COLLABORATIVE IMAGE ANNOTATION USING IMAGE WEBS

Zixuan Wang[1,*], Omprakash Gnawali[2], Kyle Heath[1], and Leonidas J. Guibas[2]

[1]Department of Electrical Engineering, Stanford University, Stanford, CA, 94305
[2]Department of Computer Science, Stanford University, Stanford, CA, 94305

ABSTRACT

The widespread availability of hand-held devices equipped with cameras has facilitated the creation of massive image collections. Our method links together image regions containing instances of the same object to form a graph called an Image Web. Such graphs represent relationships between images based on shared visual content. We demonstrate how to use Image Webs as conduits for symbolic information propagation among images. Symbolic information include annotations provided by users who perhaps have special expertise or were close to where the sensor data was captured. Such annotations can then be propagated to related images of the same object and benefit other users. Our algorithm gives similarity weights to edges in the Image Web graph. These weights are used to attenuate the relevance of annotations as they propagate along edges of the graph. Experiments show that our system supports multiple users to share images and annotate images collaboratively fast and accurately.

1. INTRODUCTION

The widespread availability of hand-held devices equipped with cameras has facilitated the creation of massive image collections. These massive data sets have remained a largely untapped resource of information because of the difficulty of automatically discovering useful structure in the image content. We build upon the work in [Heath *et al.* , 2010] which proposed building graphs called Image Webs to represent relationship between images in a collection induced by shared objects. Each node in an Image Web corresponds to a region in an image that can be matched to a region in another image. Edges in the Image Web connect regions in different images that are similar under an affine transform. These regions are extracted using a process called Affine Co-segmentation to produce match links between images. The graph also has edges connecting distinct regions that occur in the same image. We present a system that leverages this graph structure to enable users to shares semantic information about objects in large collections of images. Fig. 1 shows images are connected from a front to a back view through intermediate ones.

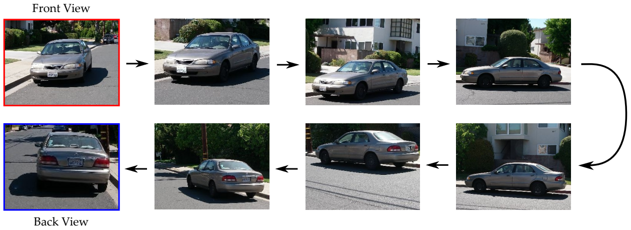


Fig. 1: An example of Image Webs

Such a collaborative annotation system can be extremely useful in the military context. When a soldier takes images of incidents and objects of interest, the soldier can be automatically notified of all the relevant information about the objects in such images. The notification can include information about the object provided by the experts or another soldier who saw the object earlier during a patrol. We automate the inference of relation between objects, even across the images, and propagate the relevant information about the objects using those relations between the objects.

The focus of this paper is to utilize the densely linked image collections to propagate symbolic information such as annotations. This work has three main contributions. First, Image Webs are built incrementally as users add new images to the server. This is different from [Heath *et al.* , 2010], which builds Image Webs over a large scale image collection offline. Second, we use Image Webs as information conduits, allowing symbolic information such as image annotation to flow over them. To our knowledge, no previous literature used image region graph to transfer annotation or other symbolic information. This is different from [Liu *et al.* , 2009], which uses image based graph to annotate images rather than regions. It fails when one image contains distinct objects such that annotating one object would affect other other objects that occur in the same image. The way to define the weights between image regions is also novel. Our algorithm gives similarity weights to edges in the Image Web graph: both appearance similarity and geometric distance between two regions are considered when defining the weight of the corresponding edge. Regions that correspond exactly across the images and regions that have similar appearance and close to each other in the same image have higher weights. These weights are used to attenuate the relevance of annotations as they propagate along edges of the graph. Third, we present a centralized system that support multiple mobile users to share and annotate images collaboratively. Mobile users within a community can share images and annotations through Image Webs. They can query unknown image and the tags of the query image are returned to users in almost real time.

This paper is organized as follows: related works are discussed in section 2. Algorithms to compute Image Webs and propagate tags are shown in section 3. System implementation is in section 4. Evaluation results are in section 5. In the last section, we discuss future work.

2. RELATED WORK

The recent breakthroughs in local feature detector and descriptors [Lowe, 2004] [Bay *et al.* , 2006] [Matas *et al.* , 2004] [Mikolajczyk & Schmid, 2004] make accurate image matching possible. A number of researchers use local features for image retrieval and image mining. The approach of [Chum & Matas, 2010] establishes pairwise matches between images using min-hash [Chum *et al.* , 2007], and then uses query expansion to find clusters of similar images. Due to the low recall ratio of the min-hash, the connectivity of the graph is low, which is not suitable for symbolic information propagation. In [Simon *et al.* , 2007] [Philbin, 2010], clustering is used to create an image graph and to select a set of canonical images summarizing the scene. There are also many useful applications based on the graph structure. In [Agarwal *et al.* , 2009], the image graph is created by computing the fundamental matrix between pair of images. The 3D point cloud is reconstructed from Internet images using structure from motion (SfM). In [Zheng *et al.* , 2009], the graph is built on image regions consisting of groups of matching features. This structure is used to find and label photo of landmarks in a large scale. In [Jing & Baluja, 2008], the PageRank algorithm is used in the image graph built from images returned by Google Product Search to re-rank and group similar products together.

Our pairwise matching algorithm uses local feature matching techniques [Lowe, 2004] [Mikolajczyk & Schmid, 2004]. Co-segmentation algorithms [Rother *et al.* , 2006] [Mukherjee *et al.* , 2009] are used in the context of simultaneously segmenting a person or object of interest from an image pair using Markov Random Field but their speed are too slow to be used in a large collection of images. In [Barnes *et al.* , 2009], dense correspondences are computed using randomized algorithm, but dense local feature extraction is expensive.

The image annotation has been explored by many researchers. In [Wang *et al.* , 2008], content-based image retrieval is used to retrieve similar images from a large scale image collection. Then a keyword search technique is used to obtain a ranked list of candidate annotations for each retrieved image. A fusion algorithm is used to combine the ranked lists into a final candidate annotation list. The candidate annotations are re-ranked using Random Walk with Restarts. In [Cusano *et al.* , 2004] [Li & Wang, 2003], statistical models are used to annotate images. In [Liu *et al.* , 2009], the authors use Nearest Spanning Chain (NSC) to measure the similarity between a pair of images and use it to propagate image annotations.

3. APPROACH

Our algorithm consists of two parts – the Image Web construction algorithm and the annotation propagation algorithm.

3.1 Construction of Image Webs

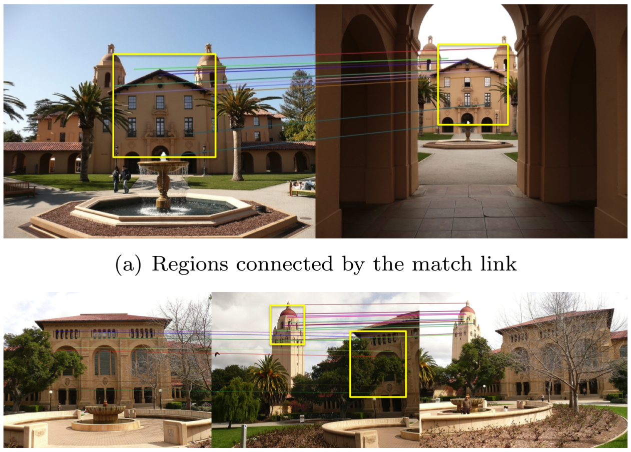
In the construction of Image Webs, we follow the first phase in [Heath *et al.* , 2010] to discover connected components by using content-based image retrieval techniques [Philbin *et al.* , 2007]. The difference is the construction step in [Heath *et al.* , 2010] is offline, but we need to build the Image Webs in real time when users upload new images to the server. In order to measure the similarity between a pair of regions, we use the matched key-points and descriptors inside each region. We briefly list the major steps of building Image Webs and show how our work is different from the previous work.

1. We use I to denote the current set of images uploaded by users and I_q is being uploaded. When a new image I_q is to be added to the current Image Webs, content-based image retrieval techniques [Philbin *et al.* , 2007] are used to select top k ranked images from I: $I_{q1}, I_{q2}, \ldots, I_{qk}$.

2. From each related image, the affine consistent matches are computed using the RANdom Sample Consensus algorithm (RANSAC) [Fischler & Bolles, 1981].

3. Match links and same image links are added to the graph. Match links connect regions which result from local feature matching and same image links which connect regions co-occurring in the same image.

Because when computing initial correspondences, non-salient features are dropped (fail in Lowe's ratio test), to recover more correspondences from these non-salient features, all features in the first image I_i are re-projected to the second image I_j using the computed affine transform. Assume p_1 in I_i is projected to $p_1' = A \cdot p_1$ in I_j, where A is the affine transform between I_i and I_j. In the 8 neighborhood of p_1', the best match p_2 in I_j is detected, which is closest to p_1 in the feature space. Different from [Heath *et al.* , 2010], we use the key-points and their descriptors to describe regions rather than shapes of co-segmentation regions. When a pair of images has geometrically consistent features matches, a match link is added between the matching regions. Same image links are added to represent the connections of regions that co-occur in the same image. These two types of links in the Image Webs, which are explored as follows when defining weights to them. Fig. 2 shows these two types of links in Image Webs.



(a) Regions connected by the match link

(b) Regions connected by the same image link

Fig. 2: Match and same image links in Image Webs

Fig. 3 shows a small part of Image Webs built from different views of the Gates building at Stanford. The user draws a bounding box and tag it and the tags are propagated through the Image Web.

3.2 Definition of weights between regions

To propagate tags through Image Webs, we define the similarity between each pair of connected regions. We define the weights of match links and same image links separately. We set the weights of links between $[0, 1]$. The higher weight reflects that two regions are more similar and tags should propagate from one region to the other. The weight between region i and region j is represented as W_{ij}.

3.2.1 Match links

Because the match links are verified by RANSAC, regions connected by the match link belong to the same object with high probability. If edge (i, j) is a match link that connects corresponding image pair, the algorithm assigns weight $W_{ij} = 1$. This means the tags should propagate through match links.

3.2.2 Same image links

To avoid tags drifting to incorrect regions, we first check whether two regions overlap. Each region consists of several features and a convex hull of these features is computed. If two convex hulls do not overlap, $W_{ij} = 0$. Fig. 4 shows an example in this case. If two convex hulls overlap, we consider the visual appearance and the degree of overlap of two regions to define the similarity.

We use an auxiliary visual vocabulary of size 1000 and represent each region using bag of words model [Sivic & Zisserman, 2003]. Each feature in the region is quantized to its nearest visual word and the region X is represented as a sparse vector $\mathbf{v}$. The appearance similarity of two regions X_i, X_j is measured by the angle spanned by two sparse vectors $\mathbf{v_i}, \mathbf{v_j}$.

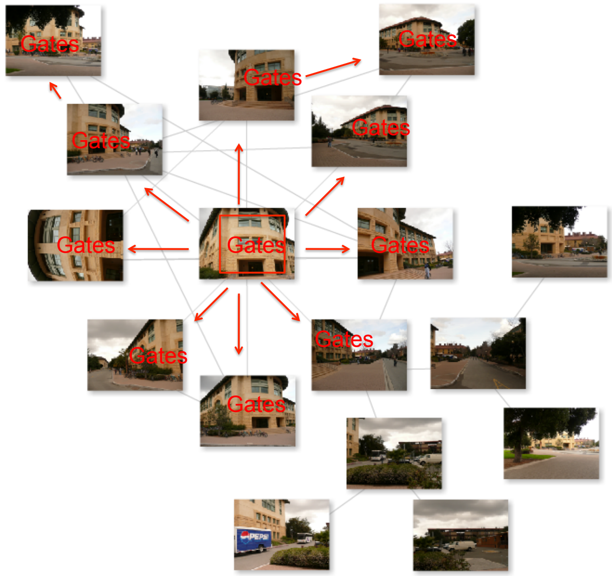


Fig. 3: An example of tag propagation

$$s_{app}(X_i, X_j) = \frac{\mathbf{v_i} \cdot \mathbf{v_j}}{|\mathbf{v_i}| \cdot |\mathbf{v_j}|} \tag{1}$$

Hausdorff distance is used to measure the geometric distance of two image regions

$$\begin{aligned} d_{geo}(X_i, X_j) = \frac{1}{\beta} \max\{ &\sup_{x_1 \in X_i} \inf_{x_2 \in X_j} d(x_1, x_2), \\ &\sup_{x_2 \in X_j} \inf_{x_1 \in X_i} d(x_1, x_2)\} \end{aligned} \tag{2}$$

where $d(x_1, x_2)$ is the Euclidean distance between the key-points x_1 and x_2. β is the normalize factor to make the geometric distance invariant to the image size.

$$\beta = \sigma_x(X_i) + \sigma_y(X_i) + \sigma_x(X_j) + \sigma_y(X_j) \tag{3}$$

$\sigma_x(X_i)$ is the standard deviation of X_i in x axis and $\sigma_y(X_i)$ is the standard deviation of X_i in y axis.

When the Hausdorff distance is small, the overlap of two regions is large.

The weight between X_i and X_j is defined as:

$$W_{ij} = s_{app} \cdot e^{-d_{geo}/2} \tag{4}$$

This means that regions that correspond exactly (discovered by computer vision techniques) and regions that have similar appearance and have significant overlap are highly related.

3.3 Propagation of tags

After having the graph of image regions, we use it to propagate image annotations provided by users. Given a set of N connected image regions $X_1, X_2, \ldots, X_N$ and c labels, we have an $N \times c$ labeling matrix Y with entries between $[0, 1]$ to indicate the probability that the region having that label. N is the total number of regions in the same connected component in Image Webs. We set the initial entries of Y from the user's manual inputs, that the user draws a bounding box on the image and give it a tag. The number of features inside the bounding box divided by the total number of features in the region is defined as the initial value for that entry.

We used an approach based on the graph-learning algorithm proposed in [Liu *et al.* , 2009], which has four steps:

1. The $N \times N$ similarity matrix W is computed using the algorithm in the previous section.

2. Symmetrically normalize W by computing
$$S = D^{-1/2} W D^{-1/2} \tag{5}$$
where D is a diagonal matrix, $D_{ii} = \sum_{j=1}^{N} W_{ij}$.

3. Do the following iteration until convergence
$$R_{t+1} = \alpha S R_t + (1 - \alpha) Y \tag{6}$$
where α is the propagation parameter (we choose $\alpha = 0.25$). To start, initialize $R_0 = Y$. The intuition is that the similarity matrix S propagates tags to related items while the label matrix Y provides expert information which are from the input of users.

4. In the final step, the algorithm generates tags for each image by merging the list of tags assigned to each region in the image. This is done by selecting the tags with the probability greater than τ/N, where τ is a threshold (we choose $\tau = 0.2$). We sort selected tags by their probabilities and present a list of tags to the user.

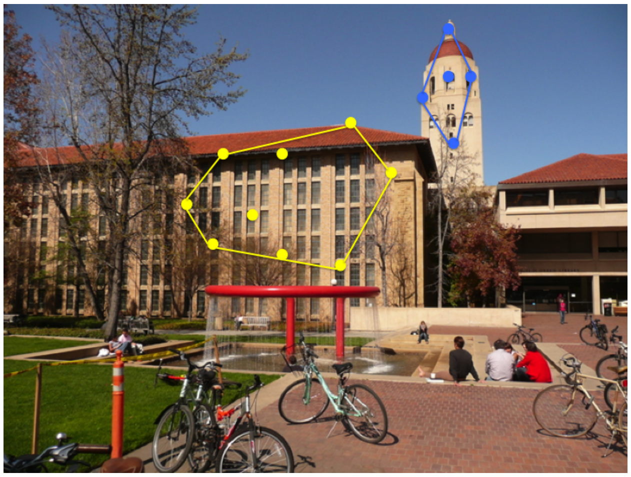


Fig. 4: Two regions do not overlap

4. IMPLEMENTATION

We implemented the collaborative annotation system on the Android 2.1 platform using the client-server architecture shown in Fig. 5. We were able to get two users to take pictures using their Android phones, tag specific objects in the images, and automatically share those annotations with the other user. The mobile client supports two actions: *Tag* and *Query*.

1. In the *Tag* action, the user draws bounding boxes on the image and give them labels. The down-sampled image with resolution 640 × 480 and the tag information including the information of the bounding boxes and other meta data such as GPS location are sent to the server. On the server side, content-based image retrieval techniques are used to select top 25 ranked images and verifies the geometric consistency. The new image is connected to images that pass the geometric verification and its tags are propagated as described in section 3.

   We keep two types of tags. One is manual tag and the other is automatic tag. Each time when new image is added to Image Webs, only manual tags are propagated in the current connected component.

2. In the *Query* action, the user can either send the down-sampled query image or a sparse vector representing the query image to server. In the evaluation section, we will compare two methods in terms of the speed and network traffic. When the down-sampled query image is sent to the server, the query image is placed in Image Webs in the same way as that in the *Tag* action and labels are pulled from it neighbors. The tags with the probability that are greater than a threshold are returned to the user and are overlayed on the corresponding

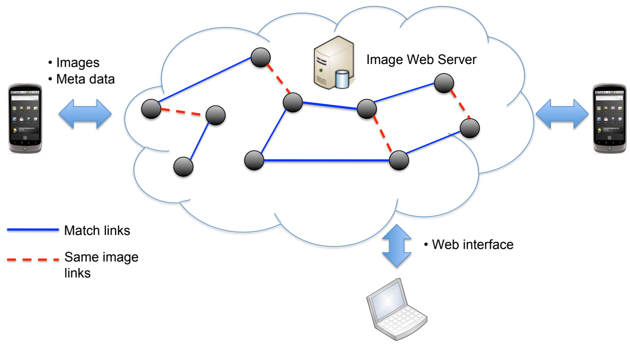


Fig. 5: The architecture that two Image Webs are connected to propagate the annotations

objects. If the sparse vector is sent, content-based image retrieval techniques are used to find the most similar image. The tags of the retrieved image are returned to the client.

We also provide a central Image Web browser web site, from which we can discover when images are inserted to the Image Web and who is tagging the objects. This gives users a global view of the structure of the Image Webs.

5. EVALUATION

We use 1 million visual vocabulary trained from 5k Oxford building dataset using approximate k-means [Philbin *et al.* , 2007]. We test our algorithm using Stanford building dataset which contains 569 images. We test it using Intel Xeon 2.26GHz PC. We also dispatch the same task to a 500 core cluster and the construction time is less than 1 minute. The dataset contains 20 different buildings on Stanford campus. The resolution of each image is 640 × 480. Table 1 shows the statistics of the construction of Image Webs. Initially the dataset has no tag information. 40 images are randomly selected from the dataset and are annotated manually by drawing a bounding box and giving it a tag. Each building is tagged at least once. Table 2 shows the results after the tag propagation.

The Oxford building dataset contains 5063 images taken from Oxford, which contains 11 different buildings. 22 images are randomly selected from the dataset and are annotated. Each building is

Table 1: Statistics of Image Webs

	Stanford	Oxford
Images	569	5063
Regions	3284	14966
Edges	13117	161420
Construction Time	2958s	25634s

Table 2: Tag propagation results

	Stanford	Oxford
Bounding boxes	40	22
Annotated images	432	681
Accuracy	84%	97%

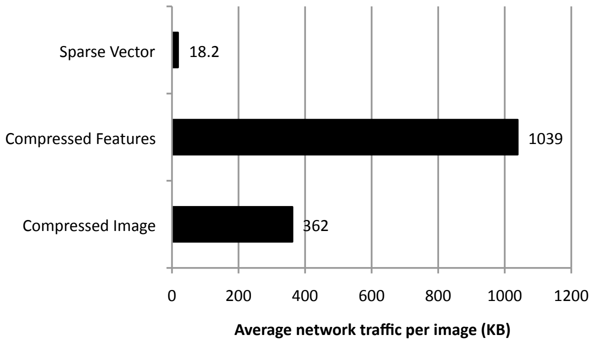


Fig. 6: Network traffic to query one image

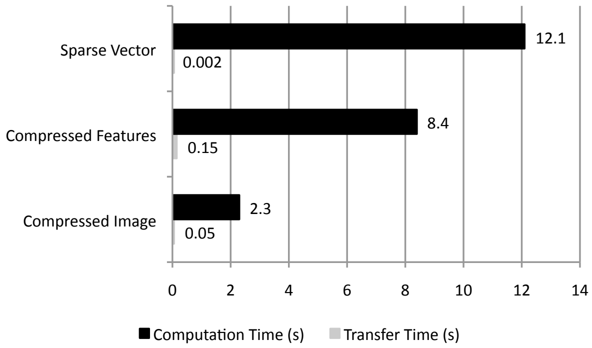


Fig. 7: Response time to query one image

tagged twice. Table 1 shows the construction information of the Image Webs. Table 2 shows the statistics of the result.

To compare the performance in querying the image label, Fig. 6 shows the average network traffic to query one image. To extract local features and represent the query image in a sparse vector saves most network traffic, which is very useful when network resource is expensive. Fig. 7 shows the average response time to query one image. Although computing the sparse vector representation of an image on mobile devices can save network bandwidth, feature extraction and quantization is still very expensive on mobile devices. This causes the total response time to be much greater than if the down-sampled image were simply sent to the server.

6. FUTURE WORK

We use the central server to build Image Webs in the current implementation. If we can extract features on the phone efficiently, we can build Image Webs on each phone in a distributed way.

We also plan to extend collaborative annotation to a large scale dataset and support more users. We also plan to design algorithms to resolve the annotation conflicts that can arise when different labels are equally likely to describe the object correctly. Such conflict resolution will require inference of semantic information about the tags aided by ontology databases.

CONCLUSION

In summary, this paper introduces a new algorithm to tag the objects in an image and automatically share those tags between multiple mobile users. We link images through similar image regions, and use different types of links to propagate symbolic information. Experiments show that our system allows multiple users to share image annotations fast and accurately.

ACKNOWLEDGMENTS

We are grateful for support from NSF POMI Expedition and AHPCRC.

References

[Agarwal *et al.* , 2009] Agarwal, S., Snavely, N., Simon, I., Seitz, S.M., & Szeliski, R. 2009. Building rome in a day. *In: Proceedings of the international conference on computer vision.*

[Barnes *et al.* , 2009] Barnes, C., Shechtman, E., Finkelstein, A., & Goldman, D.B. 2009. Patchmatch: A randomized correspondence algorithm for structural image editing. *Acm transactions on graphics*, **28**(3), 2.

[Bay *et al.* , 2006] Bay, H., Tuytelaars, T., & Van Gool, L. 2006. Surf: Speeded up robust features. *Eccv*, 404–417.

[Chum & Matas, 2010] Chum, O., & Matas, J. 2010. Large Scale Discovery of Spatially Related Images. *Ieee transactions on pattern analysis and machine intelligence.*

[Chum *et al.* , 2007] Chum, O., Philbin, J., Isard, M., & Zisserman, A. 2007. Scalable near identical image and shot detection. *Page 556 of: Proceedings of the 6th acm international conference on image and video retrieval.* ACM.

[Cusano *et al.* , 2004] Cusano, C., Ciocca, G., & Schettini, R. 2004. Image annotation using SVM. *Pages 330–338 of: Proceedings of internet imaging iv, vol. spie*, vol. 5304. Citeseer.

[Fischler & Bolles, 1981] Fischler, M.A., & Bolles, R.C. 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the acm*, **24**(6), 381–395.

[Heath *et al.* , 2010] Heath, K., Gelfand, N., Ovsjanikov, M., Aanjaneya, M., & Guibas, L.J. 2010. Image webs: Computing and exploiting connectivity in image collections. *In: Proceedings of the ieee conference on computer vision and pattern recognition.*

[Jing & Baluja, 2008] Jing, Y., & Baluja, S. 2008. Pagerank for product image search. *Pages 307–316 of: Proceeding of the 17th international conference on world wide web.* ACM.

[Li & Wang, 2003] Li, J., & Wang, J.Z. 2003. Automatic linguistic indexing of pictures by a statistical modeling approach. *Ieee transactions on pattern analysis and machine intelligence*, 1075–1088.

[Liu *et al.* , 2009] Liu, J., Li, M., Liu, Q., Lu, H., & Ma, S. 2009. Image annotation via graph learning. *Pattern recognition*, **42**(2), 218–228.

[Lowe, 2004] Lowe, D.G. 2004. Distinctive image features from scale-invariant keypoints. *Pages 91–110 of: International journal of computer vision*, vol. 60. Springer.

[Matas *et al.* , 2004] Matas, J., Chum, O., Urban, M., & Pajdla, T. 2004. Robust wide-baseline stereo from maximally stable extremal regions. *Image and vision computing*, **22**(10), 761–767.

[Mikolajczyk & Schmid, 2004] Mikolajczyk, K., & Schmid, C. 2004. Scale & affine invariant interest point detectors. *International journal of computer vision*, **60**(1), 63–86.

[Mukherjee *et al.* , 2009] Mukherjee, L., Singh, V., & Dyer, C.R. 2009. Half-integrality based algorithms for cosegmentation of images. *In: Proceedings of the ieee conference on computer vision and pattern recognition.*

[Philbin, 2010] Philbin, J. 2010. *Scalable object retrieval in very large image collections.* Ph.D. thesis, University of Oxford.

[Philbin *et al.* , 2007] Philbin, J., Chum, O., Isard, M., Sivic, J., & Zisserman, A. 2007. Object retrieval with large vocabularies and fast spatial matching. *In: Proceedings of the ieee conference on computer vision and pattern recognition.*

[Rother *et al.* , 2006] Rother, C., Minka, T., Blake, A., & Kolmogorov, V. 2006. Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. *In: Proceedings of the ieee conference on computer vision and pattern recognition.*

[Simon *et al.* , 2007] Simon, I., Snavely, N., & Seitz, S.M. 2007. Scene summarization for online image collections. *Pages 1–8 of: Proceedings of the international conference on computer vision.* Citeseer.

[Sivic & Zisserman, 2003] Sivic, J., & Zisserman, A. 2003. Video Google: A text retrieval approach to object matching in videos. *In: Proceedings of the international conference on computer vision.*

[Wang *et al.* , 2008] Wang, C., Jing, F., Zhang, L., & Zhang, H.J. 2008. Scalable search-based image annotation. *Multimedia systems*, **14**(4), 205–220.

[Zheng *et al.* , 2009] Zheng, Y.T., Zhao, M., Song, Y., Adam, H., Buddemeier, U., Bissacco, A., Brucher, F., Chua, T.S., & Neven, H. 2009. Tour the world: building a web-scale landmark recognition engine. *In: Proceedings of the ieee conference on computer vision and pattern recognition.*