NII_Hitachi_UIT at TRECVID 2017

Sang Phan [1], Martin Klinkigt [6], Vinh-Tiep Nguyen [4], Tien-Dung Mai [4], Andreu Girbau Xalabarder [5], Ryota Hinami [2], Benjamin Renoust [1], Thanh Duc Ngo [4], Minh-Triet Tran [3], Yuki Watanabe [6], Atsushi Hiroike [6], Duc A. Duong [4], Duy-Dinh Le [4], Yusuke Miyao [1], and Shin'ichi Satoh [1]

[1] National Institute of Informatics, Japan
[2] The University of Tokyo, Japan
[3] University of Science, VNU-HCMC, Vietnam
[4] University of Information Technology, VNU-HCMC, Vietnam
[5] Universitat Politècnica de Catalunya, Spain
[6] Hitachi, Ltd., Japan

1 TRECVID 2017 Instance Search: Searching Specific Persons in Specific Locations

Abstract. This paper presents the proposed system of our team for TRECVID Instance Search task. In this year system, we focus on person recognition step and scene tracking to improve both precision and recall of the system. First, instead of using face from the bottom of initial ranked list which is very weak in classification, we use face samples from the top of the ranked list which is the second highest score. Based on this strategy, the second highest score guarantee that the chosen samples are hard negative. For classification model, we also use SVM algorithm with RBF kernel instead of linear kernel. Last but not least, to improve the recall of the system, we track top returned shots using person re-identification methods. The final results show that, the proposed hard negative samples and scene tracking method help to improve performance of the system.

1.1 Introduction

Fig. 1: A query topic includes location examples (first row images) and person examples (second row images) marked by magenta boundaries. Images in the first row are examples of a pub that a user want to search. These images cover multiple views of a location with many irrelevant or noisy objects such as humans, temporary decorations. These objects may cause low retrieval accuracy due to noisy features. Images in the second row are examples of the person that the user also need to find if he appears at the pub. Programme material copyrighted by BBC.

This year, TRECVID Instance Search task (INS) [1] kept the format of compound queries: retrieving a specific person at a specific location. This type of query has many applications in practice such as: surveillance systems, personal video archive management. Figure 1 gives an example of this type of query. To

deal with this type of query, we focus on improving the accuracy of face recognition and system recall. Firstly, instead of choosing 50 shots from the bottom of the initial ranked list, we propose to use face samples from the top of the rank list for hard negative samples. In case a key frame of a shot contains many faces, we use the second highest face score as a negative sample. This approach will improve the accuracy since increasing the number of hard negative samples. Secondly, we propose to use RBF kernel instead of linear kernel as last year configuration. Lastly, to further improve the recall of the system we propose to person tracking in top 100 shots returned from the baseline system. From an anchor shot, we look back and look forward to find shots that contain the target person. This method is based on an assumption that, at the same time, person cannot move to different location quickly.

1.2 Location Search

Similar to last year system, we retrieve shots containing the query location. Our approach is to fuse rank lists of both holistic and local feature based searching systems. For the local feature based approach, we use Bag-of-Visual-Word (BOW) model for location retrieval. In order to improve the recall of the system, we proposed a filter based approach for scene tracking. We filter the similarity scores of BOW based retrieval result to track location with the assumption that shots of a location could not be changed rapidly. The filtered score is computed by the following formulas:

$$S_{norm} = \frac{s - s_{min}}{s_{max}} \tag{1}$$

$$S_{mag} = S_{norm} * S_{norm} \tag{2}$$

where, s is a score of a shot; s_{min} and s_{max} are the lowest and highest scores in top K returned shots in the location search. S_{mag} is the filtered score that would be used for reranking.

After the filtering step, shots with magnified scores that are greater than a threshold will be used in the next step.

1.3 Face reranking with deep feature and person re-identification

The second main part of the query is person identification. Face recognition is a very popular approach to identify a person. First, DPM cascade detector[2] is applied to point out locations of faces in maximum 5 keyframes per shot. Then, face images are described by a deep feature using VGG-Face descriptor[3]. After this module, each face will be represented by a 4096 dimensional feature vector. Although this feature is designed to best fit with L_2 distance metric, there still has a big gap in performance. This could be explained that, the face feature vector does not have the same weight for all components. For each face, the weights of components are different. Instead of using a linear kernel, this year we use RBF, a non-linear kernel for training step[4]. We propose to use face samples from the top of the rank list for hard negative samples. In case a key

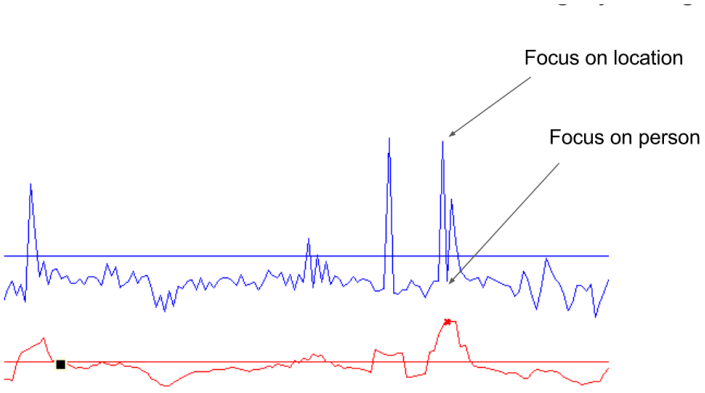

Fig. 2: Filtered score of a location retrieval. At the same location, camera may focus on location or a person in a scene. In case of focusing on a person, the location information will be lost.

frame of a shot contains many faces, we use the second highest face score as a negative sample. For classification, we use SVM algorithm with RBF kernel. Finally, to improve the recall of the system, we propose to use scene tracking with person re-identification.

1.4 Our runs submitted to TRECVID INS 2017

Table 1: Description of submitted runs for TRECVID INS 2017

RUN-ID	Description	MAP
RUN1	Similar to Run 2 using extra shots mentioned about the person based on transcript.	0.355
RUN2	Similar to Run 4 using CNN feature to remove outlier show with incorrect location.	0.377
RUN3	Location search using RANSAC with BOW model. SVM with RBF kernel for person recognition. Person tracking and person re-identification to improve recall.	0.381
RUN4	Location search using RANSAC with BOW model + SVM with RBF kernel for person recognition.	0.374

We submitted 4 automatic runs using all frames of query shots of location and person. Table 1 shows run IDs, descriptions and performances in mean average precision of 4 runs where their priority is sorted from the highest to lowest. The final result shows that, person tracking and re-identification improves the

performance a little bit. In order to significantly increase the accuracy, we should retrain the network with some more augmented data.

2 TRECVID 2017 Ad-hoc Video Search: Combining Concept Features and Dependency Features

Abstract. Ad-hoc Video Search is a challenging problem in TRECVID evaluation [5]. This is due to the high semantic gap between the text query and the video content. A rich source of semantic information is video metadata e.g. title, summary, or textual transcript provided by video owners. However, such amount of semantic information is still far from enough to fully describe video content as it can be observed by human being. Hence, it causes low accuracy in searching videos with complex query. Our approach towards enriching semantic description and presentation is combining concept-based representation and dependency-based representation. Experimental results show that dependency features are complementary to concept features for this task. However, using only dependency features is not reliable because of its sparsity in the text query as well as in the video representation.

2.1 Introduction

With the rapid growth of video data from many sources such as social sites, broadcast TVs, films, one of the most fundamental demand is to search a particular video in huge video databases. In some cases, users did not see any target video shots before. No visual example is provided. The input query could be a text string with ad-hoc description about the content they want to search. Fig 1. gives an example of this query type, "finding shots of a man lying on a tree near a beach".

To deal with AVS query type, when users describe what they are looking for by using verbal description, high-level features (i.e. semantic based features) are usually extracted to match with human language. The result of last year Video Browser Showdown has shown that, leveraging high level feature using deep convolutional neural network (CNN) is one of the-state-of the-art methods [6]. Although the performance of these neural networks are increasing every year, the number of concepts used for training is limited. On the other hand, query topics given by users are unpredictable. We also combine multiple concepts from multiple datasets including ImageNet[7], Visual Genome[8], MIT Places[9] and SUN Attribute[10] to cover most popular topics that users may be interested in.

To further capture the semantic information from the video, we propose to use the dependency matching method. Dependencies are syntactic relations like subject and object that represents a relationship between concepts. Therefore, dependency representation can convey a richer level of semantic information which can not be found from encoding individual concepts. This idea is related to our previous work [11], in which we utilized the dependencies obtained from image captions for video event detection.

2.2 Concept Extraction

In this section, we propose to extract semantic features to match with ad-hoc query given by users. Because the users may pay attention to any aspects of a

video frame, the set of semantic concepts is unknown. Figure 3 shows an example in which users may be interested in varying from single objects e.g. the man, the beach, the coconut tree to their complex relations e.g. the man lying on the tree, the tree next to the beach.

Fig. 3: Users may be interested in single objects e.g. the man, the beach, the coconut tree, or the complex relations between objects e.g. the man lying on the tree, the tree next to the beach.

Since the number of concepts is unlimited and the query of the user is unpredictable, to increase the recall of the system, we propose to extract as much semantic description and presentation of a video at frame-level as possible. Inspired by recent success of deep learning techniques, we also leverage the powerful of deep features in semantic search task. In this system, semantic concepts includes:

- Objects: ones that appear in a large enough region of the video frame with assumption that the higher salient object gives the higher score from the output activation of the pretrained deep convolutional neural network. In this paper, we use VGG-16 network proposed by K. Simonyan and A. Zisserman [12] to extract main objects. Feature maps from the output activation are aggregated together using average pooling approach.
- Scene Attributes: includes indoor/outdoor labels, building, park, kitchen etc.. In our system, the attributes are extracted from the state-of-the-art models trained on MIT scene and SUN attribute dataset [9].
- TRECVID SIN345 concepts [13]. We use the concept detection scores for the IACC.3 dataset that are shared by the ITI-CERTH team [14].
- Image captions. In order to obtain the captions for each video shots, we use popular method including NeuralTalk [15] and DenseCap [16] method.
- Using concept/dependency detector. We also train the concept and dependency detectors on the MSCOCO dataset and use these detectors to detect

concept/dependency on the AVS datasets. This is similar to the approach described in [11].

2.3 Dependency Extraction

Different from our last year systems, which is only based on concept matching or manually select co-occurrence concepts. In this year, we propose to select the co-occurrence concepts in a systematic way based on the syntactic dependencies. The motivation behind using dependency matching is simple. For instance, consider this AVS query: "Find shots of a policeman where a police car is visible". In this query, the dependency "police car" is crucial for searching. If we only use concept-based representation, we might able to search videos that contains both "car" and "police" but might not be "police car". Dependency representation can resolve this ambiguity.

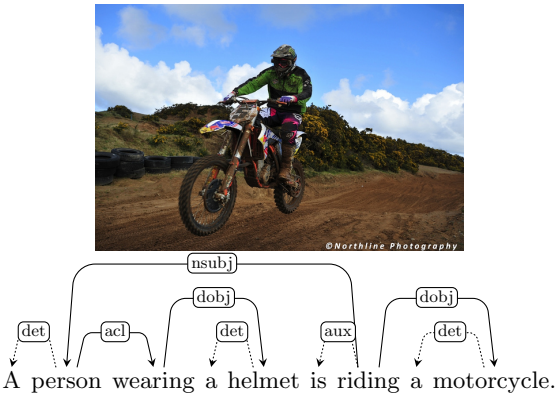

Fig. 4: Example of dependencies extracted from a text description.

Figure 4 shows examples of possible dependencies that can be extracted from the text description. The dependency tree of the caption is obtained by applying Stanford Parser [17]. In practice, we do not always have access to the full sentence description of an image. For examples, for dataset like ImageNet, Places or SUN, we only have the category labels of each image, which can be a word, a phrase, or several phrases. In this work, we directly apply the Stanford Parser [17] on those category labels to extract the dependencies, though for some classes, the dependency is not available.

2.4 Concept/Dependency Matching

After extracting semantic features including the concept-based features and dependency features, the searching task is now equivalent to text based retrieval

task. This stage is to index semantic text returned from the previous stage. A standard *TF-IDF* scheme is used to calculate weight of each word. In the online searching stage, the system computes similarity scores between query text and video semantic features using inverted index structure.

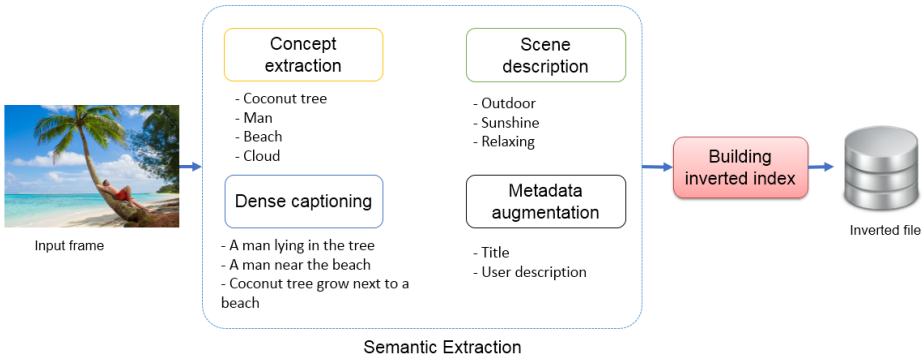

Fig. 5: Proposed system for searching based on semantic description.

Figure 5 illustrates our proposed framework for concept matching. The same system is also used for matching dependencies between the text query and the video. In summary, the pipeline of our system can be described in the following five steps.

- Step 1: Detect concepts and dependencies from text queries
- Step 2: Build the concept and dependency from the concept banks
- Step 3: Detect concepts/dependencies using the pretrained concept models
- Step 4: Calculate the dot product scores, weighted by TF-IDF
- Step 5: Late fusion to combine concepts and dependencies

2.5 Results

Table 2 shows the performance of concept and dependency matching, as well as their combinations. In general, concept-based matching performs better than dependency-based matching. This is reasonable because we observed that the dependency representation can only be obtained in 5 out of 30 queries. In the remaining queries, we could not extract any dependency from the text queries that is also appeared in the dependency vocabulary (obtained from all the concept category labels). Therefore, performance of dependency features is zero in these queries. The performance of combining both concept and dependency is more or less similar to using concept only, except for the case of SIN345 features, where we observe a significant improvement. The performance of using image captioning methods such as NeuralTalk [15] and DenseCap [16] are not good, and we did not incorporate those runs in our submitted runs.

Table 2: Results of using concept and dependency matching

Concept Bank	Concept	Dependency	Concept + Dependency
imagenet1k (1)	0.0429	0.0103	0.0443
imagenetplaces1365 (2)	0.0327	0.0110	0.0324
sin345 (3)	0.0179	0.0054	0.0213
densecap (4)	0.0059	0.0056	0.0067
neuraltalk (5)	0.0023	0.0011	0.0024
mscoco (6)	0.0178	0.0058	0.0095
(1) + (2) + (3)	0.0625	0.0221	0.0680
(1) + (2) + (3) + NII_Hitachi_UIT@AVS2016 [18]	0.0842	0.0689	**0.0857**

Table 3: Summary of AVS2017's Submitted runs

RunID	Description	Test 2016	Test 2017
1	Concept + Dependency	0.0680	**0.081**
2	Concept only	0.0625	0.077
3	NII_Hitachi_UIT@AVS2016 [18]	0.0538	0.026
4	Concept + Dependency + NII_Hitachi_UIT@AVS2016 [18]	**0.0857**	0.058

We submitted 4 automatic runs to this year Ad-hoc Video Search task. Table 3 shows run IDs, descriptions and performances in mean average precision of 4 runs where their priority is sorted from the highest to lowest. Our last year's wining system does not perform well on this year's test set. Our best run is the run that combining both concept and dependency matching. However, this result is quite below the bar when comparing with other participants. This may be the limitation of the concept-based and dependency-based matching method. In the future, we plan to learn a joint visual-semantic for the retrieval task, which can better bridge the semantic gap between the text query and the video content.

3 TRECVID 2017 Surveillance Event Detection

3.1 Abstract

In this paper, we present a retrospective system for the surveillance event detection (SED) task in TRECVID2017. In this system we combine a high-precision head detector trained by using deep learning and track detected head regions with a generic object tracker. Detected persons are classified by fusing scores of still-image classifiers from DCNN (Deep Convolutional Neural Network) and motion-images classifier using C3D (Convolutional 3D) [19]. In the resulting SED system we fine-tuned pre-trained DCNN/C3D models on the Gatwick airport dataset with extra annotations and explored optimum weights for score fusion.

3.2 SED system overview

Fig. 6 shows an overview of our SED system. This is an enhanced version of the system developed in TRECVID2016 [18].

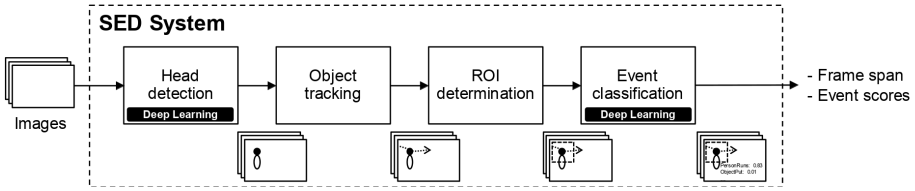

Fig. 6: NII_Hitachi_UIT surveillance event detection system.

Our SED system consists of the following four steps: (1) Head detection, (2) Object tracking, (3) ROI determination and (4) Event classification. For head detection we use the method proposed by Russell et al. by combining CNN and LSTM [20]. We used the publicly available pre-trained models and fine-tuned them on the Gatwick airport videos with extra annotations (11,970 images, 82,583 head region coordinates).

After head detection, the system associate head regions across multiple frames by using a generic object tracker proposed by Joo [21] which results in temporal coordinates of detected people.

In the following ROI determination step, the system extracts Regions Of Interest (ROI) from each frame. By using the head coordinates to calculate the upper and entire body regions by predefined ratios based on the head region size. To avoid effects similar to global camera motion, we use the same ROI of one frame in the whole sequence.

For the final event classification the system calculates scores for the target events with individual action classifiers for each body region trained using Deep Learning. As shown in Fig. 7, the system utilizes multiple classifiers: entire

body still-image (DCNN#1), upper body still-image (DCNN#2), entire body motion-images (C3D#1), and upper body motion-images (C3D#2). We used an ImageNet pre-trained VGG-19 model for DCNN, and Sports-1M pre-trained model for C3D. The system calculates event scores by fusing scores obtained from the multiple classifiers. Table 1 shows the weights for score fusion explored by grid-search using the training data.

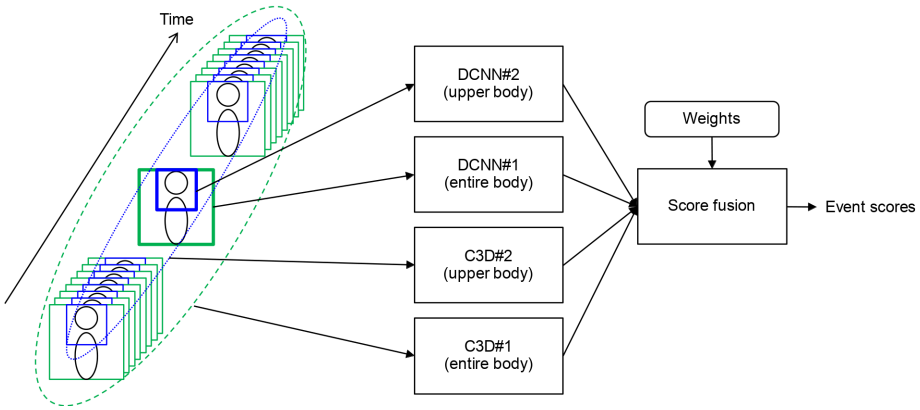

Fig. 7: Late score fusion of multiple CNN and C3D.

3.3 Evaluation results

Event	DCNN#1	DCNN#2	C3D#1	C3D#2
CellToEar	0.00	1.00	0.00	0.00
Embrace	0.10	0.00	0.85	0.05
ObjectPut	0.00	0.00	1.00	0.00
PeopleMeet	0.30	0.00	0.65	0.05
PeopleSplitUp	0.05	0.15	0.60	0.20
PersonRuns	0.45	0.00	0.55	0.00
Pointing	0.05	0.60	0.15	0.20

Table 4: Weights for score fusion for each action

Table 5 shows our evaluation results for EVAL17 provided by NIST, along with the best performance achieved by other participants and our last years results (EVAL16). In EVAL16, we used only one ROI (entire body) and C3D based event classifier. This year we could improve accuracy of system by using multiple types of ROI and still/motion-based classifiers.

	Others best		Ours best		Ours (EVAL16)	
Event	aDCR	mDCR	aDCR	mDCR	aDCR	mDCR
CellToEar	1.0000	1.0005	1.0065	0.9895	1.0200	1.0005
Embrace	0.5996	0.5996	0.9132	0.7846	0.9823	0.9746
ObjectPut	0.9503	0.9483	1.0132	0.9967	1.0132	0.9986
PeopleMeet	0.8942	0.8942	1.0092	1.0005	1.0056	0.9986
PeopleSplitUp	0.9097	0.9097	0.9582	0.9527	1.0076	0.9932
PersonRuns	0.6260	0.6260	0.9217	0.8487	1.0036	0.9896
Pointing	0.9350	0.9308	0.9979	0.9924	1.0105	1.0005

Table 5: Evaluation results of SED task (the smaller the better)

In Fig. 8 we plot the evaluation results of sub-systems with single classifier. We can confirm that the classifiers using upper body ROI or motion-images are effective for Embrace event, and classifiers using motion-images are effective for PersonRuns event. For other events, the difference between classifier is not significant and a system using score fusion achieves generally good performance.

This year we fused the scores of multiple classifiers (late fusion). In the future, we will try to combine features from multiple DCNN/C3D (early fusion).

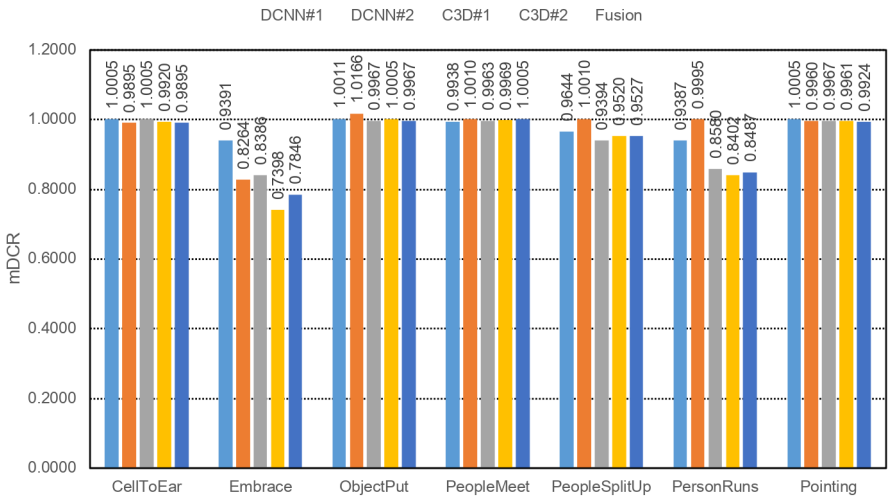

Fig. 8: Comparison of sub-systems and main-system

4 TRECVID 2017 Video-to-Text: Modal Attention Network for Describing Videos

Abstract. We present in this paper our results and analyses on Video-to-Text (V2T) task, which is a pilot task in TRECVID 2017. For the matching and ranking subtask, in our first attempt we wanted to apply our system developed for AVS in the VTT task. However, the performance as below our exceptions. We decided to follow some ideas of the MediaMill team and their VisualWord2Vec. Our final decision was to apply the improved Visual-semantic embedding proposed by Faghri et al [22]. For the description generation subtask, we use a multimodal approach which combining multiple features that are extracted from frames, spatial-temporal volumes and also from audio segments. Moreover, we also employ a modal attention mechanism in the language model that is proposed in [23] to generate better video descriptions.

4.1 Subtask 1: Matching and Ranking

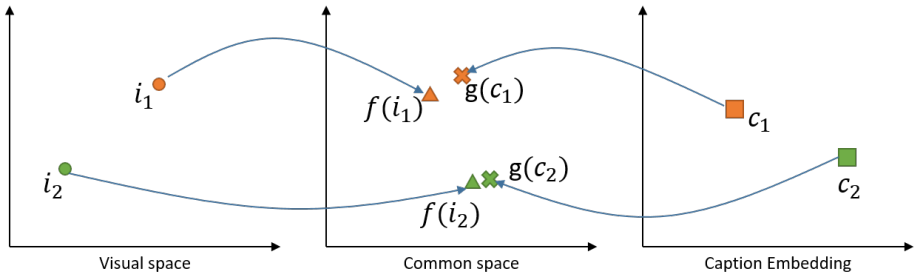

Fig. 9: Visual Semantic Embedding.

Visual-Semantic Embedding The main motivation behind visual-semantic embedding is, to map heterogeneous feature spaces into one common space indicated in Fig. 9. The heterogeneous spaces in this case are the visual features and the word-vector of the caption. Let f be the mapping of visual features to the common space and g the mapping of word-vectors to the common space. The similarity measure s in the common space is defined as the dot product:

$$s(i, c) = f(i) \cdot g(c) \tag{3}$$

where i is the visual feature vector and c is a word-vector of the caption.

In the typical visual-semantic embedding the rank loss over all samples in the mini-batch is optimized, given as:

$$l(i, c) = \sum_{\hat{c}} \max(0, \alpha - s(i, c) + s(i, \hat{c})) + \sum_{\hat{i}} \max(0, \alpha - s(\hat{i}, c) + s(i, c)) \tag{4}$$

where α is a fixed margin, $\hat{c}$ represents a negative caption for the image i and $\hat{i}$ represents a negative image for the given caption c.

Improved Visual-Semantic Embedding According to Faghri et al. optimizing the sum of miss-matched captions, might not be the best approach and came up to replace it by just taking the max of the violating caption:

$$l(i,c) = \max_{\hat{c}} \max(0, \alpha - s(i,c) + s(i,\hat{c})) + \max_{\hat{i}} \max(0, \alpha - s(\hat{i},c) + s(i,c)). \quad (5)$$

The motivation behind this is, that easily miss-matched captions should have a higher contribution to the loss, and therefore optimization, as captions which are already matched correctly.

Proposed Model We utilized two dataset, MSCOCO and MSRVTT and provided pre-trained model. For the combined MSCCOCO + MSRVTT model we applied late fusion with equal weights for both scores. Indexing of captions was adapted accordingly to fit into the TRECVid2017 VTT task. In this indexing we discovered a bug for the MSCOCO model, which was corrected later.

Results In tables 6 and 7 we show the results on the Set 2 and 5. On both sets we ranked third after the team of DL and MediaMill, who achieved 0.383 and 0.229 on set 2 and 0.773 and 0.586 on set 5. For the set 5 we have found a bug in our MSCOCO model in indexing step, leading to a low performance of this model, which also lowered the performance of the combined model with MSRVTT significantly. After fixing this indexing bug, performance of MSCOCO improved significantly by about 0.4. The combined performance of MSCOCO and MSRVTT lead to the overall third rank in this task.

Table 6: Results on Set 2 (2 caption sets)

Mean Inverted Rank (MIR)	set 2.A	set 2.B
R1 - VSE (MSCOCO)	0.141	0.133
R2 - VSE (MSRVTT)	0.128	0.129
R3 - VSE (MSCOCO) + VSE (MSRVTT)	0.185	0.187

4.2 Subtask 2: Description Generation

Problem In this subtask, our system is required to generate a natural language sentence to describe a given video, without mining knowledge of the provided descriptions in the matching and ranking task.

Table 7: Results on Set 5 (5 caption sets)

Mean Inverted Rank (MIR)	set 5.A	set 5.B	set 5.C	set 5.D	set 5.E
R1 - VSE (MSCOCO) (*buggy*)	0.025	0.032	0.057	0.039	0.035
R2 - VSE (MSRVTT) (*no bug*)	0.378	0.389	0.376	0.355	0.380
R3 - VSE (MSCOCO) + VSE (MSRVTT)	0.203	0.219	0.257	0.205	0.241
R1 - VSE (MSCOCO) (*bug fixed*)	0.435	0.493	0.424	0.434	0.422
R3 - VSE (MSCOCO) + VSE (MSRVTT)	0.526	0.563	0.519	0.516	0.508

Methods We use the MANet method proposed in [23] for generating video descriptions. MANet is a new method to combine multiple video features for the captioning task. Different from the existing work, which either combine multimodal features evenly or using a fixed weight combination, MANet uses a dynamic weighting combination that is different for each generated word. This network is illustrated in Fig. 10.

We use the following multimodal video features: VGG [12], ResNet [24], C3D [19], and audio MFCC which represents for three main different streams in video. For each feature, we apply a linear layer to learn an embedded vector that has 512 dimension. We use MANet to learn a weighting combination of those multimodal feaures at each time step. The time-dependent video representation is obtained by concatenating all the embedded features, after multiplying by the attention weighs produced by the MANet.

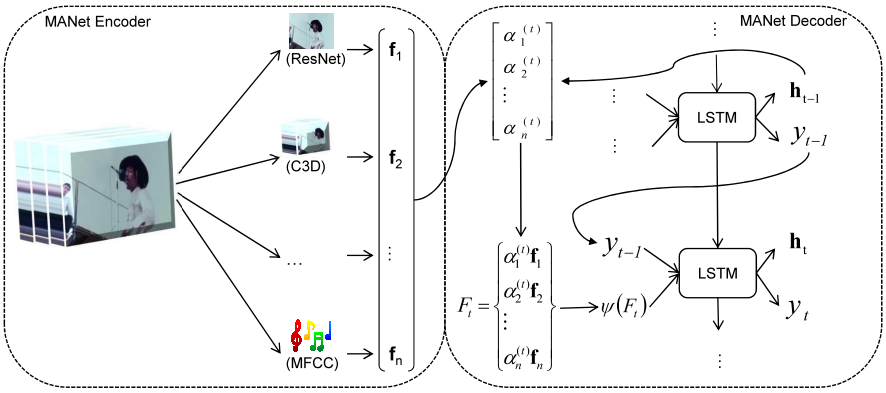

Fig. 10: Overview of MANet framework

Results We train our captioning model on MSR-VTT 2017 dataset [25], which contain around 10,000 videos. We used a subset of 497 videos from this collection

Table 8: Results of our submitted runs to the caption generation subtask

System	CIDEr
RUC_CMU	0.437
MediaMILL	0.328
INF	0.324
VIREO	0.257
NII_Hitachi_UIT run1 (MANet)	0.253
NII_Hitachi_UIT run2	0.214

as the validation set. The training is terminated by the early stopping condition. Finally, we use this model to generate captions on the VTT dataset.

Results of our description task is presented in Table 8. We report the results in terms of the CIDEr metric. Our MANet run (Run 1) performs better than the baseline that did not use MANet (Run 2). This confirms the benefit of using MANet for the description task. However, this result is still inferior to the top performing systems.

References

1. George Awad, Wessel Kraaij, Paul Over, and Shin'ichi Satoh, "Instance search retrospective with focus on trecvid," *International Journal of Multimedia Information Retrieval*, vol. 6, no. 1, pp. 1–29, 2017.
2. M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool, "Face detection without bells and whistles," in *ECCV*, 2014.
3. O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *British Machine Vision Conference*, 2015.
4. Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin, "Liblinear: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, June 2008.
5. George Awad, Asad Butt, Jonathan Fiscus, David Joy, Andrew Delgado, Martial Michel, Alan F. Smeaton, Yvette Graham, Wessel Kraaij, Georges Quénot, Maria Eskevich, Roeland Ordelman, Gareth J. F. Jones, and Benoit Huet, "Trecvid 2017: Evaluating ad-hoc and instance video search, events detection, video captioning and hyperlinking," in *Proceedings of TRECVID 2017*. NIST, USA, 2017.
6. Kai Uwe Barthel, Nico Hezel, and Radek Mackowiak, *Navigating a Graph of Scenes for Exploring Large Video Collections*, pp. 418–423, Springer International Publishing, Cham, 2016.
7. Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.
8. Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalanditis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," 2016.
9. Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva, "Learning deep features for scene recognition using places database," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., pp. 487–495. Curran Associates, Inc., 2014.
10. Genevieve Patterson and James Hays, "Sun attribute database: Discovering, annotating, and recognizing scene attributes," in *Proceeding of the 25th Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
11. Sang Phan, Yusuke Miyao, Duy-Dinh Le, and Shin'ichi Satoh, "Video event detection by exploiting word dependencies from image captions," in *26th International Conference on Computational Linguistics (COLING)*, 2016, pp. 3318–3327.
12. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.
13. George Awad, Cees G. M. Snoek, Alan F. Smeaton, and Georges Quénot, "Trecvid semantic indexing of video: A 6-year retrospective," *ITE Transactions on Media Technology and Applications*, vol. 4, no. 3, pp. 187–208, 2016, Invited paper.
14. Nikiforos Pittaras, Foteini Markatopoulou, Vasileios Mezaris, and Ioannis Patras, "Comparison of fine-tuning and extension strategies for deep convolutional neural networks," in *International Conference on Multimedia Modeling*. Springer, 2017, pp. 102–114.

15. Andrej Karpathy and Li Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3128–3137.
16. Justin Johnson, Andrej Karpathy, and Li Fei-Fei, "Densecap: Fully convolutional localization networks for dense captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4565–4574.
17. Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky, "The Stanford CoreNLP Natural Language Processing Toolkit," in *ACL*, 2014, pp. 55–60.
18. Duy-Dinh Le, Sang Phan, Vinh-Tiep Nguyen, Benjamin Renoust, Tuan A Nguyen, Van-Nam Hoang, Thanh Duc Ngo, Minh-Triet Tran, Yuki Watanabe, Martin Klinkigt, Atsushi Hiroike, Duc A Duong, Yusuke Miyao, and Shin'ichi Satoh, "NII-HITACHI-UIT at TRECVID 2016," in *TRECVID 2016 Workshop*, 2016.
19. Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri, "Learning Spatiotemporal Features with 3D Convolutional Networks," in *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 4489–4497, IEEE.
20. Russell Stewart, Mykhaylo Andriluka, and Andrew Y Ng, "End-to-end people detection in crowded scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2325–2333.
21. João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.
22. Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler, "Vse++: Improved visual-semantic embeddings," *arXiv preprint arXiv:1707.05612*, 2017.
23. Sang Phan, Yusuke Miyao, and Shin'ichi Satoh, "MANet: A modal attention network for describing videos," in *ACM Multimedia*, 2017.
24. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, 2016.
25. Jun Xu, Tao Mei, Ting Yao, and Yong 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.