Hierarchical Scene Annotation

Michael Maire[1]
mmaire@caltech.edu
Stella X. Yu[2]
stellayu@icsi.berkeley.edu
Pietro Perona[1]
perona@caltech.edu

[1] California Institute of Technology
1200 East California Blvd
Pasadena, CA 91125

[2] UC Berkeley / ICSI
1947 Center St. Ste. 600
Berkeley, CA 94704

Abstract

We present a computer-assisted annotation system, together with a labeled dataset and benchmark suite, for evaluating an algorithm's ability to recover hierarchical scene structure. We evolve segmentation groundtruth from the two-dimensional image partition into a tree model that captures both occlusion and object-part relationships among possibly overlapping regions. Our tree model extends the segmentation problem to encompass object detection, object-part containment, and figure-ground ordering.

We mitigate the cost of providing richer groundtruth labeling through a new web-based annotation tool with an intuitive graphical interface for rearranging the region hierarchy. Using precomputed superpixels, our tool also guides creation of user-specified regions with pixel-perfect boundaries. Widespread adoption of this human-machine combination should make the inaccuracies of bounding box labeling a relic of the past.

Evaluating the state-of-the-art in fully automatic image segmentation reveals that it produces accurate two-dimension partitions, but does not respect groundtruth object-part structure. Our dataset and benchmark is the first to quantify these inadequacies. We illuminate recovery of rich scene structure as an important new goal for segmentation.

1 Introduction

Supervised datasets play a central role in driving computer vision research. In multiple areas, just a handful of labeled datasets serve to inspire and benchmark progress over a timescale of many years. In object recognition, the Caltech101 [14], ImageNet [9], and PASCAL [12] datasets appear as focal points for work over the past decade. For edge detection and image segmentation, the Berkeley segmentation dataset (BSDS) [20] serves as the single standard benchmark [21]. The type of annotation available for each of these datasets determines the particular visual subtasks to which they are applicable. Object bounding boxes can benchmark detection algorithms, but aren't much use for training or evaluating segmentation. Segmented objects are more widely useful, but more time-consuming to annotate. What visual tasks are most important and what level of annotation detail is appropriate?

We present an alternative to thinking about dataset annotation in terms of a restricted set of visual tasks. A key motivation is the observation that recent work blurs the boundaries between tasks. For example, segmentation is often employed as a preprocessing step for object detection, replacing sliding windows with region candidates [5, 16]. Related efforts

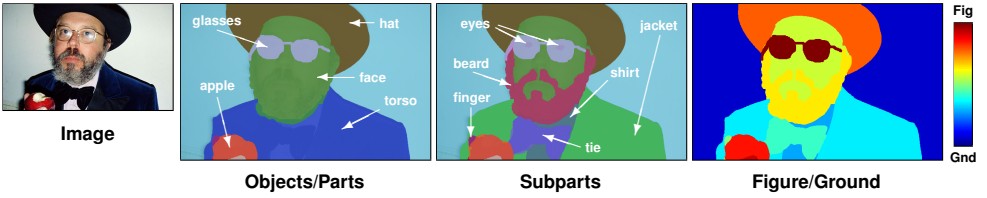

Figure 1: **Hierarchical annotation.** At the coarsest level, this scene contains two objects: a man standing in front of a wall. Looking in detail, we subdivide the man into regions for his hat, glasses, face, torso, and the apple he holds. In even more detail, his face consists of eyes, skin, and beard, and his torso is covered by a shirt, tie, and jacket. Traditional segmentation datasets label a single two-dimensional region partition. Our annotation model, outlined in Figure 2, captures the object-part-subpart decomposition, as well as the occlusion relationships (*e.g.* apple in front of jacket, glasses in front of face) present in the scene.

generate object candidates prior to invoking category-specific knowledge [1, 8, 11]. Such usage suggests viewing the output of segmentation algorithms in an object-centric context.

Segmentation is also not the only perceptual task one might perform prior to object detection. A large body of work focuses on identifying occlusion boundaries or figure-ground relationships, either building on segmentation output, or in conjunction with the segmentation task [17, 18, 19, 22, 24]. Some of this work [18, 19, 22] depends on a re-annotated subset of the BSDS with figure-ground labels on boundaries.

In the object detection realm, part-based models [7, 15] are among the current best performing techniques on the PASCAL benchmark [12]. Of these, the poselets-based approach [7] benefits from extensive supervised part annotation during training [6]. Subsequent work extends poselets to other domains [13]. Part-based models also continue their long history of relevance to the problem of articulated human pose estimation [28].

We propose a simple annotation model, in the form of a region tree, that captures a nearly complete description of any scene in terms of objects, parts, object-part containment, segmentation, and figure-ground or occlusion ordering. Our key observation is that a hierarchical, rather than flat, groundtruth representation allows one to subsume these disparate aspects into a single framework. Figure 1 illustrates the level of detail our annotation model encompasses for a typical scene, while Section 2 discusses model technical specifics.

To enable humans to efficiently create detailed annotation, we build an interactive labeling tool that provides computer-assistance at each step of the process. Similar to LabelMe [23], our tool runs as a web application, facilitating large-scale distributed annotation projects. However, both model expressiveness and level of automated assistance to users are far beyond any existing system. Our system offers interactive visualization, enforces model invariants during editing, and frees users from the tedious and time-consuming process of manual segmentation. We accelerate labeling by mixing the state-of-the-art in machine segmentation [4] with human supervision. Section 3 gives details.

Section 4 describes a dataset, annotated using our tool, containing scenes of greater complexity and scale variation than those typical of the BSDS. Section 5 uses this dataset to evaluate the gPb-UCM [4] algorithm in terms of its ability to match groundtruth object-part-subpart hierarchical boundaries. Our benchmark is the first to directly analyze hierarchical object segmentation and is complementary to the BSDS. Results indicate a need for further advancements in generic segmentation algorithms for complex scenes. Section 6 concludes.

2 Scene Model

A flat 2D partition of an image fails to capture full scene structure, since it must choose a fixed level of detail at which to label parts (*e.g.* face vs eyes/beard in Figure 1). Switching from a region partition to a region tree allows groundtruth annotation to capture multiple levels of detail. With a node for each region, the tree connects parts or subparts as children of their immediately containing object or part. Moving to a doubly ordered tree offers one additional degree of freedom: the ordering of child nodes appearing beneath a common parent. We exploit this ordering to encode local occlusion relationships between regions.

Specifically, we model a scene as a set $S = \{R_1, R_2, ..., R_n\}$ where each $R_i \subseteq I$ is a region in the image I. In general, $R_i \cap R_j$ may be nonempty. Regions may overlap in all possible ways (no overlap, partial intersection, or full containment). We organize regions $\{R_i\}$ into a tree T such that the root node of T is a dummy node representing the entire scene and each region R_i appears exactly once as a non-root node of T. Let $N(R_i)$ denote the node

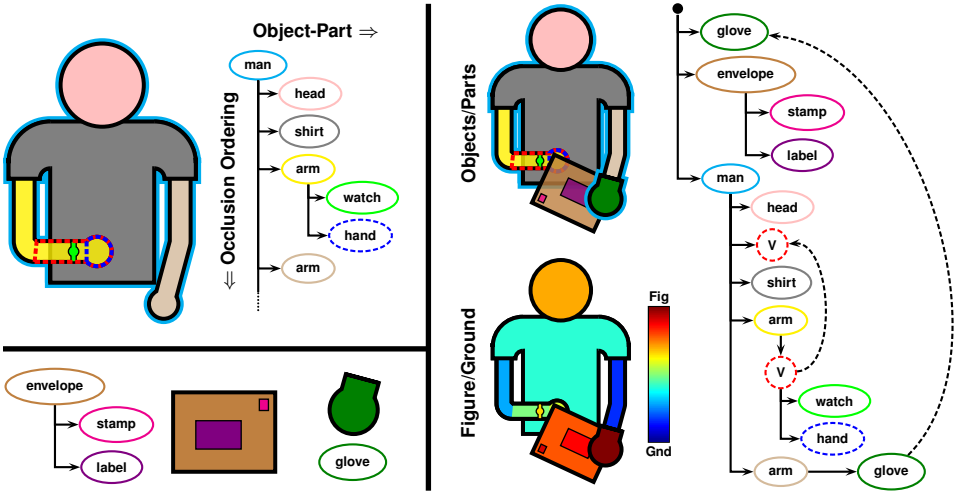

Figure 2: **Scene model.** We organize image pixels into (possibly overlapping) regions and map them to nodes in a doubly ordered tree. Parent-child links (solid arrows) denote region containment and semantic object-part or part-subpart relationships. The (top to bottom) order of siblings beneath a common parent resolves (figure to ground) occlusion ambiguities between any touching or overlapping regions. *Upper Left:* Labels indicate the object-part and occlusion axes for the man's partial annotation tree. His head occludes his shirt, which occludes both arms at the sleeves. Not captured is the occlusion of the shirt by the forearm (dashed red outline). The hand is a semantically meaningful virtual node. Virtual nodes (dashed ovals) denote subregions that do not introduce interior boundaries with their parent; there is no visible boundary along the wrist. *Lower Left:* Example trees for simple objects. *Right:* The man puts on the glove and picks up the envelope. We introduce the notion of virtual links (dashed arrows) to capture self-occlusions. We add virtual node V to capture the forearm region and a virtual link to move V in front of the shirt. Another virtual link moves the glove, now an arm subpart, in front of the envelope. Ignoring nodes with incoming virtual links, tree traversal recovers the object-part hierarchy. Moving nodes to the destinations of outgoing virtual links, preorder traversal recovers global figure-ground ordering.

of T corresponding to region R_i. $N(R_j)$ is set to be the parent node of $N(R_i)$ iff all of the following conditions are met: (1) $R_j \supset R_i$, (2) R_j and R_i have an object-part relationship, and (3) $\nexists R_k : R_j \supset R_k \supset R_i$ and R_j, R_k, R_i have an object-part-subpart relationship, respectively. If for R_i, no region R_j satisfies all three conditions, then we set $N(R_i)$ to be a child of the root node. Simply stated, T decomposes the the scene into a multilevel object-part hierarchy.

Within T, region containment is only enforced between descendent and ancestor nodes. Provided $N(R_i)$ and $N(R_j)$ do not lie on the same path from the root to a leaf, we may have both $R_i \not\subseteq R_j$ and $R_j \not\subseteq R_i$, but $R_i \cap R_j \neq \emptyset$. Since at each pixel in the image there is at most one visible object, this situation indicates an occlusion of one object by another in the scene.

To resolve occlusions, we augment T with a ranked ordering on each set of sibling nodes. Suppose node $N(R) \in T$ has m children in T denoted by $N(R_1), N(R_2), \ldots, N(R_m)$ (abusing notation and using indices $1, \ldots, m$ here for convenience). Then we store a mapping $O(R_i)$ that uniquely assigns the integers in $[1, m]$ to the regions $R_1, \ldots, R_m$. Semantically, $O(\cdot)$ provides a local occlusion ordering. Specifically, if $R_i \cap R_j \neq \emptyset$, then R_i occludes R_j if $O(R_i) < O(R_j)$, and R_j occludes R_i if $O(R_i) > O(R_j)$. If $R_i \cap R_j = \emptyset$, then they do not occlude one another and we do not care about their relative ordering assignments.

Given T and O, we can render the visible regions in the scene as follows. Working up from the leaves of T, we recursively project each set of child regions onto its parent, with the order of projection among siblings given in terms of decreasing $O(\cdot)$ value. Equivalently stated, we use the tree structure to translate the relative ordering between siblings into a global figure-ground order for the scene. A preorder tree traversal implements this operation.

Setting the color of visible regions by the order in which this traversal draws them reveals the global figure-ground structure as a heat map. Drawing visible boundaries instead, and weighting them by the level (object, part, subpart, ...) of the region they enclose, produces an ultrametric contour map (UCM) [2]. This UCM is a groundtruth hierarchical segmentation whose finest level consists of a partition of the image into visible regions. Section 5 explains the use of the groundtruth UCM in benchmarking machine segmentations.

At this point, our model consists of regions, with associated semantic labels, organized into a tree with structure along two axes: parent-child for object-part and sibling order for occlusion. Two minor extensions can improve model flexibility. First, one may want to label semantic parts that do not correspond to regions with visible boundaries. To do so without corrupting the UCM, we refer to such parts as virtual regions and flag the corresponding tree nodes. Second, our figure-ground recovery algorithm assumes the object-part and occlusion hierarchies are the same; there is no self-occlusion or inter-object wrap-around. We introduce the notion of virtual links to indicate exceptions to this assumption. They allow a tree node to appear at one location in the object-part decomposition and a different location during figure-ground recovery. Figure 2 walks through a concrete example of the full model.

We emphasize that virtual nodes and virtual links, while enabling absolute correctness, are rarely necessary. It is appropriate to balance their added complexity during the annotation process against the need to handle the special cases they address. Without them, the model still provides a good first-order approximation of object-part and occlusion relationships, and represents a great leap in expressiveness over a flat 2D region partitioning.

3 Annotation Software

Since even our most basic model presents a nontrivial annotation task, we develop a custom annotation tool that runs as a web application. Figure 3 shows the tool's scene overview

Figure 3: **Annotation interface.** Our web-based image annotation tool renders partially transparent color-coded regions (center) according to the occlusion order determined by their position in the tree (right panel). The right panel serves as a graphical interface for controlling the level of detail displayed by expand/collapsing nodes (A). More importantly, it enables rearrangement of tree structure by clicking and dragging nodes to different locations in the tree (B). Here, the user should correct the ordering of jacket and pants by dragging jacket to be behind (occluded by) pants. Buttons next to each region (C) swap to the region editing mode detailed in Figures 4 and 5 The slider in the left panel (D) peels off occlusion layers, allowing quick visualization of global figure/ground structure.

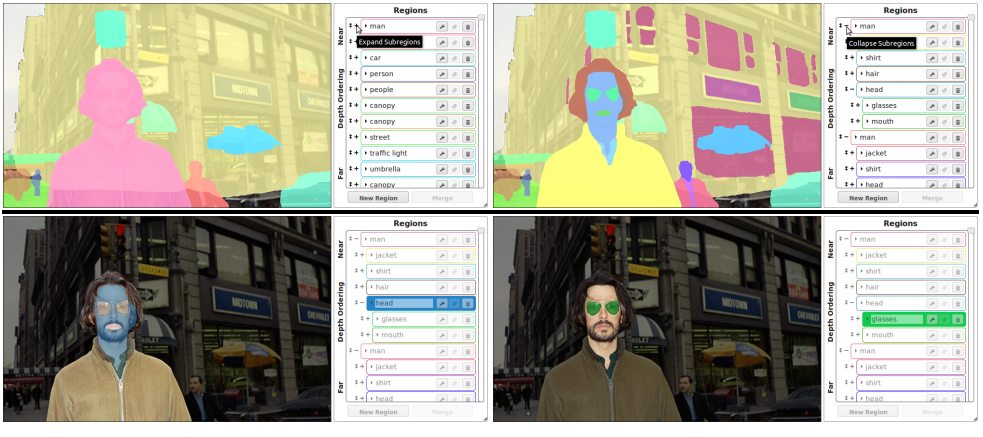

Figure 4: **Managing containment.** *Top:* Expanding and collapsing tree nodes permits easy navigation of the object-part hierarchy. *Bottom:* When editing a region, the system enforces parent-child node containment invariants. The head must contain glasses and mouth (required areas brightened) and cannot include any pixels not covered by the man's body (disallowed areas darkened). Similarly, glasses must be contained within head. These constraints are dynamic and track the tree structure. For example, dragging and dropping glasses to a different parent node in the tree would alter them in real time.

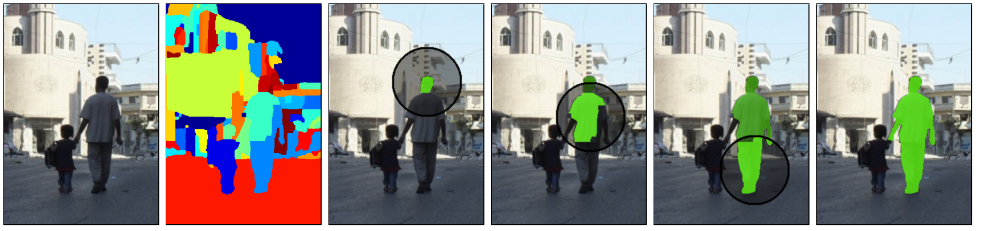

Figure 5: **Computer-assisted region annotation.** As the user moves a brush over the image, it dynamically snaps to the shape of the segment in the precomputed oversegmentation located beneath the brush center. Clicking and dragging the brush over the interior of the person expands the region to match the computer-generated boundaries. A few more clicks touch-up the result to include the missing arm. Without tediously hand-tracing boundaries, and in time comparable to drawing a bounding box, the user obtains a pixel-perfect result.

mode. As a user edits, the tool renders the scene model in real time, resolving occlusions using the algorithm described in the previous section. It provides a graphical interface for managing the object-part hierarchy, with the ability to drag and drop regions to rearrange scene structure. Moving a region relative to its siblings changes its occlusion ordering. Dropping a region onto a new parent node detaches and reattaches the subtree rooted at that region.

The system enforces the crucial invariant that the pixels covered by each child region are a subset of those covered by its parent, preserving the consistency of the hierarchy. Drag and drop operations check that the region being moved fits within its target parent. When editing a region, we enforce these constraints in the form of masks of forbidden and required pixels, as shown in Figure 4. The combination of visual feedback, drag and drop flexibility, and automatic constraint enforcement guides the user to an intuitive annotation style: define major objects in the scene, sort them by occlusion, and recursively subdivide them into parts.

The most time-consuming aspect of producing groundtruth segmentations is the tedious process of precisely tracing region outlines. We sidestep this problem by using computer vision to handle the vast majority of boundary localization work. Performance of machine algorithms for 2D image segmentation has drastically improved as measured by boundary localization accuracy [21] on the BSDS [20]. At the risk of introducing slight bias, we use an oversegmentation computed by the gPb-UCM algorithm [3] in order to assist annotators.

An important caveat is that machine segmentation is only used locally during the annotation process. When outlining a region, a user may choose to snap to a machine-computed boundary. The user may also manually draw part of the region in order to correct errors in the machine segmentation. At no point in time, however, does the machine segmentation influence the hierarchical structure the annotator places on the scene. Figure 5 illustrates the drastic annotation speed increase resulting from computer assistance with local boundaries.

4 Dataset

We use our annotation tool to groundtruth a collection of 97 photographs by artist Stephen Shore [25, 26], a dataset previously used in experiments measuring the importance of objects in complex scenes [10, 27]. This dataset is appropriate for a hierarchical scene segmentation benchmark as it contains a diversity of scene types. A typical image has more visible objects

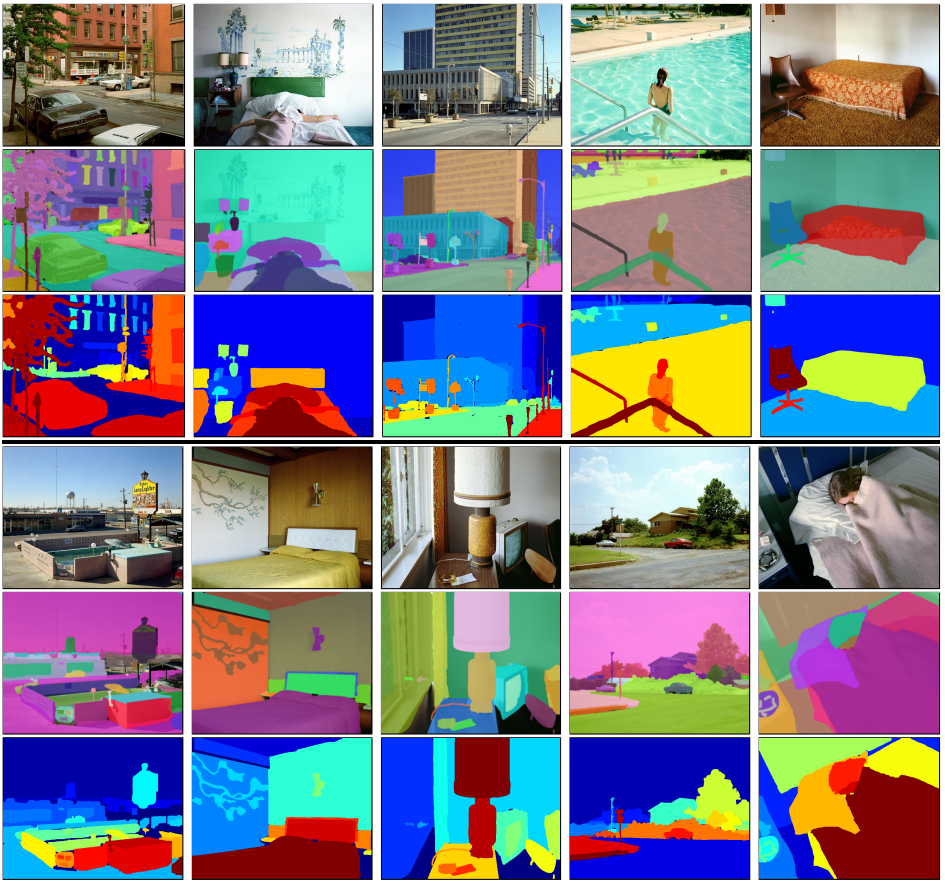

Figure 6: **Annotated scene dataset.** We show a sample of images from our dataset. Beneath each image lies a visualization of its groundtruth hierarchical segmentation, rendered at the finest level of part detail, followed by the corresponding groundtruth occlusion (figure-ground) layering. In the figure-ground map, red indicates more figural regions.

Figure 7: **Example object-part hierarchies.** Beneath each image, we visualize the first two or three levels of its groundtruth hierarchical segmentation (from objects to subparts).

than the average BSDS image, and contains detail across a wider scale range. Dataset size is comparable to the 100 images in the original BSDS test set. To reduce annotation task complexity, we ignore virtual nodes and links. Figures 6 and 7 show a sample of the dataset.

5 Benchmarks

As discussed in Section 2, our annotations can be turned into a groundtruth UCM. This UCM weights each boundary according to the level of detail at which it appears in the object-part hierarchy. Instead of having binary groundtruth boundary maps, as in the case of the BSDS, we have real-valued hierarchical groundtruth. This permits direct evaluation of the quality of the hierarchical output of machine segmentation algorithms. We can test whether the machine hierarchy arranges regions according to object-part containment.

Specifically, we examine the order in which boundaries are recalled as one varies an algorithm's boundary detection threshold. The ideal algorithm would recover the occlusion boundaries between the top-level objects in the scene first (level 1), followed by large-scale object-part boundaries (level 2), followed by finely detailed part-subpart boundaries (level 3). Recovering exactly and only the top-level object boundaries is precisely the category independent object detection problem [11], so our benchmark also evaluates that task.

Figure 8 reports overall boundary detection performance followed by level recovery order for the state-of-the-art gPb-UCM segmentation algorithm. Though our groundtruth superpixels are biased toward gPb-UCM (computer-assisted region creation), our groundtruth hierarchy is not. The middle plot shows that instead of recovering levels in order, recovery of all levels increases together, indicating a mismatch between the gPb-UCM hierarchy and the groundtruth hierarchy. Restricting analysis to the portrait scenes (right plot) yields a small improvement on this easier subset. Figure 9 provides a visual comparison.

These results are the first that directly examine the quality of the hierarchical output of a leading image segmentation approach. The gap between gPb-UCM and the groundtruth object-part hierarchies serves as a call for further research into hierarchical scene segmentation and underscores the importance of our annotation tool and dataset.

Although we do not explore it here, it is also possible to use our dataset to benchmark figure/ground assignment as we have groundtruth occlusion ordering on the regions.

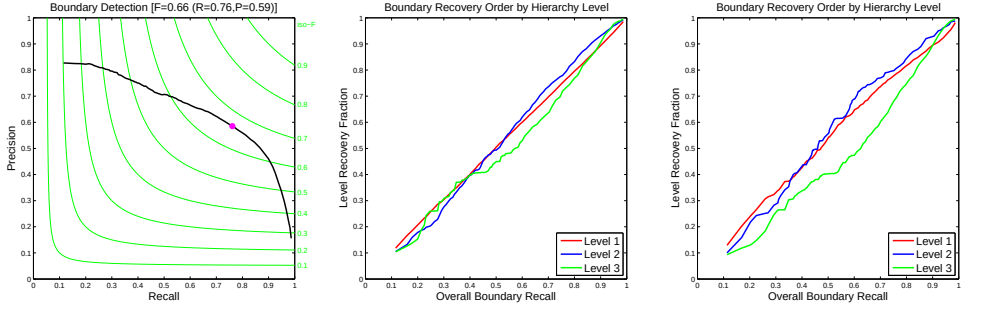

Figure 8: **Hierarchical segmentation benchmark.** Overall boundary precision-recall (left) and hierarchical boundary recovery for all scenes (middle) and portraits only (right).

Figure 9: **Comparison of hierarchical groundtruth and machine segmentations.** Residuals show differences after computing correspondence between the optimally thresholded gPb-UCM and the groundtruth UCM. Green indicates missed boundaries, red extraneous detections, and gray correct boundary localization, but incorrect hierarchical level assignment. Color intensity correlates with the magnitude of the error.

6 Conclusion

Our novel annotation representation, based on region trees, captures multiple aspects of scene structure. With a sophisticated labeling tool, we create an object-part segmentation dataset and benchmark that demonstrates deficiencies in current work and challenges for future work. Our source code, annotation tool, dataset, and benchmark are available online.

Acknowledgments. ONR MURI N00014-10-1-0933 and ARO/JPL-NASA Stennis NAS7.03001 supported this work. Part of Stella Yu's work was supported by NSF CAREER IIS-1257700. Thanks to Alex Jose and Piotr Dollar for helpful discussion on user interfaces for segmentation.

References

[1] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. What is an object? *CVPR*, 2010.

[2] Pablo Arbeláez. Boundary extraction in natural images using ultrametric contour maps. *POCV*, 2006.

[3] Pablo Arbeláez, Michael Maire, Charless Fowlkes, and Jitendra Malik. From contours to regions: An empirical evaluation. *CVPR*, 2009.

[4] Pablo Arbeláez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *PAMI*, 2011.

[5] Pablo Arbeláez, Bharath Hariharan, Chunhui Gu, Saurabh Gupta, Lubomir Bourdev, and Jitendra Malik. Semantic segmentation using regions and parts. *CVPR*, 2012.

[6] Lubomir Bourdev and Jitendra Malik. Poselets: Body part detectors trained using 3d human pose annotations. *ICCV*, 2009.

[7] Thomas Brox, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Object segmentation by alignment of poselet activations to image contours. *CVPR*, 2011.

[8] Joao Carreira and Cristian Sminchisescu. Constrained parametric min-cuts for automatic object segmentation. *CVPR*, 2010.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *CVPR*, 2009.

[10] Wolfgang Einhäuser, Merrielle Spain, and Pietro Perona. Objects predict fixations better than early saliency. *Journal of Vision*, 2008.

[11] Ian Endres and Derek Hoiem. Category independent object proposals. *CVPR*, 2010.

[12] Mark Everingham, Luc van Gool, Chris Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes (VOC) challenge. *IJCV*, 2010.

[13] Ryan Farrell, Om Oza1, Ning Zhang, Vlad I. Morariu1, Trevor Darrell, and Larry S. Davis. Birdlets: Subordinate categorization using volumetric primitives and pose-normalized appearance. *ICCV*, 2011.

[14] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *CVPR Workshop on Generative-Model Based Vision*, 2004.

[15] Pedro Felzenszwalb, Ross Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 2009.

[16] Chunhui Gu, Joseph Lim, Pablo Arbeláez, and Jitendra Malik. Recognition using regions. *CVPR*, 2009.

[17] Derek Hoiem, Andrew N. Stein, Alexei A. Efros, and Martial Hebert. Recovering occlusion boundaries from a single image. *ICCV*, 2007.

[18] Ido Leichter and Michael Lindenbaum. Boundary ownership by lifting to 2.1D. *ICCV*, 2009.

[19] Michael Maire. Simultaneous segmentation and figure/ground organization using angular embedding. *ECCV*, 2010.

[20] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *ICCV*, 2001.

[21] David Martin, Charless Fowlkes, and Jitendra Malik. Learning to detect natural image boundaries using local brightness, color and texture cues. *PAMI*, 2004.

[22] Xiaofeng Ren, Charless Fowlkes, and Jitendra Malik. Figure/ground assignment in natural images. *ECCV*, 2006.

[23] Bryan Russell, Antonio Torralba, Kevin Murphy, and William Freeman. Labelme: A database and web-based tool for image annotation. *IJCV*, 2007.

[24] Ashutosh Saxena, Sung H. Chung, and Andrew Y. Ng. 3-d depth reconstruction from a single still image. *IJCV*, 2008.

[25] Stephen Shore. *Uncommon Places*. Aperture, 2005.

[26] Stephen Shore. *American Surfaces*. Phaidon Press, 2008.

[27] Merrielle Spain and Pietro Perona. Measuring and predicting object importance. *IJCV*, 2010.

[28] Yi Yang and Deva Ramanan. Articulated pose estimation using flexible mixtures of parts. *CVPR*, 2011.