Unsupervised estimation of segmentation quality using nonnegative factorization

Roman Sandler and Michael Lindenbaum
Computer Science dept. Technion,
Haifa 32000, Israel
{romats, mic}@cs.technion.ac.il

Abstract

We propose an unsupervised method for evaluating image segmentation. Common methods are typically based on evaluating smoothness within segments and contrast between them, and the measure they provide is not explicitly related to segmentation errors. The proposed approach differs from these methods on several important points and has several advantages over them.

First, it provides a meaningful, quantitative assessment of segmentation quality, in precision/recall terms, which were applicable so far only for supervised evaluation. Second, it builds on a new image model, which characterizes the segments as a mixture of basic feature distributions. The precision/recall estimates are then obtained by a nonnegative matrix factorization (NMF) process. A third important advantage is that the estimates, which are based on intrinsic properties of the specific image being evaluated and not on a comparison to typical images (learning), are relatively robust to context factors such as image quality or the presence of texture.

Experimental results demonstrate the accuracy of the precision/recall estimates in comparison to ground truth based on human judgment. Moreover, it is shown that tuning a segmentation algorithm using the unsupervised measure improves the algorithm's quality (as measured by a supervised method).

1. Introduction

Image analysis of a typical complex scene is much simpler if the image is partitioned into semantically meaningful parts. Much effort has been dedicated to the development of segmentation algorithms. The hardest and yet most useful form of segmentation uses only the image itself and does not rely on additional information. Segmentation algorithms are usually based on characterizing every image point using some local property and seeking a partition that makes this property regular (*i.e.*, smooth or obeying some model [3]) within the parts (segments) and irregular across the boundaries between them.

Leading approaches for finding such partitions include graph cut algorithms [7, 17], hierarchical segmentation [5] and active contour algorithms [21]. See comparative evaluation in [13]. This bottom up approach is limited, and better results may be obtained using model based information [5] or human interaction [7].

The quality of a segmentation result may be evaluated by comparing it to ground truth segmentations (supervised evaluation). Alternately, the evaluations may be done without any reference segmentation at all (unsupervised evaluation). We do not consider here task-dependent evaluation, which is useful in the context of specific applications.

Supervised, or ground truth based, evaluation is commonly used for empirical comparison of algorithms. The evaluated segmentation is compared to the reference segmentations using some type of set difference (*e.g.*, [1, 18, 25, 28]). Some methods focus on the boundaries between the segments and compare them to the reference boundaries, in statistical terms of miss and false positive, or precision and recall [18]. The recently available large image databases associated with manual segmentations [19] reveal the inconsistency of human segmentations, but still allow the quantitative comparison of different approaches on a common test bed [13]. The feedback from the supervised segmentation evaluation, enables learning/optimization based design of segmentation procedures [5, 18].

Unsupervised evaluation of segmentation does not use ground truth and is based only on the information included in the image itself. It is usually based on heuristic measures of consistency, related to Gestalt laws, between the image and the segmentation. Some examples are intra-region uniformity, inter-region contrast [6, 8], specific region shape properties (*e.g.*, convexity [14]), or combinations thereof [27]. More accurate judgement is possible when a statistical characterization of the underlying perceptual context is available [1, 12].

Unsupervised evaluation is considered rather weak for evaluating segmentation [28]. It is sensitive to texture and context, lacks the very informative ground truth, and does

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

not offer a clear interpretation: unsupervised evaluation algorithms provide a measure which, supposedly, increases monotonically with the perceptual quality of the segmentation. Yet, this measure is not explicitly related to the empirical error probability provided by, say, precision/recall. Unsupervised evaluation is rarely discussed as an end in itself; see, however, [6, 8, 26, 20]. It is more commonly discussed in the context of the numerous segmentation methods (see, *e.g.*, [21, 23]). In fact, every segmentation algorithm may be interpreted as an optimization of an unsupervised quality measure. To make the resulting segmentation algorithm efficient, such quality measures are often simplistic.

This paper proposes an unsupervised method for evaluating image segmentation, which is very different from previous unsupervised approaches. First, it provides a meaningful, quantitative assessment of segmentation quality, in precision/recall terms, which were applicable so far only for supervised evaluation. Second, it builds on a new image model, which characterizes the segments as a mixture of basic feature distributions. The precision/recall estimates are then obtained by a nonnegative matrix factorization (NMF) process. A third important advantage is that the estimates, which are based on intrinsic properties of the specific image being evaluated and not on a comparison to typical images (learning), are relatively robust to context factors such as image quality or the presence of texture.

Several supervised approaches proposed the collection of distributions characterizing the different objects, either by learning them for a class of objects (*e.g.*, [15]), or by using interaction on a single image [7], and using these distributions to improve the segmentation. The use of distributions for characterizing the segments is related to the proposed approach. The proposed approach is also (weakly) related to the recent segmentation algorithm which alternatively calculates the segmentation and the segment distributions [2]. Note, however, that this method focuses on the actual segmentation and relies on the obtained (unique) segmentation to get the distributions. Our method, on the other hand, does not rely on any hard decision regarding the segmentation and is therefore more robust.

Grouping quality was evaluated, in an unsupervised way, relative to the consensus of several grouping algorithms [26, 24]. This way, quantitatively meaningful evaluation may be obtained. Like these methods, our also uses multiple segmentations. Unlike the consensus approach, it relies on an explicit model, is much less sensitive to texture and edge detection errors, and does not require any consensus between the segmentations. In fact, it would perform well even if very few segments included in the input segmentations are reasonable.

The paper continues as follows: Section 2 introduces our image model. A background on NMF and the details of our implementation are given in section 3. Section 4 suggests an application framework for the proposed measure. Experimental results, discussion and conclusions are presented in sections 5 and 6.

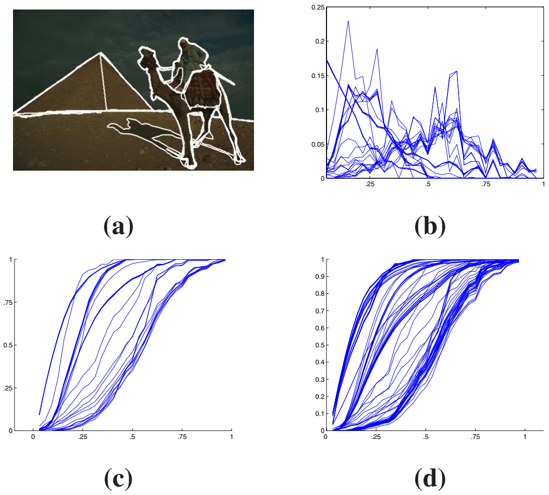


Figure 1. Edge strength distributions on different regions of an image. The regions are specified by manual (*e.g.* (a)) or automatic segmentations. Distribution densities on the segments specified by (several) manual segmentations (b). Each curve is associated with a distribution of a single segment. Cumulative distributions corresponding to manual segmentations (c). Cumulative distributions corresponding to incorrect segmentation of the same image (d). The ”thick” curves are clustered thin curves.

2. Framework

We consider the evaluation of segmentations such as those created by general purpose segmentation algorithms. Thus, a segmentation is a partition of the image into disjoint regions, separated by thin boundaries. We refer to the evaluated segmentation as the hypothesized or given segmentation.

Every point in the image may be characterized by some local properties, such as intensity, color, or texture, which may be represented by some feature vector. In this paper we used a boundary sensitive operator which provides a rough scalar characterization of texture and edge presence. (Actually, we use three operators, corresponding to texture, brightness and color. See section 3.3).

Consider a good segmentation of some image (possibly one out of several). Our basic model is that for every pixel within a particular segment, the local characterization may be regarded as an instance of a random variable, associated with some (discrete) distribution. The distributions associated with different segments are not necessarily different. The region around the boundary is considered as another segment and is characterized by another distribution. Intuitively, for boundary sensitive operators, we expect this distribution to put higher weights on high values. Note however that due to texture, the other distributions are not expected to be disjoint from the boundary distribution.

As an example, consider the distributions associated with several human segmentations of the same image (Fig. 1ab). Note that the distributions are clustered into several types. The clustering phenomenon is clearer in the less noisy cumulative representation of these distributions (Fig. 1c). The lower cumulative distribution curves, which rise only for relatively high values, are those associated with the manually marked boundaries. It should be emphasized that these distributions, characterizing the true segments and the true boundary, are not only unknown but are not even uniquely specified. We shall show, however, that estimating them leads to a quantitative, meaningful and yet unsupervised quality measure for a given segmentation.

Consider now an incorrect segmentation hypothesis. Every incorrect segment contains parts from several true segments. Therefore, we expect the incorrect segments to be characterized by mixtures of the true distributions; see Fig. 1d. The basic goal considered here is: Given a segmentation (and no ground truth), estimate its correctness. Specifically, we would like to estimate the accuracy of the inter-segment boundaries in precision/recall terms [18].

To carry out this seemingly impossible task, we first consider a simpler task. Assume that the number of true segments (including the boundary segment), k, as well as the associated distributions are known. All the mixture distributions lie in a subspace spanned by these true distributions. Therefore, given a particular hypothesized segment and its distribution, the mixture coefficients associated with the hypothesized segment may be obtained by solving an overconstrained system of linear equations. Then the precision and the recall may be easily calculated; see below.

Consider now the more difficult task, where the true distributions are not known. To find these true distributions, specify many (not necessarily correct) hypothesized segments and find their distributions. The subspace spanning all the hypothesized distributions is of dimension k and contains the true distributions. Note now, that for any choice of the true distributions, the mixture coefficient associated with every hypothesized distribution must be positive. Therefore, finding the true (hidden) distributions associated with the true segments is a nonnegative matrix factorization task.

Formally, let h_i be the operator response distribution in the i-th segment, represented as an n-bin histogram or column vector. Thus, $H = (h_1, h_2, \ldots, h_k) \in \mathfrak{R}^{n\times k}$ represents all the underlying (true) distributions on the image. Consider now some segmentation containing m segments (including the boundary). Let $H^* = (h_1^*, h_2^*, \ldots, h_m^*) \in \mathfrak{R}^{n\times m}$ be the matrix of the distributions associated with these segments. Then, H^* may be written as

$$H^* = HW, \tag{1}$$

where $W \in \mathfrak{R}^{k\times m}$ is a weight matrix. Practically, the number of true distributions is unknown and may be very large. Moreover, the measured distributions may be noisy. The factorization still holds as an approximation $H^* \approx HW$ for an effective value of k, which we estimate.

W.l.g. let h_1 and h_1^* be the histograms associated with the boundaries in the true segmentation and in the hypothesized one. Then, by definition,

$$Precision = w_{11} \quad Recall = \frac{\alpha_1 w_{11}}{\sum_j \alpha_j w_{1j}}, \tag{2}$$

where α_j is the size of the j-th segment.

Thus, the quality of a given hypothesized segmentation may be found by decomposing its operator response histogram matrix into two matrices H and W, representing the distributions associated with the true segments and the mixture coefficients, respectively.

3. Finding true segmentation distributions using nonnegative factorization

3.1. Algorithms for nonnegative factorization

The decomposition of the measured histogram matrix H^* into a mixture of basic histograms is a nonnegative matrix factorization task [16, 10, 4, 11]. This task is often formulated as follows: Given a nonnegative matrix $A \in \mathfrak{R}^{n\times m}$ and a positive integer $k < \min(m, n)$, find nonnegative matrices $H \in \mathfrak{R}^{n\times k}$ and $W \in \mathfrak{R}^{k\times m}$ which minimize the functional

$$f(H, W) = \frac{1}{2}\|A - HW\|_2^2. \tag{3}$$

The matrix pair $\{H, W\}$ is called a nonnegative matrix factorization of A, although A is not necessarily exactly equal to the product HW. Minimizing (3) is difficult for several reasons, including the existence of local minima as a result of the nonconvexity of $f(H, W)$ in both H and W, and, perhaps more importantly, the nonuniqueness of the solution. Additional information is commonly used to direct the algorithm to the desired solution [10].

The problem was introduced by Paatero [22] but got much attention only after its information theoretic formulation and the multiplicative update algorithm by Lee and Seung [16]. See the survey in [4]. The factorization is commonly done by iterative algorithms: one matrix (*e.g.*, W) is treated as a constant, getting its value from the previous iteration, while the other H is changed to reduce the cost $f(H, W)$. Then the roles of the matrices are switched. The algorithms differ mostly in the specific cost reducing iteration, and in the use of additional information.

3.2. Factorizing the histogram matrix

To carry out the factorization (1), we used a variation on the multiplicative update method [16] as well as several supporting techniques.

Much data is needed for successful factorization. Therefore, instead of factorizing the matrix H^*, associated with a single segmentation, we consider a larger matrix associated with several segmentations (of the same image!). Such segmentations are either available or may be created using a segmentation algorithm with different sets of parameters. H^* is thus redefined as an $n \times M$ matrix whose columns are the M histograms associated with all segments of all segmentations.

$$H^* = \left(h^*_1, h^*_2, \ldots, h^*_m, h^*_{m+1}, \ldots, h^*_M\right). \qquad (4)$$

The factorization (1) is now changed to $H^* = HW$, where H is an $n \times k$ matrix (unchanged) and W is a much larger $k \times M$ weight matrix. Clearly, for successful factorization, H^* should contain different combinations of true H vectors. Geometrically, the columns of H^* are points in the convex hull specified by the columns of H in $\Re^n$. To get a stable reconstruction of the convex hull, at least some of the points should be on its faces and preferably close to its vertices. Empirically, we found that the reconstruction is stable when a related condition holds: at least several of the segmentations are not completely wrong. That is, a substantial part (35% or more) of their hypothesized boundary overlaps with the true boundary. This usually means that at least one of the segments in the segmentation is correct.

For common segmentation sets, many segments are associated with very similar distributions. For an example, see Fig. 1d, where the middle cluster of curves corresponds to such similar distributions. Geometrically, this means that the center of the convex hull is over-represented. Such uneven representation leads to unstable and incorrect factorization. Following [11] we make the representation of combinations of H distributions more even by a dilution process which replaces every set of similar columns with a single representative. (Technically, two distributions are considered similar if their inner product is larger than 0.999.) Actually, some NMF algorithms, such as alternating least squares (ALS) NMF [4] as well as the algorithm used here, are not too sensitive to uneven representation. Yet, dilution decreases the size of H^* and much improves its computational efficiency. After the NMF is carried out, the full weight matrix W is found by a single least squares iteration.

Using constraints improves the accuracy of NMF tasks. Note that every column of both H and W should sum to 1. Moreover, we noticed that the many distributions associated with true segments are roughly unimodal, which suggests a parametric description. We found that approximating every one of the basic distributions as a skewed Gaussian ($h(x) = (a + bx)e^{-\frac{(x-\mu)^2}{2\sigma^2}}$) made the factorization more stable.

The results described here were obtained using the following variation on the multiplicative update method [16], which was nearly as accurate as the ALS algorithm [4] but much faster. The algorithm, formally described in Algorithm 1, makes several multiplicative update steps for each matrix and thus gets closer to the solution at each iteration. In this sense, it behaves similarly to ALS NMF algorithms.

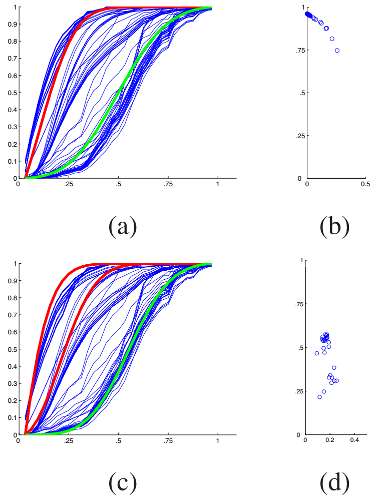
```
Algorithm 1 Factorization
Input: Histogram matrix $H^*$, model complexity $k$.
Dilute $H^*$ to $H'$ as described in 3.2.
Initialize $H \in \Re^{n \times k}$ with random columns from $H'$.
    Initialize $W \in \Re^{k \times m}$ with random values, and normalize its columns to sum to $1$.
Do $W$ and $H$ iterations until convergence. Each $W$ iteration repeats the basic $W$ update and a column normalization step several times. Each $H$ iteration repeats the basic $H$ update and a normalized skewed Gaussian fit to every column of $H$ several times. The basic updates are :
    $h_{ij} = \frac{h_{ij}(H' \cdot W^T)_{ij}}{(H \cdot W \cdot W^T)_{ij} + \epsilon} \quad w_{ij} = \frac{w_{ij}(H^T \cdot H')_{ij}}{(H^T \cdot H \cdot W)_{ij} + \epsilon}$   (5)
    where $\epsilon$ is a small constant.
Order columns of $H$ by $\mu + 2\sigma$.
Solve $H^* = HW$ for $W$ with least squares algorithm using the the obtained $H$.
Decompose $W$ into segmentation specific coefficients matrices $W_i$. Estimate the precisions ($P_i$) and the recalls ($R_i$) for each segmentation using $W_i$ and (2).
Output: $\{P_i\}, \{R_i\}$.
```

Given H, we still need to identify the distribution (column of H) associated with the boundary. We choose the distribution associated with highest $\mu + 2\sigma$ value.

3.3. Estimating model complexity using several modalities

The factorization algorithm described above decomposes the available histograms to sums of k basic histograms. We found, empirically, that assigning the correct value to the model complexity k is critical to the algorithm's success: For example, a too-high value of k may lead to a decomposition of the true boundary histogram into two or more estimated basic histograms, and to precision errors. The best value of k differs from image to image and depends on the type of boundary-sensitive operator (modality) as well.

We propose to use the consistency between modalities to specify the model complexity. In principle, if we use different boundary sensitive operators, we should still get the same precision if they function properly. In particular, we should get the same precision if the model complexities were chosen properly.



(a) (b)

(c) (d)

Figure 2. Choosing the model complexity: estimated basic histogram (brightness modality) with nonoptimal (a) and optimal (c) choices of k_1. The basic histograms are plotted in colored thick lines. The corresponding distribution of the precisions associated with different segmentations in the $(\delta_i, med_j(\hat{p}_{ij}))$ plane (b) and (d).

This criterion is not sufficient: a choice of $k = 1$ for all modalities would yield a uniform precision of 1 for them all, which is consistent but incorrect. To enhance the criterion we used the following observation: given a set of segmentations (of the same image), we know that some of them are correct while others are inaccurate. Therefore, their true corresponding precisions should differ.

We use three modalities: brightness, color and texture. The NMF (Algorithm 1) was applied to each of them separately, using three corresponding model complexities, k_1, k_2, k_3. Let $\hat{p}_{ij}$ be the estimated precision associated with the i-th segmentation and the j-th modality, and let δ_i be the standard deviation of the three precisions associated with the i-th segmentation. Let δ be the standard deviation of the median precision $\{med_j(\hat{p}_{ij})\}$, calculated over all segmentations. Then, one empirically selected way to quantify the considerations discussed above is to minimize

$$c(k_1, k_2, k_3) = \frac{max(max_i \delta_i, 0.2)}{\delta}. \qquad (6)$$

Note that the variance between modalities cannot dominate this expression even if it is very small. Limiting the numerator is necessary because, for some erroneous choices of model complexities, the three modalities may be consistent, leading to a very small $c(k_1, k_2, k_3)$. See Fig. 2.

3.4. Dealing with boundary inaccuracies

Typical segmentation algorithms distort the boundaries and provide somewhat inaccurate locations. That is, even for a segmentation providing roughly true segments, the boundary locations are inaccurate. This problem is recognized in supervised evaluation methods [18, 13], and some small location error margin is allowed.

Algorithm 2 Evaluation

```
Input: A test image $I$ and its segmentation(s) $s_i, i \in 1, \dots, M$.
1: If needed, add additional segmentations using a segmentation algorithm and different parameter sets.
2: Run three boundary sensitive operators (denoted different modalities), and measure their distribution within the segments. Construct three matrix $H_1^*, H_2^*, H_3^*$.
3: For all combinations of $k_1', k_2', k_3' \in \{2, 3, 4, 5\}$ Factorize every matrix $H_j^*$ using the corresponding $k_j'$ value, by applying Algorithm 1, and obtain the precisions $\{\hat{p}_{i,k_j'}\}$ and recalls $\{\hat{r}_{i,k_j'}\}$ of all segmentations. Choose the $(k_1, k_2, k_3)$ triple minimizing $c(k_1', k_2', k_3')$ (6).
4: Calculate: $P_i = median\left(\hat{p}_{i,k_j}\right)$ $R_i = median\left(\hat{r}_{i,k_j}\right)$
Output: $\{P_i\}, \{R_i\}$
```

Naturally, the problem arises here as well: the distribution evaluated on the inaccurate boundary is not the one characterizing the true boundary, and the distribution evaluated within a segment contains contributions from the boundary. Thus, we do not use the distribution of the boundary sensitive operator directly. Rather, to handle this difficulty, we replace the responses in the boundary points with the highest responses in their circular neighborhood ($r = 5$). Because we expect higher values on the boundary, the pixel contributing the maximal value is indeed likely to belong to the true boundary. The other segment distributions are calculated similarly, except that points which were considered when the boundary distribution was calculated are not considered this time.

A summary of the full factorization based evaluation algorithm is described in Algorithm 2

4. Application: a tool for algorithm design

As discussed above, every segmentation algorithm may be regarded as an optimization of an unsupervised segmentation quality criterion. Usually both the criterion and the optimization method depend on a set of parameters. Optimizing them for an ensemble of images may not give good segmentations for many images. The proposed unsupervised evaluation method may act as an independent referee, able to better choose a segmentation algorithm and its associated parameters for every image.

This way, the segmentation process becomes hierarchical. The external part uses the proposed evaluation, to specify a particular internal algorithm and tune its parameters. Any common algorithm may be used as the internal algo-

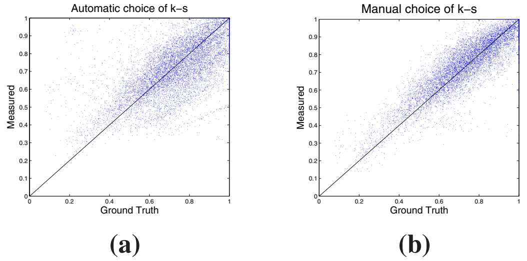


Figure 3. Precision estimation for synthetically degraded segmentations. (a) automatically found k values are used to estimate the precision. (b) optimal k values are used to estimate the precision.

rithm. Below, we consider a simplified version of such a hierarchical algorithm, where the internal algorithm itself is specified and only its parameters are optimized by the external part.In the section 5 we show that this approach indeed improves the performance of a mean-shift based segmentation algorithm [9].

We choose to optimize the internal algorithm by maximizing the F-value, $F = \frac{2P*R}{P+R}$. The F-value is considered to be a function of the parameter vector $\Omega = (\omega_1, ..., \omega_L)$, associated with the segmentation process, $s(\Omega)$. To optimize it we use a simple hierarchical gradient descent algorithm:

```
Init: Choose some parameters set $\{\Omega_i\}$, find the corresponding segmentations $\{s(\Omega_i)\}$, and use the NMF algorithm to evaluate the F values $\{F(s(\Omega_i)\}$
$\Omega = \arg\max_i F(s(\Omega_i))$, $\delta = \delta_0$
repeat
    for all $\omega_j$ do
        $\frac{\partial F}{\partial \omega_j} = \frac{F(\Omega + \delta \cdot \hat{\omega}_j) - F(\Omega - \delta \cdot \hat{\omega}_j)}{2\delta}$
    end for
    if $F(s(\Omega))$ is not a local maximum on a $\delta-$grid then
        $\Omega = \Omega + \delta \cdot \frac{\nabla F(\Omega)}{\|\nabla F(\Omega)\|}$
    else
        $\delta = \delta/2$
    end if
until $\delta < \epsilon$
```

In our implementation, the initial parameter set $\{\Omega_i\}$ was specified on a grid, and δ_0 is set as the grid spacing. $\epsilon = \delta_0/100$. It is important that the optimization be hierarchical, because F is nonconvex in the algorithm parameters. Note that the NMF runs only once. The F values calculated during the iterative process are calculated using the basic histograms computed during initialization.

5. Experiments

In all the experiments the estimated precision and recall values are obtained using Algorithm 2. Multiscale gradients were used as boundary sensitive operators. The texture gradient was based on Gabor filters. We expect, however, that the other operators would give similar results.

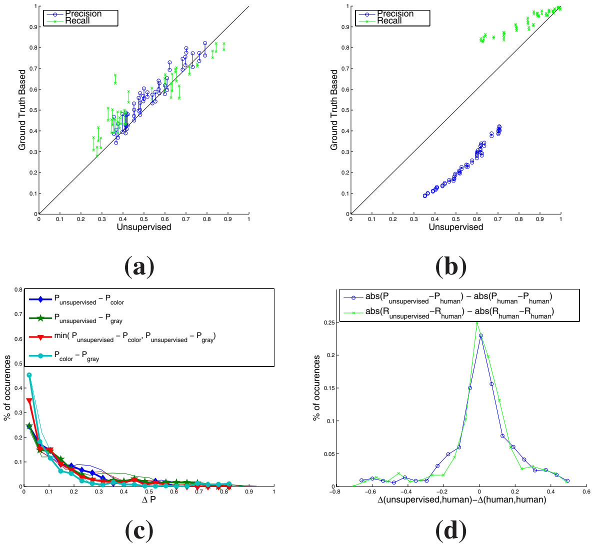


Figure 4. Precision/recall unsupervised estimates for segmentations obtained by EDISON algorithm, compared with supervised evaluation made with two types of ground truth: A typical good fit (a) and a typical erroneous fit (b). Histograms of the difference between the unsupervised estimates and the two supervised estimates, as well as the difference between the supervised estimates (c). The results are more accurate when the true precision is moderately high for at least several segmentations of the image (0.35 or larger, thick lines (c); the plot for recall is similar). A histogram of $\Delta(unsupervised, human)$ minus $\Delta(human, human)$ (d).

The accuracy of the proposed method is estimated with the help of the manual markings supplied as part of the Berkeley database, which serves as ground truth, and are considered below as "true". Note that the manual markings of different observers are different, and depend also on the type of image (color vs. grey level) used [19]. These inconsistencies naturally limit the testing accuracy.

5.1. Precision estimates for degraded images

The first experiment uses a set of degraded true segmentations, associated with known precision. Specifically, we took a manual segmentation associated with an image of the Berkeley database [18] and created 50 degraded versions of it by adding false boundary segments of random locations and lengths. (This was repeated for many manual segmentations of different images). Let L_i and L_h be the lengths of the $i-$th modified (degraded) boundary, and of the true (manually specified) boundary, respectively. The precision of the i-th segmentation hypothesis is therefore $p_i = \frac{L_i}{L_i - L_h}$. This precision, denoted synthetic precision, serves as a reference (ground truth).

The estimated precision was calculated and compared with the synthetic precision. The graph in Fig. 3a shows the estimated precision versus the synthetic one. Note that

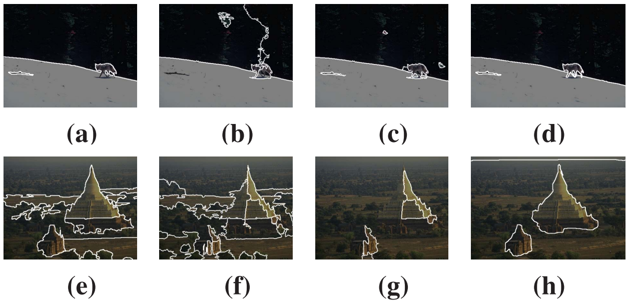


Figure 5. Mean shift segmentations for different parameter sets. The first line corresponds to an example where the proposed image-adaptive choice led to an improvement. Segmentation with ensemble-optimized parameters ($F = 0.55$)(b), and the most similar human segmentation (a). Segmentation with image adaptive parameters ($F = 0.87$) (c), and the most similar human segmentation (d). The second line shows the worst degradation caused by our algorithm. Parameter optimization over an ensemble of images led to (f) and to $F = 0.57$. Image adaptive optimization led to (c) and to $F = 0.23$. Note the high variability between the manual segmentation (e) and (h).

in most cases the approximation is very good. We found that the median of the relative error is 8.3%. Limiting our attention to the more relevant segmentations where the precision is relatively high, (0.35 or larger here), we found that the average precision error is 8.5%. This level of accuracy is comparable to the typical accuracy of manual (human) segmentation, as measured by comparing the segmentation of a single individual to that of the group [18]. Note that choosing a single human segmentation as ground truth makes the synthetic precision a little noisy. Manually optimizing the model complexities k_1, k_2, k_3 yields, obviously, more accurate precision estimates. See Fig. 3b. This improvement is not large (median: 6.7%), implying a reasonable estimate of the model complexities.

5.2. Estimating precision and recall from of automatically generated segmentations

We now turn to testing the accuracy of our precision/recall estimates applied to segmentations made by a common algorithm: the EDISON (mean shift) segmentation tool [9]. Each image was segmented 30 times using a different parameter set. (EDISON's parameters are spatial bandwidth, range bandwidth, and minimum region area).

The precision/recall were estimated for all segmentations. The unsupervised estimates were often close to those calculated using ground truth [18]. See Fig. 4a. (Two ground truth values, corresponding to "gray" and "color" markings [18], are plotted.) Sometimes the estimates are different but the unsupervised estimate is usually still monotonic in the true one. See Fig. 4b.

We compiled statistics for more than 100 images. The overall difference between the estimates and the ground

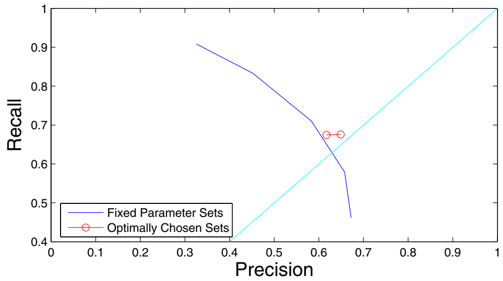


Figure 6. Precision/recall performance of constant parameter sets and the proposed algorithms on Berkeley images using EDISON.

truth precision is just a bit larger than the difference between the supervised calculation based on the two sets of ground truth. See Fig. 4c. Moreover, it seems that the larger errors are correlated with those of the humans (Fig. 4d). Note that a high histogram weight for a large value would imply that our method errs while the human observers are consistent. This does not happen frequently.

5.3. Application: optimizing EDISON

We now test the power of the proposed evaluation method for unsupervised image-specific optimization of the mean shift (EDISON) segmentation algorithm. 100 images from the Berkeley database were used. First, as a reference, we used several parameter sets suggested in [13] (and verified to be good indeed) and estimated the resulting precision/recall as described in [18]. See Fig. 6.

Then the segmentation quality was characterized, separately for each image, by the unsupervised precision/recall estimates, (not using any ground truth). This was done for 28 segmentations associated with different parameters in the 3D $(8, 16) \times (1, 4, 7, 10, 13, 16, 19) \times (100, 400)$ parameter grid (following [13]). The segmentation associated with the best (unsupervised) F value was selected. Usually, the best segmentations for the different images were not associated with the same parameters.

These best segmentations were evaluated using a supervised method (as in [18]). The performance is similar (slightly better) to that associated with the best non-image-adaptive parameter set; see Fig.6. For some images the results were better and for others they were worse; see Fig.5.

Starting from the selected parameters, we further optimized the algorithm using gradient descent, as described in section 4. The algorithm is somewhat improved; see Fig. 6.

The error is now distributed more evenly between the precision and the recall. This probably happens because, in the grid based optimization, the third parameter (minimal segment size) was chosen to be rather small (100 or 400 pixels) to prevent segmentation errors in many images with small true segments. This often led, however, to oversegmentation (low precision and high recall). The proposed adaptive choice of parameters allows the minimal segment size to be larger when necessary. This leads to more similar precision and recall values when the optimization is applied, and makes the overall segmentation better.

6. Conclusions

A fundamentally new approach to unsupervised estimation of segmentation quality is proposed. The approach builds on an intrinsic image model and a nonnegative matrix factorization process, and is able the predict precision/recall characterizations. Experiments, carried out on a large database, demonstrate the accuracy of the estimates and their application to tuning the segmentation process.

The segmentations optimized by the proposed measure are often consistent with manual segmentation. Inconsistencies often arise when the manual segmentations are themselves inconsistent. This seems to be the case when a lot of semantic knowledge is used.

An important property of the proposed hierarchical segmentation approach is the support of different segmentation methods.The best algorithm for a particular image is chosen, along with the optimal parameters. Diverse images can thus be efficiently segmented by the most efficient algorithm rather than by a complex general-purpose segmentation algorithm

References

[1] A. Amir and M. Lindenbaum. A generic grouping algorithm and its quantitative analysis. *PAMI*, 20(2):168–185, 1998.

[2] M. Andreetto, L. Zelnik-Manor, and P. Perona. Non-parametric probabilistic image segmentation. In *ICCV*, 2007.

[3] O. Ben-Shahar and S. Zucker. The perceptual organization of texture flow: A contextual inference approach. *PAMI*, 25(4):401–417, 2003.

[4] M. Berry, M. Browne, A. Langville, P. Pauca, and R. Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. *Computational Statistics and Data Analysis*, 52(1):155–173, September 2007.

[5] E. Borenstein, E. Sharon, and S. Ullman. Combining top-down and bottom-up segmentation. In *CVPRW*, page 46, 2004.

[6] M. Borsotti, P. Campadelli, and R. Schettini. Quantitative evaluation of color image segmentation results. *Pattern Recogn. Lett.*, 19(8):741–747, 1998.

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

[8] S. Chabrier, B. Emile, H. Laurent, C. Rosenberger, and P. Marche. Unsupervised evaluation of image segmentation application to multi-spectral images. In *ICPR*, pages 576–579, 2004.

[9] D. Comanicu and P. Meer. Mean shift: A robust approach toward feature space analysis. *PAMI*, 24(5):603–619, May 2002.

[10] I. Dhillon and S. Sra. Generalized nonnegative matrix approximations with Bregman divergences. In *NIPS*, volume 18, pages 283–290, 2006.

[11] D. Donoho and V. Stodden. When does non-negative matrix factorization give a correct decomposition into parts. In *NIPS*, 2003.

[12] J. H. Elder and R. M. Goldberg. Ecological statistics of Gestalt laws for the perceptual organization of contours. *J. Vis.*, 2(4):324–353, 8 2002.

[13] F. J. Estrada and A. D. Jepson. Quantitative evaluation of a novel image segmentation algorithm. In *CVPR*, pages 1132–1139, 2005.

[14] D. Jacobs. Robust and efficient detection of salient convex groups. *PAMI*, 18(1):23–37, 1996.

[15] M. P. Kumar, P. H. S. Torr, and A. Zisserman. Obj cut. In *CVPR*, pages 18–25, 2005.

[16] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. *NIPS*, 13:556–562, 2001.

[17] J. Malik, S. Belongie, T. K. Leung, and J. Shi. Contour and texture analysis for image segmentation. *IJCV*, 43(1):7–27, 2001.

[18] D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *PAMI*, 26(5):530–549, May 2004.

[19] 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. In *ICCV*, volume II, pages 416–423, 2001.

[20] P. Meer, B. Matei, and K. Cho. *Performance Characterization in Computer Vision*, chapter Input guided performance evaluation, pages 115–124. Kluwer, Amsterdam, 2000.

[21] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math*, XLII:577–685, 1989.

[22] P. Paatero and U. Tapper. Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.

[23] X. Ren and J. Malik. Learning a classification model for segmentation. In *ICCV*, pages 10–17, 2003.

[24] S. Warfield, K. Zou, and W. Wells, III. Simultaneous truth and performance level estimation (staple): An algorithm for the validation of image segmentation. *MedImg*, 23(7):903–921, July 2004.

[25] L. R. Williams and K. K. Thornber. A comparison of measures for detecting natural shapes in cluttered backgrounds. *IJCV*, 34(2-3):81–96, 1999.

[26] Y. Yitzhaky and E. Peli. A method for objective edge detection evaluation and detector parameter selection. *PAMI*, 25(8):1027–1033, August 2003.

[27] H. Zhang, S. Cholleti, S. A. Goldman, and J. E. Fritts. Meta-evaluation of image segmentation using machine learning. In *CVPR*, pages 1138–1145, 2006.

[28] Y. Zhang. A review of recent evaluation methods for image segmentation. *ISSPA*, 1:148–15, August 2001.