CPMC: Automatic Object Segmentation Using Constrained Parametric Min-Cuts

João Carreira and Cristian Sminchisescu

Abstract—We present a novel framework to generate and rank plausible hypotheses for the spatial extent of objects in images using bottom-up computational processes and mid-level selection cues. The object hypotheses are represented as figure-ground segmentations, and are extracted automatically, without prior knowledge of the properties of individual object classes, by solving a sequence of constrained parametric min-cut problems (CPMC) on a regular image grid. In a subsequent step, we learn to rank the corresponding segments by training a continuous model to predict how likely they are to exhibit real world regularities (expressed as putative overlap with ground truth) based on their mid-level region properties, then diversify the estimated overlap score using maximum marginal relevance measures. We show that this algorithm significantly outperforms the state of the art for low-level segmentation in the VOC 2009 and 2010 datasets. In our companion papers [1], [2], we show that the algorithm can be used, successfully, in a segmentation-based visual object category recognition pipeline. This architecture ranked first in the VOC2009 and VOC2010 image segmentation and labeling challenges.

Index Terms—Image Segmentation, figure-ground segmentation, learning

1 Introduction

Reliably identifying the spatial extent of objects in images is important for high-level vision tasks like object recognition. A region that covers an object fully provides a characteristic spatial scale for feature extraction, isolates the object from potentially confusing background signals and allows for information to be propagated from parts of the object to the whole. For example, a region covering a human fully makes it possible to propagate the person identity from the easier to identify face area to the rest of the body.

Given an image, the space of all possible regions, or segments that can be obtained, is exponentially large. However, in our perceived visual world not all image regions are equally likely to arise from the projection of a three-dimensional object. Objects are usually compact and this results in their projection in the image being connected; it is also common for strong contrast edges to mark objects boundaries. Such properties reduce the number of plausible object regions greatly, but may not be sufficient to unambiguously identify the optimal spatial support for each of the objects in an image.

In this paper, we follow a two step strategy by combining a figure-ground, multiple hypothesis bottom-up approach to segmentation with subsequent verification and ranking based on mid-level region properties. Key to an effective solution is the capability to leverage the statistics of real-world objects in the selection process. One possibility would be to learn the parameters of the segmentation algorithm directly, by training a machine learning model using large amounts of human annotated data. However, the local scope of dependencies and the intrinsically combinatorial nature of image segmentation diminishes the effectiveness of learning in such 'pixel spaces' as many interesting features such as the convexity and the smoothness of a region boundary are difficult to capture locally. On the other hand, once sufficient image support is available, learning to distinguish 'good' segments that represent plausible projections of real-world surfaces, from accidental image partitions becomes in principle feasible. This motivates our novel decomposition of the problem into two stages. In the first stage, we explore the space of regions that can inferred from local measurements, using cues such as good alignment with image edges. The process of enumerating regions with plausible alignment with the image contours is performed using exact combinatorial methods based on parametric max-flow. Then, in the restricted space of generated regions, we use a learned combination of advanced mid-level features in order to induce a more accurate global ranking of those regions in terms of their probability to exhibit 'object-like' regularities.

A key question, and one of our contributions, is how should image partitions be generated. Should region hypotheses be allowed to overlap with each other? Should one aim at multi-region image segmentations early? We argue that segmentation is already a sufficiently challenging problem without such constraints. It may therefore be more appropriate to enforce global inter-region spatial consistency at a later stage of processing, by higher-level routines that have access to a sufficient spatial support for this calculation. We argue that attempts to enforce complex multi-region consistency constraints early may disallow the speculative behavior necessary for sampling regions effectively, given the inherently ambiguous nature of the low-level cues one typically operates on initially. Hence, differently from most of the existing approaches to segmentation, we derive methods to generate *several independent figure-ground*

- *João Carreira is with the Faculty of Mathematics and Natural Sciences, University of Bonn. E-mail: carreira@ins.uni-bonn.de.*
- *Cristian Sminchisescu is with the Faculty of Mathematics and Natural Sciences, University of Bonn and the Institute of Mathematics (IMAR). E-mail: cristian.sminchisescu@ins.uni-bonn.de. (Corresponding author)*

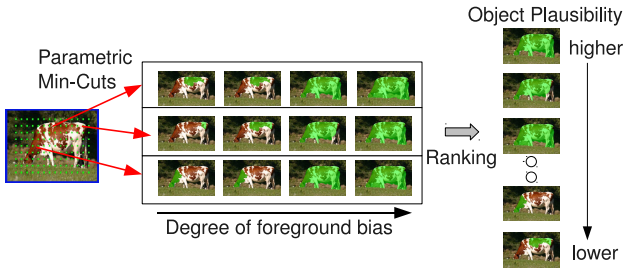


Fig. 1: Our object segmentation framework. Segments are extracted around regularly placed foreground seeds, with various background seeds corresponding to image boundary edges, for all levels of foreground bias, which has the effect of producing segments at different locations and spatial scales. The resulting set of segments is ranked according to their plausibility of being good object hypotheses, based on mid-level properties. Ranking involves first removing duplicates, then diversifying the segment overlap scores using maximum marginal relevance measures.

partitions, rather than a battery of splits of each image into multiple, non-overlapping regions[1].

Our proposed framework is depicted in fig. 1. We first solve a large number of independent binary min-cut problems on an image grid, at multiple scales. These are designed as energy functions efficiently solvable with parametric min-cut/max-flow techniques. The resulting pool of segments is minimally filtered to remove trivial solutions and ranked using a regressor trained to predict to what extent the segments exhibit the regularities typical of real-world objects, based on their low and mid-level region properties. Because standard ranking tends to place redundant instances of a same segment in close-by positions, we diversify the resulting set using Maximal Marginal Relevance measures and retain only the top ranked segments.

The quality of the list of object hypotheses returned by our algorithm is evaluated empirically by measuring how accurate they are with respect to pixel-level ground truth human annotations, in object recognition datasets. We also record performance as a function of the number of segments. Results are reported on several publicly available benchmarks: MSRC [5], the Weizmann Segmentation Database [6] and both VOC2009 and VOC2010 [7], [8] where the proposed method is shown to significantly outperform the state of the art, while at the same time using significantly fewer segments.

Several visual analysis methods may benefit from outputs like the ones provided by our algorithm. Object detectors usually scan a large number of bounding boxes in sliding window schemes [9], [10] without considering the plausibility of pixel grouping within each. Semantic segmentation algorithms [11], [12], [13], [14] incorporate the outputs of these object detectors, and may need to mediate the transition between the rectangular regions produced by the detector and the desired free-form regions that align with object boundaries. Unsupervised object discovery [15] also requires good class-independent object proposals. While the presentation focuses on the problem of object segmentation, the proposed method is general and can rank lists of segments that exhibit the statistics of non-object, 'stuff' regions such as grass or sky, as long as appropriate ground truth training data is provided.

An implementation of the proposed algorithm is made publicly available via our website [16].

Paper Organization: Section §2 reviews the related literature, §3 introduces the methodology used to generate an initial pool of segments for an image and §4 presents the segment ranking procedure. Section §5 presents experimental results and shows comparisons with the state of the art. An extension of the basic algorithm to include bounding box constraints and the corresponding results are described in §5.3. We conclude and discuss ideas for future work in §6.

2 Related Work

One of the first image segmentation approaches, published more than 40 years ago by Muerle and Allen [17], aimed to compute 'object' regions. Small patches having similar gray-level statistics were iteratively merged, starting at a seed patch. Region growing stopped when none of the neighboring candidate patches was sufficiently similar to the current region. The process was repeated until all pixels were assigned. This method took advantage of the fundamental grouping heuristic that neighboring pixels with different color are more likely to belong to different objects. However it produced very local solutions and was not able to deal with textured regions, and even less, take advantage of more sophisticated object statistics. Later, more accurate techniques emerged—good surveys can be found in [18], [19], [20]. However, most methods still pursued a single optimal segmentation of an image into a set of non-overlapping regions that covered it fully (a multi-region image partitioning). But a sufficiently good partitioning is not easy to obtain given the ambiguity of low and mid-level cues. Moreover, there were no quantitative benchmarks to gauge progress and most papers only described the merits of the output segmentations qualitatively, usually based on results obtained on a few images.

As a consequence, in the nineties, part of the recognition community lost confidence that a reliable segmentation procedure would be found and began investigating solutions that avoided bottom-up segmentation entirely [21]. This trend led to the current prevalence of bounding box detectors operating on sliding windows [9], [22]. These detectors rely on a dense evaluation of classifiers in overlapping rectangular image regions, with consistency usually enforced a posteriori by non-maxima suppression operations. Sliding window methods are effective in localizing certain objects like faces or motorbikes, but do not obviously generalize to more complex structures and cannot be easily adapted for general 3d scene understanding: *e.g.* information predicted on rectangular image regions is not sufficient for tasks such as vision-based manipulation

1. The algorithm proposed in this paper has been recently employed to generate multi-region, full image segmentations, by sampling high-scoring sets of non-overlapping figure-ground segmentations, modeled as maximal cliques, with competitive results [3], [4].

of a cup by a robot, where it is critical to precisely identify the cup handle in order to grasp it.

Such considerations made a revival of segmentation inevitable. The trend has gained momentum during the past ten years, propelled by the creation of annotated benchmarks [7], [23] and new segmentation performance metrics [7], [24]. A second important factor was the adoption of machine learning techniques to optimize performance on benchmarks. A third factor was relaxing the constraint of working with a single partitioning. A popular approach emerged by computing several independent segmentations, possibly using different algorithms. This idea was pursued by Hoiem *et al*. [25] for geometric labeling problems. Russel *et al*. [15] computed normalized cuts for different number of segments and image sizes in the context of unsupervised object discovery. By generating tens to hundreds of thousands of segments per image, Malisiewicz and Efros [26] produced very good quality regions for the MSRC dataset, by merging pairs and triplets of segments obtained using the Mean Shift [27], Normalized Cuts [28] and Felzenszwalb-Huttenlocher's (FH) [29] algorithms. Stein *et al*. [30] solved Normalized Cut problems for different number of segments, on a special affinity matrix derived from soft binary mattes, whereas Rabinovich *et al*. [31] shortlisted segmentations that reoccured, hence were potentially more stable.

The computation of multiple segmentations can also be organized hierarchically. Shi and Malik [28] recursively solve relaxations of a Normalized Cut cost based on graphs constructed over pixel nodes. Sharon *et al*. [32] proposed algebraic multigrid techniques to efficiently solve normalized cuts problems at multiple levels of granularity, where graphs with increasingly more complex features were used at coarser levels. Arbeláez *et al*. [33] derive a segment hierarchy by iteratively merging superpixels produced by an oriented watershed transform. They use the output of the learned globalPb boundary detector [34] and can represent the full hierarchy elegantly by a single ultrametric contour map. The hierarchy is a natural representation for segmentation, as it lends itself to compositional representations. However, inaccuracies in one level (due to incorrect merging of two regions from the previous level, for example), tend to propagate to all coarser levels. Therefore, given the same segmentation technique, generating a single hierarchy is likely to be less robust than using independent segmentations.

Differently, our region sampling methodology generates multiple independent binary hierarchies constrained at different positions in the image. Each level of the hierarchy corresponds to a partitioning into figure and ground, where only the figure region is retained, and regions at finer levels are nested inside coarser levels regions (this is a property induced by our parametric max-flow methodology [35]). In this way, we aim to better sample the space of plausible regions popping up at different image locations. We compute these partitionings using energies mostly related to the ones developed for interactive segmentation applications, where, however, computing a single figure-ground solution is typical. In these applications, max-flow algorithms are quite popular because they can obtain exact optima for certain energy minimization problems that involve region and boundary properties [36]. Generally the user assigns some pixels to the foreground and background regions manually and these constrain an energy function, which is optimized using a global minimization algorithm. The two steps are repeated until the set of manually assigned pixels constrain the solution sufficiently to make the resulting binary segmentation satisfactory. Variants requiring less manual interaction have been developed, such as GrabCut [37], where a simple rectangular seed around the object of interest is manually initialized and an observation model is iteratively fitted by expectation maximization (EM). Alternatively, Bagon *et al*. [38] require a user to simply click a point inside the object of interest, and use EM to estimate a sophisticated self-similarity energy.

Max-flow techniques can only globally optimize energies defined on local features such as contrast along the boundary and good pixel fit to a color or texture model. Interesting relaxation approaches exist for some energies whose minimization is NP-hard, such as curvature regularity of the boundary [39] and approximations have been developed for energies with connectivity priors [40]. However, many other more global properties, such as convexity or symmetry, are significantly more challenging to optimize directly. This motivates our segment generation and ranking procedure. We differ from existing methods not only in leveraging an efficient parametric max-flow methodology to solve for multiple breakpoints of the cost, thus exploring a much larger space of plausible segment hypotheses in polynomial time, but also in using regression methods on generic mid-level features, in conjunction with ranking diversification techniques, to score the generated segments. This fully automates the process of distilling a representative, yet compact segment pool. No manual interaction is necessary in our method.

One of the big challenges in segmentation is to leverage the statistics of real world images in order to obtain more coherent spatial results. Methods that learn low-level statistics have been applied to distinguish real from apparent contours [41], [42], [43] and similar from dissimilar superpixels [25]. Ren and Malik [44] use a random search algorithm to iteratively hypothesize segmentations by combining different superpixels, and use a classifier to distinguish good segmentations from bad ones. Pen and Veksler [45] learn to select the best segment among a small set generated by varying the value of one parameter, in the context of interactive segmentation. Models based on mid-level properties have also been learned to distinguish good from bad regions [44]. High-level shape statistics can be incorporated into binary segmentation models, usually as non-parametric distributions of templates [46], [47], [48]. Expressive part-based appearance models have also been developed [49], [50], [51], [52]. As objects in real images exhibit large variability in pose, have high intra-class variation and are often occluded, it is likely that such methods may require bottom-up initialization, which an algorithm like ours can provide. Effectively leveraging high-level shape priors in the initial steps of a visual processing pipeline may not always be feasible.

Our method aims to learn what distinguishes meaningful regions, covering full objects, from accidental pixel group-

ings. Since our original presentation at VOC2009 [53] and publication [54], related ideas have been pursued. Endres and Hoiem [55] follow a processing pipeline related to ours, but employ a learned affinity measure between superpixels, rather than pixels, and a structured learning approach on a maximum marginal relevance measure similar to the one we originally proposed to diversify ranking. To generate figure-ground segments, Levinshtein *et al*. [56] developed a procedure based on parametric max-flow principles similar to ours, but use a graph where new similarity measures are constructed on superpixels. In parallel work, Alexe *et al*. [57] learn a naive Bayes model to distinguish bounding boxes enclosing objects from those containing amorphous background, without knowledge of the shape and appearance of particular object classes. They also show how to sample bounding boxes from the model efficiently but do not provide segmentations. Salient object detection [58] approaches are also relevant to our work, but they focus on selection criteria inspired by attention mechanisms. We are instead interested in computing regions that cover every object in an image well, independently of whether they 'pop out' from the rest of the scene or not.

3 Constrained Parametric Min-Cuts (CPMC)

In order to generate a pool of segments with high probability of not missing regions with good object overlap, multiple constrained parametric min-cut (CPMC) problems are solved with different seeds and unary terms. This leads to a large and diverse pool of segments at multiple spatial scales. The segments corresponding to implausible solutions are subsequently discarded using simple ratio cut criteria. The remaining are clustered so that all but representative segments with low energy are retained, among those extremely similar. The final working set of segments is significantly reduced, but at the same time the most accurate segments are preserved.

3.1 Setting up the Energy Functions

For each image, alternative sets of pixels, called seeds, are hypothesized to belong to the foreground and the background. The foreground seeds are placed on a grid, whereas background seeds are associated with sets of pixels along the image border. For each combination of foreground and background seeds we compute figure-ground segmentations with multiple levels of foreground bias. The levels of bias are induced by varying the cost of assigning non-seed pixels to the foreground. Inference consists of finding minimum cuts for the different values of foreground bias — in fact searching over multiple foreground biases is intrinsic to our parametric max flow procedure. The optimization problem is formulated next.

Let $I(\mathcal{V}) \rightarrow R^3$ be an image defined on a set of pixels $\mathcal{V}$. As commonly done in graph-based segmentation algorithms, the similarity between neighboring pixels is encoded as edges of a weighted graph $G = (\mathcal{V}, \mathcal{E})$. Here, each pixel is a node in the set $\mathcal{V}$. The foreground and background partitions are represented by labels 1 and 0, respectively. Seed pixels $\mathcal{V}_f$ are constrained to the foreground and $\mathcal{V}_b$ to the background by setting infinity energy to any labeling where they receive the contrasting label. Our overall objective is to minimize an energy function over pixel labels $\{x_1, ..., x_N\}, x_i \in \{0, 1\}$, with N the total number of pixels. In particular, we optimize the following energy function:

$$E^{\lambda}(X) = \sum_{u \in \mathcal{V}} D_{\lambda}(x_u) + \sum_{(u,v) \in \mathcal{E}} V_{uv}(x_u, x_v) \qquad (1)$$

with $\lambda \in \mathbb{R}$, and unary potentials given by:

$$D_{\lambda}(x_u) = \begin{cases} 0 & \text{if } x_u = 1,\ u \notin \mathcal{V}_b \\ \infty & \text{if } x_u = 1,\ u \in \mathcal{V}_b \\ \infty & \text{if } x_u = 0,\ u \in \mathcal{V}_f \\ f(x_u) + \lambda & \text{if } x_u = 0,\ u \notin \mathcal{V}_f \end{cases} \qquad (2)$$

The foreground bias is implemented as a cost incurred by the assignment of non-seed pixels to background, and consists of a pixel-dependent value $f(x_u)$ and an uniform offset λ. Two different functions $f(x_u)$ are used in practice. The first is constant and equal to 0, resulting in a uniform (variable) foreground bias. The second function uses color. Specifically, RGB color distributions $p_f(x_u)$ on seed $\mathcal{V}_f$ and $p_b(x_u)$ on seed $\mathcal{V}_b$ are estimated to derive $f(x_u) = \ln p_f(x_u) - \ln p_b(x_u)$. The probability distribution of pixel j belonging to foreground is defined as $p_f(i) = \exp[-\gamma \cdot \min_j(||I(i) - I(j)||)]$, with γ a scaling factor, and j indexing representative pixels in the seed region, selected as centers resulting from a *k*-means algorithm (k is set to 5 in all of our experiments). The background probability is defined similarly. This choice of function is motivated by efficiency, being much faster to estimate compared to the frequently used Gaussian mixture model [37]. Color-based unary terms are more effective when the color of the object is distinctive with respect to the background, as well as when objects have thin parts. Uniform unary terms are more useful in the opposite case. The complementary effects of these two types of unary energy terms are illustrated in fig. 2.

The pairwise term V_{uv} penalizes the assignment of different labels to similar neighboring pixels:

$$V_{uv}(x_u, x_v) = \begin{cases} 0 & \text{if } x_u = x_v \\ g(u, v) & \text{if } x_u \neq x_v \end{cases} \qquad (3)$$

with similarity between adjacent pixels given by $g(u, v) = \exp\left[-\frac{\max(gPb(u), gPb(v))}{\sigma^2}\right]$. gPb returns the output of the multi-cue contour detector globalPb [34] at a pixel. The square distance is also an option we experimented with, instead of the $\max$ operation, with similar results. The *boundary sharpness* parameter σ controls the smoothness of the pairwise term.

The function defined by eq. 1 is submodular. Given a pair of foreground and background seeds and $f(x_u)$, the cost can be minimized exactly for all values of λ in the same complexity as a single max-flow problem, using a parametric solver [59]. In canonical form, parametric max-flow problems differ from their max-flow counterparts in that capacities from the source node are allowed to be linear functions of a parameter, here λ. As λ (effectively our foreground bias) varies there are at most $(N - 1)$ different cuts in the transformed graph, where N is the number of nodes, although for the graphs encountered in vision problems there are generally far fewer (see our study

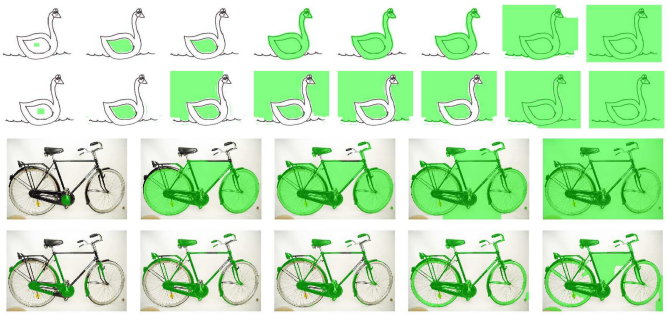


Fig. 2: Different effects of uniform and color-based unary terms. For illustration, a single foreground seed was placed manually at the same location for two energy problems, one with uniform and another with color unary terms. Shown are samples from the set of successive energy breakpoints (increasing λ values) from left to right, as computed by parametric max-flow. Uniform unary terms are used in rows 1 and 3. Color unary terms are used in even rows. Uniform unary terms are most effective in images where the background and foreground have similar color. Color unary terms are more appropriate for objects with elongated shapes.

in §3.3). The values of λ for which the cut values change are usually known as *breakpoints*. When the linear capacity functions from the source are either non-increasing or non-decreasing functions of λ, the problem is said to be monotonic. Our energy problems are monotonic because, for all unary terms, λ is multiplied by the same factor, 1. This important property implies that all cuts computed for a particular choice of source and sink seeds are nested.

In this work we use the *highest label pseudoflow* solver [60], which has complexity $O(mN\log(N))$ for image graphs with N nodes and m edges. The complexity of the CPMC procedure is thus $O(kmN\log(N))$, as we solve multiple parametric max-flow problems, for each of the k combinations of foreground and background seeds, and for different choices of $f(x_u)$. The pseudoflow implementation we used requires a set of λ parameters for which to compute cuts. For the study in §3.3, we additionally use an implementation based on Gallo *et al*. [35] in order to analyze the segmentation results produced by a push-relabel parametric max-flow solver which retrieves all breakpoints [61].

The graph construction that maps to the energy functions in (1), for each choice of foreground and background seed, augments the original problem dependency graph G with two special nodes, source s and sink t that must be in separate partitions in any binary cut [36]. The unary energy terms are encoded as edges between these special nodes and the nodes in $\mathcal{V}$.

3.2 Effect of Grid Geometry

As *foreground seeds*, we chose groups of pixels that form small solid squares. We have experimented with three different strategies to place them automatically: rectangular grid geometry, centroids of superpixels obtained with normalized cuts, and centroids of variable size regions, closest to each rectangular grid position, obtained using segments obtained by the algorithm of [29]. As shown in table 1, the performance differences are not very significant (see section §5 for details about the datasets and the evaluation criteria).

The *background seeds* are necessary in order to prevent trivial cuts that leave the background set empty. We used four different types: seeds including pixels that cover the full image boundary, just the vertical edges, just the horizontal edges and all but the bottom image edge. This selection strategy allows us to extract objects that are only partially visible, due to clipping at different image boundaries.

In practice we solve around 180 instances of problem (1) for each image, for 30 λ values each (during processing, we skip duplicate breakpoints), defined on a logarithmic scale. The set of figure-ground segmentations is further enlarged by splitting the ones with multiple connected foreground components. The final pool has up to 10,000 segments per image.

As an alternative to multiple 'hard' background seeds, it is possible to use a single 'soft' background seed. This can be a frame one pixel wide covering the border of the image, with each pixel having a finite penalty associated to its assignment to the foreground. This construction is more efficient, as it decreases the number of energy problems to solve by 75%. We used this type of background seeds in an extension of the basic algorithm, presented in section §5.3.

Seed placement	MSRC score	Weizmann score
Grid	0.85 ± 0.1	0.93 ± 0.06
NCuts	0.86 ± 0.09	0.93 ± 0.07
FH	0.87 ± 0.08	0.93 ± 0.07

TABLE 1: Effect of spatial seed distribution. The use of superpixel segmentation algorithms (*e.g.* Normalized Cuts or FH [29]) to spatially distribute the foreground seeds does not significantly improve the average covering score on the MSRC dataset, over regular seed geometries. On Weizmann, the average best F-measure is the same for all distributions, perhaps because the objects are large and any placement strategy eventually distributes some seeds inside the object.

3.3 Effect of λ Schedule

The effect of solving problem (1) for all λ values, instead of a preset logarithmic λ schedule, was evaluated on the training set of the PASCAL VOC 2010 segmentation dataset (the typical distinction into training and testing is not relevant for the purpose of this experiment, where the goal is only to analyze the number of breakpoints obtained using different search strategies). We use a 6x6 regular grid of square seeds and solve using two procedures: (1) 20 values of λ sampled on a logarithmic scale (only the distinct energy optima are recorded) and, (2) all λ values, as computed as breakpoints of (1). We have recorded the average computational time per seed, the ground truth covering score, and the number of breakpoints obtained under the two λ-search strategies. The results are shown in table 2, suggesting that a preset λ schedule is a sensible option. Using only 20 values produces almost the same covering as the one obtained using all values, it is 4 times

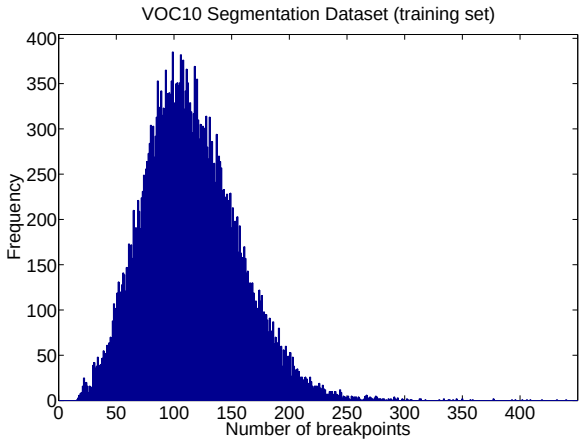


Fig. 3: Frequency of the parametric max flow breakpoints for each seed, on the training set of the VOC2010 segmentation dataset. These results were obtained using a 6x6 uniform grid of seeds. The number of breakpoints has mean 110, and a heavier tail towards a larger number of breakpoints.

faster and generates 10% of the total number of breakpoints, hence fewer segments. We also plot the distribution of the number of breakpoints per seed in fig. 3, under the same experimental conditions. The frequency of breakpoints has a dominantly unimodal (bell) shape, with mean 110, but a slightly heavier tail towards larger numbers of segments. There are never less than 15 breakpoints in this dataset.

# λ values	# breakpoints	Time (s)	Covering
20	12.3	1.8	0.713
all	114.6	7.5	0.720

# objects	1-2	3-4	5-6	7-13
# breakpoints all λ	112.19	124.60	125.29	142.83
# breakpoints 20 λ	12.27	12.64	13.08	13.45
# images	717	147	68	32

TABLE 2: Covering results obtained on the training set of VOC2010, based on a 6x6 grid of uniform seeds. The table compares the results of solving CPMC problems for 20 values of λ, sampled on a logarithmic scale, with the results obtained by solving for all possible values of λ. Shown are the average number of breakpoints per seed, and the average time required to compute the solutions for each seed. Computing all breakpoints for each seed provides modest ground truth covering improvements, at the cost of generating a larger number of segments and an increased computation time. The second table shows that images containing a larger number of ground truth objects tend to generate more breakpoints per seed.

3.4 Fast Segment Rejection

Generating a large set of segments increases the hit rate of the algorithm, but many segments are redundant or do not obey the statistics of real-world surfaces imaged by a camera. For images with large homogeneous regions, the original hypothesis generation step can also produce many copies of the same segment because of the seeding strategy — every seed placed inside the region would tend to generate the same segment for the same λ. Moreover, sometimes visually arbitrary segments are created, as artifacts of the foreground bias strength and the seed constraints employed.

We deal with these problems using a fast rejection step. We first filter very small segments (up to 150 pixels in our implementation), then sort the segments using a simple criterion (we have used the ratio cut [62] as this is scale invariant and very selective) and retain up to 2,000 of the highest scoring segments. Then we hierarchically cluster the segments using overlap as a similarity measure, to form groups with all segments of at least 0.95 spatial overlap. For each cluster, we retain the segment with the lowest energy.

The number of segments that pass the fast rejection step is usually small, being indicative of how simple or cluttered the structure of an image is. In general, simple datasets have lower average number of segments. But even in the difficult PASCAL VOC 2009 dataset, the average was 154.

4 Mid-level Segment Ranking

Gestalt theorists [63], [64] argued that properties such as proximity, similarity, symmetry and good continuation are key to visual grouping. One approach would be to model such properties in the segmentation process, as long-range dependencies in a random field model [65], [66]. However, this poses significant modeling and computational challenges. With a segment set generated using weaker constraints, leveraging Gestalt properties becomes easier: rather than guide a complex inference procedure based on higher-order, long-range dependencies, we only need to check conformance with Gestalt regularities. It is therefore interesting to explore how the qualitative Gestalt theories can be implemented in such a framework and what effects they produce in practice. An important question is whether Gestalt properties can be used to predict if segments have regularities typical of projections of real objects, without leveraging prior knowledge about the classes of objects present in the image. This is a potentially challenging decision problem, since the visual aspects of objects are extremely diverse. However, if object regularities can be identified, images could be represented by a handful of segments, which are easier to interpret and process by higher-level visual routines than a large set of pixels or superpixels.

In this work, we take an empirical approach: we compile a large set of features and annotated examples of segments of many objects from different categories, and use machine learning techniques to uncover their significance. Three sets of features (34 in total) are considered to describe each segment, representing graph, region and Gestalt properties. Graph properties, in particular variations of cut values, have long been used as cost functions in optimization methods for segmentation. Region properties encode mainly the statistics of where and at what scale objects tend to appear in images. Finally, Gestalt properties include mid-level cues like convexity and continuity, which can encode object regularities (*e.g.* objects background segments are usually non-convex and object boundaries are usually smoother than the boundaries

of accidental segments).

Graph partition properties (8 features) include the *cut* (sum of affinities along the segment boundary) [67], the *ratio cut* (sum of affinity along the boundary divided by their number) [62], the *normalized cut* (ratio of cut and affinity inside foreground, plus ratio of cut and affinity on background) [28], the *unbalanced normalized cut* (cut divided by affinity inside foreground) [32], and the *boundary fraction of low cut*, 4 binary variables signaling if the fraction of the cut is larger than a threshold, normalized by segment perimeter, for different thresholds.

Region properties (18 features) include area, perimeter, relative coordinates of the region centroid in the image, bounding box location and dimensions, major and minor axis lengths of the ellipse having the same normalized second central moments as the region, eccentricity, orientation, convex area, Euler number, diameter of a circle with the same area as the region, ratio of pixels in the region to pixels in the total bounding box, perimeter and absolute distance to the center of the image. Some of these features can be easily computed in Matlab using the *regionprops* function.

Gestalt properties (8 features) are implemented mainly as normalized histogram distances based on the χ^2 comparison metric: $\chi^2(x,y) = \sum_i \frac{(x_i - y_i)^2}{x_i + y_i}$ [68]. Let the texton histogram vector on the foreground region be t_f, and the one on the background be t_b. Then *inter-region texton similarity* is computed as the $\chi^2(t_f, t_b)$. *Intra-region texton similarity* is computed as $\sum_i \mathbf{1}(t_f(i) > k)$, with $\mathbf{1}$ the indicator function, and k a threshold, set to 0.3% the area of the foreground in our implementation. The textons are obtained using the globalPb implementation [33], which uses 65 nearest neighbor codewords.

Another two features we use are *inter-region brightness similarity*, defined as $\chi^2(b_f, b_b)$, with b_f and b_b intensity histograms with 256 bins, and *intra-region brightness similarity* defined as $\sum_i \mathbf{1}(b_f(i) > 0)$.

We also extract the *intra-region contour energy* as the sum of edge energy inside the foreground region, computed using globalPb, normalized by the length of the region perimeter. We also extract an *inter-region contour energy*, as the sum of edge energies along the boundary normalized by the perimeter.

Other Gestalt features we consider include *curvilinear continuity* and *convexity*. The first is the integral of the segment boundary curvature. We use an angle approximation to the curvature [69] on triplets of points sampled regularly (every 15 pixels in our tests). Convexity is measured as the ratio of areas of the foreground region and its convex hull.

All features are normalized by subtracting their mean and dividing by their standard deviation.

4.1 Learning

The objective of our ranking process is to identify segments that exhibit object-like regularities and discard most others. One quality measure for a set of segments with respect to the ground truth is **covering** [33]. Let S be the set of ground truth segments for an image, S' be the set of machine segments and $S'(r)$ the subset of machine segments at rank r or higher. Then, the covering of S by $S'(r)$ can be defined as:

$$C(S, S'(r)) = \frac{1}{N} \sum_{R \in S} |R| * \max_{R' \in S'(r)} O(R, R') \qquad (4)$$

where N is the total number of pixels in annotated objects in the image, $|R|$ is the number of pixels in the ground truth segment R, and O is a similarity measure between two regions.

We cast the problem of ranking the figure-ground hypotheses as regression on $\max_{R \in S} O(R, R')$, the maximum similarity a segment has with a ground truth object, against the segment features. The idea is that if regression is accurate, the generated segments most similar to each ground truth will be placed at high ranks. Then many lower ranked segments can be discarded without reducing the covering measure. As similarity measure O we use **overlap** [7]:

$$O(S, G) = \frac{|S \cap G|}{|S \cup G|} \qquad (5)$$

which penalizes both under-segmentations and over-segmentations and is scale invariant. An alternative to overlap, which we used in one of our experiments, is the **F-measure** [6]:

$$F = \frac{2RP}{P + R} \qquad (6)$$

where P and R are the precision and recall of pixels in a machine segment relative to a ground truth segment.

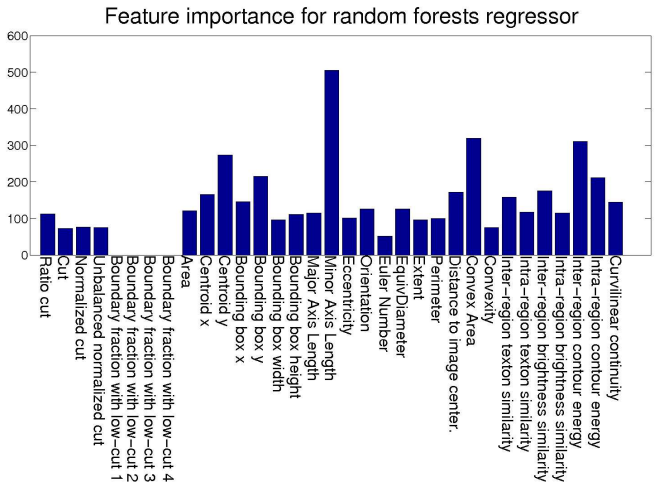


Fig. 4: Feature importance for the random forests regressor learned on the VOC2009 segmentation training set. The minor axis of the ellipse having the same normalized second central moments as the segment (here 'Minor Axis Length') is, perhaps surprisingly, the most important. This feature used in isolation results in relatively poor rankings however (see fig. 5a). The Graph properties have small importance. The 'Boundary fraction of low cut' features, being binary, do not contribute at all. Gestalt features have above average importance, particularly the contour energies.

For ranking, we experimented with both linear regression and random forests [70], a competitive non-linear model that averages over multiple regression trees. We used a random forests implementation available online [71] and used default parameters, except for the number of trees, 200, and the number of candidate variables to select from, at random, at each split node, which we set to 10.

The *importance* of our features as learned by the random forests regressor [70], is shown in fig. 4. Some region properties appear to be quite informative, particularly features such as segment width and height and the location in the image. The 'Minor Axis Length' feature, which gets the highest importance works quite poorly in isolation, however (as illustrated in fig. 5a), suggesting that some cues are only effective in conjunction with other features. Convexity and the edge energy along the boundary are also assigned large importance, as expected.

4.2 Maximum Marginal Relevance Diversification

Applying standard regression for ranking does not come without issues. Similar segments have similar features, which causes them to regress to the same values and be ranked in adjacent positions. The covering measure only considers the best overlap with each ground truth object, hence redundant segments in adjacent positions do not increase covering and tend to lower the ranks of segments that best overlap other objects. More segments then need to be retained to achieve the same score.

An effective way to deal with such effects is to **diversify** the ranking, in order to prevent that minor variations of a segment saturate the pool. We achieve this based on Maximal Marginal Relevance (MMR) measures [72]. To our knowledge this is the first application of this technique to image segmentation. Starting with the originally top-scored segment, the MMR induces an ordering where the next selected segment (with maximum marginal relevance) is the one maximizing the original score minus a redundancy measure with respect to segments already selected. This procedure is iterated until all segments have been re-ranked. The redundancy measure we employ is the overlap with the set of previously selected segments based on the MMR measure.

Formally, let H be the full set of figure-ground segmentations and $H_p \subset H$ hypotheses already selected. Let $s(H_i)$ be our predicted score for a given figure-ground segmentation and $o(H_i, H_j)$ the overlap between two figure-ground segmentations. The recursive definition for the next maximal marginal relevance selection [72] is given as:

$$MMR = \underset{H_i \in H \setminus H_p}{\operatorname{argmax}} \left[\theta \cdot s(H_i) - (1-\theta) \cdot \max_{H_j \in H_p} o(H_i, H_j)) \right]$$

The first term is the score and the second is the redundancy. Parameter θ regulates the trade-off between the predicted score and the diversity measures in the first N selections. For example with $\theta = 0$ the ranking will ignore individual scores, and select the next element in the set, having minimal overlap with any of the previously chosen elements. In contrast, with $\theta = 1$ the element with the highest score will always be selected next. The best trade-off depends on the application. If high precision is desired then a higher weight should be given to the predicted score, whereas if recall is more important, then a higher weight should be given to diversity. If θ is very small, then ranking will be close to random. For our VOC experiments we have cross-validated at $\theta = 0.75$.

5 Experiments

We study both the quality of the pool of object hypotheses generated by CPMC and the loss in quality incurred by selecting the topmost N object hypotheses, as opposed to the use of a much larger pool. We experiment with three publicly available datasets: Weizmann's Segmentation Evaluation Database [6], MSRC [5] and the VOC2009 train and validation sets for the object-class segmentation problem [7].

Weizmann consists of 100 gray-valued images having a single prominent foreground object. The goal is to generate coverage of the entire spatial support of the object in the image using a single segment, and as accurately as possible. We compare the performance of CPMC with published results from two state of the art segmentation algorithms. The results are reported using the **average best F-measure** criterion. For each ground truth object the most similar segment with respect to F-measure (eq. 6) is selected and the value of the similarity is recorded. These top similarities are then averaged.

The MSRC dataset is quite different, featuring 23 different classes, including some 'stuff' classes, such as water and grass. It has up to 11 objects present in each of its nearly 600 images. We use this dataset to evaluate the quality of the pool of segments generated, not the individual rankings.

The VOC 2009 dataset is challenging for segmentation, as it contains real-world images from Flickr, with 20 different classes of objects. The background regions are not annotated. In both MSRC and VOC2009, which contain multiple ground-truth objects per image we use the **covering** (eq. 4) with **overlap** (eq. 5) as a segment similarity measure.

5.1 Segment Pool Quality

The automatic results obtained using CPMC on the Weizmann dataset are shown in table 3a together with the previous best result, by Bagon et al [38], which additionally requires the user to click a point inside the object. We also compare with the method of Alpert *et al*. [6], which is automatic. Results for CMPC were obtained using an average of 53 segments per image. Visibly, it generates an accurate pool of segments. Results on MSRC and VOC2009 are compared in table 3b to Arbeláez *et al*. [33], which is arguably one of the state of the art methods for low-level segmentation. The methodology of the authors was followed, and we report average coverings. We use all the unique segments in the hierarchy returned by their algorithm [33] to compute the score. The pool of segments produced by CPMC appears to be significantly more accurate and has an order of magnitude fewer segment hypotheses. A filtering procedure could be used for gPb-owt-ucm to reduce the number segments, but at a potential penalty in quality. The relation between the quality of segments and the size of the ground truth objects is shown in fig. 7.

Weizmann	F-measure
CPMC	0.93 ± 0.009
Bagon *et al*. [38]	0.87 ± 0.010
Alpert *et al.* [6]	0.86 ± 0.012

(a) Average best F-measure scores over the entire Weizmann dataset. Bagon's algorithm produces a single figure-ground segmentation but requires a user to click inside the object. Alpert's results were obtained automatically by partitioning the image into one full image segmentation typically having between 2 and 10 regions. The table shows that for each image, among the pool of segment hypotheses produced by CPMC, there is usually one segment which is extremely accurate. The average number of segments that passed our fast rejection step was 53 in this dataset.

MSRC	Covering	N Segments
CPMC	0.85 ± 0.1	57
gPb-owt-ucm [20]	0.78 ± 0.15	670

VOC2009	Covering	N Segments
CPMC	0.78 ± 0.18	154
gPb-owt-ucm [20]	0.61 ± 0.20	1286

(b) Average of covering scores on MSRC and VOC2009 train+validation datasets, compared to Arbeláez *et al*. [33], here gPb-owt-ucm. Scores show the covering of ground truth by segments produced using each algorithm. CPMC results before ranking are shown, to evaluate the quality of the pool of segments from various methods.

TABLE 3: CPMC segment quality on multiple datasets.

5.2 Ranking Object Hypotheses

We evaluate the quality of our ranking method on both the validation set of the VOC2009 segmentation dataset, and on hold-out sets from the Weizmann Segmentation Database. The training set of VOC2009 consists of 750 images, resulting in $114,000$ training examples, one for each segment passing the fast rejection step. On the Weizmann Segmentation Database we randomly select 50 images, resulting in $2,500$ training examples, and we test on the remaining 50 images.

We plot curves showing how well the ground truth for each image is covered on average, as a function of the number of segments we retain per image. The segments are added to the retained list in the order of their ranking.

The curve marked as 'upper bound' describes the maximum quality measure possible given the generated segments, which can be obtained if the segments are ranked by their known overlap with ground truth. Note that on Weizmann the upper bound is flat because each image has one single ground truth object, whereas on VOC images there can be multiple objects, hence the upper bound increases as more than one segment is considered per image (on the horizontal axis). The curve labeled as 'random' is based on randomly ranked segments. It is a baseline upon which the ranking operation should improve in order to be useful.

On Weizmann we compare a random forests regressor trained on the images in that dataset with a predictor trained on VOC2009. The results in fig. 5a are similar, showing that the model is not overfitting to the statistics of the individual datasets. This also shows that it is possible to learn to rank segments of arbitrary objects, using training regions from only 20 classes. The learned models are significantly better than ranking based on the value of any single feature such as the cut or the ratio cut. On VOC2009 we have also run experiments where we have complemented the initial feature set with additional appearance and shape features — a bag of dense gray-level SIFT [73] features computed on the foreground mask, a bag of local shape contexts [74] computed on its boundary, and a HOG pyramid [75] with 3 levels computed on the bounding box fitted on the boundary of the segment, for a total of 1,054 features. In this case, we trained a linear regressor for ranking (this is significantly faster than random forests, which takes about 8 hours to train for the model with 34 features). The results are shown in fig. 5b. Clearly the new features help somewhat, producing results that are slightly better than the ones obtained by the linear regressor on the basic feature set. We will revisit them in §5.3. However, these are not better than a random forests model trained on the basic feature set. This shows that the set of basic features is already quite expressive in conjunction with nonlinear models.

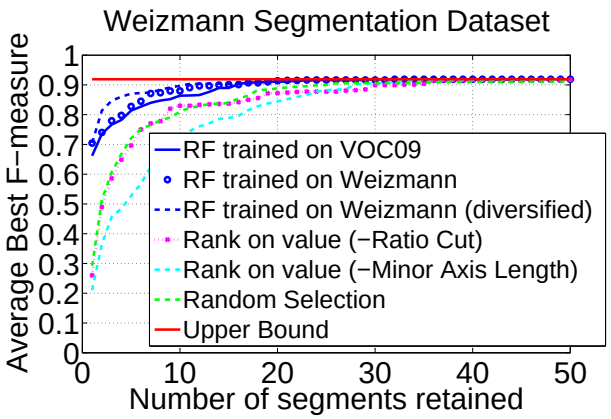
Notice that by using this ranking procedure, followed by diversification, we can obtain more accurate object hypotheses than those provided by the best existing segmentation algorithm of [33]. In fact, by using the top 7 segments produced by our ranking procedure, we obtain the same covering, 0.61, as obtained using the full hierarchy of $1,286$ distinct segments in [33].

5.3 Subframe-CPMC Extension

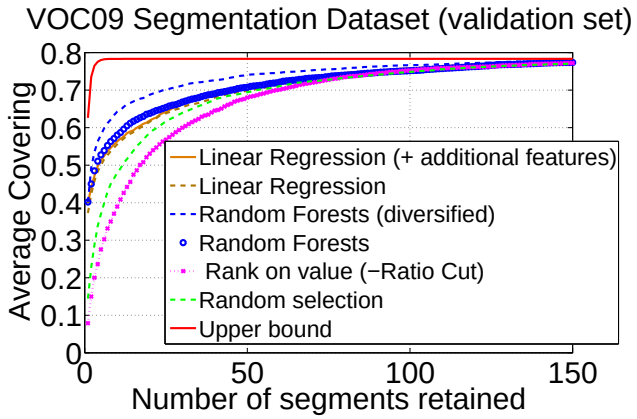
We have experimented with a different variant of the algorithm, the Subframe-CPMC, on the Pascal VOC2010 dataset. The goal was to achieve high object recall while at the same time preserve segmentation quality, with a mindset towards detection applications. To score a detection hypothesis as correct, benchmarks such as the Pascal VOC require a minimum overlap between a correctly classified region and the ground truth. In addition, benchmarks disregard the area of the ground truth regions (*e.g.* an object with 500 pixels is just as important as one occupying the full image), hence what matters is not so much achieving high *covering* scores (which explicitly take into account the size of the segments), but high *overlap*.

Subframe-CPMC uses an additional type of seed, and is configured to generate a larger number of segments. First we make the overall process faster by solving the energy problems at half the image resolution. Quantitative results were equivalent. We also changed the seeding strategy to use a single soft background seed and increased the number of foreground seeds, by using a grid of 6x6 instead of the previous 5x5. We reduced the value of the σ parameter by 30% in eq. 3, resulting in more segments due to reduced affinities between neighboring pixels.

We have also complemented the existing seeds with *subframes*, background seeds composed of the outside of rectangles covering no more than 25% of the area in the image, with a single square foreground seed in the center. These seeds constrain segments to smaller regions in the image, as they force the possible contours to lie inside the rectangular region. This is especially helpful for segmenting small objects in cluttered regions, as can be seen in fig. 7. For this type of



(a) Average best segment F-measure as we vary the number of retained segments given by our ranking procedure. Results were averaged over three different splits of 50 training and 50 testing images. Note that when working with our top-scored 5 segments per image, the results already equal the ones obtained by the interactive method of Bagon *et al*. [38]. Note also that using this learned ranking procedure, it is possible to compress the original pool of segments to a fifth (10 segments), at negligible loss of quality.



(b) Complementing the basic descriptor set with additional appearance and shape features improves the ranking slightly, but the basic set is still superior when used in conjunction with a more expressive random forests regressor. Further diversifying the ranking improves the average covering given by the first top N segments significantly.

Fig. 5: Ranking results on the Weizmann and VOC2009 datasets. Different rankers are compared with the optimal ranker ("Upper bound") and with random ranking ("Random selection").

seed we also solve problems with and without a color unary term. Two alternative types of subframe seeds were tried: a 5x5 regular grid of square subframes of fixed dimension, with width set to 40% of the image, and bounding boxes from a deformable parts detector [9], [76] with default parameters, set to the regime of high recall but low precision. For the detector, we discard class information and keep the 40 top-scored bounding boxes smaller than a threshold C, in this case 25% of the image area. Subframe energy problems are optimized efficiently by contracting all nodes corresponding to pixels belonging to background seeds into a single node, thereby reducing the size of the graph significantly.

The parameter σ, controlling the sharpness of the boundary, has an important influence on the number of generated segments. A value of 2.5 with the color-based seeds leads to 225 segments, average overlap of 0.61 and covering of 0.74, whereas for $\sigma = 1$ the method produces an average of 876 segments, average overlap of 0.69 and covering 0.76. We used $\sigma = 1$ for the uniform seeds, $\sigma = \sqrt{2}$ for the color seeds, and $\sigma = \sqrt{0.8}$ for the subframe seeds. This leads to a larger pool of segments, but also of higher quality, as noticeable in table 4.

Additional Features: Working with a larger pool of segments poses additional demands on the accuracy of ranking. An improvement we pursued was to enlarge the set of mid-level features with shape and texture descriptors. In §5.2 this was shown to improve results, but the dimensionality of these features made linear regression the most practical learning choice. A nonlinear random forests regressor on the basic feature set was still superior.

The additional shape and texture features we use are histograms, which are known to be most effective when used with certain nonlinear similarities, such as a Laplacian-RBF embedding $k(\mathbf{x}, \mathbf{y}) = \exp(-\sum |x_i - y_i|)$ [68]. Here we handle one of these similarity functions with linear regression, by first applying a randomized feature map to linearly approximate the Laplacian-RBF kernel [77], [78].

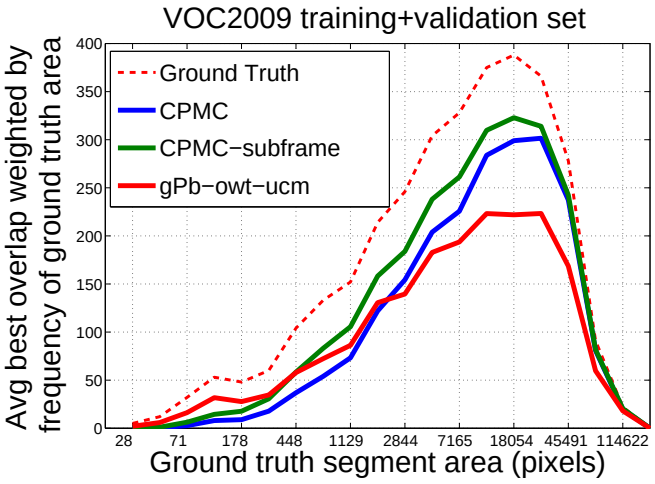


Fig. 7: Quality of the segments in the combined VOC2009 train and validation sets, as a function of the area of the ground truth segments. Object area has been discretized into 20 bins on a log scale. In the case of the ground truth curve the y-axis corresponds to the number of segments assigned in each bin (ground truth segments have an overlap value of 1 with themselves). Medium and large size objects, that are more frequent, are segmented significantly more accurately by CPMC than by gPb-owt-ucm [33]. Subframe-CPMC is competitive with gPb-owt-ucm on small objects, but generates a larger segment pool than plain CPMC (in the order of 700 instead of 150 elements).

We adjusted the extended feature set from §5.2 slightly. To represent texture we extracted two bags of words for each segment, one defined over gray-level SIFT features as

Fig. 6: Segmentation and ranking results obtained using the random forests model learned on the VOC2009 training set, with the features described in sec. §4. The green regions are the segment foreground hypotheses. The first image on each row shows the ground truth, the second and third images show the most plausible segments given by CPMC, the last two images show *the least* plausible segments, and the fourth and fifth images show segments *intermediately* placed in the ranking. The predicted segment scores are overlaid. The first three images are from the VOC2009 validation set and rows 2, 4 and 6 show the diversified rankings, with $\theta = 0.75$. Note that in the diversified ranking, segments scored nearby tend to be more dissimilar. The last three rows show results from the Weizmann Segmentation Database. The algorithm has no prior knowledge of the object classes, but on this dataset, it still shows a remarkable preference for segments with large spatial overlap with the imaged objects, yet there are neither chariots nor vases in the training set, for example. The lowest ranked object hypotheses are usually quite small reflecting perhaps the image statistics in the VOC2009 training set.

before and a new one over color SIFT features, both sampled every 4 pixels and at 4 different scales (16, 24, 36 and 54 pixels wide) to ensure a degree of scale invariance. Each feature was quantized using a 300-dimensional codebook. To represent shape we computed two pyramid HOGs, both with gradient orientation quantized into 20 bins, the first with the background segment gradients masked out on a pyramid composed of four levels, for a total of $1,700$ dimensions. The other PHOG was computed directly on the contour of the segment, with both foreground and background gradients masked out and a pyramid of three levels for a total of 420 dimensions. We map the joint vector of the two bags of words for texture features into a $2,000$-dimensional randomized feature map drawn from the Fourier transform of the Laplacian-RBF kernel [77], and process similarly the two PHOGs corresponding to shape features. We also append our original 34-dimensional feature set resulting in a total of $4,034$ features.

VOC2010 Results: The overlap measure is popular for distinguishing hits from misses in detection benchmarks. In the VOC2010 dataset we evaluate the recall under two different hit-metrics: 50% minimum segment overlap and 50% minimum bounding box overlap. Using the 50% segment overlap criterion, the algorithm obtains, on average per class, 87.73% and 83.10% recall, using 800 and 200 segments per image, respectively. Under a 50% bounding box overlap criterion, the algorithm achieves 91.90% when using 800 segments and 87.65%, for 200 segments.

The top 200 ranked segments gave on average 0.82 covering and 0.71 best overlap, which improves upon the results of CPMC without subframes on the VOC2009 (0.78 and 0.66 with all segments). These results are made possible because of the richer pools of segments, but also because the ranking is accurate. A reduction of on average around 500 segments per image results only in a loss of 0.03 average best overlap.

Details are shown in figs. 11 and 12, whereas image results are shown in fig. 9. The learned weights of the linear regressor for all features are displayed in fig.8.

Quality Measure	Grid Subframes	BB Detector	No Subframes
Overlap	0.74	0.76	0.71
Covering	0.83	0.84	0.82
N segments	736	758	602

TABLE 4: Results on the training set of the VOC2010 segmentation dataset. Color and uniform seeds are complemented with subframe seeds, either placed on a regular grid or obtained from a bounding box detector. Using a regular grid gives only slightly inferior results compared to results obtained using detector responses. Both give a large improvement in the recall of small objects, compared to models that do not use subframes. This is reflected in the overlap measure, which does not take into account the area of the segments.

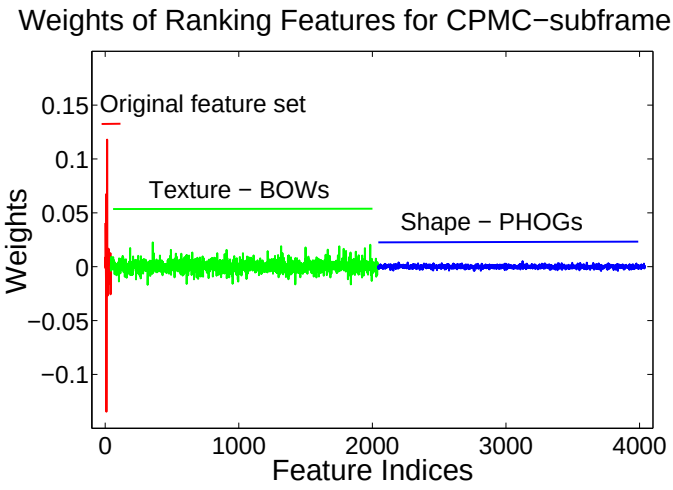


Fig. 8: Learned feature weights for the Subframe-CPMC model. The original set of mid-level features and region properties gets higher weights, texture features get intermediate weights and shape features get smaller weights. Texture features might help discard amorphous 'stuff' regions such as grass, water and sky.

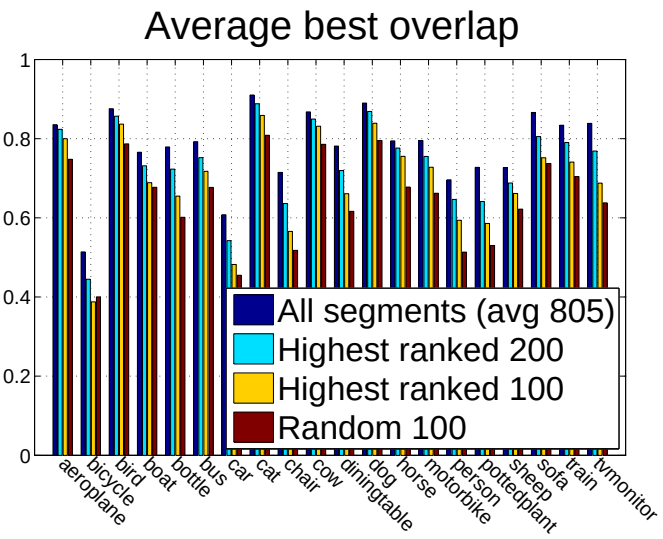


Fig. 11: Average overlap between ground truth objects and the best Subframe-CPMC segments on the validation set of VOC2010. We compare results obtained when considering all segments, just the top ranked 100 or 200 and a baseline that selects 100 segments randomly from the pool of all segments. Certain classes appear to be considerably harder to segment, such as bicycles, perhaps due to their wiry structure.

6 Conclusions

We have presented an algorithm that casts the automatic image segmentation problem as one of generating a compact set of plausible figure-ground object hypotheses. It does so by learning to rank figure-ground segmentations, using ground truth annotations available in object class recognition datasets and based on a set of low and mid-level properties. The

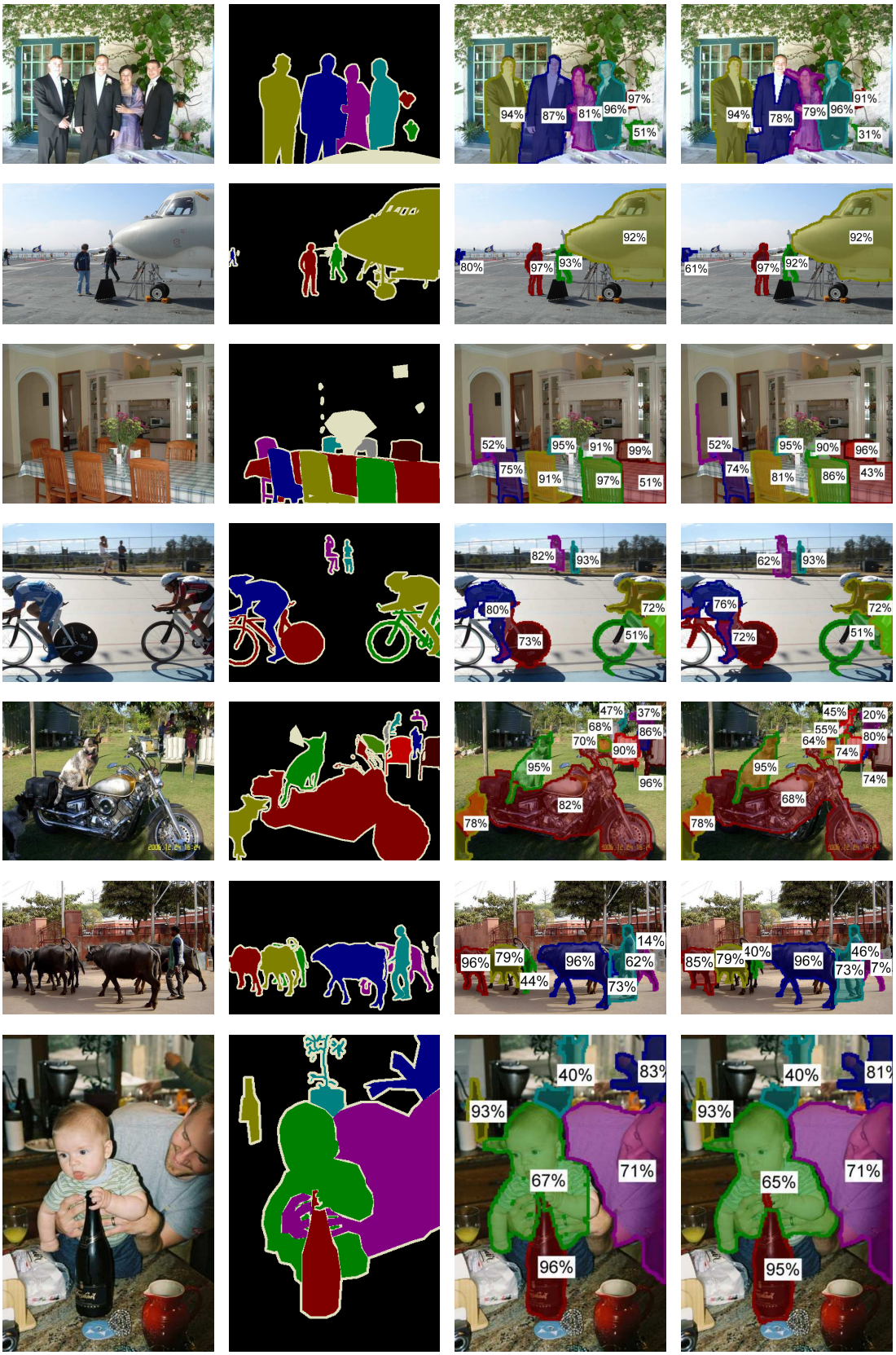


Fig. 9: Segmentation results on images from the validation set of the VOC2010 database. The **first** column contains the original images, the **second** gives the human ground truth annotations of multiple objects, the **third** shows the best segment in the Subframe-CPMC pool for each ground truth object, the **fourth** shows the best segment among the ones ranked in the top-200. The proposed algorithm obtains accurate segments for objects at multiple scales and locations, even when they are spatially adjacent. See fig. 10 for challenging cases.

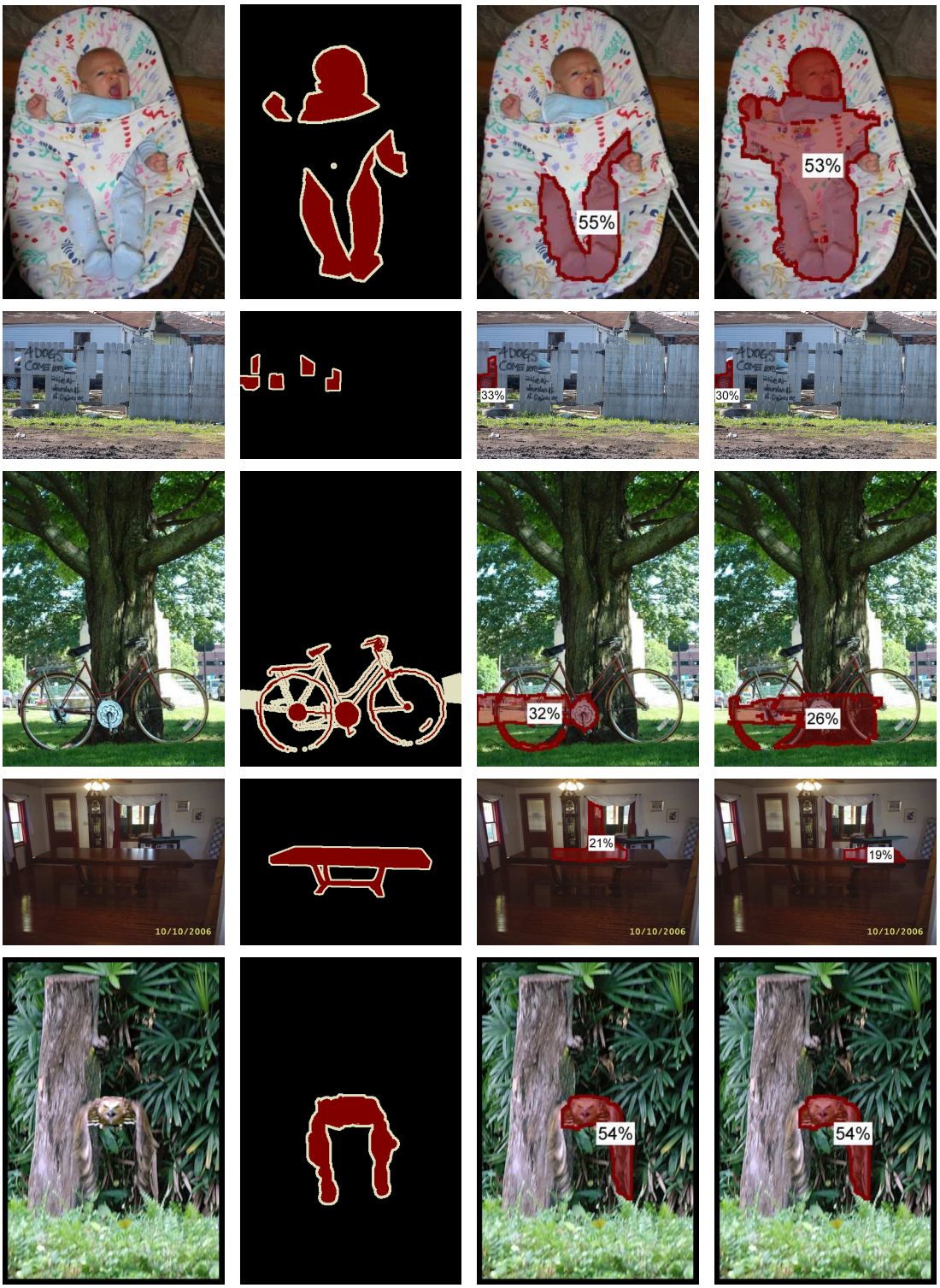


Fig. 10: Examples, taken from the validation set of VOC2010, where the CPMC algorithm encounters difficulties. The **first** column shows the images, the **second** the human ground truth annotations of multiple objects, the **third** shows the best segment in the entire Subframe-CPMC pool for each ground truth object, the **fourth** shows the best segment among the ones ranked in the top-200. Partially occluded objects (first two rows), wiry objects (third row) and objects with low background contrast (fourth and fifth row) can cause difficulties.

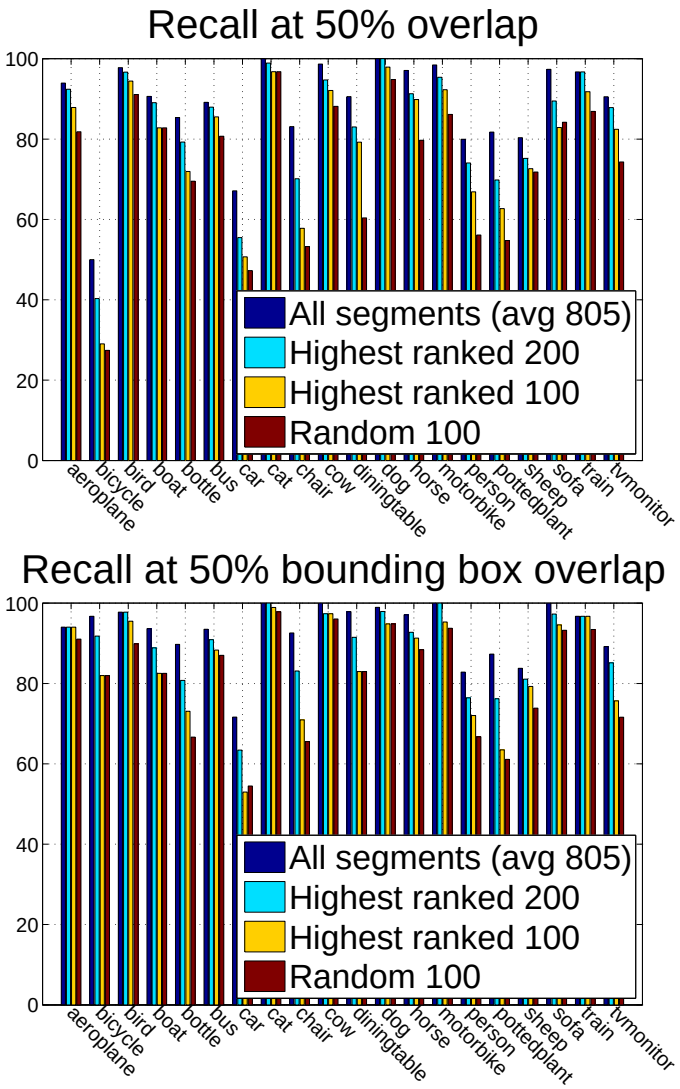


Fig. 12: Recall at 50% overlap between regions of ground truth objects and the best Subframe-CPMC segments (**top**) and between ground truth bounding boxes and best Subframe-CPMC segment bounding boxes (**bottom**). Note that bicycles are difficult to segment accurately due to their wiry structure, but there is usually some segment for each bicycle that has an accurate bounding box, such as the ones shown in the third row of fig. 2. These results are computed on the validation set of the VOC2010 segmentation dataset.

algorithm uses a very powerful new procedure to generate a pool of figure-ground segmentations — the Constrained Parametric Min-Cuts (CPMC). This uses parametric max-flow to efficiently compute non-degenerate figure-ground hypotheses at multiple scales on an image grid, followed by maximum relevance ranking and diversification. We have shown that the proposed framework is able to compute compact sets of segments that represent the objects in an image more accurately than existing state of the art segmentation methods. These sets of segments have been used successfully in segmentation-based recognition frameworks [1], [79], as well as for multi-region image segmentation [3], [4] and cosegmentation [80].

One difficulty for the current method is handling objects composed of disconnected regions that may arise from occlusion. While the energy minimization problems we solve sometimes generate such multiple regions, we chose to separate them into individual connected components, because they only rarely belong to the same object. In fact, in many such cases it may not be possible to segment the object correctly without top-down information. For exemple segmenting people embraced might require the knowledge of the number of arms a person has, and the configurations they can be in. It might be possible to handle such scenarios in a bottom-up fashion in simple situations, when cues like strong continuity may be exploited, but it appears more adequate to do this analysis at a higher level of scene interpretation.

The low-level segmentation and ranking components are also susceptible to improvement. Both components perform satisfactorily conditioned on the current state-of-the-art and datasets. One promising direction to improve the segmentation is the development of more sophisticated unary terms. Other advances may come from minimizing more powerful energy functions or the use of additional representations beyond regions. For example curves [81] may be more appropriate for objects that have long 'wiry' structures such as bicycles. The ranking component can be improved by developing better learning methodology, better features and by using more training data. At this point the segmentation component seems to allow the most improvement, but if applications set stringent constraints with respect to the maximum number of segments retained per image then ranking can become a bottleneck.

A somewhat suboptimal aspect of the proposed method is that energy minimization problems are solved independently, and the same number of problems is generated for all images, notwithstanding some having a single object and others having plenty. An interesting extension would make the process dynamic by making decisions on where and how to extract more segments conditioned on the solutions of the previous problems. This would be conceivably more efficient and would make the transition to video smoother. A sequential, conditional process could also make for a more biologically plausible control structure.

ACKNOWLEDGEMENTS

This work was supported, in part, by the European Commission, under MCEXT-025481, and by CNCSIS-UEFISCU, under project number PN II-RU-RC-2/2009.

REFERENCES

[1] F. Li, J. Carreira, and C. Sminchisescu, "Object recognition as ranking holistic figure-ground hypotheses," in *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2010.

[2] J. Carreira, F. Li, and C. Sminchisescu, "Object Recognition by Sequential Figure-Ground Ranking," *International Journal of Computer Vision*, 2012.

[3] J. Carreira, A. Ion, and C. Sminchisescu, "Image segmentation by discounted cumulative ranking on maximal cliques," Computer Vision and Machine Learning Group, Institute for Numerical Simulation, University of Bonn, Tech. Rep. 06-2010, June 2010.

[4] A. Ion, J. Carreira, and C. Sminchisescu, "Image Segmentation by Figure-Ground Composition into Maximal Cliques," in *IEEE International Conference on Computer Vision*, November 2011.

[5] J. Shotton, J. M. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation." in *European Conference on Computer Vision*, May 2006, pp. 1–15.

[6] S. Alpert, M. Galun, R. Basri, and A. Brandt, "Image segmentation by probabilistic bottom-up aggregation and cue integration," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 0, pp. 1–8, 2007.

[7] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2009 (VOC2009) Results," http://www.pascal-network.org/challenges/VOC/voc2009/workshop/index.html.

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results," http://www.pascal-network.org/challenges/VOC/voc2010/workshop/index.html.

[9] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 1627–1645, 2010.

[10] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman, "Multiple kernels for object detection," in *IEEE International Conference on Computer Vision*, September 2009.

[11] S. Gould, T. Gao, and D. Koller, "Region-based segmentation and object detection," in *Advances in Neural Information Processing Systems*, Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, Eds., December 2009, pp. 655–663.

[12] Y. Yang, S. Hallman, D. Ramanan, and C. Fowlkes, "Layered object detection for multi-class segmentation," in *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2010.

[13] L. Ladicky, P. Sturgess, K. Alaharia, C. Russel, and P. H. Torr, "What, where & how many ? combining object detectors and crfs," in *European Conference on Computer Vision*, September 2010.

[14] J. M. Gonfaus, X. Boix, J. van de Weijer, A. D. Bagdanov, J. Serrat, and J. Gonzàlez, "Harmony potentials for joint classification and segmentation," in *IEEE International Conference on Computer Vision and Pattern Recognition*, San Francisco, California, USA, June 2010, pp. 1–8.

[15] B. Russell, W. Freeman, A. Efros, J. Sivic, and A. Zisserman, "Using multiple segmentations to discover objects and their extent in image collections," *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 1605–1614, June 2006.

[16] J. Carreira and C. Sminchisescu, "Constrained parametric min-cuts for automatic object segmentation, release 1," http://sminchisescu.ins.uni-bonn.de/code/cpmc/.

[17] J. Muerle and D. Allen, "Experimental evaluation of techniques for automatic segmentation of objects in a complex scene." in *Pictorial Pattern Recognition*, 1968, pp. 3–13.

[18] R. M. Haralick and L. G. Shapiro, "Image segmentation techniques," *Computer Vision, Graphics, and Image Processing*, vol. 29, no. 1, pp. 100–132, 1985.

[19] S. Zhu, T. Lee, and A. Yuille, "Region competition: unifying snakes, region growing, energy/bayes/mdl for multi-band image segmentation," in *Computer Vision, 1995. Proceedings., Fifth International Conference on*, June 1995, pp. 416 –423.

[20] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 99, no. PrePrints, 2010.

[21] J. L. Mundy, "Object recognition in the geometric era: A retrospective," in *Toward Category-Level Object Recognition*, 2006, pp. 3–28.

[22] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE International Conference on Computer Vision and Pattern Recognition*, December 2001.

[23] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *ICCV*, vol. 2, July 2001, pp. 416–423.

[24] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "Toward objective evaluation of image segmentation algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 929–944, June 2007.

[25] D. Hoiem, A. A. Efros, and M. Hebert, "Geometric context from a single image," *IEEE International Conference on Computer Vision*, vol. 1, pp. 654–661, October 2005.

[26] T. Malisiewicz and A. Efros, "Improving spatial support for objects via multiple segmentations," *British Machine Vision Conference*, September 2007.

[27] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[28] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.

[29] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, September 2004.

[30] T. S. A. Stein and M. Hebert, "Towards unsupervised whole-object segmentation: Combining automated matting with boundary detection," in *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2008.

[31] A. Rabinovich, S. Belongie, T. Lange, and J. M. Buhmann, "Model order selection and cue combination for image segmentation," *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 1130–1137, June 2006.

[32] E. Sharon, M. Galun, D. Sharon, R. Basri, and A. Brandt, "Hierarchy and adaptivity in segmenting visual scenes," *Nature*, vol. 442, no. 7104, pp. 719–846, June 2006.

[33] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "From contours to regions: An empirical evaluation," in *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 2294–2301.

[34] M. Maire, P. Arbelaez, C. Fowlkes, and J. Malik, "Using contours to detect and localize junctions in natural images," *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 0, pp. 1–8, June 2008.

[35] G. Gallo, M. D. Grigoriadis, and R. E. Tarjan, "A fast parametric maximum flow algorithm and applications," *SIAM J. Comput.*, vol. 18, no. 1, pp. 30–55, February 1989.

[36] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient n-d image segmentation," *International Journal of Computer Vision*, vol. 70, no. 2, pp. 109–131, 2006.

[37] C. Rother, V. Kolmogorov, and A. Blake, ""grabcut": interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.

[38] S. Bagon, O. Boiman, and M. Irani, "What is a good image segment? a unified approach to segment extraction," *European Conference on Computer Vision*, pp. 30–44, October 2008.

[39] T. Schoenemann, F. Kahl, and D. Cremers, "Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation," *IEEE International Conference on Computer Vision*, 2009.

[40] S. Vicente, V. Kolmogorov, and C. Rother, "Graph cut based image segmentation with connectivity priors," *IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 0–7, 2008.

[41] C. Fowlkes, D. Martin, and J. Malik, "Learning affinity functions for image segmentation: combining patch-based and gradient-based approaches," in *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 2, June 2003, pp. II–54–61 vol.2.

[42] P. Dollár, Z. Tu, and S. Belongie, "Supervised learning of edges and object boundaries," in *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2006.

[43] J. Kaufhold and A. Hoogs, "Learning to segment images using region-based perceptual features," *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 954–961, June 2004.

[44] X. Ren and J. Malik, "Learning a classification model for segmentation," *IEEE International Conference on Computer Vision*, vol. 1, p. 10, October 2003.

[45] B. Peng and O. Veksler, "Parameter Selection for Graph Cut Based Image Segmentation," in *British Machine Vision Conference*, September 2008.

[46] V. Lempitsky, A. Blake, and C. Rother, "Image segmentation by branch-and-mincut," in *European Conference on Computer Vision*, October 2008, pp. IV: 15–29.

[47] D. Cremers, F. R. Schmidt, and F. Barthel, "Shape priors in variational image segmentation: Convexity, lipschitz continuity and globally optimal solutions," *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 0, pp. 1–6, June 2008.

[48] T. Schoenemann and D. Cremers, "Globally optimal image segmentation with an elastic shape prior," *IEEE International Conference on Computer Vision*, vol. 0, pp. 1–6, October 2007.

[49] E. Borenstein, E. Sharon, and S. Ullman, "Combining top-down and bottom-up segmentation," *Computer Vision and Pattern Recognition Workshop*, pp. 46–46, June 2004.

[50] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 259–289, 2008.

[51] A. Levin and Y. Weiss, "Learning to combine bottom-up and top-down segmentation," *International Journal of Computer Vision*, vol. 81, no. 1, pp. 105–118, 2009.
[52] M. P. Kumar, P. Torr, and A. Zisserman, "Objcut: Efficient segmentation using top-down and bottom-up cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 530–545, 2010.
[53] J. Carreira, F. Li, and C. Sminchisescu, "Ranking figure-ground hypotheses for object segmentation," oral presentation at the PASCAL VOC 2009 Workshop, available online at http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2009/workshop/.
[54] J. Carreira and C. Sminchisescu, "Constrained Parametric Min-Cuts for Automatic Object Segmentation," in *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2010.
[55] I. Endres and A. Hoiem, "Category independent object proposals," in *European Conference on Computer Vision*, September 2010.
[56] A. Levinshtein, C. Sminchisescu, and S. Dickinson, "Optimal contour closure by superpixel grouping," in *European Conference on Computer Vision*, September 2010.
[57] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?" in *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2010.
[58] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," in *CVPR*, June 2007, pp. 1 –8.
[59] V. Kolmogorov, Y. Boykov, and C. Rother, "Applications of parametric maxflow in computer vision," *IEEE International Conference on Computer Vision*, pp. 1–8, October 2007.
[60] D. S. Hochbaum, "The pseudoflow algorithm: A new algorithm for the maximum-flow problem," *Oper. Res.*, vol. 56, pp. 992–1009, July 2008.
[61] M. A. Babenko, J. Derryberry, A. V. Goldberg, R. E. Tarjan, and Y. Zhou, "Experimental evaluation of parametric max-flow algorithms," in *WEA*, 2007, pp. 256–269.
[62] S. Wang and J. M. Siskind, "Image segmentation with ratio cut," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 675–690, 2003.
[63] M. Wertheimer, "Laws of organization in perceptual forms (partial translation)," in *A sourcebook of Gestalt Psychology*, 1938, pp. 71–88.
[64] S. E. Palmer, *Vision Science: Photons to Phenomenology*. The MIT Press, 1999.
[65] S. C. Zhu and D. Mumford, "Learning Generic Prior Models for Visual Computation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 11, 1997.
[66] S.-C. Zhu, "Embedding gestalt laws in markov random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 11, pp. 1170 –1187, nov 1999.
[67] Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1101–1113, 1993.
[68] O. Chapelle, P. Haffner, and V. Vapnik, "Support vector machines for histogram-based image classification," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1055–1064, 1999.
[69] A. M. Bruckstein, R. J. Holt, and A. N. Netravali, "Discrete elastica," in *DCGI '96: Proceedings of the 6th International Workshop on Discrete Geometry for Computer Imagery*. Lyon, France: Springer-Verlag, 1996, pp. 59–72.
[70] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
[71] A. Jaiantilal, "Classification and regression by randomforest-matlab," Available at http://code.google.com/p/randomforest-matlab, 2009.
[72] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: ACM, August 1998, pp. 335–336.
[73] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
[74] S. Belongie, J. Malik, and J. Puzicha, "Shape context: A new descriptor for shape matching and object recognition," in *Advances in Neural Information Processing Systems*, November 2000, pp. 831–837.
[75] A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," *ACM International Conference on Image and Video Retrieval*, pp. 401–408, July 2007.
[76] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Discriminatively trained deformable part models, release 4," http://people.cs.uchicago.edu/ pff/latent-release4/.
[77] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *Advances in Neural Information Processing Systems*, December 2007.
[78] F. Li, C. Ionescu, and C. Sminchisescu, "Random Fourier approximations for skewed multiplicative histogram kernels," in *Lecture Notes for Computer Science (DAGM)*, September 2010, dAGM paper prize.
[79] A. Ion, J. Carreira, and C. Sminchisescu, "Probabilistic Joint Image Segmentation and Labeling," in *Advances in Neural Information Processing Systems*, December 2011.
[80] S. Vicente, C. Rother, and V. Kolmogorov, "Object cosegmentation," in *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 2217–2224.
[81] Z. Tu and S.-C. Zhu, "Parsing images into regions, curves, and curve groups," *Int. J. Comput. Vision*, vol. 69, pp. 223–249, August 2006.
[82] V. Kolmogorov and R. Zabin, "What energy functions can be minimized via graph cuts?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 147–159, Feb. 2004.
[83] J. Ponce, M. Hebert, C. Schmid, and A. Zisserman, Eds., *Toward Category-Level Object Recognition*, ser. Lecture Notes in Computer Science, vol. 4170. Springer, 2006.
[84] S. Dickinson, A. Leonardis, B. Schiele, and M. Tarr, *Object Categorization: Computer and Human Vision Perspectives*. Cambridge University Press, 2009.
[85] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.

João Carreira received a Licenciatura in Electrotechnical and Computer Engineering from the University of Coimbra, Portugal in 2005. He was a research assistant in Coimbra until 2008, when he became a PhD student at the University of Bonn. His current research interests lie in bottom-up processes for segmentation and object recognition.

Cristian Sminchisescu has obtained a doctorate in Computer Science and Applied Mathematics with an emphasis on imaging, vision and robotics at INRIA, France, under an Eiffel excellence doctoral fellowship, and has done postdoctoral research in the Artificial Intelligence Laboratory at the University of Toronto. He is a member in the program committees of the main conferences in computer vision and machine learning (CVPR, ICCV, ECCV, NIPS, AISTATS), area chair for ICCV07 and 11, and a member of the Editorial Board (Associate Editor) of IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI). He has given more than 50 invited talks and presentations and has offered tutorials on 3d tracking, recognition and optimization at ICCV and CVPR, the Chicago Machine Learning Summer School, the AERFAI Vision School in Barcelona and the Computer Vision Summer School (VSS) in Zurich. Sminchisescu's research goal is to train computers to see. His research interests are in the area of computer vision (articulated objects, 3d reconstruction, segmentation, and object and action recognition) and machine learning (optimization and sampling algorithms, structured prediction, sparse approximations and kernel methods). Recent work by himself and his collaborators has produced state-of-the art results in the monocular 3d human pose estimation benchmark (HumanEva) and was the winner of the PASCAL VOC object segmentation and labeling challenge, in 2009 and 2010.