A Comparison of Image Segmentation Algorithms

Caroline Pantofaru Martial Hebert

CMU-RI-TR-05-40

September 1, 2005

The Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Abstract

Unsupervised image segmentation algorithms have matured to the point where they generate reasonable segmentations, and thus can begin to be incorporated into larger systems. A system designer now has an array of available algorithm choices, however, few objective numerical evaluations exist of these segmentation algorithms. As a first step towards filling this gap, this paper presents an evaluation of two popular segmentation algorithms, the mean shift-based segmentation algorithm and a graph-based segmentation scheme. We also consider a hybrid method which combines the other two methods. This quantitative evaluation is made possible by the recently proposed measure of segmentation correctness, the Normalized Probabilistic Rand (NPR) index, which allows a principled comparison between segmentations created by different algorithms, as well as segmentations on different images.

For each algorithm, we consider its correctness as measured by the NPR index, as well as its stability with respect to changes in parameter settings and with respect to different images. An algorithm which produces correct segmentation results with a wide array of parameters on any one image, as well as correct segmentation results on multiple images with the same parameters, will be a useful, predictable and easily adjustable preprocessing step in a larger system.

Our results are presented on the Berkeley image segmentation database, which contains 300 natural images along with several ground truth hand segmentations for each image. As opposed to previous results presented on this database, the algorithms we compare all use the same image features (position and colour) for segmentation, thereby making their outputs directly comparable.

Contents

1 Introduction

Unsupervised image segmentation algorithms have matured to the point that they provide segmentations which agree to a large extent with human intuition. The time has arrived for these segmentations to play a larger role in object recognition. It is clear that unsupervised segmentation can be used to help cue and refine various recognition algorithms. However, one of the stumbling blocks that remains is that it is unknown exactly how well these segmentation algorithms perform from an objective standpoint. Most presentations of segmentation algorithms contain superficial evaluations which merely display images of the segmentation results and appeal to the reader's intuition for evaluation. There is a consistent lack of numerical results, thus it is difficult to know which segmentation algorithms present useful results and in which situations they do so. Appealing to human intuition is convenient, however if the algorithm is going to be used in an automated system then objective results on large datasets are to be desired.

In this paper we present the results of an objective evaluation of two popular segmentation techniques: mean shift segmentation [1], and the efficient graph-based segmentation algorithm presented in [4]. As well, we look at a hybrid variant that combines these algorithms. For each of these algorithms, we examine three characteristics:

1. *Correctness*: the ability to produce segmentations which agree with human intuition. That is, segmentations which correctly identify structures in the image at neither too fine nor too coarse a level of detail.

2. *Stability with respect to parameter choice*: the ability to produce segmentations of consistent correctness for a range of parameter choices.

3. *Stability with respect to image choice*: the ability to produce segmentations of consistent correctness using the same parameter choice on a wide range of different images.

If a segmentation scheme satisfies these three characteristics, then it will give useful and predictable results which can be reliably incorporated into a larger system.

The measure we use to evaluate these algorithms is the recently proposed Normalized Probabilistic Rand (NPR) index [6]. We chose to use this measure as it allows a principled comparison between segmentation results on different images, with differing numbers of regions, and generated by different algorithms with different parameters. Also, the NPR index of one segmentation is meaningful as an absolute score, not just in comparison with that of another segmentation. These characteristics are all necessary for the comparison we wish to perform.

Our dataset for this evaluation is the Berkeley Segmentation Database [5], which contains 300 natural images with multiple ground truth hand segmentations of each image. To ensure a valid comparison between algorithms, we compute the same features (pixel location and colour) for every image and every segmentation algorithm.

This paper is organized as follows. We begin by presenting some previous work on comparing segmentations and clusterings. Then, we present each of the segmentation algorithms and the hybrid variant we considered. Next, we describe the NPR index and present the reasons for using this measure, followed by a description of our comparison methodology. Finally, we present our results.

2 Previous Work

There have been previous attempts at numerical image segmentation method comparisons, although the number is small. Here we describe some examples and summarize how our work differs.

A comparison of spectral clustering methods is given in [8]. The authors attempted to compare variants of four popular spectral clustering algorithms: normalized cuts by Shi and Malik [13], a variant by Kannan, Vempala and Vetta [9], the algorithm by Ng, Jordan and Weiss [11], and the Multicut algorithm by Meila and Shi [10], as well as Single and Ward linkage as a base for comparison. They also combined different parts of different algorithms to create new ones. The measure of correctness used was the Variation of Information introduced in [12], which considers the conditional entropies between the labels in two segmentations. The results of this comparison were largely unexciting, with all of the algorithms and variants performing well on ‘easy’ data, and all performing roughly equally badly on ‘hard’ data.

Another attempt at segmentation algorithm comparison is presented on the Berkeley database and segmentation comparison website [5]. Here a large set of images are made available for segmentation evaluation, and a framework is set up to facilitate comparison. Comparisons currently exist between using cues of brightness, texture, and/or edges for segmentation. However, there are no current examples of comparisons between actual algorithms which use the same features. The measure used for segmentation correctness is a precision-recall curve based on the correctness of each region boundary pixel. Different boundary thresholds are used to obtain different points on the curve. The reported statistic is the F-measure, the harmonic mean of the precision and recall. This measure has the downside of considering only region boundaries instead of the regions themselves. Since region difference is a quadratic measure whereas boundary difference is a linear measure, small boundary imperfections will affect the measure more than they necessarily should.

Our current work is the first which presents a segmentation algorithm comparison within a framework which is independent of the features used, which compares each of the three traits of correctness and stability with respect to parameters and images, and which uses a measure of correctness which can compare all of the necessary segmentations in a principled and valid manner. In addition, we consider two popular but previously unevaluated algorithms.

3 The Segmentation Algorithms

As mentioned, we will compare three different segmentation techniques, the mean shift-based segmentation algorithm [1], an efficient graph-based segmentation algorithm [4], and a hybrid of the two. We have chosen to look at mean shift-based segmentation as it is generally effective and has become widely-used in the vision community. The efficient graph-based segmentation algorithm was chosen as an interesting comparison to the mean shift in that its general approach is similar, however it excludes the mean shift filtering step itself, thus partially addressing the question of whether the filtering step is useful. The combination of the two algorithms is shown as an attempt to improve the performance and stability of either one alone. In this section we describe each algorithm and further discuss how they differ from one another.

3.1 Mean Shift Segmentation

The mean shift based segmentation technique was introduced in [1] and has become widely-used in the vision community. It is one of many techniques under the heading of “feature space analysis”. The mean shift technique is comprised of two basic steps: a mean shift filtering of the original image data (in feature space), and a subsequent clustering of the filtered data points. Below we will briefly describe each of these steps and then discuss some of the strengths and weaknesses of this method.

3.1.1 Filtering

The filtering step of the mean shift segmentation algorithm consists of analyzing the probability density function underlying the image data in feature space. Consider the feature space consisting of the original image data represented as the (x, y) location of each pixel, plus its colour in L*u*v* space (L^*, u^*, v^*). The modes of the pdf underlying the data in this space will correspond to the locations with highest data density. In terms of a segmentation, it is intuitive that the data points close to these high density points (modes) should be clustered together. Note that these modes are also far less sensitive to outliers than the means of, say, a mixture of Gaussians would be.

The mean shift filtering step consists of finding the modes of the underlying pdf and associating with them any points in their basin of attraction. Unlike earlier techniques, the mean shift is a non-parametric technique and hence we will need to estimate the gradient of the pdf, $f(x)$, in an iterative manner using kernel density estimation to find the modes. For a data point $\mathbf{x}$ in feature space, the density gradient is estimated as being proportional to the mean shift vector:

$$\widehat{\nabla f}(\mathbf{x}) \propto \frac{\sum_{i=1}^{n} \mathbf{x}_i g\left(\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\|\right)}{\sum_{i=1}^{n} g\left(\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\|\right)} - \mathbf{x} \tag{1}$$

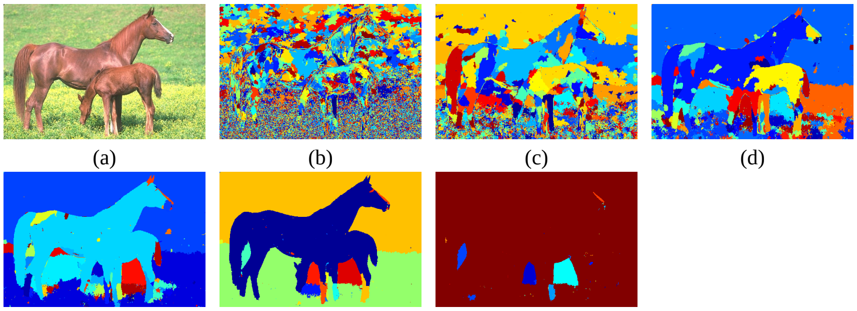
where $\mathbf{x}_i$ are the data points, $\mathbf{x}$ is a point in the feature space, n is the number of datapoints (pixels in the image), and g is the profile of the symmetric kernel G. We use the simple case where G is the uniform kernel with radius vector h. Thus the above equation simplifies to:

$$\widehat{\nabla f}(\mathbf{x}) \propto \left[\frac{1}{|S_{\mathbf{x},h_s,h_r}|} \sum_{\mathbf{x}_i \in S_{\mathbf{x},h_s,h_r}} \mathbf{x}_i\right] - \mathbf{x} \tag{2}$$

where $S_{\mathbf{x},h_s,h_r}$ represents the sphere in feature space centered at $\mathbf{x}$ and having spatial radius h_s and colour (range) radius h_r, and the $\mathbf{x}_i$ represent the data points within that sphere.

For every data point (pixel in the original image) $\mathbf{x}$ we can iteratively compute the gradient estimate in Eqn. 2 and move $\mathbf{x}$ in that direction, until the gradient is below a threshold. Thus we have found the points where $\widehat{\nabla f}(\mathbf{x}') = 0$, the modes of the density estimate. We can then replace the point $\mathbf{x}$ with $\mathbf{x}'$, the mode with which it is associated.

Finding the mode associated with each data point helps to smooth the image while preserving discontinuities. Intuitively, if two points $\mathbf{x}_i$ and $\mathbf{x}_j$ are far from each other in feature space, then $\mathbf{x}_i \notin S_{\mathbf{x}_j,h_s,h_r}$ and hence $\mathbf{x}_j$ doesn’t contribute to the mean shift vector gradient estimate and the trajectory of $\mathbf{x}_i$ will move it away from $\mathbf{x}_j$. Hence,



(a) (b) (c) (d)

(e) (f) (g)

Figure 1: Example of changing scores for different segmentation granularities: (a) Original image, (b)-(g) mean shift segmentations using scale bandwidth (h_s) 7 and color bandwidths (h_r) 3, 7, 11, 15, 19 and 23 respectively.

pixels on either side of a strong discontinuity will not attract each other. However, filtering alone does not provide a segmentation as the modes found are noisy. This "noise" stems from two sources. First, the mode estimation is an iterative process, hence it only converges to within the threshold provided (and with some numerical error). Second, consider an area in feature space larger than $S_{\mathbf{x}_j,h_s,h_r}$ and where the colour features are uniform or have a gradient of 1. Since the pixel coordinates are uniform by design, the mean shift vector will be 0 in this region, and the data points will not move and hence not converge to a single mode. Intuitively, however, we would like all of these datapoints to belong to the same cluster in the final segmentation. For these reasons, mean shift filtering is only a preprocessing step, and a second step is required in the segmentation process: clustering of the filtered data points $\{\mathbf{x}'\}$.

3.1.2 Clustering

After mean shift filtering, each data point in the feature space has been replaced by its corresponding mode. As described above, some points may have collapsed to the same mode, but many have not despite the fact that they may be less than one kernel radius apart. In the original mean shift segmentation paper [1], clustering is described as a simple post-processing step in which any modes that are less than one kernel radius apart are grouped together and their basins of attraction are merged. This suggests using single linkage clustering, which effectively converts the filtered points into a segmentation.

The only other paper using mean shift segmentation that speaks directly to the clustering is [2]. In this approach, a region adjacency graph (RAG) is created to hierarchically cluster the modes. Also, edge information from an edge detector is combined with the colour information to better guide the clustering. This is the method used in the publicly available EDISON system, also described in [2]. The EDISON system is the implementation we use here as our mean shift segmentation system.

3.1.3 Discussion

Mean shift filtering using either single linkage clustering or edge-directed clustering produces segmentations that correspond well to human perception. However, as we discuss in the experiments section, this algorithm is quite sensitive to its parameters. The mean shift filtering stage has two parameters corresponding to the bandwidths (radii of the kernel) for the spatial (h_s) and colour (h_r) features. Slight variations in h_r can cause large changes in the granularity of the segmentation, as shown in Fig. 1. By adjusting the colour bandwidth we can produce over-segmentations as in Fig. 1b which show every minute detail, to reasonably intuitive segmentations as in Fig. 1f which delineate objects or large patches, to under-segmentations as in Fig. 1g which obscure the important elements completely. This issue is a major stumbling block with respect to using mean shift segmentation as a reliable preprocessing step for other algorithms, such as object recognition. For an object recognition system to actually use a segmentation algorithm, it requires that the segmentations produced be fairly stable under parameter changes and that the same parameters produce stable results for different images, thus easing the burden of parameter tuning. In an attempt to improve stability, we consider a second algorithm.

3.2 Efficient Graph-based Segmentation

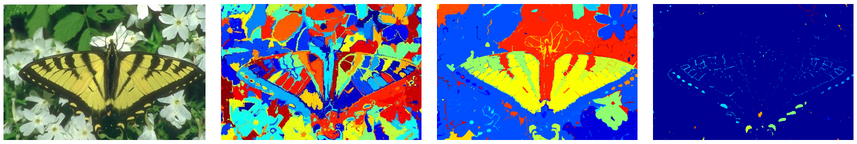
Efficient graph-based image segmentation, introduced in [4], is another method of performing clustering in feature space. This method works directly on the data points in feature space, without first performing a filtering step, and uses a variation on single linkage clustering. The key to the success of this method is adaptive thresholding. To perform traditional single linkage clustering, a minimum spanning tree of the data points is first generated (using Kruskal's algorithm), from which any edges with length greater than a given hard threshold are removed. The connected components become the clusters in the segmentation. The method in [4] eliminates the need for a hard threshold, instead replacing it with a data-dependent term.

More specifically, let $G = (V, E)$ be a (fully connected) graph, with m edges and n vertices. Each vertex is a pixel, $\mathbf{x}$, represented in the feature space. The final segmentation will be $S = (C_1, ..., C_r)$ where C_i is a cluster of data points. The algorithm is:

1. Sort $E = (e_1, ..., e_m)$ such that $|e_t| \leq |e_{t'}| \, \forall t < t'$
2. Let $S^0 = \big(\{\mathbf{x}_1\}, ..., \{\mathbf{x}_n\}\big)$, in other words each initial cluster contains exactly one vertex.
3. For $t = 1, ..., m$
    (a) Let $\mathbf{x}_i$ and $\mathbf{x}_j$ be the vertices connected by e_t.
    (b) Let $C^{t-1}_{\mathbf{x}_i}$ be the connected component containing point $\mathbf{x}_i$ on iteration $t-1$, and $l_i = \max_{\text{mst}} C^{t-1}_{\mathbf{x}_i}$ be the longest edge in the minimum spanning tree of $C^{t-1}_{\mathbf{x}_i}$. Likewise for l_j.
    (c) Merge $C^{t-1}_{\mathbf{x}_i}$ and $C^{t-1}_{\mathbf{x}_j}$ if

$$|e_t| < \min\{l_i + \frac{k}{\left|C^{t-1}_{\mathbf{x}_i}\right|}, l_j + \frac{k}{\left|C^{t-1}_{\mathbf{x}_j}\right|}\} \tag{3}$$

    where k is a constant.
4. $S = S^m$



(a) (b) (c) (d)

Figure 2: Example of changing scores for different parameters using efficient graph-based segmentation: (a) Original image, (b)-(d) efficient graph-based segmentations using scale bandwidth (h_s) 7, color bandwidth (h_r) 7 and k values 5, 25, 125 respectively.

We can make the algorithm more efficient by considering only the 100 shortest edges from any vertex instead of the fully connected graph. This does not result in any perceptible quality loss.

In contrast to single linkage clustering which uses a constant K to set the threshold on edge length for merging two components in Eqn. 3, efficient graph-based segmentation uses a variable threshold. This threshold effectively allows two components to be merged if the minimum edge connecting them does not have length greater than the maximum edge in either of the components' minimum spanning trees, plus a term $\tau = \frac{k}{\left|C^{t-1}_{\mathbf{x}_i}\right|}$. As defined here, τ is dependent on a constant k and the size of the component. Note that on the first iteration, $l_i = 0$ and $l_j = 0$, and $\left|C^0_{\mathbf{x}_i}\right| = 1$ and $\left|C^0_{\mathbf{x}_j}\right| = 1$, so k represents the longest edge which will be added to any cluster at any time, $k = l_{max}$. Also, as the number of points in a component increases, the tolerance on added edge length for new edges becomes tighter and fewer mergers are performed, thus indirectly controlling region size. However, it is possible to use any non-negative function for τ which reflects the goals of the segmentation system.

The merging criteria in Eqn. 3 allows efficient graph-based clustering to be sensitive to edges in areas of low variability, and less sensitive to them in areas of high variability. This is intuitively the property we would like to see in a clustering algorithm. However, the results it gives do not have the same degree of correctness as mean shift-based segmentation, as demonstrated in Fig. 2. This algorithm also suffers somewhat from sensitivity to its parameter, k.

3.3 Hybrid Segmentation Algorithm

An obvious question emerges when describing the mean shift based segmentation method [1] and the efficient graph based clustering method [4]: can we combine the two methods to give better results than either method alone? More specifically, can we combine the two methods to create more stable segmentations that are less sensitive to parameter changes and for which the same parameters give reasonable segmentations across multiple images? In an attempt to answer these questions, the third algorithm we consider is a combination of the previous two algorithms: first we apply mean shift filtering, and then we use efficient graph-based clustering to give the final segmentation. The result of applying this algorithm with different parameters can be seen in Fig. 3. Notice that for $h_r = 15$ the quality of the segmentation is high. Also notice that the rate of granularity change is slower than either of the previous two algorithms, even

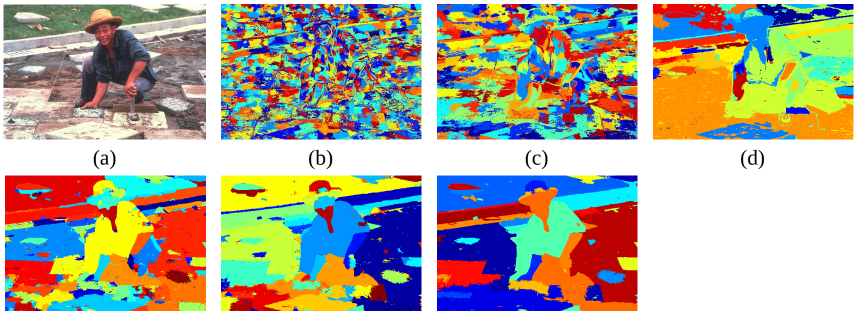

(a) (b) (c) (d)
(e) (f) (g)

Figure 3: Example of changing scores for different parameters using a hybrid segmentation algorithm which first performs mean shift filtering and then efficient graph-based segmentation: (a) Original image, (b)-(g) segmentations using scale bandwidth (h_s) 7, and color bandwidth (h_r) and k value combinations (3,5), (3,25), (3,125), (15,5), (15,25), (15,125) respectively.

though the parameters cover a wide range.

4 Evaluation Methodology

Having described each of the segmentation algorithms to be evaluated, we now present the evaluation measure and methodology to be used.

4.1 Normalized Probabilistic Rand (NPR) Index

The performance measure we use is the Normalized Probabilistic Rand (NPR) index [6], an extension of the Probabilistic Rand (PR) index introduced in [7]. In our tests, we would like to compare a test segmentation of an image S_{test} with a set of ground truth segmentations (human segmentations in this case), $S_1, ..., S_K$. We will consider a segmentation "good" if it correctly identifies the pairwise relationships between the pixels as defined in the ground truth segmentations. In other words, for any pair of pixels x_i, x_j we would like the labels of those pixels $l_i^{S_{test}}, l_j^{S_{test}}$ to be the same in the test segmentation if the labels $l_i^{S_k}, l_j^{S_k}$ were the same in the ground truth segmentations, and vice versa. We would also like to penalize inconsistencies between the test and ground truth label pair relationships proportionally to the level of consistency between the ground truth label pair relationships. These requirements lead us to the PR index:

$$\begin{aligned}\mathrm{PR}(S_{\text{test}}, \{S_k\}) = \frac{1}{\binom{N}{2}} \sum_{\substack{i,j \\ i \prec j}} \Big[& \mathbb{I}\left(l_i^{S_{\text{test}}} = l_j^{S_{\text{test}}}\right) p_{ij} \\ & + \mathbb{I}\left(l_i^{S_{\text{test}}} \neq l_j^{S_{\text{test}}}\right)(1 - p_{ij}) \Big]\end{aligned} \tag{4}$$

Let $c_{ij} = \mathbb{I}\left(l_i^{S_{\text{test}}} = l_j^{S_{\text{test}}}\right)$

Then the PR index can be written as:

$$\mathrm{PR}(S_{\text{test}}, \{S_k\}) = \frac{1}{\binom{N}{2}} \sum_{\substack{i,j \\ i \prec j}} [c_{ij} p_{ij} + (1 - c_{ij})(1 - p_{ij})] \tag{5}$$

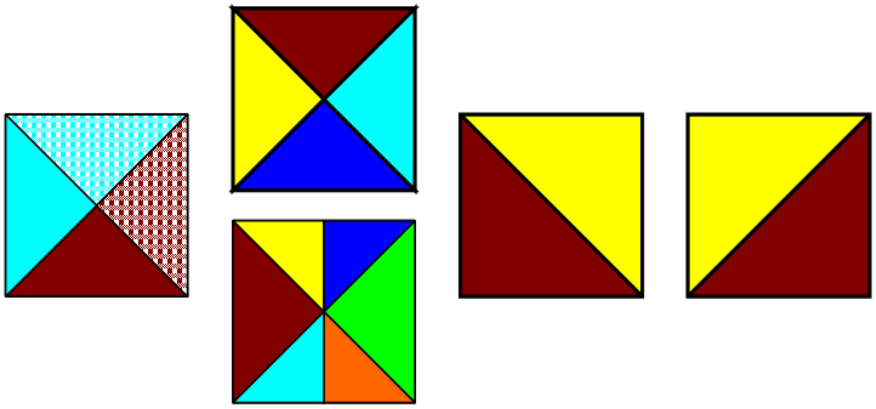
Where N is the number of pixels, and p_{ij} is the ground truth probability that $\mathbb{I}(l_i = l_j)$. In practice $p_{ij} = \frac{1}{K} \sum \mathbb{I}\left(l_i^{S_k} = l_j^{S_k}\right)$, the mean pixel pair relationship among the ground truth images. The PR index takes a value in the interval $[0, 1]$, and a PR index of 0 or 1 can only be achieved when all of the ground truth segmentations agree on every pixel pair relationship. A score of 0 indicates that every pixel pair in the test image has the opposite relationship as every pair in the ground truth segmentations, while a score of 1 indicates that every pixel pair in the test image has the same relationship as every pair in the ground truth segmentations.

As described in [7], the PR index has the desirable property that it does not allow arbitrary refinements or generalizations of the ground truth segmentations. In other words, if two pixels are in the same region in most of the ground truth segmentations, they are penalized for not being in the same region in the test segmentation, and vice versa. Also, the penalization is dependent on the fraction of disagreeing ground truth segmentations. Thus, there is a smaller penalty for disagreeing with an inherently ambiguous pixel pair than with a pixel pair on which all of the ground truths agree. An example of this refinement policy is shown in Fig. 4. Fig. 4a shows an image, Fig. 4b shows two test segmentations, and Fig. 4c shows the ground truth hand segmentations for that image. It appears as though the first ground truth labeling is based on texture, while the second is based on colour. The top test segmentation only has region divisions that exist in at least one of the two ground truth images, it has divided the image based on both colour and texture. The bottom test segmentation, however, has region divisions which exist in neither of the ground truth images. Thus, the top test segmentation will have a higher index than the bottom one.

This refinement policy is attractive for evaluating most segmentation tasks in which we would like to avoid arbitrary refining and coarsening of a segmentation. For example, for an object recognition system, it is important to differentiate between a segmentation which gives object-level regions, one which gives part-level regions, and one which gives even smaller over-segmented regions. However, if your application is not sensitive to different granularities, a different measure should be used, such as the LCE [14].

The PR index does however have one serious flaw. Note that the PR index is on a scale of 0-1, but there is no expected value for a given segmentation. That is, it is impossible to know if any given score is good or bad. The best we can hope to do is compare it to the score of another segmentation of the same image, but still we do not know if the difference between the two scores is relevant or not. Also, we certainly can not compare the score of a segmentation of one image with the score of a segmentation of another image. All of these issues are resolved with normalization to produce the Normalized Probabilistic Rand (NPR) index [6]. The NPR index uses a typical normalization scheme: if the baseline index value is the expected value of the index of any given segmentation of a particular image, then

$$\text{Normalized index} = \frac{\text{Index} - \text{Expected index}}{\text{Maximum index} - \text{Expected index}} \tag{6}$$



(a) (b) (c)

Figure 4: Synthetic example of permissible refinements: (a) Input image, (b) Segmentations for testing, and (c) ground truth set

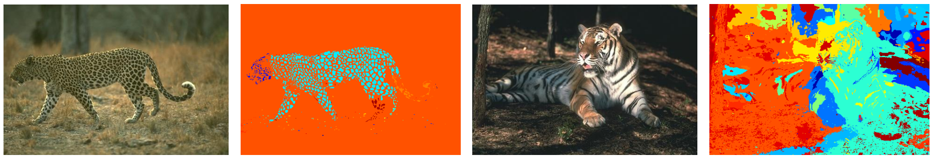


Figure 5: Examples of segmentations with NPR indices near 0.

The expected value of the normalized index is 0, so we know exactly when a segmentation is better than average or worse than average. Fig. 5 gives a couple of examples of segmentations for which the absolute value of the NPR index is small (< 0.003).

To be able to compare the segmentations of two different images, we need to include all possible images into the expectation calculation. So the expected value of the PR index as given in Eqn. 4 is:

$$\begin{aligned}\mathbb{E}\Big[\mathrm{PR}(S_{\text{test}}, \{S_k\})\Big] &= \frac{1}{\binom{N}{2}} \sum_{\substack{i,j\\ i \prec j}} \Big\{ \mathbb{E}\Big[\mathbb{I}\Big(l_i^{S_{\text{test}}} = l_j^{S_{\text{test}}}\Big)\Big] p_{ij} \\ &\qquad + \mathbb{E}\Big[\mathbb{I}\Big(l_i^{S_{\text{test}}} \neq l_j^{S_{\text{test}}}\Big)\Big](1 - p_{ij}) \Big\} \qquad (7) \\ &= \frac{1}{\binom{N}{2}} \sum_{\substack{i,j\\ i \prec j}} \Big[p'_{ij} p_{ij} + (1 - p'_{ij})(1 - p_{ij})\Big]\end{aligned}$$

Let Φ be the number of different images in the entire dataset, and K_ϕ the number of ground truth hand segmentations of image ϕ. Then p'_{ij} can be expressed as:

$$p'_{ij} = \frac{1}{\Phi} \sum_{\phi} \frac{1}{K_\phi} \sum_{k=1}^{K_\phi} \mathbb{I}\Big(l_i^{S_k^\phi} = l_j^{S_k^\phi}\Big) \qquad (8)$$

Since in computing the expected values no assumptions were made as to the number of regions in the segmentation, nor the size of the regions, and all of the ground truth data was used, the NPR indices are comparable across images, across segmentations, and across segmentation granularities. These are key properties which facilitate the comparison we perform in this paper.

4.2 Comparisons

Now that we have described the algorithms we wish to compare, and a measure for comparison, we can finally describe the specific comparisons we wish to perform. We believe that there are two key factors which allow for the use of a segmentation algorithm in a larger object detection system: correctness and stability.

Correctness is the obvious ability that we desire from any algorithm, the ability to produce results that are consistent with ground truth. Thus correctness is measured by the size of the NPR index. It has been often argued that the correctness of a segmentation is irrelevant, with the only relevant fact being whether or not the segmentation improves the recognition system. If the recognition system to be used is known, then it may be appropriate to measure the performance of the segmentation algorithm in conjunction with the rest of the system. However, there is value in weeding out potential segmentation algorithms which may give non-sensical results, and this can be done separately from the rest of the system. Also, it is often the case that it is not known beforehand which recognition algorithm will be best suited to a given problem, hence it is useful to have a generally well-behaved segmentation algorithm to use with multiple recognition algorithms. In addition, when evaluating a system with multiple components, it is often important to know which component specifically is causing a certain behaviour. Thus, we present here a comparison of the correctness of the segmentation algorithms apart from any recognition system.

Perhaps a more important indication of a segmentation algorithm's usefulness is its stability. If an algorithm gives reasonably correct segmentations on average, but is wildly unpredictable on any given image or with any given parameter set, it will be useless as a preprocessing step. We would like a preprocessing step to produce consistently correct segmentations of similar granularity so that any system built on top of it can predict its outcome. We require segmentations with low bias and low variance. There are two basic types of stability, stability with respect to parameters and stability across images. Stability with respect to parameters refers to achieving consistent results on the same image given different parameter inputs to the algorithm. In other words, we would like the algorithm to have low variability with respect to its parameters. Stability across images refers to achieving consistent results on different images given the same set of parameters. If a segmentation algorithm can be shown to be both correct and stable, then it will be a useful preprocessing step for many systems.

5 Experiments

The following plots explore each of the issues raised in the previous section. Note that the axes for each kind of plot have been kept constant so plots may be compared easily. In each experiment, the label 'EDISON' refers to the publicly available EDISON system for mean shift segmentation [2], the label 'FH' refers to the efficient graph-based

Figure 6: Examples of images from the Berkeley image segmentation database [5]

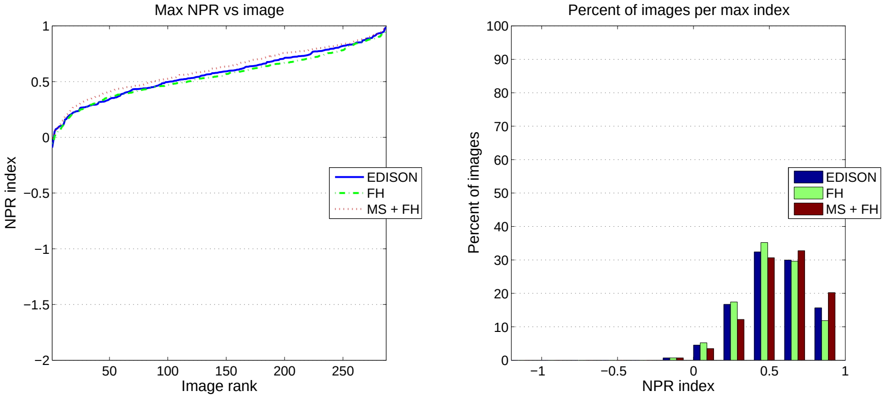
segmentation method by Felzenszwalb and Huttenlocher [4], and the label 'MS+FH' refers to our hybrid algorithm of mean shift filtering followed by efficient graph-based segmentation. All of the experiments were performed on the publicly available Berkeley image segmentation database [5], which contains 300 images of natural scenes with approximately five to seven hand segmentations of each image as ground truth. Examples of the images are shown in Fig. 6.

In all of the following plots we have fixed the spatial bandwidth $h_s = 7$, since it seems to be the least sensitive parameter and removing it makes the comparison more approachable. Also, although the FH algorithm as defined previously only had one parameter, k, we need to add two more. The FH algorithm requires the computation of the distances between points in feature space. Since our feature space consists of $\{x, y, L^*, u^*, v^*\}$, we need to put all of the dimensions into the same scale. Hence we will divide each dimension by the corresponding $\{h_s, h_r\}$ as in the EDISON system. So each algorithm was run with a parameter combination from the sets: $h_s = 7$, $h_r = \{3, 7, 11, 15, 19, 23\}$, and $k = \{5, 25, 50, 75, 100, 125\}$.

5.1 Maximum performance

The first set of experiments will examine the correctness of the segmentations produced by the three algorithms. We considered each of the three algorithms with a reasonable set of parameters. The left plot in Fig 7 shows the maximum NPR index on each image for each algorithm. The indices are plotted in increasing order for each algorithm, hence image rank 190 refers to the image with the 190th lowest index for a particular algorithm, and may not represent the same image across algorithms. The right plot in Fig 7 is a histogram of the same information, showing the number of images per maximum NPR index bin.

All of the algorithms produce similar maximum NPR indices, demonstrating that they have roughly equal ability to produce correct segmentations with the parameter set chosen. Note that there are very few images which have below-zero maximum NPR index, hence all of the algorithms almost always have the potential to produce useful results. These graphs also demonstrate that our parameter choices for each algorithm are reasonable.



(a) (b)

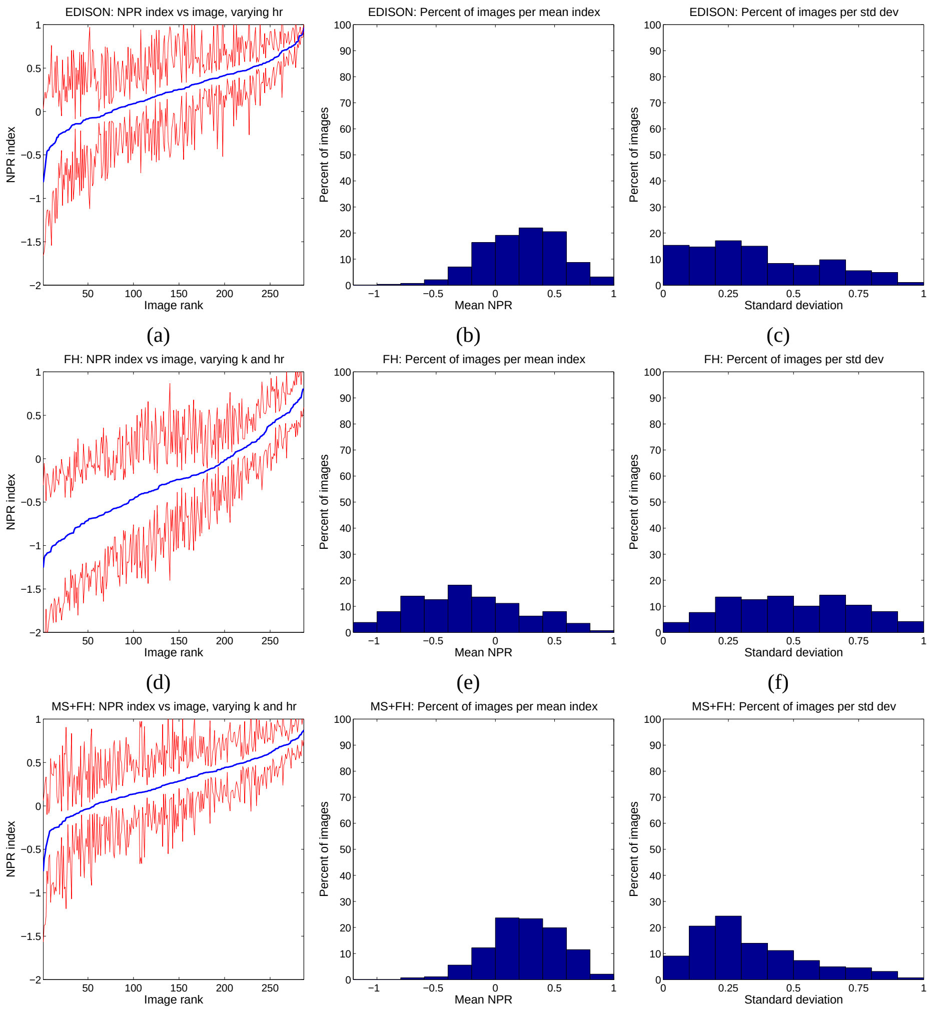
Figure 7: Maximum NPR indices achieved on individual images with the set of parameters used for each algorithm. Plot (a) shows the indices achieved on each image individually, ordered by increasing index. Plot (b) shows the same information in the form of a histogram.

5.2 Average performance per image

The next set of plots in Figs. 8-12 also consider correctness, but instead of the maximum index achieved they demonstrate the mean index achieved. The first plot in each row shows the mean NPR index on each image achieved over the set of parameters used (in increasing order of the mean), along with one standard deviation. The second plot in each row is a histogram of the mean information, showing the number of images per mean NPR index bin. An algorithm which creates good segmentations will have a histogram skewed to the right. The third plot in each row is a histogram of the standard deviations. This plot partially addresses the issue of stability with respect to parameters. A standard deviation histogram that is skewed to the left indicates that the algorithm in question is less sensitive to changes in its parameters. Using the means as a measure certainly makes us more dependent on our choice of parameters for each algorithm. However, while we can not say that we have seen the best or worst performance of any of the algorithms, we can compare their performance with identical parameters.

5.2.1 Average performance over all parameter combinations

Figure Fig. 8 shows the mean NPR plots for each of the three systems, EDISON, FH, and MS+FH, with averages taken over all possible combinations of the parameters h_r and k. Notice that these three rows of plots paint a different picture than the first set. While in terms of the maximum NPR index achievable on each image the three algorithms were comparable, in terms of mean NPR index the efficient graph-based segmentation (FH) falls behind the other two. Also note that the standard deviation histogram of the hybrid algorithm (MS+FH) is the most left-heavy, reflecting that it is the least sensitive to its parameters.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 8: Mean NPR indices achieved using each of the segmentation algorithms. The first row shows results from the mean shift-based system (EDISON), the second from the efficient graph-based system (FH), and the third from the hybrid segmentation system (MS+FH). Results from each algorithm are given for individual images over the parameter set of all combinations of $h_r = \{3, 7, 11, 15, 19, 23\}$ and $k = \{5, 25, 50, 75, 100, 125\}$. Plots (a), (d) and (g) show the mean indices achieved on each image individually, ordered by increasing index, along with one standard deviation. Plots (b), (e) and (h) show histograms of the means. Plots (c), (f) and (i) show histograms of the standard deviations.

5.2.2 Average performance over different values of the colour bandwidth h_r

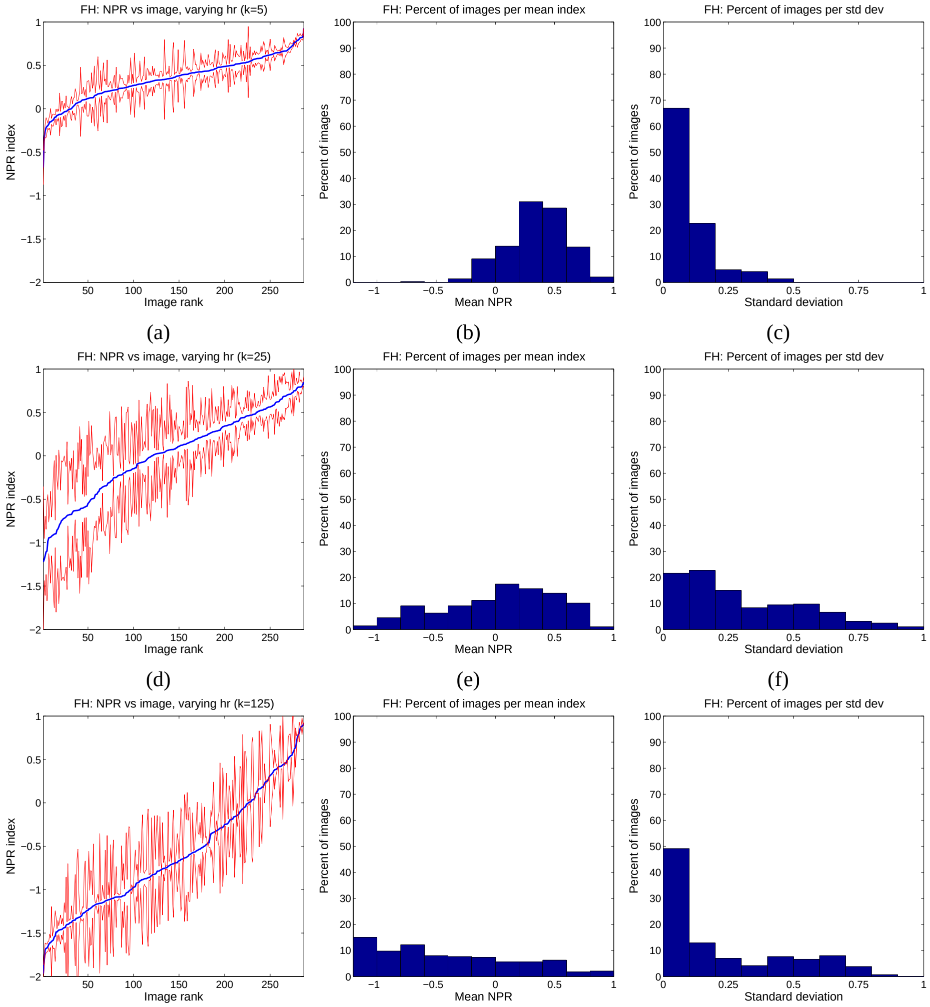
Although the above comparison is an interesting preliminary look at the data, it is actually biased since it is based on a different number of parameters for each algorithm, hence we now move on to more specific comparisons. This next comparison considers the NPR indices averaged over values of h_r, with k held constant. The plots showing this data for the EDISON method are the same as in the last section, in the first row in Fig. 8. Fig. 9 gives the plots for the efficient graph-based segmentation system (FH) for $k = \{5, 25, 125\}$. We only show three out of the six values of k in order to keep the amount of data presented reasonable. Fig. 10 gives the same information for the hybrid algorithm (MS+FH). The most interesting comparison here is between the EDISON system and the hybrid system. We would like to judge what impact the addition of the efficient graph-based clustering has had on the segmentations produced.

Notice that for $k = 5$, the performance of the hybrid (MS+FH) system in the first row of Fig. 10 is slightly better and certainly more stable than that of the mean shift-based (EDISON) system in Fig. 8. For $k = 25$, in the second row of Fig. 10, the performance is more comparable, but the standard deviation is still somewhat lower. Finally, for $k = 125$, in the third row of Fig. 10, the hybrid system performs comparably to the mean-shift based system. From these results we can see that the change to using the efficient graph-based clustering after the mean shift filtering has maintained the correctness of the mean shift-based system while improving its stability.

Looking at the graphs for the efficient graph-based segmentation system alone in Fig. 9, we can see that although for $k = 5$ the mean performance and standard deviation are promising, they quickly degrade for larger values of k. This decline is much more gradual in the hybrid algorithm.

5.2.3 Average performance over different values of k

The final set of plots of this kind in figures Fig. 11 and Fig. 12 examine the mean NPR indices as k is varied through $k = \{5, 25, 50, 75, 100, 125\}$ and h_r is held constant. Once again we only look at a representative three out of the six possible h_r values, $h_r = \{3, 7, 23\}$. Since the mean shift-based system doesn't use k, this comparison is between the efficient graph-based segmentation system and the hybrid system. It is immediately evident that the mean performance of the hybrid system is far superior to the efficient graph-based segmentation system, and that the results are much more stable with respect to changing values of k. Hence, adding a mean shift filtering preprocessing step to the efficient graph-based segmentation system is clearly an improvement.

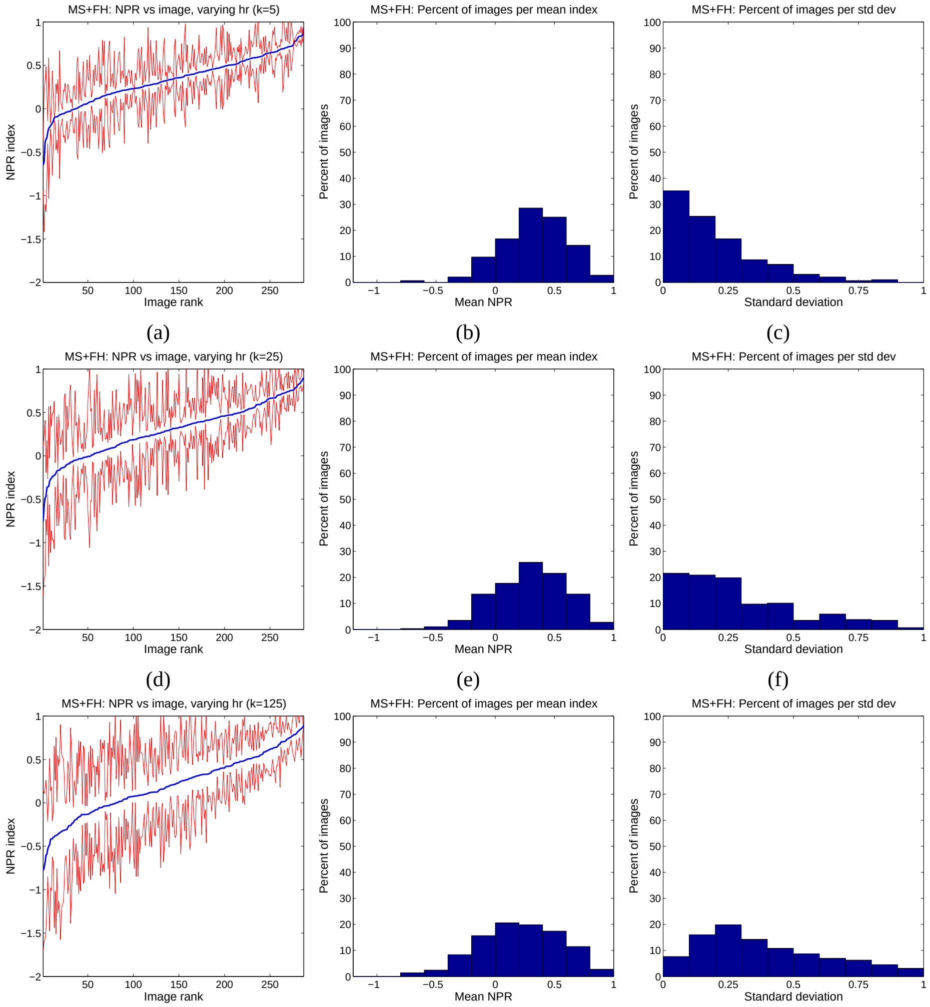


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 9: Mean NPR indices achieved using the efficient graph-based segmentation system (FH) on individual images over the parameter set $h_r = \{3, 7, 11, 15, 19, 23\}$ with a constant k. Plot (a) shows the mean indices achieved on each image individually, ordered by increasing index, along with one standard deviation. Plot (b) shows a histogram of the means. Plot (c) shows a histogram of the standard deviations.

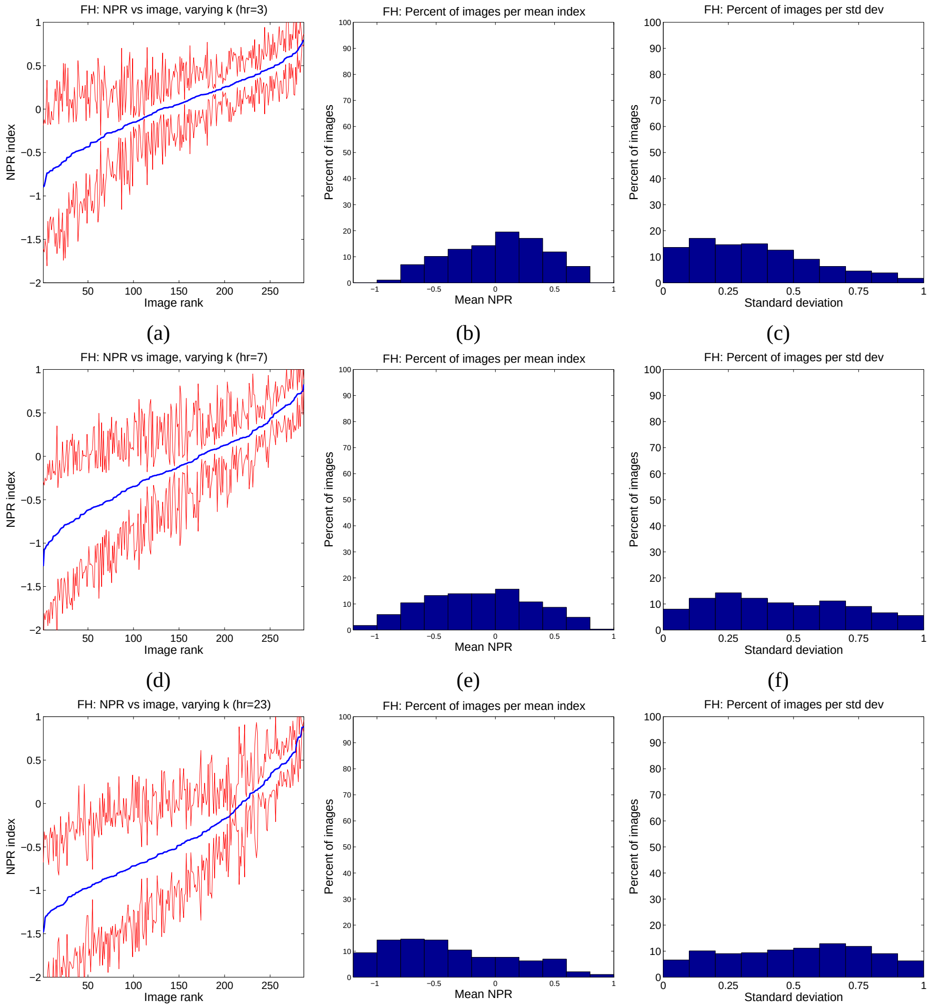


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 10: Mean NPR indices achieved using the hybrid segmentation system (MS+FH) on individual images over the parameter set $h_r = \{3, 7, 11, 15, 19, 23\}$ with a constant k. Plot (a) shows the mean indices achieved on each image individually, ordered by increasing index, along with one standard deviation. Plot (b) shows a histogram of the means. Plot (c) shows a histogram of the standard deviations.

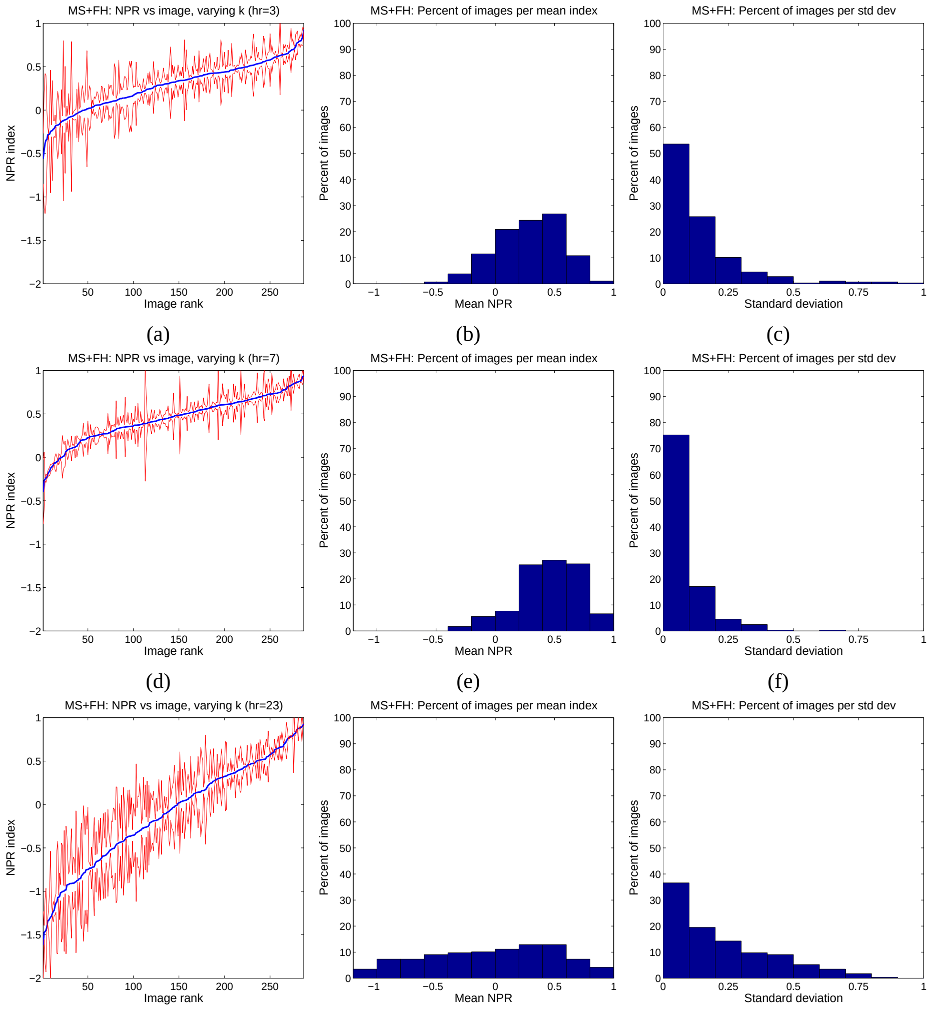


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 11: Mean NPR indices achieved using the efficient graph-based segmentation system (FH) on individual images over the parameter set $k = \{5, 25, 50, 75, 100, 125\}$ with a constant h_r. Plots (a), (d) and (g) show the mean indices achieved on each image individually, ordered by increasing index, along with one standard deviation. Plots (b), (e) and (h) show histograms of the means. Plots (c), (f) and (i) show histograms of the standard deviations.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 12: Mean NPR indices achieved using the hybrid segmentation system (MS+FH) on individual images over the parameter set $k = \{5, 25, 50, 75, 100, 125\}$ with a constant h_r. Plots (a), (d) and (g) show the mean indices achieved on each image individually, ordered by increasing index, along with one standard deviation. Plots (b), (e) and (h) show histograms of the means. Plots (c), (f) and (i) show histograms of the standard deviations.

5.3 Average performance per parameter choice

The final set of experiments looks at the stability of a particular parameter combination across images. In each experiment results are shown with respect to a particular parameter, with averages and standard deviations taken over segmentations of each image in the entire image database.

5.3.1 Average performance over all images for different values of h_r

The first three sets of graphs show the results of keeping k constant and choosing from the set $h_r = \{3, 7, 11, 15, 19, 23\}$. Fig. 13 shows the results of running the EDISON system with these parameters, averaged over the image set and with one standard deviation. Fig. 14 shows the same information for the efficient graph-based segmentation (FH) on the six possible values of k. Fig. 15 shows the same information for the hybrid (MS+FH) system.

As before, we can see that the hybrid algorithm gives slight improvements in stability over the mean shift-based system, but only for smaller values of k. We can also see that, except for $k = 5$, both the mean shift-based system and the hybrid system are much more stable across images than the efficient graph-based segmentation system.

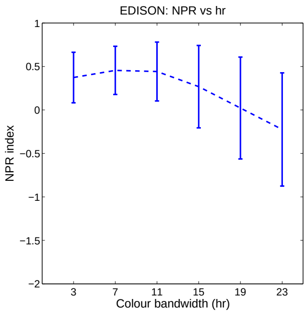
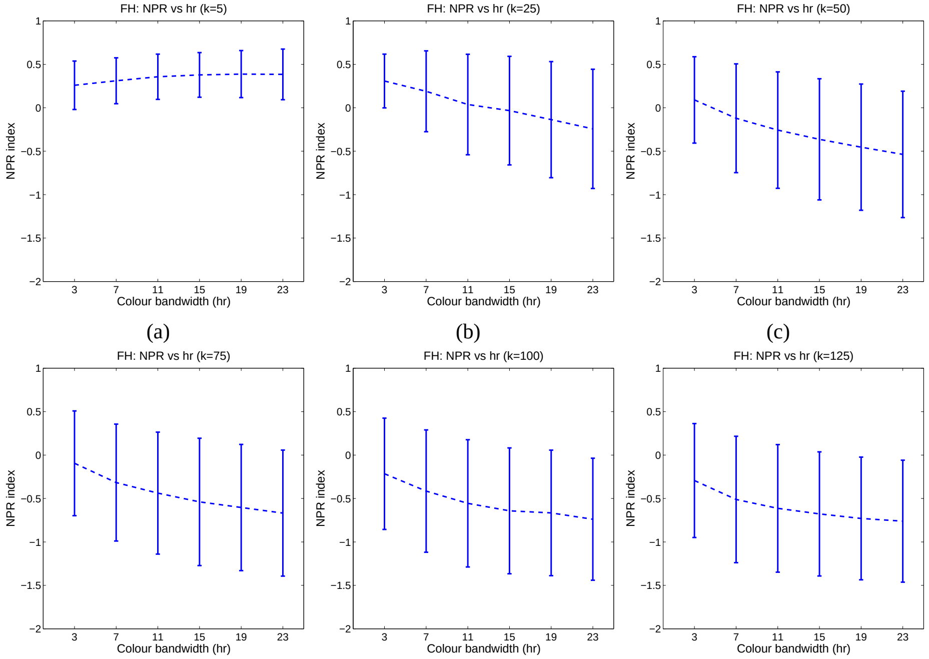


Figure 13: Mean NPR indices using the EDISON segmentation system on each colour bandwidth (h_r) over the set of images, with one standard deviation.

5.3.2 Average performance over all images for different values of k

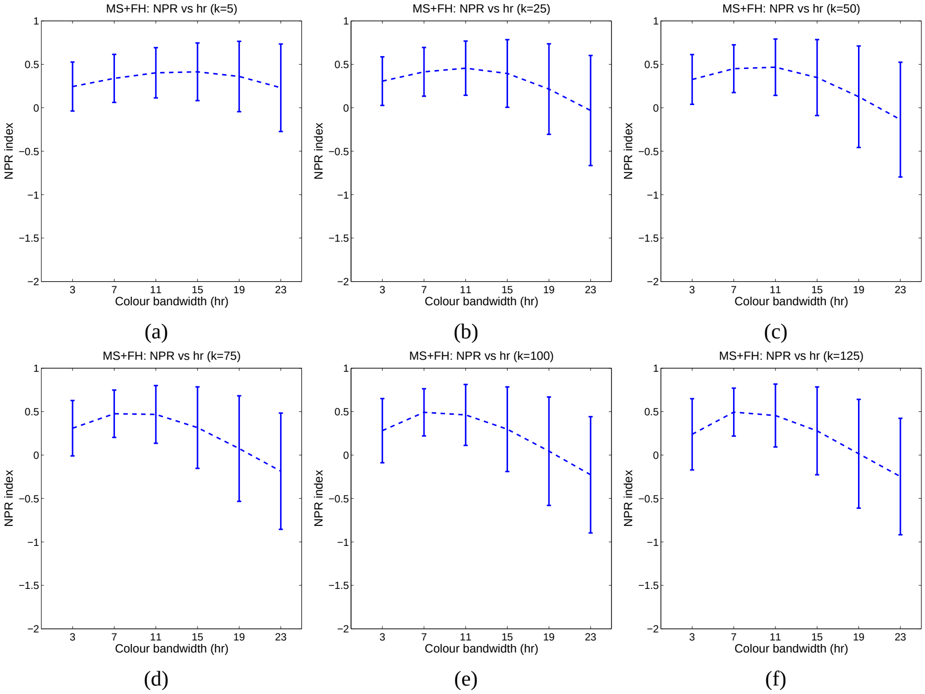
The last two sets of graphs in Fig. 16 and Fig. 17 examine the stability of k over a set of images. Each graph shows the average algorithm performance taken over the set of images with a particular h_r and each graph point shows a particular k. Once again we see that combining the two algorithms has improved performance and stability. The hybrid algorithm has higher means and lower standard deviations than the efficient graph-based segmentation over the image set for each k, and especially for lower values of h_r.



(a) (b) (c)

(d) (e) (f)

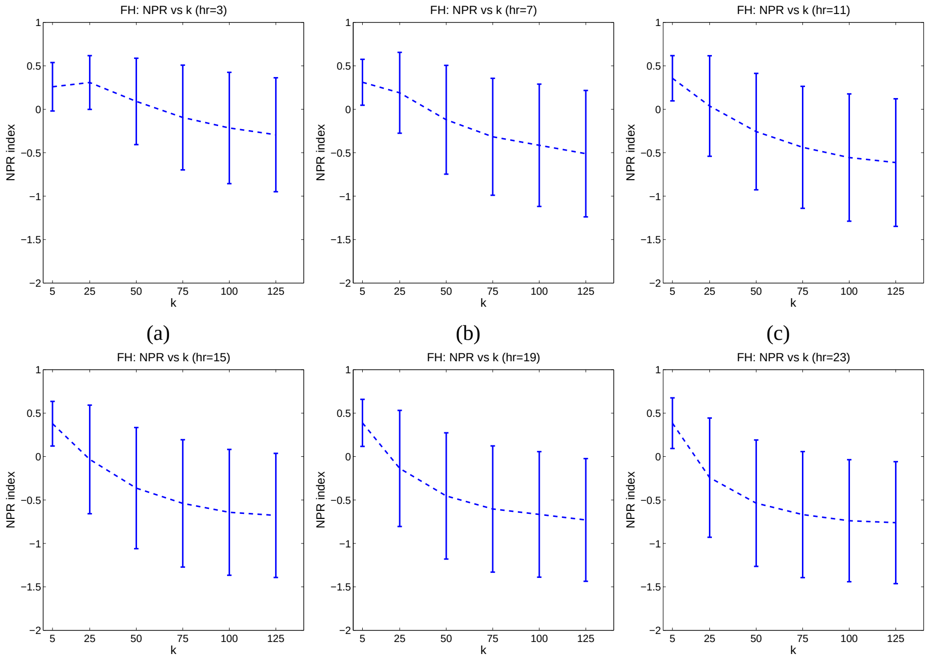
Figure 14: Mean NPR indices using graph-based segmentation (FH) on each colour bandwidth $h_r = \{3, 7, 11, 15, 19, 23\}$ over the set of images. One plot per value of k.



(a) (b) (c)

(d) (e) (f)

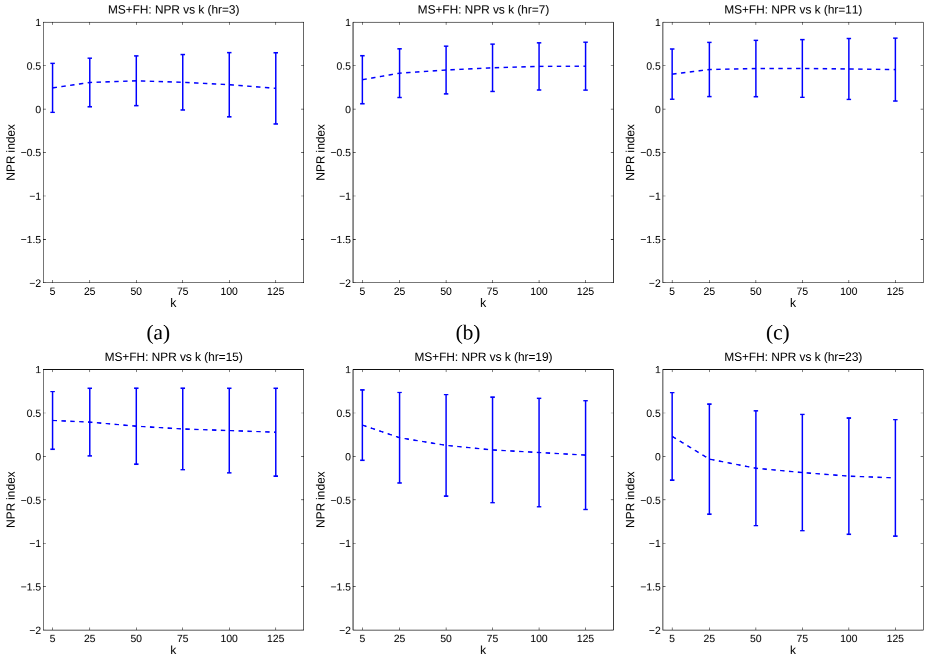
Figure 15: Mean NPR indices using hybrid segmentation (MS+FH) on each colour bandwidth $h_r = \{3, 7, 11, 15, 19, 23\}$ over the set of images. One plot per value of k.



(a) (b) (c)

(d) (e) (f)

Figure 16: Mean NPR indices using efficient graph-based segmentation (FH) on each of $k = \{5, 25, 50, 75, 100, 125\}$ over the set of images. One plot per value of h_r.



(a) (b) (c)

(d) (e) (f)

Figure 17: Mean NPR indices using hybrid segmentation (MS+FH) on each of $k = \{5, 25, 50, 75, 100, 125\}$ over the set of images. One plot per value of h_r.

6 Summary and Conclusions

In this paper we have proposed a framework for comparing image segmentation algorithms, and performed one such comparison. Our framework consists of comparing the performance of segmentation algorithms based on three important characteristics: correctness, stability with respect to parameter choice, and stability with respect to image choice. If an algorithm performs well with respect to all of these characteristics, it has the potential to be useful as part of a larger vision system. The measure used within our framework is the Normalized Probabilistic Rand index [6], which facilitates principled comparisons between segmentations of the same or different images, and generated with multiple algorithms and parameters. The NPR index does not place any restrictions on the number or distribution of regions in competing segmentations, and it generates scores which are easily interpretable, making it ideal for this task.

For our comparison task, we chose to compare two popular segmentation algorithms: mean shift-based segmentation [1] as implemented by the EDISON system [2] and a graph-based segmentation scheme [4]. We also proposed a hybrid algorithm which first performs the first stage of mean shift-based segmentation, mean shift filtering, and then applies the graph-based segmentation scheme, as an attempt to create an algorithm which preserves the correctness of the mean shift-based segmentation but is more robust with respect to parameter and image choice.

The first comparison we performed considered the correctness of the three algorithms. All three algorithms had the potential to perform equally well on the dataset [5] given the correct parameter choice. On average over the parameter set, however, the hybrid algorithm performed slightly better than the mean shift algorithm, and both performed significantly better than the graph-based segmentation. We can conclude that the mean shift filtering step is indeed useful, and that the most promising algorithms are the mean shift segmentation and the hybrid algorithm.

The second comparison we performed considered stability with respect to parameters. In this comparison, the hybrid algorithm showed less variability when its parameters were changed than the mean shift segmentation algorithm. Although the amount of improvement did decline with increasing values of k, the rate of decline was very slow and any choice of k within our parameter set gave reasonable results. Although the graph-based segmentation did show very low variability with $k = 5$, changing the value of k decreased its stability drastically.

Finally, we compared the stability of a particular parameter choice over the set of images. Once again we see that the graph-based algorithm has low variability when $k = 5$, however its performance and stability decrease rapidly with changing values of k. The comparison between the mean shift segmentation and the hybrid method is much closer here, with neither performing significantly better.

For the three characteristics measured, we have demonstrated that both the mean shift segmentation and hybrid segmentation algorithms can create realistic segmentations with a wide variety of parameters, however the hybrid algorithm has slightly improved stability. Thus, we would choose to incorporate the hybrid method into a larger system.

References

[1] D. Comaniciu, P. Meer, “Mean shift: A robust approach toward feature space analysis”, IEEE Trans. on Pattern Analysis and Machine Intelligence, 2002, 24, pp. 603-619

[2] C. Christoudias, B. Georgescu, P. Meer, “Synergism in Low Level Vision”, Intl Conf on Pattern Recognition, 2002, 4, pp. 40190

[3] B. Georgescu, I. Shimshoni, P. Meer, “Mean Shift Based Clustering in High Dimensions: A Texture Classification Example”, Intl Conf on Computer Vision, 2003

[4] P. Felzenszwalb, D. Huttenlocher,“Efficient Graph-Based Image Segmentation”, Intl Journal of Computer Vision, 2004, 59 (2)

[5] D. Martin, C. Fowlkes, D. Tal, J. Malik, “A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics”, Intl Conf on Computer Vision, 2001.

[6] R. Unnikrishnan, C. Pantofaru, M. Hebert, “A Measure for Objective Evaluation of Image Segmentation Algorithms”, CVPR workshop on Empirical Evaluation Methods in Computer Vision, 2005.

[7] R. Unnikrishnan, M. Hebert, “Measures of Similarity”, IEEE Workshop on Computer Vision Applications, 2005, pp. 394–400.

[8] D. Verma, M. Meila, “A comparison of spectral clustering algorithms”, Univ of Washington technical report, 2001.

[9] R. Kannan, S. Vempala, A. Vetta, “On clusterings - good, bad and spectral”, FOCS, 2000, pp. 367-377.

[10] M. Meila, J. Shi, “Learning segmentation by random walks”, NIPS, 2000, pp. 873-879.

[11] A. Y. Ng, M. I. Jordan, Y. Weiss, “On spectral clustering: Analysis and an algorithm”, NIPS, 2002, pp. 849-856.

[12] M. Meila, “Comparing clusterings”, Univ of Washington Technical Report, 2002.

[13] J. Shi, J. Malik, “Normalized cuts and image segmentation”, IEEE Transactions on Pattern Analysis and Machine Learning, 2000, pp. 888-905.

[14] D. Martin, “An Empirical Approach to Grouping and Segmentation”, Ph.D. dissertation, 2002, University of California, Berkeley.