Object Constellations: Scalable, Simultaneous Detection and Recognition of Multiple Specific Objects

Ser-Nam Lim[1] Gianfranco Doretto[2⋆] Jens Rittscher[1]

Visualization & Computer Vision Lab[1]
GE Global Research
Niskayuna, NY, USA 12309

Dept of CS & EE[2]
West Virginia University
Morgantown, WV, USA 26506

Abstract. Given a library of objects specified by example images, we describe a probabilistic framework for detecting multiple such objects in the scene, as well as estimating their positions and sizes. Detection of such an *object constellation* is faced with several major challenges. The approach has to be scalable to a large number of objects in the library while being robust towards outliers and noise. We propose to overcome these challenges by generating object and geometry hypotheses as priors for estimating the constellation. Generating object hypotheses avoids the need for evaluating the presence of all the objects in the image, which allows us to achieve scalability. Generating geometry hypotheses localizes the region of interests (ROIs) in which detection can be conducted. Object constellations are then estimated through a feature matching procedure, where we make recognition decision based on the number of feature matches for each object hypothesis, as well as the quality of the matches. We demonstrate the efficacy of our approach with both public datasets and TV broadcast image sequences.

1 Introduction

Given a library of objects, each of which is specified by a set of reference images, we seek to determine which of those are present in a given image. In addition to that, we also aim to estimate their location and apparent size. We do not pose restrictions on the number of items, and refer to a configuration of objects, their locations and sizes, as an *object constellation* (see Figure 1). Here, we present a framework that allows the estimation of the constellation of known objects in a given image. No specific assumptions about the reference images that describe a specific object are being made, except that they should contain only that object, or a portion of it. The requirement that our algorithm should scale with the number of objects in the library is one of the guiding principles of our approach. Ultimately, our approach should facilitate a real-time processing of large-scale image data sets or video sequences.

⋆ author was with GE Global Research when this work was performed

The described capability will enable multiple applications, one of which is the automatic annotation of large image and video archives. In addition to detecting items that belong to generic object categories, such as vehicles or buildings, it is the ability to detect specific objects, or landmarks, that will enable a semantic annotation aimed at narrowing a search down to the finest details. For instance, the scene does in many cases provide the only clue where a certain recorded event took place. We envisage that a library containing images of objects such as important buildings, monuments, and items such as logos or flags might effectively annotate visual data to support the search for a specific event. Relative size and position between objects in the scene is another important annotation enrichment, aimed at supporting search refinements. For instance, if an event was simultaneously filmed from different viewpoints, this annotation would support search queries of the same scene but with objects in flipped positions.

Focus of this paper is the formulation of a probabilistic framework that enables the robust estimation of a constellation of objects in a given image (Section 2). We deal with the problem of estimating the number of objects by first proposing a set of *object hypotheses* in a scalable way, where every object has to be proposed while it appears with other objects, maybe partially occluded, and in background clutter (Section 3.1). We also propose a set of potential positions and sizes of these objects, which we call *geometry hypothesis*. We do so with a combined multiscale object search and pruning procedure (Section 3.2). The exact objects composing the constellation are estimated by a robust procedure (Section 4), where image features, extracted from the input image, vote on each object hypothesis they think they belong to, based on how their appearances agree with the object appearance, and how their positions agree with the object position. A fully integrated system, tested in many scenarios (Section 5), demonstrates that the proposed framework provides an effective approach to the object constellation estimation problem.

1.1 Related Work

Our approach to estimating object constellations using a library of example images builds upon recent research in image retrieval and object recognition. The development of robust feature detectors [1], such as SIFT [2], provided the basis to match and register images using a very sparse set of sample points. Although not scalable to a large amount of objects, an extension of that initial line of work allows to perform object detection in clutter, after an appropriate training phase [3]. Using feature correspondences and computing similarity measures based on feature matches has become a standard approach to object recognition [4–6]. Spatial configurations of features and object parts [7, 8] have led to more robust object recognition strategies, while [9] proposes generating saliency regions in which features are then extracted for matching purpose. While such regions are analogous to our hypothesis generation we guide the search for ROIs based on recognizable object candidates, which improves robustness.

In the image retrieval domain, k-means clustering methods can be utilized to construct visual codebooks [10, 11] by grouping critical features from a represen-

Fig. 1. Object constellations. Given the input image on the left, and a library describing a set of specific objects, we would like to tell which of those are in the image, and where and how big they are. We call this information an *object constellation*.

tative set of objects. Effective indexing methods like the fast tree matching [12] and the vocabulary tree [13, 14] demonstrate that the feature based approach can be applied to large-scale image retrieval and search.

Recent work has highlighted the importance of exploiting contextual information for recognition purposes (see [15] and references therein). [16] is an example that uses it for scene segmentation and labeling. More related to our work, [17] exploits geotagged images to build a multimodal unsupervised framework that links images to Wikipedia articles, whereas [18] exploits multimodal information to automatically detect, and label landmarks in large collection of images, but they do not consider multiple objects, and their relative spatial position. Finally, recent work has considered the problem of reorganizing, and browsing large collections of images of a given site, while taking into account 3D geometry constraints [19, 20].

2 Problem Overview

Let us suppose that knowledge about the shape and appearance of a very large set of objects $\mathcal{O}$ is available. Given an image I, we would like to solve the problem of being able to tell *which of these objects are present* in the image, as well as *where they are*, and *how big they are*. If $O \in \mathcal{O}$ indicates an object, c its position, and s its size in the image, we refer to a set $C_N \doteq \{(O_i, c_i, s_i)\}_{i=1,\cdots,N}$ of N objects, together with their positions and sizes, as an *object constellation*. Therefore, given I, we would like to estimate the constellation it contains.

If we knew that I contained exactly N objects, we would have to solve a reduced problem, which could be framed as the maximum a posteriori (MAP) estimation $\hat{C}_N \doteq \arg\max p(C_N|I)$. Since N is unknown, the original problem includes a model selection step. One approach would be to solve the MAP es-

timation for several N's, and then pick the solution that optimizes a suitable information theoretic criterion [21].

Rather than computing multiple MAP estimates, and defining and computing costly information theoretic measures, we approach the original problem in two stages. First, we nominate M objects, together with their positions and sizes, which are the most likely to be present in the image I, where we assume that $M \geq N$. This is like computing the MAP estimate of the constellation C_M. Second, given I and C_M, we estimate the biggest constellation (ideally $C_N \subset C_M$) that is in accordance with the nominated hypotheses. This step simultaneously takes into account single object hypothesis appearance and shape, as well as relative object positions and orientations. It can be interpreted as a robust estimation step where undesired object hypothesis (outliers) are rejected, and correct ones are refined, which means that this step also includes an implicit form of occlusion reasoning. Section 3 addresses the first stage, and Section 4 addresses the second.

3 Constellation Hypothesis

Assuming that the constellation depicted in the image I has $N \leq M$ objects, we are interested in a fast and scalable estimation of a constellation C_M, made of hypothesized objects, positions, and scales, which will hopefully be such that $C_N \subset C_M$. In other words we want to solve $\hat{C}_M \doteq \arg\max\, p(C_M|I)$. The conditional probability rules allow to split the problem into two. The first one estimates the hypothesized objects, whereas the second one estimates the hypothesized geometry of the objects. More precisely, we need to compute

$$\hat{O}_1,\cdots,\hat{O}_M \doteq \arg\max_{O_1,\cdots,O_M \in \mathcal{O}} p(O_1,\cdots,O_M|I)\ , \tag{1}$$

$$(\hat{c}_1,\hat{s}_1),\cdots,(\hat{c}_M,\hat{s}_M) \doteq \arg\max_{(c_1,s_1),\cdots,(c_M,s_M)} p((c_1,s_1),\cdots,(c_M,s_M)|I,O_1,\cdots,O_M)\ . \tag{2}$$

Section 3.1 explains how to solve problem (1), and Section 3.2 explains how to solve problem (2).

3.1 Object Hypotheses

Since $\mathcal{O}$ is very large, we need to illustrate how (1) can be solved in a fast and scalable way. We do so by using a variation of the vocabulary tree approach described in [13]. More precisely, we represent an object $O_i \in \mathcal{O}$ by a set of *template* images $\{T_i^u\}$. Every template is a tightly cropped image of the object. The representation can be made robust against viewpoint variation and occlusions by including many templates taken from different viewpoints, and templates that do not include the entire object. Every template is then represented by a set of features $\{g_{p'}|p' \in \mathcal{P}'_{i,u}\}$, where $\mathcal{P}'_{i,u}$ is the set of template points identified by an interest point detector. We use SIFT features [2] because they are scale-invariant and highly discriminative [22].

Similar to [13], the vocabulary tree is built by running hierarchical k-means clustering, where a SIFT feature g, and a cluster center h are compared by using

the following distance

$$\Delta(g,h) = \frac{\sum_{t=1}^{16} B(g(t),h(t))}{16} \ , \tag{3}$$

which is simply the average of the Bhattacharyya distances ($B(\cdot,\cdot)$) between the 16 histograms of the SIFT feature descriptors [2].

Given the image I, we extract the SIFT features $\{f_p | p \in \mathcal{P}\}$, where $\mathcal{P}$ is the set of image points identified by the SIFT detector. At each level of the vocabulary tree the cluster with center nearest to f_p is chosen, and contributes to the composition of the relevance of the objects $O_i \in \mathcal{O}$ to the image I. In particular, we found that the original waiting of the contributions proposed in [13], which was designed for recognizing a single object occupying a significant portion of I, is not very effective in presence of clutter. This issue is fundamental because we have the ambition to detect object constellations, where wanted objects will likely appear together with unwanted ones, besides additional background clutter. Therefore, we propose the following weighting scheme

$$w_{p,m} = \exp^{-\alpha \Delta(f_p, h_m)} \ , \tag{4}$$

which essentially weighs the merit of the nearest cluster on its similarity to the feature each time, scaled by a factor α. This is in contrast with assigning the same weight indiscriminately, based only on how "rare" is a cluster (the reader can consult [13] for more details). In this way features from clutter are expected to contribute poorly to the composition of the object relevance.

Finally, we compute $\hat{O}_1, \cdots, \hat{O}_M$ as the rank-M most relevant objects scored against the image I according to the scoring function defined in [13]. The vocabulary tree with the proposed weighting scheme retains the efficiency of the original algorithm, and is now able to cope with severe clutter. Moreover, the size of the vocabulary tree can be reduced, and does not need to be enormous as it was previously necessary to alleviate problems associated with clutter.

3.2 Geometry Hypotheses

Given the image I and the object hypotheses $O_1, \cdots, O_M$, we describe how we address (2) in this section. We make the assumption that position and size of each object can be estimated independently from the other objects. Let us suppose that object O_i was hypothesized by means of the template T_i^u, for which we compute a predefined holistic appearance descriptor. We proceed by analyzing a detection window, sliding across the image I. At every position we compute the window descriptor and compare it with T_i^u by computing the L_1-norm of their difference. Also, we repeat this computation at multiple scales, i.e. for various sizes of the detection window. The point where we record the minimum discrepancy defines the hypothesized position $\hat{c}_i$ and size $\hat{s}_i$ of the object O_i. Although slow in principle, this procedure can be accelerated because it is fully parallelizable.

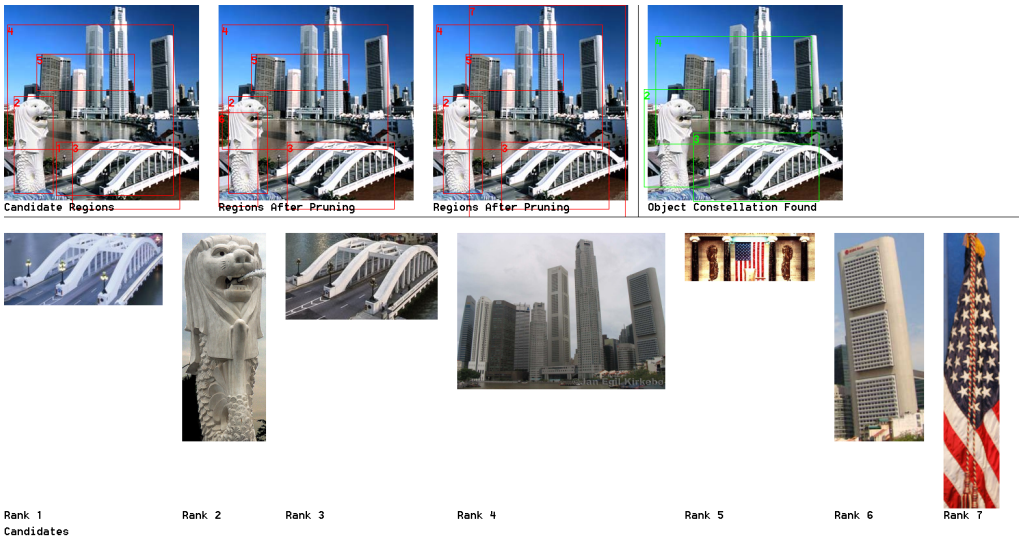


Fig. 2. Object hypothesis pruning. First row: Initial constellation hypothesis, the constellation hypothesis after pruning the rank 1 object, the final constellation hypothesis after pruning the rank 6 object, and the estimated constellation after Hungarian matching. Second row: Ranked list of object hypotheses involved in the pruning. **(Please magnify the PDF to see the details)**

In our approach we found the HOG descriptor [23] to produce excellent results with L_2-Hys block normalization. Each detection window is divided into small spatial regions called cells. We use rectangular cells, and found that varying the cell size and number according to the aspect ratio of the template improves the performance. In practice, this is done by simply imposing a given number of (approximately) square cells along the smallest size of the detection window.

Once the geometry $(\hat{c}_i, \hat{s}_i)$ of every object O_i has been computed, we need to verify whether it is "plausible". More precisely, different templates of the same object O_i might have ranked O_i within the first M positions multiple times. This is true especially when different templates represent portions of the same object. However, if O_i is fully visible in the image I, we are interested in considering only the template including the entire object, and not the ones showing the object parts. Therefore, if $O_1, \cdots, O_n$ are n object hypotheses of the same object O_i, nominated through the templates $\mathcal{T} \doteq \{T_i^{u_1}, \cdots, T_i^{u_n}\}$, we proceed as follows. For each $T \in \mathcal{T}$ we compute the fraction of its area that overlaps with the other templates in $\mathcal{T} \setminus \{T\}$. If the highest overlapping fraction is greater than a given threshold, the corresponding template, and associated object hypothesis are pruned. The object that was ranked $M+1$ now becomes part of the M object hypotheses (see Figure 2). After computing the geometry of the new object, this pruning process is repeated until all the object hypotheses

have a plausible geometry. Note that, if multiple templates have the same highest overlapping fraction, then the one with the smallest area is pruned.

Finally, the geometry hypothesis $(\hat{c}_i, \hat{s}_i)$ defines a mapping ψ_i from the coordinate system of the template to the coordinate system of I, which identifies the hypothesized *region of interest* (ROI) where O_i might be located, assuming that the four corners of the template correspond to the four corners of the ROI.

4 Constellation Estimation

Given the image I and the constellation hypothesis C_M, we are interested in estimating the constellation C_N, where we downselect to N objects from C_M to C_N, and where we perform a final refinement of the position and size of each of the objects.

To this end, we first conduct bipartite graph matching between the SIFT features belonging to the objects in C_M and those that have been extracted from I. This can be achieved by the Hungarian algorithm, which solves such an assignment problem in polynomial time. More importantly, critical to this matching step is the weighting scheme for the edges between the two sets of SIFT features, which we formulate as follows.

Given I and C_M, at every point $p \in \mathcal{P}$, we would like to estimate the weight variable $W_p \doteq (i, p')$, which depicts the merit of assigning feature f_p to the object O_i, and matching it with the feature $g_{p'}$, $p' \in \mathcal{P}'_i$. More synthetically we determine the weight as

$$W_p \doteq -\left[\left(1 - e^{-\frac{|p-\psi_i(p')|^2}{2\mu^2}}\right) + \nu\Delta(f_p, g_{p'})\right], \tag{5}$$

where the constant μ allows to calibrate the spatial weighting of the first term, and ν allows to calibrate the relative weight between first and second term. The first term quantifies how aligned O_i, and its point p', might be to the image point p. The second term quantifies the similarity between f_p and $g_{p'}$ as given in Eq. 3.

Based on the weights computed, we can obtain from the Hungarian algorithm a set of unique one-to-one matches. However, since the Hungarian algorithm always generates a number of matches that equals the cardinality of the smaller SIFT feature set, it is possible for matches that are too low in weights to "survive" as long as they are ranked high enough. For this reason, we impose a threshold below which an edge is discarded before we conduct Hungarian matching.

Matches generated by the Hungarian matching can be separated into subsets of object-specific matches, whereby each object in C_M "receives" a number of matching features from I. To finally arrive at C_N, we discard those subsets that contain less than a pre-specified number of matches. The remaining subsets are then used to refine the geometry of the recognized objects in I such that a new mapping between the recognized object and the corresponding template is computed solely based on the matched features.

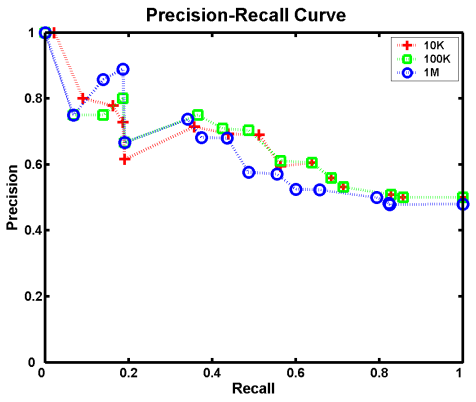


Fig. 3. Precision-recall curves. Precision-recall curves for one run and three vocabulary tree sizes. The performance remains fairly stable across different sizes.

5 Experiments and Results

In this section we are going to describe a series of experiments that highlight various aspects of our approach, which we have implemented in C++.

Object Database. We composed an object database $\mathcal{O}$ made of multiple objects represented by templates coming from various sources. We borrow 320 templates from the dataset in [13], representing 80 objects. We added the 50 templates representing the 10 building landmarks in the Oxford building dataset [24] that were used as query images of an image retrieval system. In addition 100 templates downloaded from internet, representing 33 objects, which are buildings, and other objects or popular landmarks were used. Unless otherwise specified, this is our standard object datadase, totaling 710 templates representing 203 different objects.

Test Dataset. For testing we have collected 10 short (100-500 frames) YouTube videos of TV broadcast news and documentaries, about 200 images of buildings and squares downloaded from internet, and we have added the Oxford image retrieval dataset [24], made of 5602 images of 11 different Oxford landmarks, which here we use as test images.

Weighting Scheme. With three groundtruthed YouTube videos containing a single object constellation per frame in significant clutter, we compare the proposed weighting scheme (4) against the one in [13]. In this case the object database $\mathcal{O}$ contained only 21 objects, and was represented with a vocabulary tree with $k = 6$ and 4 levels. For each sequence, Figure 4 shows the frequency at which the correct object was included within the rank 1 to 5 hypothesis. We observe that the proposed weighting scheme outperforms the former method in two sequences, and yields similar performance in the third one. We highlight also the fact that the scalability of the nomination of the object hypothesis is fully preserved even with the weighting scheme (4), and enables efficient handling of a

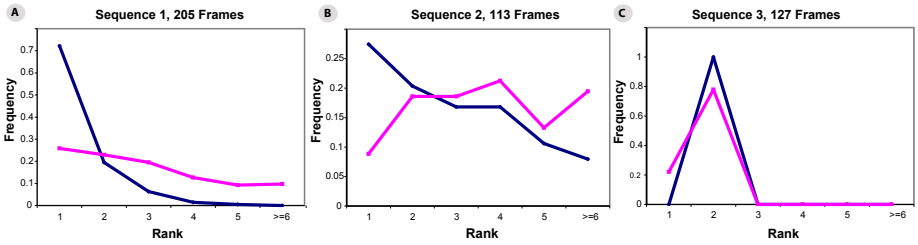
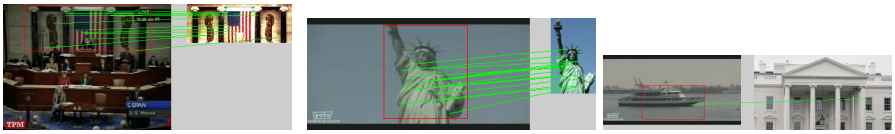


Fig. 4. Weighting scheme. Frequency at which a correct object is hypothesized within a given rank by a vocabulary tree with weighting scheme (4) (solid magenta), and with the weighting scheme in [13] (solid blue).



(a) True positive (b) Partial occlusion (c) True negative

Fig. 5. Single object constellation. Matches between input frames of YouTube videos on the left and the rank 1 object hypothesis on the right. The red bounding box is the ROI identified by the geometry hypothesis.

large database of objects. On the other hand, since the number of object hypothesis M is pre-specified, the complexity for estimating the constellation remains the same, regardless of the size of the object database.

Geometry Hypothesis. For the three YouTube videos, when the correct object was within the rank 5 hypothesis, we also computed the geometry hypothesis. The detection window had 3 cells along the smaller dimension, and the multiscale search was within a fraction of $[0.25, 1]$ of the frame size. We measured the performance by reporting how many times the object with the lowest descriptor discrepancy was not the right one, and this event happened only for one frame, highlighting the robustness of this step.

Precision and Recall. We follow the approach described in [25], and compare our method against object retrieval algorithms [26, 24, 27], on the ground of the mean Average Precision (mAP) metric computed on the Oxford dataset. Note that for this comparison a query image for an image retrieval system becomes a template of the object database, whereas a retrievable image of the retrieval system becomes a test image for our approach.

For each evaluation run, we select several different voting thresholds γ, and randomly select 100 test images. We compute a precision-recall curve, and record the area under the curve, which is the Average Precision (AP). One example

of the curve is shown in Figure 3, which shows that we have performed this experiment with three vocabulary tree settings, more precisely with 10K words ($k = 10$ and 4 levels), 100K words ($k = 10$ and 5 levels), and 1M words ($k = 10$ and 6 levels). Finally, we average the runs to obtain the mAP of 0.624298, 0.647894 and 0.627027 for the three vocabularies, respectively.

Figure 3 highlights that the weighting scheme (4) makes our approach relatively stable with respect to the vocabulary tree size. We reach a peak performance of 0.647894 at 100K words, which is less than the 1M words at which the Approximate k-means Method (AKM) [24] reaches the peak. Also, the peak of AKM, reported to be 0.645, is lower than ours. [24] also reports the peak mAP of the vocabulary tree of [13], which is only 0.439. On the other hand, our approach fails to outperform the soft assignment approach in [27], which has a peak mAP of 0.673 at 1M words. Note, however, that our vocabulary tree could be extended to include the notion of soft assignment, which is the subject of future work.

Single Object Constellation. Figure 5 shows the rank 1 object hypothesis and matches, after Hungarian matching, for three frames of YouTube videos. Figure 5(a) highlights the robustness of the approach against background clutter. Figure 5(b) shows that partial occlusions can be handled as well. Finally, Figure 5(c) shows a true negative example that can easily be rejected by the cutoff threshold.

Object Hypothesis Pruning. Figure 2 illustrates the effects of the object hypothesis pruning described in Section 3.2. Among the templates ranked 1 to 5, 1 and 3 are from the same object. The pruning process eliminates 1, and considers 6 as part of the new constellation hypothesis. 6 is a portion of 4, and is very much occluded, causing it to be pruned, and template 7 becomes part of the final constellation hypothesis. The estimated constellation is made of 3 objects, ranked 2, 3, and 4.

Hungarian Matching. Figure 6 shows, for a true positive object hypothesis and for a true negative object hypothesis, the matches computed by the Hungarian Matching (Section 4). It can be seen that while the true positive retains the correct matches, the matches of the true negative are significantly pruned, and the object hypothesis is rejected.

Robust Constellation Estimation. Figure 7 shows two other examples of constellation estimation. The object hypotheses are not ranked such that the correct object are in the first positions. In the first example they are ranked 1 and 5. In the second example the pruning had to reason up to the ranked 10 objects to produce 5 valid object hypotheses, namely 1, 3, 7, 8, 10, and the right ones are 1 and 3. In both cases the matching stage was able to reject all the unwanted object hypotheses, and refine the geometry of the true positives.

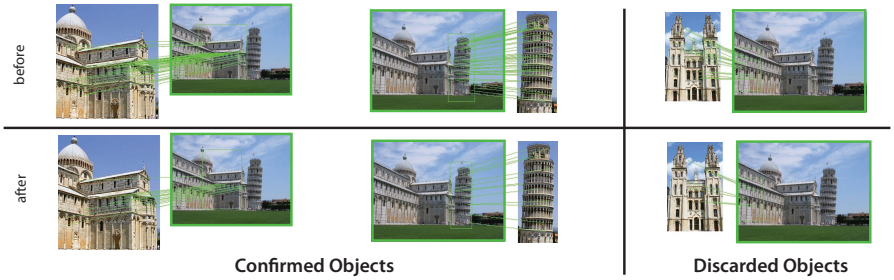


Fig. 6. Matching. Matches before and after rejecting matches that are simply too low in weight (Sec. 4). On the left is those of a true positive object hypothesis, and on the right is those of a true negative object hypothesis.

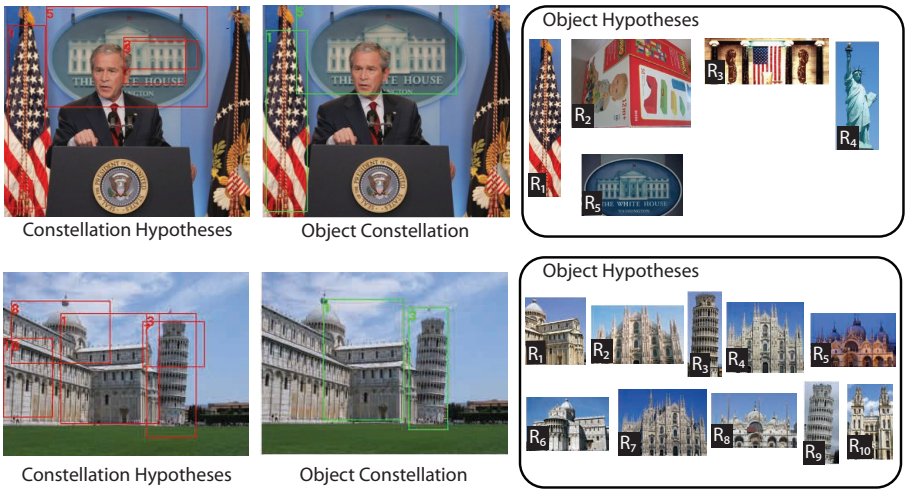


Fig. 7. Object constellation detections. Two examples showing the constellation hypothesis (leftmost images with red ROIs), composed of the object hypotheses (rightmost images in boxes), and the final constellation estimation (middle images with green ROIs).

6 Conclusions

We have proposed an approach for the estimation of object constellations, with objects defined by an image library. The framework addresses the challenges of scalability with respect to the library size, robustness against noise and background clutter, and does not pose limits to the number of simultaneously detectable objects in an image. The experiments portray the robustness and potential for this approach to go towards enabling a richer semantic annotation of

large image and video datasets, describing not only the category of the objects, but also what these objects are, together with their spatial relations and sizes. We believe that this will ultimately enable visual search systems supporting scene-specific queries.

7 Acknowledgements

This report was prepared by GE GRC as an account of work sponsored by Lockheed Martin Corporation. Information contained in this report constitutes technical information which is the property of Lockheed Martin Corporation. Neither GE nor Lockheed Martin Corporation, nor any person acting on behalf of either; a. Makes any warranty or representation, expressed or implied, with respect to the use of any information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or b. Assume any liabilities with respect to the use of, or for damages resulting from the use of, any information, apparatus, method or process disclosed in this report.

References

1. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Transactions on PAMI (2005) 1615–1630
2. Lowe, D.G.: Object recognition from local scale-invariant features. In: ICCV. Volume 2. (1999) 1150–1157
3. Lowe, D.G.: Local feature view clustering for 3D object recognition. In: CVPR. (2001)
4. Berg, A., Berg, T., Malik, J.: Shape matching and object recognition using low distortion correspondences. In: CVPR. Volume 1. (2005) 26–33
5. Moreels, P., Perona, P.: A probabilistic cascade of detectors for individual object recognition. In: ECCV. (2008) 426–439
6. Kannala, J., Rahtu, E., Brandt, S., Heikkila, J.: Object recognition and segmentation by non-rigid quasi-dense matching. In: CVPR. (2008) 1–8
7. Marszaek, M., Schmid, C.: Spatial weighting for bag-of-features. In: CVPR. Volume 2. (2006) 2118–2125
8. Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. IJCV **61** (2005) 55–79
9. Rutishauser, U., Walther, D., Koch, C., Perona, P.: Is bottom-up attention useful for object recognition? In: CVPR. Volume 2. (2004) 37–44
10. Sivic, J., Russell, B.C., Efros, A.A., Zisserman, A., Freeman, W.T.: Discovering objects and their location in images. In: ICCV. Volume 1. (2005) 370–377
11. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: ICCV. Volume 2. (2003) 1470–1477
12. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. Image and Vision Computing **22** (2004) 761–767
13. Nister, D., Stewenius, H.: Scalable recognition with a vocabulary tree. In: CVPR. Volume 2. (2006) 2161–2168

14. Chum, O., Philbin, J., Sivic, J., Isard, M., Zisserman, A.: Total Recall: Automatic query expansion with a generative feature model for object retrieval. In: ICCV. (2007)
15. Divvala, S.K., Hoiem, D., Hays, J.H., Efros, A.A., Hebert, M.: An empirical study of context in object detection. In: CVPR. (2009)
16. Lazebnik, S., Raginsky, M.: An empirical bayes approach to contextual region classi?cation. In: CVPR. (2009)
17. Quack, T., Leibe, B., Van Gool, L.: World-scale mining of objects and events from community photo collections. In: CIVR. (2008)
18. Gammeter, S., Bossard, L., Quack, T., Van Gool, L.: I know what you did last summer: object-level auto-annotation of holiday snaps. In: ICCV. (2009)
19. Snavely, N., Seitz, S.M., Szeliski, R.: Photo Tourism: Exploring photo collections in 3D. ACM Trans. on Graphics (2006) 835–846
20. Li, X., Wu, C., Zach, C., Lazebnik, S., Frahm, J.M.: Modeling and recognition of landmark image collections using iconic scene graphs. In: ECCV. (2008)
21. Claeskens, G., Lid Hjort, N.: Model Selection and Model Averaging. Cambridge University Press (2008)
22. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. PAMI **27** (2005) 1615–1630
23. Dalal, N., Triggs, B.: Histogram of oriented gradients for human detection. In: ICCV. Volume 1. (2005) 886–893
24. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: CVPR. (2007)
25. Agarwal, S., Awan, A., Roth, D.: Learning to detect objects in images via a sparse, part-based representation. PAMI **26** (2004) 1475–1490
26. Chum, O., Philbin, J., Sivic, J., Isard, M., Zisserman, A.: Total recall: Automatic query expansion with a generative feature model for object retrieval. In: ICCV. (2007)
27. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In: CVPR. (2008)