



# Multi-grained encoding and joint embedding space fusion for video and text cross-modal retrieval

Xiaotao Cui<sup>1</sup> · Jing Xiao<sup>1</sup> · Yang Cao<sup>1</sup> · Jia Zhu<sup>1</sup>

Received: 27 August 2020 / Revised: 30 November 2021 / Accepted: 4 April 2022 /

Published online: 30 May 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

## Abstract

Video-text cross-modal retrieval is significant to computer vision. Most of existing works focus on exploring the global similarity between modalities, but ignore the influence of details on retrieval results. How to explore the correlation between different forms of data from multiple angles is a key issue. In this paper, we propose a Multi-grained Encoding and Joint Embedding Spaces Fusion (MEJESF) for video-text cross-modal retrieval. Specifically, we propose a novel dual encoding network to explore not only coarse-grained feature but also fine-grained feature of modals. At the same time, giving considerations to multiple encoding and hard sample mining, a modified pairwise ranking loss function is introduced. After that, we build two joint embedding spaces and adopt them when retrieving by fusing their scores. Experiments on two public benchmark datasets (MSR-VTT,MSVD) demonstrate that our method can obtain promising performance compared to the state-of-the-art methods in video-text cross-modal retrieval. Furthermore, our network model achieves outstanding performance in zero-example video retrieval.

**Keywords** Multi-grained encoding · Joint embedding space fusion · Cross-modal retrieval · Zero-example video retrieval

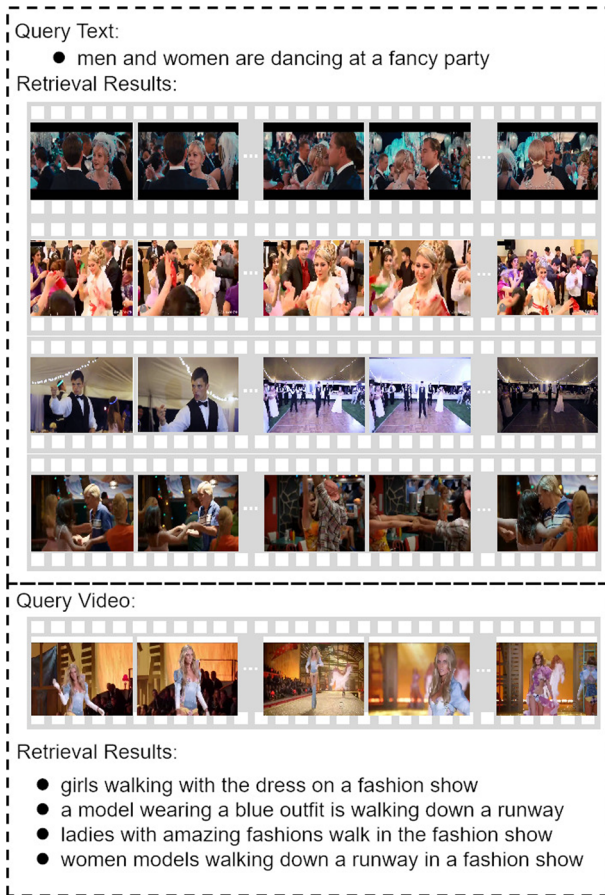
## 1 Introduction

The relationship between video and text is vital to computer vision research and many practical applications including video retrieval [22], video object segmentation [40], video caption [35] and video question answering [34]. This paper focuses on cross-modal retrieval of video and text. The goal of this work is to retrieve the correlated videos given a text sentence, and vice versa, given a random video query to find the matched sentence descriptions (Fig. 1). Because a huge gap between video data and natural language descriptions, it is challenging to explore a well visual-semantic alignment to measure the similarity of video and text in cross-modal retrieval.

---

✉ Jing Xiao  
xiaojing@scnu.edu.cn

<sup>1</sup> School of Computer Science, South China Normal University, Guangzhou, China



**Fig. 1** Illustration of Video-Text cross-modal retrieval task: the upper box shows text-to-video retrieval and the bottom box shows video-to-text retrieval

Some of the earlier methods were related to concepts. For example, Yu et al. [37] designed a word detector to generate a list of high-level concept words for per video as a semantic prior for task. However, these methods ignored the rich sequential information within both video and text.

Recently, some works have attempted to use deep neural networks to encode videos and texts in a common space. Pan et al. [23] used a deep convolution neural networks (CNN) to produce a representation of each clip from video and utilized an RNN-based network to produce sentences representation. Dong et al. [6] encoded captions into a textual embedding based on multi-scale sentence vectorization and used 3-D convolutional neural network to embed video features. Although these methods compared the similarities between video feature vectors and text feature vectors in a common space as well as image-text retrieval, they failed to take advantage of the various rich cues on videos, such as sound, actions,

scenes and faces. Synthesizing all kinds of cues can make the representation more complete and plentiful.

We argue that it is necessary to consider the various information and available cues of video. To address this issue, Mithun et al. [20] tried to incorporate multiple video cues into the model to enhance the robustness of the retrieval system. However, these models usually used pre-trained neural networks to extract features from video frames and encoded by one encoding strategy, where the information was not adequately encoded. They may fail to extract some higher level semantic information, and did not use more powerful and finer-grained representations.

In this paper, we propose a cross-modal retrieval method based on multi-grained encoding and common space fusion. The coarse-grained and fine-grained features are used to comprehensively explore the global similarity of events between modals and the local slight of objects. Multiple joint embedding spaces can complement each other and improve the accuracy of retrieval and resolve ambiguities in retrieval. In detail, we design a new encoding method firstly, and apply it to videos and sentences to obtain their respective coarse-grained features and fine-grained features. We expect to use a gated embedding block to capture the dependencies among features after mean-pooling layer as coarse-grained feature representation of the modal. In terms of fine-grained, we use bidirectional Gated Recurrent Unit (biGRU) network, a CNN layer in top to encode its features. Fine-grained feature encoding can learn the context, temporal and spatial information, local details of the modal. Considering that the coarse and fine-grained features are mapped into a joint embedding space simultaneously, the loss function is modified based on the hard triple loss, so that the loss function can comprehensively judge the semantic difference between cross modals and effectively optimize the model. Furthermore, to make better use of the multiple video cues, we construct two joint embedding spaces. The static-text space emphasizes on salient object, and the dynamic-text space concentrates on actions and events.

In summary, this paper makes the following contributions:

- (1) We propose a multi-granularity encoding network that encodes video and text input in a similar manner. Coarse-grained feature captures the global characteristics of the modal, while the fine-grained feature focuses on subtle local differences and both features are learned in the same common space. We also improve the pairwise loss function so that it can better deal with joint embedding space construction while mining hard sample.
- (2) The appearance cues and activity cues of the video and the features of text are used to construct the joint embedding spaces respectively while encoding methods of different spaces are consistent. We adopt score-based fusion for effective integration to improve retrieval accuracy.
- (3) Detailed ablation experiments and results prove the effectiveness of different components in our framework. The state-of-the-art performance for video-text cross-modal retrieval can be achieved on MSR-VTT and MSVD. The proposed network works well in zero-example video retrieval.

The remainder of this paper is organized as follows: in Section 2, we briefly review the most recent and related work. The proposed framework is explained in Section 3. Retrieval results have been analyzed and compared in Section 4. Finally, in Section 5, we conclude the paper and propose the directions for the future work.

## 2 Related work

### 2.1 The process of cross-modal retrieval

Text, image, video, audio and other modal data can be used as retrieval objects in the process of cross-modal retrieval. The retrieval process is divided into five steps:

- Step 1: Extracting the features for different modal data.
- Step 2: Constructing a shared representation of different modal through cross-modal retrieval models.
- Step 3: Calculating the distance between the shared representations of different modal data through the similarity measurement function in the public space.
- Step 4: Sorting all results from smallest to largest, the higher the correlation, the smaller the distance.
- Step 5: Returning the sorted results as the final cross-modal retrieval result.

In addition, the cross-modal retrieval model proposed in this paper also follows the above procedure.

### 2.2 Image-text retrieval

The usual methods used in image-text retrieval can be roughly divided into traditional statistical correlation analysis methods and Deep Neural Networks (DNN)-based methods. Hodosh et al. [10] firstly took the Canonical correlation analysis (CCA) to cross-modal retrieval, finding a joint embedding by maximizing the correlation between two sets of heterogeneous data. The method named Deep Canonical Correlation Analysis (DCCA) proposed by Andrew et al. [1] and other similar methods were all promoted based on CCA. Peng et al. [24] used a deep neural network with hierarchical architecture to learn the cross-modal shared representation. VSE++ proposed by Faghri et al. [8] has been used in many previous works for cross-modal retrieval which modified the pairwise ranking loss based on violations caused by the hard negatives (i.e., non-matching query closest to each training query). In recent years, some scholars have proposed some novel methods. Peng [25] considered that attention mechanism can fully explore modality-specific characteristics, and proposed MCSM. Zhang et al. [39] leveraged GANs and attention mechanisms to improve the accuracy of the retrieval. Shen et al. [26] believed that hashing technology can significantly reduce computational cost and storage. Zhang et al. [38] leveraged GANs and hashing. Xu et al. [31] proposed graph convolutional hashing (GCH) to realize fast and flexible cross-modal retrieval.

### 2.3 Video-text retrieval

Similar to the image-text cross-modal retrieval methods, most video-text cross-modal retrieval approaches also use semantic space. The work of Xu et al. [30] advocated leveraging the subject, verb, object component of the sentence, vectorization by word2vec model and used the Recursive Neural Network (RNN) to aggregate them into sentence vectors. Otani et al. [22] focused on disambiguating semantics of a sentence by web image search result. Yu et al. [37] proposed to incorporate concept words as semantic priors and used Long Short Term Memory (LSTM) to encode the video. Dong et al. [5] proposed a deep neural network named Word2VisualVec (W2VV) to predict matching based on sentence

vectorization strategy and a multi-layer perceptron. Mithun et al. [20] improved the effect of video-text-retrieval by focusing on the different cues in the video and modified the loss function. Liu et al. [15] introduced a collaborative expert model which can combine cues in video by a gating mechanism.

## 2.4 Zero-example retrieval

Recently, Zero-example retrieval has attracted wide attention in the computer vision field. In image domain, Xu et al. [33] and Chi et al. [4] used the word vectors of labels in the trained NLP models as external knowledge, and performed correlation learning, subspace learning for zero sample retrieval jointly. Xu et al. [32] proposed a model named TANSS, including two semantic feature learning subnetworks and a self-supervised semantic subnetwork, which ensured to learn more effective common semantic space. In video domain, in order to extract relevant concepts, Markatopoulou et al. [16] designed relatively complex linguistic rules and exploited pre-trained Convolution Neural Network (CNN) model to select objects and scenes appeared in video. The work of Ueki et al. [28] utilized a larger concept bank with more than 50,000 concepts. While majority of existing methods for zero-sample retrieval were based on concept, Dong et al. [7] proposed the method that employed the dual network before the common space learning and used multi-level encoding for each network.

Different from the existing approaches, our work focuses on the multi-grained features for various modals, the modification of the loss function, and the construction of two joint embedding spaces to achieve more abundant and complementary retrieval results.

## 3 The method

### 3.1 Overview of the proposed approach

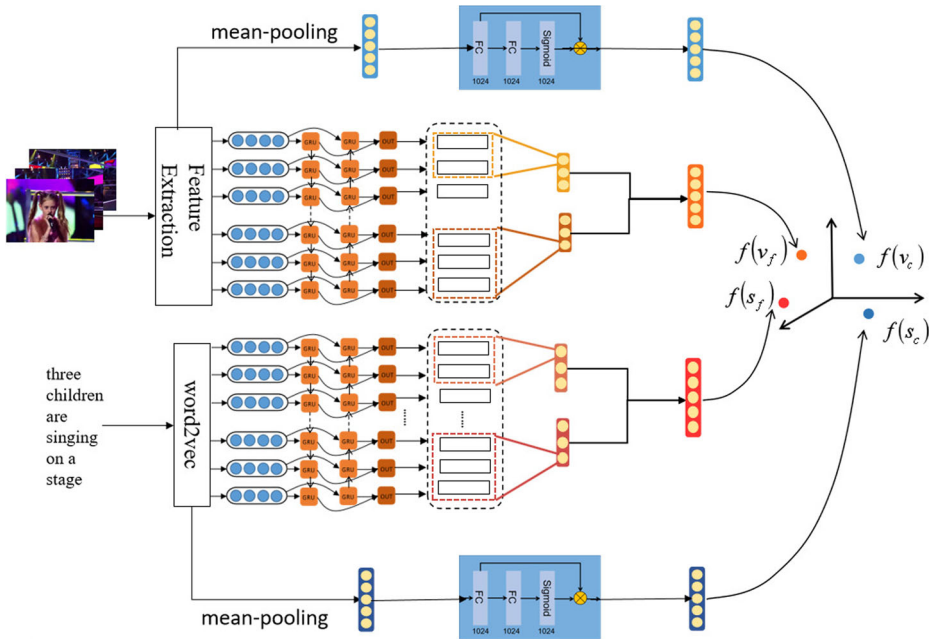
To learn the potential information and relation between videos and texts, we propose a multi-grained encoding and joint embedding space fusion network. The architecture is shown in Fig. 2. Specifically, we first describe the multi-grained encoding network of videos and sentences in Sections 3.2 and 3.3, including modal feature extraction, the methods of coarse-grained encoding and fine-grained encoding. Finally, we present our modification on the loss function and introduce the fusion step for two joint embedding spaces in Section 3.4. The main notations are described in Table 1.

### 3.2 Video-side multi-granularity encoding

A large number of frames are contained in a video. For each input video, we sample  $n$  frames according to the fixed frame interval, and extract deep features of each frame with a pre-trained model. Consequently, a video is described by a sequence of frame vectors  $\{v_1, v_2, \dots, v_n\}$ , where  $v_t$  represents the feature of the  $t$ -th frame, and the dimension is determined by different video extraction methods.

For each video, we use various networks to extract different video cues.

**Appearance feature** ResNet152 [9] is known to be effective for extracting image features. Firstly, we extract each frame of the video and scale them to  $224 \times 224$  and use them as the



**Fig. 2** Overview of the proposed model: we multi-grained encode the extracted features with mean-pooling, gated embedding block, biGRU, and parallel 1D convolutional network on top. After that the coarse-grained and fine-grained features are projected into a common space for similarity computation. Then we form two joint embedding spaces, i.e. the static-text joint embedding space and the dynamic-text joint embedding space. Finally, the sum of similarity scores is used for ranking

input to CNN. We use ResNet152 to extract 2048 dimensional appearance features of video. ResNet152 are pre-trained using ImageNet dataset.

**Action feature** I3D [2] is good at extracting spatial and temporal features from video. Taking the continuous 16 frames of a video as input, we extract 1024 dimensional action features using I3D.

**Table 1** Main notations

Notation	Description
$v_t$	the feature of the $t$ -th frame in the video
$w_1$	the word embedding vector of the $t$ -th word in the description
$f(v_c)$	the coarse-grained encoding for video
$f(v_f)$	the fine-grained encoding for video
$f(s_c)$	the coarse-grained encoding for text
$f(s_f)$	the fine-grained encoding for text
$S^*(,)$	the similarity score between a video and a text
$S_{s-t}$	the similarity score in static-text space
$S_{d-t}$	the similarity score in dynamic-text space
$S_{v-t}(v, t)$	the sum similarity score between video and text in two spaces

### 3.2.1 Coarse-grained encoding

Coarse-grained encoding pays close attention to distinguishing different categories of videos and focuses on the main objects and events described by the modal.

Mean-pooling is widely used in video encoding and it can achieve excellent results by literature review. Since mean-pooling is good at capturing visual patterns repeatedly in video, we encode the extracted feature vectors by mean-pooling layer firstly, represented as  $f(v)$ , where  $v \in \mathbb{R}^{n \times d_v}$ , and the size of  $d_v$  depends on the video cues:

$$f(v) = \frac{1}{n} \sum_{t=1}^n v_t \tag{1}$$

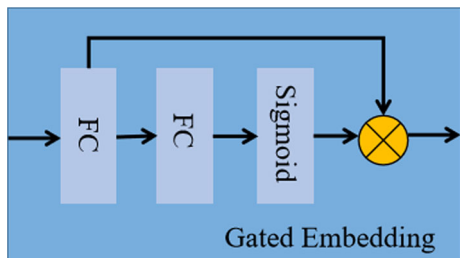
A video contains a lot of things; some are core, but some are irrelevant. In order to enhance the relationship between the most relevant objects and events and suppress the impact of irrelevant objects on retrieval, we further encode the video feature vector  $f(v)$  with a gated embedding block, which is proved to be effective to capture dependencies among features in [18]. Taking a video that a puppy and a little girl are playing frisbee in the garden as an example, the flowers in the garden is running through the entire video. The network activity for flowers feature may be high. However, in this video, the puppy and the little girl are the main characters, and playing frisbee is the main activity. The flowers feature is not so important when retrieving. The gated embedding block can learn how to reduce the visual activation of flowers and focus on puppy, little girl, and playing frisbee well. The gated embedding block is shown in Fig. 3.

The gated embedding block is composed of two fully connected layers and a sigmoid activation layer. The first fully connected layer assembles the convolution pooled features into a complete graph through the weight matrix. The second fully connected layer and the activation function sigmoid form a context gating function to adjust the output of the linear layer. Finally, the output of the first fully connected layer and the context gating function are element-wise multiplied to obtain the final video encoding vector  $f(v_c)$ . More formally we express the above process as:

$$f(v_c) = (W_1 f(v) + b_1) \circ \sigma(W_2(W_1 f(v) + b_1) + b_2) \tag{2}$$

where  $W_1 \in \mathbb{R}^{d \times d_v}$ ,  $W_2 \in \mathbb{R}^{d \times d}$ ,  $b_1, b_2 \in \mathbb{R}^d$  are learnable parameters;  $\sigma$  is a sigmoid activation and  $\circ$  is the element-wise multiplication (Hadamard product), and  $d=1024$ .

**Fig. 3** The structure of gated embedding block: consisting of two fully connected layers and a sigmoid activation layer



### 3.2.2 Fine-grained encoding

Unlike the coarse-grained features that emphasize the overall situation, the fine-grained features are more concerned with the temporal aware, the development sequence of events and the connection between objects.

Recurrent Neural Network (RNN) is effective to deal with the temporal information between video frames. We primarily use a bidirectional GRU (biGRU) for video sequence, which considers both future and past contexts for a given sequence. The biGRU consists of a forward GRU and a backward GRU, where the forward GRU is in charge of encoding video sequence in normal order and the backward GRU is responsible for encoding video sequence in reverse order.  $\vec{h}$  and  $\overleftarrow{h}$  are the forward and backward hidden states of the biGRU, respectively:

$$\begin{aligned}\vec{h}_t &= \overrightarrow{GRU}(v_t, \vec{h}_{t-1}) \\ \overleftarrow{h}_t &= \overleftarrow{GRU}(v_{m+1-t}, \overleftarrow{h}_{t-1})\end{aligned}\quad (3)$$

where  $\overrightarrow{GRU}$  and  $\overleftarrow{GRU}$  represent the forward and backward GRUs, and  $\vec{h}_t, \overleftarrow{h}_t \in \mathbb{R}^{512}$ . After that, by concatenating the forward and backward hidden states, we obtain the representation of biGRU output  $h_t = [\vec{h}_t, \overleftarrow{h}_t]$  where  $h_t \in \mathbb{R}^{1024}$ . A video can be mapped as  $H = \{h_1, h_2, \dots, h_n\}$ , with a size of  $n \times 1024$ .

To capture deeper hidden features and enhance the capabilities of the network, we build convolutional networks (CNN) on the top of biGRU, because CNN can distinguish the slight disparities between videos. In this paper, 1-d CNN [11] is used.

The feature map  $H$  generated by the biGRU is used as the input of an 1-d convolutional block  $Conv1d_{k,r}$ , where  $r = 512$  filters of size  $k$ , with  $k \geq 2$ . After that, an  $m \times r$  feature map is generated. We further apply ReLU activation function on the feature map and employ max-pooling layer which can compress the feature map to vector  $c_k$  of fixed length  $r$ . The representation that focuses on local features  $f(v_f)$  is concatenated by two  $c_k$  where  $k = 2$  and  $k = 3$ . The encoding process can be expressed as:

$$\begin{aligned}H_2 &= ReLU(Conv1d_{k,t}(H)) \\ c_k &= max - pooling(H_2) \\ f(v_f) &= [c_2, c_3].\end{aligned}\quad (4)$$

### 3.3 Text-side multi-granularity encoding

The encoding process of the text is consistent with the video as a whole.

For word representation, we use a 300-dimensional pre-trained word2vec word embedding model. A sentence of length  $m$ , can be represented by a word sequence  $\{w_1, w_2, \dots, w_m\}$  where  $w_t$  represents the word embedding vector of the  $t$ -th word in the description.

Similar to coarse-grained encoding of video (as Section 3.2.1), the word sequence passes through a mean-pooling layer and a gate embedding block to obtain the coarse-grained feature  $f(s_c)$ .



The fine-grained encoding module of text is similar to video encoding in Section 3.2.2. The formation process of  $f(s_f)$  is as follows:

The sentence vector embedding by word2vec is fed to biGRU network to acquire context-related encoding, and two one-dimensional convolution blocks with  $k=2,3$  are used to enhance local patterns of text.

Videos and sentences are essentially series of items (frames or words), which allows us to design a dual encoding network to handle two modalities. Such design is more concise and more symmetrical. However, in order to maintain the uniqueness of each modal, the parameters of video and text encoding network are not shared.

### 3.4 Joint embedding learning

#### 3.4.1 Loss function

The pairwise ranking model is employed to learn the joint embedding space. Specifically, we utilize the hard triplet loss as the fundamental loss function, which is widely used in face recognition field. Given a positive video-text pair  $(s, v)$ , the loss function of individual features is defined as follows:

$$\begin{aligned} \mathcal{L} = & \sum_{s^-} [\alpha - S(s, v) + S(s^-, v)]_+ \\ & + \sum_{v^-} [\alpha - S(s, v) + S(s, v^-)]_+ \end{aligned} \tag{5}$$

where  $[x]_+ = \max(0, x)$  and  $\alpha$  is a margin constant.  $S(s, v)$  denotes the similarity score of matched video  $v$  and text  $s$ ;  $S(s, v^-)$  denotes the similarity score between the query text and video, where the video does not match the query text but is most similar to it in the batch. And vice versa with  $S(s^-, v)$ . The cosine similarity is used to calculate the similarity score.

In our method, we do not fuse the features from different granularity, which results in two kinds of cross-modal features: the coarse-grained features  $(s_c, v_c)$  and the fine-grained features  $(s_f, v_f)$ ; the loss is modified as:

$$\begin{aligned} \mathcal{L} = & \sum_{s^-} \left[ \alpha - S^*(s_{c,f}, v_{c,f}) + S^*(s_{c,f}^-, v_{c,f}) \right]_+ + \\ & \sum_{v^-} \left[ \alpha - S^*(s_{c,f}, v_{c,f}) + S^*(s_{c,f}, v_{c,f}^-) \right]_+ \end{aligned} \tag{6}$$

where  $S^*(s_{c,f}, v_{c,f}) = \lambda S(f(s_c), f(v_c)) + (1 - \lambda)S(f(s_f), f(v_f))$  and  $\lambda$  is the tradeoff weight.

Calculating the similarity of  $(f(s_c), f(v_c))$  can help compare the global consistency between query object and matching sample. The purpose of calculating the similarity of  $(f(s_f), f(v_f))$  is to distinguish the slight difference between positive sample and non-matching sample. The weighted combination of the two similarities makes the hardest negative sample most similar to the positive sample, thus better measuring the optimization model.

The training process of the MEJESF is shown in the Algorithm 1:

**Algorithm 1** Training process of MEJESF.**Input:** training dataset  $D_{train}$ , batch size  $N$ , learning rate  $r$ ,  $\lambda$ , max epoch  $T$ .

- 1: Extract video features by pre-trained network:  $v = \{v_1, v_2, \dots, v_n\}$ .
- 2: Extract text features by pre-trained network:  $s = \{w_1, w_2, \dots, w_m\}$ .
- 3: **repeat**
- 4:     **for**  $t = 1$  to  $T$  **do**
- 5:         Coarse-grained encoding for video:  $v \rightarrow f(v_c)$ , and fine-grained encoding for video:  $v \rightarrow f(v_f)$ .
- 6:         Coarse-grained encoding for text:  $s \rightarrow f(s_c)$ , and fine-grained encoding for text:  $s \rightarrow f(s_f)$ .
- 7:         Update the model by equation (6).
- 8:     **end for**
- 9: **until** MEJESF model converges.
- 10: **return** The optimized MEJESF model.

**3.4.2 Score-based fusion of joint embedding spaces**

”Who does what” is the basic content of a video. To make full use of multiple cues of the video, and pay attention to the object (who) and events (what) of the video during retrieval, we form two joint embedding spaces.

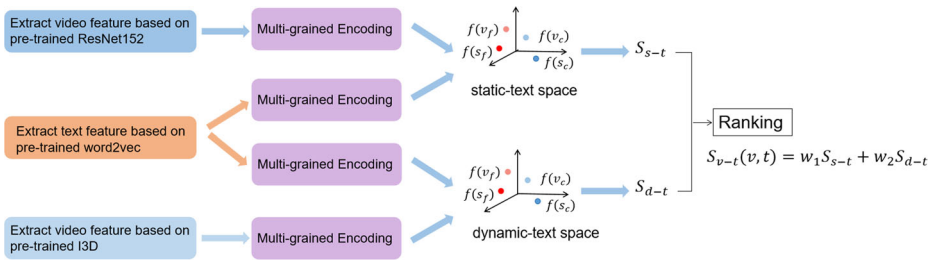
The coarse and fine grained features of videos extracted by ResNet152 and the coarse and fine grained features of text extracted by word2vec are mapped to the static-text joint embedding space where emphasizes the salient object. Another common space is named dynamic-text space. The video multi-granularity features in this space are obtained from features extracted by I3D. The dynamic-text space centers on actions and events of video. The same loss function is used to train both embedding models.

The use of multiple cues and combining or fusing retrieval flow in some way has been proved to be effective in traditional retrieval methods. At the time of retrieval, the early fusion method merges each cue before matching, while the late fusion method firstly performs matching on the different cues and then fuses the matching results. Late fusion can potentially support an adaptive fusion approach. Late fusion in retrieval can be roughly divided into score-based fusion and ranking-based fusion. Score-based fusion sums the similarity scores of query and results in joint spaces, and ranking-based fusion sums the ranking positions of retrieval resulting in different joint embedding spaces. Donald et al. [17] proved that the score-based fusion performed better in multiple features by experiments. Therefore, we utilize the score-based late fusion method for retrieval, as shown in Fig. 4.

In video-to-text retrieval, given a query video, we calculate similarity score with all sentences in the test dataset in both embedding spaces, and use the sum of the similarity scores for final ranking. Vice versa for text-to-video retrieval. This process can be expressed as:

$$S_{v-t}(v, t) = a_1 S_{s-t} + a_2 S_{d-t} \quad (7)$$

where  $S_{s-t}$  denotes the similarity score between video and text in static-text space,  $S_{d-t}$  denotes the similarity score between video and text in dynamic-text space. And  $S_{v-t}(v, t)$  represents the similarity sum score. It is worth noting that the similarity scores of



**Fig. 4** Score fusion of joint embedding spaces.  $S_{s-t}$  and  $S_{d-t}$  represent the similarity scores of modal samples in static-text and dynamic text spaces respectively.  $S_{v-t}$  is the score for final ranking

the multiple joint embedding spaces have the same weight in our paper.  $a_1$  and  $a_2$  are both 0.5.

The retrieval process of the MEJESF is shown in the Algorithm 2:

---

**Algorithm 2** Retrieval process of MEJESF.

---

**Input:** test dataset  $D_{test}$ .

- 1: Extract video features by pre-trained network:  $v = \{v_1, v_2, \dots, v_n\}$ .
  - 2: Extract text features by pre-trained network:  $s = \{w_1, w_2, \dots, w_m\}$ .
  - 3: Load the parameters of the optimal model for static-text space and encode videos and texts.
  - 4: Calculate the similarity of each text and video in the object text space, get  $S_{s-t}$ .
  - 5: Load the parameters of the optimal model for dynamic-text space and encode videos and texts.
  - 6: Calculate the similarity of each text and video in the object text space, get  $S_{d-t}$ .
  - 7:  $S_{v-t}(v, t) = S_{s-t} + S_{d-t}$  as the final score.
  - 8: Return the retrieval results.
- 

## 4 Experimental evaluation

### 4.1 Datasets

**MSR-VTT** [29] is the largest public dataset originally developed for video description generation tasks. It contains of 10k video clips and each clip is described by 20 natural language sentences. There are two ways to split the data for MSR-VTT in retrieval. Following the official data split, 6,513 videos are used for training and 2,990 videos for full testing. The rest of them (497 videos) are used for validation. The other split way is following [36]. The test set uses 1,000 randomly sampled text-videos pairs. Because the test data is randomly selected in this 1k division method, the cross-validation experiment is conducted as the final result. To evaluate the model more fairly, our experiments consist of these two data splits.

**MSVD** [3] is collected from YouTube videos, which have been used in several previous works on video captioning and action recognition. MSVD contains of 80K English descriptions for 1970 videos. For fair comparison, we follow the prior work [22], with 1,200,

100 and 670 video clips for training, validation, and test respectively. We randomly pick 5 sentences per test video.

## 4.2 Evaluation metrics

We report rank-based performance metrics including recall at rank ( $R@K$ -higher is better), Median rank ( $MedR$ -lower is better) and mean rank ( $MeanR$ -lower is better). Take video-to-text retrieval as an example,  $R@K$  ( $K=1,5,10$ ) reports the correct text ranks in the Top- $K$  retrieved results of the video query.  $MedR$  calculates the median rank of the first relevant sentence in the query results. In the same way,  $MeanR$  is the mean rank of all correct descriptions.

## 4.3 Implementation details

Our experiments are implemented based on PyTorch. We use a GTX 1080Ti GPU to train our model. We adopt Stochastic Gradient Descent (SGD) to optimize model's parameters with a mini-batch size of 128. The base learning rate is set as  $2e-3$ . The margin  $\alpha$  in the loss is 0.2. The tradeoff weight  $\lambda$  of the loss function is 0.5. We choose the best sum of recalls on the validation set as the final model.

## 4.4 Descriptions of the compared methods

Several state-of-art methods are compared in the experiments. Among these methods, CCA is a traditional method; VES, VES++ are based on ranking loss; W2VV, LJEMC, Dual-encoding, JSFusion, JMDV, ST, LJRv are DNN-based methods, and MoEE, HowTo100M are expert collaboration algorithms. These compared methods are briefly introduced as follows:

- VES [12]: unifies joint embedding models with multimodal neural language models.
- VES++ [8]: uses hard negative mining to learn visual-semantic embedding for cross-modal retrieval.
- W2VV [6]: learns to predict a visual feature representation from multi-scale sentence vectorization and multi-layer perceptron.
- LJEMC [20]: uses more video features in the retrieval framework including object, action, sound.
- Dual-encoding [7]: proposes a dual deep encoding network that encodes videos and queries into powerful dense representations of their own.
- JSFusion [36]: leverages hierarchical attention mechanisms that learn to promote well-matched representation patterns while prune out misaligned ones in a bottom-up manner.
- MoEE [18]: proposes a model with ability to handle missing input modalities during training.
- HowTo100M [19]: uses the collaborative expert model to collect different trained expert information, such as semantic embedding, ASR of video and OCR features.
- CCA [10]: learns a common space for different media types, which is able to maximize the correlation of them.
- JMDV [30]: proposes a unified framework consists of a compositional semantics language model, a deep video model and a joint embedding model to joint video and the corresponding text sentences.

- ST [13]:proposes a training sentence representation method skip though, which uses an encoder to model the current sentence and two independent autoregressive decoders to model the previous sentence and the next sentence respectively.
- LJRJ [22]:proposes a high-level concept word detector that can be integrated with any video-to-language models.

#### 4.5 Comparisons with state-of-the-art methods


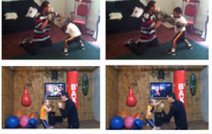

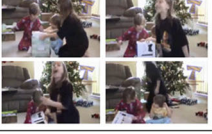
**MSR-VTT** Table 2 shows the experiment results and the performance comparison between our approach and the state-of-the-art methods on MSR-VTT dataset. We can observe that no matter of which data split method is using, our method outperforms the other approaches in terms of all metrics for both video-to-text retrieval and text-to-video retrieval. Compared with LJEMC, our algorithm has improved. It may be because that MEJESF emphasizes the fine-grained encoding of the same cue feature, which not only focuses on the global similarity, but also focuses on the influence of details on the retrieval. Dual-Encoding enriches the feature expression through multi-level encoding, but ignores the impact of multiple joint embedding spaces on the overall results and our algorithm takes both into account. Figure 5 shows the top-5 retrieved results for some exemplar queries. Figure 6 shows the qualitative results of text-to-video retrieval.

**MSVD** Table 3 shows the results of video-to-text retrieval on MSVD dataset and Table 4 shows the results of text-to-video retrieval task. It is clearly observed that our approach achieves the best performance with the  $R@1$  of 23.1 and 18.9 in text retrieval and video retrieval, which is about 12.2% and 7.4% higher than Dual-Encoding. However, our result of  $R@5$  is a little worse than Dual-Encoding. The possible reason is that we selected the best sum of recalls on the validation set as the optimal model for retrieval. The optimal model is the best overall, but does not mean the best locally. In general, MEJESF achieves competitive performance on all the evaluation metrics.

**Zero example retrieval** To verify the performance of the proposed model in zero example retrieval, we use the MSVD test set to assess the models trained on MSR-VTT. Table 5 shows the performance comparison. Our best result at  $R@1$  is 21.3 in video-to-text

**Table 2** Video-to-text and text-to-video retrieval results on MSR-VTT dataset


Method	Test-set	Video-to-text retrieval					Text-to-video retrieval				
		$R@1$	$R@5$	$R@10$	$MedR$	$Meand$	$R@1$	$R@5$	$R@10$	$MedR$	$Meand$
VSE [12]	full	7.7	20.3	31.2	28.0	185.8	5.0	16.4	24.6	47.0	215.1
VSE++ [8]	full	10.2	25.4	35.1	25.0	228.1	5.7	17.1	24.8	65.0	300.8
W2VV [6]	full	11.8	28.9	39.1	21.0	-	6.1	18.7	27.5	45.0	-
LJEMC [20]	full	12.5	32.1	42.4	16.0	134.0	7.0	20.9	29.7	38.0	213.8
Dual-Encoding [7]	full	13.0	30.8	43.3	15.0	-	7.7	22.0	31.8	32.0	-
MEJESF(ours)	full	13.8	33.8	46.5	13.0	103.2	7.7	22.2	32.1	30.0	152.9
JSFusion [36]	1k	-	-	-	-	-	10.2	31.2	43.2	13.0	-
MoEE [18]	1k	-	-	-	-	-	12.9	36.4	51.8	10.0	-
HowTo100M [19]	1k	-	-	-	-	-	14.9	40.2	52.8	9.0	-
MEJESF(ours)	1k	31.2	58.1	69.8	4.0	26.6	17.7	42.4	54.1	8.0	44.3

Query Video	Retrieval Result
	Rank 1: A man talks about the food he is putting on a plate. ✓ Rank 2: A man puts together a plate of food. ✓ Rank 3: A man is serving food in the plate with eggs carrot and cabbage. ✓ Rank 4: A man is preparing a food dish. ✓ Rank 5: Someone giving a recipe for a dinner dish. ✓
	Rank 1: Little kids training for boxing. ✓ Rank 2: People are boxing. ✓ Rank 3: A man boxing with his child. ✓ Rank 4: Fathers practicing boxing/martial arts with their children. ✓ Rank 5: Kids are sparring with boxing gloves on. ✓
	Rank 1: The girl sitting at her desk laughs. ✓ Rank 2: A video of a man and girl talking. ✓ Rank 3: An asian lady laughs in front of her computer before scowling at a guy. ✓ Rank 4: A group of people playing a video game together. ✗ Rank 5: A young woman singing a song for audience. ✗
	Rank 1: A girl is screaming loudly. ✓ Rank 2: Person showing off stroller. ✗ Rank 3: Children are opening christmas gifts. ✓ Rank 4: A kid celebrates a present opening. ✓ Rank 5: A girl is talking to another girl angrily. ✗

**Fig. 5** Text retrieval examples on MSR-VTT. For each query video, it shows top five retrieved sentences using our model. In the figure, the top-5 sentences retrieved from the 1th and 2nd videos are their ground-truth captions, and the top-5 sentences retrieved from the 3rd and 4th videos have wrong captions. Take the fourth retrieval result as an example, the second-ranked caption identifies “gift” as “stroller”; the fifth-ranked caption understands the little girl’s “happily talking” and “pleasant bouncing” movement as “angry performance”

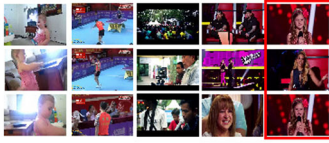
**Correct**

Q: A young girl petting a dog that is laying on a couch.




**Wrong**

Q: A girl is talking to the judges on a game show.




8th

Q: Three women are walking on the street and talking with a man.



Q: A woman and two kids are playing and exercising



17th

**Fig. 6** Video retrieval examples on MSR-VTT. The left shows top five videos retrieved using our method. Video in red bounding box is the ground truth video. Examples of retrieval errors are shown on the right. Taking the search in the upper right corner as an example, the top four retrieval results put the focus on “girl”, “game”, “talking”, and “show” respectively. There is lack of comprehensive consideration of these factors in the retrieval

**Table 3** Video-to-text retrieval results on MSVD dataset

Method	$R@1$	$R@5$	$R@10$	$MedR$	$Meand$
CCA [27]					245.3
JMDV [30]					224.1
ST [13]	2.99	10.9	17.5	77.0	241.0
LJRV [22]	9.85	27.1	38.4	19.0	75.2
VSE [12]	15.8	30.2	41.4	12.0	84.8
VSE++ [8]	21.2	43.4	52.2	9.0	79.2
W2VV [6]	17.9	-	49.4	11.0	57.6
Dual-Encoding [7]	20.6	42.8	58.8	8.0	38.9
MEJESF(ours)	23.1	48.1	60.3	6.0	36.0

retrieval and 15.3 in text-to-video retrieval for the cross-dataset scenario, which makes the improvement over the most recent report [7] by 13.9% and 20.5%.

Observing the experiment results, it is found that the result of video-to-text are always better than the result of text-to-video. It is because in the retrieval, the same video corresponds to multiple correct text descriptions, and each text description only corresponds to one correct video.

#### 4.6 Ablation studies

We conduct several ablation experiments on MSR-VTT to investigate the effectiveness of each component of our model.

**Influence of multi-grained encoding** Table 6 shows the result of investigating contribution of each module in coarse-grained and fine-grained manners. The 1st row methods are encoded by mean-pooling layer in both video feature and text feature. And the results of adding the gate embedding block to encode modal's coarse-grained feature are displayed in the second row. Then the results with encoding fine-grained feature just with biGRU are shown in the 3rd row. Furthermore, Li et al. [14] fused the features from different angles in image-text retrieval. We follow Li et al.'s approach, and fuse the coarse and fine-grained features of the same modal in joint embedding space, and the 5th row shows the result.

**Table 4** Text-to-video retrieval results on MSVD dataset

Method	$R@1$	$R@5$	$R@10$	$MedR$	$Meand$
CCA [27]					251.3
JMDV [30]					236.3
ST [13]	2.6	11.6	19.3	51.0	106.0
LJRV [22]	7.7	23.4	35.0	21.0	49.7
VSE [12]	12.3	30.1	42.3	14.0	57.7
VSE++ [8]	15.4	39.6	53.0	9.0	43.8
Dual-Encoding [7]	17.6	47.1	59.5	7.0	34.8
MEJESF(ours)	18.9	46.1	60.5	7.0	25.3

**Table 5** Performance of zero-example video retrieval, trained on MSR-VTT, tested on MSVD

Encoding Strategies	Video-to-text retrieval					Text-to-video retrieval				
	<i>R@1</i>	<i>R@5</i>	<i>R@10</i>	<i>MedR</i>	<i>Meand</i>	<i>R@1</i>	<i>R@5</i>	<i>R@10</i>	<i>MedR</i>	<i>Meand</i>
VSE [12]	15.4	31	42.4	19.0	128	11.0	28.6	39.9	18.0	48.7
VSE++ [8]	20.8	37.6	47.8	12.0	108.3	13.8	34.6	46.1	13.0	48.4
Dual-Encoding [7]	18.7	37.2	45.7	15.0	142.6	12.7	32.0	43.8	15.0	52.7
MEJESF(ours)	21.3	41.9	55.8	8.0	36.4	15.3	37.6	51.9	10.0	28.4

Through comparison experiments, it can be observed that our method (shown in the 4th row) which considers both global information and subtle details has better effect.

**Influence of different joint embedding spaces** The results of different joint embedding spaces are assessed in Table 7. The 1st, 2nd rows show the results using different video features. Each video feature is multi-grained encoded, and then mapped into a joint embedding space with text features using the modified loss. The 1st row and 2nd row also represent the retrieval accuracy of a single embedding space. Miechet et al. [18] proposed a method that connected multiple features of the media firstly and employed a single embedding space for retrieval. To test the effectiveness of multiple joint embedding spaces, we also perform similar experiments. Comparing the 1st row (static-text space), the 2th row (dynamic-text space) and 3th row (con(appearance - action)-text space), our proposed method achieves 17.9% improvement in *R@1* for text retrieval and 2.6% improvement for video retrieval compared to the best model among the single embedding space. Therefore, the score-based fusion strategy for joint embedding spaces is effective.

## 4.7 Further analyses

### 4.7.1 The tradeoff parameter

To clarify the effect of the tradeoff parameter  $\lambda$  in the loss function in Equation (6), we illustrate the performance curves with different values of  $\lambda$  in Fig. 7. From the figures, we can see that the performance of *R@1*, *R@5*, *R@10* all increase with the increase of  $\lambda$ , when  $\lambda$  varies in a range from 0.2 to 0.5. The value of *R@1*, *R@5*, *R@10* gradually decrease when  $\lambda$  is greater than 0.5. Therefore the best performance is achieved when  $\lambda$  is about 0.5.

**Table 6** Ablation study: impact of different encoding strategies on MSR-VTT

Encoding Strategies	Video-to-text retrieval					Text-to-video retrieval				
	<i>R@1</i>	<i>R@5</i>	<i>R@10</i>	<i>MedR</i>	<i>Meand</i>	<i>R@1</i>	<i>R@5</i>	<i>R@10</i>	<i>MedR</i>	<i>Meand</i>
coarse grained-mean_pooling	10.6	26.4	37.8	21.0	145.1	5.7	18.6	27.5	43.0	218.0
+gate embedding block	11.9	31.4	43.0	16.0	125.2	6.6	20.8	30.2	35.0	193.2
+fine grained										
+biGRU	11.7	30.2	40.9	16.0	126.2	6.8	20.9	30.4	34.0	196.9
+CNN(ALL)	13.8	33.8	46.5	13.0	103.2	7.7	22.2	32.1	30.0	152.9
CON(coarse and fine grained)	12.6	31.3	44.2	15.0	107.2	6.9	21.1	30.5	34.0	174.8



**Table 7** Ablation study: influence of different joint embedding spaces on MSR-VTT

Joint embedding spaces	Video-to-text retrieval					Text-to-video retrieval				
	<i>R</i> @1	<i>R</i> @5	<i>R</i> @10	<i>MedR</i>	<i>Meand</i>	<i>R</i> @1	<i>R</i> @5	<i>R</i> @10	<i>MedR</i>	<i>Meand</i>
Static-Text Space	11.0	27.6	37.7	21.0	164.8	6.6	19.5	28.1	41.0	191.6
Dynamic-Text Space	9.4	25.0	35.0	26.0	176.9	5.4	17.7	26.2	48.0	215.6
CON(appearance-action)-Text Space	11.7	30.1	41.8	16.0	116.2	7.5	22.1	32.0	31.0	157.0
All	13.8	33.8	46.5	13.0	103.2	7.7	22.2	32.1	30.0	152.9

### 4.7.2 Convergence in practice

We visualize the loss function of MEJESF in (6) with increasing the number of epochs in Fig. 8. It can be seen from the figure that the loss function of MEJESF converges on both static-text space and dynamic-text space.

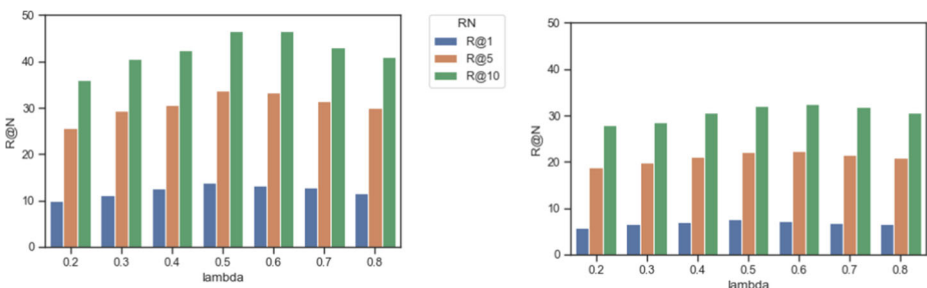
### 4.8 FLOPs

In the stage of training and learning feature representation of different modals, MEJESF mainly includes coarse-grained encoding module and fine-grained encoding module. In deep learning model, floating-point operations (FLOPs) [21] are often used to measure the complexity of the model.

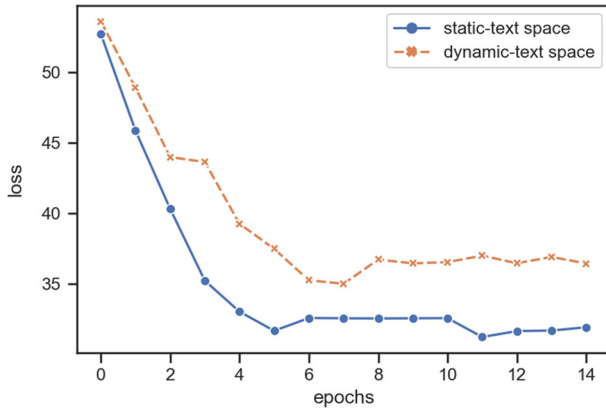
Coarse-grained encoding: coarse-grained encoding is mainly composed of a mean pooling layer, two fully connected layers and a sigmoid activation function. A full connection layer FLOPs is  $O(2 \times N_i \times N_o)$ , the two full connection layers are  $O(2 \times 2 \times N_i \times N_o)$ , where  $N_i$  and  $N_o$  is the number of input neurons and output neurons respectively.

Fine-grained encoding: in the process of fine-grained encoding, the model first passes through a biGRU network, and the biGRU network is composed of two GRU networks. Then the obtained matrix is passed through the convolution network to obtain the final feature representation. Therefore, the FLOPs of fine-grained encoding is  $O(2 \times (2 \times C_i \times k^2) \times C_o \times H \times W)$ .

where  $C_i$  is the number of input channels,  $k$  is the length of convolution kernel,  $C_o$  is the number of output channels,  $H \times W$  is the length and width of the output characteristic graph.



**Fig. 7** Experiments on influence of the  $\lambda$  in formula (7) on MSR-VTT. Similar results for other datasets



**Fig. 8** Convergence of MEJESF in practice

MEJESF includes the processing of video and text. Therefore, the FLOPs is the sum of fine-grained encoding of video, coarse-grained encoding of video, fine-grained encoding of text and coarse-grained encoding of text. The FLOPs of MEJESF as:

$$O\left(2\left(2 \times 2 \times N_i \times N_o + 2 \times \left(2 \times C_i \times k^2\right) \times C_o \times H \times W\right)\right)$$

## 5 Conclusion

In this paper, an end-to-end model for video-text cross-modal retrieval is presented. It consists of multi-grained feature encoding, joint embedding space fusing to reduce the semantic gap between diverse modals. In particular, we propose a novel dual encoding method which encodes the coarse-grained feature and fine-grained feature of modals. Considering interaction of features and hard sample mining, the loss function is modified. Furthermore, we build two joint embedding spaces and fuse them when retrieving. Extensive experiments demonstrate that the proposed model achieves promising performance improvements in both video-text cross-modal retrieval and zero-example video retrieval. We will work to improve the model to better complete the task of zero-example video retrieval in the future work.

**Acknowledgements** This work was supported in part by the National Natural Science Foundation of China under Project No. 62177015, in part by the Key-Area Research and Development Program of Guangdong Province No. 2019B111101001 and in part by the Science and Technology on Information System Engineering Laboratory No. WDZC 20205250410.

## Declarations

**Conflict of Interests** The authors declare that there is no conflict of interest regarding the publication of this article.

## References

1. Andrew G, Arora R, Bilmes J, Livescu K (2013) Deep canonical correlation analysis. In: the 30th International conference on machine learning, pp 1247–1255

2. Carreira J, Zisserman A (2017) Quo vadis, action recognition: a new model and the kinetics dataset. In: IEEE conference on computer vision and pattern recognition, pp 6299–6308
3. Chen DL, Dolan WB (2011) Collecting highly parallel data for paraphrase evaluation. In: the 49th annual meeting of the association for computational linguistics: human language technologies, proceedings of the conference, pp 190–200
4. Chi J, Peng Y (2018) Dual adversarial networks for zero-shot cross-media retrieval. In: the 27th international joint conference on artificial intelligence, pp 663–669
5. Dong J, Li X, Snoek CeesGM (2016) Word2visualvec: Image and video to sentence matching by visual feature prediction. arXiv:[1604.06838](https://arxiv.org/abs/1604.06838)
6. Dong J, Li X, Snoek CGM (2018) Predicting visual features from text for image and video caption retrieval. *IEEE Trans Multimed* 20(12):3377–3388
7. Dong J, Li X, Xu C, Ji S, He Y, Yang G, Wang X (2019) Dual encoding for zero-example video retrieval. In: the IEEE conference on computer vision and pattern recognition, pp 9346–9355
8. Faghri F, Fleet DJ, Kiros JR, Fidler S (2017) Vse++: Improved visual-semantic embeddings. arXiv:[1707.05612](https://arxiv.org/abs/1707.05612)
9. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: the IEEE conference on computer vision and pattern recognition, pp 770–778
10. Hodosh M, Young P, Hockenmaier J (2013) Framing image description as a ranking task: Data, models and evaluation metrics. *J Artif Intell Res* 47:853–899
11. Kim Y (2014) Convolutional neural networks for sentence classification. In: the 2014 conference on empirical methods in natural language processing, pp 1746–1751
12. Kiros R, Salakhutdinov R, Zemel RS (2014) Unifying visual-semantic embeddings with multimodal neural language models. arXiv:[1411.2539](https://arxiv.org/abs/1411.2539)
13. Kiros R, Zhu Y, Salakhutdinov R, Zemel RS, Urtasun R, Torralba A, Fidler S (2015) Skip-thought vectors. In: advances in neural information processing systems 28: annual conference on neural information processing systems 2015, pp 3294–3302
14. Li W, Zheng Y, Zhang Y, Feng R, Zhang T, Fan W (2020) Cross-modal retrieval with dual multi-angle self-attention. *J Assoc Inf Sci Technol* 72(1):46–65
15. Liu Y, Albanie S, Nagrani A, Zisserman A (2019) Use what you have: Video retrieval using representations from collaborative experts. arXiv:[1907.13487](https://arxiv.org/abs/1907.13487)
16. Markatopoulou F, Galanopoulos D, Mezaris V, Patras I (2017) Query and keyframe representations for ad-hoc video search. In: the 2017 ACM on international conference on multimedia retrieval, pp 407–411
17. Mc Donald K, Smeaton AF (2005) A comparison of score, rank and probability-based fusion methods for video shot retrieval. In: the 4th international conference on image and video retrieval, pp 61–70
18. Miech A, Laptev I, Sivic J (2018) Learning a text-video embedding from incomplete and heterogeneous data. arXiv:[1804.02516](https://arxiv.org/abs/1804.02516)
19. Miech A, Zhukov D, Alayrac J-B, Tapaswi M, Laptev I, Sivic J (2019) Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. arXiv:[1906.03327](https://arxiv.org/abs/1906.03327)
20. Miithun NC, Li J, Metz F, Roy-Chowdhury AK (2018) Learning joint embedding with multimodal cues for cross-modal video-text retrieval. In: the 2008 ACM on international conference on multimedia retrieval, pp 19–27
21. Molchanov P, Tyree S, Karras T, Aila T, Kautz J (2016) Pruning convolutional neural networks for resource efficient inference. arXiv:[1611.06440](https://arxiv.org/abs/1611.06440)
22. Otani M, Nakashima Y, Rahtu E, Heikkil J, Yokoya N (2016) Learning joint representations of videos and sentences with web image search. In: the european conference on computer vision, pp 651–667
23. Pan Y, Mei T, Yao T, Li H, Rui Y (2016) Jointly modeling embedding and translation to bridge video and language. In: the IEEE conference on computer vision and pattern recognition, pp 4594–4602
24. Peng Y, Huang X, Qi J (2016) Cross-media shared representation by hierarchical learning with multiple deep networks. In: the 25th international joint conference on artificial intelligence, pp 3846–3853
25. Peng Y, Qi J, Yuan Y (2018) Modality-specific cross-modal similarity measurement with recurrent attention network. *IEEE Trans Image Process* 27(11):5585–5599
26. Shen X, Shen F, Sun Q-S, Yang Y, Yuan Y-H, Shen HT (2016) Semi-paired discrete hashing: Learning latent hash codes for semi-paired cross-view retrieval. *IEEE Trans Cybern* 47(12):4275–4288
27. Socher R, Li F (2010) Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora. In: the IEEE conference on computer vision and pattern recognition, pp 966–973
28. Ueki K, Hirakawa K, Kikuchi K, Ogawa T, Kobayashi T (2017) Waseda meisei at trecvid 2017: Ad-hoc video search. In: the 2017 TREC video retrieval evaluation
29. Xu J, Mei T, Yao T, Rui Y (2016) Msr-vtt: A large video description dataset for bridging video and language. In: the IEEE conference on computer vision and pattern recognition, pp 5288–5296

30. Xu R, Xiong C, Chen W, Corso JJ (2015) Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In: the 29th AAAI conference on artificial intelligence, pp 2346–2352
31. Xu R, Li C, Yan J, Deng C, Liu X (2019) Graph convolutional network hashing for cross-modal retrieval. In: the 28th international joint conference on artificial intelligence, pp 10–16
32. Xu X, Lu H, Song J, Yang Y, Shen HT, Li X (2019) Ternary adversarial networks with self-supervision for zero-shot cross-modal retrieval. *IEEE Trans Cybern* 50(6):2400–2413
33. Xu X, Song J, Lu H, Yang Y, Shen F, Huang Z (2018) Modal-adversarial semantic learning network for extendable cross-modal retrieval. In: the 2018 ACM on international conference on multimedia retrieval, pp 46–54
34. Xue H, Chu W, Zhao Z, Cai D (2018) A better way to attend: Attention with trees for video question answering. *IEEE Trans Image Process* 27(11):5563–5574
35. Yang Y, Zhou J, Ai J, Bin Y, Hanjalic A, Shen HT, Ji Y (2018) Video captioning by adversarial lstm. *IEEE Trans Image Process* 27(11):5600–5611
36. Yu Y, Kim J, Kim G (2018) A joint sequence fusion model for video question answering and retrieval. In: the european conference on computer vision, pp 471–487
37. Yu Y, Ko H, Choi J, Kim G (2017) End-to-end concept word detection for video captioning, retrieval, and question answering. In: the IEEE conference on computer vision and pattern recognition, pp 3165–3173
38. Zhang J, Peng Y, Yuan M (2018) Sch-gan: Semi-supervised cross-modal hashing by generative adversarial network. *IEEE Trans Cybern* 50(2):489–502
39. Zhang X, Zhou S, Feng J, Lai H, Li B, Pan Y, Yin J, Yan S (2017) Hashgan: attention-aware deep adversarial hashing for cross modal retrieval. [arXiv:1711.09347](https://arxiv.org/abs/1711.09347)
40. Zhuo T, Cheng Z, Zhang P, Wong Y, Kankanhalli M (2019) Unsupervised online video object segmentation with motion property understanding. *IEEE Trans Image Process* 29:237–249

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.