Cutout-Search: Putting a Name to the Picture

Dhruv Batra[1] batradhruv@cmu.edu
Adarsh Kowdle[2] apk64@cornell.edu
Devi Parikh[1] dparikh@cmu.edu
Tsuhan Chen[2,1] tsuhan@ece.cornell.edu

[1]Carnegie Mellon University [2]Cornell University

Abstract

We often come across photographs with content whose identity we can no longer recall. For instance, we may have a picture from a football game we went to, but do not remember the name of the team in the photograph. A natural instinct may be to query an image search engine with related general terms, such as 'football' or 'football teams' in this case. This would lead to many irrelevant retrievals, and the user would have to manually examine several pages of retrieval results before he can hope to find other images containing the same team players and look at the text associated with these images to identify the team. With the growing popularity of global image matching techniques, one may consider matching the query image to other images on the web. However, this does not allow for ways to focus on the object-of-interest while matching, and may cause the background to overwhelm the matching results, especially when the object-of-interest is small and can occur in varying backgrounds, again, leading to irrelevant retrievals.

We propose Cutout-Search*, where a user employs an interactive segmentation tool to cut out the object-of-interest from the image, and use this* Cutout-Query *to retrieve images. As our experiments show, this leads to retrieval of more relevant images when compared to global image matching leading to more specific identification of the object-of-interest in the query image.*

1. Introduction

How often do we find ourselves staring at pictures from old photo-collections, unable to recall the names of the teams playing in the football match we went to? Or the name of the famous landmark pictured in the backdrop? Or the scientific name of a bird or butterfly we photographed?

In the era where google has become a verb, the natural solution would be to use a vague (but related) query term (*e.g.*, 'football teams', 'famous landmarks', or 'bird species') with an image search engine (like Google, Live or Flickr), and hope that we see an image similar to ours in the first few pages of the retrieval, and perhaps the associated text descriptions (tags, file names, *etc.*) give us clues to helps us answer our real question: What is *this* called? However, since the query term is general and semantically a level higher, it is unlikely to find relevant images without requiring us to manually examine a prohibitively large number of images. For example, Figure 2 shows the top few images returned by Live Image Search for the query 'soccer player'. We can imagine the number of images we would have to examine to find one similar to the query image, with a query term this vague.

Figure 1: "Which team was playing in the football game we went to?", "Which landmark is that in the backdrop?", "Which bird species is this?" We often come across pictures on the web or in our personal photo collections, unable to identify the content.

This is because this is precisely the scenario where text-tag based search fails. Text-tag based search assumes that users have a text description available, and seek a visual description for this text. But what if we don't *know* what we are looking for? The scenario we are interested in is where we seek a text description for a specific visual entity.

Recent works in internet vision and image retrieval [11, 12] have shown success at scene-matching, and this would seem to be a promising direction for finding images similar to query image that the user has. However, in many cases, the region of interest to users is small in comparison with the image size, and global image-level matching would be overwhelmed by the overall *scene* of the image, potentially de-emphasizing the very object the user was interested in. This is especially troublesome in cases where the object might appear in front of vastly varied backgrounds (*e.g.*

Figure 2: Putting a Name to a Picture: We consider a scenario where a user is unable to recall the name of the team in his photograph. The first block of images shows actual retrieval results from Live Image Search using a vague (but related) query text 'soccer player'. We can imagine that it would be a hopeless exercise to try and find an image containing the same team, so as to look at the associated text to identify the team. The second block of images are actual results returned by the 'Show Similar Images' feature in Live Image Search . We can see that even though most images match at a scene-level, they do not contain the same team/foreground. We propose *Cutout-Search* where a user first uses an interactive segmentation tool to create a cutout of the object-of-interest, thus guiding the search towards more relevant images, as seen the third block. The associated text of these relevant results can be analyzed to precisely identify the object, as seen in the histograms.

animals in natural outdoor surroundings, and structured indoors). Figure 2 shows an example retrieval when using the 'show similar images' feature in Live Image Search.

Interestingly, this problem of ambiguity in determining the region of interest to the user has been faced for a long time in the field of image segmentation. In this paper, we

borrow from the successful ideas of interactive segmentation [5, 20], and propose the notion of *Cutout-Queries*. We propose an interface where users can scribble on images to create object cutouts, and then use these as queries for the dataset. This Cutout-query now holds visual information about the object-of-interest alone, and thus can be expected to provide more relevant retrievals, as seen in Figure 2. Text analysis on these relevant retrievals would allow for more accurate and specific identification of the query object.

At a higher level, this notion of Cutout-queries can be generalized to *Cutout-Tags*. Just like current photo-sharing websites (like Flickr) allow users to tag their uploaded images with text keywords, which are later used to search these images, we envision an interface where users can also scribble on images after uploading them to create cutouts for various objects in the image. These cutouts are then treated as visual tags for this image, much in the same way keywords are treated as text tags. Figure 2 shows an example image that is tagged with both text-tags and a few possible Cutout-tags. As demonstrated in our experiments, our proposed idea of using Cutout-queries does not require Cutout-tags for matching. However the notion of *Cutout-Search* opens doors to a new medium of interaction for internet-users and forms for interesting research questions.

The rest of this paper is organized as follows. Section 2 describes some related work. We describe our proposed approach in Section 3, followed by a description of the experimental set up and results in Sections 4 and 5. We conclude our paper after some discussion in Sections 6 and 7.

2. Related Work

Boykov and Jolly [5] posed interactive segmentation as a discrete optimization problem solved by graph-cut energy minimization. Rother *et al*. [20] extend this technique to a simpler user interaction in the form of bounding boxes. Any of these standard approaches can be used to obtain the cutouts in our proposed framework.

Several works have leveraged the associations between visual and textual data on the web. Berg *et al*. [3] build a large dataset of labelled faces using news photographs and corresponding captions. Berg and Forsyth [4] use textual and visual information to re-rank results from Google image search and build classifiers for several animal categories. Satoh and Kanade [23] associate names and faces in video using a transcription.

Object instance recognition has been applied to the task of retrieving relevant information from the web for a user capturing a picture of an object-of-interest [17] such as CD-cases. These methods are applicable in scenarios where the user can actively photograph a known object they wish to know more about, such as information about the music album or band. In contrast, our work deals with the scenario where the user already has a photograph in hand. Another related work is that of Sivic *et al*. [24], who use ideas from text retrieval to perform efficient object retrieval in video. While this work contains interesting ideas in the retrieval framework (which could be adopted in our matching framework), their end goal is retrieval, whereas we also analyze text associated with returned results to identify the query.

Several works [19, 25] have explored the use of automatic segmentation to extract the object-of-interest to guide the retrieval process. For a detailed survey please refer to Datta *et al*. [10].

3. Approach

Our basic approach is as follows. Given an object in an image a user wishes to identify, the user scribbles on this query image to cut out the object-of-interest. This cutout is used to perform segment matching from images in our database, and the top few relevant images are retrieved. The work closest to our retrieval setup is that of "Blobworld" [8], where an initial segmentation is performed on the database images, and then segment matching is used for retrieval. The key difference is that in our work, we use object cutout to achieve the segmentation of the query and the multiple segmentation approach of Russell *et al*. [22] to be robust to segmentation errors on the database images. Finally, the text associated with the retrieved images is analyzed and a few keywords are displayed as suggestions for identifying the cutout.

Interactive segmentation. In order to extract the cutout from the query image, any existing interactive segmentation tools [5, 20] can be used. We cast our binary labelling problem as an energy minimization problem solved via graph cuts. We work with an over-segmentation of the image. The task is to label each superpixel as foreground or background. We construct a graph over these superpixels, where adjacent superpixels are joined by an edge. Associated with this graph is an energy which is a weighted combination of a data-term and an edge-term. We model the data-term as the negative log-likelihood of the features extracted at a superpixel given the class model. Our features are mean Luv colour features extracted over superpixels, and the class model is a Gaussian Mixture Model (GMM). The edge-term is modeled as a contrast sensitive Potts model using the learnt distances proposed by Batra *et al*. [1]. Finally, we use Graph-cuts to efficiently compute the MAP labels for all superpixels, using the implementation provided by Boykov *et al*. [6, 7, 14]. Example cutouts obtained using this algorithm can be seen in Figure 3.

Multiple segmentations. We use the multiple segmentation approach of Russell *et al*. [22] to create a large pool of segments from images in our dataset so that segment match-

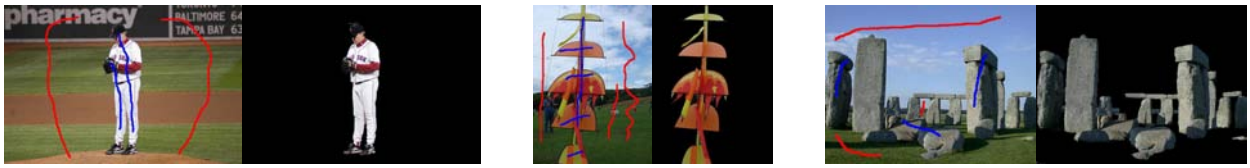


Figure 3: Example results of the interactive segmentation algorithm used. In each pair of images, the left image indicates the scribbles provided by the user, and the right shows the cutout obtained.

ing with the cutout query can be performed. Similar to Malisiewicz *et al*. [15] we use normalized cuts [9] to obtain a total of 9 different segmentations per image by varying the number of segments 6, 9 and 12, and the image scale to be 100%, 50% and 25%. We also compute similar multiple segmentations of the Cutout-query.

Features. We describe each superpixel with color, texture and shape features. We use the color and texture features of Hoiem *et al*. [13] and shape features similar to Malisiewicz *et al*. [16].

Relevance scoring scheme. In order to determine the relevant images to the provided Cutout-query, we compute a relevance score for each image in the dataset. Intuitively, an image that contains many segments that match the query segments should be assigned a high score. We compute the relevance score for an image i as follows:

$$\phi_i = \sum_f \sum_{s_q} \sum_{s_i} \exp\left(\frac{-d_{qi}^f}{m_q^f}\right) \quad (1)$$

where f indicates all the features (color, texture, shape), s_i refers to all the segments in the set of multiple segmentations corresponding to image i, s_q refers to all the segments corresponding to the Cutout-query, d_{qi}^f refers to the squared Eucledian distance between each query segment and a segment in image i using feature f and m_q^f is the average of the squared distance d_{qi}^f over all s_i using feature f.

Text analysis. We analyze the text associated with the most relevant images to find potential words to identify the Cutout-query. Several text analysis tools can be used to analyze text-tags associated with retrieved images [3,4], or descriptions or captions associated with images. In our implementation we simply analyze the filenames of the images. We collect all the file names of the top relevant images, and for each pair of names, find the longest common substring. We build a histogram of all such substrings to find how often a substring was found to be the longest common substring in all pairs of relevant images. Although quite simplistic, this method was found to work well in all our experimental scenarios, and serves as a proof of concept.

4. Experiments

In our experiments, we present several scenarios where a user attempts to identify what a particular object-of-interest in an image is. We compare our proposed Cutout-search, to global-matching based image search. Using a vague but related text query provided poor results, and is not reported here.

4.1. Dataset

We downloaded several image collections using Live Image Search. For each image collection, we combined images downloaded using a generic query a user is likely to give, with a more specific query that corresponds to the specific entity the user is interested in, but does not know and is attempting to identify.

For one collection, we download 200 images using the query 'soccer', and 100 images each for 'soccer argentina', 'soccer astonvilla', 'soccer brazil', 'soccer chelsea' and 'soccer newcastle'. For another collection, we downloaded 200 images using the query 'butterfly' and 100 each for 'butterfly california sister', 'butterfly monarch' and 'butterfly owl'. For a third collection, we combined 200 images for 'fish' with 100 each for 'fish kissing gourami', 'fish yellow cichlid'; and for a fourth collection we downloaded 200 images of 'flower' and 100 images for 'flower sunflower'.

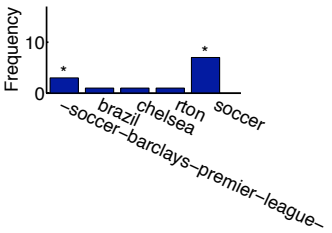
4.2. Baseline: global matching

We compare with the baseline approach of global image matching, where the query image is matched to all images in the dataset globally, to retrieve a few closest matches. As with Cutout-search, the text associated with these retrieved images is analyzed to identify the object in the query image. For global matching we use the color histogram of the image in the Lab space following [12] and gist features [18] that describes the global spatial layout of texture in the image. The matching score of an image i is computed as

$$\phi_i^g = \sum_f \exp\left(\frac{-d_{qi}^f}{m_q^f}\right) \quad (2)$$

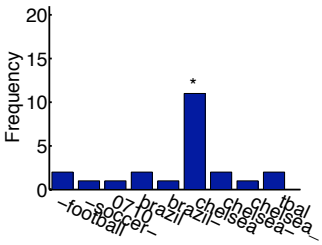
where d_{qi}^f is the squared eucledian distance of the query image to image i using feature f, and m_q^f is the average of the squared distance d_{qi}^f over all images i using feature f.

(a) Global-matching based retrieval



(b) Global-matching based text histogram

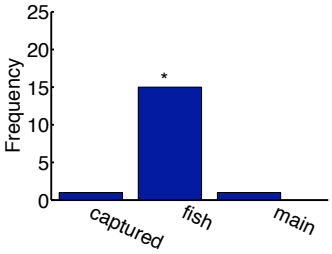
(c) Cutout-search retrieval



(d) Cutout-search text histogram

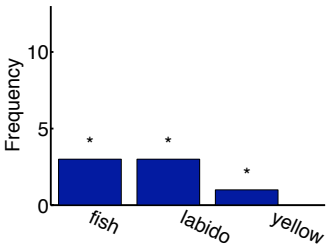
Figure 4: Dataset: Soccer, Query: 'Chelsea', a team: Global-image based matching finds the tags 'soccer' and '-soccer-barclays-premier-league'. The second tag is a reflection of a bias in our dataset as several of the teams are part of the English Premier League. Our proposed method can successfully identify the team 'chelsea'.

(a) Global-matching based retrieval



(b) Global-matching based text histogram

(c) Cutout-search retrieval



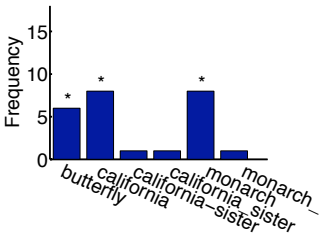
(d) Cutout-search text histogram

Figure 5: Dataset: Fish, Query: 'Yellow Cichlid', a type of fish. We can see that global-matching based retrieval returns the generic tag, 'fish'. Our proposed method not only find the tags 'yellow', 'fish' but also 'labido'. Interestingly the scientific name of this fish is 'Labidochromus Caerleus', which was not part of the query text used to download the dataset.

5. Results

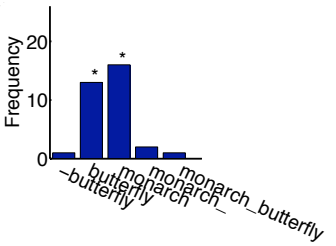
We now show results for queries on all four datasets: soccer, fish, butterfly and flower. For all our results, the top left image is the query image or cutout. We display 13 most relevant images. We show the histogram of the top 10 substrings found when analyzing the text associated to these images (strings with fewer than 3 characters were ignored), and the recommended text words for the input query image/cutout are marked with asterisk '*'. In all cases (Figures 4 to 8), we see that the global image matching is unable to identify a specific textual description of the query im-

(a) Global-matching based retrieval



(b) Global-matching based text histogram

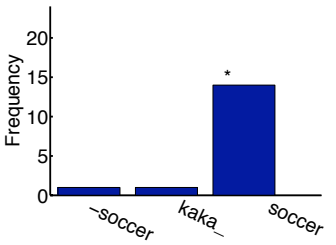
(c) Cutout-search retrieval



(d) Cutout-search text histogram

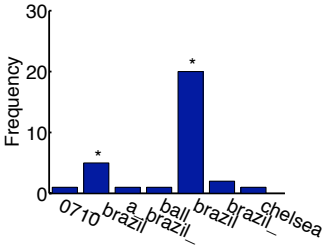
Figure 6: Dataset: Butterfly, Query: 'Monarch', a type of butterfly. Note that global-matching based retrieval is confused between two different kinds of butterflies, 'Monarch' and 'California Sister', while the proposed method is able to correctly identify the butterfly type.

(a) Global-matching based retrieval



(b) Global-matching based text histogram

(c) Cutout-search retrieval



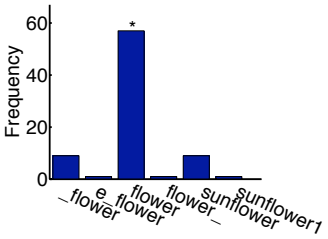
(d) Cutout-search text histogram

Figure 7: Dataset: Soccer, Query: 'Brazil', the team: Global-image based matching only finds the tag 'soccer' while our proposed method can also identify the specific team 'brazil'.

age, while Cutout-search can do so. On average, across the 5 scenarios we evaluated, we found that global-matching based search retrieved 3.2 relevant images in the top 13, while Cutout-search more than doubled that to 7.4.

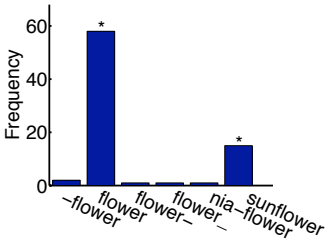
Soccer dataset. For the soccer dataset, we use two query images containing a player from the team 'Chelsea' and 'Brazil' that the user is hoping to identify. The results obtained can be seen in Figures 4 and 7. In both cases we see that the global-matching based retrieval returns generic soccer images and only can provide a very vague description to the user. In Figure 4 we can see that the global matching returns images that are similar to the query image at the scene level, most of them containing a football player at similar scales in the foreground, with a blured/smoooth background. However, the specific object-of-interest is not reliably retrieved. Cutout-search on the other hand can return images specific to the teams, and for both queries, iden-

(a) Global-matching based retrieval



(b) Global-matching based text histogram

(c) Cutout-search retrieval



(d) Cutout-search text histogram

Figure 8: Dataset: Flower, Query: 'Sunflower', a type of flower: Global-image based matching only finds the tag 'flower' while our proposed method can also identify the specific category 'sunflower'.

tify for the user the team photographed in the image by focussing the matching on the cutout. In Figure 4, we see that images containing the foreground object are retrieved, even though they have drastically varying global image layouts.

Fish dataset. For the fish dataset, we select a query image containing a 'Yellow Cichlid' fish that the user wishes to identify. The results obtained are shown in Figure 5. We see that the global-matching based search can not provide the user with any more information than the generic tag 'fish'. Cutout-search on the other hand is not only able to recommend the tag 'yellow', but also 'labido', which is interestingly part of the scientific name of this fish 'Labidochromus Caerleus'. It should be noted that this was not part of the query text used to collect the dataset, and thus is an interesting discovery.

Butterfly dataset. For the butterfly dataset, we consider a scenario where the user has a picture of the 'Monarch' butterfly that he wishes to identify. The results obtained can be seen in Figures 6. In the global-matching based results we see that two different types of butterflies, 'Monarch' and 'California Sister', have high peaks, thus providing the user with confusing information. Cutout-search on the other hand can clearly identify the specific type of butterfly, 'Monarch'. The second peak is 'butterfly', which is more general, but also relevant.

Flower dataset. For the flower dataset, we consider a photograph of 'sunflower' as the query image. The results

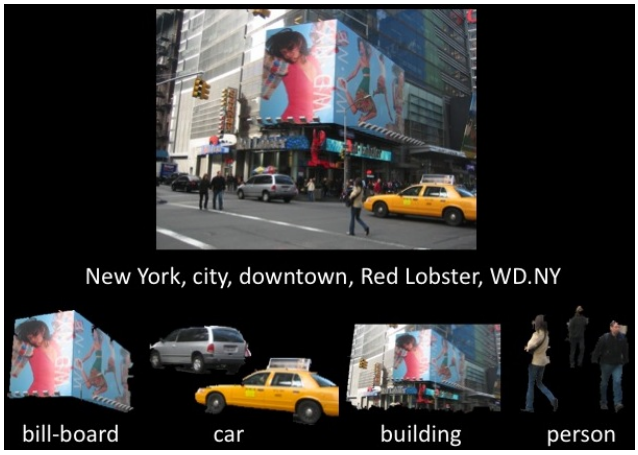


Figure 9: Our envisioned scenario where users upload pictures with Cutout-tags (in addition to image-level and perhaps cutout-level text-tags), effectively providing us with scene parsing through interactions.

shown in Figure 8 similarly demonstrate the benefit of using Cutout-search over global-matching.

6. Discussion

One may argue that if instead of using hundreds of images, the dataset consisted of millions of images, even global matching would provide relevant images to our query image. However, as we show in Figure 2, a search engine with access to millions of images on the web did not retrieve relevant images. Moreover, using cutouts allows for signif-

icantly more candidate matches and far fewer false negatives, since the background has now been explicitly suppressed, and the matching focuses only on the foreground object-of-interest. This could allow for fast and approximate matching techniques, without hurting the precision of the retrieval.

It should be noted that while we demonstrate the use of Cutout-search for the specific purpose of identifying unknown image content, it can also be used to refine regular image search where a user is trying to gather more information even of a known object. Cutout-queries and tags simply allow the user specify which parts of the image are of interest, allowing for a more guided-search.

We can see that this process of creating Cutout-tags can be thought of as *interactive* scene-parsing [2], where the user breaks down the scene into visually meaningful regions by scribble based annotations. This may alleviate the need for multiple segmentations on the dataset. This is related to efforts in collecting a large dataset with online collaborative labelling such as LabelMe [21].

7. Conclusion

In this paper we propose the notion of *Cutout-Search*. We motivate this with a scenario where a user does not know the identity of a certain object in the image. In our experimental results we show how using global image matching to retrieve relevant images, and using the text associated with these images to infer the identity of the object-of-interest in the query image performs poorly. Instead, if we use a cutout of the object-of-interest using an interactive image segmentation tool, and use this *Cutout-Query* to match segments in the database, the identity of the object-of-interest can be determined significantly more reliably. This could potentially lead to a new mode of interaction in the internet photo-sharing world, where in addition to text tags, users provide *Cutout-Tags* associated with the representative objects in their uploaded photos. Future work involves large scale experiments to test the Cutout-search notion, and explore further research directions involving partial matching algorithms, fast (perhaps approximate) search procedures.

References

[1] D. Batra, R. Sukthankar, and T. Chen. Semi-supervised clustering via learnt codeword distances. In *BMVC*, 2008.

[2] A. Berg, F. Grabler, and J. Malik. Parsing images of architectural scenes. In *CVPR*, 2007.

[3] T. L. Berg, A. C. Berg, J. Edwards, M. Maire, R. White, Y.-W. Teh, E. Learned-Miller, and D. A. Forsyth. Names and faces in the news. In *CVPR*, 2004.

[4] T. L. Berg and D. A. Forsyth. Animals on the web. In *CVPR*, 2006.

[5] Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. *ICCV*, 2001.

[6] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *PAMI*, 26(9):1124–1137, 2004.

[7] Y. Boykov, O. Veksler, and R. Zabih. Efficient approximate energy minimization via graph cuts. *PAMI*, 20(12):1222–1239, 2001.

[8] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: image segmentation using expectation-maximization and its application to image querying. *PAMI*, 24(8):1026–1038, 2002.

[9] T. Cour, F. Benezit, and J. Shi. Spectral segmentation with multiscale graph decomposition. In *CVPR*, 2005.

[10] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 2008.

[11] J. Hays and A. A. Efros. Scene completion using millions of photographs. *SIGGRAPH*, 2007.

[12] J. Hays and A. A. Efros. im2gps: estimating geographic information from a single image. In *CVPR*, 2008.

[13] D. Hoiem, A. A. Efros, and M. Hebert. Recovering surface layout from an image. *IJCV*, 75(1), 2007.

[14] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *PAMI*, 26(2):147–159, 2004.

[15] T. Malisiewicz and A. A. Efros. Improving spatial support for objects via multiple segmentations. In *BMVC*, 2007.

[16] T. Malisiewicz and A. A. Efros. Recognition by association via learning per-exemplar distances. In *CVPR*, 2008.

[17] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006.

[18] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.

[19] Rajashekhara and S. Chaudhuri. Segmentation and region of interest based image retrieval in low depth of field observations. *Image Vision Computing*, 2007.

[20] C. Rother, V. Kolmogorov, and A. Blake. "Grabcut": interactive foreground extraction using iterated graph cuts. *SIGGRAPH*, 2004.

[21] B. Russell, A. Torralba, K. Murphy, and W. Freeman. Labelme: a database and web-based tool for image annotation. *MIT, AI Memo*, 2005.

[22] B. C. Russell, A. A. Efros, J. Sivic, W. T. Freeman, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *CVPR*, 2006.

[23] S. Satoh and T. Kanade. Name-it: Association of face and name in video. In *CVPR*, 1997.

[24] J. Sivic and A. Zisserman. Efficient visual search for objects in videos. *Proceedings of the IEEE*, 96(4):548–566, 2008.

[25] K. Vu, K. A. Hua, and W. Tavanapong. Image retrieval based on regions of interest. *KDE*, 2003.