Depiction Invariant Object Matching

Anupriya Balikai
anubalikai@gmail.com
Peter M. Hall
pmh@cs.bath.ac.uk

Department of Computer Science
University of Bath
Bath, UK

Abstract

We are interested in matching objects in photographs, paintings, sketches and so on; after all, humans have a remarkable ability to recognise objects in images, no matter how they are depicted. We conduct experiments in matching, and conclude that the key to robustness lies in object description. The existing literature consists of numerous feature descriptors that rely heavily on photometric properties such as colour and illumination to describe objects. Although these methods achieve high rates of accuracy in applications such as detection and retrieval of photographs, they fail to generalise datasets consisting of mixed depictions. Here, we propose a more general approach for describing objects invariant to depictive style. We use structure at a global level, which is combined with simple non-photometric descriptors at a local level. There is no need for any prior learning. Our descriptor achieves results on par with existing state of the art, when applied to object matching on a standard dataset consisting of photographs alone and outperforms the state of the art when applied to depiction-invariant object matching.

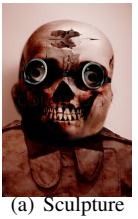
1 Introduction

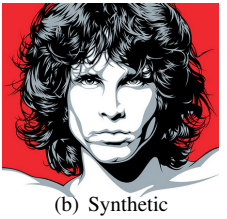
Object matching is an area that has received continuous and consistent attention within Computer Vision. The state of the art is now robust to challenges such as occlusions, viewpoint invariance, pose invariance and so on. However, little attention has been given to matching objects across depictive styles. Yet, as shown in Figure 1, a face is a face is a face, no matter what style of depiction has been chosen to represent it. A majority of current object matching methods are heavily tuned towards recognising photographs, and cannot be easily generalised to handle inputs of varying depiction.

In order to find common ground between instances of an object depicted in varying styles, a highly generic form of representation is required. We provide details below, but briefly we use structure – a graph of nodes and arcs — to describe objects in a global sense. This is because structure encodes the relationship between parts in a manner that is independent of depiction. Individual parts are described using self-similarity descriptor (SSD) [13] that are augmented with descriptions of relative geometry. These descriptors are also independent of photometric details.

The importance of structure has been underlined as a basis for generic representation of objects [1, 9, 10, 21], and has previously been used for matching and detection of objects, albeit not invariant to depiction. The connected segmentation tree [7] captures adjacency and

http://dx.doi.org/10.5244/C.26.56


(a) Sculpture


(b) Synthetic


(c) Tattoo


(d) Drawing


(e) Oil


(f) Watercolour

Figure 1: A face is a face is a face, no matter how it is depicted.

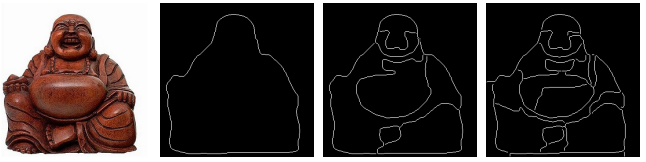
containment relationships of segmented regions in an image, and is useful for learning the common structure in objects from the same class.

There is little work on matching across depictive styles, but the self-similarity descriptor (SSD) of Scechtman and Irani [13] has proven to be capable in that regard. As originally used, SSD matching employs a "star graph" to match; we allow a more general structure and (as mentioned above and detailed below) augment SSDs with geometry. Evidence for the utility of structure to our problem come from Bai *et al* [20], who build a depiction invariant classifier using structure as the only feature. However, the classifier yields broad classes, which motivates our richer description of parts. Previous research into depiction invariant matching labels graph nodes with qualitative shapes [6]. However, the particular set of shapes used were presumed to be useful by the algorithm designers, and finding the optimal qualitative shape for any given object part is expensive and not straight forward. Recent work has solved both of these problems [19]. Impressive results have been obtained for matching similar images of different depictions based on learning [14]. In this work, SVM weights are learned for salient parts of a given input image, against a dataset consisting of millions of images taken from the world wide web as "negatives".

The main contribution of this paper is to show that it is possible to perform depiction invariant object matching without the need for any learning. Matching results are shown using two methods – one that minimises an energy function between graphs representing a pair of objects, and another that uses a sliding window approach to search for suitable matches in the second image, followed by a max-sum solver. This second approach is our own, and so represents a further contribution. Results obtained show that the proposed descriptor provides performance comparable to the state of that art on a dataset that contains only photographs, and outperforms existing methods on a dataset consisting of mixed depictions.

2 Object Description

This section gives details of our depiction-invariant description for visual objects. In overview, we use a hierarchical segmentation to obtain a structural descriptor in which nodes are the



(a) Input (b) Level 1 (c) Level 2 (d) Level 3

Figure 2: Segmentation hierarchy. Image of a laughing Buddha shown in (a), and three levels of the corredponding segmentation hierarchy in (b), (c), (d).

segmented regions. Next, each region is described using non-photometric features.

2.1 Hierarchical Segmentation

In this paper, every image is represented as a tree of hierarchical regions generated by using the output of any hierarchical segmentor. Graphs are a commonly used representation for segmentation hierarchies, and especially useful to encapsulate the property of *structure*. Although a number of hierarchical segmentors were used for experiments, the state of the art segmentor based on ultrametric contour maps [4] was found to be the most consistent and has been used to showcase all results in this paper.

To construct the segmentation hierarchy of an image, a large set of regions are first obtained using the oriented watershed transform introduced in [4]. Next, contours in the transformed image are weighted according to the similarity between the intersecting edges by using the contour's gPb strength [12]. These segmentations are then organised into a nested collection of segmentations by a greedy graph-based region merging algorithm, to obtain a representation termed as the ultrametric contour map [3].

One disadvantage of the above representation is the depth of the segmentation hierarchy, which was found to be an average of 89 per image across all images in the Caltech 101 object categories dataset [8]. In order to use hierarchical segmentations to match across images, a lower number of levels in the trees was found to be advantageous in terms of both accuracy and efficiency. The Laplacian Graph Energy [11] has previously been effective for reducing the number of levels in a hierarchy by an order of magnitude, with little or no loss [15], and is used for improved object representations in this paper. Such segmentation trees, obtained from a combination of the Oriented Watershed Transform, Ultrametric Contour Maps and component-wise Laplacian Graph Energy, are termed as OWT-UCM-cLGE trees for brevity. After applying a reduction by selecting a subset of levels in the tree, we choose only 4 levels to represent any image. This was found to be sufficient to obtain hierarchies of objects in the Caltech 101 object categories dataset, such that no salient object part was over-segmented or under-segmented. Figure 2 shows an example of an image segmented into a three-level hierarchy.

2.2 Segmentation Graph

Building a hierarchical graph $G =< V, E >$ out of the segmentation hierarchy is straightforward. Each segmented region in the hierarchy is represented by a vertex $v \in V$, where V is the set of all vertices. The set of edges E contains edges $\{e_{ij} = v_i v_j | v_i, v_j \in V\}$ that connect vertices v_i and v_j. v_i and v_j may belong to the same level, or neighbouring levels

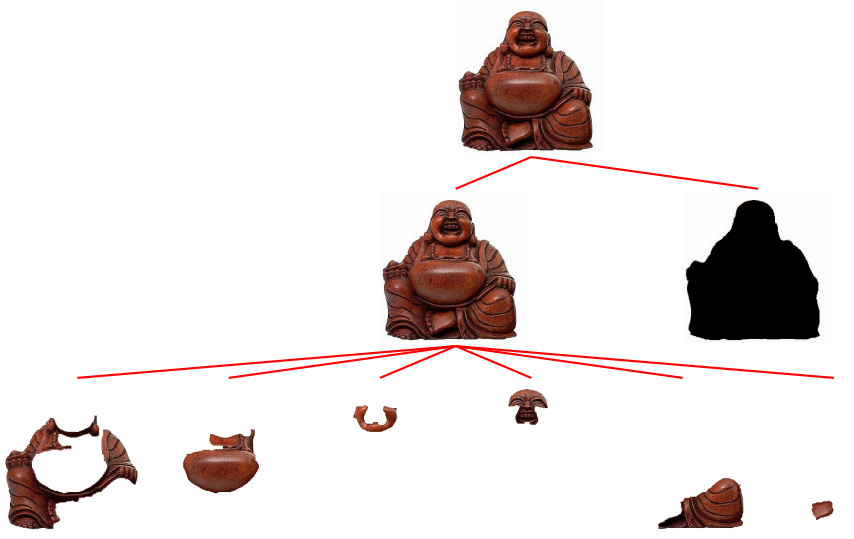


Figure 3: Segmentation graph. First two levels of the segmentation graph obtained for the hierarchy shown in Figure 2. In Level 1, the object is separated from the background.

of the hierarchy. Vertices at the same level are connected by an edge if their corresponding regions share a boundary in the segmented image. Parent-child connectivity is determined by checking segments that intersect across two consecutive levels of the hierarchy. Graph G now contains an efficient representation of the spatial arrangement of segmented regions, and encapsulates relations of adjacency and containment between the regions.

A consistent observation across a number of image segmentation methods is the over-segmentation of highly textured parts of the image. It has been common practice to ignore small regions as noise and exclude them from the segmentation hierarchy [2, 17]. This sometimes results in the loss of meaningful parts of the image leading to an incomplete representation. This problem is countered using an intuitive fix. Prior to exclusion of nodes from the hierarchy via area-based thresholds, adjacency checks are applied to the set of over-segmented or noisy regions. If a pair of over-segmented regions are connected by an edge, which effectively means they are neighbouring regions in the segmented image, they are combined to form a single larger region.

Consider $\omega \in V$ as the set of regions whose area is less than a pre-defined threshold and are possibly over-segmented. If any two regions $v_p, v_q \in \omega$ are connected by an edge $e_{pq} = v_p v_q \in E$, then they are combined to form a single region v_{pq}. Vertices v_p, v_q and edge e_{pq} are removed from ω. v_{pq} is then added as a new node in ω and inherits the adjacencies of v_p and v_q. This step is repeated until there are no neighbouring regions remaining in ω. Once this condition is satisfied, every vertex $v \in \omega$ and its edges are added to G if the associated region's area now exceeds the specified threshold.

2.3 Region Description

In terms of adjacency and containment, graph G encapsulates the structure of an object at a global level. The next step involves description of each of the segmented regions at a

Figure 4: An example of matching across depictions using the self-similarity descriptor. Using features extracted from a template (shown on left), an exhaustive pixel-wise search is carried out on a set of test images (shown on right) using a sliding window approach, across a number of scales. In each case, the green box highlights the region that minimised the cost of matching the template. From this set of faces captured in varying depictions, it is clear that the SSD provides a useful description for depiction-invariant matching.

local level. In order to match across depictions, the descriptor must make use of strictly non-photometric features, a claim which is substantiated in results presented later in this paper. The self-similarity descriptor (SSD), through edges and repetitive patterns across the described region, captures the internal geometric layout of local self-similarities [13].

In order to compute the SSD for a given pixel, a correlation surface is obtained by computing the sum of squared differences between a small and a large patch, both centred at the same pixel. The smaller patch is moved across the larger patch and the sum of squared differences of all overlapping pixels is computed to obtain the correlation surface associated with the centre pixel. The correlation surface is then transformed to a log polar representation and features are binned into 20 angles and 4 radial intervals. As a result of this procedure [1], each pixel is represented by a feature vector of 80 dimensions, based on the parameters we have chosen in our experiments.

Figure 4 examines the usefulness of SSD as a depiction-invariant region descriptor. Given a template image (shown on the left), extracted features are exhaustively matched on each of a set of images from the same object class (shown on the right) using a pixel-wise sliding window approach over a number of scales. The green box shown in each case highlights the region that minimises the cost of matching to the template. Though there is significant variation in texture, colour and other photometric properties across the illustrated set of images, this does not pose a constraint for accurate matching using SSD's.

SSD's have previously been applied to object matching and detection [5, 13], but these methods are of the order of $O(n^4)$, which is computationally rather expensive. The following section examines a couple of optimised methods to match objects, showing significant reduction in computational load with minimal or no loss of accuracy.

In addition to SSD's, two geometric features are included to describe every region – the ratio of the areas of the region and its parent, and the orientation of the line that connects the centroid of the region to the centroid of its parent.

[1] Extraction of SSD's is covered in detail in [13]

3 Matching

In this section, two methods are presented for matching across pairs of images. The first method, based on Feature Correspondence Graph Matching [16], is used to compute a mapping between the segmentation graphs of the two objects. However, inconsistent segmentation of different instances of an object brings about failure cases with this method. To counter this problem, a new approach to object matching is introduced, based on maximising the overall quality of a Markov Random Field [18] created by using a sliding window search.

3.1 Global Optimisation

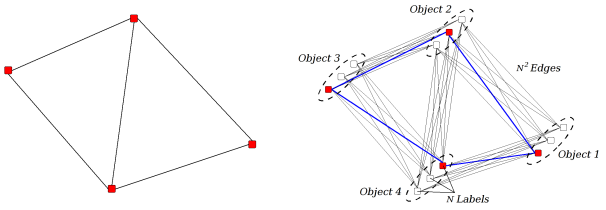
Segmented regions of the hierarchy are likely to be of varying sizes, which gives rise to feature vectors of variable lengths. In order to simplify feature matching, a fixed-length feature vector is obtained for every region by simply averaging the SSD features for all pixels within the region. Although averaged features are not the most descriptive (especially at the top of the hierarchy), they are sufficiently descriptive through the hierarchy for the purpose of matching objects. On the other hand, structural stability is higher towards the upper end of the hierarchy, and makes up for the loss of information encountered due to averaging.

Given two segmentation graphs, $G1 =< V1, E1 >$ and $G2 =< V2, E2 >$ obtained from a pair of images to be matched, the aim is to compute a mapping $f : G1 \rightarrow G2$ – a set of correspondences between regions of the two graphs that minimises matching cost. In order to maintain structural coherence, a constraint is applied such that $(f(v_i), f(v_j))$ is an arc of $G2$ if and only if (v_i, v_j) is an arc of $G1$, for $v_i, v_j \in V1$ and $f(v_i), f(v_j) \in V2$. Region similarity is encapsulated in a $V1 \times V2$ matrix S, where $S(i, j)$ is the $L1$ distance between features extracted from the region corresponding to $i \in V1$ and $j \in V2$. A state of the art graph matching algorithm [16], that carries out dual decomposition to compute a global minimum, is applied to find the mapping f.

3.2 Sliding Window Search

This is a method of our invention. For a given pair of images to be matched, the segmentation graph $G1 =< V1, E1 >$ is only computed for one of the two objects. Each region $v \in V1$ is applied as a template for conducting a sliding window based search through the second image, with local matching cost computed at every step using the $L1$ distance between the features extracted for v and the intersected region from image 2. The N-best matches for each region are collected to compute a Markov Random Field, in a similar manner to a previous method for matching anatomical structures [7].

Each region $v \in V$ forms a vertex in the MRF, called an object, and consists of N fields or labels, with associated qualities. The quality of a label is inversely proportional to the cost of matching the label to its corresponding object. The labels of two adjacent nodes are fully connected by N^2 edges. The quality of an edge is computed as the weighted sum of its length and orientation similarities to the corresponding edge in the model graph. An example of such a graph is shown in Figure 5. The best match in the second image corresponding to $G1$ can be found by computing the max-sum of label and edge qualities on the MRF.



(a) Sample Graph (b) Corresponding MRF

Figure 5: An example illustrating the construction of an MRF, given N-best matches for each vertex of a graph.

4 Experiments

Experiments are conducted on three datasets of images. The first, called *photos only*, is a subset of 400 images across 5 categories chosen from the Caltech 101 dataset [8]. The second, called *art only* is a new dataset that consists of 300 non-photographic images of the same 5 object categories. The third dataset is a union of *photos only* and *art only* and is called the *mixed* dataset.

In order to label ground truth, best matches were chosen between segmentation graphs of all possible pairs of images from the same class. Given two images, one of three labels were assigned for every region in either image –

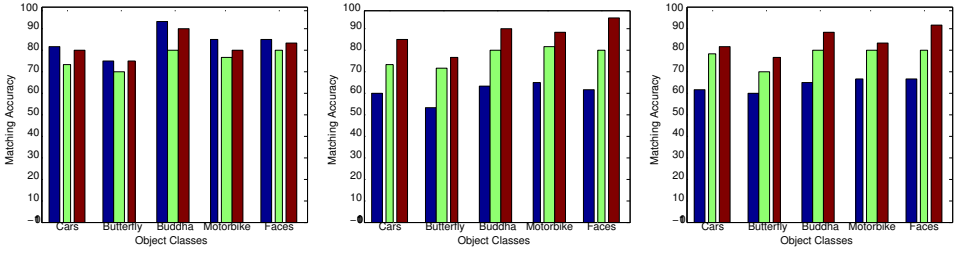
- A corresponding region that matches in the other image.
- A corresponding region that matches, albeit with some difficulty such as partial visibility.
- No match.

Accuracy of matching was measured by checking against labelled ground truth. A pair of images of the same class is said to be matched correctly if at least 75% of matched regions agree with human labelling. The overall matching accuracy is simply the ratio of correctly matched object pairs to the total number of all possible object pairs in every class.

To compare results with the state of the art, experiments were carried out to compare performance of object matching using photometric features from [1], and the non-photometric features examined in 2.3. Various configurations of the features and matching methods were used for analysing performance:

1. OWT-UCM-cLGE segmentation trees, photometric features, global optimisation. matcher
2. OWT-UCM-cLGE segmentation trees, SSD features, global optimisation matcher.
3. OWT-UCM-cLGE segmentation tree, SSD features, sliding window search.

All 3 methods were applied to all 3 datasets, and the results can be compared in Figure 6. It can be seen that Method 1 performs best on the *photos only* dataset, closely followed by Method 3. With the use of photometric features on a purely photographic dataset, this result concurs with what one would expect. The comparable performance of Method 3 on this dataset indicates that depiction-invariant features are sufficient to match photographs. Methods 2 and 3 clearly outperform Method 1 with the *art only* and *mixed* datasets, hence verifying that photometric features are not robust to variation in deptictive styles.



(a) Photos (b) Art (c) Mixed

Figure 6: Matching accuracies obtained for graph matching with photometric features (blue), graph matching with SSD features (green) and sliding window search with SSD features (red), for 5 object categories over the three datasets.

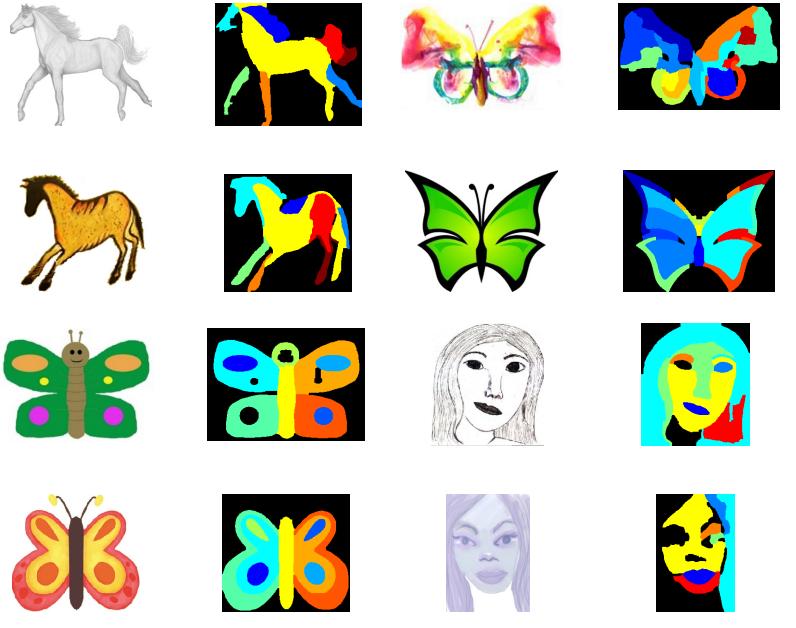


Figure 7: Results obtained using FCGM with SSD features on various samples of the art dataset. Each pair of images is colour-coded to show regions that have been matched.

Examples of object matching with SSD features and global optimisation are shown in Figure 7, with colour codes to represent matched regions. For instance, in Figure 7(a), two images of horses depicted differently have been matched by parts. The head in the first image matches the head in the second image (shown in blue), and the same follows for the body (shown in yellow) and other parts.

4.1 A Further Study

Recent work [14] examines the use of exemplar-based SVM's to perform depiction-invariant matching using HOG features, and provides impressive results on a large dataset retrieved from the internet. In order to compare performance, two variants of this method – (a) the

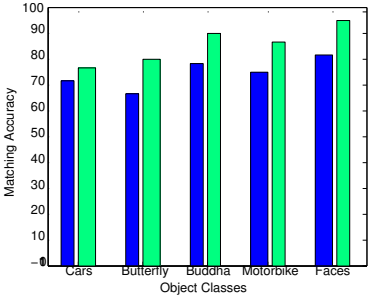


Figure 8: Results obtained when exemplar SVM's are used to match objects in the mixed dataset. Results with HOG features are shown in blue, and those with SSD features are shown in green.

original configuration [14], and (b) by replacing HOG with SSD features – are applied for matching objects in the *mixed* Dataset. A comparison of the results, shown in Figure 8, indicate that the objects are better described using SSD's.

5 Summary and Conclusion

The main contribution of this paper lies in the introduction of an object description that is consistent across depictive styles. Segmentation-based hierarchies are used to represent the global structure of the object, which remains invariant to depiction. Locally, each region is described using non-photometric and structural features.

Three datasets were used to evaluate performance – *photos only*, *art only* and *mixed* – the latter two containing depictions of objects in varying styles. Experiments in which the matching algorithm is constant but image representation changes (from photometric to our proposal) shows that matching performance compares well with state of the art for *photos only*, but exceeds that for the other two – the improvement being most marked for *art only*. Our sliding window matcher, using our descriptor, improves on the global optimisation matcher in all cases.

We conclude that description is important to the problem of cross-domain matching, and that learning is not necessary to achieve that task.

References

[1] N. Ahuja and S. Todorovic. Learning the taxonomy and models of categories present in arbitrary images. In *International Conference in Computer Vision (ICCV)*, 2007.

[2] N. Ahuja and S. Todorovic. Connected segmentation tree - a joint representation of region layout and hierarchy. In *Computer Vision and Pattern Recognition (CVPR)*, 2008.

[3] P. Arbelaez. Boundary extraction in natural images using ultrametric contour maps. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2006.

[4] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. From contours to regions: An empirical evaluation. In *Computer Vision and Pattern Recognition (CVPR)*, 2009.

[5] S. Bagon, O. Brostovski, M. Galun, and M. Irani. Detecting and sketching the common. In *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[6] A. Balikai, P.L. Rosin, Y.-Z. Song, and P.M. Hall. Shapes fit for purpose. In *British Machine Vision Conference (BMVC)*, 2008.

[7] R. Donner, B. Micusik, G. Langs, and H. Bischof. Sparse mrf appearance models for fast anatomical structure localisation. In *British Machine Vision Conference (BMVC)*, 2007.

[8] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding (CVIU)*, 2007.

[9] S. Fidler, M. Boben, and A. Leonardis. Learning hierarchical compositional representations of object structure. *Object Categorization: Computer and Human Vision Perspectives*, 2009.

[10] S. Geman. Hierarchy in machine and natural vision. In *Scandinavian Conference on Image Analysis (SCIA)*, 1999.

[11] Ivan Gutman and Bo Zhou. Laplacian energy of a graph. *Linear Algebra and its Applications*, 2006.

[12] M. Maire, P. Arbeláez, C. Fowlkes, and J. Malik. Using contours to detect and localize junctions in natural images. In *Computer Vision and Pattern Recognition (CVPR)*, 2008.

[13] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *Computer Vision and Pattern Recognition (CVPR)*, 2007.

[14] Abhinav Shrivastava, Tomasz Malisiewicz, Abhinav Gupta, and Alexei A. Efros. Data-driven visual similarity for cross-domain image matching. *SIGGRAPH ASIA*, 2011.

[15] Y.Z. Song, P. Arbelaez, P. Hall, C. Li, and A. Balikai. Finding semantic structures in image hierarchies using laplacian graph energy. *European Conference on Computer Vision (ECCV)*, 2010.

[16] L. Torresani, V. Kolmogorov, and C. Rother. Feature correspondence via graph matching: Models and global optimization. *European Conference on Computer Vision (ECCV)*, 2008.

[17] A. Torsello, A. Albarelli, and M. Pelillo. Matching relational structures using the edge-association graph. In *Image Analysis and Processing (ICIAP)*, 2007.

[18] T. Werner. A linear programming approach to max-sum problem: A review. *Pattern Analysis and Machine Intelligence (PAMI)*, 2007.

[19] Q. Wu and P. Hall. Prime shapes in natural images. In *British Machine Vision Conference (BMVC)*, 2012.

[20] B. Xiao, Y. Z. Song, A. Balikai, and P. M. Hall. Structure is a visual class invariant. In *Structural, Syntactic, and Statistical Pattern Recognition (SSSPR)*. 2008.

[21] L.L. Zhu, C. Lin, H. Huang, Y. Chen, and A. Yuille. Unsupervised structure learning: Hierarchical recursive composition, suspicious coincidence and competitive exclusion. In *European Conference on Computer Vision (ECCV)*, 2008.