SVGraph: Learning Semantic Graphs from Instructional Videos

Madeline C. Schiappa  
Center for Research in Computer Vision  
University of Central Florida  
Orlando, FL  
madelineschiappa@knights.ucf.edu

Yogesh S. Rawat  
Center for Research in Computer Vision  
University of Central Florida  
Orlando, FL  
yogesh@crcv.ucf.edu

Abstract—In this work, we focus on generating graphical representations of instructional videos. We propose a self-supervised, interpretable approach that does not require any annotations for graphical representations, which would be expensive and time consuming to collect. We attempt to overcome ‘black box’ learning limitations by presenting Semantic Video Graph or SVGraph, a multi-modal approach that utilizes narrations for semantic interpretability of the learned graphs. SVGraph 1) relies on the agreement between multiple modalities to learn a unified graphical structure with the help of cross-modal attention, and 2) assigns semantic interpretation with the help of Semantic Assignment, which captures the semantics from narration. We perform experiments on multiple datasets and demonstrate the interpretability of SVGraph in semantic graph learning.

Index Terms—multimodal learning, deep learning, interpretability, graph learning, video understanding

I. INTRODUCTION

The internet today hosts millions of instructional videos which can be made useful when analyzed automatically. However, designing an approach that learns an interpretable representation of videos without annotation is one of the biggest challenges. Most of the existing research in video domain has focused on tasks such as action detection [7], [44], temporal activity detection [13], [26] and retrieval [14], [26], [49]. However, these models are trained on large datasets with annotations specific to the task. This leads to issues of annotation cost [4], [52], annotation bias [10], [36], lack of generalization [22], [48], and lack of robustness [16], [17]. Learning without annotations is often done by self-supervised learning (SSL), where a model is pre-trained on large-scale datasets without the need of labels [21], [37] using a learning objective which is derived from the training samples itself [3], [23], [31]. Using multiple modalities with SSL for video has been found effective in downstream tasks related to visual question-answering, retrieval, and action recognition [2], [30], [42]. However, none of these existing approaches have shown interpretability or a semantic understanding of the video.

In image domain, scene graphs have been proposed to provide interpretable learning in the form of graphs for entities and their relations [8], [51]. The challenge with these approaches is they require in-depth annotations that describe different entities and their relations to each other. Because they are also in the image domain, they do not directly translate to video with the addition of the temporal dimension. Obtaining frame-by-frame annotations for videos is very challenging and requires extensive resources in terms of time, cost and computation. This inspired us to develop an approach which learns interpretable graphical structure from videos without the availability of such annotations while taking advantage of longer duration of video.

We propose SVGraph which utilize multiple modalities to learn an interpretable representation. The multiple modalities will not always complement each other but sometimes they will also share information. For example, if a user is performing a task, the narration/audio will supplement that information (Figure 1). More specifically in the case of instructional videos, a user is more likely to narrate sub-activities when they appear on screen. This may indicate that visual stimuli at that time is important and should be attended to. Motivated by this, we utilize cross-modal attention which facilitates one modality to attend the other. This enables enhancement of each modality with the help of other before their integration.

SVGraph utilize cross-modal embeddings and graph neural networks [15] to model a video using multiple signals while generating an interpretable version of the learned representations. It learns joint-embeddings between visual, audio and textual signals obtained from automatic speech recognition (ASR) using cross-modal attention, without the need of any annotations. It uses multi-modal embeddings to build an interpretable graph with the help of convolutions for neighborhood-based message-passing and our Semantic Assignment. We make the following contributions in this work,

• We introduce SVGraph, a novel approach for building a graphical representation of long instructional videos.
• SVGraph is trained using a self-supervised objective using multiple modalities without the need of annotations.
• We propose a novel Semantic Assignment mechanism.
which provides interpretability to the learned graph.

We perform our experiments on 4 different datasets and show both qualitative as well as quantitative evaluation validating the contributions of various components of SVGraph and also demonstrate its effectiveness in semantic graph learning.

II. RELATED WORKS

A. Learning Complex Activities in Videos

Several approaches extend deep clustering methods [5] where cluster assignments act as labels for images to sub-action detection in video. Each time segment from a video is assigned to a cluster and that cluster assignment acts as a sub-action label [25], [38], [46]. A problem with these approaches is that the sub-action labels per video are not representing sub-actions at a global/dataset level, but rather at the local/video level. Work by [25] addressed this by assuming sub-activities will propose a new task for these methods, to learn an interpretable embeddings, focusing on the joint-embedding space [29], uses contrastive comparisons [9] between different modality from the same video [32], [34], [42]. These approaches use [37]. One such approach is to predict whether a signal came C. Multi-Modal Representation Learning

and relations embeddings through iterative updates, which we graph, reducing complexity. These architectures learn object optimized graph generation by not assuming a fully-connected diffi
cult to directly apply to video. Some works [27], [51] have [50]. These approaches are often highly spatial based [12], [50]. The most common approaches focus on refining the activities. [6], [19]. While these approaches are able to capture both local and global information, they are not interpretable.

B. Scene Graphs

Scene graphs are methods that model the interactions between objects in a scene using dense annotations. They often use a pipeline of object detection for nodes, graph generation and then iterative updates of relationships between those nodes [1], [12]. The most common approaches focus on refining the initial node embeddings extracted from object detection [8], [27], [50]. These approaches are often highly spatial based [12], [50] and the graphs are initialized as a fully-connected structure. These approaches ignore temporal relationships, making it difficult to directly apply to video. Some works [27], [51] have optimized graph generation by not assuming a fully-connected graph, reducing complexity. These architectures learn object and relations embeddings through iterative updates, which we extend to video using SVGraph.

C. Multi-Modal Representation Learning

Several works have employed a cross-modal learning objective to guide respective embeddings into a joint space [37]. One such approach is to predict whether a signal came from the same video [32], [34], [42]. These approaches use visual embeddings to predict whether a given text and/or audio embedding from a set belongs to the respective video. Another uses contrastive comparisons [9] between different modality embeddings, focusing on the joint-embedding space [29], [35], [41]. These aim to maximize the similarity between an embedding from one modality to the embedding of another. We propose a new task for these methods, to learn an interpretable graph from the multi-modal embeddings.

III. METHOD

Given a video \( V \in \mathbb{R}^{T \times C \times H \times W} \) where \( C \) is channels, \( T \) is time and \( H \times W \) are height and width of the frames, we want to learn a graph \( G \) that comprises of nodes \( NE \). Each video is first divided into short clips with a sequence of frames which are encoded using a 3D CNN resulting in visual encodings \( M_v \). Each video’s aligned audio is extracted as Mel Spectrograms and encoded using a 2D CNN resulting in audio features \( M_a \). Given an encoded video \( M_v \in \mathbb{R}^{T \times C} \) and an encoded audio \( M_a \in \mathbb{R}^{T \times C} \), we first learn a multi-modal embedding \( Z \in \mathbb{R}^{T \times C} \). Next, words \( W \) are encoded using Word2Vec from the narration in a video are used to initialize latent nodes \( N \). Given a series of encoded words \( N \in \mathbb{R}^{T \times N \times C} \), we attend to \( N \) using \( Z \). The semantic node embeddings \( NE \in \mathbb{R}^{T \times N \times C} \) are further trained to learn the overall graph embedding, refining the semantic node embeddings \( NE \). Readout on the refined node embeddings \( NE \) is performed resulting in both a compact graph embedding \( G \) and indices that map where the most relevant node embeddings were in the original set of nodes. Using Semantic Assignment, these indices are used to map the original words \( W \) to the maximally relevant node embeddings \( NE \). Using the graph embedding for each video, we use a triplet loss between the original video graph embedding \( G_v \), an augmented version \( \hat{G}_i \) and a randomly selected video \( G_j \) to ensure consistency in learning. To generate \( \hat{G}_i \), each input video \( V \) is augmented and encoded to generate \( \hat{M}_v \). An overview of the proposed approach is shown in Figure 2.

A. Cross-Modal Learning

SVGraph uses multiple modalities to encourage learning to be focused on the most relevant activities in a video. We therefore utilize attention mechanisms for our multi-modal embeddings, attending to one modality via the other. In Figure 3 is the two-branch attention mechanism where each branch uses one modality \( M_1 \) to attend to the other \( M_2 \). Each branch’s procedure is similar to self-attention in [45] but with the focus of cross-attention between two signals. Input for this module are the modalities’ feature representations extracted from their respective encoders, either a 3D CNN or a 2D CNN. These features comprise of \( T \) time segments and a feature vector for each segment. We start by linearly projecting each modal feature vector in their respective branches. Then attention values \( \alpha \) are used to correlated the embeddings which are calculated by taking the dot-product between \( M_1 \) and \( M_2 \). Then \( \alpha \) is multiplied by the linear projection of the modality we are attending to in the respective branch resulting in a refined feature embedding for each modality \( Z_1 \) and \( Z_2 \). This results in the original embeddings being refined where the points of greater similarity between the two modalities are emphasized. The output of each branch, \( Z_1 \) and \( Z_2 \), is then aggregated resulting in a final output which is a joint embedding of the two modalities \( Z \).

B. Semantic Attention

In order to make our graph interpretable, we learn semantically relevant features for our nodes. We start by initializing nodes \( N \) of the graph by using extracted features from the
with the same time segments as the other modalities and \( N \) we are attending to our semantic nodes. This approach is similar to a one branch attention mechanism as \( \alpha \). These values measure the correlation between the semantic nodes extracted from narration and the multi-modal features where larger values indicate stronger similarity. We then attend to each node in \( N \) with the attention values in \( \alpha \) resulting in \( \hat{N} \). This results in updates to original embedding where the nodes of greater similarity between the multi-modal embeddings are emphasized.

C. Message-Passing

In order to learn relationships between the semantic node features \( \hat{N} \), we utilize message-passing. Convolutional layers can be used in a Message-Passing Neural Network (MPNN) framework to learn from graphs [15]. We use iterative convolutional layers to allow message-passing between neighbouring nodes and time segments. This will encourage the model to learn interactions between nodes and between a node over time and the maximally relevant nodes and time segments. We use depthwise-convolutions [11] to split the input and filter into groups, convolve each input with their respective filter and finally stack the convolved outputs together [19]. This procedure is shown in Algorithm 1.

D. Semantic Assignment

To make SVGraph interpretable, we propose Semantic Assignment. During message-passing of the learned semantic node embeddings \( n \in \hat{N} \) with feature vectors \( NE_n \in \hat{N} \), indexes \( I \) refer to the maximally relevant nodes for the current instructional activity during the max pool operation. In order to represent interactions between objects, we interpret verbs/states as edges and their respective features as edge features. To do this, we must first map our selected nodes \( N_I \) back to the original words so we can use their semantic meaning to assign nodes to entities or actions/states. An overview of this mechanism is shown in Figure 4.

The extraction of \( I \) is shown in Algorithm 1 and the Semantic Assignment is shown in Algorithm 2. Semantic Assignment uses the indices output from Algorithm 1 to map backwards and retrieve the selected words \( W_I \) from \( W \). With the final set of words, we can use their respective features \( \hat{N} \) to build graphs and to assign semantic meaning. For directed graphs it is important to consider the difference between features over time \( \hat{N} \in \mathbb{R}^{T \times N} \). To maintain time \( t \in T \), we treat each occurrence

\[
\text{Algorithm 1: Message Passing}
\]

\[
\text{Input: } \hat{N} \in \mathbb{R}^{T \times N \times C} : \text{Initialized Nodes} \\
\text{Output: } I, \hat{N} : \text{Selection Indices, Nodes} \\
\text{for } l = 1 \text{ to } L \text{ do} \\
\text{1: Convolve Features: } \hat{N} = \text{CONV}(\hat{N}) \\
\text{2: MP over } T [20]: \hat{N} = \text{TimeCONV}(\hat{N}) \\
\text{3: MP over } N [19]: \hat{N} = \text{NodeCONV}(\hat{N}) \\
\text{4: MaxPool: } \hat{N}, I = \text{MaxPool}_{t \in T, n \in N}(\hat{N}) \\
\text{end} \\
\text{return } I, \hat{N}
\]
of a word at different time segments as separate nodes \( n_t \in T \). Multiple occurrences of a word in one time segment are treated as the same by aggregating their respective activation values as:

\[
\hat{N}E_{n_t} = \sum_{n \in N_t} NE_{n},
\]

where \( N_t \) are all nodes that occur in time \( t \). For undirected graphs, we want to compare nodes across all time segments. To do this, we calculate the average feature vector for each word over all occurrences as,

\[
NE_{n} = \frac{\sum_{t,T} \sum_{n \in N_t} \hat{N}E_{n_t}}{\sum_{t,T} \sum_{n \in N_t} 1}.
\]  

For both cases, the feature represents the activation of attention values for each node, which also indicates its relevance.

### E. Objective function

In order to train the model, a readout procedure is performed on the graph to get an aggregate graph representation for the instructional video. This readout procedure takes the refined semantic node embeddings \( \hat{N}E \) and aggregates them by a series of convolutions and max pooling operations to focus on the maximally relevant time segments and concepts. In order to train in a self-supervised fashion, we perform augmentation [9] on the input video \( M_v \) to generate a positive sample \( \hat{M}_v \). Using our framework we extract a graph embedding from readout \( \hat{G}_t \) using the original \( M_v \) and \( \hat{G}_t \) from the augmented version \( \hat{M}_v \). We use a triplet loss [18] where \( \hat{G}_t \) is the positive sample and \( \hat{G}_j \) is a randomly selected other video from the same batch as the negative sample. This learning objective aims to maximize the distance between the negative pair and minimize the distance between the positive pair while promoting discriminant feature learning.

### IV. Experiments

#### A. Experimental Setup

For visual embeddings we use an I3D model initialized with weights pre-trained on ImageNet [7]. For the audio embedding branch we use a 2D CNN initialized with weights pre-trained on acoustic scenery [24]. We adapted augmentations from [9] for video to generate positive samples for the triplet loss during training. We trained for 50 epochs using stochastic gradient descent with a momentum of 0.9, weight decay of \( 1 \times 10^{-7} \), and an cyclical learning rate [39] that used a base learning rate of 0.01 and maximum learning rate of 0.1.

1) Datasets: We perform our experiments on four different datasets. **HowTo100M** [30] is a large-scale dataset containing narrated instructional videos collected from YouTube. We chose a subset of videos that are under the activity category of ‘home and garden’, ‘hobbies and crafts’, and ‘computers and electronics’. Text is extracted from the ASR, or manually provided narration, downloaded from YouTube. In total, there are 19,662 videos used for training our approach. **COIN** [43] is comprised of YouTube instructional videos. These videos have task labels, allowing us to make comparisons between different and similar tasks. The COIN dataset does not provide video narrations, therefore we used the video ids to retrieve ASR from YouTube. This resulted in 1,382 videos for training and 94 videos for testing. **YouCook2** [53] is a large, task-oriented, instructional video dataset for cooking. To collect the narrations, we again used the video ids to download ASR from YouTube resulting in 662 videos for training and 238 for testing from 89 cooking recipes. **UCF101** [40] is an action recognition dataset. These clips are short and do not have long-complex activities. These videos are annotated with action classes and therefore we use this dataset for our ablation experiments. We focused on videos that have audio signal available resulting in 4,839 videos for training and 1,944 videos for testing.

2) Metric: To evaluate the quality of the learned graphs we measure the ability of our model to minimize the distance between videos from similar categories and maximize the distance between videos from different categories. To do this, we use a metric adopted from the natural language processing (NLP), the rouge-n metric. We use the rouge-1 [28], or the unigram overlap, between the nodes of a pair of videos. The complex activity recognition task on YouCook2 is evaluated using Precision@K [53] for K=5 and 10.

#### B. Graphical Analysis

1) Quantitative Analysis: We compare the overlap of nodes in graphs from videos of the same and different tasks using the rouge-1. When there is complete overlap, rouge-1 equals 1, and when there is no overlap it equals 0. Figure 7 shows the model is learning to pick maximally relevant nodes that are applicable to the specific task. While there will be some overlap based on words being unique to the task, the model may learn more generic terms that carry over all tasks. For all tasks, the average rouge-1 node overlap for COIN for same-task graphs is 86.67 and 35.87 for different-task graphs. For YouCook2, rouge-1 for same-tasks graphs is 82.53 and 37.47 for different-task graphs. The higher similarity for same tasks compared to different for overall tasks, and the similar pattern found when observing specific tasks as shown in Figure 7, indicates that the model is learning to differentiate on key concepts between tasks.
2) **Qualitative Analysis:** Fig. 5 shows graphs learned for corresponding instructional videos. We observe that the key concepts appear in the video as important segments of instruction. For example in Fig. 5, the instructor in the video recommends using a cloth if the shower head is too tight to unscrew using a plier. The colors around sample frames are also around the nodes that illustrate that relationship. The most important concepts of the instruction appear to be repeated in the graph, demonstrating the model’s ability to select nodes that are most relevant. Fig. 6 show visual comparisons between videos from YouCook2 on how to make margarita pizza and pancakes respectively. We observe that the graphs for making pizza on the right in Fig. 6 have more content in common and the structure is more similar when compared with a different recipe such as ‘making pancakes’ on the left in Fig. 6.

An example of an undirected graph is in Fig. 8 visualizing an unlabeled activity from the HowTo100M. Node features are aggregated over time, therefore nodes that are activated in more time segments have a greater importance to the overall activity. We treat all words as nodes where edges are the strength of a relationship based on the cosine similarity. Node importance is visualized by the size of the node, where the greater the relevance the larger the node. The graph shown in Fig 8 visualizes the most important relationships towards...
Fig. 7: A visualization comparing videos from different tasks in both the YouCook2 dataset and the Coin dataset. The values shown are the average number of shared nodes between tasks. The darker the color, the greater the similarity.

Fig. 8: Undirected graphs for two instructional videos. These graphs aggregate node embeddings over time to generate an undirected graph. The larger the node size, the greater the importance. The thicker the edges, the stronger the relationships between two nodes.

Fig. 9: Aggregate undirected graphical representations of videos that either instruct the same, or different, task. Videos on the top row reflect a disjoint, two-clustered graphical representation while videos in the bottom row reflect an integrated graphical representation.

we see more connections and overlap between the node features while in the different activities there is absolutely no overlap or similarity between the nodes. This shows SVGraph is learning to distinguish between different activities at a global level.

3) User Study: To further evaluate the quality of learned graphs, we performed an user study. We surveyed three different aspects of our work: 1) graph to video matching, 2) video to graph matching, and 3) graph quality. We first presented users with a graph and asked them to choose which of the given videos the graph best represented with an option that the graph represented none of the provided videos. We then did the reverse and presented them with a video and a choice of multiple graphs to match with an option of no graph. Finally, we presented graph and video pairs and asked users to rate how well the respective graph represented its video on a scale from 1-10. Video and graph samples were randomly selected from the Coin and Youcook2 dataset. For each question, four options were presented to the user. The results for the 30 participants are shown in Figure 10. Users were better able to match a graph to a video but struggled with the reverse. Overall, the correct match was made a majority of the time. When users were asked to rate the quality of randomly chosen graph-video pair from a scale of 1 to 10, the average rating was 6.

C. Ablations and Discussion

1) Self-Supervised Objective: We experimented with multiple self-supervised loss functions to train our approach: cosine-triplet loss [18], angular-cosine triplet loss [47], and a noise-contrastive estimation (NCE) [33]. Each loss uses an augmented version of the frames as a positive sample while all other samples in the batch as negative samples. We also tested a cross-modal NCE loss that treat the visual and audio branch separately when attending to text. The resulting joint embedding between audio+text NEa and video+text NEv are used as positive samples for the NCE loss. Figure 11 shows a visualization of learned features for sub-activities in the COIN dataset extracted from each model. The cosine triplet loss shows the most distinct groupings of the activities while the cross-modal shows the most separation of different activities. Without the cross-modal positive samples, the NCE loss alone does not show a good separation of different tasks.

2) Video and Audio Joint Embedding: To better understand if using both audio and video will improve learning, we trained SVGraph up to the visual-audio joint embedding VA on the YouCook2 dataset, predicting on the overall video activity. We used a clip length of 128 and a fixed word length of 15. We
TABLE I: Ablations for audio and visual features on YouCook2 dataset precision on recipe. Including audio features improves learning.

| Method                  | Precision |
|-------------------------|-----------|
| SVGraph w/Video         | 6.5% 12.6%|
| SVGraph w/Audio         | 7.1% 13.7%|
| SVGraph w/Video+Audio   | 9.1% 16.1%|

TABLE II: Ablations on learning joint-embeddings for action-recognition on UCF101. Using cross-modal attention on visual and audio features improves accuracy.

| Method                      | Accuracy |
|----------------------------|----------|
| I3D [7]                    | 33.2%    |
| Self-Attention [45]        | 63.2%    |
| One-Branch Attn. Sum       | 21.1% 29.1% |
| One-Branch Attn. Multi     | 30.9% 41.3% |
| One-Branch Attn. Concat    | 68.2% 78.7% |

compare results to a video only prediction using I3D [7]. The results in Table I indicate that using both audio and video improve performance compared to using one over the other.

3) Cross-Modal Attention Mechanism: We use UCF101 [40] which is annotated with action classes for this ablation. We only used videos that contained audio, resulting in 4893 for training and 1944 for testing. Visual features and audio features are extracted in the same way as prior experiments. For efficiency, we reduce clip lengths to $t = 16$ with a smaller batch size of 8. A joint-embedding is learned through either multiplication, summation, concatenation, a one-branch attention focused on attending to video $M_v$ or our proposed cross-modal attention. Because our cross-modal attention combines the attended outputs of the two branches, we experimented with the different variations. The one-branch attention module was used where video features $M_v$ were the query and the audio features $M_a$ were the key. It was also used for self-attention on $M_v$ only. The resulting joint-embedding between visual and audio features is then passed through a FC layer to predict activity. The results in Table II further indicate that using both audio and video improves performance and cross-modal attention with concatenation performed.

4) Global vs. Local Representation: To analyze how local or global the learned representations are between tasks, we combined the nodes and node embeddings of two videos that share the same instructional task according to their HowTo100M task description. Figure 9 shows two aggregated graphical representations, each of two videos that are labelled to be performing the same task. The top row of Figure 9 shows videos described as how to make a stuffed giraffe, but on inspection, one is instructing users on a “Design Deluxe Sewing Studio Playset” and the other is providing a review on “My Baby’s Heartbeat Bear (Giraffe)”. When observing their combined graphical representation, there are two distant clusters that formed, further supporting their difference. The bottom row of Figure 9 shows videos described as how to make berry card and both videos were instructing that activity. When observing their combined graphical representation, there is a more intertwined graphical representation. While our approach is instance-based, there appears to be global learning of complex activities based on the aggregate graphical representation.

V. CONCLUSION

We propose a new task, representing instructional videos in semantically meaningful graphical form without the use of annotations. To solve this problem, we propose SVGraph, a framework to address this challenge that uses multiple-modalities from visual, sound and text in video. We incorporated cross-modal attention to improve the learning of joint-embeddings between these modalities. We proposed a novel technique Semantic Assignment to make these representations semantically interpretable. While it is a challenging problem, we demonstrate its feasibility which opens up an interesting research direction in video understanding.

ACKNOWLEDGEMENT

This research is based upon work supported by KAIROS Program No. FA8750-19-2-1004. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of DARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

REFERENCES

[1] Agarwal, A., Mangal, A., et al.: Visual relationship detection using scene graphs: a survey. arXiv preprint arXiv:2005.08045 (2020)
[2] Alwassel, H., Mahajan, D., Korbar, B., Torresani, L., Ghanem, B., Tran, D.: Self-supervised learning by cross-modal audio-video clustering. Advances in Neural Information Processing Systems 33 (2020)
[3] Benaim, S., Ephrat, A., Lang, O., Mosseri, I., Freeman, W.T., Rubinstein, M., Irani, M., Dekel, T.: Speednet: Learning the speediness in videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
[4] Cai, L., Xu, X., Liew, J.H., Foo, C.S.: Revisiting superpixels for active learning in semantic segmentation with realistic annotation costs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2021)
[5] Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018)
[6] Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised learning of visual features by contrasting cluster assignments. Advances in Neural Information Processing Systems 33 (2020)
[7] Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)
[8] Chen, T., Yu, W., Chen, R., Lin, L.: Knowledge-embedded routing network for scene graph generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2019)

[9] Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. PMLR (2020)

[10] Chen, Y., Joo, J.: Understanding and mitigating annotation bias in facial expression recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021)

[11] Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2017)

[12] Dai, B., Zhang, Y., Lin, D.: Detecting visual relationships with deep relational networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)

[13] Ding, X., Wang, N., Li, J., Gao, X.: Weakly supervised temporal action localization with segment-level labels. In: Chinese Conference on Pattern Recognition and Computer Vision (PRCV). Springer (2021)

[14] Duan, H., Zhao, Y., Xiong, Y., Liu, W., Lin, D.: Omni-supervised webly-supervised learning for video recognition. In: European Conference on Computer Vision. Springer (2020)

[15] Gilmer, J., Schoenholz, S.S., Riley,PF.,Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry (2017)

[16] Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., et al.: The many faces of robustness: A critical analysis of out-of-distribution generalization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021)

[17] Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations. Proceedings of the International Conference on Learning Representations (2019)

[18] Hermans*, A., Beyer*, L., Leibe, B.: In Defense of the Triplet Loss for Person Re-Identification. arXiv preprint arXiv:1703.07737 (2017)

[19] Hussein, N., Gavves, E., Smールders, A.W.M.: VideoGraph: Recognizing minutes-long human activities in videos. ArXiv (2019)

[20] Hussein, N., Gavves, E., Smールders, A.W.: Timeception for complex action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2019)

[21] Jing, L., Tian, Y.: Self-supervised visual feature learning with deep neural networks: A survey (2020)

[22] Kim, D., Saito, K., Oh, T.H., Plummer, B.A., Sclaroff, S., Saenko, K.: Cds: Cross-domain self-supervised pre-training. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (October 2021)

[23] Knights, J., Harwood, B., Ward, D., Vanderkop, A., Mackenzie-Ross, O., Moghadam, P.: Temporally coherent embeddings for self-supervised video representation learning. In: 2020 25th International Conference on Pattern Recognition (ICPR) (2021)

[24] Koutini, K., Eghbal-zadeh, H., Widmer, G.: Receptive-field-regularized cnn variants for acoustic scene classification. In: Detection and Classification of Acoustic Scenes and Events (2019)

[25] Kukleva, A., Kuehne, H., Sener, F., Gall, J.: Unsupervised learning of action classes with continuous temporal embedding (2019)

[26] Kipfnikl, O., Wei, X., Rigoli, G.: You only watch once: A unified cnn architecture for real-time spatiotemporal action localization (2019)

[27] Li, Y., Ouyang, W., Zhou, B., Shi, J., Zhang, C., Wang, X.: Factorizable net: an efficient subgraph-based framework for scene graph generation. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)

[28] Lin, C.Y., Hovy, E.: Automatic evaluation of summaries using n-gram co-occurrence statistics. In: Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (2003)

[29] Ma, S., McDuff, D., Song, Y.: Unpaired image-to-speech synthesis with multimodal information bottleneck. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2019)

[30] Miech, A., Zhukov, D., Alayrac, J.B., Tapaswi, M., Laptev, I., Sivic, J.: HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips (2019)

[31] Misra, I., Zitnick, C.L., Hebert, M.: Shuffle and Learn: Unsupervised Learning Using Temporal Order Verification. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Computer Vision – ECCV 2016. Springer International Publishing, Cham (2016)

[32] Mithun, N.C., Li, J., Metze, F., Roy-Chowdhury, A.K.: Learning joint embedding with multimodal cues for cross-modal video-text retrieval. ICMR 2018 - Proceedings of the 2018 ACM International Conference on Multimedia Retrieval (2018)

[33] Mnih, A., Kavukcuoglu, K.: Learning word embeddings efficiently with noise-contrastive estimation. In: Advances in neural information processing systems (2013)

[34] Owens, A., Efros, A.A.: Audio-visual scene analysis with self-supervised multisensory features. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)

[35] Peyre, J., Laptev, I., Schmid, C., Sivic, J.: Detecting unseen visual relationships using analogies. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2019)

[36] Rodrigues, F., Pereira, F.: Deep learning from crowds. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 32 (2018)

[37] Schiappa, M.C., Rawat, Y.S., Shah, M.: Self-supervised learning for Person Re-Identification. arXiv preprint arXiv:2001.06464 (2022)

[38] Sener, F., Yao, A.: Unsupervised Learning and Segmentation of Complex Activities from Video. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2018)

[39] Smith, L.N.: Cyclic learning rates for training neural networks. In: 2017 IEEE winter conference on applications of computer vision (WACV). IEEE (2017)

[40] Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)

[41] Sun, C., Baradel, F., Murphy, K., Schmid, C.: VideoBERT: A joint model for video and language representation learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2019)

[42] Tang, Y., Ding, D., Rao, Y., Zheng, Y., Zhang, D., Zhao, L., Lu, J., Zhou, J.: COIN: A large-scale dataset for comprehensive instructional video analysis. CoRR abs/1903.02574 (2019)

[43] Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2018)

[44] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems 30 (2017)

[45] VidalMata, R.G., Scheirer, W.J., Kukleva, A., Cox, D., Kuehne, H.: Joint visual-temporal embedding for unsupervised learning of actions in untrimmed sequences (2021)

[46] Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W.: Cosface: Large margin cosine loss for deep face recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2018)

[47] Wang, J., Lan, C., Liu, C., Ouyang, Y., Qin, T., Lu, W., Chen, Y., Zeng, W., Yu, P.: Generalizing to unseen domains: A survey on domain generalization. IEEE Transactions on Knowledge and Data Engineering (2022)

[48] Xie, S., Sun, C., Huang, J., Tu, Z., Murphy, K.: Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification (2018)

[49] Xu, D., Zhu, Y., Choy, C.B., Fei-Fei, L.: Scene graph generation by iterative message passing. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2017)

[50] Yang, J., Lu, J., Lee, S., Batra, D., Parikh, D.: Graph-r-cnn for scene graph generation. In: Proceedings of the European conference on computer vision (ECCV) (2018)

[51] Yang, L., Zhang, Y., Chen, J., Zhang, S., Chen, D.Z.: Suggestive annotation: A deep active learning framework for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer (2017)

[52] Zhou, L., Xu, C., Corso, J.J.: Towards automatic learning of procedures from web instructional videos. In: AAAI Conference on Artificial Intelligence (2018)