Scalable Cascade Inference for Semantic Image Segmentation

Paul Sturgess[1]
paul.sturgess@brookes.ac.uk
L'ubor Ladický[2]
lubor@robots.ox.ac.uk
Nigel Crook[1]
ncrook@brookes.ac.uk
Philip H. S. Torr[1]
philiptorr@brookes.ac.uk

[1] Department of Computing
Oxford Brookes University
Oxford,UK

[2] Department of Engineering Science
University of Oxford
Oxford, UK

Abstract

Semantic image segmentation is a problem of simultaneous segmentation and recognition of an input image into regions and their associated categorical labels, such as person, car or cow. A popular way to achieve this goal is to assign a label to every pixel in the input image and impose simple structural constraints on the output label space. Efficient approximation algorithms for solving this labelling problem such as α-expansion have, at best, linear runtime complexity with respect to the number of labels, making them practical only when working in a specific domain that has few classes-of-interest. However when working in a more general setting where the number of classes could easily reach tens of thousands, sub-linear complexity is desired. In this paper we propose meeting this requirement by performing cascaded inference that wraps around the α-expansion algorithm. The cascade both divides the large label set into smaller more manageable ones by way of a hierarchy, and dynamically subdivides the image into smaller and smaller regions during inference. We test our method on the SUN09 dataset with 107 accurately hand labelled classes.

1 Introduction

Semantic image segmentation (SIS) is a problem of simultaneous segmentation and recognition of an input image into regions and their associated categorical labels, such as person, car or cow. A popular way to achieve this goal is to assign a label to every pixel in the input image and impose simple structural constraints on the output label space. Such approaches have been successfully formulated as pairwise conditional random fields (CRF) [20] and higher order CRFs [14, 16]. These approaches are now practically solvable for some problems due to advances in inference techniques [8, 11, 15]. Currently the α-expansion [2] algorithm has proved to be perhaps the most efficient approximation algorithm for the SIS problem and is amongst the state-of-the art for quantitative performance [5, 16]. Empirically the algorithm's runtime is linear in the number of labels, making it practical only when working in a specific domain that has few classes-of-interest ($10-20$ for example). However when working

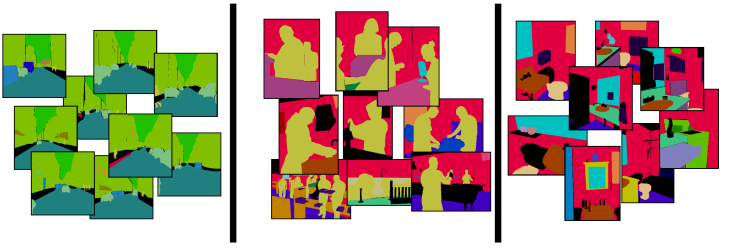


Figure 1: **Image Label Domains**: Taking the domain of labels from all the groups of depicted ground truth images together gives many labels, but each group has a smaller subset, and each image has yet a smaller subset of labels from its group. In fact the SUN09 dataset has only seven labels on average per image, and the full domain of labels is 107. This makes using a cascade, that can reject sets of labels, an attractive approach for scaling up the inference methods for semantic image segmentation to many labels.

in a more general setting where the number of classes could easily reach tens of thousands [5], sub-linear complexity is required. In this paper we propose to meet this requirement by dividing the large label set into smaller more manageable ones, and then only solving for some of these subsets. Since the SIS problem is concerned with categorical labels a natural way to subdivide the label set is by building a hierarchy, or taxonomy. Given a hierarchy we propose a cascade architecture that can reject whole portions of the label space at the early stages of the optimisation, which could be considered a type of *energy-aware* variable selection process [13]. We also dynamically subdivide the image into smaller and smaller regions during inference to gain further efficiency. The use of a cascade is motivated by the observation that even with a large label domain, a single image will usually only contain a small subset of classes. This can easily seen by viewing Fig. 1 where three sets of manually labelled images are shown, that have a variety of object classes. Even though over all the groups there is a relatively large number of classes there is no image with more than seven labels. We demonstrate the effectiveness of the approach with quantitative evaluation of performance on the SUN09 database [9] that has 107 labels. This dataset has the largest number of classes hand labelled by a single *expert* (as opposed to LabelMe [18]) for the semantic image segmentation application, making the ground truth of a high quality. Some examples can seen in Fig. 1 and Fig. 3.

2 The Potts Labelling Problem for Semantic Image Segmentation

The labelling over an undirected graph, $G(V,\mathcal{E})$, is defined as the function $f : V \rightarrow \Delta^{|V|}$, where Δ is a discrete label set, or domain, with associated metric distance $d : \Delta \times \Delta \rightarrow [0,1]$. In SIS, Δ is a categorical set of labels such as *car, van* and *horse*, and each vertex is associated with a pixel in the image. An assignment cost $c(v,l)$ is specified for each vertex $v \in V$ and label $l \in \Delta$. The cost of the solution $f : V \rightarrow \Delta^{|V|}$, denoted $Q(f)$, consists of the assignment cost and a set of edge weights w, that typically enforce consistency between a group of vertices. For SIS these groups usually consist of a set of 4 or 8 nearest pixel neighbours. The

overall cost that is to be minimised is given by:-

$$\text{minimize:} \quad Q(f) = \sum_{v \in V} c(v, f(v)) + \sum_{(u,v) \in \mathcal{E}} w(u,v) \cdot d(f(u), f(v)) \tag{1}$$

subject to:

$$\begin{aligned} & f : v \to \alpha && \forall v \in V,\ \exists \alpha \in \Delta \\ & d(\alpha, \alpha) = 0 && \forall \alpha \in \Delta \\ & d(\alpha, \beta) = d(\beta, \alpha) \geq 0 && \forall \alpha, \beta \in \Delta \\ & d(\alpha, \beta) \leq d(\alpha, \gamma) + d(\gamma, \beta) && \forall \alpha, \beta, \gamma \in \Delta, \\ & w(u,v) \geq 0 && \forall u, v \in V. \end{aligned}$$

We say that f^*_Δ is a local minimum for a Potts labelling problem that has been defined on the domain Δ and variables V.

Potts Distance Metric: In SIS it is important to preserve object boundaries so the metric distance function $d(f(u), f(v))$ that we use is the Potts model:

$$d(f(u), f(v)) = \begin{cases} 0 & \text{if } f(u) = f(v), \\ \lambda & \text{otherwise.} \end{cases} \tag{2}$$

The function $w(i, j)$ is an edge feature based on the difference in colours of neighbouring pixels [1], typically defined as:

$$w(u,v) = \theta_p + \theta_v \exp(-\theta_\beta ||I_u - I_v||_2^2), \tag{3}$$

where I_u and I_v are the color vectors of pixel u and v respectively. θ_p, θ_v, $\theta_\beta \geq 0$ are model parameters learned using training data. We refer the interested reader to [1, 17, 20] for more details. The Potts labelling problem is a specific case of a general Markov, or Conditional Random Field, that is particularly useful for SIS due to its edge persevering properties that are important at occlusion boundaries of objects.

3 Cascaded Inference

In order to obtain scalable SIS we propose a to perform cascade style inference as depicted in Fig. 2. In this section we specify the details of our approach. First we define two general functions:

$$\begin{aligned} \textit{variable selection} \quad & T_\delta : V \to V', \\ \textit{variable assignment} \quad & T_V : \delta \to \delta', \end{aligned}$$

that can be applied respectively to the variables (vertices) and the label domain of the cost function Q Eq. 1; T_δ transforms the current set of variables, given a domain; T_V modifies the current domain, given some variables. We can specify these transformation functions in different ways such that their evaluation performs a move for many move making algorithms, such as α-expansion and $\alpha\beta$-swap [2], γ-expansion [10], and multilabel-swap [3]. Here we

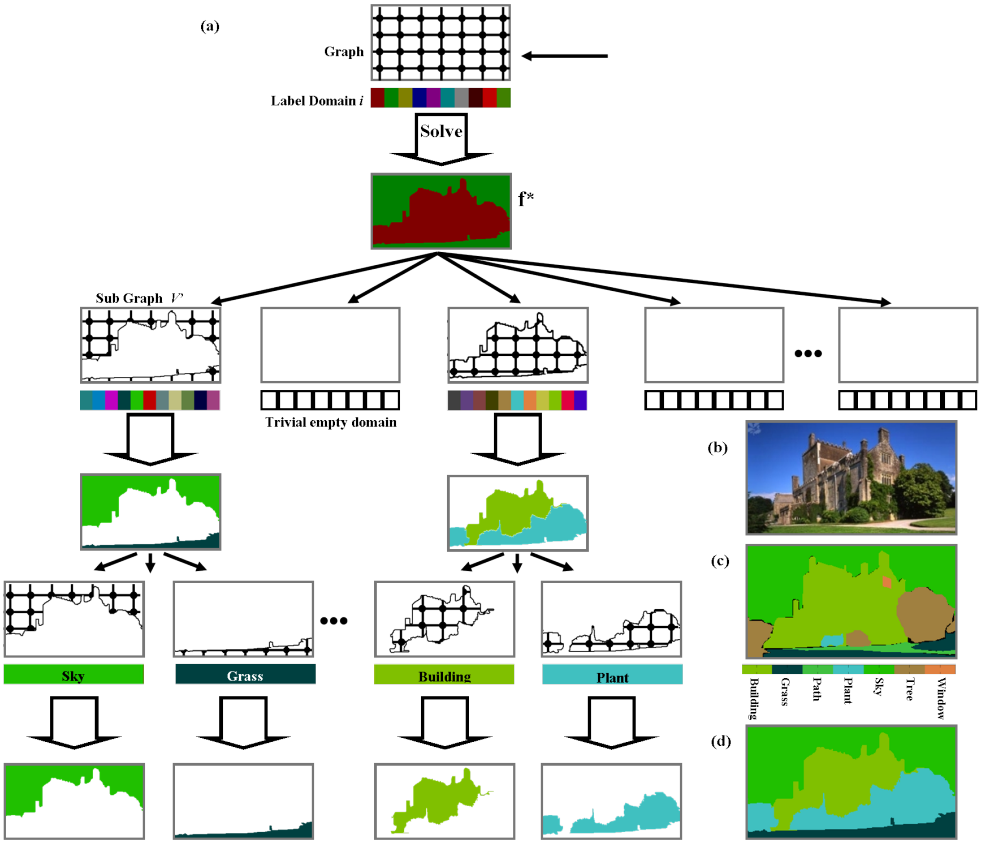


Figure 2: **Cascaded Inference**:(a) shows our cascade inference on a test example from the SUN09 dataset. Our cascade is able to reject large portions of the large label space resulting in sun-linear complexity at run-time. (b) shows the raw input image. (c) is the hand labelled ground truth and (d) is our final output after combining the results at the leaf nodes of the tree

are interested in specifying them in order to perform cascaded inference over a tree structured label space, or taxonomy τ. We define such a space with reference to an unstructured domain Δ as recursive subdivision into disjoint subsets δ such that the root node contains all the elements of Δ and leaf nodes contain the elementary labels $l \in \Delta$. Now, let δ denote a group of siblings, that is a set of children that share the same direct parent in the tree and thus forms a sub-domain of Δ. Also let $\pi(\delta)$ signify the domain that the shared parent belongs to, i.e. If the domain $\Delta = \{cat, dog, car, van\}$, then we could have the following groupings that form our tree; The head node would be $everything = \{cat, dog, car, van\}$ and it may have two children, such as $animal = \{cat, dog\}$ and $vehicle = \{car, van\}$. In turn these would then have two leaf nodes as children. Then $\pi(vehicle)$ points to the domain $everything$ and $\pi(dog)$ points to the label domain $\{cat, dog\}$. Thus a tree defines a set of domains $\{\delta_1, ...\delta_{n+1}\}$, where n is the number of sibling groups. For convenience we also maintain an index δ_i^j to the j^{th} elementary label contained within the i^{th} domain, i.e. $vehicle^1 = car$, as does $everything^3$. Given these notations variable selection and assignment based on a tree is

then defined as:

$$T_v(\delta) = \begin{cases} \delta_i & \text{if } \delta_i^j \in f^*_{\pi(\delta)} \text{ and } \delta \neq \emptyset \\ \emptyset & \text{otherwise,} \end{cases} \tag{4}$$

$$T_\delta(v) = v \in \{\mathbf{I}(f(T_\delta(v)) \neq \inf\} \tag{5}$$

where $\emptyset$ is the empty set, $\mathbf{I}$ is an indicator function, $f^*_{\pi(\delta)}$ is a given solution for the a labelling problem defined on the domain $\pi(\delta)$ and variables V' and

$$f(T_\delta(v)) = \begin{cases} c^\tau(v, f(v)) & \text{if } f(v) \in \delta \\ \infty & \text{otherwise} \end{cases}, \tag{6}$$

$$c^\tau(v, f(v)) = \arg\min_{f(v) \in \delta_i} c(v, f(v))). \tag{7}$$

For the first layer of the tree $f^*_{\pi(\delta)}$ is trivial since $\pi(\delta)$ is the single label domain of the head node, i.e. $f : V \to [1]$. This means that we have to solve a k label problem at the start of our cascade, where k is the number of children of the head node. In our running example this would be the $\{animal, vehicle\}$ domain on all variables V of the original graph. However when we visit all the nodes in the tree in the following fashion:-

for all i minimize:

$$Q(f) = \sum_{v \in T_{\delta_i}(v)} c^\tau(v, f(v)) + \sum_{(u,v) \in \mathcal{E}'} w(u,v) \cdot d(f(u), f(v)) \tag{8}$$

subject to:

$$\begin{aligned} & f : v \to \alpha && \forall v \in T_{\delta_i}, \exists \alpha \in T_v \\ & d(\alpha, \alpha) = 0 && \forall \alpha \in T_v \\ & d(\alpha, \beta) = d(\beta, \alpha) \geq 0 && \forall \alpha, \beta \in T_v \\ & d(\alpha, \beta) \leq d(\alpha, \gamma) + d(\gamma, \beta) && \forall \alpha, \beta, \gamma \in T_v \\ & w(u,v) \geq 0 && \forall u, v \in T_{\delta_i}, \end{aligned}$$

many sub-problems will be trivial such as:- no labels, $|\delta| = \emptyset$; a single label $|\delta| = 1$; no finite cost variables $\forall v \in T_v : c(v, f_\delta(v)) = \infty$. In these cases, we need not evaluate the function at all, saving computation time. In the cases where the cost is non-trivial with binary $\delta = \{\alpha, \beta\}$, or a multi-class domain with $|\delta| > 2$ and $\exists v \in T_v : c(v, f_\delta(v)) \neq \infty$. The cost function remains metric since we only modify the data term $c(.,.)$ [3], thus we can approximately solve it using α-expansion or other suitable methods. We show in 4 that our cascaded approach achieves a good approximation, $Q(\bigcup_{i \in leafs} Q(f^*_{\delta_i})) \approx Q(f^*_\Delta)$.

4 Empirical Evaluation

Dataset We are working in a fully supervised setting, so our experiments are performed on the 10,000 image subset of the SUN09 dataset [2]. The dataset has a sufficient amount of fully labelled per-pixel ground truth for a large set of 107 object classes. This is the same set as used for the experiments in [4]. We processed the polygon based layered representation to a single layer using the simple heuristics outlined in [18] as many images in the SUN09 dataset contain overlapping polygons. Examples of the dataset can be seen in Fig. 3 and Fig. 1.

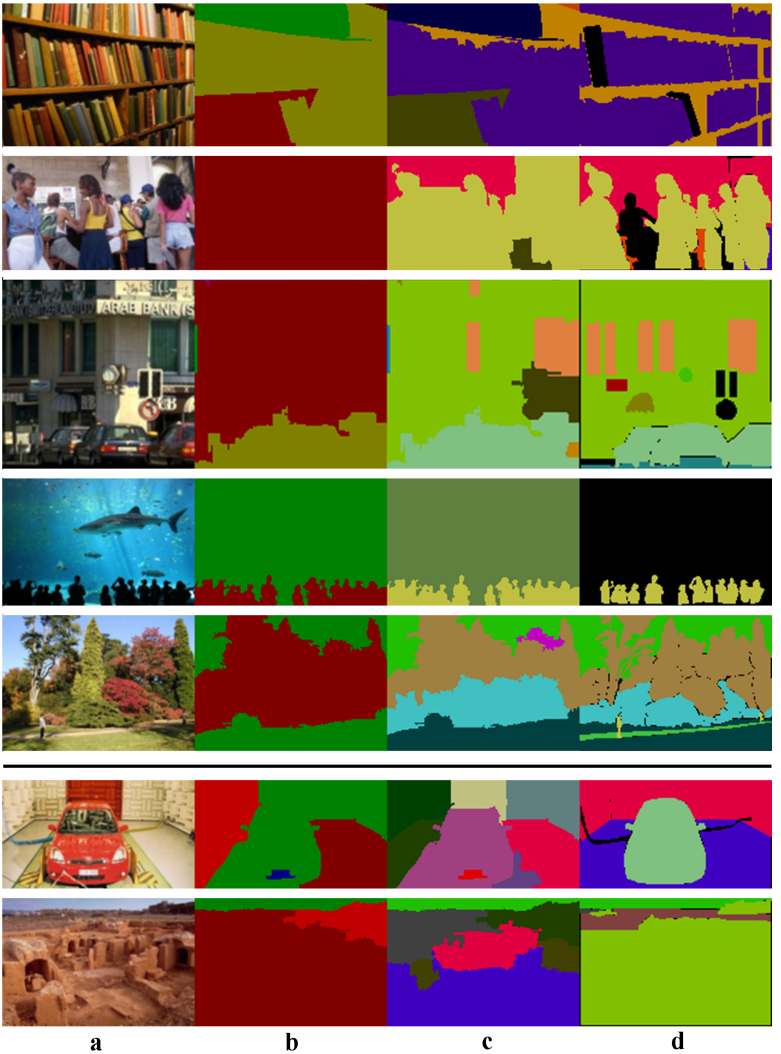


Figure 3: **Select Qualitative Results**: (a) shows some of the original images in the SUN09 test set, that are varied in content. (b) shows the partitioning of the image at the top level of feature sharing hierarchical clustering with around 11 labels per sub-domain. (c) shows the final output of our cascade. (d) The hand labelled ground truth images from the SUN09 dataset. The bottom two rows show some failure cases

Tree There number of possible k-ary tree's that can be generated from an n-label problem is exponential. This makes it infeasible to try all possible tree's and thus here we rely on prior knowledge. First we split our label set into roughly equally sized partitions, to give a balanced tree. In practice α-expansion has been demonstrated for SIS labelling problems with cardinality of around ten labels so we use this as our maximum node size. This then leads to a simple 2-layer tree. In order to decide which labels to group we use two common techniques. The first is the use of our own expert knowledge to manually group the labels,

denoted *CascALE Expert*. For the second tree we experiment with one based on visual similarity. To measure the visual similarity between different object classes we re-use the feature sharing matrix found whilst training the unary potentials with the joint-boost algorithm [21], denoted *CascALE Sharing*. We first find the ten classes with the most shared features and create ten clusters. Then we greedily add the other classes by visiting each cluster in a round robin fashion and add in the class that shares the most features with the current cluster. We do this until all classes have been assigned to a cluster.

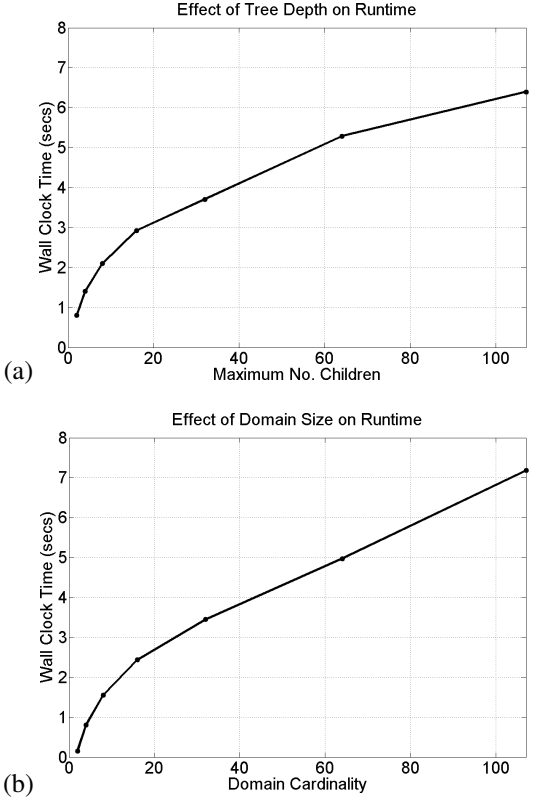


Figure 4: **Effects of tree depth**: (a) shows how the depth of our cascade effects its runtime on the full 107 labels. (b) shows how a flat approach scales as we increase the domain cardinality. Interestingly we see that the two graphs have similar trends. This indicates that we predict the runtime of our proposed cascade based on the number of labels and the depth of the tree.

Empirical Runtime We evaluated the our cascade in terms of efficiency with random tree's ranging in depth. Fig. 4 shows how the flat models runtime complexity grows linearly with the number of labels and that our approach grows linearly with the size of the sub-domains.

Empirical Accuracy In order to evaluate the effectiveness of our approach we compare to two baselines, the first being the contextual model of Torralba [4]. We use this as a

Method	Global	Recall	VOC	mean time secs
Torralba	33.0	10.6	6.2	-
ALE [19]	53.6	17.4	10.7	78
CascALE Expert	49.3	16.7	10.0	6
CascALE Sharing	52.8	15.2	9.6	4

Figure 5: **Global Results** This table shows how our method compared to state-of-the-art approaches

baseline as it frames our results well with respect to a very different method of performing recognition. They tackle the problem as one of bounding each object with a box, a.k.a, object detection [9]. In order to compare we take each bounding box as a segment and use layering heuristics to deal with overlapping regions, this follows the procedure of VOC challenge for evaluating detectors for SIS [7]. Our cascade essentially approximates inference over a pairwise cost function, this can be trivially extended to work with a higher order model AHCRF [16] that is amongst state-of-the art for the SIS problem [7] (In fact we implement our method as a wrapper around their publicly available ALE[1] library). This then gives us a strong second baseline to compare to that is competitive to with detector based approaches. We report the performance on global number of correct pixels, recall, and accuracy (a.k.a intersection-union (VOC) [12]) measures speed per-image and the averages over the whole dataset in Fig. 5. We also show close up of per-class recall over all the labels and the average recall per image over the whole test set in Fig. 4 as well as qualitative results in Fig. 3.

5 Conclusion

In this paper we have proposed a scalable cascaded inference method for the semantic image segmentation problem. The proposed approach is predictable in its runtime performance allowing the tree depth to be specified in advance given the available computational power. Also the per-class recall of our scalable approach is on par with the much less scalable flat approach. Our algorithm performs well due to the nature of scenes, that is that the usually only contain a few select categories. We exploited this by allowing many labels, and whole domains of labels to be rejected, furthermore we get a real boost by simultaneously partitioning the image space, and allowing different partitions to take on different sub-domains.

This work is supported by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, IST-2007-216886. P. H. S. Torr is in receipt of Royal Society Wolfson Research Merit Award.

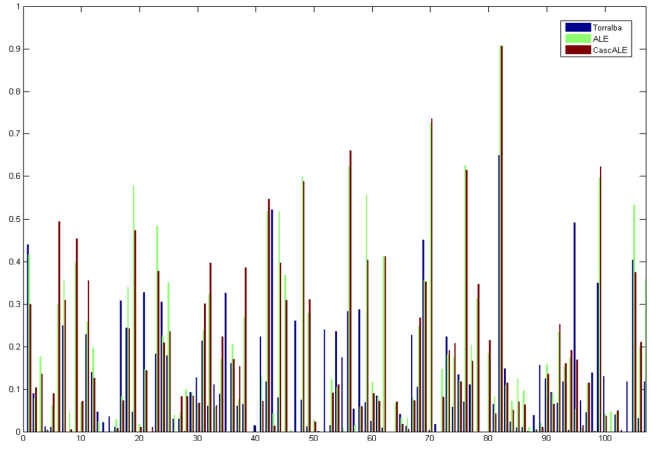
References

[1] Y. Boykov and M. P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *ICCV*, pages I: 105–112, 2001.

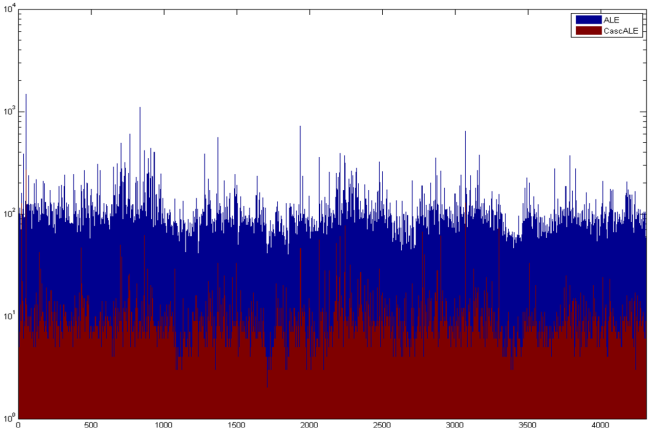
[2] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23:2001, 2001.

[3] Peter Carr and Richard Hartley. Solving multilabel graph cut problems with multilabel swap. In *DICTA*, pages 532–539, 2009.

[1] ALE is available from http://cms.brookes.ac.uk/staff/PhilipTorr/ale.htm



(a)



(b)

Figure 6: **Detailed Quantitative Results**: Here we see a close up look at the performance over all 107 classes (a), and the speedup over all test images in the dataset (b). These results are from CascALE with feature sharing trees at a shallow depth of 2 giving around 11 classes per sub-domain. We can see that the performance is stable across the whole dataset with respect the the flat model.

[4] Myung Jin Choi, Joseph J. Lim, Antonio Torralba, and Alan S. Willsky. Exploiting hierarchical context on a large database of object categories. In *CVPR*, 2010.

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

[6] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2011 (VOC2011) Results. http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html.

[7] Mark Everingham, Luc Gool, Christopher K. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vision*, 88:303–338, June 2010. ISSN

0920-5691. doi: http://dx.doi.org/10.1007/s11263-009-0275-4. URL http://dx.doi.org/10.1007/s11263-009-0275-4.

[8] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient Belief Propagation for Early Vision. In *CVPR*, volume 1, pages 261–268, 2004.

[9] P. F. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Conference on Computer Vision and Pattern Recognition*, 2008.

[10] S. Gould, F. Amat, and D. Koller. Alphabet soup: A framework for approximate energy minimization. In *Conference on Computer Vision and Pattern Recognition*, pages 903–910, 2009.

[11] D. Greig, B. Porteous, and A. Seheult. Exact maximum a posteriori estimation for binary images. *RoyalStat*, B: 51(2):271–279, 1989.

[12] Hedi Harzallah, Cordelia Schmid, Frédéric Jurie, and Adrien Gaidon. Classification aided two stage localization. In *PASCAL Visual Object Classes Challenge Workshop, in conjunction with ECCV*, oct 2008. URL http://lear.inrialpes.fr/pubs/2008/HSJG08.

[13] Taesup Kim, Sebastian Nowozin, Pushmeet Kohli, and Chang D. Yoo. Variable grouping for energy minimization. In *CVPR*, pages 1913–1920, 2011.

[14] P. Kohli, L. Ladicky, and P. H. S. Torr. Robust higher order potentials for enforcing label consistency. In *CVPR*, 2008.

[15] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *PAMI*, 28(10):1568–1583, 2006.

[16] Lubor Ladicky, Chris Russell, Pushmeet Kohli, and P. H. S. Torr. Associative hierarchical crfs for object class image segmentation. In *ICCV*, 2009.

[17] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: interactive foreground extraction using iterated graph cuts. In *SIGGRAPH*, pages 309–314, 2004.

[18] Bryan C. Russell, Bryan C. Russell, Antonio Torralba, Antonio Torralba, Kevin P. Murphy, Kevin P. Murphy, William T. Freeman, and William T. Freeman. Labelme: A database and web-based tool for image annotation. *MIT AI Lab Memo*, 2005.

[19] Chris Russell, Lubor Ladicky, Pushmeet Kohli, and P. H. S. Torr. Exact and approximate inference in associative hierarchical networks using graph cuts. *UAI*, 2010.

[20] 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 *ECCV*, pages 1–15, 2006.

[21] A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *CVPR*, volume 2, pages 762–769, Washington, DC, June 2004.