Score Propagation Based on Similarity Shot Graph for Improving Visual Object Retrieval

Juan Manuel Barrios
ORAND S.A.
Santiago, Chile
juan.barrios@orand.cl

Jose M. Saavedra
ORAND S.A.
Santiago, Chile
jose.saavedra@orand.cl

ABSTRACT

The Visual Object Retrieval problem consists in locating the occurrences of a specific entity in an image/video dataset.

In this work, we focus on discovering new occurrences of an entity by propagating the detection scores of already computed candidates to other video segments. The score propagation follows the edges of a pre-computed Similarity Shot Graph (SSG). The SSG connects video segments that are similar according to some criterion. Four methods for creating the SSG are presented: two based on computing and comparing low-level visual features, one based on comparing text transcriptions, and other based on computing and comparing high-level concepts.

The score propagation is evaluated on the INS 2014 dataset. The results show that the detection performance can be slightly improved by the proposed algorithm. However, the performance is variable and depends on the properties of the SSG and the target entity. It is part of the future work to automatically decide the kind of SSG that will be used to propagate scores given a set of detection candidates.

1. INTRODUCTION

The Visual Object Retrieval problem consists in locating the occurrences of a specific entity in an image/video dataset. The entity is defined by one or more visual examples and optionally the location of the entity in each example. In the case of video datasets, the object retrieval system reports all the shots or video segments where the entity is visible and optionally a bounding box with the location of its occurrence. The entity may be some person, some building, the trademark logo of some company, a location, etc. Figure 1 shows two visual examples defining the entity "wheelchair" to be searched in the INS 2014 dataset. The visual object retrieval problem differs from near duplicate detection, because an object may occur in scenarios or contexts that may have not been specified by the visual examples. It also differs from the classification problem because it intends to locate a specific instance of a class rather than occurrences of generic classes, e.g., an object retrieval system may look for a specific dog instead of the generic class of dogs.

Most of current research focuses on developing effective and efficient methods for generating candidates from large datasets. The bag-of-words provides an effective and efficient approach for processing large video datasets. Different variations and improvements to this approach have been proposed, like hierarchical codebooks [7], Hamming embedding [5], or accumulating differences to codewords [4].

This work focuses on discovering new occurrences of an object given a set of already computed candidates. A Similarity Shot Graph (SSG) is pre-computed for a video dataset. The graph connects shots according to some criterion, like low-level visual similarity, similarity of speech or subtitles, or even similarity based on shared semantic concepts. The graph is used to propagate detection scores of candidates to other similar shots in the dataset in order to discover new occurrences.

Some works use a pre-computed graph to improve search results or to provide insights on a dataset. In the case of image collections, a graph can be computed in order to connect images that represent the same object from different viewpoints [10]. Also the graph can be used to enhance the descriptor of an image by using the codewords from other images in the neighborhood of the graph [13] [2]. In the case of video collections, a graph can be computed to connect duplicated video sequences in the collection [11] [9]. Unlike these examples, our approach does not restrict the graph to near-duplicates. In the experimental section we test different kinds of similarity to compute a SSG.

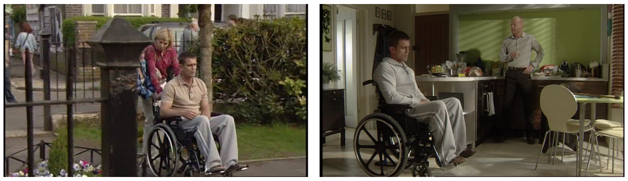

Figure 1: Two visual examples of "9125 this wheelchair with armrests" from INS 2014 dataset. Programme material ©BBC.

2. SIMILARITY SHOT GRAPH

A shot is a series of interrelated consecutive frames in a video scene. Given the set S of n shots $S = \{s_1, ..., s_n\}$ from

SLAM'15, October 30, 2015, Brisbane, Australia.
© 2015 ACM. ISBN 978-1-4503-3749-6/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2802558.2814644.

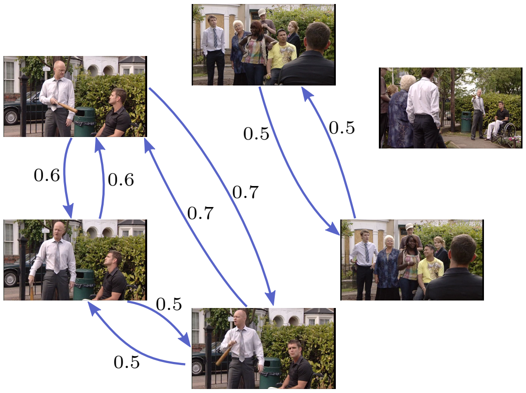

Figure 2: Similarity Shot Graph for some shots in the same scene. The weights represent similarity between shots according to some criterion. Programme material ©BBC.

a video collection, the Similarity Shot Graph (SSG) is defined as a weighted graph where the set of nodes corresponds to S and the edge weights $w(s_i, s_j)$ represents the degree of similarity between shots s_i and s_j. The weights are normalized to the range $[0, 1]$, where zero means no similarity and the unitary weight means high similarity. Different criteria can be use to define similarity between shots, e.g., visual similarity of frames, similarity of acoustic tracks, similarity of visible objects, similarity between speech, etc. Note that the SSG is created offline prior to any query definition.

Usually, a shot division produces fine-grained segmentation of a scene. If a static object is visible in a shot, it may be expected that the object will also be visible in other shots from the same scene.

In this work, we evaluate three criteria to construct a SSG: low-level visual similarity, as reported by a near duplicate detection system, speech similarity, defined as the degree of closeness of the words spoken during a time interval, and concepts similarity, defined as the degree of closeness of the distribution of elements detected by a concepts detector.

In the case of low-level visual similarity, a simple method for building a SSG is by computing a global descriptor for a representative frame of each shot. A k-NN search for each frame retrieves other similar frames in the dataset. Then, the distances are converted to similarity values which correspond to the edge weights. This method produces a SGG that connects shots that are visually alike, as usually happens during a scene (see Figure 2). This method can be improved by selecting three frames per shot *start*/*middle*/*end* instead of a single frame. A global descriptor is computed for every selected frame and then a k-NN search is performed for everyone. Then, a graph edge (s_i, s_j) is created if simultaneously the frame $start(s_j)$ is one of the k-NN of $start(s_i)$, $middle(s_j)$ is one of the k-NN of $middle(s_i)$, and $end(s_j)$ is one of the k-NN of $end(s_i)$, and the edge weight is the average similarity of the three matched frames. This method produces a stronger similarity than comparing a single keyframe but weaker than a duplicate detection.

In the case of speech similarity, all the words spoken inside the shot boundaries are accumulated to create a vector following the *tf-idf* model. Hence, for each shot a single vector summarizes all the speech in the shot. Thereafter, for each vector the k most similar vectors whose cosine similarity is above a threshold are retrieved. The edge weight is defined as the cosine similarity value between the two shots. The spoken words can be obtained by using subtitles, closed captions, transcriptions, or as the result of an ASR process.

In the case of object detection, a general concept detector is used to detect concepts on frames. The detected concepts are accumulated to create a *tf-idf* vector for each shot. The graph computation is analogous to the speech similarity but replacing spoken words by detected concepts.

3. SCORE PROPAGATION

Given a detection score for each shot $R{=}\{(s_i, \text{score}_i), i \in \{1, ..., n\}\}$, $s_i \in S$, $\text{score}_i \geq 0$, the score propagation algorithm uses the SSG to compute a new set of detection scores $R'{=}\{(s_j, \text{score}'_j), j \in \{1, ..., n\}\}$ where:

$$\text{score}'_j = \text{score}_j + \sum_{i=1}^{n} \text{score}_i \cdot w(s_i, s_j)$$

The rationale for the score propagation is to improve the detection of objects that are related to the context, i.e., when the object is somehow linked to environment where it was detected. In this case, the context is represented by the criterion used to build the SSG. For example, if the SSG is based on low-level visual similarity then the detections will propagate to other shots that are visually alike, which may be a good option for static or fixed objects (e.g., a decoration), but if the SSG is based on speech similarity, the detections will propagate to shots where the speech is similar, which may be a good option for a character or a location. It is an open issue to automatically decide the type of SSG to use for each object, in the future work we outline an idea to address this issue.

We should note that the SSG may also propagate noisy scores, even decreasing the effectiveness of the system, especially if the baseline contains several false alarms. Hence, the SSG should be sparse, containing edges only between those shots that are indeed similar. A direct solution for this issue is to set a threshold ω and to propagate scores only on edges where $w(s_j, s_i) \geq \omega$. Another option is to moderate the effect of the propagation by scaling down the propagated scores by a factor $\gamma < 1$, thus the score propagation becomes:

$$\text{score}'_j = \text{score}_j + \gamma \sum_{i=1}^{n} \text{score}_i \cdot \hat{w}(s_i, s_j)$$

$$\hat{w}(s_i, s_j) = \begin{cases} w(s_i, s_j) & \text{if } w(s_i, s_j) \geq \omega \\ 0 & \text{otherwise} \end{cases}$$

In addition, the SSG can also be used to propagate user decisions. In the case of interactive systems or systems with user feedback, when a user states that a candidate shot is correct or incorrect, the decision (represented as an increase or decrease of the score) can be propagated to other shots following the edges in the SSG. We should mention we tested a propagation algorithm considering more than one level of connectivity in the graph without satisfactory results.

4. EVALUATION

TRECVID is an evaluation sponsored by the National Institute of Standards and Technology (NIST) with the goal

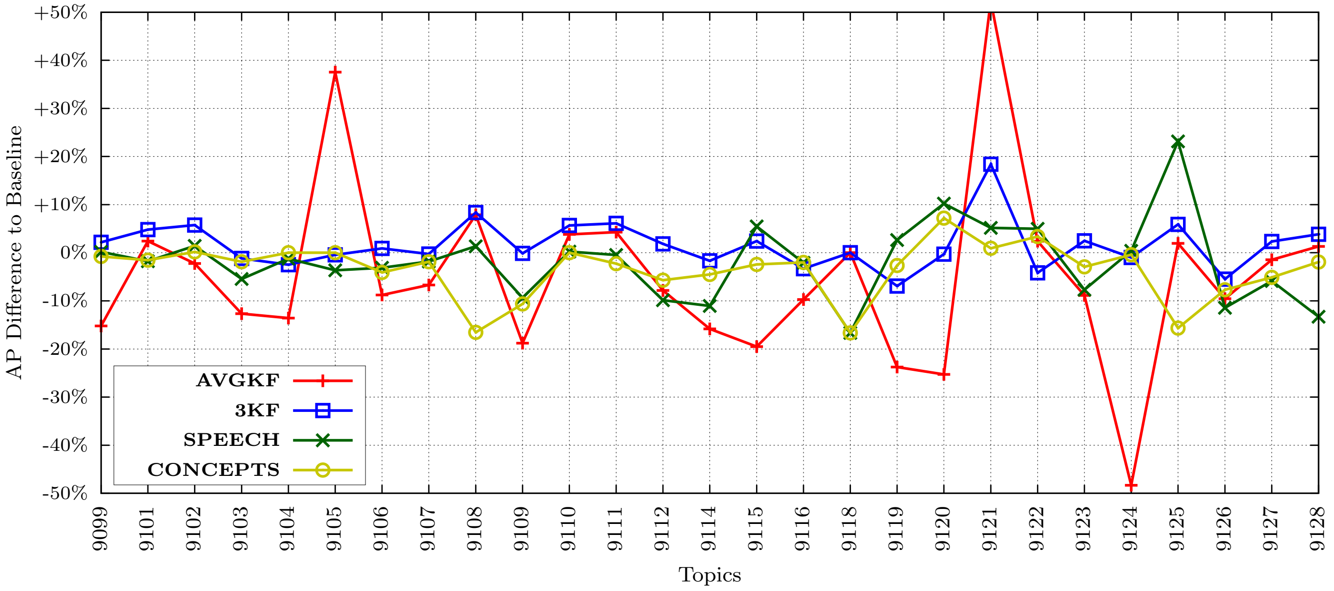

Figure 3: Improvements on average precision (AP) achieved by score propagation of the Baseline candidates using four SSGs compared to the AP achieved by the Baseline.

SSG	MAP	Difference to Baseline
Baseline	0.222	–
AVGKF	0.212	-4%
3KF	0.227	+2%
SPEECH	0.213	-4%
CONCEPTS	0.211	-5%

Table 1: Results on the 27 topics when a SSG is used for score propagation on the Baseline results.

of encouraging research in video information retrieval [12]. Since 2010 TRECVID includes an evaluation task devoted to the visual object retrieval problem, called Instance Search (INS). The INS task consists in retrieving shots from a video collection containing a target object, called a *topic*. The topic is defined by visual examples and a brief textual description. A visual example comprises a still image and a mask, which outlines the region in the still image where the topic is visible. INS 2014 evaluated 27 topics with four visual examples per topic [8]. The dataset is the BBC EastEnders collection, which consists in 244 videos with a total extension of 435 hours. The dataset contains 471,526 shots with an average shot length of 3.3 seconds. For each topic a system must report the top 1000 shots with a detection score.

The names of the 27 evaluated topic at INS 2014 are the following: **9099** a checkerboard band on a police cap, **9101** a Primus washing machine, **9102** this large vase with artificial flowers, **9103** a red, curved, plastic ketchup container, **9104** this woman, **9105** this dog, Wellard, **9106** a London Underground logo, **9107** this Walford East Station entrance, **9108** these 2 ceramic heads, **9109** a Mercedes star logo, **9110** these etched glass doors, **9111** this dartboard, **9112** this HOLMES lager logo on a pump handle, **9114** a red public mailbox, **9115** this man, **9116** this man, **9118** a Ford Mustang grill logo, **9119** this man, **9120** a wooden park bench, straight-backed, with flat arm rests, **9121** a Royal Mail red vest, **9122** this round watch with black face and black leather band, **9123** a white plastic kettle with vertical blue window, **9124** this woman, **9125** this wheelchair with armrests, **9126** a Peugeot logo, **9127** this multicolored bust of Queen Victoria, **9128** this F pendant.

In order to evaluate the impact of score propagation, we use as a Baseline the participation at INS 2014 of the ORAND team [3]. Briefly, the Baseline was computed following these steps: sampling five frames per second for each video, extracting CSIFT [1] descriptors for all selected frames and for all visual examples. For each local descriptor in a visual example compute the list of the k-NN in the database (k=100). Finally, a voting algorithm ranks each shot according to the number of nearest neighbors they contain for each visual example, weighting the votes by the position of the query vector (inside or outside the mask). The Baseline achieves a Mean Average Precision (MAP) 0.222.

In this experiment we compute four SSGs for the EastEnders collection, and then we measure the result of propagating the scores in the Baseline to similar shots. The four SSG are: **AVGKF**, which represents a low-level visual similarity graph whose edge weights are computed as the difference between the average frame of shots (frames are reduced to 12×9 and compared with distance L_1); **3KF**, which represents a low-level visual similarity graph based on three descriptors per shot as described in Section 2; **SPEECH**, which stores the similarity between shots based on the transcription texts provided by BBC; and **CONCEPTS**, which represents the similarity between shots computed by comparing the concepts detected using the Caffe framework with the pre-trained model AlexNet [6] on sampled frames for each shot, where the concepts are those defined in the ImageNet collection.

The four graphs were restricted to create edges only between shots from the same video with a similarity higher than ω=0.5. Their sizes are: **AVGKF** contains 21.7 million edges, **3KF** has 0.57 million edges, **SPEECH** has 2.2 million edges, and **CONCEPTS** has 6.4 million edges. The maximum number of edges is more than $2 \cdot 10^{11}$, thus these graphs are all sparse (their densities are lower than 0.01%).

The Table 1 shows the result of using a SSG for propagating scores on the Baseline. The **3KF** graph achieves on average a slight improvement, however it is not significant, while the other three graphs produce a small decrease in effectiveness. The **3KF** graph is stricter at creating edges than the other three graphs, hence it contains less and more meaningful edges, while the other three graphs contains more edges that can propagate noise or produce false positives. The propagation weight γ was fixed to 0.5.

The Figure 3 depicts the increase/decrease of the Average Precision on each of the 27 topics in proportion to the Average Precision achieved by the Baseline. The performance of the score propagation depends on the topic and on the type of graph. In particular, the **AVGKF** graph shows some big improvements in topics **9105** and **9121** where the occurrences in both topics are highly related to a physical location, while in topics **9119** and **9124** the decrease is due to the target objects (background cast members) are not related to the background hence the propagation adds false positives. The **3KF** graph does not show that extreme behavior mainly due to its reduced number of edges which produces score propagation only at a small scale without a big impact on the Baseline. Both **SPEECH** and **CONCEPTS** produces too many false positives, thus in general they reduce the effectiveness compared to the Baseline. However, there are some topics where both graphs produces an improvement compared to the Baseline, like topics **9120** and **9125**, where the score propagation based on high-level features can improve the detection effectiveness.

5. CONCLUSIONS AND FUTURE WORK

We have proposed a mechanism for discovering new occurrences of an object by propagating the score detection to other shots that are similar. The rationale for the score propagation is to improve the detection of objects that are somehow related to a context. The contexts are represented by a graph which links video segments that are similar according to some criterion. We presented three criteria to build the Similarity Shot Graph: a low-level visual similarity, a speech-based similarity, and similarity of detected concepts. The results show that the performance of the score propagation is highly dependent on the properties of the target object. In addition, the SSG can also be used in other tasks like propagating user feedback or propagating recommendations.

It is an open issue to decide which kind of context will be useful to discover more occurrences of an object. In principle, this issue could be delegated to the user who has to decide the properties of the target objects. However, we think the best performing graph can be guessed by comparing the already discovered occurrences and the edges in the graph: if there is a fit between already known occurrences and SSG edges then some properties on the target object could be assumed. More research is needed in order to gain insight about this issue and the scenarios were the score propagation can be successfully applied.

6. ACKNOWLEDGEMENTS

We are grateful for financial support from CONICYT through projects PAI-781204026 and PAI-781204025 as well as CORFO-INNOVA through the project 15ITE2-38948. The results of the Baseline were completed thanks to the High Performance Computing infrastructure of the National Laboratory for High Performance Computing (NLHPC) ECM-02.

7. REFERENCES

[1] Feature Detectors and Descriptors: The State Of The Art and Beyond. Feature Detection Code., 2010. `http://kahlan.eps.surrey.ac.uk/featurespace/web/`.

[2] R. Arandjelovic and A. Zisserman. Three things everyone should know to improve object retrieval. In *Proc. of CVPR*, pages 2911–2918. IEEE, 2012.

[3] J. M. Barrios, J. M. Saavedra, F. Ramirez, and D. Contreras. Orand at trecvid 2014: Instance search and multimedia event detection. In *Proc. of TRECVID*. NIST, USA, 2014.

[4] J. Delhumeau, P.-H. Gosselin, H. Jégou, and P. Pérez. Revisiting the vlad image representation. In *Proc. of ACM Multimedia*, pages 653–656. ACM, 2013.

[5] H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *Proc. of ECCV*, pages 304–317. Springer, 2008.

[6] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[7] D.-D. Le, C.-Z. Zhu, S. Poullot, V. Q. Lam, D. A. Duong, and S. Satoh. National institute of informatics, japan at trecvid 2011. In *Proc. of TRECVID*. NIST, USA, 2011.

[8] P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A. F. Smeaton, and G. Quéenot. Trecvid 2014 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2014*. NIST, USA, 2014.

[9] L. Pang, W. Zhang, H.-K. Tan, and C.-W. Ngo. Video hyperlinking: Libraries and tools for threading and visualizing large video collection. In *Proc. of ACM Multimedia*, pages 1461–1464. ACM, 2012.

[10] J. Philbin and A. Zisserman. Object mining using a matching graph on very large image collections. In *Proc. of ICVGIP*, pages 738–745. IEEE, 2008.

[11] S. Poullot, M. Crucianu, and O. Buisson. Scalable mining of large video databases using copy detection. In *Proc. of ACM Multimedia*, pages 61–70. ACM, 2008.

[12] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *Proc. of MIR*, pages 321–330. ACM, 2006.

[13] P. Turcot and D. G. Lowe. Better matching with fewer features: The selection of useful features in large database recognition problems. In *Proc. of ICCV*, pages 2109–2116. IEEE, 2009.