to-image translation), our work is the first to extend the usage of cycle consistency for learning visual-textual embeddings. The work of Chen and Zitnick [53] is relevant to ours, as their model can both generate textual captions and reconstruct visual features given an image representation. However, their model lacks the inverse cycle mapping, i.e. text-to-image-to-text, which can be jointly learned in our model. Last but not least, these existing works did not consider matching latent embeddings during the cycle-consistent scheme.

3. Proposed cycle-consistent embeddings

In this section, we present the proposed CycleMatch model with cycle-consistent embeddings for matching visual and textual representations.

3.1. System architecture

Fig. 3 depicts an overview of the CycleMatch architecture. The entire network consists of three components: feature encoder, feature embedding and feature matching. First of all, given an input image I_i and text T_i, we employ individual feature encoders to extract the visual feature $\boldsymbol{v}_i = En_{\text{img}}(I_i)$ and textual feature $\boldsymbol{t}_i = En_{\text{text}}(T_i)$. Then, we develop several fully-connected (FC) layers (i.e. $FC_{\text{I2T}}^{(j)}$) to perform the *Image-to-Text* (I2T) mapping and several other FC layers (i.e. $FC_{\text{T2I}}^{(j)}$) for the *Text-to-Image* (T2I) mapping. Let $f_{\text{I2T}}(\cdot)$ and $f_{\text{T2I}}(\cdot)$ represent the mapping functions for I2T and T2I, respectively. In addition, connecting FC_{I2T} and FC_{T2I} can form two cycle mappings between the visual and textual feature spaces. Specifically, given $\boldsymbol{v}_i$, we first transform it to be $f_{\text{I2T}}(\boldsymbol{v}_i)$ in the textual feature space and then learn its reconstructed feature $f_{\text{T2I}}(f_{\text{I2T}}(\boldsymbol{v}_i))$ in the visual feature space. Moreover, we also correlate intermediate features derived from $FC_{\text{I2T}}^{(3)}$ and $FC_{\text{T2I}}^{(3)}$, so as to learn a latent feature space. Similarly, $\boldsymbol{t}_i$ is used to start another cycle mapping. In a nutshell, each cycle mapping can learn dual, reconstructed and latent embeddings in a cyclic fashion.

3.2. Formulation

Next, we will detail the above three embeddings and formulate their loss functions separately. The entire network contains two cycle-consistent embedding branches: one for *image-to-text-to-image* (I2T2I) mapping and the other for *text-to-image-to-text* (T2I2T) mapping. Here, we take the I2T2I mapping for an example.

3.2.1. Dual embedding

In a dataset collection with N image-text pairs, the input $\boldsymbol{v}_i$ is fed into the first layer $FC_{\text{I2T}}^{(1)}$, where $i = 1, \ldots, N$. By using the following layers $FC_{\text{I2T}}^{(j)}(j = 2, 3, 4)$, the network finally generates a dual embedding $f_{\text{I2T}}(\boldsymbol{v}_i)$ in the textual space, which has the same dimension as the ground-truth textual feature $\boldsymbol{t}_i$. Then, we normalize the two features and compute their similarity using the

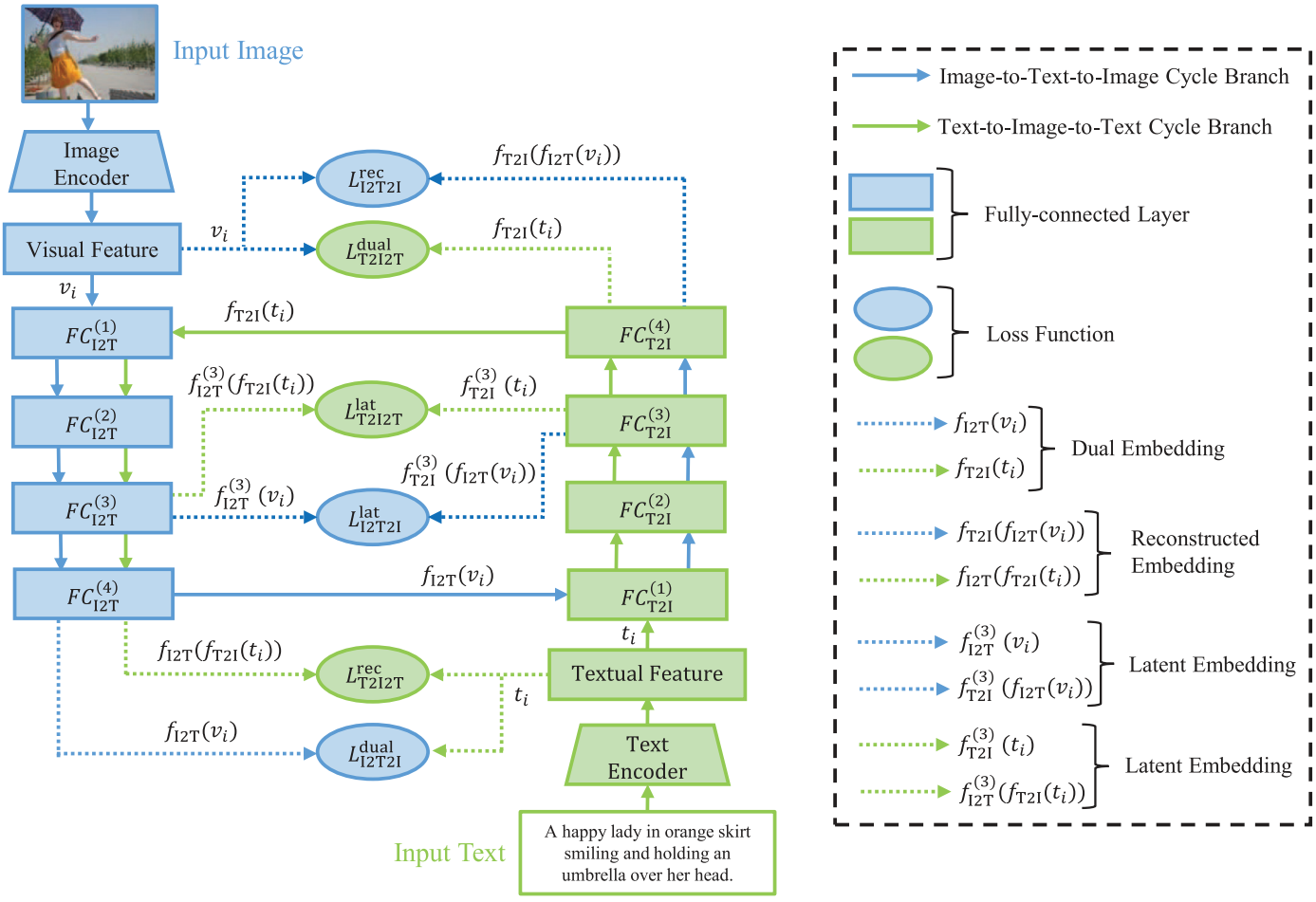


Fig. 3. Overview of the proposed CycleMatch. It develops two cycle branches for visual-textual embeddings. For each branch, it is divided into two sub-branches from the *fourth* FC layer (i.e. $FC_{\mathrm{IT}}^{(4)}$ and $FC_{\mathrm{TI}}^{(4)}$). One sub-branch continues accomplishing the dual mapping to the target feature space, while the other sub-branch is used to perform the reconstructed mapping back to the source feature space. In this way, the cycle branches allow to jointly learn dual, reconstructed and latent embedding features. We can train the network end-to-end by optimizing several ranking loss functions simultaneously.

cosine distance

$$s(f_{\mathrm{I2T}}(\boldsymbol{v}_i), \boldsymbol{t}_i) = \frac{f_{\mathrm{I2T}}(\boldsymbol{v}_i) \cdot \boldsymbol{t}_i}{||f_{\mathrm{I2T}}(\boldsymbol{v}_i)|| \cdot ||\boldsymbol{t}_i||}. \tag{1}$$

Notably, larger scores indicate more similar samples. During training, it is important to construct a number of negative pairs, in addition to the positive pair. Thereby, we search for the top K negative samples in a mini-batch for both $f_{\mathrm{I2T}}(\boldsymbol{v}_i)$ and $\boldsymbol{t}_i$, which are denoted with $f_{\mathrm{I2T}}(\boldsymbol{v}_{i,k}^-)$ and $\boldsymbol{t}_{i,k}^-$, respectively, where $k = 1, \ldots, K$. To learn dual mappings, we need to employ a pairwise ranking loss function with respect to positive and negative pairs:

$$L_{\mathrm{I2T2I}}^{\mathrm{dual}} = \sum_{i=1}^{N}\sum_{k=1}^{K}\Big\{\max\left[0, m - s(f_{\mathrm{I2T}}(\boldsymbol{v}_i), \boldsymbol{t}_i) + s(f_{\mathrm{I2T}}(\boldsymbol{v}_i), \boldsymbol{t}_{i,k}^-)\right] + \alpha \max\left[0, m - s(f_{\mathrm{I2T}}(\boldsymbol{v}_i), \boldsymbol{t}_i) + s(f_{\mathrm{I2T}}(\boldsymbol{v}_{i,k}^-), \boldsymbol{t}_i)\right]\Big\}, \tag{2}$$

where m is a margin parameter that defines a threshold to constrain the positive and negative pairs. α adjusts the weights of the two loss terms. Ideally, the matched distance $s(f_{\mathrm{I2T}}(\boldsymbol{v}_i), \boldsymbol{t}_i)$ should be smaller than any of the unmatched distances $s(f_{\mathrm{I2T}}(\boldsymbol{v}_i), \boldsymbol{t}_{i,k}^-)$ and $s(f_{\mathrm{I2T}}(\boldsymbol{v}_{i,k}^-), \boldsymbol{t}_i)$.

3.2.2. Reconstructed embedding

Despite the fact that the task in this work is about cross-modal matching, it is important as well to ensure intra-modal consistency, that is, related images (or texts) should have closer distances than unrelated ones. Hence, we explore reconstructed mappings to maintain the intra-modal semantic consistency, in addition to learning inter-modal correlations with dual mappings. We cascade the dual and reconstructed mappings to form an intra-modal autoencoder and minimize the reconstruction error based on the ranking loss instead of the traditional Euclidean loss. Specifically, we feed $f_{\mathrm{I2T}}(\boldsymbol{v}_i)$ into $FC_{\mathrm{T2I}}^{(j)}$, to produce a reconstructed embedding feature $\tilde{\boldsymbol{v}}_i$ in the visual feature space with

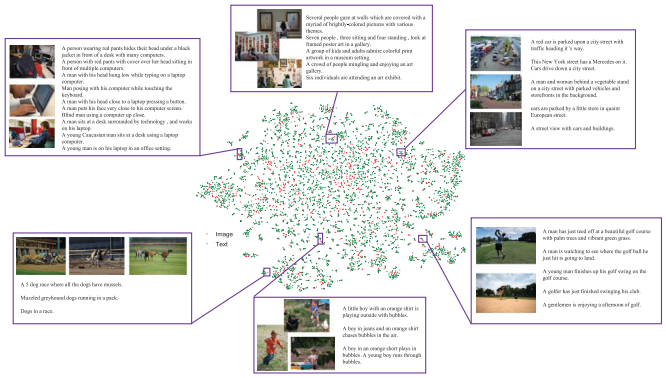
$$\tilde{\boldsymbol{v}}_i = f_{\mathrm{T2I}}(f_{\mathrm{I2T}}(\boldsymbol{v}_i)) = f_{\mathrm{T2I}} \circ f_{\mathrm{I2T}}(\boldsymbol{v}_i). \tag{3}$$

The ranking loss for making the reconstructed embedding feature $\tilde{\boldsymbol{v}}_i$ match with the original visual feature $\boldsymbol{v}_i$ can be written as follows

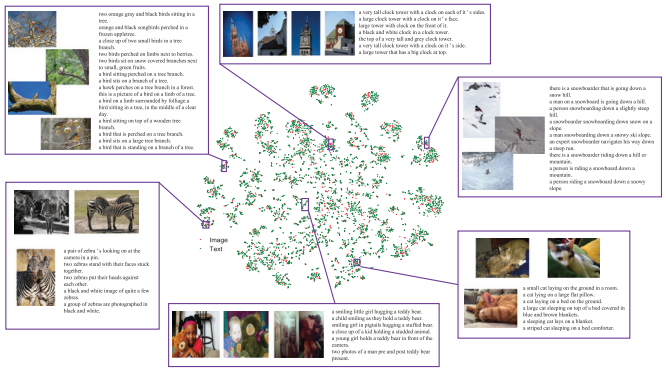
$$L_{\mathrm{I2T2I}}^{\mathrm{rec}} = \sum_{i=1}^{N}\sum_{k=1}^{K}\Big\{\max\left[0, m - s(\tilde{\boldsymbol{v}}_i, \boldsymbol{v}_i) + s(\tilde{\boldsymbol{v}}_i, \boldsymbol{v}_{i,k}^-)\right] + \alpha \max\left[0, m - s(\tilde{\boldsymbol{v}}_i, \boldsymbol{v}_i) + s(\tilde{\boldsymbol{v}}_{i,k}^-, \boldsymbol{v}_i)\right]\Big\}. \tag{4}$$

Since $L_{\mathrm{I2T2I}}^{\mathrm{rec}}$ also has an effect on the parameters of $FC_{\mathrm{I2T}}^{(j)}$, the reconstructed mappings can help to improve the learning of dual mappings as well.

Moreover, we employ the t-SNE algorithm [54] to visualize our embedding features. Fig. 4 shows the embedding maps with the test data from Flickr30K and MSCOCO, respectively. We show some original images and texts corresponding to the embedding features. First, the images and texts in each local window demonstrate high



(a) Flickr30K



(b) MSCOCO

Fig. 4. Visualization of our embedding features. For each dataset, we pick 1000 images (red) and 5000 texts (green). Some images and texts corresponding to the embedding features are shown in local windows, from which we can observe not only correlations between cross-modal samples, but also relations between intra-modal samples. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

semantic correlations. In addition, these images themselves have similar visual content, and these texts themselves contain related descriptions. This observation is consistent with our motivation that a robust embedding method should be able to consider both inter-modal correlations and intra-modal consistency.

3.2.3. Latent embedding

Furthermore, we exploit a latent feature space to enhance the correlations between the dual and reconstructed mappings. Latent embeddings are able to distill common semantic information from visual and textual representations. Specifically, we make use of the intermediate representations from the third FC layers, i.e. $FC^{(3)}_{\text{I2T}}$ and $FC^{(3)}_{\text{T2I}}$. When $\boldsymbol{v_i}$ passes through $FC^{(3)}_{\text{I2T}}$, we can extract an intermediate feature $f^{(3)}_{\text{I2T}}(\boldsymbol{v}_i)$. Also, the dual embedding $f_{\text{I2T}}(\boldsymbol{v}_i)$ passes through $FC^{(3)}_{\text{T2I}}$ to generate another intermediate feature $f^{(3)}_{\text{T2I}}(f_{\text{I2T}}(\boldsymbol{v}_i))$. The ranking loss for matching latent embeddings thereby becomes

$$\begin{aligned} L^{\text{lat}}_{\text{I2T2I}} = \sum_{i=1}^{N}\sum_{k=1}^{K}\Big\{ & \max\Big[0, m - s\big(f^{(3)}_{\text{I2T}}(\boldsymbol{v}_i), f^{(3)}_{\text{T2I}}(f_{\text{I2T}}(\boldsymbol{v}_i))\big) \\ & + s\big(f^{(3)}_{\text{I2T}}(\boldsymbol{v}_i), f^{(3)}_{\text{T2I}}(f_{\text{I2T}}(\boldsymbol{v}^-_{i,k}))\big)\Big] \\ & + \alpha \max\Big[0, m - s\big(f^{(3)}_{\text{I2T}}(\boldsymbol{v}_i), f^{(3)}_{\text{T2I}}(f_{\text{I2T}}(\boldsymbol{v}_i))\big) \\ & + s\big(f^{(3)}_{\text{I2T}}(\boldsymbol{v}^-_{i,k}), f^{(3)}_{\text{T2I}}(f_{\text{I2T}}(\boldsymbol{v}_i))\big)\Big]\Big\}. \end{aligned} \tag{5}$$

Similar to the above I2T2I branch, it is straightforward to express the matching losses in the T2I2T branch, including $L^{\text{dual}}_{\text{T2I2T}}$, $L^{\text{rec}}_{\text{T2I2T}}$ and $L^{\text{lat}}_{\text{T2I2T}}$. In Fig. 5, we show the six loss functions for learning cycle-consistent embeddings.

3.2.4. Full objective

During training, we need to incorporate all the loss functions jointly. The full objective is to minimize the total loss:

$$\arg\min_{W_{\text{I2T}}, W_{\text{T2I}}} L_{\text{total}} = L^{\text{dual}}_{\text{I2T2I}} + L^{\text{rec}}_{\text{I2T2I}} + L^{\text{lat}}_{\text{I2T2I}} + L^{\text{dual}}_{\text{T2I2T}} + L^{\text{rec}}_{\text{T2I2T}} + L^{\text{lat}}_{\text{T2I2T}}, \tag{6}$$

where W_{I2T} and W_{T2I} indicate the parameters in $FC^{(j)}_{\text{I2T}}$ and $FC^{(j)}_{\text{T2I}}$, respectively. They are unshared due to the specialization of two different modalities.

To demonstrate the effectiveness of our CycleMatch, we utilize the t-SNE [54] algorithm to visualize the embedding features learned in the visual, textual and latent feature spaces, separately. As shown in Fig. 6, we randomly select 100 image-text pairs from the Flickr30K dataset [55]. From all the feature maps, we can visibly observe high similarities between two matched samples.

4. Late-fusion inference

By performing cycle-consistent embeddings, we can represent one sample with a set of three different features, for instance, $\{\boldsymbol{v}_i, f_{\text{I2T}}(\boldsymbol{v}_i), f^{(3)}_{\text{I2T}}(\boldsymbol{v}_i)\}$ for an image. Since the reconstructed embedding $\tilde{\boldsymbol{v}}_i$ and the other latent embedding $f^{(3)}_{\text{T2I}}(f_{\text{I2T}}(\boldsymbol{v}_i))$ are related to $\boldsymbol{v}_i$ and $f^{(3)}_{\text{I2T}}(\boldsymbol{v}_i)$, we do not consider them for simplicity. Each of the three features can be used to measure an image-text matching score. Instead of using only one score, it is encouraged to leverage different scores together to achieve a more robust inference. This is driven by the late-fusion technique [56] in multimedia retrieval, which is a simple and efficient approach to combine the prediction scores of individual features. In this work, we present two effective late-fusion approaches, namely average fusion and adaptive fusion.

4.1. Average fusion

Given a query image I_q, we extract three features $\{\boldsymbol{v}_q, f_{\text{I2T}}(\boldsymbol{v}_q), f^{(3)}_{\text{I2T}}(\boldsymbol{v}_q)\}$. Similarly, an arbitrary text T_i in the dataset can be described with $\{\boldsymbol{t}_i, f_{\text{T2I}}(\boldsymbol{t}_i), f^{(3)}_{\text{T2I}}(\boldsymbol{t}_i)\}$. We can compute three similarity scores between I_q and T_i:

$$\begin{cases} \text{visual score}: s^{(1)}(\boldsymbol{v}_q, \boldsymbol{t}_i) = s(\boldsymbol{v}_q, f_{\text{T2I}}(\boldsymbol{t}_i)), \\ \text{textual score}: s^{(2)}(\boldsymbol{v}_q, \boldsymbol{t}_i) = s(f_{\text{I2T}}(\boldsymbol{v}_q), \boldsymbol{t}_i), \\ \text{latent score}: s^{(3)}(\boldsymbol{v}_q, \boldsymbol{t}_i) = s(f^{(3)}_{\text{I2T}}(\boldsymbol{v}_q), f^{(3)}_{\text{T2I}}(\boldsymbol{t}_i)). \end{cases} \tag{7}$$

Then we combine the three scores to obtain an average fusion score as follows

$$s^{avg}(\boldsymbol{v}_q, \boldsymbol{t}_i) = \frac{\sum_{j=1}^{3} s^{(j)}(\boldsymbol{v}_q, \boldsymbol{t}_i)}{3}. \tag{8}$$

It is similar to compute the fusion score $s^{avg}(\boldsymbol{t}_q, \boldsymbol{v}_i)$ in terms of a query text T_q.

4.2. Adaptive fusion

To study the importance of different features, we further learn adaptive weights when combining the three scores. As suggested in [57], the score curve by using a superior feature can be sorted in an "L" shape, while the curve by using an inferior feature tends to gradually descend. In addition, the area under the curve can be used as an indicator to measure the weight of the corresponding feature. Driven by this observation, we can use the sorted score curves of the above three features to decide their weights. Specifically, we utilize each of the three features to compute the score

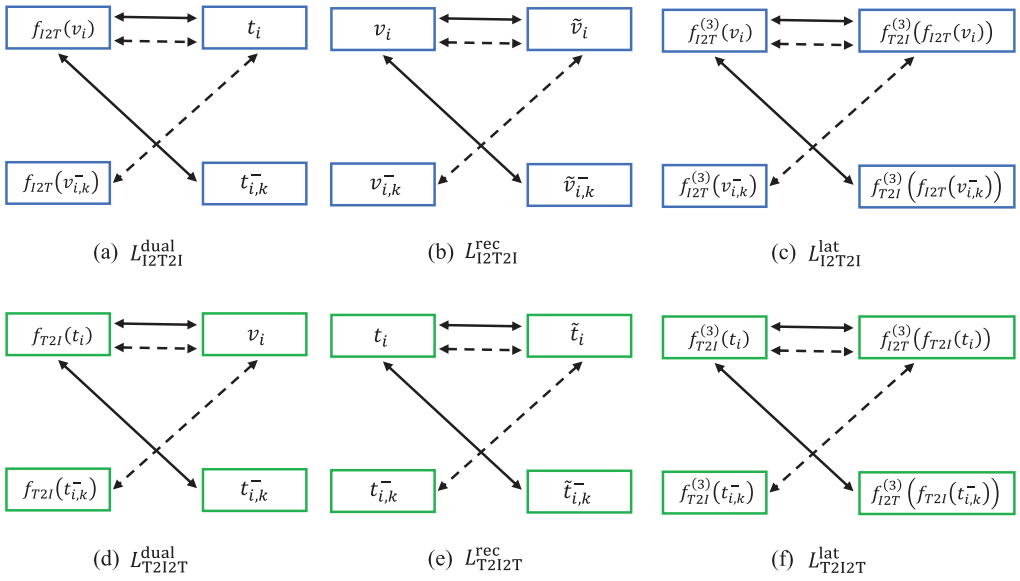


Fig. 5. Conceptual illustration of loss functions for training CycleMatch. The first row includes the loss functions in the I2T2I cycle and the second row is for the T2I2T cycle.

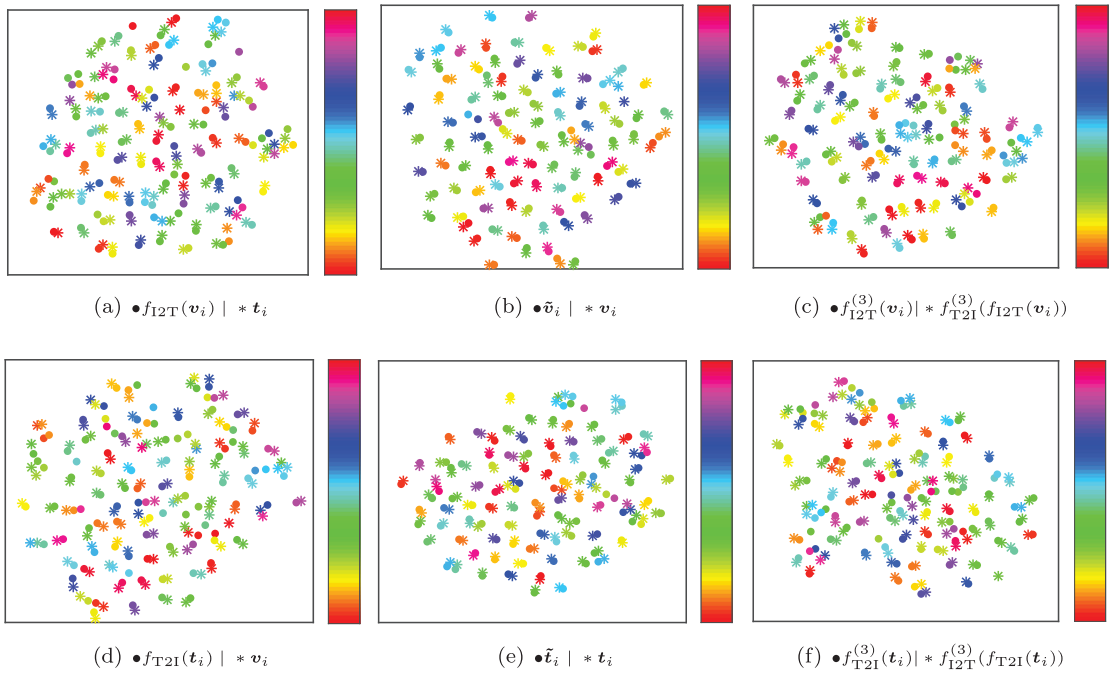
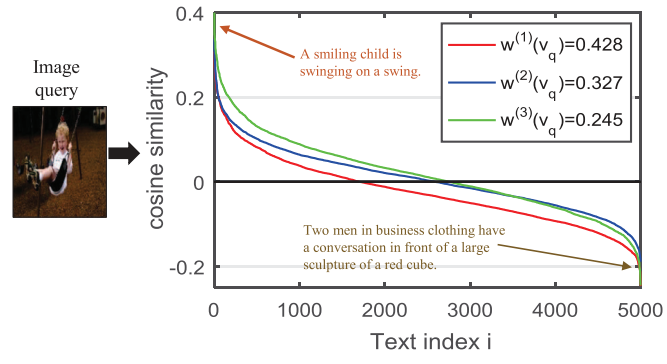


Fig. 6. Visualization of our embedding features by using 100 image-text pairs in Flickr30K [55]. The first and second rows represent the embedding features learned in the I2T2I and T2I2T branches respectively. In each feature map, matched samples are shown with the same color. In (a), (d), the dual embedding features ('•') can match with the corresponding target features ('*'); In (b)(e), the reconstructed embedding features ('•') look closely similar to the source features ('*'). In (c)(f), the two latent embedding features ('•' and '*') can learn to correlate with each other as well.

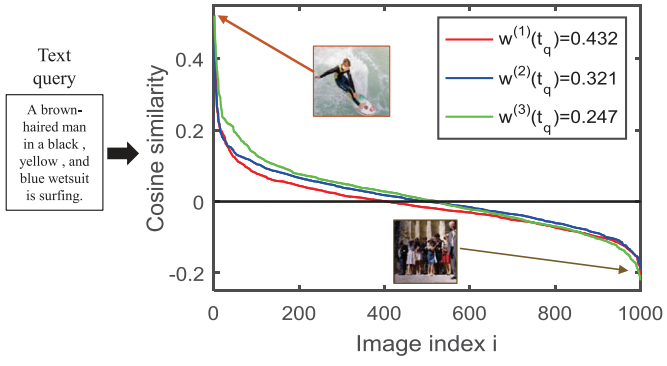
curve of a query image I_q to all the text samples. Then, we sort the score curves and compute their areas with respect to the horizontal axis. In Fig. 7, we show three sorted score curves for either a query image or text.

Our adaptive fusion method is inspired by the late fusion in [57], due to its parameter-free property and efficient computation. However, our method has two major differences from [57]. First, Zheng et al. [57] attempt to integrate different features, including BoW, Color and GIST features. In contrast, we construct a unified network to extract multiple embedding features, which have close relations to each other. Second, In [57], they use the total curve to compute the area. However, we compute only the positive area above the axis and omit the negative one.[1] This way can help to decrease the effect of long tails of the curves. For example in Fig. 7 (b), the three curves have almost similar negative areas, based on which it is hard to distinguish the weights of the three features. Hence, adding the negative area with the positive one will narrow the gap of significance of different features and fail to learn robust adaptive weights. In the experiments, we show the advantage of our method over [57].

[1] The similarity scores in this work are based on the cosine distance, ranging from −1 to 1.



(a) Image-to-text retrieval



(b) Image-to-text retrieval

Fig. 7. Illustration of the sorted score curves based on three different features. (a) For the query image, the first curve (in red) forms the smallest area above the X axis, so the corresponding feature (i.e. visual embedding feature) can have the largest weight (0.428). We show a matched text at the beginning of the curves and an unmatched text at the end of the curves. (b) Similarly, we demonstrate a text query example. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Formally, the positive area associated with the jth feature can be approximated by

$$area_{+}^{(j)}(\boldsymbol{v}_q) = \sum_{i=1}^{N} \max\left[0, s^{(j)}(\boldsymbol{v}_q, \boldsymbol{t}_i)\right]. \tag{9}$$

Smaller positive area means that the corresponding feature should have greater weights. Hence, the adaptive weights of I_q *w.r.t.* the three features can be expressed with

$$w^{(j)}(\boldsymbol{v}_q) = \frac{1}{area_{+}^{(j)}(\boldsymbol{v}_q)}. \tag{10}$$

In addition, we normalize the three weights to make sure $\sum_{j=1}^{3} w^{(j)}(\boldsymbol{v}_q) = 1$. Finally, the adaptive fusion score for matching I_q and T_i becomes

$$s^{adt}(\boldsymbol{v}_q, \boldsymbol{t}_i) = \sum_{j} w^{(j)}(\boldsymbol{v}_q) \cdot s^{(j)}(\boldsymbol{v}_q, \boldsymbol{t}_i). \tag{11}$$

Likewise, we demonstrate a text query T_q in the right of Fig. 7, and show its adaptive weights, $w^{(j)}(\boldsymbol{t}_q)$. Notice that our adaptive fusion approach can achieve specific weights for different query samples. It is an unsupervised and efficient manner without adding extra parameters and manual tuning. In the experiments, we analyze the effects of these two late-fusion approaches on the inference of cross-modal retrieval.

5. Discussion

Although the cycle-consistent idea has been adopted in many problems, it should not decrease the novelty of our work. In this section, we mainly aim to state our similarities and differences compared to the prior works like CycleGAN.

Similarities: Essentially, cycle-consistent learning is a variant of the auto-encoder model, which mainly aims to construct a cyclic mapping to reconstruct the input data. Both CycleGAN and CycleMatch are motivated by the idea of cycle-consistent learning, even though they focus on addressing different tasks.

Differences: Our proposed CycleMatch uses the idea of cycle-consistent learning, but it still has task-specific novelties and differences from CycleGAN.

- First, CycleGAN integrates a cycle-consistency loss with an adversarial loss to perform intra-modal representation learning, i.e. image-to-image translation between two image sets. In contrast, our CycleMatch is proposed to address the problem of cross-modal representation learning between image and text sets. In prior works, the effectiveness of cycle-consistent learning has not been well investigated in the context of cross-modal tasks. Our work is the first to extend cycle-consistent learning to address the task of image-text matching.
- Second, our reconstructed embedding is learned with the ranking loss, instead of the traditional Euclidean loss in CycleGAN. Notably, the ranking loss aims to reconstruct the relations among data samples rather than the original features. We find that the ranking loss is more suited for the matching task compared to the Euclidean loss.
- Third, CycleMatch is a novel network architecture that is different from CycleGAN. Notably, CycleMatch is not based on the GAN model. In addition, we consider the latent embedding representations, which are not taken into account in CycleGAN.
- Lastly, we contribute to proposing late-fusion inference in order to integrate multiple embedding features learned in the model. This robust and efficient inference is performed in the test stage and will not complicate the training procedure. The results in our experiments verify the effectiveness of the late-fusion inference. However, CycleGAN does not provide a robust inference in its test stage.

In summary, more and more papers [50–52,58,59] are making use of cycle-consistent learning to solve a variety of problems, such as domain adaptation, video retargeting and zero-shot learning. These works make promising contributions to the field, even though they are primarily or partially inspired by CycleGAN. In the future, we believe more related works will be encouraged in the field.

6. Experiments

First, we compare CycleMatch with various baseline models to verify its effectiveness. In addition, we present in-depth analysis on the two late-fusion approaches. Moreover, our results can be competitive with the state-of-the-art performance for cross-modal retrieval on two well-known datasets. Finally, we present additional ablation study on the effect of feature encoders and variance of test splits.

6.1. Experimental setup

We introduce the dataset protocols, evaluation metrics, network details, training details and time complexity, involved in our experimental setup.

6.1.1. Dataset protocols

The experiments are performed on two well-known datasets. (1) Flickr30K [55] consists of 31,783 images and each image is associated with five different sentences. We use the dataset split of [60], namely 29,783 training images, 1000 validation images and 1000 test images. (2) MSCOCO [61] is one of the largest multi-modal datasets, which includes 82,783 training images and 40,504 validation images. We pick five ground-truth sentences for each image. 1,000 test images are selected from the validation set [60]. Notice that some works [9,17,38] merge the remaining validation images into the training set, to further increase the performance. However, we keep only using the original training set for fairness.

6.1.2. Evaluation metrics

For evaluating the performance of cross-modal retrieval, we adopt the common metric R@K, which measures the recall rate of a correctly retrieved ground-truth at top K retrieved candidates. Generally, K is set to 1, 5 and 10 for both image-to-text and text-to-image retrieval.

6.1.3. Network details

In terms of the image encoder, we employed the powerful ResNet-152 [8] pre-trained on the ImageNet dataset [62]. Besides, we recast the CNN model to its fully convolutional network (FCN) counterpart, which can capture rich region representations. The last layer of the FCN model is spatially averaged to generate a 2048 dimensional visual representation. To extract the textual representation, we utilized the pre-trained RNN encoder proposed in [63]. It can represent one sentence with a 4096 dimensional feature vector. Currently, we did not fine-tune the feature encoders during the training.

As for the two groups of four FC layers in CycleMatch (i.e. $FC^{(j)}_{\mathrm{I2T}}$ and $FC^{(j)}_{\mathrm{T2I}}$), the channels of the first three layers are fixed as [2048,512,512]. Note that, $FC^{(4)}_{\mathrm{I2T}}$ should have the same dimension as the textual feature and $FC^{(4)}_{\mathrm{T2I}}$ should be equal to the size of the visual feature.

6.1.4. Training details

We implemented the proposed approach based on the Caffe library [64]. It is important to shuffle the training samples randomly during the data preparation stage. The hyper-parameters are evaluated on the validation set of each dataset. We trained the model using SGD with a mini-batch size of 500, a weight decay of 0.0005, a momentum of 0.9 and an initialized learning rate of 0.1. The learning rate is divided by 10 when the decrease in loss stabilizes. We set $\alpha = 2$ and $m = 0.1$ in all the experiments. The number of negative samples in each min-batch is 50. The whole training procedure terminates after 60 epochs for both datasets.

In Fig. 8, we show the training loss of the six loss functions on the two datasets. It can be observed that the loss tend to converge during the training epochs.

6.1.5. Time complexity

We use the total loss in Eq. (6) to perform the training procedure. Each loss term is a simple and efficient ranking loss that is widely used in retrieval tasks. We used a Titan X card with 12GB to train all models in the experiments. For the full CycleMatch model, training required about 19 h on the Flickr30K dataset and 47 h on the MSCOCO dataset, respectively.

6.2. Comparisons with baseline approaches

To demonstrate the superiority of our approach, we designed several baseline models (see Table 1) based on the same network settings and training hyper-parameters as CycleMatch. In terms of inference, LatentMatch is evaluated with only the latent score. However, all the other models have both visual and textual scores. For consistency we utilize the average fusion approach to accomplish their inference. Table 2 reports the cross-modal retrieval performance of these models on both Flickr30K and MSCOCO. Overall, CycleMatch surpasses LatentMatch and DualMatch with significant improvements, and achieves overall superior performance over other variants of CycleMatch. In the next, we can report the results from several aspects.

6.2.1. Impact of reconstructed embeddings

First, we explain the benefit of constructing the cycle-consistent embeddings in our model. Primarily, cycle-consistent learning used in our model can benefit the dual embeddings. As can be seen in Fig. 3, the reconstructed embeddings are built on top of the dual embeddings, therefore the reconstruction loss can help the training of the dual embeddings. The main difference between DualMatch and CycleMatch(w/o latent) is that the latter model introduces a reconstructed mapping upon the traditional dual mapping. As reported in Table 2, the performance gap shows between DualMatch and CycleMatch(w/o latent) verifies the benefit of adding reconstructed embeddings in a cyclic fashion.

In addition to the above quantitative evaluation, we show image-to-text retrieval results as well to qualitatively compare the two methods. As shown in Fig. 9, CycleMatch (w/o latent) can retrieve more accurate text descriptions than DualMatch, given the same query image. According to both quantitative and qualitative comparisons, it shows the improvements achieved by adding the cycle-consistency in our model.

6.2.2. Impact of latent embeddings

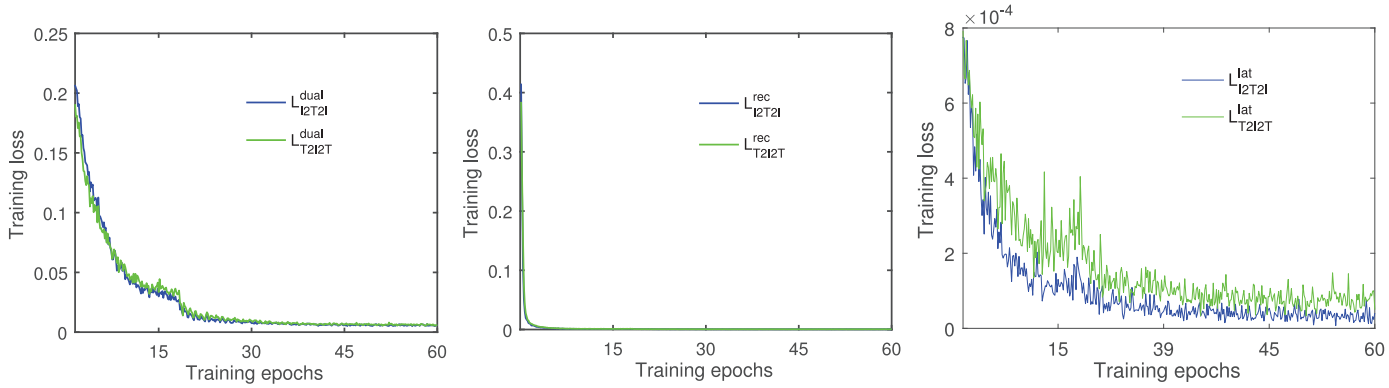
By comparing the results of CycleMatch and CycleMatch(w/o latent), we find that integrating the latent embeddings into CycleMatch brings further improvements over all metrics. For example, R@5 shows about 2% gains for both I → T and T → I. Although using only latent embeddings (i.e. LatentMatch) is inferior to other models, it is beneficial to adopt them to improve other embedding methods like CycleMatch.

Moreover, we conduct an experiment below to test the effect of using different fully-connected (FC) layers on the latent embeddings. Apart from using the layer $FC^{(3)}$, we also test the latent embedding based on $FC^{(1)}$ or $FC^{(2)}$. In Table 3, we show the results by using three different FC layers. It can be seen that the both $FC^{(2)}$ and $FC^{(3)}$ features show better results than $FC^{(1)}$. Although $FC^{(1)}$ feature has more dimensions, its representation power is less than $FC^{(2)}$ and $FC^{(3)}$. One main reason is that $FC^{(1)}$ is the first layer in the network, but $FC^{(2)}$ and $FC^{(3)}$ are closer to the high-level semantics. In addition, $FC^{(3)}$ has slight improvements over $FC^{(2)}$. Based on these results, we decide to construct the latent embedding with the $FC^{(3)}$ features.

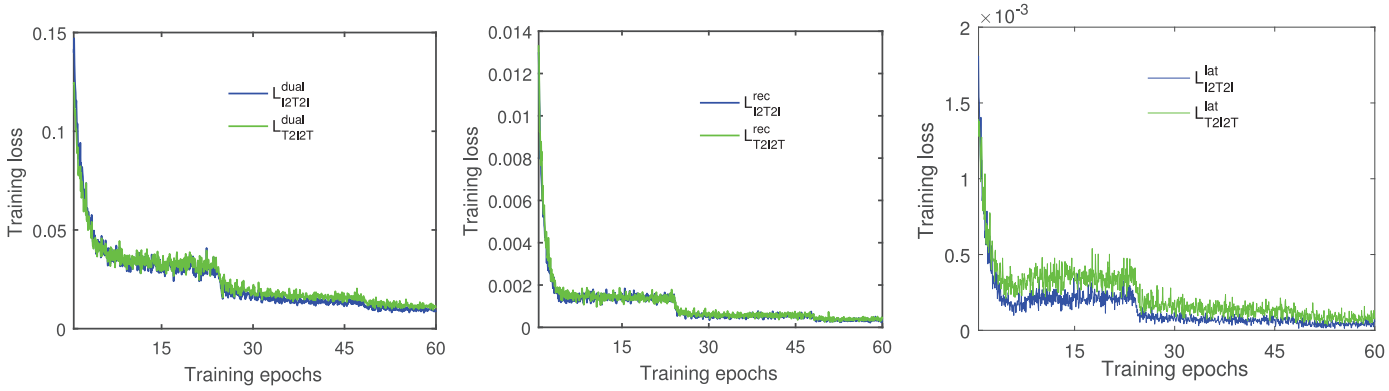
6.2.3. Impact of cycle branches

Both CycleMatch(I2T2I) and CycleMatch(T2I2T) can outperform LatentMatch and DualMatch, even though only one cycle-consistent embedding branch is used. By comparing these two models, CycleMatch(I2T2I) performs better for I → T retrieval, while CycleMatch(T2I2T) yields better results for T → I retrieval. When we incorporate the two cycle branches jointly, namely CycleMatch, it achieves overall superior performance over any single cycle branch on both datasets.

In addition to the R@K performance, we further present the matching scores computed by using our embedding features. To be specific, we randomly select 100 image-text pairs from the test set, and compute the similarity between one image and text. As shown in Fig. 10, matched image-text pairs (with the same index) have greater similarity scores than unmatched ones.



(a) on the Flickr30K dataset



(b) on the MSCOCO dataset

Fig. 8. Illustration of training loss cost during training CycleMatch on the Flickr30K and MSCOCO datasets.

Table 1
Summary of various embedding methods for image-text matching.

Embedding methods	Main description
LatentMatch	A latent embedding model by matching $f^{(3)}_{I2T}(\boldsymbol{v}_i)$ and $f^{(3)}_{T2I}(\boldsymbol{t}_i)$.
DualMatch	A dual embedding model by learning two dual mappings: I→T and T→I.
CycleMatch(w/o latent)	An ablation model without latent embeddings between dual and reconstructed mappings.
CycleMatch(I2T2I)	An ablation model with an I2T2I cycle branch and an I→T dual mapping.
CycleMatch(T2I2T)	An ablation model with a T2I2T cycle branch and a T→I dual mapping.
CycleMatch	A fully implemented model with two cycle branches.

Table 2
Comparison of different embedding approaches for cross-modal retrieval on Flickr30k and MSCOCO. Higher R@K numbers are better, where $K = 1, 5, 10$. The full CycleMatch method outperforms others on both datasets.

Method	Flickr30K dataset						MSCOCO dataset					
	Image to Text			Text to Image			Image to Text			Text to Image		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
LatentMatch	49.7	77.4	85.0	37.8	69.8	80.6	53.9	82.9	90.8	43.0	75.8	85.9
DualMatch	53.4	80.5	87.1	40.1	70.9	81.0	56.3	83.5	91.5	45.5	76.7	87.5
CycleMatch(w/o latent)	56.8	81.7	90.3	41.1	72.5	81.3	58.5	84.0	92.4	46.9	78.3	88.7
CycleMatch(I2T2I)	57.0	82.4	**91.0**	42.4	73.6	82.0	**61.1**	85.5	93.1	46.3	79.3	89.0
CycleMatch(T2I2T)	56.4	81.9	90.6	**43.2**	74.3	82.6	59.7	84.7	92.6	**47.6**	79.7	89.6
CycleMatch	**57.8**	**83.3**	90.9	**43.2**	**74.8**	**83.8**	60.5	**86.3**	**93.7**	47.2	**80.3**	**90.4**

6.3. Analysis of late-fusion inference

Recall that CycleMatch contains visual, textual and latent scores for inference (Section 4). In this experiment, we compare three strategies to study the effect of two late-fusion inference approaches on the retrieval performance of CycleMatch. Specifically, the one-score strategy uses only a single visual score; the two-score strategy integrates visual and textual scores together; the three-score strategy combines all three scores by further adding the latent score. Table 4 reports the results of the three strategies. For the two-score and three-score strategies, we present the results of using the average and adaptive fusion, respectively. From the results, we can make the following observations:

Fig. 9. Image-to-Text retrieval results on the datasets, (a) Flickr30K and (b) MSCOCO. The ground-truth descriptions are in green. By comparison, CycleMatch (w/o latent) achieves more accurate results than DualMatch. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3
Evaluation on the effect of using different fully-connected layers on the latent embedding. The two-score adaptive fusion is used here. By comparison, fc(3) is the best one for learning the latent embedding on most measurements.

Method	Flickr30K dataset						MSCOCO dataset					
	Image to Text			Text to Image			Image to Text			Text to Image		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
latent embedding with $FC^{(1)}$	57.2	82.1	90.6	42.1	73.5	83.3	59.5	85.0	93.4	46.9	79.3	89.4
latent embedding with $FC^{(2)}$	58.2	83.3	91.9	43.3	74.9	84.3	60.7	86.4	94.0	47.5	80.5	90.4
latent embedding with $FC^{(3)}$	58.6	83.6	91.6	43.6	75.3	84.2	61.1	86.8	94.2	47.9	80.9	90.9

(1) The two-score strategy improves the one-score counterpart with 1%–3% gains. As the visual and textual scores match the samples in two different feature spaces, their complementary scores are able to improve the inference quality.
(2) The adaptive fusion outperforms the average one in terms of both two-score and three-score strategies. Although their performance gap over the R@K measurements is not significant, the adaptive fusion is an efficient method without imposing extra parameters and manual tuning. In addition, the inference time of the adaptive fusion is close to that of the average fusion.
(3) The three-score strategy fails to achieve further improvements over the two-score one. We attribute this to the fact that, the latent score measures the similarity between

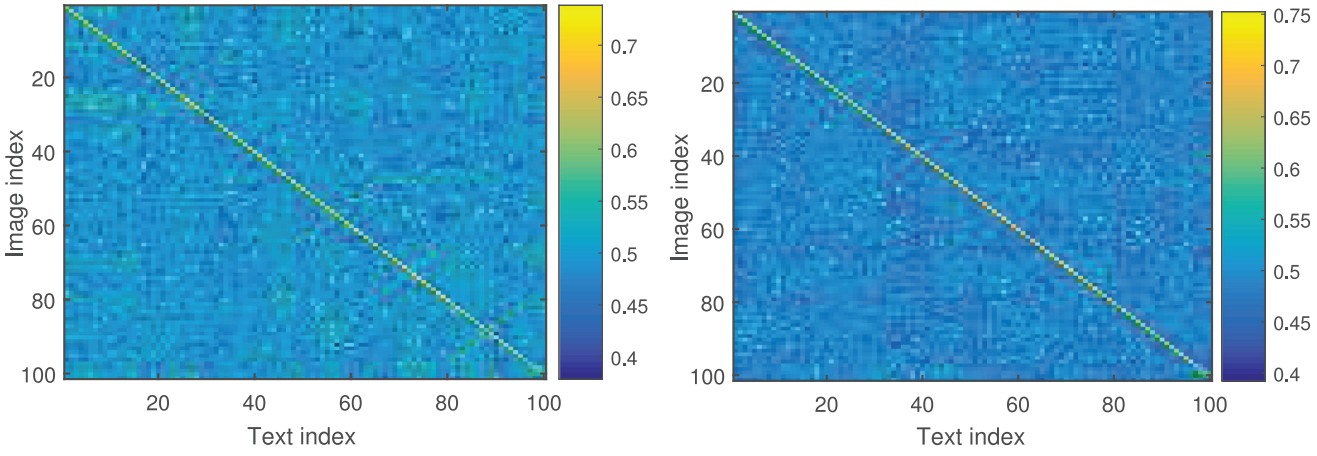


Fig. 10. Similarity matrix of 100 image-text pairs from the test set. The related images and texts have the same index numbers. The diagonal line demonstrates high inter-modal correlations for matched image-text pairs. The original cosine scores are re-scaled to be [0,1].

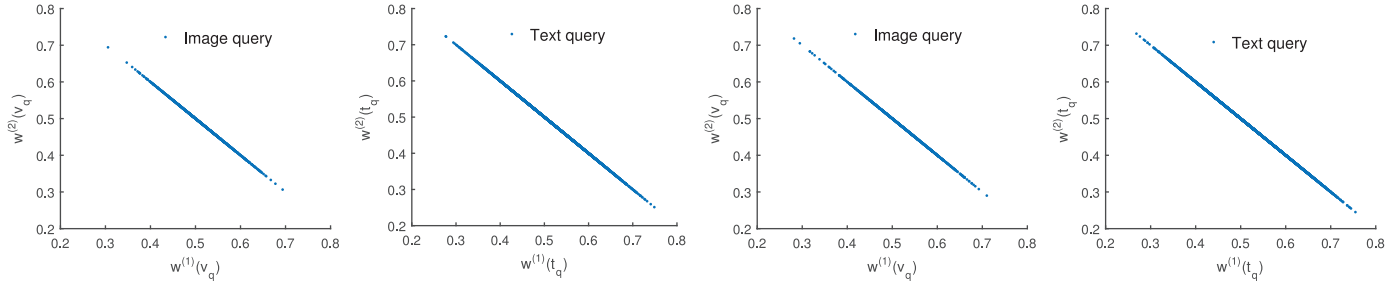


Fig. 11. Visualization of adaptive weights for 1000 image queries and 5000 text queries on Flickr30K(a, b) and MSCOCO (c, d). Each dot in the maps is a query sample, having two weights for the adaptive fusion. Note that $w^{(1)}(\cdot)+w^{(2)}(\cdot)=1$. The weights of query samples are mostly gathered between 0.4 and 0.6. It suggests that both visual and textual scores play an important role in the inference results.

Table 4
Evaluation on the effect of different inference strategies on the R@K measurements. The two-score strategy based on the adaptive fusion achieves the best results (in bold face).

Inference method	Flickr30K dataset						MSCOCO dataset					
	Image to Text			Text to Image			Image to Text			Text to Image		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
One-score, without fusion	54.8	82.6	90.1	40.1	70.9	81.0	58.6	85.5	92.6	45.5	78.3	88.7
Two-score, average fusion	57.8	83.3	90.9	43.2	74.8	83.8	60.5	86.3	93.7	47.2	80.3	90.4
Two-score, adaptive fusion	**58.6**	83.6	**91.6**	**43.6**	**75.3**	**84.2**	**61.1**	**86.8**	94.2	**47.9**	80.9	**90.9**
Three-score, average fusion	57.4	83.5	91.0	43.2	74.7	83.9	59.7	86.0	94.0	46.9	80.6	89.8
Three-score, adaptive fusion	57.8	**83.8**	91.2	43.5	74.7	84.0	61.0	86.4	**94.5**	47.8	**81.0**	90.7

$f^{(3)}_{\text{I2T}}(\boldsymbol{v}_i)$ and $f^{(3)}_{\text{T2I}}(\boldsymbol{t}_i)$. However, we do not use a direct matching loss between them during training CycleMatch. Although adding this latent score for inference will not bring further performance gains, learning the latent embeddings in CycleMatch is still important for improving the entire embedding procedure. As we discussed earlier, CycleMatch performs better than the variant without latent embeddings, namely CycleMatch(w/o latent).

As we can see, the two-score adaptive fusion achieves the best results. In Fig. 11, we further present and analyze the two adaptive weights (i.e. $w^{(1)}(\cdot)$ and $w^{(2)}(\cdot)$), which are learned in the two-score adaptive fusion for visual and textual scores. Fig. 11(a,b) and (c,d) shows the weights for Flickr30K and MSCOCO, respectively. For I2T retrieval, we illustrate the adaptive weights of 1000 image queries, namely $w^{(1)}(v_q)$ and $w^{(2)}(v_q)$; for T2I retrieval, we show all the weights of 5000 text queries, denoted as $w^{(1)}(t_q)$ and $w^{(2)}(t_q)$. Notice that, each dot in Fig. 11 represents a query sample that learns individual weights based on its score curves. It can be seen that most samples have weights ranging from 0.4 to 0.6, which suggests that both visual and textual scores have an important impact on the inference results.

Comparison with the late fusion in [57]. This experiment is used to compare the results of our adaptive fusion and the one in [57]. Recall that our method computes only the positive area above the axis, while the method in [57] considers both positive and negative areas. As reported in Table 5 below, the results with only a positive area are better in the context of both two-score and three-score fusion cases, even though the performance gap between the two methods is slight.

Table 5
Comparison of two different methods for computing adaptive-fusion weights. The method by using only a positive area are better that of using both positive and negative areas.

Inference method	Flickr30K dataset						MSCOCO dataset					
	Image to Text			Text to Image			Image to Text			Text to Image		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Two-score, positive&negative areas	58.1	83.3	91.2	43.2	75.0	83.8	60.7	86.3	93.8	47.4	80.5	90.6
Two-score, positive area	58.6	83.6	91.6	43.6	75.3	84.2	61.1	86.8	94.2	47.9	80.9	90.9
Three-score, positive&negative areas	57.5	83.6	91.0	43.3	74.7	83.8	60.5	86.0	94.2	47.3	80.6	90.4
Three-score, positive area	57.8	83.8	91.2	43.5	74.7	84.0	61.0	86.4	94.5	47.8	81.0	90.7

Table 6
Comparison with the state-of-the-art approaches on Flickr30K. In addition, we present the image and text encoders used in these approaches. Our CycleMatch (the two-score adaptive fusion) achieves better results on R@K measurements (in boldface).

Method	Image encoder	Text encoder	Image to Text			Text to Image		
			R@1	R@5	R@10	R@1	R@5	R@10
DCCA [32]	AlexNet	TF-IDF	16.7	39.3	52.9	12.6	31.0	43.0
DVSA [34]	AlexNet	RNN	22.2	48.2	61.4	15.2	37.7	50.5
UVSE [36]	VGG-19	RNN	23.0	50.7	62.9	16.8	42.0	56.5
mCNN [35]	VGG-19	CNN	33.6	64.1	74.9	26.2	56.3	69.6
VQA-aware [65]	VGG-19	RNN	33.9	62.5	74.5	24.9	52.6	64.8
GMM-FV [33]	VGG-16	GMM+HGLMM	35.0	62.0	73.8	25.0	52.7	66.0
m-RNN [60]	VGG-16	RNN	35.4	63.8	73.7	22.8	50.7	63.1
RNN-FV [66]	VGG-19	RNN	35.6	62.5	74.2	27.4	55.9	70.0
HM-LSTM [13]	AlexNet	RNN	38.1	–	76.5	27.7	–	68.8
DSPE [9]	VGG-19	HGLMM	40.3	68.9	79.9	29.7	60.1	72.1
sm-LSTM [11]	VGG-19	RNN	42.5	71.9	81.5	30.2	60.4	72.3
VSE++ [67]	ResNet-152	RNN	43.7	–	82.1	32.2	–	72.1
DualCNN [38]	ResNet-152	ResNet-152	44.2	70.2	79.7	30.7	59.2	70.8
RRF-Net [12]	ResNet-152	HGLMM	47.6	77.4	87.1	35.4	68.3	79.9
2WayNet [16]	VGG-16	GMM+HGLMM	49.8	67.5	–	36.0	55.6	–
DAN [10]	ResNet-152	RNN	55.0	81.8	89.0	39.4	69.2	79.1
CycleMatch (Ours)	ResNet-152	RNN	**58.6**	**83.6**	**91.6**	**43.6**	**75.3**	**84.2**

Table 7
Comparison with the state-of-the-art approaches on MSCOCO. In addition, we present the image and text encoders used in these approaches. Our CycleMatch (the two-score adaptive fusion) outperforms other approaches by achieving promising results (in boldface).

Method	Image encoder	Text encoder	Image to Text			Text to Image		
			R@1	R@5	R@10	R@1	R@5	R@10
STV [68]	VGG-19	RNN	33.8	67.7	82.1	25.9	60.0	74.6
DVSA [34]	AlexNet	RNN	38.4	69.9	80.5	27.4	60.2	74.8
GMM-FV [33]	VGG-16	GMM+HGLMM	39.4	67.9	80.9	25.1	59.8	76.6
m-RNN [60]	VGG-16	RNN	41.0	73.0	83.5	29.0	42.2	77.0
RNN-FV [66]	VGG-19	RNN	41.5	72.0	82.9	29.2	64.7	80.4
BiLSTM-Max [63]	ResNet-101	RNN	42.6	75.3	87.3	33.9	69.7	83.8
mCNN [35]	VGG-19	CNN	42.8	73.1	84.1	32.6	68.6	82.8
UVSE [36]	VGG-19	RNN	43.4	75.7	85.8	31.0	66.7	79.9
HM-LSTM [13]	AlexNet	RNN	43.9	–	87.8	36.1	–	86.7
order-embeddings [69]	VGG-19	RNN	46.7	–	88.9	37.9	–	85.9
DSPE [9]	VGG-19	HGLMM	50.1	79.7	89.2	39.6	75.2	86.9
VQA-aware [65]	VGG-19	RNN	50.5	80.1	89.7	37.0	70.9	82.9
DualCNN [38]	ResNet-50	ResNet-50	52.2	80.4	88.7	37.2	69.5	80.6
sm-LSTM [11]	VGG-19	RNN	53.2	83.1	91.5	40.7	75.8	87.4
2WayNet [16]	VGG-16	GMM+HGLMM	55.8	75.2	–	39.7	63.3	–
RRF-Net [12]	ResNet-152	HGLMM	56.4	85.3	91.5	43.9	78.1	88.6
VSE++ [67]	ResNet-152	RNN	58.3	–	93.3	43.6	–	87.8
CycleMatch (Ours)	ResNet-152	RNN	**61.1**	**86.8**	**94.2**	**47.9**	**80.9**	**90.9**

6.4. Comparisons with state-of-the-art approaches

In Tables 6 and 7, we present a comprehensive comparison with previous papers where they reported the cross-modal retrieval performance on Flickr30K and MSCOCO. It can be seen that our CycleMatch (the two-score adaptive fusion) outperforms recent state-of-the-art approaches [10,12,67] with promising improvements on both datasets. It is worth noting that these approaches employ different feature encoders that have a significant influence on the performance. For a clear comparison, we further list the image and text encoders used in these approaches. In the following experiments, we will study the effect of different feature encoders on the performance of CycleMatch.

To boost the performance, recent several approaches [17,38,43,67] further fine-tune the image encoders during training their models. Their results with fine-tuning the image encoders achieve better performance on MSCOCO than Flickr30K. We should know that it is feasible to fine-tune the image encoders while training our CycleMatch, which can help to further improve our results. In addition, the fine-tuning process will maintain the findings we mentioned as above. More importantly, our results on the Flickr30K dataset can even compete with the fine-tuned results in

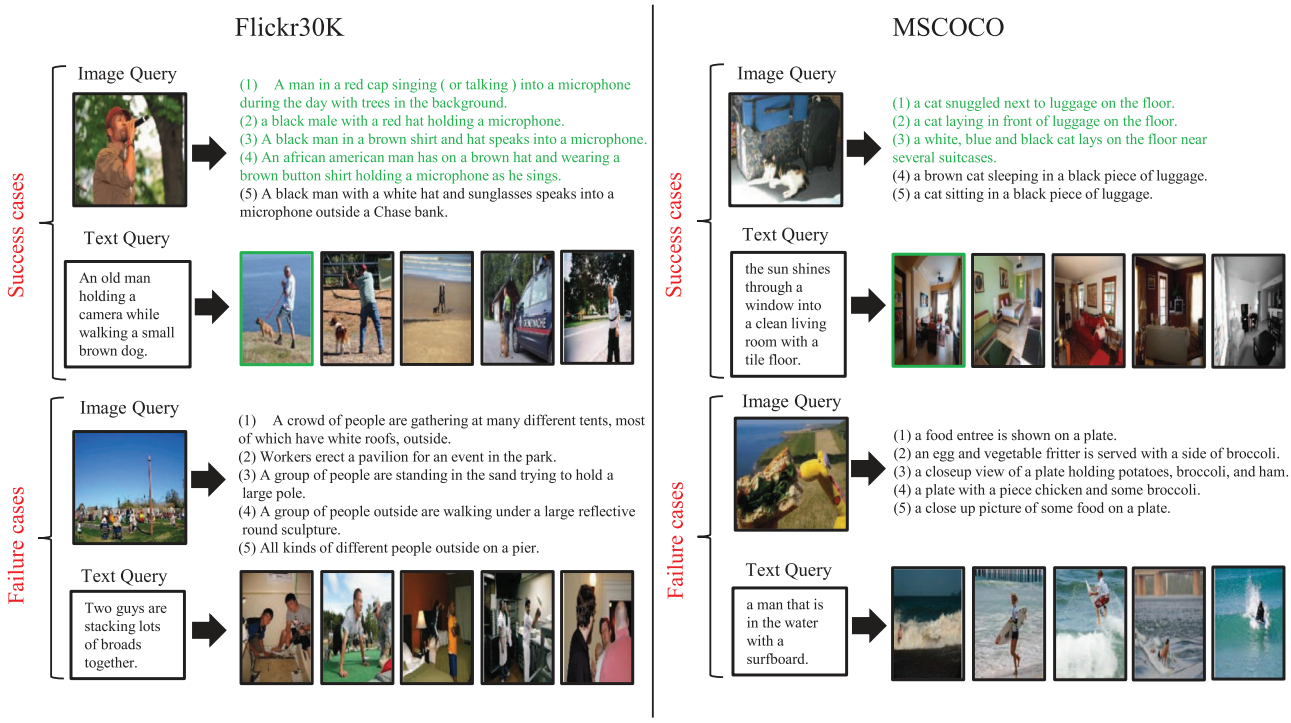


Fig. 12. Qualitative results of our CycleMatch on Flickr30K and MSCOCO. Given one query, the top-5 candidates are retrieved. In the success cases, the correct matches are highlighted with green. In the failure cases, our method can still retrieve some reasonable candidates related to the query. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 8
Evaluation on the effect of different feature encoders on the performance of CycleMatch. By comparison, ResNet-152 is a superior image encoder and RNN is a more powerful text encoder.

Image encoder	Text encoder	Flickr30K						MSCOCO					
		Image to Text			Text to Image			Image to Text			Text to Image		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Effect of image encoders													
VGG-19	RNN	51.4	80.6	88.1	38.5	71.0	81.3	55.1	83.5	91.3	43.7	76.7	88.4
ResNet-152	RNN	58.6	83.6	91.6	43.6	75.3	84.2	61.1	86.8	94.2	47.9	80.9	90.9
Effect of text encoders													
ResNet-152	word2vec	48.1	78.7	87.4	37.7	70.8	81.1	55.9	83.8	91.8	44.7	79.1	87.7
ResNet-152	HGLMM	54.5	81.6	90.9	41.3	73.1	82.8	58.4	85.5	93.4	46.2	80.3	89.4
ResNet-152	RNN	58.6	83.6	91.6	43.6	75.3	84.2	61.1	86.8	94.2	47.9	80.9	90.9

[17,38,43,67]. On the MSCOCO dataset, the fine-tuned approaches [17,38,43,67] merge the validation images into the training set to further increase the performance. However, we still use the original training set for a fair comparison with other prior approaches. Notice that, the accuracies of image-to-text retrieval are higher than those of text-to-image retrieval, because one image is annotated with several texts but one text matches with only one image.

In addition to the quantitative evaluation, we present our image-to-text and text-to-image retrieval examples in Fig. 12, which includes both success and failure cases. For each query sample, the top-5 candidates are retrieved, of which the ground-truth samples are highlighted in green. We notice that, the retrieved candidates are semantically related to the query sample in some extent, even for the failure cases.

6.5. Effect of feature encoders

As shown in Fig. 3, we extract visual and textual features from off-the-shelf feature encoders. The proposed CycleMatch can be compatible with diverse feature encoders, but it is still encouraged to study the effect of different feature encoders on the performance. We report the results in Table 8.

Considering the image encoders, we use the VGG-19 and ResNet-152 models to extract the visual features and compare their results. We can see that, ResNet-152 has a considerable improvements over VGG-19 on all measurements, especially for R@1 accuracies. This shows the benefit of using more powerful CNN models for improving the visual embeddings. In addition, the feature dimension with ResNet-152 (i.e. 2,048) is lower than that with VGG-19 (i.e. 4,096). Therefore, in this work we take the ResNet-152 model as the preferable image encoder.

In terms of the text encoders, we test another two encoders apart from the RNN encoder. The first one is word2vec [70], which describes each word in the sentence with a 300-dimensional feature vector. We then compute the average of all the word features to represent the sentence feature. The second one is an expensive representation based on the Hybrid Gaussian-Laplacian mixture model (HGLMM) [33]. Specifically, HGLMM computes a 18,000-dimension feature vector with 30 centers (i.e. $300 \times 30 \times 2$). Similar to [9], we further reduce it to a 6,000-dimension feature vector in order to decrease the training complexity. As shown in Table 8, the RNN encoder is more powerful than both word2vec and HGLMM. In addition, the feature dimension based RNN (i.e. 4,096) is feasible and practical during training CycleMatch.

Table 9
Evaluation on the effect of fine-tuning the image encoder during training CycleMatch.

Image encoder	Flickr30K						MSCOCO					
	Text encoder	Image to Text			Text to Image			Image to Text			Text to Image	
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
VGG-19 without fine-tune	51.4	80.6	88.1	38.5	71.0	81.3	55.1	83.5	91.3	43.7	76.7	88.4
VGG-19 with fine-tune	54.8	83.1	90.5	42.3	74.8	84.5	60.2	87.3	94.0	49.3	81.2	91.8

Table 10
Evaluation on the effect of different test splits on the performance of CycleMatch. The results on the MSCOCO dataset show that CycleMatch can achieve high mean accuracy and low standard deviation.

Image encoder	Text encoder	Image to Text			Text to Image		
		R@1	R@5	R@10	R@1	R@5	R@10
ResNet-152	RNN	60.23 ± 1.46	88.08 ± 1.19	94.88 ± 0.77	47.73 ± 0.91	81.89 ± 0.88	91.41 ± 0.62

6.6. Effect of fine-tuning image encoders

In this experiment, we perform the fine-tuning (ft) process for the VGG-19 image encoder. The results in Table 9 show considerable improvements for all R@K measurements. Similarly, fine-tuning ResNet-152 can bring further improvements as well, while it is out of the scope in our work.

6.7. Variance of test splits

For a fair comparison, we employ the standard data split including 1000 test images that are captured from the validation set [12,60]. However, no prior work has studied the effect of using different test splits on the retrieval performance. To this end, we perform 100 times of evaluations on the MSCOCO dataset. For each evaluation, we randomly select 1000 images from the validation set and test the results with the proposed CycleMatch. As shown in Table 10, our results show high mean accuracy and low standard deviation. This reveals the proper stability of our approach for cross-modal retrieval. It is worth mentioning that we cannot conduct this experiment on the Flickr30K dataset, as its test set (i.e. including only 1000 images) has been already fixed.

7. Conclusions

In this paper, we have developed a novel embedding method for the multi-modal task of matching visual and textual representations. We proposed cycle-consistent embeddings to learn both intra-modal correlations and intra-modal consistency. Our approach taking advantage of multiple embedding techniques is able to outperform any single embedding method. The experimental results have demonstrated the superiority of our method over other embedding methods. In addition, we have presented two simple and efficient late-fusion approaches to increase the inference quality. The late-fusion inference can integrate different matching scores together without increasing the training complexity. Finally, our approach has shown state-of-the-art performance for cross-modal retrieval on Flickr30K and MSCOCO. In the future, we will take into account local relations when matching images and sentences, for example, semantic correlations between visual regions and phases. One potential solution is to exploit the attention mechanism to localize the objects corresponding to the phase description.

Acknowledgments

This work was supported by the LIACS Media Lab at Leiden University under Grant 2006002026 and the National Natural Science Foundation of China under Grant 61872379. We are also grateful to the support of NVIDIA with the donation of GPU cards.

References

[1] D. Rafailidis, S. Manolopoulou, P. Daras, A unified framework for multimodal retrieval, Pattern Recognit. 46 (12) (2013) 3358–3370.
[2] V.E. Liong, J. Lu, Y.-P. Tan, Cross-modal discrete hashing, Pattern Recognit. 79 (2018) 114–129.
[3] M. Malinowski, M. Rohrbach, M. Fritz, Ask your neurons: a neural-based approach to answering questions about images, in: IEEE ICCV, 2015, pp. 1–9.
[4] S. Reed, Z. Akata, H. Lee, B. Schiele, Learning deep representations of fine-grained visual descriptions, in: IEEE CVPR, 2016, pp. 49–58.
[5] L. Gao, Z. Guo, H. Zhang, X. Xu, H.T. Shen, Video captioning with attention-based lstm and semantic consistency, IEEE Trans. Multimed. 19 (9) (2017) 2045–2055.
[6] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: NIPS, 2012, pp. 1097–1105.
[7] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, ICLR, 2015.
[8] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE CVPR, 2016, pp. 770–778.
[9] L. Wang, Y. Li, S. Lazebnik, Learning deep structure-preserving image-text embeddings, in: IEEE CVPR, 2016, pp. 5005–5013.
[10] H. Nam, J.-W. Ha, J. Kim, Dual attention networks for multimodal reasoning and matching, in: IEEE CVPR, 2017, pp. 299–307.
[11] Y. Huang, W. Wang, L. Wang, Instance-aware image and sentence matching with selective multimodal lstm, in: IEEE CVPR, 2017, pp. 2310–2318.
[12] Y. Liu, Y. Guo, E.M. Bakker, M.S. Lew, Learning a recurrent residual fusion network for multimodal matching, in: IEEE ICCV, 2017, pp. 4107–4116.
[13] Z. Niu, M. Zhou, L. Wang, X. Gao, G. Hua, Hierarchical multimodal lstm for dense visual-semantic embedding, in: IEEE ICCV, 2017, pp. 1881–1889.
[14] F. Feng, X. Wang, R. Li, Cross-modal retrieval with correspondence autoencoder, in: ACM MM, 2014, pp. 7–16.
[15] A. Habibian, T. Mensink, C.G. Snoek, Videostory: a new multimedia embedding for few-example recognition and translation of events, in: ACM International Conference on Multimedia (MM), 2014, pp. 17–26.
[16] A. Eisenschtat, L. Wolf, Linking image and text with 2-way nets, in: IEEE CVPR, 2017, pp. 4601–4611.
[17] J. Gu, J. Cai, S. Joty, L. Niu, G. Wang, Look, imagine and match: improving textual-visual cross-modal retrieval with generative models, IEEE CVPR, 2018.
[18] D. He, Y. Xia, T. Qin, L. Wang, N. Yu, T. Liu, W.-Y. Ma, Dual learning for machine translation, in: NIPS, 2016, pp. 820–828.
[19] J.-Y. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: IEEE ICCV, 2017, pp. 2223–2232.
[20] H. Hotelling, Relations between two sets of variates, Biometrika 28 (1936) 321–377.
[21] P.L. Lai, C. Fyfe, Kernel and nonlinear canonical correlation analysis, IEEE TPAMI 10 (05) (2000) 365–377.
[22] P. Mineiro, N. Karampatziakis, A randomized algorithm for cca, Neural Information Processing Systems (NIPS) workshop, 2014.
[23] T. Michaeli, W. Wang, K. Livescu, Nonparametric canonical correlation analysis, in: International Conference on Machine Learning (ICML), 2016, pp. 1967–1976.
[24] D.R. Hardoon, S.R. Szedmak, J.R. Shawe-taylor, Canonical correlation analysis: an overview with application to learning methods, Neural Comput. 16 (12) (2004) 2639–2664.
[25] Y. Gong, Q. Ke, M. Isard, S. Lazebnik, A multi-view embedding space for modeling internet images, tags, and their semantics, IJCV 106 (2) (2014) 210–233.
[26] V. Ranjan, N. Rasiwasia, C.V. Jawahar, Multi-label cross-modal retrieval, in: IEEE ICCV, 2015, pp. 4094–4102.

[27] A. Salvador, N. Hynes, Y. Aytar, J. Marin, F. Ofli, I. Weber, A. Torralba, Learning cross-modal embeddings for cooking recipes and food images, in: IEEE CVPR, 2017, pp. 3020–3028.
[28] L. Ma, Z. Chen, L. Xu, Y. Yan, Multimodal deep learning for solar radio burst classification, Pattern Recognit. 61 (2017) 573–582.
[29] S. Karaoglu, R. Tao, T. Gevers, A.W.M. Smeulders, Words matter: scene text for image classification and retrieval, IEEE Trans. Multimed. 19 (5) (2017) 1063–1076.
[30] X. Bai, M. Yang, P. Lyu, Y. Xu, Integrating scene text and visual appearance for fine-grained image classification with convolutional neural networks, CoRR (2017). 1704.04613.
[31] G. Andrew, R. Arora, K. Livescu, J. Bilmes, Deep canonical correlation analysis, in: ICML, 2013, pp. 1247–1255.
[32] F. Yan, K. Mikolajczyk, Deep correlation for matching images and text, in: IEEE CVPR, 2015, pp. 3441–3450.
[33] B. Klein, G. Lev, G. Sadeh, L. Wolf, Associating neural word embeddings with deep image representations using fisher vectors, in: IEEE CVPR, 2015, pp. 4437–4446.
[34] A. Karpathy, F.-F. Li, Deep visual-semantic alignments for generating image descriptions, in: IEEE CVPR, 2015, pp. 3128–3137.
[35] L. Ma, Z. Lu, L. Shang, H. Li, Multimodal convolutional neural networks for matching image and sentence, in: IEEE ICCV, 2015, pp. 2623–2631.
[36] R. Kiros, R. Salakhutdinov, R.S. Zemel, Unifying visual-semantic embeddings with multimodal neural language models, NIPS Workshop, 2014.
[37] Y. Liu, L. Liu, Y. Guo, M.S. Lew, Learning visual and textual representations for multimodal matching and classification, Pattern Recognit. 84 (2018) 51–67.
[38] Z. Zheng, L. Zheng, M. Garrett, Y. Yang, Y. Shen, Dual-path convolutional image-text embedding, CoRR (2017). 1711.05535.
[39] B. Wang, Y. Yang, X. Xu, A. Hanjalic, H.T. Shen, Adversarial cross-modal retrieval, in: ACM Multimedia, 2017, pp. 154–162.
[40] S. Rastegar, M. Soleymani, H.R. Rabiee, S. Mohsen Shojaee, Mdl-cw: a multimodal deep learning framework with cross weights, in: IEEE CVPR, 2016, pp. 2601–2609.
[41] V. Vukotić, C. Raymond, G. Gravier, Bidirectional joint representation learning with symmetrical deep neural networks for multimodal and crossmodal applications, in: ICMR, 2016, pp. 343–346.
[42] E. Kodirov, T. Xiang, S. Gong, Semantic autoencoder for zero-shot learning, in: IEEE CVPR, 2017, pp. 3174–3183.
[43] Y. Huang, Q. Wu, L. Wang, Learning semantic concepts and order for image and sentence matching, IEEE CVPR, 2018.
[44] F. Wang, Q. Huang, L. Guibas, Image co-segmentation via consistent functional maps, in: IEEE ICCV, 2013, pp. 849–856.
[45] T. Zhou, P. Krähenbühl, M. Aubry, Q. Huang, A.A. Efros, Learning dense correspondence via 3d-guided cycle consistency, in: IEEE CVPR, 2016, pp. 117–126.
[46] C. Godard, O. Mac Aodha, G.J. Brostow, Unsupervised monocular depth estimation with left-right consistency, in: IEEE CVPR, 2017, pp. 270–279.
[47] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: NIPS, 2014, pp. 2672–2680.
[48] Z. Yi, H. Zhang, P. Tan, M. Gong, Dualgan: unsupervised dual learning for image-to-image translation, in: IEEE ICCV, 2017, pp. 2849–2857.
[49] T. Kim, M. Cha, H. Kim, J.K. Lee, J. Kim, Learning to discover cross-domain relations with generative adversarial networks, in: ICML, 2017, pp. 1857–1865.
[50] R. Felix, B.V. Kumar, I. Reid, G. Carneiro, Multi-modal cycle-consistent generalized zero-shot learning, in: ECCV, 2018, pp. 21–37.
[51] G. Zhang, M. Kan, S. Shan, X. Chen, Generative adversarial network with spatial attention for face attribute editing, in: V. Ferrari, M. Hebert, C. Sminchisescu, Y. Weiss (Eds.), ECCV, 2018, pp. 422–437.
[52] H. Tang, W. Wang, D. Xu, Y. Yan, N. Sebe, Gesturegan for hand gesture-to-gesture translation in the wild, in: ACM Multimedia, 2018, pp. 774–782.
[53] X. Chen, C.L. Zitnick, Mind's eye: a recurrent visual representation for image caption generation, in: IEEE CVPR, 2015, pp. 2422–2431.
[54] L. van der Maaten, G. Hinton, Visualizing data using t-sne, J. Mach. Learn. Res. 9 (2008) 2579–2605.
[55] P. Young, A. Lai, M. Hodosh, J. Hockenmaier, From image descriptions to visual denotations: new similarity metrics for semantic inference over event descriptions, in: ACL, 2014, pp. 67–78.
[56] K. Nandakumar, Y. Chen, S.C. Dass, A. Jain, Likelihood ratio-based biometric score fusion, IEEE TPAMI 30 (2) (2008) 342–347.
[57] L. Zheng, S. Wang, L. Tian, F. He, Z. Liu, Q. Tian, Query-adaptive late fusion for image search and person re-identification, in: IEEE CVPR, 2015, pp. 1741–1750.
[58] A. Bansal, S. Ma, D. Ramanan, Y. Sheikh, Recycle-gan: unsupervised video retargeting, ECCV, 2018.
[59] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, T. Darrell, CyCADA: cycle-consistent adversarial domain adaptation, in: ICML, 2018, pp. 1989–1998.
[60] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, A. Yuille, Deep captioning with multimodal recurrent neural networks (m-rnn), ICLR, 2015.
[61] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, C.L. Zitnick, Microsoft coco: common objects in context, in: ECCV, 2014, pp. 740–755.
[62] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. Fei-Fei, Imagenet large scale visual recognition challenge, IJCV 115 (3) (2015) 211–252.
[63] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, A. Bordes, Supervised learning of universal sentence representations from natural language inference data, in: EMNLP, 2017, pp. 670–680.
[64] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: convolutional architecture for fast feature embedding, in: ACM MM, 2014, pp. 675–678.
[65] X. Lin, D. Parikh, Leveraging visual question answering for image-caption ranking, in: ECCV, 2016, pp. 261–277.
[66] G. Lev, G. Sadeh, B. Klein, L. Wolf, RNN fisher vectors for action recognition and image annotation, in: ECCV, 2016, pp. 833–850.
[67] F. Faghri, D.J. Fleet, R. Kiros, S. Fidler, VSE++: improved visual-semantic embeddings, CoRR (2017). 1707.05612.
[68] R. Kiros, Y. Zhu, R.R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, S. Fidler, Skip-thought vectors, in: NIPS, 2015, pp. 3294–3302.
[69] I. Vendrov, R. Kiros, S. Fidler, R. Urtasun, Order-embeddings of images and language, ICLR, 2016.
[70] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: NIPS, 2013, pp. 3111–3119.

Yu Liu received the B.S. degree and M.S degree from School of Software Technology, Dalian University of Technology, Dalian, China, in 2011 and 2014, respectively, and the Ph.D. degree in Leiden Institute of Advanced Computer Science (LIACS), Leiden University, in 2018. His current research interests include computer vision and deep learning, especially, image classification, image retrieval and multi-modal matching. He has published papers in international conferences and journals including CVPR, ICCV, ICMR, Pattern Recognition, TMM and MTAP, and obtained a best paper award at the 23rd International Conference on MultiMedia Modeling (MMM) in 2017.

Yanming Guo is now a lecturer in the College of System Engineering, National University of Defense Technology. He received his B.S. and M.S. degrees from the National University of Defense Technology, in 2011 and 2013, respectively, and the Ph.D. degree in Leiden Institute of Advanced Computer Science (LIACS), Leiden University, in 2017. His current interests include computer vision, natural language processing and deep learning. He has served as reviewers of some journals and conferences, such as TNNLS, TMM, Neurocomputing, MTAP and ICPR.

Li Liu received the B.S. degree in communication engineering, the M.S. degree in photogrammetry and remote sensing and the Ph.D. degree in information and communication engineering from the National University of Defense Technology (NUDT), China, in 2003, 2005 and 2012, respectively. She joined the faculty at the NUDT in 2012, where she is currently an Associate Professor with the College of Information System and Management. During her PhD study, she spent more than two years as a Visiting Student at the University of Waterloo, Canada, from 2008 to 2010. From 2015 to 2016, she spent ten months visiting the Multimedia Laboratory at the Chinese University of Hong Kong. From 2016 to 2018, she is visiting Machine Vision Group at the University of Oulu, Finland. Dr. Liu was a cochair of International Workshops at ACCV2014, CVPR2016, and ICCV2017. She was a guest editor of the special issue on RoLoD: Robust local descriptors for computer vision for the journal of Neurocomputing. She is a guest editor of the special issue on Compact and Efficient Feature Representation and Learning in Computer Vision for IEEE Transactions on Pattern Analysis and Machine Intelligence. Her papers have currently over 1200 citations in Google Scholar. Her current research interests include texture analysis, image classification, object detection and scene understanding. She serves as an Associate Editor of The Visual Computer Journal.

Erwin M. Bakker is co-director of the LIACS Media Lab at Leiden University. He has published widely in the fields of image retrieval, audio analysis and retrieval and bioinformatics. He was closely involved with the start of the International Conference on Image and Video Retrieval (CIVR) serving on the organizing committee in 2003 and 2005. Moreover, he regularly serves as a program committee member or organizing committee member for scientific multimedia and human-computer interaction conferences and workshops.

Michael S. Lew is co-head of the Imagery and Media Research Cluster at LIACS and director of the LIACS Media Lab. He received his doctorate from University of Illinois at Urbana-Champaign and then became a postdoctoral researcher at Leiden University. One year later he became the first Leiden University Fellow which was a pilot program for tenure track professors. In 2003, he became a tenured associate professor at Leiden University and was invited to serve as a chair full professor in computer science at Tsinghua University (the MIT of China). He has published over 100 peer reviewed papers with three best paper citations in the areas of computer vision, content-based retrieval, and machine learning. Currently (September 2014), he has the most cited paper in the history of the ACM Transactions on Multimedia. In addition, he has the most cited paper from the ACM International Conference on Multimedia Information Retrieval (MIR) 2008 and also from ACM MIR 2010. He has served on the organizing committees for over a dozen ACM and IEEE conferences. He served as the founding the chair of the ACM ICMR steering committee and had served as chair for both the ACM MIR and ACM CIVR steering committees. In addition he is the Editor-in-Chief of the International Journal of Multimedia Information Retrieval (Springer) and a member of the ACM SIGMM Executive Board which is the highest and most influential committee of the SIGMM.