

# Ensemble Combination for Solving the Parameter Selection Problem in Image Segmentation

Pakaket Wattuya and Xiaoyi Jiang

Department of Mathematics and Computer Science  
University of Münster, Germany  
{wattuya,xjiang}@math.uni-muenster.de

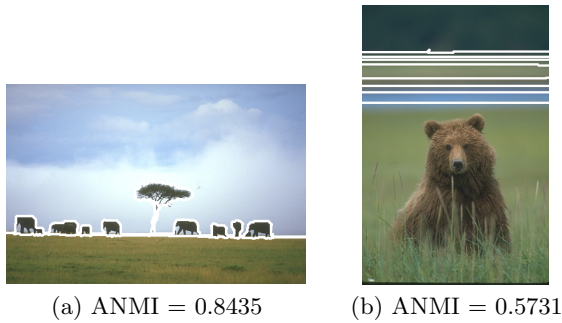
**Abstract.** Unsupervised image segmentation is of essential relevance for many computer vision applications and remains a difficult task despite of decades of intensive research. In particular, the parameter selection problem has not received the due attention in the past. Researchers typically claim to have empirically fixed the parameter values or train in advance based on manual ground truth. These approaches are not optimal and lack an adaptive behavior in dealing with a particular image. In this work we adopt the ensemble combination principle to solve the parameter selection problem in image segmentation. It explores the parameter space without the need of ground truth. The experimental results including a comparison with ground truth based training demonstrate the effectiveness of our framework.

**Keywords:** Image segmentation, parameter selection, ensemble combination concept, multiple segmentation combination.

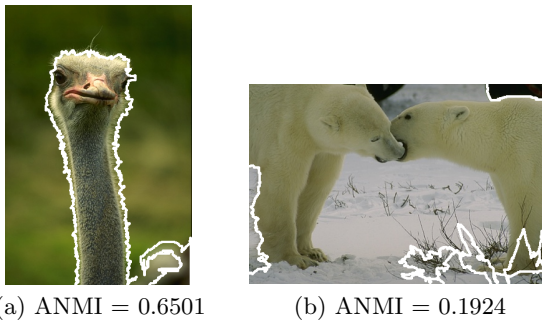
## 1 Introduction

Segmentation algorithms mostly have some parameters and their optimal setting is not trivial. One fundamental problem is in fact to find suitable parameter values, preferably on a per-image basis. This need can be illustrated by the two pairs of images shown in Figure 1 and Figure 2. Each pair is segmented using the FH algorithm [3] based on exactly the same parameter set (see Section 3.3 for the meaning of these parameters). However, while Figure 1(a) and 2(a) show a nearly perfect segmentation, we obtain a very bad segmentation in Figure 1(b) and 2(b). Note that in this work we use the average normalized mutual information (ANMI) [9] (to be defined later) to quantitatively evaluate the quality of a segmentation against corresponding human ground truth segmentations.

Despite of its importance, the parameter selection problem has not received the due attention in the past. Researchers typically claim to have empirically determined the parameter values (in an ad-hoc manner). More systematically, the optimal parameter values can be trained in advance based on manual ground truth. A subspace of the parameter space is explored to find out the best parameter setting (with the largest average performance measure). Since fully exploring



**Fig. 1.** Segmentations produced by the FH algorithm [3] given the same set of parameter values ( $\sigma = 0.7$ ,  $k = 700$ ,  $min = 1500$ )



**Fig. 2.** Segmentations produced by the FH algorithm [3] given the same set of parameter values ( $\sigma = 0.5$ ,  $k = 700$ ,  $min = 1500$ )

the subspace can be very costly, space subsampling [8] or genetic search [2] has been proposed. While this approach is reasonable and has been successfully practiced in several applications, its fundamental disadvantage is the assumption of ground truth segmentation. The manual generation of ground truth is always painful and thus a main barrier of wide use in many situations. Another class of methods assumes a segmentation quality measure, which is used to control a parameter optimization process [1].

In this work we propose a novel framework of parameter handling based on ensemble combination. No ground truth is assumed in our framework. The fundamental idea is not to specify a particular parameter set. Instead, we compute a set of segmentations according to a subspace sampling of the parameter space and then to combine them into a final segmentation. This framework will be detailed in the next section. Then, we report the results of a case study based on a particular segmentation combination method in Section 3. Afterwards, comparison experiments are described in Section 4 to demonstrate that our framework even outperforms the ground truth based training approach. Finally, some discussions conclude this paper.

## 2 Ensemble Combination Framework

Ensemble combination has been successfully applied to pattern classification [7] and prototype learning [5]. We adopt this general principle to implicitly exploring the (segmentation) parameter space without the need of ground truth. It is assumed that we know a reasonable subspace of the parameter space (i.e. a lower and upper bound for each parameter), which is sampled into a finite number  $\mathcal{N}$  of parameter settings. Then, we run the segmentation procedure for all the  $\mathcal{N}$  parameter settings and compute a combined result of the  $\mathcal{N}$  segmentations. The rationale behind our framework is that the combination result tends to be a good segmentation within the explored parameter subspace.

In this framework we do not explicitly determine the optimal parameter setting for a particular image. Instead, we try to reach an optimum out of the segmentation ensemble. One possibility is to compute an average, or more formally generalized median [5]. To some extent our combination method presented in the next section follows this line. The experimental results reported there show that the combined segmentation outperforms the majority of the input segmentations and is in many cases even superior to the best input segmentation. Given the fact that the optimal parameter setting may substantially vary among different images, our framework intends to achieve the highly desired adaptive behavior in dealing with a particular image.

## 3 Case Study: Multiple Segmentation Combination

For this case study the multiple segmentation combination algorithm in [11] is applied, which is briefly described in the following. The basis of this algorithm is the random walker algorithm for image segmentation [4]: Given (i) a graph  $G = (V, E, w)$ , where a node  $v_i$  corresponds to a pixel  $p_i$ , an edge  $e_{ij}$  connects two neighboring pixels in 4-neighborhood and a weight  $w_{ij}$  indicates similarity between the neighboring pixels  $v_i$  and  $v_j$  and (ii) a small number of  $K$  pixels with user-defined labels (or seeds), the algorithm labels unseeded pixels by resolving the probability that a random walker starting from each unseeded pixel will first reach each of the  $K$  seed points. A final segmentation is derived by selecting for each pixel the most probable seed destination for the random walker.

In principle there exists a natural link of our problem at hand to the random walker based image segmentation. The input segmentations provide strong hints about where to *automatically* place some seeds. Given such seed regions, we are then faced with the same situation as image segmentation with manually specified seeds and can thus apply the random walker algorithm [4] to achieve a quality final segmentation.

### 3.1 Multiple Segmentation Combination Algorithm

To develop the multiple segmentation combination algorithm [11] based on the random walker we need three components: (i) Generating a graph to work with,

(ii) extracting seed regions, and (iii) computing a final combined segmentation result.

### Component 1: Graph Generation

A weight  $w_{ij}$  indicates how probably the two pixels  $p_i$  and  $p_j$  belong to the same image region, similar to the co-association weight in [12]. Clearly, this can be guided by counting the number  $n_{ij}$  of initial segmentations, in which  $p_i$  and  $p_j$  share the same region label. Thus, we define the weight function as a Gaussian weighting:

$$w_{ij} = \exp(-\beta \cdot (1 - \frac{n_{ij}}{\mathcal{N}})) \quad (1)$$

where  $\beta$  is only a single parameter of an algorithm.

### Component 2: Seed Region Generation

We describe a two-step strategy to automatically generate seed points: (i) extracting candidate seed regions from  $G$ ; (ii) grouping them to form final seed regions to be used in the combination step.

Candidate seed region extraction: We build a new graph  $G^*$  by preserving those edges with weight  $w_{ij} = 1$  only ( $p_i$  and  $p_j$  have the same label in all  $\mathcal{N}$  initial segmentations) and removing all other edges. This step basically retains those edges between two neighboring nodes (pixels) which are most likely to belong to the same region. Then, we detect all *connected subgraphs* in  $G^*$  and regard them as a set of initial seed regions which are further reduced in the next step.

Grouping candidate seed regions: The number of candidate seed regions from the last step is typically higher than the true number of regions in an input image. Thus, a further reduction is performed by iteratively selecting the two candidate seed regions with the highest similarity value and merging them to build one single (possibly spatially disconnected) candidate seed region. Then, the similarity values between a new merged region and all remaining candidate seed regions are recomputed. The similarity  $s(C_i, C_j)$  between candidate seed regions  $C_i$  and  $C_j$  is defined as following:

$$s(C_i, C_j) = \overline{\{n_{ij}/\mathcal{N} \mid (p_i, p_j) \in C_i \times C_j\}}$$

where  $\overline{A}$  denotes the average of the set  $A$ .

A strategy for determining the final number  $K$  of seeds is to generate a series of results with  $K \in [K_{\min}, K_{\max}]$  seed regions. Then, each initial result is fed to the ensemble combination part of our algorithm (Component 3) to achieve a total of  $K_{\max} - K_{\min} + 1$  combination segmentations. Finally, we select an optimal one as the final combination segmentation (Section 3.2).

### Component 3: Segmentation Ensemble Combination

Given the graph  $G$  constructed from the initial segmentations and  $K$  seed regions, we apply the random walker algorithm [4] to compute the combination segmentation. The computation of random walker probabilities can be exactly performed without the simulation of random walks, but by solving a sparse,



symmetric, positive-definite system of equations. Each unseeded pixel is then assigned a  $K$ -tuple vector, specifying the probability that a random walker starting from that pixel will first reach each of the  $K$  seed regions. A final segmentation is derived by assigning each pixel the label of the largest probability.

### 3.2 Optimization of $K$

The quality of combination results can be evaluated in terms of consistency with the input ensemble. A good combination should share as much information as possible with the given  $\mathcal{N}$  segmentations [9]. We follow the concept of mutual information to quantify the statistical information shared between two segmentations. An optimality criterion is proposed to determine the optimal  $K$ .

The *normalized mutual information*  $\phi^{(NMI)}$  between two segmentations  $S_a$  and  $S_b$  is defined by

$$\phi^{(NMI)}(S_a, S_b) = \frac{\sum_{h=1}^{|S_a|} \sum_{l=1}^{|S_b|} |R_{h,l}| \log \frac{n \cdot |R_{h,l}|}{|R_h| \cdot |R_l|}}{\sqrt{\sum_{h=1}^{|S_a|} |R_h| \log \frac{|R_h|}{n} \sum_{l=1}^{|S_b|} |R_l| \log \frac{|R_l|}{n}}}$$

where  $R_h$  and  $R_l$  are regions from  $S_a$  and  $S_b$ , respectively,  $R_{h,l}$  denotes the common part of  $R_h$  and  $R_l$ , and  $n$  is the image size. The value domain of  $\phi^{(NMI)}$  is  $[0, 1]$ . Then, an information-share measure between a *single* segmentation  $\hat{S}$  and a *set*  $\Lambda = \{S_1, \dots, S_N\}$  of  $\mathcal{N}$  input segmentations can be defined as the average normalized mutual information (ANMI):

$$\phi^{(ANMI)}(\hat{S}, \Lambda) = \frac{1}{\mathcal{N}} \sum_{q=1}^{\mathcal{N}} \phi^{(NMI)}(\hat{S}, S_q)$$

We propose to select the *optimal* combination segmentation  $S^{k_{opt}}$  as the one with maximal average mutual information:

$$S^{k_{opt}} = \arg \max_{\hat{S}} \phi^{(ANMI)}(\hat{S}, \Lambda) \tag{2}$$

where  $\hat{S}$  covers all possible segmentations with  $K \in [K_{min}, K_{max}]$ .

If we replace  $\hat{S}$  by a universe  $\mathcal{U}$  of all possible segmentations of an image, then  $S^{k_{opt}}$  would represent the optimal segmentation in accordance with the generalized median concept of the input ensemble [5]. Therefore, our approach can be regarded as an approximation of generalized median segmentation by investigating the subspace of  $\mathcal{U}$  consisting of the combination segmentations for all possible  $K \in [K_{min}, K_{max}]$  only.

### 3.3 Experimental Results

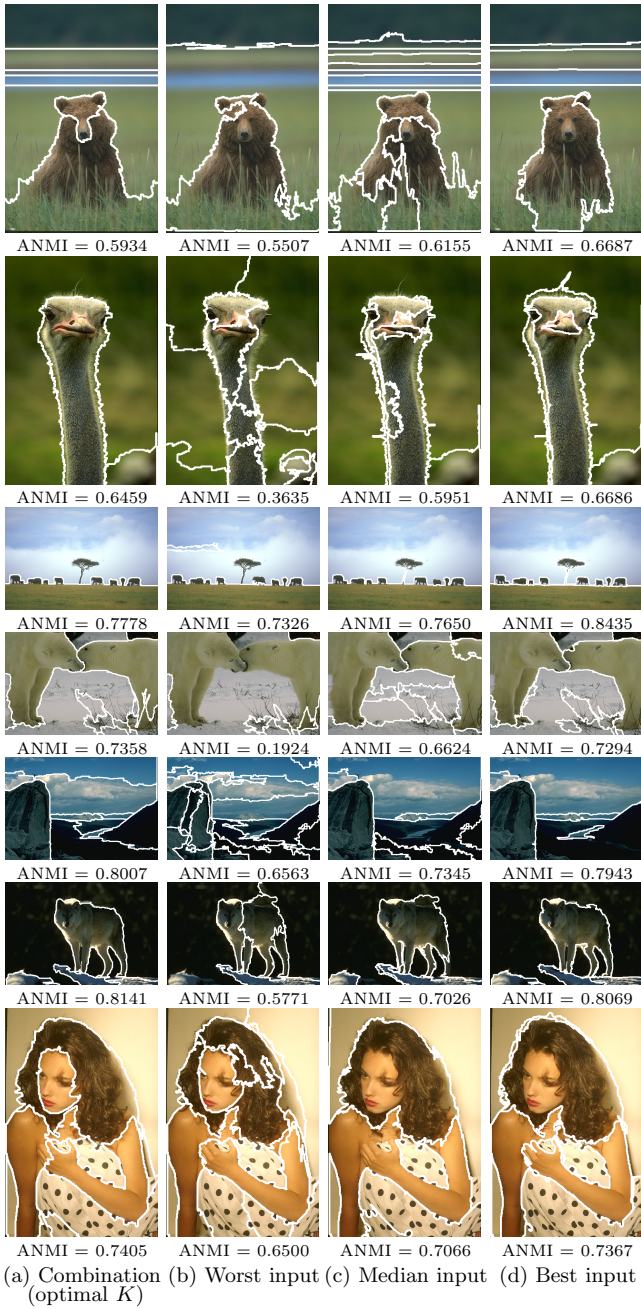
We tested our approach on 300 natural images of size  $481 \times 321$  from the Berkeley segmentation dataset [6], each having multiple manual segmentations. We apply  $\phi^{(\text{NMI})}$  to quantitatively evaluate the segmentation quality against the ground truth. One segmentation result is compared to all manual segmentations and the ANMI is reported. Larger ANMI values indicate better combination results that share more information with the ground truth.

The graph-based image segmentation algorithm [3] was used as the baseline segmentation method. This decision was made because of its competitive segmentation performance and high computational efficiency. The algorithm has three parameters: smoothing parameter ( $\sigma$ ), a threshold function ( $k$ ), and a minimum component size (*min\_area*). We obtained 24 segmentations of an image by varying the parameter values (fixing *min\_area* = 1500 and varying  $\sigma = 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$  and  $k = 150, 300, 500, 700$ ).

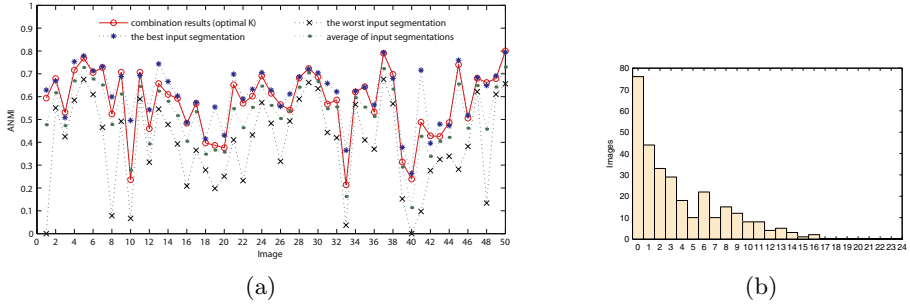
Figure 3 shows seven examples of combined segmentations produced by our method: column (a) using the optimality criterion in Eq. (2) to automatically determine  $K$ . For comparison purpose we also show the input segmentation with the worst, median and the best ANMI values (column (b)-(d)). Generally, we can observe a substantial improvement of our combination compared to the median input segmentation. These results demonstrate that we can obtain an “average” segmentation which is superior to the - possibly vast - majority of the input ensemble. Figure 4(a) shows a per-image performance of the first 50 images in the dataset, compared with the worst/best/average inputs. The combination performance tends to reach the best input segmentation. In some cases the combined segmentation even outperforms the entire input ensemble. This case is confirmed by Figure 4(b), which shows a statistic  $f(n)$ , indicating the number of images among the 300 test images, for which the combination segmentation is worse than the  $n$  best input segmentations. Remarkably, the combination segmentation outperforms all 24 input segmentations in  $f(0) = 78$  cases. For 70% (210) of all 300 test images, the goodness of our solution is beaten by at most 5 input segmentations only. This statistic is a clear sign of combination quality of our approach.

Another perspective is given in Figure 5, showing the average performance of all 300 images with regard to each of the 24 individual configurations (parameter settings). This implies that for all 24 parameter settings the combination approach always achieved improved results in average.

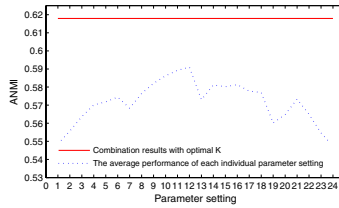
There is a single parameter of our combination algorithm,  $\beta$  in Eq. (1). A value of  $\beta$  affects the smoothness of contours in a combination result. A result with a lower  $\beta$  value is a smoother version of that with a higher value. A systematic study of the effect of  $\beta$  values over combination results was undertaken by running our algorithm with  $\beta = [10, 30, 60, 90, 120]$  on 100 images. The average standard deviation of performance of combination results is 0.0050. We thus conclude that the influence of  $\beta$  is rather small and set  $\beta$  to 30 for all our experiments.



**Fig. 3.** Exploring parameter space: (a) Combined segmentation with optimal  $K$ ; (b)-(d) show three input segmentations with the worst, median and the best ANMI scores, respectively



**Fig. 4.** Comparison (per image): Worst/best/average inputs & combination result. Combined using an optimality criterion (Eq. (2)). (b)  $f(n)$ : Number of images for which the combination result is worse than the best  $n$  input segmentations.



**Fig. 5.** Average performance of combined results over 300 images for each individual parameter setting

## 4 Combination Approach vs. Parameter Training

### 4.1 Automated Parameter Training

In recent years automated parameter training has become popular, mainly by probing a subspace of the parameter space by means of quantitatively comparing with a training image set with (manual) ground truth segmentation [2,8]. Assume that a reasonable parameter subspace is specified and sampled into a finite number  $\mathcal{N}$  of parameter settings. For each parameter setting candidate a performance measure is computed in the following way:

- Segment each image of the training set based on the parameter setting;
- Compute a performance measure by comparing the segmentation result with the corresponding ground truth;
- Compute the average performance measure over all training images.

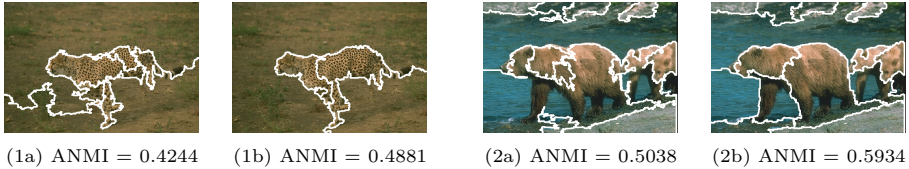
The optimal parameter setting is given by the one with the largest average performance measure.

### 4.2 Experimental Results

In this experiment series we investigate if our combination approach remains advantageous even if ground truth is available and the parameter training is

**Table 1.** Average performance measures of parameter training and combination approach on 3 test sets

Test set	Parameter training approach		Combination approach (Optimal $K$ )
	Training data	Testing data	
1	0.5716	0.5936	0.6252
2	0.5887	0.5921	0.6208
3	0.6144	0.5793	0.6078
average	0.5916	0.5883	0.6179

**Fig. 6.** Comparison of segmentation results produced by (a) trained parameters and (b) combination algorithm with optimality criterion in Eq. (2)

thus possible. The same experimental setting as in the combination approach is used (described in Section 3.3). The parameter subspace is sampled into 24 parameter settings. We apply a 3-fold cross validation in the training process described above. The dataset is randomly partitioned into 3 groups (100 images each). One group forms a 100-images training set while the rest two groups form a 200-images test set. By this way we have 3 different training sets with their corresponding test sets. The training procedure is then run 3 times on each training set to find its optimal parameter setting among the 24 parameter setting candidates.

The average performance measure over 100 images of each training set and 200 images of each test set are listed in the second and third column of Table 1, respectively. The fourth column of Table 1 shows the average performance of combination approach on each test set. It clearly demonstrates that the combination approach is even superior to automated parameter training. Firstly, the combination approach needs no ground truth. Secondly, even in case of ground truth, the combination approach is able to produce segmentations (on test data) with higher average performance than those of the training approach. This is an indication that the trained parameters based on manual ground truth lack an adaptive ability for dealing with the variation of an input image. Figure 6 shows comparison of two examples of segmentation results produced by (a) trained parameters and (b) combination algorithm with optimality criterion in Eq. (2).

## 5 Conclusion

In this work we have taken a step towards solving the parameter selection problem in image segmentation. Since empirically fixing the parameter values or

training in advance based on manual ground truth are not optimal and lack of adaptive behavior for dealing with the problem in a more general context, we have proposed to apply the concept of ensemble combination for exploring the (segmentation) parameter space without the need of ground truth. We verified our framework in a case study of segmentation combination. The experimental results confirm our expectation. Without using any ground truth information, our technique is able to produce segmentations with higher average quality than the training approach.

The focus of our current work is region-based image segmentation. It should be mentioned that our concept of ensemble combination is a general one. Given the demonstrated power we expect that it will be helpful towards solving the parameter selection problem in numerous other contexts. One such example is to explore the parameter space in a double contour detection problem [10]. We will consider further application scenarios in future.

## References

1. Abdul-Karim, M.-A., Roysam, B., Dowell-Mesfin, N.M., Jeromin, A., Yuksel, M., Kalyanaraman, S.: Automatic selection of parameters for vessel/neurite segmentation algorithms. *IEEE Trans. on Image Processing* 14(9), 1338–1350 (2005)
2. Cingue, L., Cucciara, R., Levialdi, S., Martinez, S., Pignalberi, G.: Optimal range segmentation parameters through genetic algorithms. In: *Proc. of 15th Int. Conf. on Pattern Recognition, Barcelona, vol. 1*, pp. 474–477 (2000)
3. Felzenszwalb, P., Huttenlocher, D.: Efficient graph-based image segmentation. *Int. Journal of Computer Vision*, 167–181 (2004)
4. Grady, L.: Random walks for image segmentation. *IEEE Trans. Patt. Anal. Mach. Intell.* 28(11), 1768–1783 (2006)
5. Jiang, X., Münger, A., Bunke, H.: On median graphs: Properties, algorithms, and applications. *IEEE Trans. Patt. Anal. Mach. Intell.* 23(10), 1144–1151 (2001)
6. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: *Proc. 8th Int. Conf. Computer Vision, vol. 2*, pp. 416–423 (2001)
7. Kuncheva, L.I.: *Combining Pattern Classifiers*. Wiley Interscience, Hoboken (2004)
8. Min, J., Powell, M., Bowyer, K.W.: Automated performance evaluation of range image segmentation algorithms. *IEEE Trans. on SMC – Part B* 34(1), 263–271 (2004)
9. Strehl, A., Ghosh, J.: Cluster ensembles – A knowledge reuse framework for combining multiple partitions. *J. Machine Learning Research* 3, 583–617 (2002)
10. Wattuya, P., Jiang, X.: A class of generalized median contour problem with exact solution. In: Yeung, D.-Y., Kwok, J.T., Fred, A., Roli, F., de Ridder, D. (eds.) *SSPR 2006 and SPR 2006*. LNCS, vol. 4109, pp. 109–117. Springer, Heidelberg (2006)
11. Wattuya, P., Jiang, X., Praßni, S., Rothaus, K.: A random walker based approach to combining multiple segmentations. In: *Proc. of 19th Int. Conf. on Pattern Recognition, Florida* (2008)
12. Fred, A., Jain, A.K.: Combining multiple clusterings using evidence accumulation. *IEEE Trans. Patt. Anal. Mach. Intell.* 27(6), 835–850 (2005)