A Holistic Approach for Data-Driven Object Cutout

Huayong Xu[1], Yangyan Li[3], Wenzheng Chen[1], Dani Lischinski[2], Daniel Cohen-Or[3], and Baoquan Chen[1]

[1] Shandong University
[2] Hebrew University of Jerusalem
[3] Tel Aviv University

Abstract. Object cutout is a fundamental operation for image editing and manipulation, yet it is extremely challenging to automate it in real-world images, which typically contain considerable background clutter. In contrast to existing cutout methods, which are based mainly on low-level image analysis, we propose a more *holistic* approach, which considers the entire shape of the object of interest by leveraging higher-level image analysis and learnt global shape priors. Specifically, we leverage a deep neural network (DNN) trained for objects of a particular class (chairs) for realizing this mechanism. Given a rectangular image region, the DNN outputs a probability map (P-map) that indicates for each pixel inside the rectangle how likely it is to be contained inside an object from the class of interest. We show that the resulting P-maps may be used to evaluate how likely a rectangle proposal is to contain an instance of the class, and further process good proposals to produce an accurate object cutout mask. This amounts to an automatic end-to-end pipeline for catergory-specific object cutout. We evaluate our approach on segmentation benchmark datasets, and show that it significantly outperforms the state-of-the-art on them.

1 Introduction

Object cutout is a fundamental operation in image editing and manipulation [3, 35, 37], an operation which graphics artists perform routinely. Performing this operation in a completely automatic fashion involves solving two classical and challenging computer vision tasks: object detection and semantic segmentation. Furthermore, in some scenarios an automatic approach is infeasible, since the user's intent is difficult to predict. Thus, a variety of interactive cutout tools have been proposed over the years, e.g., [24, 22, 28]. A common approach is to let the user indicate the object of interest with a bounding box, and attempt to proceed automatically from this minimal input to obtain an accurate cutout mask [28].

However, these tasks of detection and segmentation, which the human visual system accomplishes with ease, are notorious for being surprisingly hard for a computer program. They are especially challenging when the object of interest is

Fig. 1. A cluttered scene with four chairs (a); an aggregated P-map visualizing object detection (b); local P-maps inside proposed rectangles (c) ; cutouts produced with the aid of our local P-maps (d) ; cutouts produced using GrabCut, for the same rectangles (e) .

located in front of a cluttered background, which may contain many distractions, such as other objects with similar low-level statistics to the foreground object. Such an example is demonstrated in Fig. 1(a), where the background contains chairs identical in appearance to those in the foreground.

Methods that are based mainly on low-level image analysis tend to fail when the foreground object and the background are not statistically separable, and when salient separating edges cannot be easily detected. Sparse user input, such as a bounding box [28] or a pair of scribbles [22], is not sufficient to overcome these difficulties, as demonstrated in Fig. 1(e). A more *holistic* approach, which considers the whole shape rather than its pieces by leveraging higher-level image analysis and global object shape priors, has a better chance of coping with these challenging scenarios.

Recent advances in deep neural networks (DNNs) have shown promising results in solving various image understanding tasks, such as classification, detection and segmentation [29]. However, object cutout presents DNNs with three additional challenges. Firstly, the network should learn a large variety of detailed shape priors, which differ significantly among different object classes. Secondly, the solution space is high-dimensional, since the images operated upon and the resulting cutout masks are required to be of high resolution. Thirdly, the cutout masks have sharp boundaries. For example, the state-of-the-art DNN-based instance-level object segmentation approach of [21] achieves 24.5% AP^r at 0.5 IoU[4], on the chair class, which is far from being useful for graphics applications.

In this work, we leverage a Convolution-Deconvolution (DeconvNet) DNN [25]. However, we train it specifically using objects of a particular class (chairs). By

[4] AP is short for *average precision*, which is the area under precision-recall (PR) curve. IoU is short for Intersection over Union, i.e., $A(P \bigcap G)/A(P \bigcup G)$, where P and G are segmentation prediction and ground truth, respectively, while $A(\bullet)$ indicates their areas. To measure the precision of segmentation, AP^r is used, which is *region* based AP. Here, a segmentation is considered to be positive when it reaches 0.5 IoU.

focusing the training on a particular class, we reduce the learning difficulty and push it to learn more detailed shape priors. Moreover, to provide a more exhaustive coverage of the class in the training phase, we leverage synthetic imagery generated from ShapeNet [33]. Given a rectangular image region, the trained network generates a map (of the same resolution as the input region), where each pixel indicates the likelihood of belonging to the object. We refer to such maps as *P-maps* for short.

We show that the resulting P-maps are useful for a number of vision tasks and applications.

First, given a set of proposals (generated by any state-of-the-art method), we are able to evaluate and rank it better using the P-map. This capability enhances automatic location of chairs in an image. Second, getting back to the original motivation for our work, we are able to use the P-map to guide an iterative graphcut process [28] towards an accurate object cutout (see Fig. 1(d)). Thus, the approach described in this paper amounts to an end-to-end solution for automatic object cutout.

We use chairs as our running example, as they represent a family of shapes that have a rich variability of geometry and topology, and pose a challenge to state-of-the-art DNNs. Our technique is specifically designed to deal with cluttered images, learning to extract the foreground shape from a background that may contain objects with similar local statistics. We show that our holistic shape prior based approach considerably improves the accuracy of the resulting cutouts, compared to the current state-of-the-art, especially for cluttered images.

2 Related Work

Over the past few decades, tremendous amount of research have been devoted to studying how to faithfully perceive objects in images. Significant progress has been made on several sub-tasks towards this goal, including object recognition, object detection, and semantic segmentation, from which still only a coarse understanding of the scene can be established. In this section, we briefly review advances made in these directions and discuss their connections to the task of instance-level object cutout.

Image segmentation is the process of partitioning an image into multiple segments of similar appearance. The problem can be formulated as a clustering problem in color space [4]. To incorporate more spatial constrains into the process, the image may be modeled as a graph, converting image segmentation into a graph partition problem. The weights on the graph edges can either be inferred from pixel colors [10] or from sparse user input, as an addition [28]. Algorithms have been proposed for efficiently computing the partition, even when the pixels are densely connected (DenseCRF) [16]. Such methods are capable of inferring a sharp segmentation mask from sparse of fuzzy probabilities, and thus are widely used as a post-process for methods that produce segmentation probability maps.

Semantic segmentation. Instead of grouping pixels only by appearance, semantic segmentation forms segments by grouping pixels belonging to same semantic objects; thus, a single segment might contain heterogeneous appearances. Since such segmentation depends on semantic understanding of the image content, state-of-the-art methods operate by running classification neural networks on patches densely sampled from the image in order to predict the semantic label of their central pixels [23, 26, 36]. Instead, Noh et al. [25] proposed a DeconvNet to directly output a high resolution semantic segmentation. We leverage DeconvNet for solving the more challenging object cutout problem by adapting and training it extensively on objects from a specific class.

Object cutout. Object cutout further pushes semantic segmentation from category-level to instance-level. The additional challenge is that objects with similar appearance may hinder the cutout accuracy for individual instances. The state-of-the-art addresses the object cutout problem by solving it jointly with detection [13, 21], object number prediction [20], or by explicitly modeling the occlusion interactions between different instances [30, 2]. Though significant progress has been made recently, the performance on some object categories is still very low. In this work, we take advantage of being able to utilize training data synthesized from 3D models [31], and focus on leveraging rich holistic shape priors for addressing segmentation ambiguities.

3D object retrieval and view estimation. Recently, exciting advances in image based 3D object retrieval and object view estimation have made [1, 19, 31]. Such efforts are quite related to object cutout, as the retrieved 3D model can be rendered in the estimated view to approximate the object in the image, thus providing a strong prior for cutout. However, we found that the gap between projected proxies and accurate cutout masks cannot be easily bridged. One reason is that there are only few models in the existing shape databases that match well with real world objects. The inherent mismatch between 3D database and real world objects, plus the introduced retrieval and view estimation errors, render it infeasible to compute object cutout through such an approach, in general cases.

Object detection. Object detection is usually done in two steps: object bounding box proposal generation and proposal evaluation. Proposal generation yields a set of bounding boxes that potentially contain objects [34, 38, 17]. Proposal evaluation typically extracts features from the image patches contained in the proposed bounding rectangle, and estimates the confidence of the image patches to belong to objects of certain classes. R-CNN [11] is an representative work in object detection and several works extended it to further improve efficiency and accuracy [14, 12, 27]. We show that the P-maps generated by our category-specific DeconvNet can benefit proposal evaluation for improving object detection and subsequent object cutout.

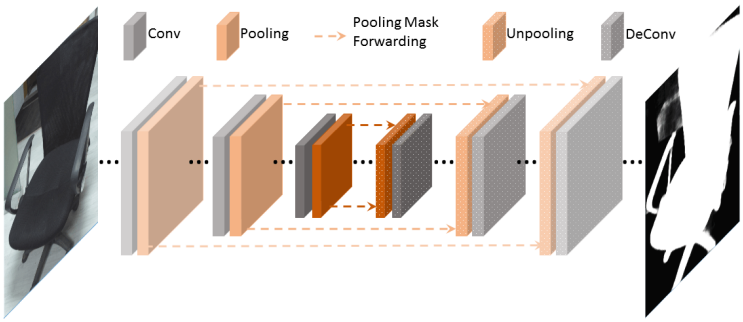

Fig. 2. A schematic illustration of the DeconvNet architecture. Records of the max pooling operations that occur during the first convolutional half, are forwarded to the subsequent deconvolution half of the network.

3 Instance Probability Maps

In this section, we introduce our method for generating instance probability maps. The term "instance" indicates that the maps aim to locate specific instances of a particular object class, rather than only detect the presence of such an object in the image. These probability maps, which will be referred to as P-maps, specify for each pixel its likelihood of belonging to an object instance. As we show in later sections, they allow efficient detection and consequent cutouts of objects, as well as the retrieval of 3D shapes.

Our P-maps are based on the non-trivial observation that although an image of an object may be high-dimensional, the underlying object can often be represented by a compact feature vector. Dosovitskiy et al. [7] show that a DNN can be trained to generate object images from given object type, viewpoint, and color. This raises the expectation that neural networks can detect the presence of an object, encode it into a rather low-dimensional feature vector, from which it then should be possible to "reconstruct" the object, or its binary cutout mask. The premise of this approach is that the extraction of this low-dimensional representation in fact "peels off" the background clutter.

However, only rather fuzzy images can be reconstructed if the feature vector is extracted from real-world cluttered images, instead of a clean feature vector consisting of object type, viewpoint, and color [8]. To generate a sharper image or cutout mask, additional information must be passed into the reconstruction process, and we build our approach upon the DeconvNet architecture proposed by Noh et al. [25], which we found to be better suited for cluttered scenes. In this network, not only the feature vector, but also additional information about the feature extraction process is forwarded into the reconstruction process, which greatly improves the reconstruction sharpness.

More specifically, the feature extraction part of our network (see Fig. 2) is composed of convolutional layers and pooling layers, which gradually encode the input as a 4096-dimensional feature vector. This feature vector is then taken by

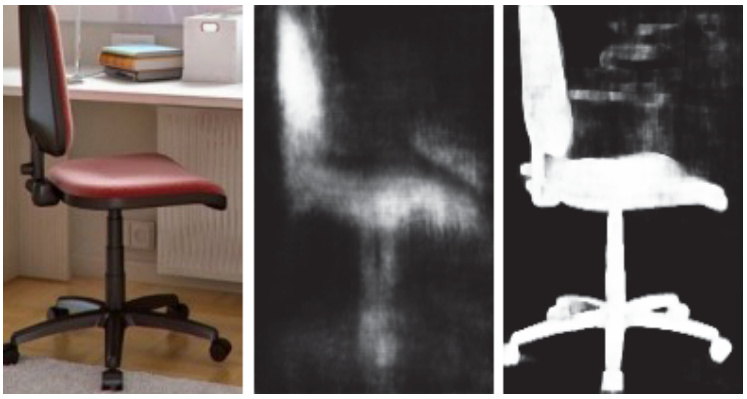

Fig. 3. . Given an input image (left), DNN trained extensively with large amount of images from a particular class can learn to "reconstruct" a fuzzy image while ignoring background clutters (middle). Pooling mask forwarding in DeconvNet greatly improves the sharpness of output cutout probability maps (right).

the reconstruction part of the network composed of deconvolutional layers and unpooling layers, which gradually reconstruct the P-map. Importantly, the pooling masks, which record the full history of the pooling operations, are forwarded into the unpooling layers. The pooling mask forwarding relieves the difficulty in learning how to perform a sharp reconstruction, thus greatly outperforming approaches that only use the feature vector. See Fig. 3 for a visual comparison of results with and without the use of pooling masks.

The original DeconvNet was proposed for solving a semantic segmentation problem using 21 classes. We adapt it to solve our instance-level segmentation problem by changing its last layer to output only two channel images: one for foreground and one for background. Then a softmax function over these two channels gives the foreground/background probability for each pixel. DeconvNet was originally trained on PASCAL VOC 2012 [9] data, where the number of segmented images is not particularly high, since image segmentation is a hard task for crowd sourcing. When narrowing the data to a specific category, it is insufficient to inject enough shape priors into the trained model. Instead, we choose to train the network using a much larger number of synthetic images with ground truth cutout masks, which are generated completely automatically by rendering 3D models. In the reminder of the paper, we refer our adapted DeconvNet as DNCS (DeconvNet-Class-Specific). As we shall see, the amount and quality of our training data enables the trained network to learn a powerful shape prior, which makes it possible to perform well even in the presence of considerable background clutter.

Fig. 4. Comparison of instance probability maps and the resulting cutout masks generated by various baseline methods and by our approach. It may be seen that our DNCS is more successful at injecting the learnt shape priors into the probability map generation. Furthermore, our GrabCut+P cutout method makes more effective use of the probability maps to produce a cutout, compared to DenseCRF.

4 Proposal Evaluation

The ability to generate high-quality instance probability maps over rectangles of roughly the expected object size in the image is useful not only for generating

Fig. 5. An aggregated P-map for an entire image can be generated by accumulating instance probability map from bounding box proposals (b). By weighting the proposals with $\mathcal{X}_{CNN}$ an even better aggregated P-map can be generated (c).

Fig. 6. P-map enhanced chair detection results. Note that since our P-map "sees" the individual chairs, it can locate chairs well, even with heavy background clutter.

accurate binary cutout masks (Section 5), but also helpful for locating object instances from a given scene image, referred to as *detection task* in computer vision. Given a proposal, we are able to evaluate and rank it better when using the corresponding P-map, thus improve detecting object out of an entire image.

Proposal evaluation on RGB-P images. A proposal is a rectangular region in a large image, which is deemed likely to contain an object of interest. There are many methods that generate proposals, whose objective is to avoid performing an exhaustive search over the entire image. We show that using an RGB-P image, where the fourth P channel is computed by the instance cutout DNN, benefits such proposal evaluation methods. More formally, let I_b be a rectangular proposal, its evaluation by a function $\mathcal{X} : I_b \rightarrow \mathcal{R}$, maps the input proposal to a real value that indicates the confidence of having an object of a specific class contained in it. In our case, the function $\mathcal{X}$ is no more than a binary classifier that tells how likely the proposal depicts a chair.

We train the classifier $\mathcal{X}$ with synthetic images, and we generate many rectangular proposals with any state-of-the-art methods. Since in our synthetic images we know the ground truth bounding boxes of the objects, we can easily generate positive and negative examples. We treat proposals with more than 80% over-

lap with the ground truth bounding boxes as positive samples, and the rest as negative samples. For each proposal, we also compute its P-channel.

We trained two classifiers: $\mathcal{X}_{SVM}$ and $\mathcal{X}_{CNN}$. For $\mathcal{X}_{SVM}$, we extract AlexNet [18] CNN features (4096 dimensions, the output of $fc7$ layer) from the RGB channels, and HoG [6] features (24304 dimensions) from the P-map, which are then reduced with PCA to 4096 dimensions. Then we concatenate the CNN features and the PCA reduced HoG features for training a linear Support Vector Machine (SVM). The $\mathcal{X}_{CNN}$ classifier is an end-to-end CNN approach, where we add a fourth channel to the filters of the first convolutional layer of AlexNet to adapt the additonal P-channel, and fine tune the network to work as a binary classifier.

The effect of proposal evaluation is visualized in an aggregated P-map in Fig. 5, where we generate an aggregated P-map, by running instance cutout in all proposals, accumulating the resulting P-maps with weights from the confidences given by $\mathcal{X}_{CNN}$, and normalizing the result. Another example of such a map is shown in Fig. 1(b). It is clear that our P-map enhanced proposal evaluation can greatly narrow down attentions to chair regions. We compare the performance of our two classifiers with versions trained without using the P channel, and found that both classifiers perform better when P channels is used (see Table 2). This is a strong evidence that the P-channels are effectively improving the proposal evaluation. As can be seen from Fig. 6, chairs, even with heavy background clutters can be well located by our P-map powered detection.

5 Cutout Mask Extraction

Given a P-map generated by DNCS within a proposal rectangle, our goal is now to generate a binary cutout mask for the object of interest contained therein. We achieve this goal by adapting the iterative graphcut approach (GrabCut) of Rother et al. [28].

The original GrabCut algorithm uses the bounding rectangle to initialize two GMM color models, one for the background, based on colors outside the rectangle, and one for the foreground, based on colors inside the rectangle. The minimum graphcut is then computed [15], using the two color models to determine the unary (data) term for each pixel. The process is then repeated iteratively using the result from the previous iteration to update the background and foreground GMMs, instead of the initial rectangle.

The above process will generally fail to converge to an accurate cutout mask whenever there is a significant overlap between the background and foreground color models, which will happen if the background contains objects with similar colors to those of the foreground object, as demonstrated in Fig. 1(e). However, armed with our P-map we can initialize the background and foreground color models in a much more precise fashion.

Specifically, we first convert the continuous P-map into an initial binary foreground mask, by computing the minimum graphcut where the unary term at each pixel is determined by our P-map. Denoting by p_i the P-map value of pixel i, we set the foreground likelihood to $P_i^F = p_i^\alpha$ and the background

Table 1. IoU comparison of various instance cutout probability map generation methods with various post-processing methods. In *ShapeView*, the cutout probability map is generated by rendering images of similar shape retrieval in estimated viewpoint. *DeconvNet (original)* is the DeconvNet model trained on 21 classes of images from PASCAL VOC 2012. *GrabCut + P* is our method described in Section 5.

	GrabCut	ShapeView	DeconvNet (original)	DNCS	
				DenseCRF	GrabCut + P
PASCAL 2012	45.6	46.2	39.8	49.4	**52.1**
Our Benchmark	58.1	63.3	59.6	78.9	**81.5**

likelihood to $P_i^B = (1 - p_i)^\alpha$, where $\alpha = 2.3$. The resulting binary mask is then used to initialize the two GMM color models, instead of the bounding rectangle. In subsequent iterations, we set the unary term to a weighted combination of the value predicted by the GMM color model and the P-map likelihood, with the latter's weight decreasing as the iterations progress:

$$\begin{aligned} CP_i^F &= GMM_i^F \exp(-wP_i^B) \\ CP_i^B &= GMM_i^B \exp(-wP_i^F), \end{aligned} \tag{1}$$

where GMM^F and GMM^B are the color models for the foreground and background, respectively. The weight $w = b/k$, where k is the iteration number and $b = 25$ was empirically tuned to reduce the influence of P^F and P^B as the iterations progress.

Fig. 1(d,e) compares two results produced using our P-map enhanced GrabCut (d) with those of the original GrabCut approach (e). It may be seen that the latter includes in the cutout mask parts of the background which have similar appearance to the foreground chair (in fact, these are parts of identical chairs in the background), while our approach produces a nearly perfect cutout mask.

6 Experiments

In this section, we quantitatively evaluate the performance of our instance cutout approach, and compare it against several other baseline methods. We also quantitatively evaluate the boost in object detection performance enabled by the use of our P-maps.

6.1 Evaluation of Instance Cutout

Dataset and evaluation metric. We evaluate our instance cutout performance on two chair image datasets. One is from PASCAL VOC 2012, which contains 175 chair images with ground truth cutout annotations. We found this dataset to be highly challenging for the cutout task, as it contains not only background

Fig. 7. Examples of instance probability maps generated by retrieving similar shapes and rendering them from the corresponding predicted viewpoints (ShapeView probability maps).

clutter, but also heavy occlusion, thus many of the chair instances are only partially visible. Occlusion also makes it more challenging for object detection to providing reasonably good proposals, since a rather complete presence of the object of interest is expected. In addition, we have prepared another benchmark, with 418 chair images, which contains considerable background clutter, but fewer occlusions. We evaluate different approaches using the Intersection over Union (IoU) metric, which measures the ratio between the areas of intersection and union of ground truth and predicted cutout masks. Higher IoU score indicates better cutout accuracy.

Baseline methods. Recent advances in image based 3D object retrieval [19] and object view estimation [31] provide an potential solution for generating an instance probability map, by retrieving similar shapes and rendering them from the predicted viewpoint. The rendered images approximate the underlying object in the input image, and thus can be used as probability maps for instance cutout. More specifically, we pick top 5 retrievals, render them as binary images from the predicted viewpoints, weight the rendered images by the retrieval confidence, and then overlay them into a normalized instance cutout probability map. We refer to this approach as "ShapeView"; see Fig. 7 for examples of the resulting instance probability maps. Another baseline to our approach are the probability maps generated by the original DeconvNet, which was trained for semantic segmentation with 21 classes. In our comparisons we use GrabCut [28] to generate a cutout mask directly from a given image with a proposal rectangle,

Table 2. Object proposal evaluation accuracy of classifiers $\mathcal{X}_{SVM}$ and $\mathcal{X}_{CNN}$ on RGB images and RGB-P images. Augmenting the image with a P-channel boosts the performance of both classifiers.

	RGB Images	RGB-P Images
$\mathcal{X}_{SVM}$	69.6	87.9
$\mathcal{X}_{CNN}$	65.6	86.5

while DenseCRF is used for generating a cutout mask from instance probability maps.

We compare our P-map enhanced GrabCut method (Section 5) applied on the P-maps generated by DNCS model against the original GrabCut, and DenseCRF applied on probability maps generated using the ShapeView approach and the original DeConvNet. The quantitative results are summarized in Table 1, while Fig. 4 shows a visual comparison using nine examples from our benchmark. Note that our method outperforms the baseline methods on both the PASCAL VOC 2012 dataset (by 5.9%) and on our benchmark (by 18.2%) (see Table 1). The full set of the test images and the results of these methods is included in our supplementary materials. The performance boost on our benchmark is much higher, since our network was trained with synthetic images that exhibit considerable background clutter, but no occlusions. This suggests an interesting future work direction on synthesizing images with realistic occlusion patterns for training occlusion-aware DNNs. Note that the ShapeView baseline method we proposed also consistently outperforms the original DeConvNet. This may be explained by the fact that it is trained on many classes, and thus cannot learn a sufficiently strong shape prior for each class.

6.2 Evaluation of Object Proposal Evaluation

We evaluate the performance of the $\mathcal{X}_{SVM}$ and $\mathcal{X}_{CNN}$ classifiers described in Section 4 on 35154 proposals generated by the Selective Search method [34]. These proposals were generated from 52 images from our benchmark, with each of the images containing a single chair. We measure the accuracy by the average recall on positive and negative samples.

We compare our P-map enhanced $\mathcal{X}_{SVM}$ and $\mathcal{X}_{CNN}$ classifiers against those trained without P-maps, and found that the use of P-maps greatly enhances proposal evaluation accuracy, as reported in Table 2. Our experiment suggests that the instance cutout task should be more tightly coupled with object detection tasks, as the improvement in one benefits the other.

6.3 Comparison to Seeing 3D Chairs

We also compare chair detection performance based on Selective Search + $\mathcal{X}$ with that proposed in Seeing 3D Chairs [1]. Given an image, Seeing 3D Chairs

Table 3. Comparison of top-k detection accuracy between Seeing 3D Chairs, and our P-map powered detection pipeline.

	Top-1	Top-2	Top-3	Top-4	Top-5
Seeing 3D Chairs	13.86	24.67	28.11	28.78	30.61
Selective Search + $\mathcal{X}_{SVM}$	21.73	28.76	35.49	40.16	43.49
Selective Search + $\mathcal{X}_{CNN}$	20.29	31.58	38.41	44.86	49.37

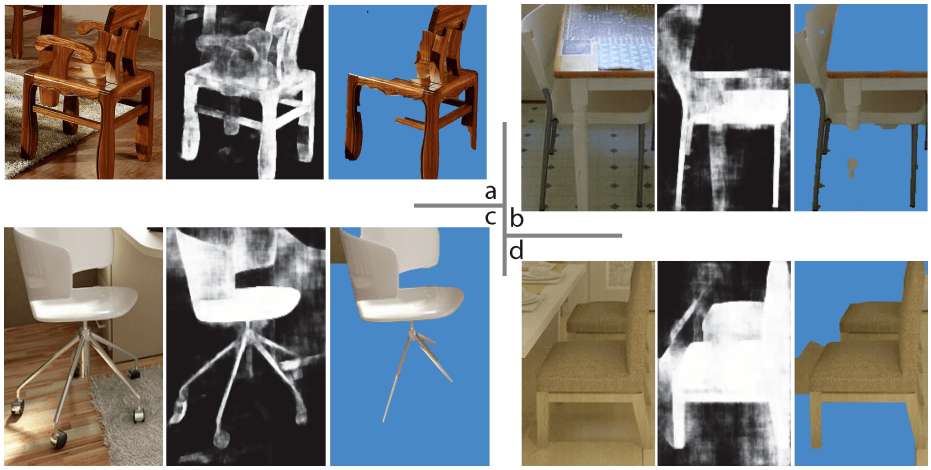

Fig. 8. Failure cases. We found several sources of errors in our cutout masks: (a) Chairs that are rarely seen in training data might be misunderstood by the DNN; (b) Occlusions pose additional challenges over background clutter; (c) The binary mask generation step sometimes eliminates thin structures even though they are preserved in the probability map; (d) Strong similarities between objects might result in highly confusing situation from specific view points.

outputs a ranked list of chair proposals. We generate chair proposals with Selective Search and then rank them with our classifiers. We compare the top-k detection accuracy of these approaches on the first 100 chair images from PASCAL VOC 2012. The results are reported in Table 3. Note that Seeing 3D Chairs is also an approach extensively trained on the chair class, yet we show that our P-map powered approach achieves better accuracy.

7 Conclusions

Many computer graphics applications depend on accurate object cutouts. Facilitating automatic cutout extraction remains extremely challenging, since it cannot rely on low-level image analysis alone, and necessarily requires some degree of high-level semantic analysis. The P-maps that we introduced aim to provide some of the latent semantics to assist in the extraction of cutouts. The

presented network aims to encode in the P-maps the essence of the shape prior with rich variability of geometry and topology.

The semantic information that P-maps carry was shown to be effective not only directly for cutouts, but also for locating the target object. We have shown that they significantly improve the evaluation of proposals, which are again means to enhance and accelerate a variety of applications that require image analysis.

The claim to fame of the P-maps is their competence to deal with cluttered images, where the target object has "rivals" in its background. Our network was designed explicitly to deal with these types of distractions, and together with our modified GrabCut approach makes a substantial step toward automatic and accurate instance cutout.

Nevertheless, our approach has its limitations. First, it is category specific, and requires training on the target class. It is intensively data-driven, which implies that a large amount of annotated data is required. For chairs, the problem is less significant since large 3D datasets are readily available. However, there are always peculiar shapes (see Fig. 8 (a)). For many other object classes there is no comparable availability of rich enough 3D models, yet. Second, the relative size of target object in the input image should be in an expected range, defined by the training set. Arguably, a more significant limitation of our technique is occlusion (see Fig. 8 (b)). While cluttering is handled well, occlusion remains a hurdle. For this reason, our performance advantage on the challenging PASCAL VOC 2012 benchmark is somewhat more modest. One of the challenges we encountered in training for occlusion is to realistically synthesize it, which is left for future work. Another limitation is demonstrated in Fig. 8 (c), where the final binary mask generation step sometimes fails capture thin structures, even though they are present in the P-map.

We believe that more fundamental processing can benefit from similar semantic layers. For example, image-based 3D shape retrieval, 2D-3D correspondence, or fitting and registering 3D proxies into an image. The P-maps or possibly similar semantic layers have the potential to boost the performance of applications that link 2D to 3D. We would also like to explore the potential of P-maps for enhance other low-level image processing operations, such as edge detection, where the saliency of the edge is augmented or amplified by the P-channel.

Acknowledgement We would first like to thank all the reviewers for their valuable comments and suggestions. This work is supported in part by grants from National 973 Program (2015CB352501), NSFC-ISF(61561146397), Shenzhen Knowledge innovation program for basic research (JCYJ20150402105524053).

References

1. Aubry, M., Maturana, D., Efros, A., Russell, B.C., Sivic, J.: Seeing 3D chairs: exemplar part-based 2D-3D alignment using a large dataset of CAD models. In: Proc. CVPR, IEEE (2014) 3762–3769

2. Chen, Y.T., Liu, X., Yang, M.H.: Multi-instance object segmentation with occlusion handling. In: Proc. CVPR. (2015) 3470–3478
3. Chen, T., Cheng, M.M., Tan, P., Shamir, A., Hu, S.M.: Sketch2photo: Internet image montage. ACM Trans. Graph. **28** (2009) 124:1–124:10
4. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. PAMI **24** (2002) 603–619
5. Dai, J., He, K., Sun, J.: Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In: Proc. ICCV. (2015)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. CVPR. Volume 1., IEEE (2005) 886–893
7. Dosovitskiy, A., Springenberg, J.T., Brox, T.: Learning to generate chairs with convolutional neural networks. In: Proc. CVPR, IEEE (2015) 1538–1546
8. Dosovitskiy, A., Brox, T.: Inverting visual representations with convolutional networks. arXiv preprint arXiv:1506.02753 (2015)
9. Everingham, M., Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV **88** (2009) 303–338
10. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. IJCV **59** (2004) 167–181
11. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proc. CVPR, IEEE (2014) 580–587
12. Girshick, R.: Fast r-cnn. In: Proc. ICCV. (2015)
13. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Simultaneous detection and segmentation. In: ECCV. Springer (2014) 297–312
14. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. In: ECCV. Springer (2014) 346–361
15. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts? IEEE PAMI **26** (2004) 147–159
16. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. In Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., Weinberger, K., eds.: NIPS. Curran Associates, Inc. (2011) 109–117
17. Krahenbuhl, P., Koltun, V.: Learning to propose objects. In: Proc. CVPR. (2015) 1574–1582
18. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proc. NIPS. (2012) 1097–1105
19. Li, Y., Su, H., Qi, C.R., Fish, N., Cohen-Or, D., Guibas, L.J.: Joint embeddings of shapes and images via CNN image purification. ACM Trans. Graph. (2015)
20. Liang, X., Wei, Y., Shen, X., Yang, J., Lin, L., Yan, S.: Proposal-free network for instance-level object segmentation. arXiv preprint arXiv:1509.02636 (2015)
21. Liang, X., Wei, Y., Shen, X., Jie, Z., Feng, J., Lin, L., Yan, S.: Reversible recursive instance-level object segmentation. arXiv preprint arXiv:1511.04517 (2015)
22. Li, Y., Sun, J., Tang, C.K., Shum, H.Y.: Lazy snapping. ACM Trans. Graph. **23** (2004) 303–308
23. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proc. CVPR. (2015) 3431–3440
24. Mortensen, E.N., Barrett, W.A.: Intelligent scissors for image composition. In: Proc. 22nd Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '95, New York, NY, USA, ACM (1995) 191–198
25. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: Proc. ICCV. (2015)

26. Papandreou, G., Chen, L.C., Murphy, K., Yuille, A.L.: Weakly-and semi-supervised learning of a dcnn for semantic image segmentation. In: Proc. ICCV. (2015)
27. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: Proc. NIPS. (2015)
28. Rother, C., Kolmogorov, V., Blake, A.: "GrabCut": Interactive foreground extraction using iterated graph cuts. ACM Trans. Graph. **23** (2004) 309–314
29. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A., Fei-Fei, L.: Imagenet large scale visual recognition challenge. IJCV **115** (2015) 211–252
30. Silberman, N., Sontag, D., Fergus, R.: Instance segmentation of indoor scenes using a coverage loss. In: ECCV. Springer (2014) 616–631
31. Su, H., Qi, C.R., Li, Y., Guibas, L.J.: Render for CNN: Viewpoint estimation in images using CNNs trained with rendered 3D model views. In: Proc. ICCV. (2015)
32. Su, H., Huang, Q., Mitra, N.J., Li, Y., Guibas, L.: Estimating image depth using shape collections. ACM Trans. Graph. **33** (2014) 37:1–37:11
33. Su, H., Yi, E., Savva, M., Chang, A., Song, S., Yu, F., Li, Z., Xiao, J., Huang, Q., Savarese, S., Funkhouser, T., Hanrahan, P., Guibas, L.: Shapenet: An ongoing effort to establish a richly-annotated, large-scale dataset of 3d shapes. http://shapenet.org (2015)
34. Uijlings, J.R.R., Sande, K.E.A., Gevers, T., Smeulders, A.W.M.: Selective search for object recognition. IJCV **104** (2013) 154–171
35. Xu, K., Zheng, H., Zhang, H., Cohen-Or, D., Liu, L., Xiong, Y.: Photo-inspired model-driven 3D object modeling. ACM Trans. Graph. **30** (2011) 80:1–80:10
36. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., Torr, P.H.S.: Conditional random fields as recurrent neural networks. In: Proc. ICCV. (2015)
37. Zheng, Y., Chen, X., Cheng, M.M., Zhou, K., Hu, S.M., Mitra, N.J.: Interactive images: Cuboid proxies for smart image manipulation. ACM Trans. Graph. **31** (2012) 99:1–99:11
38. Zitnick, C., Dollàr, P.: Edge boxes: Locating object proposals from edges. In: ECCV. Volume 8693. Springer International Publishing (2014) 391–405