IMAGE CLUSTERING THROUGH COMMUNITY DETECTION ON HYBRID IMAGE SIMILARITY GRAPHS

Symeon Papadopoulos[1,3], *Christos Zigkolis*[1,3], *Giorgos Tolias*[2], *Yannis Kalantidis*[2], *Phivos Mylonas*[2], *Yiannis Kompatsiaris*[1], *Athena Vakali*[3]

[1]Informatics and Telematics Institute, CERTH, 57001, Thessaloniki, Greece
[2]Image, Video and Multimedia Systems Laboratory, NTUA, 15780, Athens, Greece
[3]Department of Informatics, Aristotle University, 54124, Thessaloniki, Greece

ABSTRACT

The wide adoption of photo sharing applications such as Flickr© and the massive amounts of user-generated content uploaded to them raises an information overload issue for users. An established technique to overcome such an overload is to cluster images into groups based on their similarity and then use the derived clusters to assist navigation and browsing of the collection. In this paper, we present a community detection (i.e. graph-based clustering) approach that makes use of both visual and tagging features of images in order to efficiently extract groups of related images within large image collections. Based on experiments we conducted on a dataset comprising publicly available images from Flickr©, we demonstrate the efficiency of our method, the added value of combining visual and tag features and the utility of the derived clusters for exploring an image collection.

Index Terms— content-based image retrieval, image clustering, visual similarity, tags, community detection

1. INTRODUCTION

The rising popularity of photo sharing applications over the web has led to the generation of huge amounts of personal image collections. The lack of supporting mechanisms for efficient browsing, search and retrieval of content within them deteriorates considerably the overall image browsing experience and the user satisfaction. For that reason, a set of content navigation technologies, such as tagging, related image suggestion and clustering have become popular in such applications. Image clustering in particular is an extremely valuable feature for photo sharing sites, since it enables a top-down exploratory process during image browsing. At the same time, it improves user experience by (a) returning faster and more accurate results and (b) enabling organization of the personal content.

Despite the recognized value of using image clustering to aid user navigation and browsing in photo sharing sites [1], its use has been rather limited to date due to a series of issues faced by conventional clustering techniques. The most profound among these limitations is the high computational complexity of existing approaches, which renders them impractical for use in large-scale problems. In addition, many of the clustering techniques operate in a supervised way, i.e. they require the number of clusters to be provided a priori as input, which further complicates their application on user contributed content. Finally, it is always a challenging task to interpret the clustering results and assess their utility to real users.

This work was supported by the European Commission under contract FP7-215453 WeKnowIt.

Motivated by the above observations, we introduce in this work the use of *community detection* methods [2] (i.e. a kind of graph-based clustering) on image similarity graphs in an attempt to address the aforementioned limitations of previous image clustering schemes. In our approach, we make use of both visual and tag similarities among the images of a large collection in order to derive a hybrid image similarity graph. We use this graph to extract meaningful image clusters by use of community detection, i.e. by finding groups of nodes that are more densely connected to each other than to the rest of the network. The proposed scheme is computationally efficient and does not require the number and size of clusters as a priori parameters. Furthermore, we conduct a two-layer evaluation on the clustering results in order to demonstrate that the derived clusters contain images that (a) are geographically very close to each other, and (b) are perceived by users as semantically relevant to each other.

In Section 2, we briefly review previous works on the problem of web image clustering. The proposed framework is described in Section 3. We then present the evaluation of the proposed method in Section 4. and conclude in Section 5.

2. RELATED WORK

There has been considerable interest in web image clustering for some time now. Many efforts were based on a single aspect of images, either solely on their visual features [3] or on their textual descriptions [4]. Cai et al. were among the first to make use of multiple types of similarity between images, namely visual, text and hyperlink-based similarity [1]. However, they combined the different image similarities in a cascaded scheme. Gao et al. [5] proposed a multi-objective optimization technique to simultaneously use both visual and text-based image similarities in the clustering process.

The recent advent of Web 2.0 technologies that endowed photo sharing applications with social tagging features revamped the interest in image clustering by use of visual, tagging and additional user-contributed information. For instance, Moëllic et al. [6] make use of both tagging and visual similarity features to either perform two independent image clusterings or to combine them in a single clustering scheme through an early fusion approach. Furthermore, Quack et al. [7] organize geotagged flickr images into clusters corresponding to real-world objects or events based on their visual, tag and spatial proximity.

The main shortcoming of previous approaches is their reliance on complicated clustering schemes, such as spectral graph partitioning, that suffer from either or both of the following problems: (a) high computational and memory requirements and (b) need for setting the number of clusters as an algorithm parameter. Therefore,

their applicability to large photo collections is rather limited. In our framework, we make use of *community detection* to efficiently identify clusters of images on an image similarity graph that is constructed based on both visual and tag features. Apart from being efficient, the proposed algorithm assigns to clusters only those images that are related to each other, while leaving out outliers, thus increasing the precision of the derived clusters. To our knowledge, this is the first time that such a graph-based clustering scheme is applied on a tagged photo collection.

3. PROPOSED IMAGE CLUSTERING FRAMEWORK

The proposed image clustering framework relies on the creation of two image graphs representing two kinds of similarity between images of the collection, i.e. based on their visual features and their tags, respectively. These are detailed in paragraphs 3.1 and 3.2. Subsequently, an efficient graph-based clustering scheme, described in paragraph 3.3, is applied on the union of the two graphs in order to identify sets of nodes (i.e. image clusters) that are more densely connected to each other than to the rest of the network.

3.1. Visual similarity graph creation

For the representation of the visual content of a given image, a set of local visual features are detected and a descriptor is extracted from their surrounding area. In our approach, we selected the SURF (Speeded-Up Robust Features) [8] features to represent the visual properties of the images, since they have been proven to achieve high repeatability and distinctiveness.

Building a Hierarchical Vocabulary: Having collected SURF descriptors from a large corpus of images, we use a clustering process to create a visual vocabulary. A visual vocabulary is analogous to a typical language vocabulary, with an image corresponding to a piece of text. In the same way that text may be decomposed to a set of words, an image can also be decomposed to a set of *visual* words. Then, in order to compare two images, the sets of their corresponding visual words may be compared instead.

Using the K-means algorithm in a large number of descriptor vectors would be significantly slow, thus we decide to use a Hierarchical K-means (HKM) approach followed by an agglomerative merging step (Reciprocal Nearest Neighbor algorithm [9]) in order to create more discriminative visual words [10]. Each level of the aforementioned hierarchical structure forms a visual vocabulary and all levels a vocabulary tree. Each local feature is mapped to a visual word by descending the tree. Mapping all features to visual words, each image can be represented by the Bag of Words (BoW) model and the TF-IDF weighting scheme [11].

Matching and Spatial Verification: To compute the similarity between two given images we use a histogram intersection. Exploiting the sparsity of the BoW representation when using a 10^4 vocabulary, an inverted file structure is used to speed-up the matching process. A spatial verification step follows so as to re-rank the top ranked images. We use a deterministic modification of RANSAC to determine the affine transformation between image pairs (Figure 1).

Using single correspondences for hypothesis generation makes it feasible to evaluate all possible transformations based on the tentative correspondences, thus removing the randomness of the algorithm [12]. Although the initial hypothesis is a similarity transformation, we finally estimate an affine transformation with a simple Local-Optimization step [13]. Given the point correspondences between the two images, the algorithm consists of the following steps:

Fig. 1. The inliers found between two images.

- Randomly select a tentative correspondence from the set of the ones not selected so far. This is a correspondence between two circular regions found with the SURF detector $C \leftrightarrow C'$.
- Based on $C \leftrightarrow C'$ and on the transformations H_1, H_2 which transforms C, C' correspondingly to the unit circle the overall similarity transformation is $H = H_2^{-1} H_1$.
- Using H calculate the symmetric transfer error E_i for each correspondence and find the set I for which $E_i < \theta$. This is the set of inliers.
- If $|I|$ is the highest so far then use the Local-Optimization, solve for affine transformation and use it to recalculate I.

A similarity value proportional to the number of inliers is assigned to each pair of images having more inliers than a predefined threshold (empirically set to 10). In the end, for each such pair of images, we insert a weighted edge in the visual similarity graph.

3.2. Tag similarity graph creation

The creation of the tag similarity graph is based on the co-occurrences of tags in the context of images. In principle, it is possible to employ alternative tag-based similarity measures; for instance, by representing images as vectors in the tag vector space (the well-known BoW approach as described in subsection 3.1), it is possible to use the cosine similarity measure. More sophisticated approaches, e.g. Latent Semantic Indexing are also applicable. However, using the tag co-occurrence is considerably more efficient when taking into consideration update requirements (e.g. when a new image is tagged it is much simpler to compute its tag similarity with other images based on co-occurrence than based on cosine similarity).

We process the image-tag associations to build an inverted index, which maintains for each tag a list of images that are annotated with it. Each possible pair of images in this list leads to the creation of an undirected edge between these two images on the image graph. The edge is weighted by the number of times these two images are found together in a tag list. Tags that are associated with very long image lists (i.e. used very frequently to tag images[1]) are not considered in this process. In that way, we avoid to insert spurious/obvious edges in the tag similarity graph. Moreover, this leads to considerable computational gains, since the number of all possible pairs in a list of length n (that we avoid to consider) is $\frac{n \cdot (n-1)}{2} \propto n^2$.

After the creation of the tag similarity graph, we filter out edges with co-occurrence frequency below a certain threshold (empirically

[1] We select the top 5% of tags ranked by frequency.

selected, commonly set to 2 or 3). Such a filtering step aims at removing associations among images that are not common and in addition makes the problem of graph clustering easier from a computational perspective (since the resulting graph is sparser).

3.3. Graph-based clustering

Once the visual and tag image graphs are created, we combine them by into a hybrid image similarity graph by forming the union of the two graphs. We then perform the graph-based clustering of the images by use of community detection [2], i.e. by identifying regions on the graph with high connection density. We have experimented with two community detection methods, the SCAN algorithm [14] and a refinement of it [15], which entails a cluster expansion step by maximizing a local cluster quality measure. The basic community detection step is based on the concept of structural similarity between nodes. The structural similarity σ between nodes v and w is defined as:

$$\sigma(v,w) = \frac{|\Gamma(v) \cap \Gamma(w)|}{\sqrt{|\Gamma(v)| \cdot |\Gamma(w)|}} \quad (1)$$

where $\Gamma(v)$ is the *structure* of node v, i.e. the set comprising the neighbors of the node and the node itself as elements.

Communities are then defined as groups of μ nodes that have a structural similarity value of at least ϵ between each other ((μ,ϵ)-cores [14]). Therefore, the proposed community detection scheme relies on two parameters (μ and ϵ). These parameters have an intuitive meaning: μ is related to the minimum size (in nodes) of the communities that will be discovered and ϵ determines the "relatedness" that nodes of the community are expected to have to each other. In general, increasing μ will result in fewer and larger communities, while increasing ϵ makes the clustering scheme more selective (i.e. flags many nodes as outliers).

The above community detection process is augmented by a community expansion step based on optimizing a local measure of cluster quality (ratio of cluster in-degree over out-degree). In [15] we found that such a step adds relevant members to existing communities, increasing the recall performance of the resulting clustering.

4. EXPERIMENTS

We conducted our experiments on a set of 128,714 geotagged images located within the metropolitan area of Barcelona. Five clustering schemes were tested: (a) three based on SCAN [14]; the first being applied on a visual-only similarity graph (SCAN-VIS), the second on a tag-only similarity graph (SCAN-TAG) and the third on the combined graph (SCAN-HYB), (b) two methods, namely EXP-VIS and EXP-TAG, were based on deriving expanded clusters [15] from the visual-only and tag-only similarity graphs respectively. We employed two complementary evaluation techniques: (a) an indirect, yet objective, criterion, based on the geographical position of the elements of each cluster, and (b) a subjective one, based on the relevance of the cluster results as perceived by users.

4.1. Geospatial cluster coherence

Since we have relatively accurate geo-location information for many of the images of our dataset, it is possible to estimate the mutual proximity of the members of each image cluster produced by our method. The more related the images of the cluster are to each other, the more geographically close to each other they will be; therefore, the average geodesic distance between members of each image cluster and the cluster center serves as an indirect, yet objective, measure of cluster quality. Table 1 presents the results of the five clustering variants we employed. In general, we note that all clustering schemes produce geospatially coherent clusters (spanning regions of a radius in the order of 100-400 meters). The visual-only clusters (SCAN-VIS and EXP-VIS) are more tightly localized. However, having image clusters spanning a larger geographical area is not necessarily a mishap. For instance, we noted that a cluster containing images of a city neighborhood spanned an area of 2km which was to be expected since the pictures were taken from different spots around the neighborhood. In addition to this quantitative cluster evaluation, we attempted to match several of the clusters with city landmarks (Figure 2). In many cases this was possible, which indicates that our method could be used within a landmark detection framework.

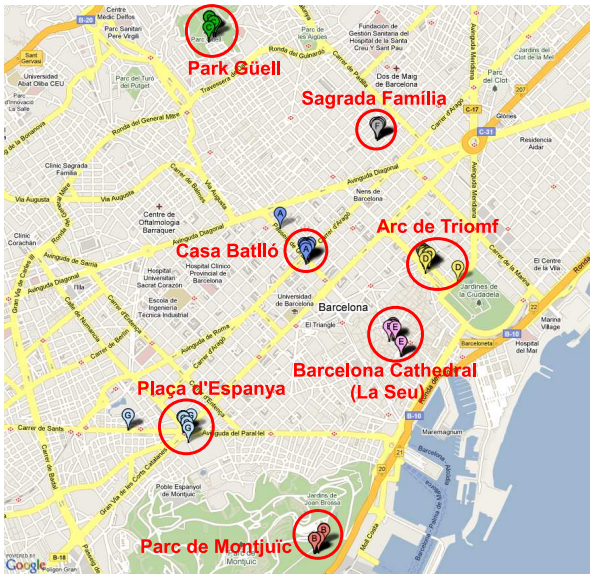

Fig. 2. City landmarks identified as image clusters by our scheme.

Table 1. Geospatial performance of clustering scheme variants. The third and fourth columns represent intra-cluster geodesic distance mean and standard deviation (in meters).

Algorithm	# communities	μ_d (m)	σ_d (m)
SCAN-VIS	1189	132	377
SCAN-TAG	945	419	689
SCAN-HYB	1265	341	613
EXP-VIS	1189	127	363
EXP-TAG	945	441	711

4.2. User study

A study involving 20 users was conducted on a subset of the derived image clusters in order to assess the perceived relevance of the produced clusters. We clustered images by use of the aforementioned five clustering schemes. We found 20 clusters with sufficient overlap among the five methods (so as to correspond to the same concept). We call each set of five such matching clusters a *cluster pool*. Then, we created a scheduling scheme such that each cluster pool would be assessed by two users. In that way, we could compute precision and recall for each cluster pool (recall was computed with respect to

Table 2. IR performance of different clustering schemes

Algorithm	Precision	Recall	F-measure	κ-statistic
SCAN-VIS	0.980	0.178	0.301	**0.925**
SCAN-TAG	0.910	0.197	0.323	0.688
SCAN-HYB	0.898	**0.246**	**0.387**	0.637
EXP-VIS	**0.985**	0.178	0.301	0.895
EXP-TAG	0.929	0.201	0.331	0.709

the total number of relevant images of the whole cluster pool).

According to the user study results (Table 2), all image clusterings are characterized by very high precision scores ($\geq 90\%$). Visual-only clusterings are characterized by superior precision ($\approx 98\%$), but suffer from low recall. Tag-only clusterings behave in an IR-complementary way, yielding higher recall rates at lower precision. The hybrid scheme strikes a balance between these two by offering the highest possible recall at a small precision penalty, achieving the highest mean F-measure of all methods (Figure 3). Thus, in terms of retrieval quality, the hybrid approach enhances the amount of retrieved results by incorporating information from the accompanying image tags, whereas it only slightly affects the accuracy of the visual-only results. In addition, the estimated κ statistics for all methods are significant (> 0.60) revealing substantial inter-annotator agreement, with the highest κ values achieved by visual clustering schemes. The latter is expected since in most cases visual clusters contained images of substantial visual similarity. We further confirmed the superior performance of hybrid clusters by inspecting several tens of clusters (apart from the ones used in the study) produced by the competing schemes. However, we also observed cases where the tag-only and hybrid clusters contained images that were irrelevant to the cluster topic. The most prominent reasons for such failed cases were imprecise tagging information and tag polysemy.

5. CONCLUSIONS

We presented an efficient graph-based clustering scheme for efficient tagged photo retrieval. The scheme uses both visual similarity features and tag co-occurrence information in order to extract enhanced image clusters from large photo collections at a modest computational and memory cost. We demonstrated the accuracy of the derived clusters by means of a two-layer evaluation making use of both objective and subjective cluster quality measures.

6. REFERENCES

[1] D. Cai, X. He, Z. Li, W.Y. Ma, and J.R. Wen, "Hierarchical clustering of www image search results using visual, textual and link information," in *MULTIMEDIA '04*, New York, NY, USA, 2004, pp. 952–959, ACM.

[2] Santo Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3-5, pp. 75–174, 2010.

[3] Y. Chen, J.Z. Wang, and R. Krovetz, "Content-based image retrieval by clustering," in *MIR '03*, New York, NY, USA, 2003, pp. 193–200, ACM.

[4] S. M. Koh and L.-T. Chia, "Web image clustering with reduced keywords and weighted bipartite spectral graph partitioning," in *PCM*. 2006, vol. 4261 of *LNCS*, pp. 880–889, Springer.

[5] B. Gao, T.Y. Liu, T. Qin, X. Zheng, Q.S. Cheng, and W.Y. Ma, "Web image clustering by consistent utilization of visual features and surrounding texts," in *MULTIMEDIA '05*, NY, USA, 2005, pp. 112–121, ACM.

[6] P.A. Moëllic, J.E. Haugeard, and G. Pitel, "Image clustering based on a shared nearest neighbors approach for tagged collections," in *CIVR '08*, NY, USA, 2008, pp. 269–278, ACM.

[7] T. Quack, B. Leibe, and L. Van Gool, "World-scale mining of objects and events from community photo collections," in *CIVR '08*, New York, NY, USA, 2008, pp. 47–56, ACM.

[8] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Comput. Vis. Image Underst.*, vol. 110, no. 3, pp. 346–359, 2008.

[9] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *Int. J. of Computer Vision*, vol. 77, no. 1, pp. 259–289, 2008.

[10] B. Fulkerson, A. Vedaldi, and S. Soatto, "Localizing objects with smart dictionaries," in *ECCV 2008*. Springer.

[11] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *ICCV*, 2003, vol. 2, pp. 1470–1477.

[12] J. Philbin, O. Chum, M. M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *CVPR*. IEEE Comp. Society, 2007, vol. 1, pp. 47–56.

[13] O. Chum, J. Matas, and J. Kittler, "Locally optimized ransac," *LNCS*, pp. 236–243, 2003.

[14] X. Xu, N. Yuruk, Z. Feng, and T.A.J. Schweiger, "Scan: a structural clustering algorithm for networks," in *KDD '07*, New York, NY, USA, 2007, pp. 824–833, ACM.

[15] S. Papadopoulos, Y. Kompatsiaris, and A. Vakali, "Leveraging collective intelligence through community detection in tag networks," in *CKCaR Workshop*, 2009.

(a) SCAN-VIS

(b) SCAN-TAG

(c) SCAN-HYB

Fig. 3. Sample image clusters.