

Image Segmentation by Using Linear Spectral Clustering

N. Sandeep Reddy¹, Mr. Shaik Umar Faruq², K. Rajya Lakshmi³

Digital Systems and Computer Electronics, QUBA College of Engineering and Technology, Nellore,
Andhra Pradesh, India¹

Head of Electronics and Communication Engineering Department, QUBA College of Engineering and Technology,
Nellore, Andhra Pradesh, India²

Assistant Professor, QUBA College of Engineering and Technology, Nellore, Andhra Pradesh, India³

Abstract: In this paper a super pixel segmentation algorithm called Linear Spectral Clustering (LSC), which produces compact and uniform super pixels with low computational costs. Basically, a normalized cuts formulation of the super pixel segmentation is adopted based on a similarity metric that measures the color similarity and space proximity between image pixels. However, instead of using the traditional eigen-based algorithm, we approximate the similarity metric using a kernel function leading to an explicitly mapping of pixel values and coordinates into a high dimensional feature space. We revisit the conclusion that by appropriately weighting each point in this feature space, the objective functions of weighted K-means and normalized cuts share the same optimum point. As such, it is possible to optimize the cost function of normalized cuts by iteratively applying simple K-means clustering in the proposed feature space. LSC is of linear computational complexity and high memory efficiency and is able to preserve global properties of images. Experimental results show that LSC performs equally well or better than state of the art super pixel segmentation algorithms in terms of several commonly used evaluation metrics in image segmentation.

Keywords: Linear Spectral Clustering (LSC), super pixel segmentation, eigen-based algorithm.

1. INTRODUCTION

“Vision is the process of discovering from images what is present in the world, and where it is.” - **David Marr**

Image segmentation is an important image processing, and it seems everywhere if we want to analyze what inside the image. For example, if we seek to find if there is a chair or person inside an indoor image, we may need image segmentation to separate objects and analyze each object individually to check what it is. Image segmentation usually serves as the pre-processing before image pattern recognition, image feature extraction and image compression. Researches of it started around 1970, while there is still no robust solution, so we want to find the reason and see what I can do to improve it.

Image segmentation is used to separate an image into several “meaningful” parts. It is an old research topic, which started around 1970, but there is still no robust solution toward it. There are two main reasons, the first is that the content variety of images is too large, and the second one is that there is no benchmark standard to judge the performance. For example, in figure 1.1, we show an original image and two segmented images based on different kinds of image segmentation methods. The one of figure 1.1 (b) separates the sky into several parts while the figure 1.1 (c) misses some detail in the original image. Every technique has its own advantages also disadvantages, so it’s hard to tell which one is better.

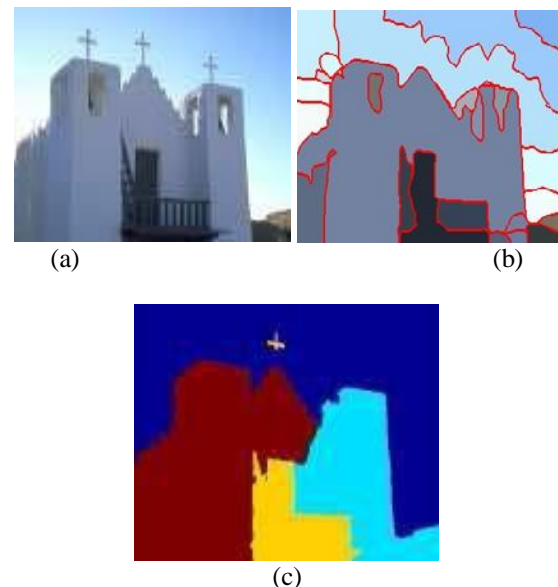


Figure 1.1: (a) is the original image; (b) and (c) are the segmentation results

There are tons of previous works about image segmentation. From these surveys, we could simply separate the image segmentation techniques into three

different classes (1) Feature-space based method, (2) image-domain based method, and (3) edge-based method. The feature-space based method is composed of two steps, feature extraction and clustering. Feature extraction is the process to find some characteristics of each pixel or of the region around each pixel, after we get some symbolic properties around each pixel, clustering process is executed to separate the image into several “meaningful” parts based on these properties.

Image-domain based method goes through the image and finds the boundary between segments by some rules. The main consideration to separate two pixels into different segments is the pixel value difference, so this kind of methods couldn't deal with textures very well. Split and merge, region growing, and watershed are the most popular methods in this class. The third class is edge-based image segmentation method, which consists of edge detection and edge linking.

Although there have been many kinds of existed methods, some common problem still can't be solved. For class (1), the accurate boundaries between segments are still hard to determine because features take properties around but not exactly on each pixel. Class (2) only uses the pixel value information, which may result in over-segmentation on texture regions. Finally the edge detection process makes class (3) always suffer the over-segmentation problem. In our project, we adopt the “normalized cut framework” for image segmentation, which finds the best cutting path from the global view (the whole image view) rather than by local thresholds and is expected to have better segmentation results than other methods.

The term super pixel was introduced by Ren and Malik in 2003 [RM03] and is used to describe a group of pixels similar in color or other low-level features. The concept of super pixels is motivated by two important aspects, which have been adapted widely. Firstly, pixels do not represent natural entities but are merely a result of discretization [RM03]. And secondly, the number of pixels prevents many algorithms from being feasible [RM03]. At this point, Ren and Malik introduce super pixels as more natural entities – grouping pixels which perceptually belong together.

The task of segmenting an image into super pixels is widely referred to as over segmentation or super pixel segmentation. While recent algorithms are explicitly designed to generate super pixel segmentations, others were initially intended for classical image segmentation or clustering. Overall, super pixels may have different properties which first of all impose a visual difference.

II. EXISTING SYSTEM

In early studies, algorithms designed for image segmentation were directly used for generating super pixels, such as FH, mean shift and quick shift. In FH, each

super pixel is represented by a minimum spanning tree and two super pixels are merged if the maximum weight of edges inside the trees is larger than the minimum weight of edges connecting them. Mean shift and quick shift are two mode seeking methods attempting to maximize a density function by shifting pixels towards areas of higher density. Pixels converging to the same mode formulate a super pixel. These algorithms offer no explicit control over the size and number of the super pixel and compactness is not considered. Super pixels thus produced are usually of irregular sizes and shapes and tend to overlap with multiple objects.

Previous researches show that algorithms which do not consider the spatial compactness usually lead to under segmentation, especially when there is poor contrast or shadow. Among the four algorithms mentioned above, Ncuts is the only one that implicitly takes compactness into consideration. However, the high computational complexity has limited its applicability. To solve this problem, several other approaches have been proposed to generate compact and regular super pixels with relatively low computational complexity.

III. PROPOSED SYSTEM

Linear Spectral Clustering (LSC), which not only captures perceptually important global image properties, but also runs in linear complexity with high memory efficiency. In LSC, we map each image pixel to a point in a ten dimensional feature space in which weighted K-means is applied for segmentation. Non-local information is implicitly preserved due to the equivalence between the weighted K-means clustering in this ten dimensional feature space and normalized cuts in the original pixel space. LSC is of linear computational complexity and high memory efficiency and is able to preserve global properties of images. Experimental results show that LSC performs equally well or better than state of the art super pixel segmentation algorithms in terms of several commonly used evaluation metrics in image segmentation.

Super pixels have actively been used for a wide range of applications because of:

- Super pixels should respect object boundaries
- Super pixels should be generated as efficiently as possible
- Super pixels should not lower the achievable performance² of subsequent processing steps

2. Super Pixel Segmentation Algorithms

Many existing algorithms in computer vision use the pixel-grid as the underlying representation. For example, stochastic models of images, such as Markov random fields, are often defined on this regular grid. Or, face detection is typically done by matching stored templates to every fixed-size (say, 50x50) window in the image.

The pixel-grid, however, is not a natural representation of visual scenes. It is rather an "artifact" of a digital imaging process. It would be more natural, and presumably more efficient, to work with perceptually meaningful entities obtained from a low-level grouping process. For example, we can apply the Normalized Cuts algorithm to partition an image into, say, 500 segments (what we call super pixels). Such a super pixel map has many desired properties:

- It is computationally efficient: it reduces the complexity of images from hundreds of thousands of pixels to only a few hundred super pixels.
- It is also representationally efficient: pairwise constraints between units, while only for adjacent pixels on the pixel-grid, can now model much longer-range interactions between super pixels.
- The super pixels are perceptually meaningful: each super pixel is a perceptually consistent unit, i.e. all pixels in a super pixel are most likely uniform in, say, color and texture.
- It is near-complete: because super pixels are results of an over segmentation, most structures in the image are conserved. There is very little loss in moving from the pixel-grid to the super pixel map.
- It is actually not novel to use super pixels or atomic regions to speed up later-stage visual processing; the idea has been around the community for a while. What we have done is: (1) to empirically validate the completeness of super pixel maps; and (2) to apply it to solve challenging vision problems such as finding people in static images.

3. FUNCTIONAL DESCRIPTION OF THE PROJECT

In LSC, we map each image pixel to a point in a ten dimensional feature space in which weighted K-means is applied for segmentation. Non-local information is implicitly preserved due to the equivalence between the weighted K-means clustering in this ten dimensional feature space and normalized cuts in the original pixel space. Simple weighted K-means clustering in the feature space can be used to optimize the segmentation cost function defined by normalized cuts. Figure 3.1 shows some super pixel segmentation results of LSC.

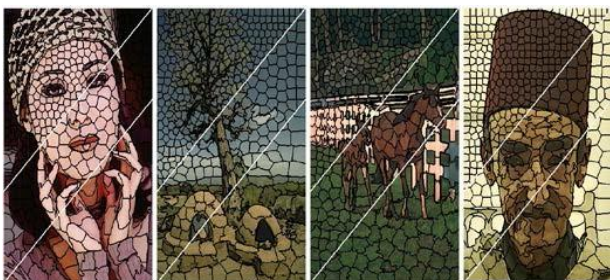
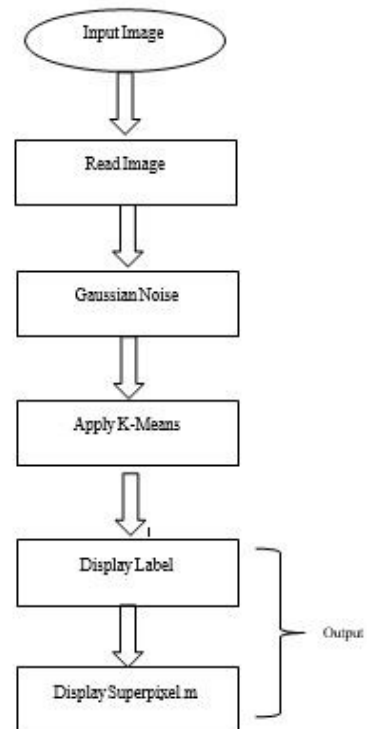


Figure 3.1 Images segmented into 1000/500/200 super pixels using the proposed LSC algorithm

The LSC super pixel segmentation algorithm which not only produces super pixels with state of the art boundary adherence but also captures global image properties. The LSC algorithm is proposed based on the investigation of the relationship between the objective functions of normalized cuts and weighted K-means. We find that optimizing these two objective functions are equivalent if the similarity between two points in the input space is equal to the weighted inner product between the two corresponding vectors in an elaborately designed high dimensional feature space. As such, simple weighted K-means clustering in this feature space can be used to replace the highly complex eigen-based method for minimizing the normalized cuts objective function. Comparing to the weighted kernel K-means clustering [7], LSC avoids the calculation of the large kernel matrix and the convergence condition can be naturally satisfied. By further limiting the search space of the weighted K-means, LSC achieves a linear complexity while retaining the high quality of the generated super pixels.

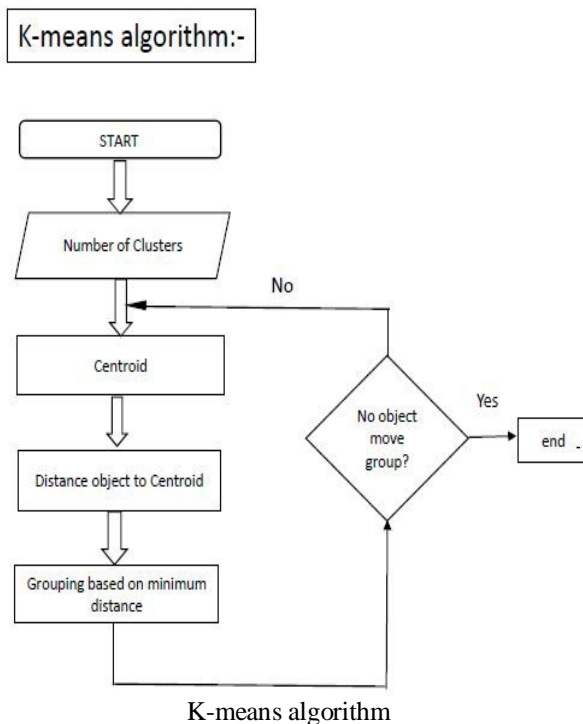
Until now, we have explicitly define a ten dimensional feature space such that weighted K-means clustering in this feature space is approximately equivalent to normalized cuts in the input space. Noticing that under the similarity function defined in both the kernel matrix for weighted kernel K-means and the affinity matrix in the normalized cuts will be highly dense, leading to high computational complexity when using existing methods. In contrast, by directly applying weighted K-means in the ten dimensional feature space, the objective function of the normalized cuts can be efficiently optimized.

FLOW CHART



LSC Super pixel Segmentation Algorithm

- 1: Map each point $p = (l_p, \alpha_p, \beta_p, x_p, y_p)$ to a ten dimensional vector $\phi(p)$ in the feature space.
- 2: Sampling K seeds over the image uniformly at fixed horizontal and vertical intervals v_x and v_y .
- 3: Move each seed to its lowest gradient neighbour in the 3×3 neighborhood.
- 4: Initialize weighted mean m_k and search center c_k of each cluster using the corresponding seed.
- 5: Set label $L(p) = 0$ for each point p .
- 6: Set distance $D(p) = \infty$ for each point p .
- 7: **repeat**
- 8: **for** each weighted means m_k and search center c_k
- 9: **do**
- 10: **for** point p in the $\tau_{vx} \times \tau_{vy}$ neighbourhood of c_k in the image plane **do**
- 11: $D =$ Euclidean distance between $\phi(p)$ and m_k in the feature space.
- 12: **if** $D < d(p)$ **then**
- 13: $d(p) = D$
- 14: $L(p) = k$
- 15: **end if**
- 16: **end for**
- 17: Update weighted means and search centers for all clusters.
- 18: **until** weighted means of K cluster converge.
- 19: Merge small super pixels to their neighbours.



In this section, we will present the LSC super pixel segmentation algorithm which not only produces super pixels with state of the art boundary adherence but also captures global image properties. The LSC algorithm is

proposed based on the investigation of the relationship between the objective functions of normalized cuts and weighted K-means. We find that optimizing these two objective functions are equivalent if the similarity between two points in the input space is equal to the weighted inner product between the two corresponding vectors in an elaborately designed high dimensional feature space. As such, simple weighted K-means clustering in this feature space can be used to replace the highly complex eigen-based method for minimizing the normalized cuts objective function. Comparing to the weighted kernel K-means clustering, LSC avoids the calculation of the large kernel matrix and the convergence condition can be naturally satisfied. By further limiting the search space of the weighted K-means, LCS achieves a linear complexity while retaining the high quality of the generated super pixels.

4. EXPERIMENTS

I compare LSC to eight state of the art super pixel segmentation algorithms including SLIC [1], SEEDS [2], Ncuts [23], Lattice [15], ERS [12], Turbopixel [11], EneOpt1 and EneOpt0 [20]. For all the eight algorithms, the implementations are based on publicly available code. Experiments are performed on the Berkeley Segmentation Database [13] consisting of three hundred test images with human segmented ground truth. The boundary adherence of super pixels generated by different algorithms are compared using three commonly used evaluation metrics in image segmentation: under-segmentation error (UE), boundary recall (BR) and achievable segmentation accuracy (ASA).

Among the three metrics, UE measures the percentage of pixels that leak from the ground truth boundaries. It actually evaluates the quality of super pixel segmentation by penalizing super pixels overlapping with multiple objects. The definition of UE used in [1] is adopted here. Lower UE indicates that fewer super pixels straddle multiple objects. BR measures the fraction of ground truth boundaries correctly recovered by the super pixel boundaries.

A true boundary pixel is regarded to be correctly recovered if it falls within 2 pixels from at least one super pixel boundary point. A high BR indicates that very few true boundaries are missed. ASA is defined as the highest achievable object segmentation accuracy when utilizing super pixel as units [12].

By labelling each super pixel with the ground truth segments of the largest overlapping area, ASA is calculated as the fraction of labelled pixels that are not leaked from the ground truth boundaries. A high ASA indicates that the super pixels comply well with objects in the image. Figure 4 shows the experimental results which are average values over all the 300 test images in the Berkeley segmentation database.

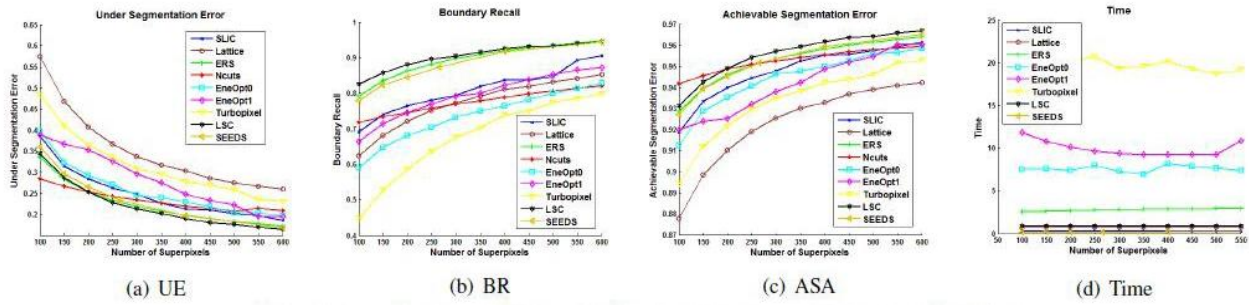


Figure 4. Quantitative evaluation of different superpixel segmentation algorithms.

Computational efficiency is also an important factor for evaluating the performance of super pixel segmentation algorithms. In our experiment, we calculate the average running time for different algorithms and the results are shown in Figure 4(d). All the experiments are performed on a desktop PC equipped with an Intel 3.4 GHz dual core processor and 2GB memory. The time consumption of the Ncuts algorithm [23] is much higher than that of the other methods and is therefore omitted in Figure 4(d).

It can be observed that in terms of the boundary adherence, the proposed LSC is comparable to the state of art the algorithms. For relative large number of super pixels, LSC performs the best. Also, LSC is of linear complexity and is among the algorithms with the highest time efficiency. In addition, qualitative experiments demonstrate that LSC performs the best. We select the five algorithms (SEEDS, Ncuts, SLIC, ERS and LSC) that achieve the lowest UE values when $K = 400$ for visual comparison. According to Figure 4, these five algorithms generally outperform the remaining three algorithms in terms of UE, BR as well as ASA. Some detail segmentation results are emphasized to facilitate close visual inspection. Intuitively, LSC has achieved the most perceptually satisfactory segmentation results for different kind of images.

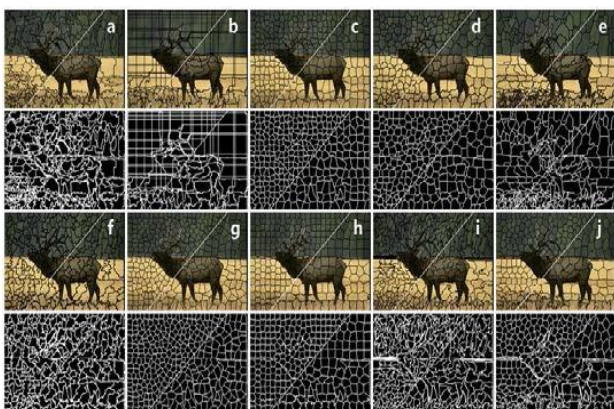


Figure 6.1 Comparison of different super pixel segmentation algorithms. (a)SEEDS, (b) Lattice, (c) Turbopixel, (d) EneOpt0, (e) EneOpt1, (f) quick shift, (g) Ncuts, (h) SLIC, (i) ERS and (j) LSC. The image is segmented into 400/200 super pixels.

5. RESULTS

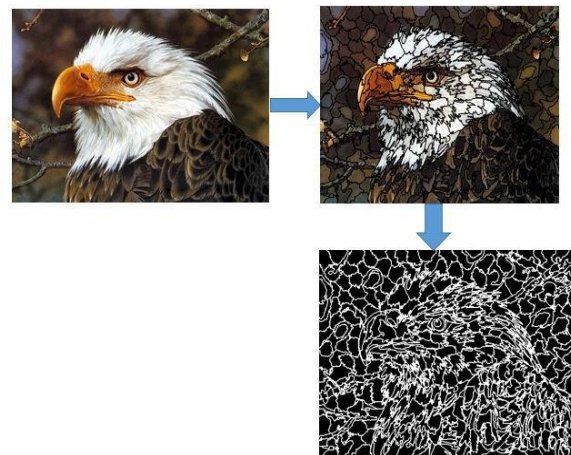


Figure 5 Segmentation result by using linear spectral clustering

6. CONCLUSION

In this project a novel super pixel segmentation algorithm, LSC, this produces compact and regular shaped super pixels with linear time complexity and high memory efficiency. The most critical idea in LSC is to explicitly utilize the connection between the optimization objectives of weighted K-means and normalized cuts by introducing a elaborately designed high dimensional space. As such, LSC achieves both boundary adherence and global image structure perseverance through simple local feature based operations. Experimental results show that LSC generally over-performs most state of the art algorithms both quantitatively and qualitatively.

REFERENCES

- [1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic super pixels compared to state-of- the-art super pixel methods. *IEEE Trans. on PAMI*, 34(11):2274– 2281, 2013.
- [2] M. Bergh, X. Boix, G. Roig, B. Capitan, and L. V. Gool. Seeds: Super pixels extracted via energy- driven sampling. *Proc. of ECCV*, 7578:13–26, 2012.
- [3] Y. Boykov and V. Kolmogrov. An experimental comparison of min- cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. on PAMI*, 26(9):1124–1137, 2001.
- [4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on PAMI*, 23(11):1222– 1239, 2001.



- [5] D. Comaniciu and P. Meer. Mean shift: a robust approach towards feature space analysis. *IEEE Trans. on PAMI*, 24(5):603–619, 2002.
- [6] N. Cristianini and J. Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press New York, NY, USA, 2000.
- [7] I. Dhillon, Y. Guan, and B. Kulis. Weighted graph cuts without eigenvectors: a multilevel approach. *IEEE Trans. On PAMI*, 29(11):1944–1957, 2007.
- [8] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.
- [9] D. Hoiem, A. Efros, and M. Hebert. Automatic photo pop-up. *ACM Trans. on Graphics*, 24(3):577–584, 2005.
- [10] V. Kolmogorov. What energy functions can be minimized via graph cuts? *IEEE Trans. on PAMI*, 26(2):147–159, 2004.
- [11] A. Levinstein, A. Stere, K. Kutulakos, D. Fleet, S. Dickinson, and K. Siddiqi. Turbopixel: fast superpixels using geometric flow. *IEEE Trans. on PAMI*, 31(12):2209–2297, 2009.
- [12] M. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa. Entropy rate superpixel segmentation. *Proc. of CVPR*, pages 2097–2104, 2011.
- [13] 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. *Proc. of ICCV*, 2:416–423, 2001.
- [14] A. Moore, S. Prince, and J. Warrell. Lattice cut – constructing super pixels using layer constraints. *Proc. of CVPR*, pages 2117–2124, 2010.
- [15] A. Moore, S. Prince, J. Warrell, U. Mohammed and G. Jones Super pixel lattices. *Proc. of CVPR*, pages 1–8, 2008.
- [16] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: analysis and an algorithm. *Proc. of NIPS*, pages 849–856, 2001.
- [17] X. Ren and J. Malik. Learning a classification model for segmentation. *Proc. of ICCV*, 1:10–17, 2003.
- [18] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. on PAMI*, 22(8):888–905, 2000.
- [19] J. Tighe and S. Lazebnik. Superparsing: scalable non parametric image parsing with super pixel. *Proc. of ECCV*, 5:352–365, 2010.
- [20] O. Veksler, Y. Boykov, and P. Mehrani. Super pixels and supervoxels in an energy optimization framework. *Proc. Of ECCV*, pages 211–224, 2010.
- [21] A. Veldadi and S. Soatto. Quick shift and kernel methods for mode seeking. *Proc. of ECCV*, pages 705–718, 2008.
- [22] S. Wang, H. Lu, F. Yang, and M. Yang. Super pixel tracking. *Proc. of ICCV*, 1:1323–1330, 2011.
- [23] S. Yu and J. Shi. Multiclass spectral clustering. *Proc. of ICCV*, 1:313–319, 2003.