Video Retrieval of
Specific Persons in Specific Locations

Andrea Calafell Orós, Universitat Autònoma de Barcelona
Supervised by: Eva Mohedano, Kevin McGuinness and Noel E. O’Connor (Dublin City University)
Xavier Giró-i-Nieto (Universitat Politècnica de Catalunya)

Abstract
This thesis explores good practices for improving the detection of specific people in specific places, as defined in the TRECVID 2016 benchmark for instance search in large scale video datasets. An approach combining convolutional and recurrent neural networks has been explored to perform face detection, but other more conventional methods have provided better results by exploiting a deformable parts-based model. A CNN is also used to obtain the face descriptors and, with the purpose of helping in the face recognition, the temporal redundancy of video sequences is exploited to expand the visual queries. The retrieval of specific places is solved with a state of the art engine based on convolutional features and a bag of words model. Furthermore, in order to be able to evaluate the different configurations in the non-labelled dataset provided by TRECVID, a user interface was developed to annotate the images and be able to compute a precision of the search engine. Finally, different fusion and normalization strategies have been explored with the aim of combining the scores obtained from the person recognition with the ones obtained in the place recognition.

Index Terms
Instance search, face detection, face recognition, place recognition, query expansion, annotation tool, fusion

I. INTRODUCTION
Scenarios involving large video collections, such as surveillance, law enforcement or even personal video organization have been present in our life for a long time. In the field of computer vision, many approaches have been developed in order to automatically index this type of content, for example by developing detection and recognition of people or objects at a local spatial scale, or the recognition of locations at a more global scale. The size of video databases has grown exponentially with the technological advances in the capture, transmission and storage of this type of media. This big data problem requires efficient and scalable solutions also for the computer vision indexing tools.

In this thesis, a novel approach to retrieve specific people in specific places is developed. This problem is defined as a challenge in TRECVID Instance search 2016 [23], [3], a scientific benchmark with dozens of participants from around the world. This benchmark is organized by the National Institute of Standards and Technology (NIST) in USA, and since 2001 has become the reference scientific forum for video retrieval. Both the persons and locations to be retrieved are defined by TRECVID with a small collection of labelled videos that contain the query persons or the query locations, but never the specific query persons and locations in the same video shot. As an example, Figure 1 depicts a query topic composed by ”Person: Jim, Location: Pub”. Notice that both persons and locations are defined by visual examples and that, in the case of the persons, the query image is complemented with a binary mask.

A first goal of this thesis was obtaining a baseline approach in order to participate in TRECVID Instance Search 2016 (deadline on July 1, 2016). The following objectives were fixed for this first baseline: (a) testing some face detectors and finding the most suitable face detector for the TRECVID data, (b) combining the face detector with the most suitable approach for face recognition, and (c) increasing the query images by exploiting the temporal redundancy of the neighbouring frames in the query video shots. Once the TRECVID challenge was submitted, a second stage of the work aimed at improving the baseline results. However, one of the challenges of TRECVID is that the provided dataset is non-labelled, which means that we are not able to obtain a measure to evaluate our system. For this reasons, this second part was divided in these two tasks: (d) develop a user interface to annotate a portion of the TRECVID dataset, and (e) use these annotations to quantitatively assess different types of normalization and fusion strategies between the person and the location scores.

In order to present all the methods, as well as the results and conclusions, the remaining of the report have been organized as follows: Section II overviews the state of the art in instance search, Section III presents the adopted methodology, Section IV includes the experiments, and finally Section V draws a summary of the work and draws its final conclusions.
II. STATE OF THE ART

This section presents a review of the state of the art on retrieval and instance search. A classic content-based image or video retrieval system [24] follows some basic and well-known steps. Firstly a descriptor for the query image, and for the whole target dataset. Secondly, a matching is assessed by comparing the dataset descriptors with the query descriptor. To do that an appropriate similarity metric has to be chosen based on the descriptors and requirements. Finally, the ranking is obtained, where the relevant images are expected to occupy the top positions, and the non-relevant in the bottom. This basic pipeline is presented in Figure 2.

A typical way to do that is using hand-crafted features in a bag of visual words approach (BoW), by detecting the keypoints in the image, obtaining its local representation and clustering them into a feature space, where the words are counted obtaining a new vector representation of visual words. An example can be found in [31], where the model has been enhanced with sophisticated techniques such as query foreground/background weighting, or using techniques of asymmetric distances like in [34], or using larger vocabularies as in [18].
However, the most substantial improvements in BoW approaches are those involving spatial reranking stages. For instance, Zhou et al. \cite{33} propose a fast spatial verification technique which benefits from the BoW encoding to choose tentative matching feature pairs between the query and the target image. Or Zhang et al. \cite{32}, who introduce an elastic spatial verification step based on triangulated graph model.

An alternative to use hand-crafted features is to exploit the fully connected layers of a convolutional neural networks information as a representation for the images. Babenko et al. \cite{5} showed how such features can reach similar performance to hand-crafted features encoded with Fisher Vectors for image retrieval. Razavian et al. \cite{19} later used several image sub-patches as input to a pretrained CNN to extract features from different locations of the image. Similarly, Liu et al. \cite{15} used features from fully connected layers evaluated on image sub patches to encode images using Bag of Words. However, these methods were not performing as well as traditional methods, mostly because the fully connected layers are more oriented to classify, and the use that they are having here is not classify but represent the image to be able to compare between them.

As a next stage convolutional layers instead of fully connected layer were used to generate the descriptors, reported significant gains in performance. However, in a convolutional layer there is a volume of neurons. A typical approach to deal with it is to do sum/max pooling of all the activation of the convolutional layers, obtaining the feature vector. Razavian et al. \cite{21} performed spatial max pooling on the feature maps of a convolutional layer of a pre-trained CNN to produce a descriptor of the same dimension as the number of filters of the layer. In this context, Babenko and Lempitsky \cite{4} proposed a compact descriptor based on sum pooling of convolutional feature maps preprocessed with a Gaussian center prior. And Kalantidis et al. \cite{12} proposed to apply directly non-parametric spatial and channel-wise weighting schemes to the convolutional features before sum pooling.

Several authors have tried to exploit local information in images by passing multiple image sub patches through a CNN to obtain local features from either fully connected \cite{19, 15} or convolutional \cite{10} layers. Although many of these methods perform well in retrieval benchmarks, they are significantly more computationally costly since they require CNN feature extraction from many image patches, which slows down indexing and feature extraction at retrieval time.

Alternatively, the convolutional features for the full image can be extracted and the activations of the different neuron arrays across all feature maps can be treated as local features. Then, when the image has to be analysed locally, there is no need to forward pass every region of the image through the network. If the location of the object of interest is known, this location can be map on the feature layers and build the feature vector applying max-pooling on these particular regions. Ng et al. \cite{16} propose to use VLAD \cite{11} encoding of features from convolutional layers to produce a single image descriptor. Arandjelovi et al. \cite{1} choose to adapt a CNN with a layer especially trained to learn the VLAD parameters. Tolias et al. \cite{28} introduce a feature representation based on the integral image to quickly max pool features from local patches of the image and encode them in a compact representation. They introduce a local analysis of multiple image patches, which is only applied to the top elements of an initial ranking. They propose an efficient workaround for sub patch feature pooling based on integral images, which allows them to quickly evaluate many image windows. Their approach improves their baseline ranking and also provides approximate object localizations. They apply query expansion using images from the top of the ranking after the reranking stage, although they do not use the obtained object locations in any way to improve retrieval performance. Mohedano et al. \cite{15} also treat the features in a convolutional layer as local features extracted at different locations in an image, using Bag of Words encoding to take advantage of sparse representations for fast retrieval in large-scale databases. Bag of words models require constructing a visual codebook to map vectors to their nearest centroid. Thus, they use k-means on local CNN features to fit this codebook. Each local CNN feature in the convolutional layer is then assigned to its closest visual word in the learned codebook. This procedure generates the assignment map that relates each local CNN feature with a visual word. This approach is the one used in the place recognition part of this thesis. They apply weak spatial verification to each target window using a spatial pyramid matching strategy. Unlike \cite{28}, they use the object localizations obtained with spatial search to mask out the activations of the background and perform query expansion using the detected object location. Figure \fig{3} shows its pipeline.

\[\text{Fig. 3: The Bag of Local Convolutional Features pipeline}\]

In order to learn a CNN for the retrieval task, instead of optimizing the network for classification, a mapping between the CNN descriptors and a space where the descriptors coming from the same images are close have to be learnt. Working on this,
the Siamese network [25], [22], learn a function that maps input patterns into a target space such that l2-norm in the target space approximates the semantic distance in the input space.

III. METHODOLOGY

This section presents the four contributions of this thesis. These four studies, combined with an off-the-shelf bag of words engine [15] used for location retrieval, build the complete architecture to retrieve specific persons in specific locations.

Section III-A describes how a feature vector is extracted from each person appearing in each keyframe of the target database. Section III-B presents the approach to enrich the query person features with a query expansion scheme. Section III-C shows the graphical user interface used to annotate the dataset and be able to obtain quantitative results. Finally, Section IV-D defines the different strategies to normalize and fuse the obtained person and the location scores for each image in the target dataset.

All the methods are designed and tested using the dataset provided by the well-known TRECVID Instance Search 2016 challenge [23], [3]. The goal of TRECVID Instance Search task is to find particular persons in specific places in a large video collection, and return a ranked list of video shots in which the person appears in the indicated place. Figure 4 shows our framework used to solve the TRECVID 2016 Instance Search task.

A. Face detection and representation

Persons depicted by the images in the database are represented in our scheme by the visual features of their faces. These features are extracted in a two-stage process: the faces are firstly detected by means of a bounding box, and a feature vector for each bounding box is generated with an off-the-shelf face descriptor.

1) Face Detection: Multiple face detection algorithms were considered before selecting the most appropriate for our application.

The first approach tested was ReInspect [26], an end-to-end approach that jointly predicts the objects in an image. This approach first encodes the image into a feature map from the top level of the GoogLeNet CNN [27]. This feature map is divided into a 15x20 grid of 1024 dimensional feature descriptors, where each cell in the grid has a receptive field of size 139x139. Reinspect has been trained to produce a set of distinct bounding boxes in the center 64x64 region, in order to be large enough to capture challenging local occlusion interactions. Then, 300 distinct LSTM controllers are run in parallel, one for each cell of the grid. The purpose of these LSTM units is to act as a controller that propagates the information between decoding steps and control the location of the next output. Thus, in each step, the GoogLeNet features are concatenated with the output of the previous LSTM unit, and the next LSTM unit is fed with this result. When a LSTM is unable to find another box in the region with a confidence above a pre-specified threshold, a stop symbol is produced. The propagation of the information through the LSTM units avoids multiple predictions of the same target, and merging or non-maximum suppression techniques are not needed, which makes this system and end-to-end trainable approach. Figure 5 shows an overview of the full process.
The second considered strategy used non-deep learning methods for face detection contained in the Menpo software library. Menpo is a python wrapper that implements some state of the art techniques and facilitates their usage and comparison. Four algorithms contained in Menpo were considered in this thesis:

- **Dlib**: is a library configured to find human faces that are looking approximately towards the camera. It works by running a sliding window classifier over an image pyramid. In particular, they use a linear classifier over a HOG pyramid, following the discussion presented in [7], and improving the HOG features as in [8].

- **OpenCV**: the library that can be found in OpenCV is also used to perform frontal face detection. It uses an implementation using Haar feature-based cascade classifiers, very similar to the one presented in [29].

- **PICO**: this library, called Pixel Intensity Comparison-based Object detection (PICO), is a modification of the standard Viola-Jones object detection method, also configured to find frontal human faces. The basic idea is to scan the image with a cascade of binary classifiers at all reasonable positions and scales. An image region is classified as an object of interest if it successfully passes all the members of the cascade. Each binary classifier consists of an ensemble of decision trees with pixel intensity comparisons as binary tests in their internal nodes.

- **FFLD**: this approach is an evolved version of the one presented in [7]. The main difference is that it uses Deformable-Part-Models (DPM) and propose to adapt the integral images to the task of face detection by using simple magnitude channels and color channels (LUV color space).

2) **Face Representation**: Once the detections are performed, the faces are cropped and a feature vector is needed to compute distances between the query images and the detected faces in the target dataset. We used the state of the art VGG-Face Descriptors [17] for this purpose. VGG-Face CNN was trained using a triplet loss scheme in order to learn a distinctive and compact projection for the score vector. It comprises 11 blocks, each containing a linear operator followed by one or more non-linearities such as ReLU and max pooling. The first eight such blocks are convolutional layers, while the last three blocks contain fully connected layers.

B. **Temporal Query Expansion**

With the aim of improving the robustness of our feature vectors for specific person retrieval, we explored the query expansion strategy in our scenario. The basic idea of query expansion [2], [6] is to extend the visual examples that define the query with additional examples. In our case, we exploited the temporal correlation of the frames in the query video to detect and use more faces.

Two main steps are needed to perform query expansion. The first one is to extract all the keyframes of the query shots provided, obtaining a sequence like the one presented in Figure 6. And the second one is to obtain a mask to use in all the newly obtained keyframes.

The masks of the persons provided by TRECVID for the four query images are very fit to their corresponding keyframe. For this reason, we applied a dilation on the original mask to take into account the small movements that a person could do within neighboring frames. Then, as only the detection of the face is needed, and in order to avoid false positives, the mask is divided in three parts and only the upper part is used to find the faces among all the possible faces detected in the query frame. An example of the pipeline is shown in Figure 7.
Fig. 6: Sequence of keyframes of one shot

Fig. 7: Mask creation pipeline

C. Annotation Tool for Person and Location Ranked Lists

This section describes the user interface developed to annotate the images of TRECVID and, this way, be able to compute a precision metric from the obtained ranked lists.

1) Front-end: The front-end of this interface was modified to be able to quickly annotate a large amount of keyframes, assisted by the search engine described in this report. Figure 8 shows a screenshot of the interface. It was programmed using HTML5 and AngularJS. The interface is divided in the following parts:

- The User box in the center-upper part, where the user can write his username to save the annotations.
- The Play and Pause buttons in the right-upper part, that allow to pause or resume the timer showed next to them.
- A drop list containing the IDs of the query topics in the left side. The user can select the topic that he wants to run the search for.
- The Search button that the user has to press to see the results of the selected topic.
- The results panel in the middle, where the retrieved keyframes are displayed. This results are the ones that has to be annotated.
- Above the results panel, the buttons that the user can click to annotate as positive, clear the annotations, or annotate as negative, respectively, over all the keyframes displayed. The images can also be annotated individually by clicking over each of them one.
- The Next and Back buttons to switch between the result pages.
- An expandable tag Examples that displays the four image example for the person.
- An expandable tag Saved displaying the positive images annotated by the user.

2) Back-end: The back-end was also modified in order to adapt it to the new annotation requirements for the TRECVID Instance Search task. As explained at the beginning of Section III this benchmark requires to create a system by fusing two approaches, one for persons and another one for locations. It means that three possible approaches should be evaluated separately: one for persons, one for locations and one assessing both persons and locations. Internally, the three types of annotations can be internally managed with only two lists: one for faces and another one for locations.

The annotation tool was launched from a Python script where four parameters were indicated: the type of annotation (persons, locations or both), query topic, ranked list to be assessed and amount of top $K$ images of the ranked list to be annotated. Once the interface was running, the typical flow was the following:

1) The user provide his username so his previous annotations can be retrieved and the new ones saved.
2) The system compares the log of previous annotations with the top $K$ elements in the rank list under evaluation. Only those images that have not been previously annotated for the current topic and annotation type are shown to the user.
3) The user manually annotates the keyframes with the described interface described in the Front-end section.
4) Once the annotations are been completed, the system is capable of computing the evaluation metrics.
D. Fusion and normalization strategies

This section presents the different normalization techniques used for both the person and the location scores, as well as the different fusion strategies used between them.

1) Normalization: Three techniques have been proposed to normalization the person and location scores so that their values would be comparable in the same scale:

- **Z-score**: this approach assumes a Gaussian distribution of the scores and normalize them based on an estimation of the mean ($\mu$) and standard deviation ($\sigma$) of the scores distribution. The source score $x_i$ is normalized into $z_i$ according to Equation 1:

$$z_i = \frac{x_i - \mu}{\sigma}$$  \hspace{1cm} (1)

- **Max-Min**: this normalization defines a linear transformation between the maximum and maximum scores, by mapping all values within the $[0,1]$ range. The normalized score is computed according to Equation 2, where $x = x_1, .., x_n$

$$z_i = \frac{x_i - \text{min}(x)}{\text{max}(x) - \text{min}(x)}$$  \hspace{1cm} (2)

- **Extreme Value Theory [20]**: this theory models the tails of a data distribution, so it is applied when dealing with extreme values. This assumption matches the retrieval problem, where the scores of the relevant samples correspond to the upper tail of the whole distribution of scores. This theory assumes that extreme values will follow a Weibull distribution, a curve that is determinated by two parameter $k$ and $\lambda$ and defined by the probability density function (pdf) in Equation 3.

$$f(x) = \frac{k}{\lambda} \left( \frac{x}{\lambda} \right)^{(k-1)} e^{-\left( \frac{x}{\lambda} \right)^k}$$  \hspace{1cm} (3)

This scores can be normalized through the Cumulative Distribution Function (CDF), shown in Equation 4. Considering the non-normalized score as the input $x$, this curve will map any value to a comparable range $[0,1]$.

$$\bar{x} = F(x) = 1 - e^{-\left( \frac{x}{\lambda} \right)^k}$$  \hspace{1cm} (4)

2) Fusion: Some strategies have been proposed to apply fusion between person and location scores:

- **Linear combination**: for each keyframe, add the corresponding scores for both the person and the place to obtain a final result. In this case, different weights can be tested for the two parts.

- **Maximum**: take the maximum value between the person and the location scores.
- **Minimum**: take the minimum value between the person and the location scores.
- **Product**: in this case the scores can be interpreted as a probabilities, so a product can be applied.

## IV. Experiments and Results

This section explains the different experiments that have been performed in this thesis, as well as their results. Section IV-A provides details of the dataset used in TRECVID 2016 Instance Search. The qualitative study about face detector and query expansion developed before the TRECVID submission on July 1st 2016 are presented in Sections IV-B and IV-C respectively. In the second block of experiments in Section IV-D a quantitative analysis is provided for the fusion strategies, thanks to the annotation of a subset of the data with the presented user interface. These experiments explore different strategies of fusion of the scores based on the person and location scores.

### A. Dataset

The participants to TRECVID Instance Search are provided with approximately 244 video files (totally 300 GB, 464 h) with associated metadata, each containing a week’s worth of BBC EastEnders programs in MPEG-4/H.264 format. Additionally, each video is divided in different shots of short duration (between 5 seconds and 2 minutes).

**People and locations query set.** The query set consists on 30 different topics, each one including: (1) a person textual description among a total of 7 persons. (2) A location textual description among a total of 10 locations. (3) 4 image examples of the person. (4) Between 6 and 12 image examples for the location. (5) For each one of the 4 person image queries, a binary mask is also provided. (6) The shot where the keyframes belongs.

**Target database.** In order to handle the large amount of video information provided, a target database was built by uniformly extracting keyframes for every shot with a sample rate of 1/4 fps. The resulting dataset contained 1.5M of keyframes and had a total size of 120GB. Thanks to the annotation tool described in Section III-C, a total amount of 3,991 shots have been annotated for the persons, and a total of 1,528 shots have been annotated for locations. It corresponds to 794 common shots annotated.

### B. Qualitative Study of Face Detectors

1) **Data adaptation for ReInspect:** The first experiments are performed using ReInspect, presented in Section III-A. This network has an input resolution of 640x480, smaller than the resolution of TRECVID images (768x576). To address this difference in the input resolution, two configurations were considered: the first one consisted in changing both the input size of the network, which also implied changing the grid size, and the size of our images. This allowed us to use the images without padding. Several experiments changing this size were done, and a final size of 320x224 was found as the best one, based on visual inspection of the results. The results of this configuration can be found in Figure 9.

![Fig. 9: Results of ReInspect in some query images](image)

As it can be observed, even if this is the best size for this configuration, results are not good enough because there are many missing detections. Thus, a second configuration can be done keeping the input size of the network and changing the size of the query image, adding padding to reach the input size. In this case, and also after a qualitative evaluation, the best size for the image has been also 320x224, adding padding to be 640x480. In Figure 10 the results can be found. As it can be observed, they are slightly better than in the previous configuration, but they still present many false negatives.

ReInspect uses a GoogLeNet network fine-tuned to detect faces in the images. The images used to fine-tune it are like the ones presented in Figure 11. As can be observed these faces are quite small in contrast with the ones that can be found in TRECVID images. Thus, the network is not prepare to detect such a big faces and for this reason fails in TRECVID images.

2) **Menpo library evaluation:** The next experiments are using the libraries of Menpo, presented in Section ???. The goal is to decide which face detection is the most suitable for our images. As in the previous subsection, results are analysed qualitatively, since no ground truth data is available. The first library used is *Dlib*, and the results can be observed in Figure 12. Some false negative are still observed, but an improvement can be observed with respect to ReInspect results.

---

^2Both, the image examples for person and locations are keyframes from the first video out of the 224 videos provided by TRECVID.
The next experiment is using OpenCV. In Figure 13 a similar performance to the one founded using Dlib is observed. As we can observe there are still some false negative.

Finally, the last experiment is done using FFLD. The results are presented in Figure 15. An improvement with respect to all the previous results is observed, and only a few false positive appear. In order to solve the problem of the false negative, an approach equalizing the image before using FFLD is propose, and as it can be observed in Figure 16 it improves a little bit the detection.

The main issues with the three first libraries, is that they are used to find human faces that are looking more or less towards the camera, but in our images the faces are slightly turned. So, it is observed that a well trained vanilla DPM approach obtains best results in our TRECVID dataset that other approaches based on the well-known Viola-Jones approach.

C. Qualitative Evaluation of Query Expansion

As proposed in Section III-B a query expansion is performed in order to obtain more query faces and improve the face recognition. Figure 17 is an example of how, using the proposed approach, some false positive can be avoided. Then, Figure 18 shows the results for the query expansion that has been used as an example in Section III-B and we can see how the faces are correctly detected.
Even though the query expansion itself works quite well, in the general pipeline the results do not improve as we expected. The main problem is that the new query faces that we can extract from the query shots are very similar. As a consequence, the retrieved images are very similar to the query and the final results end up being the same.

D. Quantitative Evaluation of Person-Location Fusion

This section explores the results obtained combining the different strategies proposed in Section IV. In this case, quantitative results were generated for a better analysis of the performance of our system.

The annotations of the TRECVID dataset used in this work were not available at the time of writing this thesis. Moreover, the huge amount of keyframes included in the target dataset (1.2 million) did not allow a full annotation with the available resources. For these two reasons, an annotation tool was developed to annotate a part of it that would allow comparing the results of our different configurations, as previously described in Section III-C.

The user interface allows annotating the top $K$ elements of each ranked list generated by the different configurations presented in this work. This annotation allows computing the Precision at position $K$ ($P@K$) for each topic, as defined in Equation 5, where $M$ is the set of relevant images in the dataset for the query topic, and $N_K$ is the set of the top $K$ elements in the ranked list under evaluation.

\[
P@K = \frac{|M \cap N_K|}{K}
\]  

(5)

The mean $P(K)$ across the whole set of query topics $Q$ defines the Mean Precision at $K$ ($mP(K)$), as defined in Equation 6.

\[
mP@K = \frac{1}{|Q|} \sum_{i=1}^{Q} P_i@K
\]  

(6)

The first step in order to evaluate the fusion strategies is to make sure that the different parts work well separately. This implies evaluating the rankings from both parts before making the fusion. The mean precision over the firsts 100 keyframes is presented in Table I. The used rankings correspond to the ones used for the TRECVID Instance Search 2016 submission. Three rows can be observed, the first one is the results for the person, the last one is the one for location, and the results in the middle correspond to apply query expansion. Thus, even though the query expansion based on neighbouring temporal frames did not work as expected, another alternative was using a pseudo-relevance feedback [30]. This strategy corresponds to using as a new queries the faces from the top 20 keyframes in the person ranking for each topic.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>mP@100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Persons</td>
<td>90.36</td>
</tr>
<tr>
<td>Persons + top20</td>
<td>94.53</td>
</tr>
<tr>
<td>Locations</td>
<td>94.10</td>
</tr>
</tbody>
</table>

TABLE I: Results of the parts separately over 100 keyframes

The mP@100 obtained are quite high in all the parts. However, during the annotation of locations an issue in the location rankings was observed. Even though the precision over 100 was high, after the top 100 images the retrieved keyframes were
totally random, probably due to a problem or bug in the off-the-shelf software running the location ranking. Thus, to be able to observe improvements between the different fusion and normalization strategies, we opted to evaluate precision over the top 50, and the new results for the three cases presented in the previous table can be observed in Table II. As the results are higher applying the query expansion, this rankings will be the ones used in the next experiments when referring to the person rankings.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>mP@50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Persons</td>
<td>91.73</td>
</tr>
<tr>
<td>Persons + top20</td>
<td>95.80</td>
</tr>
<tr>
<td>Locations</td>
<td>98.53</td>
</tr>
</tbody>
</table>

**TABLE II: Results of the parts separately over 50 keyframes**

1) **Normalization**: The different types of normalizations were tested and its results presented in Table III. It contains four columns: the first one indicating the normalization type, the second and the third one that correspond to evaluating only the people and the locations on the rankings obtained from the fusion, and the last one, which is evaluates if the specific person is in the specified location.

<table>
<thead>
<tr>
<th>Normalization type</th>
<th>Person mP@50</th>
<th>Location mP@50</th>
<th>Final mP@50</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>73.00</td>
<td>45.86</td>
<td>22.26</td>
</tr>
<tr>
<td>Max-min</td>
<td>68.53</td>
<td>42.13</td>
<td>13.36</td>
</tr>
<tr>
<td>Z-score</td>
<td>95.60</td>
<td>9.46</td>
<td>9.26</td>
</tr>
<tr>
<td>Extreme value</td>
<td>15.80</td>
<td>97.06</td>
<td>15.46</td>
</tr>
</tbody>
</table>

**TABLE III: Results of applying different normalization**

The one that gives the highest results is without applying any kind of normalization. In the case of the z-score normalization, even though it has been reported to work very well in many cases, it fails if the distribution of this data is not gaussian. As it can be observed in the locations distribution example in Figure 19, the location scores do not satisfy this. In the following experiments no normalization is applied, as it has proven to give the highest scores.

2) **Weighting linear combination**: Different weights for the person scores and the location scores has been applied during the linear combination of scores. In Table IV we can observe how the results are higher when the location has more weight. This is probably due to more images in the person rankings have a high score, while in the location rankings less images have
high score. Thus, when the fusion is applied without weighting the person rankings are predominating, while weighting higher the location score this difference is a little bit compensated.

<table>
<thead>
<tr>
<th>Person weight</th>
<th>Location weight</th>
<th>Person mP@50</th>
<th>Location mP@50</th>
<th>Final mP@50</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>0.7</td>
<td>26.06</td>
<td>94.13</td>
<td>22.53</td>
</tr>
<tr>
<td>0.4</td>
<td>0.6</td>
<td>44.60</td>
<td>75.86</td>
<td>23.46</td>
</tr>
<tr>
<td>0.5</td>
<td>0.5</td>
<td>73.00</td>
<td>45.86</td>
<td>22.26</td>
</tr>
<tr>
<td>0.6</td>
<td>0.4</td>
<td>91.86</td>
<td>23.46</td>
<td>18.86</td>
</tr>
</tbody>
</table>

**TABLE IV:** Results of applying different weights to linear combination

3) Maximum and Minimum: The results of applying Maximum and Minimum fusion can be observed in Table V. As expected the Maximum fusion does not work well, as it takes the maximum score for each shot between the person and the location rankings, which means that, if as supposed the person rankings are higher, the results will be biased towards the persons ignoring the locations. A prove of that is reflected in the fact that the Person mAP is higher for the Maximum, while the Location mAP is lower. In contrast, the Minimum fusion works better. This is because by taking the minimum, the non-relevant results are totally discarded, giving preference to those where both the person and the location is relevant.

<table>
<thead>
<tr>
<th>Normalization type</th>
<th>Person mP@50</th>
<th>Location mP@50</th>
<th>Final mP@50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum</td>
<td>95.80</td>
<td>7.53</td>
<td>7.53</td>
</tr>
<tr>
<td>Minimum</td>
<td>20.46</td>
<td>95.06</td>
<td>20.06</td>
</tr>
</tbody>
</table>

**TABLE V:** Results of applying maximum and minimum fusion
V. CONCLUSIONS

The main aim of this thesis was to obtain a baseline to be able to participate in TRECVID Instance Search challenge. On July, just two months after starting the work, we submitted four configurations of the results obtained with our pipeline. For this reason, we consider that the main goal of the thesis was accomplished.

The first idea was to use convolutional neural networks to perform face detection. However, the main problem in this case was that we did not have any training dataset, so we had to use a pre-trained network for face detection. This supposed a drawback, as the network that we chose, which was Relinspect, was trained to detect small faces in crowded scenes, the situation very different to our video frames where the faces are large.

Then, we decided to use other techniques wrapped in Menpo. In the four cases presented in Menpo, the results were better than Relinspect. However, the three approaches based on Viola-Jones were slightly worse than the FFLD approach, which is based on a vanilla DPM, combining the information of the image in LUV color-space. In addition, an image equalization helped to improve the false negatives that we still found in the images.

A query expansion was also performed using the query shots, and the new masks to detect only the faces that we needed. However, all the faces in a shot are very similar, and it did not improve the results as we expected.

After the TRECVID submission, a user interface was developed in order to obtain a measure of the system, that allows us to evaluate the different parts and obtain some conclusions about the weak points of our submission. Using this UI, different normalization and fusion strategies were tested achieving the best results by no using any normalization and weighting the linear combination, giving more weight to the location scores.

During these last experiments, a possible issue in the location part was found. Thus, as a future work we would like to analyse it deeper and try to improve the location rankings.

ACKNOWLEDGMENT

I would like to thank my tutors, Xavier Giro-i-Nieto and Noel E. O’Connor to make possible my collaboration in this project, as well as all their effort to make my Erasmus in Dublin possible. I would also like to thank Kevin McGuinness and Eva Mohedano for their patience when guiding me and teaching me, and most of all, for helping me whenever their could.

The video frames from BBC Eastenders video used in this document are programme material copyrighted by BBC.

REFERENCES


