Cross-Category Highlight Detection via Feature Decomposition and Modality Alignment

Zhenduo Zhang
Platform Technology Department, OVBU, PCG, Tencent, China
ericzdzhang@163.com

Abstract

Learning an autonomous highlight video detector with good transferability across video categories, called Cross-Category Video Highlight Detection (CC-VHD), is crucial for the practical application on video-based media platforms. To tackle this problem, we first propose a framework that treats the CC-VHD as learning category-independent highlight feature representation. Under this framework, we propose a novel module, named Multi-task Feature Decomposition Branch which jointly conducts label prediction, cyclic feature reconstruction, and adversarial feature reconstruction to decompose the video features into two independent components: highlight-related component and category-related component. Besides, we propose to align the visual and audio modalities to one aligned feature space before conducting modality fusion, which has not been considered in previous works. Finally, the extensive experimental results on three challenging public benchmarks validate the efficacy of our paradigm and the superiority over the existing state-of-the-art approaches to video highlight detection.

Introduction

Video highlight detection, which aims to automatically detect attractive moments within a video, has been a research hotspot in recent years. Video highlight detection has many downstream applications, including video summarization, video recommendation, and video editing. Despite the great successes in this field, the existing methods generally focused on training a highlight detector for a specific video category (e.g., surfing, skiing, parkour, etc.) while ignoring the transferability of a highlight detection model across different video categories (Gygli, Song, and Cao 2016; Hong et al. 2020; Badamdorj et al. 2021; Wei et al. 2022). The problem of learning a highlight detector with good transferability across video categories is called Cross-Category Video Highlight Detection (CC-VHD). The most related work in promoting the transferability of highlight detection is the work DL-VHD (Xu et al. 2021), treating the CC-VHD in an Unsupervised Domain Adaptation (UDA) (Pan and Yang 2010) way in which one adapts the knowledge learned from one labeled source domain (the source video category with supervision) to another one unlabeled target domain (the unsupervised target video category). The main issue of the proposed problem setting is that the highlight detector can be trained on only one specific category and transferred to other categories, which cannot fully utilize the annotations of the training data containing many video categories. In fact, it is a waste of resources for precious labeling data. Besides, when adopting the highlight detector in the actual scenario, it is hard and time-consuming to seek the best source categories for training.

Motivated by the weaknesses of the above approaches, we aim to overcome the poor transferability of a highlight detection model and fully utilize all the annotated training data to further promote the highlight detector’s performance. The video category can be seen as the biased attribute in video highlight detection, and the highlight estimation should be bias-independent to guarantee transferability. It is feasible to remove the category bias and obtain a category-independent highlight feature representation to promote transferability. Hence, we propose a paradigm to treat the CC-VHD as a problem of learning category-independent highlight feature representation, using annotated video segments of all categories, instead of treating it in an Unsupervised Domain Adaptation (UDA) way.

To implement this paradigm, we propose a novel module named Multi-task Feature Decomposition Branch (MFDB), which disentangles video features into two independent components: highlight-related component and category-related component. The MFDB module jointly conducts three tasks: label prediction, cyclic feature reconstruction, and adversarial feature reconstruction, to guarantee the compactness and independence of feature decomposition.

Previous works have done many works in exploring how to fuse the information from different modalities to promote the highlight detection (Hong et al. 2020; Badamdorj et al. 2021), while they rarely take the alignment of multi-modal features into consideration. The audio encoder and visual encoder in this task are usually pretrained from different video sources, and their feature spaces may be far away from each other. ALBEF (Li et al. 2021) and CLIP2TV (Gao et al. 2021) show that aligning different modalities before fusion will promote the performance of video retrieval and other downstream tasks, such as VQA and NLVR. This motivates us to align the visual and audio modalities to one aligned
feature space before fusion in the highlight detection task.

Finally, we conduct extensive experiments on popular video highlight benchmarks to validate the effectiveness and superiority of our paradigm.

To sum up, the contributions of this work are:

• We first propose to formulate the Cross-Category Video Highlight Detection as a problem of learning category-independent highlight representation.

• We propose a novel module Multi-task Feature Decomposition Branch which jointly conducts label prediction, cyclic feature reconstruction, and adversarial feature reconstruction to guarantee the compactness and independence of highlight feature and category feature.

• We propose to align the visual and audio modalities to an aligned feature space before fusion to promote the performance of highlight detection.

• Extensive experiments on popular video highlight benchmarks demonstrate the effectiveness of our paradigm and the superiority over the other existing approaches.

Related Work

Video Highlight Detection

Video highlight detection is a task that assigns each video segment a highlight score. The existing works can be divided into two categories according to the way of supervision. The supervised methods (Gygli, Song, and Cao 2016; Jiao et al. 2018; Sun et al. 2016; Yao, Mei, and Rui 2016) need the highlight annotations of all segments in a video. Since the annotation for training videos is a time-consuming and laborious task, weakly-supervised approaches have played an important role in recent years. Different effective weak supervisory signals have been employed to define highlights, such as the frequent occurrence of specific segments within a video category (Panda et al. 2017a; Potapov et al. 2014a; Yang et al. 2015b), the video duration (Xiong et al. 2019) and the segment bag information (Hong et al. 2020). From the perspective of the training task, the training task of video highlight detection can be divided into two classes: classification task (Rochan et al. 2020) and ranking task (Sun et al. 2016; Garcia del Molino and Gygli 2018; Jiao et al. 2018; Gygli, Song, and Cao 2016; Wang et al. 2020). For the classification task, the network tries to classify video clips according to whether they are highlights or not. It is also popular to adopt a ranking training task, where we train the ranking network to rank highlight clips higher than non-highlight clips. The transferability of highlight detection has attracted the attention of researchers recently. The most related work in promoting the transferability of highlight detection is the work (Xu et al. 2021), treating the task in an Unsupervised Domain Adaptation (UDA) way and adapting the knowledge learned from the source video category to the target video category.

Feature Decomposition

Feature Decomposition (Zhang et al. 2019; Bao et al. 2018; He et al. 2019; Singh, Ojha, and Lee 2019; Tran, Yin, and Liu 2017) models the explanatory factors from diverse data variation, which has drawn considerable attention in many fields, such as face recognition, person re-identification, and generative tasks. Previous works utilized annotated attribute data to decompose representations into identity-related and identity-independent information (pose, viewpoint, illumination, etc.) for recognition and identification tasks. (He et al. 2019) learned invariant feature representations of heterogeneous face images by minimizing Wasserstein distance between cross-modality distributions. (Hou, Li, and Wang 2021) decomposed the feature representation into age-related feature and identity-related feature by minimizing the mutual information between two components.

Approach

Figure 1 illustrates the overall framework of our proposed approach. The framework inputs are the sampled frames from the video and the audio signal. We denote the training batch size is $B$ and the number of sampled frames per video is $N_f$. The frameset of the $k_{th}$ video is denoted as $I_k$. We feed the $I_k$ to the video encoder, a vision transformer (Vaswani et al. 2017; Dosovitskiy et al. 2020), and obtain the feature of $N_f$ frames. Then the $N_f$ frame features are processed by the temporal transformer to capture the relationship among the sampled frames and we get the $N_f$ frame embeddings $\{v_k^i\}_{i=1}^{N_f}$. We note that the parameter set of the overall visual feature extractor is $\theta_V$.

We feed the audio signal of the $k_{th}$ video to the audio encoder and a projection MLP to obtain the audio embeddings and project them to have the same dimension as the visual embeddings. Since the audio signals have different lengths, the number of audio embeddings differs within the batch. To perform batch training, we need to ensure that the number of audio embeddings is the same for each video. Therefore, we first determine the minimum number $N_a$ of audio embeddings for each video in the current batch. Then we uniformly sample the audio embeddings for each video to obtain $N_a$ audio embeddings. The final audio embeddings of the $k_{th}$ video is denoted as $\{a_k^i\}_{i=1}^{N_a}$. We note that the parameter set of the overall audio feature extractor is $\theta_A$.

We attempt to fuse the visual and audio modalities via the Merged Attention Model (Dou et al. 2022), where the visual and audio features are simply concatenated together, then fed into a single transformer (Vaswani et al. 2017) block. We note that the parameter set of the Merged Attention Model is $\theta_M$.

The Multi-task Feature Decomposition Branch (MFDB) can decompose a feature $z_n$ into two independent components: the category-related feature $c_n$ and the highlight-related feature $h_n$. We employ two MFDBs on top of the visual encoder and the Merged Attention Model, and the two MFDBs do not share parameters.

The Visual-Audio Alignment procedure aligns the visual and audio modalities to an aligned feature space before fusion, which is implemented via a contrastive learning paradigm.
Multi-task Feature Decomposition Branch

The core of our approach to tackle the Cross-Category Highlight Detection problem is to decompose the original feature of the $m_{th}$ video, $z_m$, into two independent components: the category-related feature $c_n$ and the highlight-related feature $h_n$. When inferring, we only utilize the highlight-related feature $h_n$ to obtain good transferability across video categories. We denote the feature decomposition module as $\Pi = (\Pi_c, \Pi_h)$, where $\Pi_c$ and $\Pi_h$ are modeled by two-layer MLPs. Given an original feature $z_n$, $\Pi$ decomposes $z_n$ into two components $c_n$ and $h_n$ as $(c_n, h_n) = \Pi(z_n) = (\Pi_c(z_n), \Pi_h(z_n))$.

We consider that the original feature $z_n$ is properly decomposed by $\Pi$ if the following four constraints are satisfied:

$C_1$: $c_n$ and $h_n$ can reconstruct the original feature $z_n$ via proper function $\Pi$ to guarantee the compactness of decomposition.

$C_2$: We suppose that the highlight feature from the $m_{th}$ video ($m \neq n$) is $h_m$. The $n_{th}$ video and $m_{th}$ video have the same highlight label $y^h_m = y^h_n$. Then $c_n$ and $h_m$ can reconstruct the original feature $z_n$ via $\Pi$ to guarantee the highlight invariance to category.

$C_3$: We cannot neither reconstruct $c_n$ from $h_n$ via a function $\tau_c$ nor reconstruct $h_n$ from $c_n$ via a function $\tau_h$, which can guarantee the independence of the two components.

$C_4$: $c_n$ can predict the category label and $h_n$ can predict the highlight label to guarantee $c_n$ and $h_n$ encode the category information and highlight information, respectively.

**NOTE:** We assume that the parameter set of the feature extractor that generates $z_n$ is $\theta_z$. If the MFDB is employed in the visual branch, $\theta_z$ is the parameter set of visual encoder $\theta_V$. If the MFDB is employed in the fusion branch, $\theta_z$ is the parameter set of visual encoder, audio encoder and the Merged Attention Model, $\theta_z = \{\theta_V, \theta_A, \theta_M\}$.

![Figure 1: An overview of the proposed approach.](image1.png)

We implement the above 4 constraints via four optimization objectives, where $E[\cdot]$ calculates the mean loss within the batch.

To implement $C_1$, we propose to employ a cyclic reconstruction task as Equation 1. $\Pi'$ is modeled via a two-layer MLP and its input is the concatenation of $c_n$ and $h_n$.

$$
\min_{\theta_z, \theta_{\Pi}'} \mathcal{L}_{\text{reco}}^{\text{self}} = E \left[ \|z_n - z_n^{\text{self}}\|^2 \right]
$$

(1)

To implement $C_2$, we propose to employ an cyclic reconstruction task as Equation 2, where $m \neq n$ and $y^h_m = y^h_n$.

$$
\min_{\theta_z, \theta_{\Pi}'} \mathcal{L}_{\text{reco}}^{\text{cross}} = E \left[ \|z_n - z_n^{\text{cross}}\|^2 \right]
$$

(2)

To implement $C_3$, we propose to employ an adversial reconstruction task as Equation 3. $\tau_c$ and $\tau_h$ are both modeled.

![Figure 2: An overview of the proposed Multi-task Feature Decomposition Branch.](image2.png)
by two-layer MLPs.

\[
\min_{\theta_c, \theta_h} \max_{\tau_c, \tau_h} L_{adv}^{adv} = E \left[ \| c_n - \tau_c (h_n) \|_2 + \| h_n - h_{\tau_h} \|_2 \right] \\
= E \left[ \| c_n - \tau_c (h_n) \|_2 + \| h_n - \tau_h (c_n) \|_2 \right] \quad (3)
\]

At training time, we seek the parameters \( \{ \theta_c, \theta_h \} \) to maximize the adversarial reconstruction loss \( L_{adv}^{adv} \) to make \( c_n \) and \( h_n \) independent to each other, while simultaneously seeking \( \{ \tau_c, \tau_h \} \) to minimize \( L_{adv}^{adv} \) to reconstruct one component from another one. In order to optimize Equation 3, we exploit Gradient Reversal Layer (GRL) (Ganin et al. 2016) to connect \( \{ \theta_c, \theta_h \} \) and \( \{ \tau_c, \tau_h \} \). In the forward pass and inverts the gradient sign during the backward pass, pushing the parameters to maximize the output loss. If we denote the GRLs of the category feature and highlight feature as \( g_c \) and \( g_h \), the optimization of Equation 3 can be rewritten as Equation 4:

\[
\min_{\theta_c, \theta_h} \max_{\tau_c, \tau_h} L_{adv}^{adv} = E \left[ \| c_n - \tau_c (g_h (h_n)) \|_2 \right] + \quad (4)
\]

\[
E \left[ \| h_n - \tau_h (g_c) (c_n) \|_2 \right]
\]

To implement \( C_4 \), we try to predict the category label using \( c_n \) and the highlight label using \( h_n \). \( \sigma_c \) and \( \sigma_h \) denote the corresponding category classifier and the highlight classifier. \( \sigma_c \) is a multi-class classifier, and the classification loss for the category is calculated as Equation 5.

\[
\min_{\theta_c, \theta_h, \sigma_c, \sigma_h} L_{cls} = E \left[ - y_n^c \log \hat{y}_n^c \right] = E \left[ - y_n^c \log \sigma_c (c_n) \right] \quad (5)
\]

\( \sigma_h \) is a binary-class classifier, and the output of it can either represent the highlight score in the ranking way or the probability of highlight class in the classification way. In order to better capture the highlight relationship between different video clips, we adopt an additional self-attention module (SA) (Badamgarj et al. 2021) when predicting the highlight label. The input to the SA module is a set of highlight features from different videos, i.e. \( \{ h_{n0}, \ldots, h_{n0+N_G} \} \) and the output highlight logits are \( \{ \hat{y}_{n0}^h, \ldots, \hat{y}_{n0+N_G}^h \} = \sigma_h \left( \text{SA} \left( \{ h_{n0}, \ldots, h_{n0+N_G} \} \right) \right) \), \( N_G \) is the length of highlight feature set. The category classification loss is Equation 6.

\[
\min_{\theta_c, \theta_h, \sigma_c, \sigma_h} L_{cls} = E \left[ - y_n^h \log \hat{y}_n^h \right] \quad (6)
\]

The overall loss of MFDB module is Equation 7.

\[
L_{MFDB} = L_{self} + L_{cross} + L_{adv}^{adv} + L_{cls} + L_{cls}^{cls} \quad (7)
\]

Visual-Audio Alignment

The audio and visual encoders are usually pretrained from different sources in practice. For instance, the audio encoder PANN (Kong et al. 2020) is pretrained on AudioSet (Gemmeke et al. 2017) and the visual encoder ViT (Dosovitskiy et al. 2020) is pretrained on Kinect-400 (Carreira and Zisserman 2017) dataset. The gap in the source domain leads to the fact that their feature spaces may be far from each other. The success in ALBEF (Li et al. 2021), CLIP2TV (Gao et al. 2021) motivates us to align the visual and audio modalities to an aligned feature space before fusion in the highlight detection task. We average the visual and audio embeddings to get the visual and audio representation of the video.

\[
v_k = \frac{1}{N_f} \sum_{i=1}^{N_f} v_i \quad \text{and} \quad a_k = \frac{1}{N_a} \sum_{i=1}^{N_a} a_i \quad \text{. The Visual-Audio Alignment loss is calculated by Equation 8, where } \tau \text{ is the learnable temperature widely used in contrastive learning.}
\]

\[
L_{Align} = \frac{1}{2} E \left[ \frac{v_i a_i}{\tau} - \frac{v_i a_i}{\tau} + \epsilon_i ^{v_i a_i / \tau} \right] \quad (8)
\]

Overall Loss Function

We add two different MFDB modules on the visual branch and the fusion branch, and the losses of the two MFDB modules are denoted as \( L_{MFDB}^{v} \) and \( L_{MFDB}^{i} \). Hence, the loss of the framework is shown in Equation 9, where \( \{ \lambda_i \}_{i=1}^{3} \) are the weights of the losses.

\[
L = \lambda_1 L_{MFDB}^{v} + \lambda_2 L_{MFDB}^{i} + \lambda_3 L_{Align} \quad (9)
\]

Experiments

Experiment Settings and Compared Methods

We evaluate our approaches on three popular benchmark datasets, i.e., YouTube Highlights (Sun et al. 2016), TVSum (Song et al. 2015) and CoSum (Chu, Song, and Jaimes 2015), for video highlight detection. YouTube Highlights contains six event-specific categories, and there are approximately 100 videos in each event. TVSum is an available video summarization benchmark dataset, collected from YouTube and crawled by an event-specific queried tag. TVSum consists of 50 videos grouped into ten categories (5 videos per category). CoSum contains 51 videos covering 10 events.

We use the ViT-32 (Dosovitskiy et al. 2020) pretrained on the Kinect-400 (Carreira and Zisserman 2017) as the video encoder to extract visual features of the sampled frames and use the PANN (Kong et al. 2020) pretrained on the AudioSet (Gemmeke et al. 2017) to extract the audio embeddings of audio clips. We sample 16 frames uniformly from the video frames of a video. We train our model using Adam, with a learning rate of \( 1 \times 10^{-4} \). The weights of the losses in Equation 9 are \( \{ \lambda_i \}_{i=1}^{3} = \{ 1.0, 0.5, 1.0 \} \), which are selected by GridSearch strategy. The size of the highlight feature set fed into the self-attention layer in the MFDB module, which is the \( N_G \) mentioned above, is set to 16. The source code will be released.

We compare our methods with the following state-of-the-art video highlight detection baselines on three datasets. We introduce six weakly-supervised approaches, i.e. RRAE(V) (Yang et al. 2015a), SG(V) (Mahasseni, Lam, and Todorovic 2017), DSN(V) (Panda et al. 2017b), VESD(V) (Cai et al. 2018), LIM-s(V) (Xiong et al. 2019) and MINI-Net(V) (Hong et al. 2020) for comparison. Besides, nine supervised video highlight detection methods, i.e. Video2GIF(V) (Gygli, Song, and Cao 2016), LSVM(V) (Sun et al. 2016), KVS(V) (Potapov et al. 2014b), DPP(V) (Gong et al. 2014), sLSTM(V) (Zhang et al. 2019) etc.
et al. 2016), SM(V) (Gygli, Grabner, and Van Gool 2015), JVAL(VA) (Badamdorj et al. 2021), DL(V) (Xu et al. 2021) and PLD(V) (Wei et al. 2022) are also involved for comparison. The “V” and “VA” in parentheses after each method indicate whether the corresponding method is based on visual modality or visual-audio multi-modality.

Comparisons with the State-of-the-art Methods

We compare our approach with the current state-of-the-art methods on three popular benchmarks, Youtube Highlights, TVSum and CoSum, which are shown in Table 1, Table 2 and Table 3. Firstly, we compare our overall framework that utilizes both visual and audio modalities with the current state-of-the-art methods. On the Youtube Highlights Dataset, our overall framework gains the best performance in four of six categories. The average mAP in all the categories increases from 0.730 in PLD to 0.749. On the TVSum Dataset, our approach obtains the best performance on seven of ten categories, and the average top-5 mAP raises from 0.771 in PLD to 0.815. Similarly, our model surpasses or equal the state-of-the-art methods on seven of ten categories on the CoSum Dataset. The average top-5 mAP score is also improved, from 0.946 in PLD to 0.961. Although our model does not gain the best performance in some specific categories, our model can be employed to detect highlights for all video classes instead of training one expert model for each category, which contributes to the good transferability across video categories.

Since some of the SOTA methods for comparison, such as DL (Xu et al. 2021) and PLD (Wei et al. 2022), only use the visual modality rather than both visual and audio modalities, we also utilize a single visual modality to make fair comparisons with the current methods. When only using visual modality, the Visual-Audio Alignment, the Merged Attention Model, and the MFDB module for fused features are not included in our framework, and we only use the visual encoder and the MFDB module for the visual feature. From Table 1, our model uses a single visual modality and gain 0.737 average mAP over all categories, which also outperforms the other current methods significantly. Likewise, our model using a single visual modality can also exceed the other state-of-the-art methods on the average top-5 mAP on TVSum and CoSum datasets. The average top-5 mAP on TVSum and CoSum are 0.806 and 0.954, respectively.

The comparison between our model using a single visual modality and the one using both modalities demonstrates that the audio modality can help improving the highlight detection performance on the three datasets.

Ablation Studies of MFDB Module

Necessity of MFDB The Multi-task Feature Decomposition Branch aims to decompose the original feature into two nearly independent components: category-related and highlight-related features. Next, we will verify the necessity of using the Multi-task Feature Decomposition Branch. For a fair comparison, we replace the MFDB module with a simple Multi-Task Learning Branch (MTLB), which jointly predicts category labels and highlight labels and cannot conduct feature decomposition. The highlight features are also processed by a self-attention module to capture the relationship among clips. The details of the multi-task learning head are shown in Figure 3. It is proper to use the MTLB, which only lacks the feature decomposition process compared with the MFDB. Besides, multi-task learning is a typical transfer learning approach for learning better representation.
of the category, the highlight feature will have better transferability across categories, and the overall performance in all categories will be improved. From Table 4, the average highlight detection performance overall categories on three benchmarks can be significantly promoted when utilizing either a single visual modality or visual-audio modalities. The results sufficiently prove our assumption that learning the category-independent highlight feature can help improve the overall performance over different categories.

Except for the quantitative experiments to compare the performance of the highlight detector, we also conduct a qualitative comparison between MFDB and MTLB. We randomly sample 40 to 60 video clips from each category in Youtube Highlights Dataset. The t-SNE visualization results of their highlight features is shown in Figure 4. In Figure 4(a), the highlight features are still entangled with the category though they have a certain level of highlight discrimination. Within each category, the highlight clip features and the non-highlight clip features roughly distribute in two separate feature spaces. However, some highlight clip fea-

<table>
<thead>
<tr>
<th>Topic</th>
<th>Weakly supervised</th>
<th>Supervised</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RRAE (V)</td>
<td>LIM-s (V)</td>
</tr>
<tr>
<td>dog</td>
<td>0.49</td>
<td>0.59</td>
</tr>
<tr>
<td>gymnastics</td>
<td>0.35</td>
<td>0.41</td>
</tr>
<tr>
<td>parkour</td>
<td>0.50</td>
<td>0.67</td>
</tr>
<tr>
<td>skating</td>
<td>0.25</td>
<td>0.58</td>
</tr>
<tr>
<td>skiing</td>
<td>0.22</td>
<td>0.48</td>
</tr>
<tr>
<td>surfing</td>
<td>0.49</td>
<td>0.51</td>
</tr>
<tr>
<td>Average</td>
<td>0.383</td>
<td>0.644</td>
</tr>
</tbody>
</table>

Table 1: Highlight detection results (mAP) of weakly-supervised and supervised methods on the YouTube Highlights dataset.
Figure 5: Visualization for Highlight Detection

![Visualizations for Highlight Detection](image)

Table 4: Ablation study of the necessity of MFDB. The average mAP is used in the YouTube Highlights, and the average top-5 mAP is used in the TVSum and CoSum.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Visual MTLB</th>
<th>Visual MFDB</th>
<th>Visual-Audio MTLB</th>
<th>Visual-Audio MFDB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Youtube Highlights</td>
<td>0.623 ↑</td>
<td>0.737 ↑</td>
<td>0.652 ↑</td>
<td>0.749 ↑</td>
</tr>
<tr>
<td>TVSum</td>
<td>0.646 ↑</td>
<td>0.806 ↑</td>
<td>0.673 ↑</td>
<td>0.815 ↑</td>
</tr>
<tr>
<td>CoSum</td>
<td>0.865 ↑</td>
<td>0.954 ↑</td>
<td>0.874 ↑</td>
<td>0.961 ↑</td>
</tr>
</tbody>
</table>

Table 5: Ablation study of where to apply MFDB. The average mAP is used in the YouTube Highlights, and the average top-5 mAP is used in the TVSum and CoSum.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Fusion Only</th>
<th>Fusion &amp; Visual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Youtube Highlights</td>
<td>0.742 ↑</td>
<td>0.749 ↑</td>
</tr>
<tr>
<td>TVSum</td>
<td>0.808 ↑</td>
<td>0.815 ↑</td>
</tr>
<tr>
<td>CoSum</td>
<td>0.957 ↑</td>
<td>0.961 ↑</td>
</tr>
</tbody>
</table>

Table 6: Ablation study of the Visual-Audio Alignment. The average mAP is used in the YouTube Highlights, and the average top-5 mAP is used in the TVSum and CoSum.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>w/o $\mathcal{L}_{Align}$</th>
<th>with $\mathcal{L}_{Align}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Youtube Highlights</td>
<td>0.740 ↑</td>
<td>0.749 ↑</td>
</tr>
<tr>
<td>TVSum</td>
<td>0.811 ↑</td>
<td>0.815 ↑</td>
</tr>
<tr>
<td>CoSum</td>
<td>0.958 ↑</td>
<td>0.961 ↑</td>
</tr>
</tbody>
</table>

We conducted an ablation study to explore the effect of the Visual-Audio Alignment process, and the experiments are illustrated in Table 6. From this table, we can verify that aligning the visual and audio features before fusing them can promote the learning of highlight detection. Instead of fusing the features from two different modalities and source domain, the visual-audio alignment help to pull the feature spaces of visual feature and audio feature to be close to each other first.

Conclusion

In this work, we first attempt to learn the highlight video detector with good transferability via learning the category-independent highlight representation, which can make full use of the annotated video segments of all categories. To implement this idea, we propose the Multi-task Feature Decomposition Branch, which disentangles the features into highlight and category features, which are nearly independent. Besides, we propose to pull the feature spaces of different modalities to be close before fusion, which benefits the multimodal representation learning of the video.
References


3532


