Informedia@TRECVID 2018

Video to Text Description

Jia Chen¹, Shizhe Chen², Qin Jin² and Alexander Hauptmann¹ ¹Carnegie Mellon University, ²Renmin University of China

In this section of the notebook, we present our system in the TRECVID Video to Text description generation task. The optimization target is critical to train the encoder-decoder based video captioning models. However, there are two main limitations of the most widely used cross-entropy (CE) function as the training target, namely exposure bias and mismatched targets in training and testing. Therefore, we propose to utilize the reinforcement-learning algorithm with different rewards to improve the video captioning performance, which has achieved substantial gains over the CE-trained baselines.

Ad-hoc Video Search with Discrete and Continuous Representations

Po-Yao Huang, Junwei Liang, Vaibhav, Xiaojun Chang and Alexander Hauptmann

Carnegie Mellon University

We present two different approaches we developed for cross-modal retrieval in the Ad-hoc video search (AVS) task. Our system is a fully-automatic systemutilizing no in-domain data nor annotation. We jointly utilize representations in the discrete semantic space learned from multiple mutually exclusive source domains as well as continuous representations in the textual-visual joint-embedding space. We encode textual queries and videos in these spaces and perform search and retrieval. We achieved 12.59 inferred average precision (IAP) on the AVS 2016 validation set and achieved 8.7 IAP and ranked 2nd in the 2018 AVS task.

Activities in Extended Video

Jia Chen, Jiang Liu, Junwei Liang, Ting-Yao Hu, Wenhe Liu, Wei Ke, Wayner Barrios, Haroon Idrees, Donghyun Yoo, Yaser Sheikh, Ruslan Salakhutdinov, Kris Kitani, Dong Huang, and Alexander Hauptmann

Carnegie Mellon University,

We present a generic event detection system for the SED task of TRECVID 2014. It consists of two parts: the retrospective system and the interactive system. The retrospective system uses STIP, MoSIFT and Improved Dense Trajectories as low level features, and uses Fisher Vector encoding to represent shots generated by sliding window approach. Linear SVM is used to perform event detection. To improve performance, we applied several spatial schemas to generate the fisher vectors in our experiments. For the interactive system, we applied a general visualization scheme for all the events and a temporal locality based search method for user feedback utilization. Among primary runs of all teams, our retrospective system ranked 1st for 3 of 7 events in terms of actual DCR.

Informedia@TRECVID 2018 Video to Text Description

Jia Chen¹, Shizhe Chen², Qin Jin² and Alexander Hauptmann¹ ¹Carnegie Mellon University, ²Renmin University of China

Abstract

In this notepaper, we present our system in the TRECVID Video to Text description generation task. The optimization target is critical to train the encoder-decoder based video captioning models. However, there are two main limitations of the most widely used cross-entropy (CE) function as the training target, namely exposure bias and mismatched targets in training and testing. Therefore, we propose to utilize the reinforcement learning algorithm with different rewards to improve the video captioning performance, which has achieved substantial gains over the CE-trained baselines.

1 Introduction

The video captioning task aims to automatically generate a natural language sentence to describe the video content, which is one the of ultimate goals of video understanding. Based on the encoder-decoder framework [1] and the recent large-scale video captioning datasets [2], the video captioning task has achieve great breakthroughs in recent years [3].

However, most of the models are optimized with maximum likelihood estimation on the training dataset, which utilizes the cross-entropy (CE) function as the training target. Such training target contains two main limitations, namely exposure bias and target mismatching. The exposure bias is that only the predicted sequences are available in the testing phase which are unseen in the training phase. The target mismatching is that the evaluation metrics of captions such as CIDEr [4] etc. are not in accordance with the CE targets for training the captioning system.

In order to overcome the two limitations, reinforcement learning algorithms have been proposed to optimize the captioning systems such as Self-critical sequence training [5] and have achieved the state-of-the-art performance in image captioning tasks. The reinforcement learning algorithm treats the captioning process as an action sequence. Each action is to generate a word and the target is to achieve high rewards on the automatic caption evaluation metrics.

In this work, we explore reinforcement learning algorithms with different rewards to optimize video captioning systems, including the accuracy reward and a novel diversity reward. The direct accuracy reward is a weighted average of automatic metrics (Bleu, Meteor, Rouge, Cider - BMRC) to provide more robust evaluation on the quality of the generated captions. Although the BMRC reward improves the accuracy of the predicted captions, it narrows down the sentence space and loses diversity. Therefore, we propose to fuse it with a diversity reward to embrace both accuracy and diversity from the reinforcement learning. Our approach has achieved substantial improvements over the CE baselines on the Trecvid dataset, which demonstrates the effectiveness of the proposed rewards for reinforcement learning.

2 Methodology

In this section, we first describe the structure of our video captioning model, and then introduce the reinforcement learning based training approach.

2.1 Video Captioning Model

Mean-pooling Caption Model (MP) is the basic encoder-decoder model for video captioning, which consists of a multimodal encoder and vanilla LSTM decoder. The multimodal encoder fuses the averaged image and motion features into a video-level multimodal representation x. Then the LSTM decoder generates word sequences based on x as follows:

$$h_t = f(h_{t-1}, w_{t-1}; \theta_d) \text{ for } t = 1, \dots, N_w$$
(1)

where f is the LSTM update function [6], h_t is the state of LSTM, w_t is the t-th word and θ_d is the parameter in LSTM. Then the probability of the groundtruth word can be expressed as:

$$\Pr(w_t | \mathbf{x}, w_0, \dots, w_{t-1}) = \operatorname{Softmax}(W_d h_t + b_d)$$
(2)

where W_d, b_d are parameters to be learned.

Temporal Attention Caption Model (TA) learns to focus on the relevant temporal frames to generate the word. Assuming $\{v_1, ..., v_n\}$ are the visual features in different frames. We compute the visual context $\phi_t(v)$ via the following attention mechanism to predict the *t*-th word:

$$\phi_t(v) = \sum_{i=1}^N \alpha_i^{(t)} v_i \tag{3}$$

$$\alpha_i^{(t)} = \exp\{e_i^{(t)}\} / \sum_{j=1}^n \exp\{e_j^{(t)}\}$$
(4)

$$e_i^{(t)} = w^{\mathrm{T}} \mathrm{tanh}(W_a h_{t-1} + U_a v_i + b_a)$$
 (5)

where w, W_a, U_a and b_a are the parameters. Then $\phi_t(v)$ is concatenated with the previous word embedding as the input of the LSTM decoder:

$$h_t = f(h_{t-1}, [w_{t-1}; \phi_t(v)]; \theta_d) \text{ for } t = 1, \dots, N_w$$
(6)

2.2 Reinforcement Learning Optimization

In this section, we introduce the proposed rewards for reinforcement learning optimization.

Accuracy reward: We utilize a weighted average of different automatic caption metrics as the accuracy reward for the predicted sequences $r = \sum_{i} w_i r_i$, where r_i , $i = \{1, 2, 3, 4\}$ denotes the Bleu@4, METEOR, Rouge and CIDEr respectively. The training goal is to minimize the negative of the expected reward as follows:

$$L_{accuracy} = -\mathbb{E}_{s \sim \Pr(\theta)}[r] \tag{7}$$

Based on the self-critical REINFORCE algorithm proposed in [5], we could calculate the gradient of the model and optimize the policy for word predictions.

Diversity reward: The $L_{accuracy}$ reward improves the caption accuracy but leads to narrow exploration spaces for word predictions. Therefore, we fuse a novel diversity reward to increase the diversity of the sentence expressions. We utilize $d(s_i, s_j)$ to compute the expressive distance of two sentences s_i and s_j . It does not need to involve any kind of semantic level measurement and simple gram counting such as BLEU suffices. The goal of the diversity reward is to increase the expressive distance among the generated captions, which is:

$$L_{diversity} = -\mathbb{E}_{s_i, s_j \sim \Pr(\theta)}[d(s_i, s_j)] \tag{8}$$

The accuracy and diversity reward are complementary with each other to generate video captions.

3 Experiments

We employ the TGIF video captioning dataset as our training set [2], and TRECVID VTT 2017 as our validation set to select the best models. Table 1 summarizes the statistics of our training and validation set.

For the video representation, we extract features from different modalities including Resnet200 feature [7] from image modality and I3D feature [8] from motion modality.

Table 1: Statistics of the training set and validation set used in our work.

	# videos	# captions	# unique words	avg. sent length
training	101,413	125,019	12,392	10.56
validation	1,880	5,270	5,660	14.08

Table 2: Captioning performance of different models on the Trecvid16 and Trecvid17 validation set.

		Trecvid16			Trecvid17		
	target	BLEU4	Meteor	CIDEr	BLEU4	Meteor	CIDEr
	CE	8.24	15.01	40.05	7.10	12.42	27.55
vanilla	accuracy	8.94	15.60	46.86	7.76	13.17	31.32
	accuracy ¹	-	-	-	7.68	13.58	31.80
	accuracy+diversity ¹	-	-	-	8.06	13.85	32.53
attention	CE	9.33	15.26	44.02	7.63	12.47	28.88
	accuracy	9.22	15.79	50.10	7.41	12.98	32.11

Table 2 presents the captioning performance on the Trecvid16 and Trecvid17 datasets for validation. We can see that the reinforcement learning algorithms with the proposed rewards significantly outperform the CE baselines for different captioning structures. The testing performance on the Trecvid18 dataset is shown in Table 3.

Table 3: Captioning performance of different models on the Trecvid18 testing set.

model	target	train set	BLEU	Meteor	CIDEr
vanilla	accuracy+diversity	TGIF+Trecvid16	0.024	0.231	0.416
attention	accuracy	TGIF	0.018	0.221	0.408

References

- [1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.
- [2] Yuncheng Li, Yale Song, Liangliang Cao, Joel Tetreault, Larry Goldberg, Alejandro Jaimes, and Jiebo Luo. Tgif: A new dataset and benchmark on animated gif description. In *Proceedings of* the IEEE Conference on Computer Vision and Pattern Recognition, pages 4641–4650, 2016.
- [3] Qin Jin, Shizhe Chen, Jia Chen, and Alexander Hauptmann. Knowing yourself: Improving video caption via in-depth recap. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 1906–1911. ACM, 2017.
- [4] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern* recognition, pages 4566–4575, 2015.
- [5] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. Selfcritical sequence training for image captioning. In *CVPR*, volume 1, page 3, 2017.
- [6] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [8] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Computer Vision and Pattern Recognition (CVPR)*, 2017 IEEE Conference on, pages 4724–4733. IEEE, 2017.

¹The training set is the combination of TGIF and trecvid16.

Informedia@TRECVID 2018: Ad-hoc Video Search with Discrete and Continuous Representations

Po-Yao Huang, Junwei Liang, Vaibhav, Xiaojun Chang and Alexander Hauptmann Carnegie Mellon University Pittsburgh, PA 15213

Abstract

We present two different approaches we developed for cross-modal retrieval in the Ad-hoc video search (AVS) task. Our system is a fully-automatic system utilizing no in-domain data nor annotation. We jointly utilize representations in the discrete semantic space learned from multiple mutually exclusive source domains as well as continuous representations in the textual-visual joint-embedding space. We encode textual queries and videos in these spaces and perform search and retrieval. We achieved 12.59 inferred average precision (IAP) on the AVS 2016 validation set and achieved 8.7 IAP and ranked 2nd in the 2018 AVS task.

1 Introduction

In this abstract paper we summarize Informedia's system for TRECVid 2018 Ad-hoc Video Search (AVS). We developed systems using two types of representations for retrieval followed by late fusion to generate the final output. Specifically, we extend our semantic pool [4] by utilizing models pre-trained on multiple image/video classification datasets to generate discrete visual semantic representation for each video and query accordingly. In contrast, we utilize joint-embedding learned from retrieval datasets with parallel text-image or text-video pairs to generate continuous representations.

Specifically, for discrete semantics, we transfer the models trained on other large scale datasets, such as YFCC, ImageNet Shuffle, Google Sports, and Kinetics for detecting both static and dynamic visual concepts. Each video in the target domain (IACC.3) is then represented as a concatenated discrete vector where each dimension is an interpretable class names defined in the source domain. As will be introduced in the following section, we construct a synonym set (synset) for matching text queries to the video representation.

Parallel to visual semantics, learning continuous textual-visual representations have been proven useful to encode and retrieve multi-modal instances [13; 7; 3]. We build up our models with multi-hop intra- and inter-modal attention and learn the joint embedding on multiple retrieval datasets such as Flickr30K, MS-COCO, MS-VTT where image/video-text pairs are available. We then directly encode queries and videos in AVS into the joint-embedding space then perform retrieval by measuring the distance in such space.

The results on the 2016 validation set show different effects of these two approaches from various datasets on different event queries. While the discrete semantics are good at detecting and retrieving videos with distinctive concepts fitting the query exactly, the continuous embeddings capture a wider varieties of videos correlate to the latent concepts which may be interpreted by a query.

Additionally, we design a fully-automatic webly-label learning system that requires no annotation to perform a user search on the test set. Detailed algorithm for curriculum design and model training can be referred to our webly-labeled learning paper [9; 8].



Figure 1: Pipeline for zero-shot video retrieval using discrete semantic representation.

For fusing different models, We leverage the automatic leave-one-out weighting scheme or manual assigning weights in different runs. We achieved 12.59 inferred average precision (IAP) on the AVS 2016 validation set. For AVS 2018 testing set, we report 8.7 mIAP and rank 2nd in the competition.

2 Datasets

We use the following datasets for training discrete semantic representation:

- 1. YFCC [18], 609 concepts.
- 2. ImageNet Shuffle [10], 12073 concepts.
- 3. UCF101 [16], 101 concepts.
- 4. Kinetics [6], 400 concepts.
- 5. Place [22], 365 concepts.
- 6. Google Sports [5], 478 concepts.

The following dataset have been used for training continuous representation:

- 1. MSCOCO, [21], 123,287 image-text pairs
- 2. Flickr30K [14], 31,783 iamge-text pairs.
- 3. MSR-VTT [20], 10,000 video-text pairs.

3 Discrete Semantic Representation

Figure 1 shows the pipeline to retrieve videos given an event query using discrete semantic representations. The bottom half of the pipeline shows the process where each of the video in the collection is represented as a concept vector using the visual concepts. This part is offline. The top half of the pipeline shows the process of generating a semantic query given any event query.

3.1 Semantic Query Generation

Given a query, we first extract all the verb and noun phrases from the query using a parser¹, we call these query terms. The semantic query generation (SQG) module does concept expansion and matching to convert the text query to a semantic query which is then used to create a ranklist of videos by computing the similarity between the semantic query and the extracted representation for the videos. Algorithm 1 outlines the pseudo-code for SQG. We now explain the two key operations used in the algorithm.

¹https://nlp.stanford.edu/software/lex-parser.shtml

3.1.1 Matching

Given the visual concept and the query term, this module gives a score for the term and visual concept pair. We explore three ways to compute this similarity score,

- Exact Match, in this approach we simply count the number of words which are common between a query term and a concept to get the score for the pair.
- Synset similarity score, this is based on the shortest path that connects two words in the wordnet taxonomy.
- Explicit Semantic Analysis (ESA), here we compute the similarity score between the query term and the concept using ESA [1].

3.1.2 Expansion

Given the query terms, this module utilizes Wordnet [12] to expand each term using the synsets. A synset is a set of synonym words which are defined in Wordnet for every word in the lexicon.

Algorithm 1 Semantic Query Generation

Input: (Q, C): query terms $q_i \in Q$ and concepts $c_i \in C$ **Output:** v, semantic query where $v \in \mathcal{R}^{|C|}$

 $Q' = Q + \operatorname{expand}(Q)$ $S = \operatorname{zeros}(|Q'|, |C|)$ for c_i in C do for q_j in Q' do $S[j, i] = \operatorname{match}(q_j, c_i)$ v = S.max(axis = 1)return v

3.2 Learning Semantics

We utilize a deep neural network to train semantic concept detectors [4] using multiple video datasets including YFCC100M, UCF101, Place365, Kinetics, SIN346 and image dataset, ImageNet Shuffle. We extract semantic features for all video shots in the test set.

4 Continuous Representation for Cross-modal Retrieval

Continuous representation in the joint embedding space has demonstrated impressive performance in cross-modal retrieval [7; 13]. Its representative power comes from learning the latent alignments in the continuous joint embedding space with parallel image-text or video text pairs. This type of approach achieves state-of-the-art performance in datasets such as Flick30K and MS-COCO. In contrast to discrete representation in which projection to each dimension can be disentangled and understood semantically, the main cons of continuous is that the learned embedding typically non-interpretable.

For AVS, we leverage the continuous representation learned from external datasets with parallel image-text or video-text pairs to encode the in-domain query and videos into the joint embedding space for retrieval. We use state-of-the art-CNN models as the backbone to extract visual feature and the textual representation are trained from scratch. We leverage two types of attention networks to learn the joint embedding space: (i) Intra-modal attention network and (ii) Inter-modal attention network. If the {*query*, *value*, *keyvalue*} in the attention mechanism belongs to the same modality (text or image) then we categorize it as intra-modal attention otherwise inter-modal. We re-implement and simplified two widely-used state-of-the-art attention networks including the Dual Attention Network [13] (with intra-modal attention) and Stacked Cross-Attention Network [7] (with inter inter-modal attention) to learn the joint embedding. Figure 2 shows the comparison of the two types of attention networks. Now we elaborate more details about the models.



Figure 2: Attention networks for learning continuous representations

4.1 Feature Extraction and Learning

We use ResNet-152 [2] to extract low level visual features and Faster R-CNN [15] to extract regional of interests and its corresponding features for training joint embeddings. We randomly initialize the word embedding matrix and truncate the word tokens appears less than 4 times across datasets. We fix the visual extractor and train the textual encoder on Flickr30K, MSCOCO, MSR-VTT.

4.2 Intra-modal Attention Network

We utilize an intra-modal attention network similar to dual attention network (DAN) [13], which is a multi-hop, intra-modal attention model, to train joint embeddings for image and text. We train the model on the Flickr30K dataset in which each image is given 5 description sentence. We use a linear layer as the visual encoder and a bi-directional gated recurrent unit (Bi-GRU) as the text encoder. For a image-text pair, the encoder in individual modalities first encode the content as $\mathbf{v}^{(0)}, \mathbf{t}^{(0)}$ by averaging the encoded vectors in individual modalities. At each hop, a memory in each modalities then be use to attend the content and generate new memory $\mathbf{v}^{(i)}, \mathbf{t}^{(i)}$. The attention weight is calculated with a softmax function over the inner product of the previous memory vector and the encoded context in each modality. Specifically,

$$\alpha_i^{(t)} = \text{Softmax}(\mathbf{v}^{(t)}\mathbf{v}_i) \tag{1}$$

$$\mathbf{v}^{(t+1)} = \sum_{i=0}^{N} \alpha_i \mathbf{v}_i,\tag{2}$$

for the visual part. Similarly, for the text part, we calculate:

$$\beta_i^{(t)} = \text{Softmax}(\mathbf{t}^{(t)}\mathbf{t}_i) \tag{3}$$

$$\mathbf{t}^{(t+1)} = \sum_{i=0}^{N} \beta_i \mathbf{t}_i \tag{4}$$

These memories then be used for measure the similarity between the image-text pair. We train the model with typical max-margin loss with hard-negative mining for hardest $\hat{\mathbf{v}}, \hat{\mathbf{t}}$. Specifically, we minimize the following loss function:

$$l(\mathbf{v}, \mathbf{t}) = \sum_{\mathbf{v}} \left[\alpha - S(\mathbf{v}, \mathbf{t}) + S(\mathbf{v}, \hat{\mathbf{t}}) \right]_{+} + \sum_{\mathbf{t}} \left[\alpha - S(\mathbf{v}, \mathbf{t}) + S(\hat{\mathbf{v}}, \mathbf{t}) \right]_{+}$$
(5)

Given a video query, we use our intra-modal attention network to extract joint embeddings from each video clip's keyframe and match to the text query to get a ranking score.

4.3 Inter-Attention Network

Attention mechanism across different modalities has recently be shown effective for cross-modal retrieval. For AVS, we utilize the stacked cross-attention network [7] to learn the joint embedding. Similar to intra-modal attention, we first encode text and image/video through textual and visual encoders. Instead of using modal-wise memory, the attention is perform crossing different modalities. We generate textually attend visual vectors \mathbf{t}_v and visual attended text vectors \mathbf{v}_t then aggregate them to maximize the similarity for a given pair via computing the trace of inner products. Namely,

$$\mathbf{s}_{t-v} = \operatorname{Trace}(\mathbf{V}^T \mathbf{V}_t) \tag{6}$$

$$\mathbf{s}_{v-t} = \operatorname{Trace}(\mathbf{V}^T \mathbf{V}_t) \tag{7}$$

Similar to training intra-modal attention, we use max-margin loss and hard-negative mining to improve the learned continuous representation.

5 Webly-label Learning for AVS

In this year's Ad-hoc Video Search task, we design a fully-automatic webly-label learning system that requires no annotation to perform a user search on the test set. The system is the same as last year. Detailed algorithm for curriculum design and model training can be referred to our webly-labeled learning paper [9; 8].

5.1 Implementation and System Description

Our system consists of video collection, feature extraction, curriculum design, model training and query search.

Video Collection Since our system requires no manual annotation for ad-hoc queries, it automatically collects Internet videos based on the textual queries for training query models. Given a user query, our system first refines the queries (currently we only strip out the "find shots of" prefix of the official queries) suitable for the video crawler to search for relevant videos on popular video hosting sites like Youtube using their search engine API. Then the system downloads these videos along with their user-generated textual metadata (including titles, descriptions, comments, etc.) into our Internet Video Collection. The test videos (IACC.3) can also be included in this collection since they too have metadata. However, we didn't use that in our submission due to the quality being too low (very few meaningful metadata in the IACC.3 data).

Feature Extraction We use the Inception Resnet model [17] pre-trained on ImageNet. We first extract DCNN features from the keyframes of the videos then use average-pooling to get video-level features. pool5 layer output (2048 dimension) and the prob layer output (1000) are used and concatenated. Explicit feature mapping [19] (order 3 with chi-square kernel) is used to expand the features into higher dimension to avoid using kernel classifiers for speeding up.

Curriculum Design In curriculum design phrase, our system tries to rank the training videos by their relevance to the query from the Internet Video Collection based on the prior knowledge extracted from their textual metadata. Specifically, we consider each video's metadata as a document and utilize word2vec [11] and BM25 algorithm to retrieve the relevant videos. We use a phrase table extracted from GoogleNews corpus for word tokenization.

Model Training In model training phrase, we utilize webly-labeled learning algorithm [9] to learn one-versus-all query model, where the model is refined iteratively from easy to hard samples. The best model is selected based on empirically setting the selection threshold to p (It means that we will select the model trained with half of the total collection retrieved during the curriculum design phrase). The final model is transformed to primal form to speed up query search.

Query Search Finally, after query models are trained, we apply them to the test video shots that are longer than 3 seconds. Average late fusion is used for the final results.



Figure 3: Performance comparison on AVS 2016 validation set. **Sem-EM** stands for the model using discrete semantics with exact match for SQG. **SYN** represents its synset enhancement. **CAN** stands for cross-modal attention Network, **DAN** is dual (intra-modal) attention network. Res stands for ResNet visual feature.

6 Experiments and Submitted Runs

In this part we summarize the details of the submitted runs and show the quantitative and qualitative results on AVS 2016 validation set where labels are available. We then discuss the characteristics of the two types of representation and their pros and cons. Interactive visualization for the fused final submission outputs for AVS 2016 and AVS 2018 are also available online.

Our system achieves 12.6 mIAP, which outperforms 2017 AVS winner who reported 10.2 mIAP by (23.5%) on the 2016 validation set. For AVS 2018, we ranked 2nd place with 8.7 mIAP with **INF_2**.

INF_1 This run utilizes semantic features and joint embedding with automatic weights for fusion.

INF_2 This run utilizes semantic features and joint embedding with manual weights for fusion.

INF_3 This run is INF_1 plus webly learning ranklist with automatic weights.

INF_4 This run is INF_1 plus webly learning ranklist with manual weights.

INF_5 This run utilizes only the webly learning approach and no additional dataset annotation is used.

6.1 Quantitative Results

Table 1 shows the quantitative results on the AVS 2016 validation set. As can be seen, models using continuous representation outperforms models with discrete representations in general. However, for different events, different models are preferred. We observed that for events with out-of-vocabulary (OOV) text queries, models the continuous representations are more robust in comparison to models with discrete semantics. For example, in event 1512 *Find shots of palm trees* the synset model fails because of OOV. On the contrary, discrete semantics are robust if the semantics contains exactly the concept in the concept as in event 1510 *Find shots of a sewing machine*. The constrain of OOV indicates that the models with different types of representations could be complementary, which can be observed in the model fusion experiments.

Automatic fusion weights are determined by leave-one-out of 11 models including webly-label learning, discrete representation models, and continuous representation models. Examining the fusion results, the two types of model are complementary to each other and they together can achieve a much better results in 2016 validation set.

Qeury	Sem-EM (small)	Sem-EM	Sem-Syn	CAN	CAN-Res	DAN	Fusion
1501	4.77	4.76	4.75	7.66	5.11	4.05	-
1502	0.00	0.01	0.00	34.12	24.00	15.68	-
1503	4.33	19.51	26.38	17.01	9.93	15.92	-
1504	0.19	0.63	1.87	2.78	4.28	2.78	-
1505	0.40	0.37	0.00	0.07	0.05	0.09	-
1506	0.00	1.01	5.66	0.74	0.01	0.07	-
1507	1.37	2.01	45.64	15.44	23.64	45.35	-
1508	1.54	1.65	0.88	5.03	4.14	3.20	-
1509	0.89	0.85	3.28	12.86	6.49	11.45	-
1510	0.00	18.17	8.03	0.01	8.77	1.69	-
1511	0.04	0.05	0.18	1.75	0.69	0.38	-
1512	1.15	2.56	1.23	11.95	1.57	3.19	-
1513	0.06	0.23	3.73	0.91	0.26	4.67	-
1514	8.28	10.45	0.00	4.81	5.96	6.85	-
1515	0.03	0.04	0.05	0.35	0.16	0.10	-
1516	0.00	0.00	0.01	1.58	0.00	0.00	-
1517	0.44	1.07	0.22	9.00	3.71	3.56	-
1518	18,14	17.36	12.28	7.25	8.26	14.69	-
1519	15.04	15.27	0.85	9.27	13.75	17.58	-
1520	36.26	33.64	31.77	5.16	4.34	12.67	-
1521	3.91	5.21	2.58	5.12	1.40	3.39	-
1522	6.00	5.44	3.08	5.49	6.58	1.44	-
1523	2.71	2.96	6.63	0.72	0.88	2.48	-
1524	0.14	0.24	0.30	1.50	4.11	1.49	-
1525	4.43	5.92	6.82	2.69	0.49	4.13	-
1526	0.00	0.04	0.00	16.42	2.00	0.73	-
1527	3.92	6.29	3.90	0.15	0.23	1.40	-
1528	0.00	3.70	1.90	9.27	2.04	3.35	-
1529	0.09	0.53	0.12	0.46	2.44	0.08	-
1530	0.29	3.26	3.33	0.02	0.09	2.39	-
IAP	3.81	5.44	5.85	6.32	4.85	6.16	12.59

Table 1: Detailed comparison of models with continuous and discrete representations on 2016 AVS validation set. **Sem-EM** stands for the model using discrete semantics with exact match for SQG. **SYN** represents its synset enhancement. **CAN** stands for cross-modal attention Network, **DAN** is dual (intra-modal) attention network. Res stands for ResNet visual feature.

6.2 Qualitative Results

In Figure 4 we visualize the searching results of our system in AVS 2016 validation set. In Figure 4a, the discrete representation outperforms continuous representation if the exact concept in the text query is in the semantic pool. The models with continuous representation are better handling OOV and compositionality as in Figure 4b and Figure 4c in which the combination of "woman" and "glasses" are captured successfully.

We also observed some instability among the mIAP metric. For example, in Figure 4d and Figure 4e both CAN and SYN generates reasonable search results but the difference among the scores is unexpected. This limitation implies that it may not be feasible to use annotation from previous AVS as the newly retrieved shot may fall outside of the annotated pool. This phenomena is confirmed by comparing the results of our automatic and manual fusion in which the later achieves better performance in AVS 2018.

To promote related research toward more robust video search, we made two interactive websites for visualizing our AVS 2016² and AVS 2018³ results. A screen shot is shown in Figure 5.

²http://vid-gpu7.inf.cs.cmu.edu:2016

³http://vid-gpu7.inf.cs.cmu.edu:2018

CAN: 0.01



SYN: 8.03 (sewing machine in the semantic pool)



(a) 1510 Find shots of a sewing machine



SYN: 1.23 (palm trees: OOV)



(b) 1512 Find shots of palm trees

CAN: 16.42



(c) 1526 Find shots of a woman wearing glasses

CAN: 11.95

SYN: 45.24



(d) 1507 Find shots of a choir or orchestra and conductor performing on stage CAN: 7.25 ??



SYN: 45.24



(e) 1518 one or more people at train station platform

Figure 4: Visualization of the retrieved shots on AVS 2016 validation set

1507 => a choir or orchestra and conductor performing on stage

Sun 11 November 2018





Figure 5: Visualization Interface

7 Conclusion

We have demonstrated the detailed realization of our hybrid cross-modal retrieval system with discrete and continuous multimodal embedding for AVS 2018. Our system has achieved 12.6 mIAP, which outperforms 2017 by 23.5% on the 2016 validation set. For AVS 2018, we have beed ranked 2nd place as 8.7 mIAP with the **INF_2** submission.

References

- [1] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJcAI*, volume 7, pages 1606–1611, 2007.
- [2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770– 778, 2016.
- [3] P.-Y. Huang, J. Liang, J.-B. Lamare, and A. G. Hauptmann. Multimodal filtering of social media for temporal monitoring and event analysis. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, ICMR '18, pages 450–457, New York, NY, USA, 2018. ACM.
- [4] P.-Y. Huang, Y. Yuan, Z. Lan, L. Jiang, and A. G. Hauptmann. Video representation learning and latent concept mining for large-scale multi-label video classification. arXiv preprint arXiv:1707.01408, 2017.
- [5] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.
- [6] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017.
- [7] K.-H. Lee, X. Chen, G. Hua, H. Hu, and X. He. Stacked cross attention for image-text matching. *arXiv preprint arXiv:1803.08024*, 2018.
- [8] J. Liang, L. Jiang, and A. G. Hauptmann. Webly-supervised learning of multimodal video detectors. 2017.
- [9] J. Liang, L. Jiang, D. Meng, and A. Hauptmann. Learning to detect concepts from webly-labeled video data. 2016.

- [10] P. Mettes, D. C. Koelma, and C. G. Snoek. The imagenet shuffle: Reorganized pre-training for video event detection. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, ICMR '16, pages 175–182, New York, NY, USA, 2016. ACM.
- [11] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing* systems, pages 3111–3119, 2013.
- [12] G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.
- [13] H. Nam, J.-W. Ha, and J. Kim. Dual attention networks for multimodal reasoning and matching. *arXiv preprint arXiv:1611.00471*, 2016.
- [14] B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *Int. J. Comput. Vision*, 123(1):74–93, May 2017.
- [15] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99, 2015.
- [16] K. Soomro, A. R. Zamir, M. Shah, K. Soomro, A. R. Zamir, and M. Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, page 2012.
- [17] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. 2017.
- [18] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li. YFCC100M: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.
- [19] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *IEEE transactions on pattern analysis and machine intelligence*, 34(3):480–492, 2012.
- [20] J. Xu, T. Mei, T. Yao, and Y. Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [21] T. yi Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, and C. L. Zitnick. Microsoft coco: Common objects in context.
- [22] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems* 27, pages 487–495. Curran Associates, Inc., 2014.

Informedia@TRECVID 2018: Activities in Extended Video

Jia Chen Jiang Liu Junwei Liang Ting-Yao Hu Wenhe Liu Wei Ke Wayner Barrios Haroon Idrees Donghyun Yoo Yaser Sheikh Ruslan Salakhutdinov Kris Kitani Dong Huang Alexander Hauptmann

Carnegie Mellon University

Abstract

We present an event detection system which shares many similarities with the standard object detection pipeline. It is composed of four modules: feature extraction, event proposal, event classification and event localization. We study each module separately by evaluating several candidate options given oracle input using intermediate evaluation metric. There is mismatch gap between train and test when we integrate the module into the whole system as in the test stage the module won't receive oracle input. Furthermore, we discover that all the gaps between different modules counts and they are the major bottleneck for a system developed in this way. We apply quick fixes to some of the gaps in our final system.

1 System Overview

Event detection task in TRECVID2018[1] is about spatial-temporal localizing of the event and classifying the event while object detection is about localizing the object and classifying the object. This shows that event detection task share much similarity with object detection. Motivated by the similarity, we propose to build our event detection pipeline upon object detection pipeline. To be specific, our event detection pipeline composes four module in total: feature extraction, event proposal, event classification and event localization.

feature extraction It extracts features that are used in rest components.

event proposal It filters out candidate tubes, which are temporal sequences of bounding boxes, for events. These tubes are called event proposal.

event classification it classifies the event for each tube

event localization it refines the tube's temporal-spatial information

The conceptual diagram of the four modules in our system is shown in Figure 1.

2 Feature Extraction

We finetune the 3DResNet[4] feature following the standard procedure[2] in video classification task using event labels on the groundtruth event tubes. Basically, we randomly crop 64 frames from each groundtruth event tube and assing it with the event label in the training stage. In the validation stage, we predict events every 64 frames on the groundtruth tube and mean pooling the predictions to get one prediction for every groundtruth event tube. We split the groundtruth event rubes into two even parts randomly 4 times, which result in 4-split for cross validation. Given that the number of groundtruth event tube instances is not large, we only tune the last block of 3DResNet. This



Figure 1: System overview

feature	mAP
pretrained 3DResNet from Kinetics	0.157
finetuned 3DResNet	0.219
Table 1: Feature finetuning	

preliminary experiment is conducted on 12 events rather all the 19 events. As shown in Table 1, the finetuned 3DResNet performs much better than the pretrained feature.

3 Event Proposal

As all the 19 events involve either person or vehicle object, we use this prior knowledge to build the event proposal module starting from the object detection step. The output of this step is person and vehicle bounding box for each frame. The immediate natural next step is to associate detected object across frames, which is tracking. The output of this step is person tracklet and vehicle tracklet. Finally, we derive event proposal by designing heuristics on the tracklets. The output of this step is event proposal.

3.1 Object Detection

We utilize faster RCNN with feature pyramid network on ResNet-101 as backbone for object detection, in which RoIAlign is used to extract features for Region-of-Interest. Full model is shown in Figure 2. We apply object detection on every k frame from the videos. Full resolution images are input to the model and we train our model using the full 15 object class annotation from the the VIRAT dataset.



Figure 2: Object Detection Model

Table 2:	Person	tracking	performance	in MOTA

scene id	0000	0002	0400	0401	0500
online	79.6	68.5	49.8	43.4	75.8
offline	92.1	71.9	66.4	55.8	83.7

Table 3: Three event proposal ty	pes
----------------------------------	-----

event proposal type	events
person only	Transport_HeavyCarry, Interacts, Riding, Talk-
	ing, Activity_carrying, Specialized_talking_phone,
	Specialized_texting_phone, Entering, Exiting,
	Closing, Opening
vehicle only	Vehicle_turning_left, Vehicle_turning_right, Vehi-
	cle_u_turn
person-vehicle interaction	Closing, Closing_trunk, Entering, Exiting, Load-
	ing, Open_Trunk, Opening, Unloading

3.2 Tracking

Our tracking module receives and processes the raw video frames and object detection results. It models the motion of object bounding boxes by applying Kalman filter, and adopts kernelized correlation filter (KCF) to keep track on current object when the object detection and motion analysis are noisy.

In our system, we implement two tracking algorithms based on two types of scenarios: online and offline. The algorithm for online scenario receives the video frames and detection results in a step-wise way, while the one for offline scenario has all the information of the video. We investigated the the performance of two tracking algorithms on ActEV training and validation sets. Table 3.2 summarize the experiment results of person tracking in Multi-object Tracking Accuracy (MOTA).

3.3 Heuristic Event proposal

We divide all the 19 events to three types for event proposal: person only, vehicle only, and personvehicle interaction. The union of three types cover all the events but they are not exclusive as some events such as Entering belong to both person only and person-vehicle interaction. Table 3.3 shows the map between event and event proposal type. For person only and vehicle only event proposals, we directly borrow the center trajectories of the corresponding object tracklet as the center trajectories of event proposals. The whole event proposal tube is generated by extract a square bounding box centered the trajectory. The size of the square bounding box is set to the geometric mean of the height and width of the detector object plus a padding value to include more context. The padding is indispensable as it helps to include the interacted small object or the same kind of object in events such as Talking and Interact.

As for person-vehicle interaction event, we use a heuristic to combine person and vehicle tracklets into event proposals. Our heuristic is derived from the following observation: for person-vehicle interact event, the person and vehicle involved in the event is close to each other spatially in a period of time. As illustrated in Figure 3.3, the x axis is the time dimension and the y axis is the spatial dimension. The blue curve is the person trajectory and the red curve is the vehicle trajectory. In the black bounding box, the spatial distance between person and vehicle trajectory is consistently smaller than the threshold θ with in the temporal window [t1, t2]. This constitutes a valid person-vehicle interact event proposal. The heuristic of combining object tracklets for person-vehicle interaction proposal consists just one parameter: the spatial distance threshold θ , which could be estimated from the groundtruth. To generate the tube of the person-vehicle interaction proposal, we use the union bounding box of both objects at each frame and then do the same squaring and padding operation as in person only and vehicle only events.

Some event proposals generated in this way may cover a very long period of time. For example, two men could keep talking with each other for one hour. We denote these proposals as parent proposals,



Figure 3: Illustration of heuristics for event proposals with person-vehicle interact type

	# proposal	recall@ $IoU = 0.1(0.2 * 0.5)$
w/o background modeling	53,406	90.0%
w background modeling	28,783	84.3%

Table 4: Recall of event proposal

which need to be further broken into more child proposals of short period for the classification module. We apply sliding window with size 192 frames and stride 64 on the parent event to generate child proposals. The number of proposals generated and the recall of the groundtruth at IoU = 0.1(0.2 * 0.5) is shown in Table 4. IoU = 0.1(0.2 * 0.5) is the spatial-temporal Intersection over Union with spatial IoU at 0.5 and temporal IoU at 0.2. This is also the value in the matching kernel of official evaluation script. As shown in the first row of Table 4, our event proposal strategy is not only simple but effective, which achieves 90% recall.

We could further improve the precision of event proposals as we observer that the generated event proposals involve many still objects. Such proposals could be filtered with computationally cheap background modeling algorithm before feeding into the advanced classifer. We use the mixtured of Gaussian algorithm[6] implemented in openCV to filter out proposals of still objects. As we see in the second row of Table 4, background modeling helps us get rid of near half of the event proposals and the recall still achieves 84.3%.

4 Event Classification

The input of the event classification module is the child proposal in the test stage. In the training stage, we use the groundtruth event tube instead. In this module, we compare two network architectures: Convolutional RNN and RNN.

The convolutional RNN[5] is an extension of RNN designed to do spatial-temporal sequence prediction. Though we only need to do temporal sequence prediction given child proposal, we use Convolutional RNN to learn an intermediate long-term spatial-temporal feature respresentation before classification. To verify the effectiveness of convolution RNN over RNN, we run experiments on two datasets, VIRAT and SED. In SED, we have 10x more instances than VIRAT. Both classification datasets is construted by mixing the groundtruth event tubes with 5x non-trivial background tubes. Non-trivial means that the background trubes contain object that also appear in the foreground event. The experiments are conducted with I3D[2] on RGB stream pretraine don Kinetics dataset. As shown in Table 5, we see that convolutional RNN outperforms RNN on both datasets. Furthermore, the improvement is more significant with more data, which is not surprising as convolutional RNN contains more parameters than RNN. The performance of both network architecture has already been tuned with their best hyper-parameter.

dataset	model	mAP	GAP
VIRAT	RNN	0.253	0.339
VIRAT	convolutional RNN	0.266	0.356
SED	RNN	0.808	0.792
SED	convolutional RNN	0.861	0.845

Table 5: Comparison of RNN and convolution RNN

method	0.2rfa	0.15rfa	0.1rfa
NMS	0.867	0.890	0.907
greedy merge + NMS	0.834	0.860	0.876

5 Event Localization

After running the classifier on the child proposals, which is generated by sliding window on parent proposals, we need aggregate the predicted result for the final detection output. We experiment with two methods in this module. The first method is non-maximum suppression (NMS), which is directly borrowed from object detection used but applied on the sliding window output. The merit of this method is that it avoids redundant output. However, it has the limitation that all the output is of the fixed temporal length, which is not very reasonable. The second method is to greedily merge the sliding windows into one as long as the prediction score is above the threshold. We apply the second method event-wise. As shown in Table 6, combining greedy merge with NMS achieves the best performance on pmiss at different rfa levels.

6 Mismatch Gap between Modules

We mainly focus on two kinds of gaps:

- 1. the gap between event classification and event proposal
- 2. the gap between feature extraction and event classification

6.1 Gap between event classification and event proposal

The event classification module is trained and validated on the groundtruth event tubes. However, in the test stage, the input of classification module is the generated event proposal. This introduces the gap: the difference between groundtruth event tubes and the generated event proposal tubes. We run the following experiment to quantify this gap by exploiting the fact that the classifier performs perfectly on the groundtruth event tubes in the training videos. We run our whole pipeline on the training set of videos and evaluated on official metric. If the gap matters, we expect to see two results:

- 1. a significant performance gap between the result on the train set and perfect performance, which is 0
- 2. there is not a significant performance between the result on train set and on the validation set as the gap cancels out the difference of classification performance on train and validation set.

As shown in Table 7, both results are observed, which indicates the gap between event classification and event proposal does matter.

To fix this gap, we also generated proposals on the training videos and and add these proposals into the training set in a hard-negative mining way. To be specific, we run our pipeline on the training videos. We add miss detections (MD) and false alarms (FA) to the train set of classifier. We finetune the classifier on the augmented training set. As shown in Table 8, we see that the performance is improved by more than 2 points on most rfa levels after augmenting the training set in this hard negative mining way.

video set	0.2rfa	0.15rfa	0.1rfa
validation	0.867	0.890	0.907
train	0.766	0.797	0.821

Table 7: Gap between event classification and event proposal

 Table 8: Fixing the gap between event classification and event proposal

	0.2rfa	0.15rfa	0.1rfa
gap	0.872	0.884	0.924
fix gap	0.841	0.865	0.883

6.2 Gap between feature extraction and event classification

As we first finetune the feature with a simple one-step classifier and then train the multi-step convolutional RNN with feature fixed, we further study whether this incurs the gap between feature extraction and event classification. As shown in Table 9, comparing the first two rows shows that for the same model, tuning model together with the feature achieves much better performace, which is not surprising. Comparing the middle two rows shows that convolutional RNN is a stronger model than one-step classifier with feature fixed. However, the last two rows shows that tuning the feature with the classifier together on a weaker model performs better than fixing the feature and learning a stronger model alone. As we discover this issue very late in the competition, we are not able to train convolutional RNN together with the feature extractor in time. But this does show that the gap between feature extraction and event classification does matter.

7 Final submission system

After tuning each module separately and fixing some gaps between them, we submit two systems. The first system is trained only on the train set. The second system in trained not only on the train set but also on the validation set. To be specific, we finetune the classifier in the second system from the first system using both train and validation set. We finetune the classifier for 15 epochs and average the weight of models in epoch 5, 10, 15 following the practices of modeling averages[3]. This is the only difference between the two systems. The result is shown in Table 7. The performance on AD metric is similar to the result on validation set. However, the AOD result is very different from our result on validation set. We are still investigating this issue.

8 Conclusion

We designed an event detection pipeline based on the object detection pipeline, which is composed of four modules: event proposal, feature extraction, event classification and event localization. For each module, we compared several candidates and choose the best into the final pipeline. As each module is trained separately i.e. not end2end, we study the gap between modules and find that all gaps matter. We derive some quick fixes to some of the gaps and more work will be done along this direction in the future.

feature	model	end2end	#event	mAP
pretrained 3DResNet	one-step classifier	Ν	12	0.157
3DResNet	one-step classifier	Y	12	0.219
pretrained I3D	one-step classifier	N	19	0.160
pretrained I3D	convolutional RNN	Ν	19	0.233
3DResNet	one-step classifier	Y	19	0.270
3DResNet finetuned by one-step classifier	convolutional RNN	Ν	19	0.261

Table 9: Gap between feature extraction and event classification

Table 10: Final submission result

	AD	AOD
system-1	0.844	0.951
system-2	0.879	0.971

References

- [1] G. Awad, A. Butt, K. Curtis, Y. Lee, J. Fiscus, A. Godil, D. Joy, A. Delgado, A. F. Smeaton, Y. Graham, W. Kraaij, G. Quénot, J. Magalhaes, D. Semedo, and S. Blasi. Trecvid 2018: Benchmarking video activity detection, video captioning and matching, video storytelling linking and video search. In *Proceedings of TRECVID 2018*. NIST, USA, 2018.
- [2] J. Carreira and A. Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. CoRR, abs/1705.07750, 2017.
- [3] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, and A. G. Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.
- [4] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri. A closer look at spatiotemporal convolutions for action recognition. *CoRR*, abs/1711.11248, 2017.
- [5] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810, 2015.
- [6] Z. Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on, volume 2, pages 28–31. IEEE, 2004.

Acknowledgement

The work reported here was supported in part through the financial assistance award 60NANB17D156 from U.S. Department of Commerce, National Institute of Standards and Technology, by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DOI/IBC) contract number D17PC00340 and by DARPA grant FA8750-18-2-0018 funded under the AIDA program.

The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation herein. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DARPA, DOI/IBC, or the U.S. Government.