ITI-CERTH participation in TRECVID 2018

Konstantinos Avgerinakis, Anastasia Moumtzidou, Damianos Galanopoulos, Georgios Orfanidis, Stelios Andreadis, Foteini Markatopoulou, Elissavet Batziou, Konstantinos Ioannidis, Stefanos Vrochidis, Vasileios Mezaris, Ioannis Kompatsiaris

Information Technologies Institute/Centre for Research and Technology Hellas,
6th Km. Charilaou - Thermi Road, 57001 Thermi-Thessaloniki, Greece
{koafgeri, moumtzid, dgalanop, g.orfanidis, andreadisst, markatopoulou, batziou.el, kioannid, stefanos, bmezaris, ikom}@iti.gr

Abstract

This paper provides an overview of the runs submitted to TRECVID 2018 by ITI-CERTH. ITI-CERTH participated in the Ad-hoc Video Search (AVS), Instance Search (INS) and Activities in Extended Video (ActEV) tasks. Our AVS task participation is based on a method that combines the linguistic analysis of the query with concept-based and semantic-embedding representations of video fragments. The INS task is performed by employing VERGE, which is an interactive retrieval application that integrates retrieval functionalities that consider mainly visual information. For the ActEV task, we deploy a novel activity detection algorithm that is based on human detection in video frames, goal descriptors, dense trajectories, Fisher vectors and a discriminative action segmentation scheme.

1 Introduction

This paper describes the recent work of ITI-CERTH[1] in the area of video analysis and retrieval. TRECVID [1] has always been a target initiative for ITI-CERTH given that is is one of the major evaluation activities in the domain of video. In the past, ITI-CERTH participated in the Search task under the research network COST292 (TRECVID 2006, 2007 and 2008) and in the Semantic Indexing (SIN) task (also known as high-level feature extraction task - HLFE) under the MESH (TRECVID 2008) and K-SPACE (TRECVID 2007 and 2008) EU-funded research projects. In 2009 ITI-CERTH participated as a stand-alone organization in the SIN and Search tasks, in 2010 and 2011 in the KIS, INS, SIN and MED tasks, in 2012, 2013, 2014 and 2015 in the INS, SIN, MED and MER tasks ([2], [3], [4], [5]), in 2016 and 2017 in the AVS, MED, INS and SED tasks ([6], [7]) of TRECVID. Based on the acquired experience from previous submissions to TRECVID, our aim is to evaluate our algorithms and systems in order to improve them. This year, ITI-CERTH participated in four tasks: AVS, INS and ActEV. In the following sections we will present in detail the employed algorithms and the evaluation for the runs we performed in the aforementioned tasks.

2 Ad-hoc Video Search

2.1 Objective of the Submission

The goal in the TRECVID 2018 AVS task [8] is the development of techniques for retrieving a ranked list of 1000 test shots for each ad-hoc query that are mostly related to it. Our system for the AVS 2018 task is based on ITI-CERTH AVS 2017 system and [9]. Taking into account the previous year results, we modified our system considering two different directions. Firstly, we examine performing a simpler linguistic analysis of the query, and secondly, we extend the pool of available visual concept

[1] Information Technologies Institute - Centre for Research and Technology Hellas

detectors in order to cover a wider range of concepts from different categories (objects, events, places etc.).

2.2 System Overview

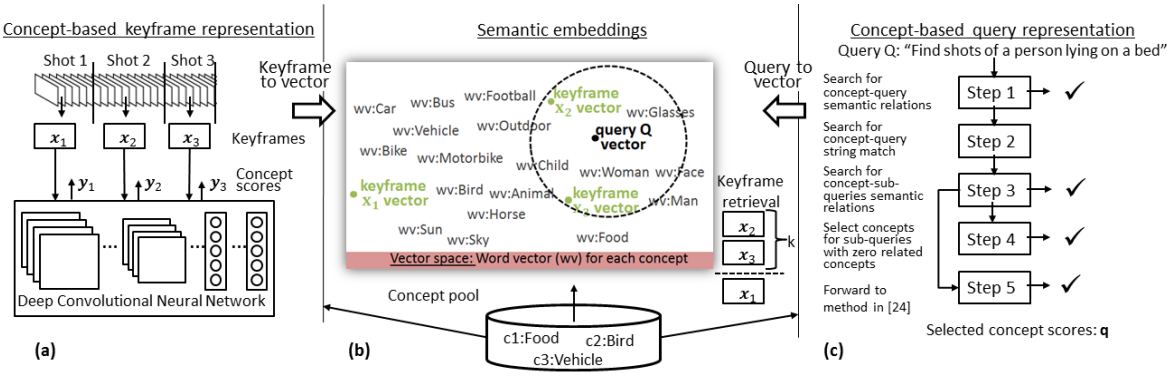

Figure 1: Developed AVS system (modified from [9]).

An overview of the system we developed for the AVS task is presented in Fig. 1. Similarly to our AVS 2017 system, our system consists of three main sub systems. Firstly, the concept-based keyframe representation component (Fig. 1 (a)) annotates every video shot with semantic concepts using deep learning, which results a vector representation that corresponds to the concepts that are depicted in the video shot. In addition, the concept-based query representation component (Fig. 1 (c)) transforms a given text query into an another vector of concepts. Subsequently, both query and video shot concept-based representations are transformed to semantic-embedding representations (Fig. 1 (b)) as described in [9]. Finally, given a test query, after the concept-based keyframe representations have been calculated, our system measures their distance from the concept-based query representation, by calculating their euclidean distance. Similarly, the distance between the semantic embedding keyframe representations and the semantic embedding query representation is calculated and the two distance vectors are combined in terms of arithmetic mean. The 1000 keyframes with the smallest distance are then retrieved. The main components of the above process are further explained below.

2.2.1 Concept-based Keyframe Representation

The concept-based keyframe representation component of our system annotates each video shot with concepts from a predefined concept pool. The output of this component is one vector for each TRECVID AVS test video shot that indicates the probability that each of the concepts in the pool appears in the video shot. Specifically, one keyframe was extracted from each video shot of the TRECVID AVS test set and annotated from a set of predefined visual concept detectors. A key upgrade of our current AVS system is the introduction of a large set of visual concept detectors. In our 2017 system the overall pool of concept detectors consisted of 4 different concept pools. Each video shoot was annotated based on 1000 ImageNet [10], 345 TRECVID SIN [11] concepts, 500 event-related concepts, and 205 place-related concepts. Beside these pools, in our current system, each video shot was also annotated by 239 FCVID concepts, 365 place-related concepts, 1365 "hybrid" concepts, 4 additional sets of ImageNet concepts, and 3 additional concepts derived from TRECVID SIN.

Similarly to our previous year's system, to obtain scores regarding the 1000 ImageNet concepts, we applied five pre-trained ImageNet deep convolutional neural networks (DCNNs) on the AVS test keyframes: i) AlexNet [12], ii) GoogLeNet [13], iii) ResNet [14], iv) VGG Net [15] and v) a DCNN that we trained according to the 22-layer GoogLeNet architecture on the ImageNet "fall" 2011 dataset for 5055 categories (where we only considered in AVS the subset of 1000 concepts out of the 5055 ones). The output of these networks was averaged in terms of arithmetic mean to obtain a single score for each of the 1000 concepts. To obtain the scores regarding the 345 TRECVID SIN concepts we fine-tuned (FT) the ResNet pretrained ImageNet network on the 345 concepts using the TRECVID AVS

development dataset and the extension strategy proposed in [16]. We applied the fine-tuned network on the AVS development dataset and we used as a feature (i.e., a global keyframe representation) the output of the last hidden layer to train one Support Vector Machine (SVM) per concept. Subsequently, we applied this FT network on the AVS test keyframes to extract features, and used them as input to the trained SVM classifiers in order to gather scores for each of the 345 concepts. To obtain scores for the event- and place-related concepts we applied the publicly available DCNNs that have been fine-tuned on the EventNet [17], FVCID [18], Places-205 [19], Places-365 [20] and Hybrid-1365 [21] datasets. To enlarge the number of ImageNet concept detectors we additionally used 4 pre-trained (GoogLeNet) models from [22]: ImageNet4437, ImageNet8201, ImageNet12988, and ImageNet4000. Furthermore, we created 3 derived SIN concepts (*woman*, *man* and *people*) by merging existing SIN concepts. For example, to create the concept *woman* all the female-oriented concepts (i.e. adult female human, female human face closeup, female child etc.) were merged and the score for this concept was calculated by max-pooling the individual scores. In a similar way the remaining two derived concepts were created.

Consequently, a 33142-element concept vector was created for each test keyframe. Each element of this vector corresponds to one concept, from the 33142 available concepts, and indicates the probability that this concept appears in the video shot.

2.2.2 Concept-based Query Representation

The concept-based query representation of our system extracts cues from each query and represents it as a vector of related concepts. Given the above pool of 33142 concepts and the textual description of the query, our method identifies a set of concepts C_Q that most closely relate to the query. Specifically, the selected concepts form a vector where each element of this vector indicates the degree that each concept is related to the query. For this reason, in our current system we examined three different linguistic analysis approaches for cue extraction, which are described bellow:

(i) Firstly, similarly to our previous year's AVS, a sequence of steps was followed. In the first step, we search for one or more high-level concepts that are semantically similar to the entire query, using the Explicit Semantic Analysis (ESA) measure [23]. If such concepts are found (according to a threshold θ) we assume that the entire query is well described by them; the selected concept(s) is (are) added in to set C_Q (which is initially empty), and no further action is taken. If this is not the case, in the second step we examine if any of the concepts in our concept pool appears in the query by string matching, and if so these concepts are added in C_Q. The third step transforms the original query to a set of elementary "subqueries". We define a "subqueries" as a meaningful smaller phrase or term that is included in the original query. To infer "subqueries", conventional natural language processing procedures (NLP), e.g., part-of-speech tagging, stop-word removal etc., are used, together with a task-specific set of NLP rules. For example, if the original query contains a sequence in the form of "Noun - Verb - Noun", this triad is considered to be a subquery. Then in the fourth step, we check for concepts that are semantically similar to any of the "subqueries" by calculating the ESA relatedness between each "subquery" and the concepts in our pool. If there are concepts that exceed the threshold θ (the same threshold as in the first step of this process), then these are added into the set C_Q. Otherwise, if C_Q is still empty, in a final step the original query and all the subqueries are used as input to the zero-example event detection pipeline [24], which jointly considers all this input and attempts to find the concepts that are most closely related to it.

(ii) In a second approach, we modify the third step of the above process in order to decompose the query in to simpler "subqueries". More specifically, as "subqueries" we considered only the Noun Phrases [25] that are included in the initial query. The rest of the procedure described above for C_Q calculation remains intact.

(iii) In a third approach, a much simpler, one-step linguistic analysis process was used, in place of the procedure described above. Simple keywords were extracted by finding the nouns that are included in each query. Then for each extracted keyword, the nearest concept from the concept pool according to their word2vec representations were found and these concepts were added to C_Q.

Then, the query's concept vector was formed by the corresponding scores of the selected concepts of C_Q. For the first two approaches, if a concept has been selected in steps 1, 3, 4 or 5 the corresponding vector's element was assigned with the relatedness score (calculated using the ESA measure), whereas if it has been selected in step 2 it was set equal to 1. Finally, for the third approach, for each concept of C_Q the corresponding vector's element was assigned with the similarity of the concept and the corresponding keyword (calculated as the Euclidean distance of their representation).

2.2.3 Video Shot Retrieval

The third component of our system retrieves for each query the 1000 test shots that are mostly related with it. Specifically, the distance between the query's concept vector (Section 2.2.2) and the keyframe's concept vector (Section 2.2.1) for each of the test AVS keyframes is calculated. Similarly, the distance between the semantic embedding representations of the query and each keyframe is calculated and the two distance vectors are combined in terms of arithmetic mean.

2.3 Description of Runs

Four AVS runs were submitted in order to evaluate the potential of the aforementioned approaches on the TRECVID 2018 AVS dataset [8]. The submitted runs are briefly described below:

- ITI-CERTH 1: The combination (late fusion by arithmetic mean) of runs 2 and 4 (explained below).
- ITI-CERTH 2: Concept-based query representation: cues extraction from the query by considering only nouns; matching each cue with the most semantically related visual concept from the concept pool (approach (iii) of Section 2.2.2). Concept-based keyframe representation as described in Section 2.2.1.
- ITI-CERTH 3: Concept-based query representation as in approach (ii) of Section 2.2.2. Concept-based keyframe representation as described in Section 2.2.1.
- ITI-CERTH 4: Concept-based query representation as in approach (i) of Section 2.2.2. Concept-based keyframe representation as described in Section 2.2.1.

2.4 Ad-hoc Video Search Task Results

Table 1: Mean Extended Inferred Average Precision (MXinfAP) for all submitted runs for the fully-automatic AVS task.

Submitted run:	ITI-CERTH 1	ITI-CERTH 2	ITI-CERTH 3	ITI-CERTH 4
MXinfAP	0.043	**0.047**	0.040	0.034

Table 1 summarizes the evaluation results of the aforementioned runs in terms of the Mean Extended Inferred Average Precision (MXinfAP). Our team submitted only fully-automatic runs. We can see in Table 1 that the simple keyword extraction on the input query (run ITI-CERTH 2) leads to the best results. Overall, the large number of the available concept detectors seems to cover the majority of visual cues that a query could relate to, and for this reason the decomposition of the query in simple keywords (ITI-CERTH 2) is sufficient and outperforms approaches involving more complicate linguistic analysis (ITI-CERTH 3 & ITI-CERTH 4).

3 Instance Search

3.1 Objective of the Submission

According to the TRECVID guidelines, the instance search (INS) [26] task represents the situation, in which the user is searching inside a video collection for video segments of a specific person in a

specific place/scene. The user is provided with two sets of visual examples; the first contains the specific person and the second the specific location. The collection of videos used in the INS task are provided by BBC and they are part of the EastEnders TV series (Programme material BBC).

ITI-CERTH participates in the TRECVID 2018 INS task by submitting a single run that incorporated several algorithms that consider mostly visual information. The system and algorithms developed are integrated in VERGE[1] interactive video search engine.

3.2 System Overview

The INS task, like any process of image search, can be achieved through an interactive tool that will provide users with the capability to efficiently retrieve relevant images. VERGE (Fig. 2) is a Web user interface that serves as an image retrieval application and is able to incorporate different search methods. The integrated modalities for this year involve: (a) Visual Similarity Search, (b) High Level Visual Concept Retrieval, (c) Face Detection and Face Retrieval, (d) Scene Similarity Search, and (e) Multimodal Fusion. Regarding the implementation, the system has been built with popular Web technologies, i.e. HTML5, CSS, PHP, JavaScript, jQuery, and open-source libraries, such as Bootstrap and Kendo UI.

A screenshot of the VERGE application is illustrated in Fig. 2, along with an indication of its components. As it can be easily seen, the interface is separated in two main parts, namely a toolbar on the top and a results panel that covers most of the screen.

The toolbar contains a multitude of useful features. Starting from the left, a burger icon toggles a sliding menu where various search options appear. In detail, users can initiate the retrieval procedure with the complete set of video shots, with landscape or general concepts, with predefined scene clusters and topics. Next to the icon, there is a text field that investigates whether the given input (keywords) exists in the metadata of the videos. Furthermore, there is a slider to modify the size of the images and a timer that counts down the minutes delimited for the INS task submission.

Moving to the basic component of VERGE, videos results are displayed in a grid view through a shot-base representation and are sorted by scores of similarity in a descending order. When users hover on an image, additional search capabilities are offered to them. Visual, face, scene, and fusion similarities can be performed by clicking the respective icon. Moreover, there is a check button to select shots and submit them to the contest.

Fig. 2 can also serve as a demonstration of the core functionality of VERGE. Inspired by the queries of the INS task, Face Retrieval and Scene Similarity Search are combined in the Multimodal Fusion and its outcome is visible in the screenshot: frames that contain both a specific female character and a convenience store.

3.2.1 Visual Similarity Search Module

The visual similarity search module performs content-based retrieval using deep convolutional neural networks (DCNNs). The approach followed was the same with the one in TRECVID 2017 INS task [7]. Thus, we trained GoogleNet [13] on 5055 ImageNet concepts and the, we used the output of the last pooling layer, with dimension 1024, as a global keyframe representation. In order to achieve fast retrieval of similar images, we constructed an IVFADC index for database vectors and then computed K-Nearest Neighbours from the query file. Finally, search is realized by combining an inverted file system with the Asymmetric Distance Computation [27].

3.2.2 High Level Visual Concept Retrieval

This module facilitates search by indexing the video shots based on high level visual concept information, such as water, aircraft, landscape and crowd. The concepts that are incorporated into the system are the 346 concepts studied in the TRECVID-2015 INS task using the techniques and the algorithms described in detail in [5] Section 2 (Semantic Indexing).

Apart from the 346 TRECVID concepts, a set of 356 scene categories using the VGG16 CNN network was used for scene/ place recognition [20].

[1] http://mklab-services.iti.gr/verge/trec2018/

Figure 2: The VERGE Web User Interface

3.2.3 Face Detection and Face Retrieval Module

This module should identify human faces in images and it should also capture the face features from the faces that have already been recognized. For the face detection part the algorithm that was applied was [28], while the algorithm used for face retrieval module was the VGG-Face CNN descriptors which were computed using the VGG-Very-Deep-16 CNN architecture described in [29]. As feature vector, we considered the last FC layer with size 2622. Eventually, the face features were used for constructing an IVFADC index similar to the one created in Section 3.2.1 that allows fast face retrieval.

3.2.4 Scene Similarity Search Module

In this module we use the feature vector from the fully connected layer of the VGG-16 Deep Convolutional neural Network [30] trained on Places 365 dataset [20]. We also use the output of the softmax layer of the same architecture. The size of the feature vectors are 4096 and 365 respectively. Eventually, the scene features were used for constructing an IVFADC index similar to the one created in Section 3.2.1 that allows fast scene retrieval.

3.2.5 Multimodal Fusion Module

Given that the aim of INS task is to retrieve a specific person in a specific place, this module fuses the DCNN-based face descriptors and the DCNN-based scene descriptors in a late fusion approach. The method used for late fusion is the same with the one in TRECVID 2017 INS task [7]. Briefly, this method fuses the two descriptors (or modalities) using a non-linear graph-based fusion approach [31]. For a query shot, an initial filtering stage is applied that keeps only the $top-i$ relevant images according to the dominant modality and then computes an $i \times i$ similarity matrix and an $i \times 1$ similarity vector per modality. The similarity matrices and vectors are fused in a non-linear and graph-based way, providing a fused relevance score vector s_q for the retrieved shots. Eventually, two ranked lists with retrieved shots were created after considering both modalities as dominant, which are merged into a single list using combMax late fusion.

3.3 Instance Search Task Results

We submitted a single run (I_A_B_ITI_CERTH_1) to the interactive INS task, that utilized the aforementioned algorithms. According to the TRECVID guidelines, the number of topics were 21 and the time duration for the run was set to five minutes. Table 2 contains the mean average precision for the runs submitted the last four years in TRECVID INS task. We can see that our results are lower than the previous 2 years and fall down to 2015 levels, however these results are neither characteristic of our algorithm's performance nor comparable to previous years as we didn't have the time to perform the whole set of experiments in this task due to lack of time. Nonetheless, we understand that it is essential that we change our fusion strategy amongst the two modalities of our approach, namely face and scene recognition, so that we can reach other teams retrieval levels in the future.

Table 2: MAP of INS task.

Run IDs	Mean Average Precision
I_A_B_ITI_CERTH_1 (2018)	0.064
I_A_ITI_CERTH_1 (2017)	0.135
I_A_ITI_CERTH_1 (2016)	0.114
I_A_ITI_CERTH_1 (2015)	0.064

4 Activities in Extended Video

For the ActEV (Activities in Extended Video) task, we deploy a novel activity detection algorithm that is based on human/vehicle detection in video frames, HoG-HoF (Histogram of Gradient-Histogram of optical Flow) [32] features, GMM model (Gaussian mixture models), Fisher vectors and an activity SVM (Support Vector Machines) classifier.

4.1 Objective of the Submission

The ActEV evaluation addresses the problem of detection of activities in multi-camera streaming videos. ActEV is an extension of the annual TRECVID Surveillance Event Detection (SED) evaluation. The data used come for this task are an unreleased portion of the VIRAT dataset provided in MPEG-4 format. The dataset is split in three parts: training, validation and test set. The number of videos, frames and seconds of each dataset are shown in Table 3. From the test set a subset was used for the constrained video detection task (Reference segmentation task). The number of targeted activities to be detected was 12 and are shown in Table 4. The actual tasks in ActEV was 2: Activity Detection (AD) and Activity-Object Detection (AOD), while the object of interest was persons and vehicles. The first task involves the correct detection and recognition of each acitivity, meaning the desired outcome of the module would be correct temporal boundaries of a targeted activity as well as correct recognition of the type of activity performed. The second task addition to the previous goals involved also the correct spatial detection of the person/objects involved in the activity, meaning the detection of actual bounding boxes (bboxes) of each person/object involved in the activity in every frame.

4.2 System Overview

The developed system was primarily focusing on the AOD task. The corresponding AD task used the same approach without providing to the final results the bboxes of the objects. We rely on the robustness of object detector to focus on objects of interest. We focus on all 12 target activities with a single module although the objects involved in them was varying. Some activities involved the use of just one kind of objects like for example Vehicle turning left which involved vehicle while other activities involved both types of objects like Open truck which obviously involved both a person and a vehicle.

Type of dataset	Number of videos	Number of frames	Duration (secs)
Train	64	267139	8904.63
Validate	54	201953	6731.76
Test (all)	96	290207	9673.56
Test (Reference segmentation)	48	147037	4901.23

Table 3: TrecVID ActEV dataset

Target activities			
Closing	Closing trunk	Entering	Exiting
Loading	Open Trunk	Opening	Transport HeavyCarry
Unloading	Vehicle turning left	Vehicle turning right	Vehicle u-turn

Table 4: Target activities in TrecVID ActEV

4.3 Activities in Extended Video System

4.3.1 Training of the activity recognition system

For the training of our system we used the provided videos and the frames during which each activity took place as well as the corresponding bboxes. All other frames were ignored in this procedure. More specifically, the steps that were followed for the training part were:

1. A bbox preprocessing step which involved the merging of multiple bboxes participating in a single action. In our case this could include only a person with a car. The merged bbox was an enlarged bbox which included inside both objects. If a single object participated in this activity its bbox was used instead.
2. Lower level HoG-HoF features, were extracted from the activity bbox (enlarged or from a single object) after rescaling the area included in bbox to predined dimensions.
3. The features extracted from each bbox for 15 consecutive frames were concatenated to form an activity feature.
4. A Gaussian Mixture Model (GMM) was created using every activity feature in order to formulate a new data representation.
5. A Fisher vector was created for each activity by passing all activity features belonging to the same activity to the GMM.
6. An activity classifier was created by training 12 1-vs-All linear SVMs using the Fisher vectors as samples.

4.3.2 Testing of the activity recognition system

For the inference part our system was slightly modified and differentiated from the training part mainly because of the absence of bboxes and activity boundaries. Depending on the existing of activity boundaries the task were split into two separate tasks:

1. Activity-Object Detection in Extended Videos (phase AOD-ActEV).
2. Activity-Object Detection using Reference Segmentation (Phase-AOD-RefSeg).

The main difference being that in the former task no activity boundary information was given. This meant that the videos ought to be parsed and the number, duration and initiation time ought to be deducted by the video itself. This posed an extra difficulty in the object-activity detection task. The later phase provided the system with the activity boundaries in which each activity took place. By this way, the extra information of the number of activities to be detected was also provided indirectly.

4.3.3 Testing for the Phase AOD-ActEV

Since in this phase there is no known information about the activities and objects to be detected besides the type of objects and activity that should be detected the process had to parse the videos in inquiring way so as to being able to detect both useful activities and objects but also ignore trivial objects. The object detector was used to fetch the necessary bboxes but the later were filtered to get rid of stationary objects which did not perform any action. We define an arbitrary frame span of 15 frames to compare the bboxes. For all matched bboxes, both in object type and bbox coordinates a further comparison was performed to identify the bboxes which referred to stationary object. Those were not taken into consideration any longer. The other object were used for creating the activity features.

The next steps were followed in this phase:

1. An object detector based on Faster R-CNN [33] which was trained on 1) a subset of UAV123 dataset [34] manually annotated which involved persons, group of persons and cars, 2) a subset of the actual trecVid ActEV training set also manually annotated for object detection was applied on the first frame of the video.
2. A multi-object tracker which tracked the resulting by the object detector bounding boxes for 15 frames.
3. Low-level feature extraction for the bboxes that continued to be detected by object detector, to create dense trajectories using HoG-HoF descriptors.
4. Concatenation of the low-level features to create the activity features. Activity features were assigned to the same activity as long as the detected bboxes every 15 frames were overlapping enough.
5. Creation of Fisher vector for each candidate activity. Activity was inferred by using the SVM classifier.

4.3.4 Testing for the Phase AOD-RefSeg

In this phase the provision of activity boundaries posed some changes in the system architecture:

1. The search for activities was limited to the frames provided in the reference segmentation.
2. Only a single activity was detected in the final result for each activity frame span.

For the reference segmentation AOD/AD task the steps were:

1. The same object detector was applied on the first frame of each activity producing bboxes for all object being detected.
2. A multi-object tracker which tracked the resulting by the object detector bounding boxes for 15 frames.
3. A second detection was performed after 15 frames in order to increase the accuracy of detected bboxes. The bboxes which matched the previous bboxes were used to extract low-level features for 15 frames. Those features were concatenated to create an activity feature.
4. For the following frames to the end of activity an activity feature was created every 15 frames.
5. Creation of Fisher vector for each candidate activity. Activity was inferred by using the SVM classifier and assigning just a single activity to the activity span for the highest score of SVM.

In this phase also a similar threshold was set for the stationary objects, so as not be taken into consideration.

min max	$mean\ P_{miss}@Rate_{FA=.15}$	$mean\ P_{miss}@Rate_{FA=1}$
min	0.6181246	0.4405567
-	$mean\ NMIDE@Rate_{FA=.15}$	$mean\ N-MIDE@Rate_{FA=1}$
min	0.07795288	0.111617
-	$mean\ P_{miss}@Rate_{FA=.15}$	$mean\ P_{miss}@Rate_{FA=1}$
max	0.9994005	0.99807
-	$mean\ NMIDE@Rate_{FA=.15}$	$mean\ N-MIDE@Rate_{FA=1}$
max	0.5794604	0.6667269

Table 5: Activity detection results

min max	$mean\ P_{miss}@Rate_{FA=1AD}$	$mean\ P_{miss}@Rate_{FA=1AOD}$
min	0.4536572	0.5576526
max	0.99807	0.9994005

Table 6: Activity vs Activity Object detection results

4.4 Activities in Extended Video (ActEV) Results

The results of Activity Detection task are shown at Table 5 where the minimum and maximum performance of the primary system is presented. On the other hand, at Table 6 a comparison of the results of Activity Object Detection task vs the Activity Detection task are shown while finally at Figure 3 the Activity Object Detection results are graphically shown for every activity.

The results are evaluated using the following metrics:

- $P_{miss}(\tau)$: the probability of missed detections at the activity presence confidence score threshold τ.
- $Rate_{FA}(\tau)$: the rate of false alarms at the presence confidence score threshold τ.
- $N-MIDE$: Normalized Multiple Instance Detection Error which counts the accuracy of temporal localization of detection instances is.

The proposed method combines deep neural network module with more traditional approaches like HOG-HOF and SVM classification. The success of the method was based heavily on the high and robust performance of object detection module which in our case was not secured. Certain aspects that seem promising are the almost perfect linear discriminality of training activity fisher vector, which achieved accuracy above 99%. This means that there is great potential for accurate activity classification if object detection was more robust. Due to time limitation the model was not trained for the time that was necessary and the generalization of the model is disputed. At this aspect the presented results are not a good representative of proposed method and its full potential.

5 Conclusions

In this paper we reported the ITI-CERTH framework for the TRECVID 2018 evaluation [8]. ITI-CERTH participated in the AVS, INS and ActEV tasks in order to evaluate new techniques and algorithms. Regarding the AVS task, our approach with a simpler keyword extraction procedure, in combination with the large pool of concepts, outperforms approaches involving more complicate linguistic analysis. At Activities in Extended Video (ActEV) task a method combining deep neural network object detection module with traditional HOG-HOF feature extraction and SVM fisher vectors activity classification. Though the results are not the expected ones some aspects of the process seem promising and we intend to intensify our effort for finer system tuning and proper model training in the future. At INS task, in the future, not only we plan to improve our face and scene recognition algorithms, so as to get improved results for retrieving BBC movies database at feature representation

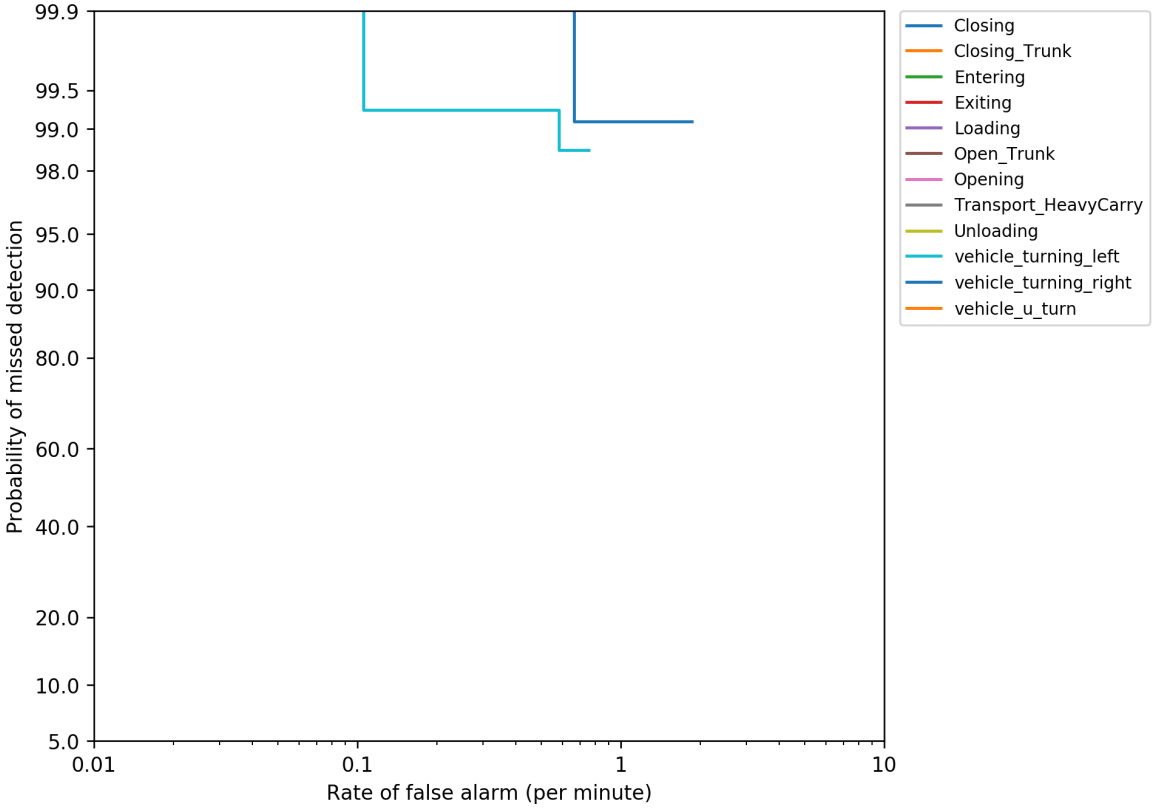

Figure 3: Activity Object Detection results

level, but also we plan to build a more advanced fusion approach for leveraging the outcome of the two above and show more True Positives (TP) to the end-user of the Verge system.

6 Acknowledgements

This work was partially supported by the European Commission under contracts H2020-693092 MOVING, H2020-779962 V4Design, H2020-700475 beAWARE and H2020-022330 ROBORDER.

References

[1] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *MIR '06: Proc. of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

[2] A. Moumtzidou, N. Gkalelis, and P. Sidiropoulos et al. ITI-CERTH participation to TRECVID 2012. In *TRECVID 2012 Workshop*, Gaithersburg, MD, USA, 2012.

[3] F. Markatopoulou, A. Moumtzidou, and C. Tzelepis et al. ITI-CERTH participation to TRECVID 2013. In *TRECVID 2013 Workshop*, Gaithersburg, MD, USA, 2013.

[4] N. Gkalelis, F. Markatopoulou, and A. Moumtzidou et al. ITI-CERTH participation to TRECVID 2014. In *TRECVID 2014 Workshop*, Gaithersburg, MD, USA, 2014.

[5] F. Markatopoulou, A. Ioannidou, and C. Tzelepis et al. ITI-CERTH participation to TRECVID 2015. In *TRECVID 2015 Workshop*, Gaithersburg, MD, USA, 2015.

[6] F. Markatopoulou, A. Moumtzidou, and D. Galanopoulos et al. ITI-CERTH participation in TRECVID 2016. In *TRECVID 2016 Workshop*, Gaithersburg, MD, USA, 2016.

[7] F. Markatopoulou, A. Moumtzidou, D. Galanopoulos, and K. Avgerinakis et al. ITI-CERTH participation in TRECVID 2017. In *TRECVID 2017 Workshop*. NIST, USA, 2017.

[8] G. Awad, A. Butt, K. Curtis, J. Fiscus, et al. Trecvid 2018: Benchmarking video activity detection, video captioning and matching, video storytelling linking and video search. In *Proceedings of TRECVID 2018*. NIST, USA, 2018.

[9] F. Markatopoulou, D. Galanopoulos, Vasileios M., and I. Patras. Query and keyframe representations for ad-hoc video search. In *Proc. of the 2017 ACM on Int. Conf. on Multimedia Retrieval.*, ICMR '17, pages 407–411. ACM, 2017.

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[11] A. F. Smeaton, P. Over, and W. Kraaij. High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements. In Ajay Divakaran, editor, *Multimedia Content Analysis, Theory and Applications*, pages 151–174. Springer Verlag, Berlin, 2009.

[12] A. Krizhevsky, I. Sutskever, and G. E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, Vi. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR 2015*, 2015.

[14] 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.

[15] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[16] N. Pittaras, F. Markatopoulou, V. Mezaris, and I. Patras. Comparison of fine-tuning and extension strategies for deep convolutional neural networks. In *International Conference on Multimedia Modeling*, pages 102–114. Springer, 2017.

[17] G. Ye, Y. Li, H. Xu, D. Liu, and S. Chang. Eventnet: A large scale structured concept library for complex event detection in video. In *ACM MM*, 2015.

[18] Y.-G. Jiang, Z. Wu, J. Wang, X. Xue, and S.-F. Chang. Exploiting feature and class relationships in video categorization with regularized deep neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(2):352–364, 2018.

[19] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *NIPS*, pages 487–495, 2014.

[20] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[21] B. Zhou, A. Khosla, A. Lapedriza, A. Torralba, and A. Oliva. Places: An image database for deep scene understanding. *CoRR*, abs/1610.02055, 2016.

[22] P. Mettes, D. C. Koelma, and C. GM Snoek. The imagenet shuffle: Reorganized pre-training for video event detection. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, pages 175–182. ACM, 2016.

[23] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*, volume 7, pages 1606–1611, 2007.

[24] D. Galanopoulos, F. Markatopoulou, V. Mezaris, and I. Patras. Concept language models and event-based concept number selection for zero-example event detection. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, pages 397–401. ACM, 2017.

[25] R. R. Shah, Y. Yu, A. D. Shaikh, and R. Zimmermann. Trace: Linguistic-based approach for automatic lecture video segmentation leveraging wikipedia texts. In *Multimedia (ISM), 2015 IEEE International Symposium on*, pages 217–220. IEEE, 2015.

[26] George Awad, Wessel Kraaij, Paul Over, and Shinichi Satoh. Instance search retrospective with focus on trecvid. *International Journal of Multimedia Information Retrieval*, 6(1):1–29, 2017.

[27] H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(1):117–128, January 2011.

[28] Peiyun Hu and Deva Ramanan. Finding tiny faces. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1522–1530. IEEE, 2017.

[29] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *Proc. of the British Machine Vision Conference (BMVC)*, 2015.

[30] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[31] I. Gialampoukidis, A. Moumtzidou, D. Liparas, S. Vrochidis, and I. Kompatsiaris. A hybrid graph-based and non-linear late fusion approach for multimedia retrieval. In *Content-Based Multimedia Indexing (CBMI), 2016 14th International Workshop on*, pages 1–6. IEEE, 2016.

[32] Rizwan Chaudhry, Avinash Ravichandran, Gregory Hager, and René Vidal. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1932–1939. IEEE, 2009.

[33] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[34] Matthias Mueller, Neil Smith, and Bernard Ghanem. A benchmark and simulator for uav tracking. In *European conference on computer vision*, pages 445–461. Springer, 2016.