Free-Shape Polygonal Object Localization*

Xiaolu Sun, C. Mario Christoudias, and Pascal Fua

CVLab, EPFL, Lausanne, Switzerland
{xiaolu.sun,mario.christoudias,pascal.fua}@epfl.ch

Abstract. Polygonal objects are prevalent in man-made scenes. Early approaches to detecting them relied mainly on geometry while subsequent ones also incorporated appearance-based cues. It has recently been shown that this could be done fast by searching for cycles in graphs of line-fragments, provided that the cycle scoring function can be expressed as additive terms attached to individual fragments. In this paper, we propose an approach that eliminates this restriction. Given a weighted line-fragment graph, we use its *cyclomatic number* to partition the graph into managebly-sized sub-graphs that preserve nodes and edges with a high weight and are most likely to contain object contours. Object contours are then detected as maximally scoring elementary circuits enumerated in each sub-graph. Our approach can be used with *any* cycle scoring function and multiple candidates that share line fragments can be found. This is unlike in other approaches that rely on a greedy approach to finding candidates. We demonstrate that our approach significantly outperforms the state-of-the-art for the detection of building rooftops in aerial images and polygonal object categories from ImageNet.

1   Introduction

Polygonal objects ranging from fields and rooftops in aerial images to signs, furniture, and facades in ground-level views are prevalent in man-made environments. They have received much attention since the very beginning of the Computer Vision field, starting with the Blocks World [35]. Many early approaches formulated the problem of finding them in terms of perceptual grouping of edges that exhibit the right geometry [21]. Over the years, it has become apparent that only looking at edges was insufficient and that, to distinguish valid polygonal regions from spurious ones, it was indispensable to also consider the pixels these edges enclose [5].

Many recent algorithms do this by treating image edges or line fragments as nodes of graphs whose cycles represent closed contours. Delineating polygonal regions is then accomplished by finding those cycles that minimize an appropriate objective function. Even though the number of potential cycles can grow very large even in moderately-sized graphs, this can be done efficiently when the objective function can be written as a sum of terms, one for each edge of the cycle [45,47]. However, this is limiting because using more complex non-linear
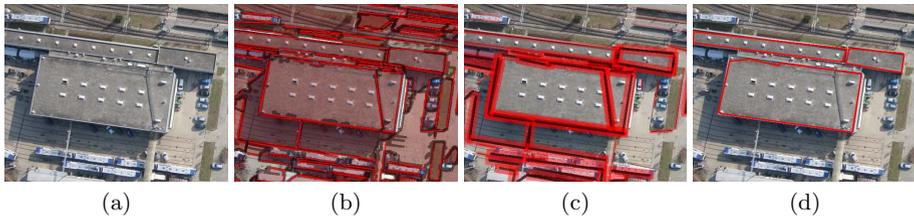

Fig. 1. Free-shape polygonal object detection: (a) Aerial image of a set of buildings and (b) detected image line fragments that define the nodes of our line fragment graph and their associated image regions. (c) Graph node and edge weights are used to quickly prune the search space and focus on sub-regions likely to contain object contours. Thicker line fragments and darker red highlights reflect larger node and edge weights respectively, each line fragment highlighted by its maximum incident edge weight. (d) We detect polygonal objects as high scoring circuits in the partitioned line fragment graph. Detected rooftops are displayed in red. Best viewed in color.

objective functions, such as those based on kernel SVMs, is required in many real-world scenarios. Furthermore, when looking for multiple objects, these approaches tend to rely on finding the best candidate, removing the corresponding edges, and then finding the next one. This precludes finding shapes that share edges, which is important in densely packed environments.

In this paper, we overcome these limitations using a graph search algorithm that partitions the graph of line fragments into smaller ones using discriminative node and edge weighting functions that encode how likely a line fragment or line fragment pair is to belong to an object. We constrain the size of each sub-graph using its easy to compute *cyclomatic number* [14] to limit the number of its cycles. Object contours are then found as elementary circuits in each sub-graph. As we show, for typical image, line-fragment graphs, object contours can be found within relatively small-sized sub-graphs having only a few elementary circuits, that can be enumerated efficiently. Fig. 1 illustrates our approach. In contrast to previous methods, it lets us use generic shape and appearance cues to score each cycle that are not restricted to linearly additive measures and can easily generate multiple hypotheses that share some edges. As seen in our experiments, this yields significantly better accuracy at no increase in computational cost.

We evaluate our approach for the detection of building rooftops in aerial images and other polygonal object categories from ImageNet [12], and explore the use of Histogram of Oriented Gradients [11] and normalized color histogram representations as cycle scoring objective functions, each of which are non-additive. As seen in our experiments, our approach significantly outperforms recent polygonal and free-shape object detection methods [47,43].

2   Related Work

Early approaches detected shapes in images using perceptual saliency criteria to group image edgels [20,33,39]. An iterative optimization method to group

image edgels based on local curvature and curvature variation was proposed in [39]. Similar ideas were explored in [20,33] that investigated measures such as co-curvilinearity and co-circularity for perceptual grouping. Spectral methods for grouping the elements of the resulting edgel graph [34,36,38] and probabilistic approaches useful for incorporating more global dependencies [10,13,16] and edgel detection uncertainty [9] have also been proposed. While useful for finding dominant shapes in images, these methods have largely focused on shape saliency and less on finding objects of a particular shape.

Voting methods based on the Hough Transform can be used to detect image contours of a specific shape [2,3,24,28]. With these approaches, each edgel votes for its shape parameters and shape instances are detected as peaks in the resulting hypothesis voting space. The Hough Transform has been demonstrated for the detection of simple shapes including lines, circles, and rectangles [24,28] and regular polygons whose edges inscribe a circle [3]. It has also been applied for the detection of arbitrary shapes [2], but becomes computationally prohibitive for complex shapes involving many parameters.

To overcome these limitations many methods have been proposed that leverage annotated images to learn models of object shape [4,8,7,19,25,32,40]. Statistical shape models define flexible and rich representations capable of efficiently modeling and detecing objects with a complex geometry. Initial approaches defined holistic or "top-down" models that incorpated global object shape statistics [4,8,7], a prevalent example being the Active Shape Model [8] that leverages dominant modes of object shape variation to define a deformable object template. More recent methods utilize local models of object geometry and learn a grammar of object parts that are individually detected in the image and then fused in a bottom-up fashion [19,25,32,40]. Although versatile, these methods have largely focused on modelling object shape and less on appearance.

Segmentation-driven detection methods comprise an alternative class of techniques that exploit image region or appearance cues to generate object hypothesis obtained from a bottom-up segmentation of the image [43,17,6,37,26]. [17,6] form object region hypothesis from multiple figure-ground image segmentations each obtained using either varying segmentation parameters or with different foreground seed locations. Similarly, [43] employs several hierarchical image segmentations computed across various image representations and grouping criteria. These methods largely focus on deriving category-independent region proposals, however, which although related is a different problem than what we address in this paper. Also, most of them do not account for region geometry.

Recent methods have focused on finding polygonal or *free-shape* objects using both object shape and appearance [44,46]. [46] extends the branch-and-bound method of [27] to find k-sided bounding polygons with a bag-of-words appearance model [41]. Similarly, [44] proposes a branch-and-cut algorithm to efficiently find the best scoring free-shape object region. While efficient, these methods rely on a linearly additive objective function. Yet many measures of interest involve non-additive scoring functions and therefore cannot be used in conjunction with

these methods. In contrast, our approach can be employed for polygonal object detection with *any* scoring function.

Probably the closest approach to ours is the ratio-contour algorithm [45,47,48] and related approaches that formulate salient boundary detection as finding minimum cost cycles in a graph [5,15,29]. Finding the minimum cost cycle of a graph, however, can only be done efficiently when the scoring function can be written as a sum of terms, one for each edge of the cycle [45]. Furthermore, when searching for multiple objects, these approaches employ a *greedy* optimization. Each subsequent solution is found by removing the previous best solution from the search space, which precludes finding shapes with common nodes or edges. In contrast, our approach enables the use of generic shape and appearance measures beyond linearly additive ones and can easily generate multiple, overlapping hypotheses.

3   Polygonal Object Detection

Our goal is to find polygonal objects of arbitrary complexity. We start from line fragments and treat these fragments as nodes of a graph whose edges encode geometric relationships. Cycles in this graph define polygonal shapes that enclose an image region. Our problem then amounts to finding the best possible such cycles in terms of a suitable objective function. We are particularly interested in finding *elementary circuits*, i.e., connected cycles whose vertices have degree two, as these generally correspond to well defined object boundaries that are each comprised of a single simple cycle[1]. In general, this is difficult because, even in relatively small graphs, the number of potential cycles can be exceedingly large. For a fully connected graph with n nodes, the number of elementary circuits can grow faster than 2^n [23], which is computationally prohibitive for most real life applications. In this section, we first describe our graph construction approach in more detail. We then outline our graph-partitioning algorithm to efficiently search the potentially large space of all possible graph cycles.

3.1   Graph Construction

As discussed above, our first step is to extract line fragments and use them to build graphs such as the one of Fig. 2. As polygonal outlines should be evaluated not only according to their geometry but also to the color and texture of the area they enclose, we take our line fragments to be straight segments extracted using a hough-style algorithm from the boundaries of Maximally Stable Extremal Regions (MSERs) [30]. We find MSERs using a combined edge and intensity image whose edge scores are computed with the method of [1] as described in [42]. This results in a better over-segmentation of the image that more faithfully respects its underlying contours.

The detected image line fragments define the nodes of our graph. Each line fragment is oriented in a clockwise direction along the boundary of its associated

---

[1] Elementary circuits can contain self-intersections, however, this can be easily checked during their enumeration.
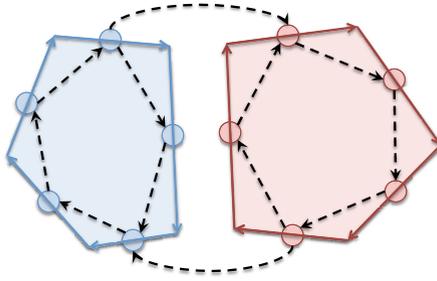
Fig. 2. Line fragment graph. Line segments are shown for two neighboring image regions colored according to their associated region that define the nodes of the graph. Each line fragment is oriented in a clockwise direction and we instantiate directed edges between nearby line fragments of compatible orientation, displayed as dashed, black arrows. Elementary circuits in the graph correspond to closed polygonal image contours whose image regions all lie within the same contour.

MSER region and we instantiate directed edges between nearby line fragments of compatible orientation, as illustrated in Figure 2. The distance between two directed lines is defined as the minimum distance between the *head* and *tail* endpoints of each line. In our implementation, we declare two lines as *nearby* if their distance is within 50 pixels of one another. A pair of lines are of a *compatible orientation* if they are nearby and their minimum head-tail distance is less than their head-head and tail-tail distances. Elementary circuits in the graph define closed polygonal outlines that enclose their respective MSERs. The large number of cycles in this graph is in general prohibitive making exact search difficult. In what follows, we outline an efficient search algorithm for addressing this hard optimization problem.

3.2   Graph-Partition Search Algorithm

More formally, let I be an image and $\{v_i\}_{i=1}^{N}$ the detected line fragments in I. To formulate our search problem we define a weighted, directed graph $G = \langle V, E \rangle$ whose nodes are the detected line fragments $v_i \in V$ and directed edges $e_{ij} \in E$ are defined between nearby line fragments. The node and edge weight functions, $u(v) : V \to \mathbb{R}$ and $w(e) : E \to \mathbb{R}$, encode how likely a line fragment or pair of line fragments is to belong to an object contour that are learned from labeled object contours as described in Section 3.4.

Each elementary circuit $c \in G$ defines a closed, polygonal outline. Let $f(c)$ be a scoring function that represents the likelihood that c truly is the outline of an object. We wish to find the image circuit or set of image circuits c^* that maximize f:

$$c^* = \mathrm{argmax}_{c \in G} f(c). \qquad (1)$$

In the special case of additive scoring functions defined over sums of the node and edge weights, u and w, this maximum can be computed in polynomial time

[45,47]. Yet, there are many scoring functions of interest that are non-additive. For such functions finding the globally optimal solution of Eq. 1 is difficult. Instead, we propose to find local optima over a partitioning of the line fragment graph to result in an efficient search algorithm and address this challenging optimization problem. We first provide a brief overview of cycle space graph theory that we use to obtain an upper bound on the number of cycles in a graph and then present our graph-partition search algorithm.

Consider a spanning tree T of G. For each non-tree edge $e \in G$ let c_e be the path connecting the endpoints of e in T. c_e is an elementary circuit of G, also referred to as a *fundamental circuit*. Together the c_e form a basis for the *cycle space* of G [14]. Let $c \in \{0, \pm 1\}^{|E|}$ define a cycle in G such that each entry of c is ± 1 if the corresponding directed edge in G belongs to it and is 0 otherwise[2]. Any cycle c can be expressed as a linear combination of the fundamental circuits c_e:

$$c = C_e \mathbf{x}, \qquad (2)$$

where $C_e \in \{0, \pm 1\}^{|E|} \times \{0, \pm 1\}^m$, each column of C_e is a fundamental circuit c_e of G and $\mathbf{x} \in \{0, \pm 1\}^m$ are the coefficients of c. $m = |E| - |V| + 1$ is the *cyclomatic number* of G and is the dimension of its cycle space.

While any cycle can be expressed as a linear combination of cycle basis, for a directed graph not all linear combinations of cycle basis produce a valid graph cycle. Let $\mathcal{X} = \{\mathbf{x} \in \{0, \pm 1\}^m \mid C_e \mathbf{x}$ is a cycle in $G\}$ define the set of valid cycle coefficients. Our search problem can be re-expressed in terms of the cycle space of G as,

$$\mathbf{x}^* = \mathrm{argmax}_{\mathbf{x} \in \mathcal{X}} f(C_e \mathbf{x}). \qquad (3)$$

Eq. 3 defines our problem in its most general form, in that it allows for the maximization of *any* scoring function f. Finding an exact solution, however, is difficult and would require time exponential in m. Moreover, although m can be used to obtain an upper bound on the number of cycles in a graph, a graph's cycle space can span generic cycles beyond elementary circuits. Instead, we propose an approximate solution based on a partitioning of the line fragment graph and restrict our search to elementary circuits that can be found efficiently for each sub-graph using [23].[3]

A key insight behind our approach is that object contours are typically contained within relatively small sized sub-graphs of the line fragment graph. Our goal is then to partition the graph into smaller sub-graphs that are most likely to contain object contours. Unlike other approximate search methods, most of which are based on a greedy merging of high scoring contour fragments [10,13,16], we prune weak edges so as to preserve the flexibility for the remaining fragments to re-form cycles according to more generic measures. This is accomplished by using the node and edge weight functions u and w to iteratively preserve nodes and edges with a high weight and discard those with a

---

[2] For ease of notation, we use c to denote both elementary circuits and generic graph cycles.

[3] When enumerating elementary circuits, we additionally constrain our search to ones that do not contain intersecting line fragments or graph edges.

low one. This results in an efficient search algorithm that operates on a set $G' = \{G_i = \langle V_i, E_i \rangle\}_{i=1}^{P}$, $V_i \subseteq V$ and $E_i \subseteq E$, of P small-sized sub-graphs obtained by a partitioning of G that retains nodes and edges with a large weight.

We can then re-express the original optimization of Eq. 1 with respect to this partitioning and enumerate the cycles of G as those of each of its sub-graphs,

$$c_{approx}^* = \text{argmax}_{c \in G'} f(c). \tag{4}$$

The worst-case complexity of our algorithm is $O(2^{m_{\max}})$, where $m_{\max}$ is the cyclomatic number of the largest sized sub-graph. Although this already represents a significant savings for $m_{\max} \ll m$, the resulting algorithm can still be restrictive. Thankfully the number of elementary circuits in a typical line fragment sub-graph is significantly smaller than $2^{m_{\max}}$ and they can be enumerated rather efficiently[4].

Our graph search algorithm is summarized as Algorithm 1. The node weight function u is first used to prune low-scoring nodes from the graph whose weight is below a threshold μ. The edge weight function w is then recursively thresholded to divide G into P sub-graphs each having a cyclomatic number of at most $m_{\max}$, such that the lowest scoring edges are removed from the graph. Together μ and $m_{\max}$ define the parameters to our search algorithm that can be used to tradeoff computational cost with approximation quality. As evidenced in our experiments, our graph-partition based search algorithm results in empirically good solutions using generic weight measures while maintaining a relatively low computational cost.

3.3   Cycle Scoring Functions

The cycle scoring functions are used to model the global appearance and shape of an object and select the most promising cycles. We employ a Histogram of Oriented Gradients (HOG) representation in combination with a Support Vector Machine (SVM) classifier to define our cycle scoring function. Provided labeled object contours, we learn a HOG-SVM cycle scoring function for each object class as

$$f_{\text{HOG}}(c) = \sum_i \alpha_i K(\Psi(c_i), \Psi(c)), \tag{5}$$

where c_i are circuit support vectors learned from training data and α_i their corresponding weights, and $K(\cdot)$ is a pre-specified kernel or similarity function. $\Psi(c)$ is a HOG appearance vector computed over its rectangular extent in the image, found by orienting c about its dominant orientation and normalizing it to a canonical scale.

The HOG-SVM scoring function cannot be decomposed over edge weights of G, particularly in the case of non-linear kernel functions. In fact, even for a linear kernel function, as it is scaled and oriented about each cycle, the HOG cycle

---

[4] In our experiments, setting $m_{\max} = 40$ resulted in approximately 3k elementary circuits per image on average.

Algorithm 1. Graph-Partition Search Algorithm

Input: An image I, line fragment graph $G = \langle V, E \rangle$, cycle score function $f(c) :$ $\{0,1\}^{|E|} \rightarrow \mathbb{R}$, and node and edge weight functions $u(v) : V \rightarrow \mathbb{R}$ and $w(e) : E \rightarrow \mathbb{R}$.

Parameters: Node weight threshold μ and sub-graph cyclomatic number threshold $m_{\max}$.

Output: Object contours c^*.

Initialize $C = \emptyset$.
if cyclomatic number of $G > m_{\max}$ **then**
    $\tau = \min_{e \in E} w(e)$.
    $V' = \{v \in V | u(v) \geq \mu\}$.
    $E' = \{e_{ij} \in E | (v_i, v_j \in V') \wedge (w(e_{ij}) > \tau)\}$.
    Define $G' = \langle V', E' \rangle$.
    for each connected component G_i of G' **do**
        $c_i = \text{GraphPartitionSearch}(G_i, f, u, w, \mu, m_{\max})$
        Add c_i to C.
    end for
else
    Enumerate elementary circuits C in G using [23].
end if

if initial call to GraphPartitionSearch **then**
    $c^* = \text{argmax}_{c \in C} f(c)$.
else
    $c^* = C$.
end if

Return: c^*.

---

feature is non-additive. The HOG representation has been widely applied for rectangular subwindow search [11,18]. In this work, we extend it for polygonal object detection. This can offer distinct advantages over the commonly used additive measures based on a bag-of-words [47], particularly for polygonal object classes whose shape is a discriminative cue.

In addition to HOG, we also consider a normalized color histogram feature, that for each RGB channel bins the color values within the cycle and normalizes each histogram to sum to one, and a bag-of-words representation as in [47] except with a Radial Basis Function (RBF) kernel. Similar to HOG, these define non-additive image measures that can be exploited by our approach.

3.4 Node and Edge Weight Functions

The node and edge weight functions encode how likely a line-fragment or line-fragment pair belong to an object that we use to guide the partitioning of the graph into meaningful sub-graphs. We use RBF-kernel SVMs to learn the node and edge weight functions from labeled line fragments and line fragment pairs based on their local geometry and appearance. A line fragment is labeled as
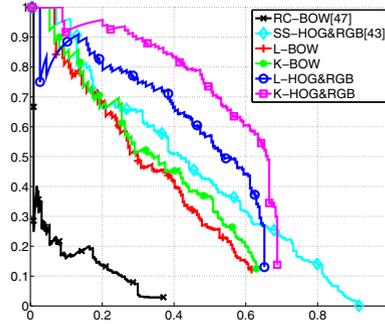
Fig. 3. *Precision-recall curves for building rooftop detection.* Results are displayed for the different settings of our approach and the baselines. The HOG and color RBF measure achieves the best performance, significantly outperforming the baselines and linear bag-of-words. Best viewed in color.

positive if it lies along the boundary of a ground-truth object instance and negative otherwise. Similarly, a pair of adjacent line fragments are labeled as positive if they both lie on the object boundary, and as negative if one lies on the boundary and the other does not.

We characterize the local appearance of a line fragment using a HOG descriptor computed over its scale and orientation normalized MSER region. We additionally use MSER color histogram features computed separately over both RGB and YCbCr channels and the MSER stability. For line pairs we compute a HOG descriptor over the scale and orientation normalized region defined by the combination of their MSERs and the absolute difference between their color histograms. Local geometry is then encoded using the angle and relative distance between adjacent lines as in [15]. The feature vector of a single line fragment or line fragment pair is then formed by concatenating its features into a single vector provided as input to the SVM.

4   Experiments

In this section, we demonstrate our approach for the purpose of detecting polygonal objects in man-made environments. We first consider the detection of building rooftops in aerial images. While there is an extensive literature on this topic (*e.g.*, for a survey see [31]), in this paper we focus on techniques that detect rooftops as cycles in line segment graphs [5,22] and compare to the state-of-the-art graph cycle detection method [47]. We then demonstrate our approach for more generic object detection using ImageNet [12]. In what follows, we first discuss our datasets, experimental setup and baselines, and then present our results.

Fig. 4. *Rooftop Detection.* Rooftop detections obtained by our approach and the baselines are displayed at a 50% recall rate. Correct detections are shown in green and false ones in red. Our approach significantly outperforms the baselines. Unlike bag-of-words, HOG not only encodes object appearance but also its global shape which can offer a better description of the polygonally shaped rooftops in these images and likely accounts for its better performance over ratio-contour. Additionally, false detections result in missed rooftops for ratio-contour, that do not degrade the recall of our approach even when using a bag-of-words scoring function. Best viewed in color.

4.1   Datasets

We use two datasets to evaluate our approach. The first consists of 65 aerial images of rural scenes containing several building rooftops many of which exhibit a fairly complex polygonal geometry. Each image is of size 1000×750 pixels. An example is shown in Figure 1. The second includes images from 10 different object categories from ImageNet [12]. They are sign, screen, remote control, cleaver, computer mouse, ipod, wine bottle, mug, beer bottle, and lampshade. We selected around 100 images per category, which were randomly split into equal-sized training and testing sets. We manually labeled the ground-truth contours of the objects in each image.

4.2   Experimental Setup and Baselines

We experiment with both additive and non-additive cycle scoring functions: a linear and RBF-kernel SVM using bag-of-words (BOW) features, which we refer to as **L-BOW** and **K-BOW** in our experiments and linear and RBF-kernel SVM with HOG and RGB color features referred to as **L-HOG&RGB** and **K-HOG&RGB**. Of these, **L-BOW** is the only additive one. For bag-of-words,

Table 1. Average precision on ImageNet. The average precision of each method is shown along with the mean average precision (mAP) across each category of ImageNet. Our approach results in a significant improvement over the baselines and linear bag-of-words.

Method/Category	sign	screen	remote	cleaver	mouse	ipod	wine	mug	beer	l-shade	mAP
RC-BOW [47]	0.37	0.43	0.42	0.06	0.10	0.27	0.08	0.16	0.15	0.05	0.21
SS-HOG&RGB [43]	0.35	0.43	0.46	0.30	0.37	0.38	0.47	**0.38**	**0.47**	0.34	0.39
L-BOW	0.42	0.25	0.60	0.32	0.26	0.38	0.29	0.15	0.17	0.17	0.30
K-BOW	0.47	0.30	0.54	0.34	0.33	0.40	0.36	0.19	0.27	0.32	0.35
L-HOG&RGB	0.49	0.49	0.63	0.42	0.50	0.32	0.47	0.20	0.34	0.49	0.44
K-HOG&RGB	**0.54**	**0.59**	**0.64**	**0.47**	**0.54**	**0.41**	**0.58**	0.27	0.40	**0.51**	**0.49**

we used SIFT keypoint descriptors and a dictionary containing 500 visual words computed with k-means. The cycle SVMs are trained from labeled samples found with our graph search algorithm. Cycles having an overlap greater than 60% with the ground-truth are labeled as positive and those with less than 50% overlap as negative. The *percent overlap* between two contours is computed as the area of their intersection divided by that of their union. For our graph search algorithm, we use a conservative node weight threshold of $\mu = -1$ and only disregard nodes that are highly unlikely to belong to an object and we cross-validate m_{max} with values $m_{max} = 10, 15, 20, 25, 30, 35, 40$ using 5-fold cross-validation on the training data.

We compare our approach with the state-of-the-art free shape object detection method–ratio-contour [47]. The ratio-contour algorithm is limited to additive measures, such as bag-of-words feature counts or area. We therefore evaluate it using a bag-of-words feature representation as described in [47] with the same MSER line fragments used by our approach, referred to as **RC-BOW** in our experiments. For multiple object detection, ratio-contour applies a greedy search that removes the optimal cycle from the graph and then is re-run to find the next one. In practice, we run it on each image until it cannot find anymore cycles. We also compare to the selective search algorithm of [43] as it is a representative approach that provides a similar or favorable performance to many of the recent segmentation-driven detection techniques [43], referred to as **SS** in our experiments. We use the code provided by the authors with the 'Fast' setting of their approach that resulted in a manageable number of object region hypothesis, on average 5.5k per image, and scored its regions using an K-HOG&RGB SVM classifier as this was our best performing scoring function.

We evaluate the baselines and each setting of our approach using precision-recall curves where detection accuracy is measured by the percentage overlap between the detected and ground-truth contours. As in previous work, a detection is considered to be correct if the detection accuracy is greater than 50%.

4.3   Results

Rooftop Dataset: Figure 3 displays the precision-recall curves for each method on this dataset. Our approach with the HOG and color RBF scoring function

consistently yields the best performance. Unlike bag-of-words, HOG encodes both global shape and appearance which is important for detecting polygonal objects. Additionally color helps avoid false detections such as those belonging to grass or dark shadow regions. When only using bag-of-words, introducing a non-linear kernel function also results in a significant improvement. Ratio-contour, however, is limited to a linear bag-of-words classifier, and cannot take advantage of non-additive scoring functions.

Furthermore, even when using the same linear bag-of-words scoring function, our approach still outperforms ratio-contour. This is due to ratio-contour's greedy nature. False detections can result in missed detections. This is illustrated by Figure 4 that displays the detections obtained by our approach along with those of ratio-contour on a set of representative images. Compared with the combined HOG and color scoring function, linear bag-of-words results in many more false detections. These are often higher scoring than cycles corresponding to true object contours, which for ratio-contour results in deleted building rooftop hypotheses. By contrast, our approach is not affected by this problem.

Our approach also outperforms selective search both in detection accuracy and quality, for which example detections are also displayed in Figure 4. This is in part due to our use of geometry for forming region hypothesis resulting in cleaner polygonal outlines, but can also be attributed to our use of discriminative, category-specific node and edge weight functions that help to quickly reduce and focus region hypothesis to those likely to be an object and increase accuracy.

ImageNet Dataset: Table 1 shows the average precision obtained for each one of the 10 ImageNet categories we have worked with. The corresponding precision-recall curves are displayed in Figure 5. Our approach achieves the best performance using a HOG and color RBF classification function and yields a
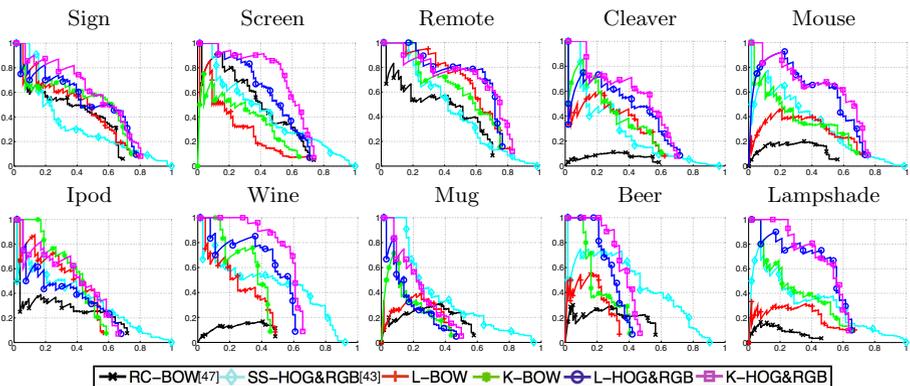

Fig. 5. *Precision-recall curves on Imagenet.* The results for the different settings of our approach and the baselines are shown for each category. Our approach obtains a significant improvement over the baselines and linear bag-of-words across all categories. Best viewed in color.

Fig. 6. *Polygonal object detection with ImageNet.* The detections obtained by our approach with the combined HOG and color RBF scoring function and the baselines are shown for example images from each category at 50% recall. Correct detections are shown in green and false ones in red. The baselines result in many false and missed detections that are significantly reduced with our approach. Best viewed in color.

significant improvement over the baselines, especially ratio-contour and linear bag-of-words, once again demonstrating the importance of non-additive measures like HOG. However, even for linear bag-of-words our approach exhibits a higher average precision than ratio-contour, which further illustrates the advantages of our graph search algorithm and suggests that it is a better way to prune the graph than relying on greedy search. We outperform selective search on 8 of the

10 datasets, with the exception of mug and beer bottle, as MSER detection was unreliable on these datasets, ratio-contour being similarly affected.

The detections returned by both our approach and the baselines on a set of example images from each category are shown in Figure 6. Ratio-contour suffers from significantly more false and missed detections than our approach. This is in part because for many of these categories, shape is an informative cue that bag-of-words does not capture and in part because of the greedy nature of the ratio-contour algorithm, which is not well suited to the detection of multiple objects. Selective search also results in more false detections and it often detects noisy object contours as it does take into account their geometry.

The computational requirements of our algorithm are comparable to those of ratio-contour. The rooftop dataset contains fairly large images that produce dense line-fragment graphs. For large graphs, ratio-contour is also costly and in fact takes longer than our approach. To process the rooftop dataset on one core of an Intel(R) Xeon(R) CPU @ 2.90 GHz, ratio-contour took 2 to 3 minutes per image on average, whereas our approach took about 1 minute. By contrast, for the ImageNet images of size 500×375, both our approach and ratio-contour took about 8 seconds per image on average.

5   Conclusion

This paper presented a graph-cycle based object localization algorithm that unlike previous approaches can exploit generic shape and appearance cues for polygonal object detection. We use the cyclomatic number to define an efficient graph-patition search algorithm and detect object boundaries as maximum scoring elementary circuits in a partitioned, image line fragment graph whose subgraphs preserve high-scoring node and edge weights. Our graph search algorithm can be used with *any* cycle scoring function to detect multiple polygonal objects in an image. We evaluated our approach for the detection of building rooftops in aerial images and other polygonal object categories from ImageNet. On these datasets, our approach achieved a significant improvement over the baselines due to its ability to leverage non-additive scoring functions that go beyond local measures of shape and appearance, and to consider multiple overlapping, hypotheses. Interesting avenues of future work include a broader exploration of cycle scoring functions and the use of alternative graph-cycle optimization strategies.

References

1. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour Detection and Hierarchical Image Segmentation. PAMI 33(5), 898–916 (2011)
2. Ballard, D.H.: Generalizing the Hough Transform to Detect Arbitrary Shapes. PR 13(2), 111–122 (1981)
3. Barnes, N., Loy, G., Shaw, D.: The Regular Polygon Detector. PR 43(3), 592–602 (2010)
4. Belongie, S., Malik, J., Puzicha, J.: Shape Matching and Object Recognition Using Shape Contexts. PAMI 24(24), 509–522 (2002)

5. Bignone, F., Henricsson, O., Fua, P., Stricker, M.: Automatic Extraction of Generic House Roofs from High Resolution Aerial Imagery. In: Buxton, B.F., Cipolla, R. (eds.) ECCV 1996. LNCS, vol. 1064, Springer, Heidelberg (1996)
6. Carreira, J., Sminchisescu, C.: Constrained Parametric Min-Cuts for Automatic Object Segmentation. In: CVPR 2010 (2010)
7. Chui, H., Rangarajan, A.: A New Point Matching Algorithm for Non-Rigid Registration. CVIU 89(2-3), 114–141 (2003)
8. Cootes, T., Taylor, C., Cooper, D., Graham, J.: Active Shape Models: Their Training and Application. CVIU 61(1), 38–59 (2005)
9. Cox, I.J., Rehg, J.M., Hingorani, S.L.: A Bayesian Multiple Hypothesis Approach to Contour Segmentation. IJCV 11, 5–24 (1993)
10. Crevier, D.: A Probabilistic Method for Extracting Chains of Collinear Segments. Image and Vision Computing (1999)
11. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: CVPR 2005 (2005)
12. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A Large-Scale Hierarchical Image Database. In: CVPR 2009 (2009)
13. Dickson, W.: Feature Grouping in a Hierarchical Probabilistic Network. Image and Vision Computing (1991)
14. Diestel, R.: Graph Theory. Springer (2005)
15. Elder, J., Zucker, S.: Computing Contour Closure. In: Buxton, B.F., Cipolla, R. (eds.) ECCV 1996. LNCS, vol. 1064, Springer, Heidelberg (1996)
16. Elder, J.H., Krupnik, A., Johnston, L.A.: Contour Grouping with Prior Models. PAMI 25(25), 661–674 (2003)
17. Endres, I., Hoiem, D.: Category Independent Object Proposals. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 575–588. Springer, Heidelberg (2010)
18. Felzenszwalb, P., Mcallester, D., Ramanan, D.: A Discriminatively Trained, Multiscale, Deformable Part Model. In: CVPR 2008 (2008)
19. Ferrari, V., Jurie, F., Schmid, C.: From Images to Shape Models for Object Detection. IJCV 87, 284–303 (2010)
20. Guy, G., Medioni, G.: Inferring Global Perceptual Contours from Local Features. IJCV 20(1/2), 113–133 (1996)
21. Huertas, A., Cole, W., Nevatia, R.: Detecting Runways in Complex Airport Scenes. CVGIP 51(2) (1990)
22. Izadi, M., Saeedi, P.: Three-Dimensional Polygonal Building Model Estimation from Single Satellite Images. IEEE Trans. Geosci. Remote Sens. (2012)
23. Johnson, D.B.: Finding All the Elementary Circuits of a Directed Graph. SIAM 4(1) (1975)
24. Jung, C.R., Schramm, R.: Rectangle Detection Based on a Windowed Hough Transform. In: CGIP 2004 (2004)
25. Jurie, F., Schmid, C.: Scale-Invariant Shape Features for Recognition of Object Categories. In: CVPR 2004 (2004)
26. Kumar, M.P., Koller, D.: Efficiently Selecting Regions for Scene Understanding. In: CVPR 2010 (2010)
27. Lampert, C., Blaschko, M., Hofmann, T.: Beyond Sliding Windows: Object Localization by Efficient Subwindow Search. In: CVPR 2008 (2008)
28. Loy, G., Zelinsky, A.: Fast Radial Symmetry for Detecting Points of Interest. PAMI 25(8), 959–973 (2003)
29. Mahamud, S., Williams, L.R., Thornber, K.K., Xu, K.: Segmentation of Multiple Salient Closed Contours from Real Images. PAMI (2003)

30. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust Wide-Baseline Stereo from Maximally Stable Extremal Regions. IVC 22(10), 761–767 (2004)
31. Mayer, H.: Automatic Object Extraction from Aerial Imagery, a Survey Focusing on Buildings. CVIU 74(2), 138–149 (1999)
32. Opelt, A., Pinz, A., Zisserman, A.: A Boundary-Fragment-Model for Object Detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 575–588. Springer, Heidelberg (2006)
33. Parent, P., Zucker, S.W.: Trace Inference, Curvature Consistency, and Curve Detection. PAMI 11(8) (1989)
34. Perona, P., Freeman, W.T.: A Factorization Approach to Grouping. In: Burkhardt, H.-J., Neumann, B. (eds.) ECCV 1998. LNCS, vol. 1406, pp. 655–670. Springer, Heidelberg (1998)
35. Roberts, L.: Machine Perception of Three-Dimensional Solids. Ph.D. thesis (1965)
36. Robles-Kelly, A., Hancock, E.R.: A Probabilistic Spectral Framework for Grouping and Segmentation. PR 37(7), 1387–1405 (2004)
37. Russakovsky, O., Ng, A.Y.: A Steiner Tree Approach to Object Detection. In: CVPR 2010 (2010)
38. Sarkar, S., Boyer, K.L.: Quantitative Measures of Change Based on Feature Organisation: Eigenvalues and Eigenvectors. CVIU 71(1), 110–136 (1998)
39. Shashua, A., Ullman, S.: Structural Saliency: the Detection of Globally Salient Structures Using a Locally Connected Network. In: ICCV 1988 (1988)
40. Shotton, J., Blake, A., Cipolla, R.: Contour-Based Learning for Object Detection. In: ICCV 2005 (2005)
41. Sivic, J., Russell, B., Efros, A., Zisserman, A., Freeman, W.: Discovering Objects and Their Location in Images. In: ICCV 2005 (2005)
42. Sun, X., Christoudias, M., Lepetit, V., Fua, P.: Real-Time Landing Place Assessment in Man-Made Environments. MVA (2013)
43. Uijlings, J.R.R., van de Sande, K.E.A., Gevers, T., Smeulders, A.W.M.: Selective Search for Object Recognition. IJCV 104(2), 154–171 (2013)
44. Vijayanarasimhan, S., Grauman, K.: Efficient Region Search for Object Detection. In: CVPR 2011 (2011)
45. Wang, S., Kubota, T., Siskind, J.M., Wang, J.: Salient Closed Boundary Extraction with Ratio Contour. PAMI 27(4) (2005)
46. Yeh, T., Lee, J., Darrell, T.: Fast Concurrent Object Localization and Recognition. In: CVPR 2009 (2009)
47. Zhang, Z., Cao, Y., Salvi, D., Oliver, K., Waggoner, J., Wang, S.: Free-Shape Subwindow Search for Object Localization. In: CVPR 2010 (2010)
48. Zhang, Z., Fidler, S., Waggoner, J., Dickinson, S., Siskind, J.M., Wang, S.: Supderedge Grouping for Object Localization by Combining Appearance and Shape Information. In: CVPR 2012 (2012)