2013

# Cluster Ensemble-Based Image Segmentation

Xiaoru Wang
*Beijing University of Posts and Telecommunications*

Junping Du
*Beijing University of Posts and Telecommunications*

Shuzhe Wu
*Beijing University of Posts and Telecommunications*

Xu Li
*Beijing University of Posts and Telecommunications*

Fu Lo
*Portland State University*, lif@pdx.edu



## Citation Details

Wang, X., Du, J., Wu, S., Li, X., & Li, F. (2013). Cluster ensemble-based image segmentation. International journal of advanced robotic systems, 10(7), 297.

INTECH
open science | open minds

OPEN ACCESS
ARTICLE

International Journal of Advanced Robotic Systems

# Cluster Ensemble-based Image Segmentation

Regular Paper

Xiaoru Wang[1,*], Junping Du[1], Shuzhe Wu[1], Xu Li[1] and Fu Li[2]

1 Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia, Beijing University of Posts and Telecommunications, Beijing, China
2 Department of Electrical and Computer Engineering, Portland State University, Portland, OR, USA
* Corresponding author E-mail: wxr@bupt.edu.cn

Received 3 Jul 2012; Accepted 19 Jun 2013

DOI: 10.5772/56769



Abstract Image segmentation is the foundation of computer vision applications. In this paper, we propose a new cluster ensemble-based image segmentation algorithm, which overcomes several problems of traditional methods. We make two main contributions in this paper. First, we introduce the cluster ensemble concept to fuse the segmentation results from different types of visual features effectively, which can deliver a better final result and achieve a much more stable performance for broad categories of images. Second, we exploit the PageRank idea from Internet applications and apply it to the image segmentation task. This can improve the final segmentation results by combining the spatial information of the image and the semantic similarity of regions. Our experiments on four public image databases validate the superiority of our algorithm over conventional single type of feature or multiple types of features-based algorithms, since our algorithm can fuse multiple types of features effectively for better segmentation results. Moreover, our method is also proved to be very competitive in comparison with other state-of-the-art segmentation algorithms.

Keywords Cluster Ensemble, Hypergraph, Image Segmentation, PageRank

## 1. Introduction

Image segmentation is the foundation of computer vision applications. Its purpose is to partition the image into several independent, meaningful and semantically related regions. An effective and accurate image segmentation algorithm is crucial for many applications, such as content-based image retrieval, object recognition, and object tracking. It also facilitates higher-level image analysis and understanding.

Image segmentation is a hot research topic in academia and industry, where many algorithms have been proposed and evaluated, such as threshold-based segmentation [1], edge-based segmentation [2], region growth segmentation [3, 4], graph-based segmentation [5] and clustering-based segmentation [6, 7]. These algorithms usually have two common problems: 1) a lack of efficient methods for fusing different image features during segmentation and 2) the spatial semantics of images are ignored in the algorithms.

The visual features of images include global features, such as colour and texture, and local features, such as SIFT features. During segmentation, each visual feature

has a different effect on different scenes. Algorithms based on a single type of feature can produce good results for some categories of images, but they cannot be applied to broad categories with good results. Fan [8] suggested that fusing multiple types of features could improve the performance and effectiveness of segmentation algorithms. Traditional segmentation methods [9, 10] usually employ a multidimensional feature vector based on several global features, such as colour and texture. However, the dimensions of these features are different, so the segmentation result may be affected more by features with higher dimensions. The other features might only have a limited effect on improving the final segmentation performance. Malisiewicz and Efros [11] did not use a multidimensional feature vector and instead they proposed to calculate the similarity between regions based on a single feature, before fusing the similarities using a positive linear combination function for segmentation. The problem of assigning a weight to the similarity for each feature is very challenging with this algorithm.

The "Bag-of-Words" concept is used widely for text analysis. Recently, it was introduced into image feature extraction and analysis. Most researchers [12-14] follow the approach of using clustered affine-invariant point descriptors as visual words. With this model, images are treated as documents, where each image is represented by a histogram of visual words. Cao [15] and Perronnin [16] produced visual words using global features (colour, texture and shape) and local features (SIFT), respectively. Each region had one visual word based on the global features and a set of visual words based on the local features. The global features and the local features are different, so the visual words produced are also very different. Simply combining these visual words cannot fully leverage the effectiveness of each feature, so the segmentation performance is hindered.

The spatial semantic information of pixels or regions is ignored by most existing segmentation methods. The spatial relationship of the words in a text may not affect content distillation seriously, but the spatial characteristics of images are critical for image segmentation. For example, two connected regions will usually be merged into one during segmentation if their visual features are similar, e.g., an ocean-sky image has the sky region at the top and the ocean region at the bottom. These two connected regions are similar in terms of their visual features. However, they are different objects semantically. If the segmentation process only considers the visual similarity and ignores the spatial information, the result could be incorrect.

After an in-depth analysis of these two common problems, we considered that the construction of a high-dimensional vector (visual words) from several features is inadequate because they cannot fully exploit every feature during segmentation. Instead, a better approach is to combine the segmentation results from every feature and deliver a better final result. Inspired by the cluster ensemble idea, we have built several subsegmentation tasks where each works on a single type of feature. Each feature may deliver the best result for some categories of images, so each subsegmentation task will deliver the best result for some images. The cluster ensemble method can enhance the strengths of some features and circumvent their weaknesses. We also considered that spatial information is a latent semantic for images, so it could be an effective approach for addressing the "semantic gap" issue between low-level visual features and high-level semantics. Therefore, this approach could combine the subsegmentation results effectively to provide the best final segmentation and achieve a much more stable performance over broad categories of images.

Based on the analysis above, we propose a novel cluster ensemble-based image segmentation algorithm. The major contributions of our work are as follows. 1) To improve the quality and stability of segmentation and overcome the problem of fusing different features, we introduce the cluster ensemble concept into image segmentation technology. We use a single but different type of feature, such as colour, texture or SIFT feature, to segment the image separately (subsegmentation), before the subsegmentation results are represented as a hypergraph model. The final segmentation is achieved using a spectral clustering algorithm with this hypergraph model. This algorithm effectively combines the subsegmentations based on different features, which could avoid the limitations of algorithms based on a single type of feature, a feature vector, or visual words and this approach could achieve a much stable performance for broad categories of images. Our algorithm also scales better because we could add more types of features if we find they are good for certain categories of images. The results showed that this method was highly robust to noise, exceptions and variable samples. 2) To exploit spatial information, we used the PageRank idea from Internet applications during image subsegmentation. First, we used the Normalized Cut (N-Cut) [17] algorithm to segment an image into several regions. The regions of the image are treated like web pages on the Internet and the links between the web pages (neighbouring regions) are computed based on the similarity between regions. In this algorithm, the importance of each page is calculated according to the semantic similarity between the linked pages. This is different from the original PageRank algorithm [18], which only considers the number of links. The merging process for regions selects the most similar page based on the semantic similarity from all the linked pages. The

effectiveness and speed of region merging is also improved by selecting the most semantically similar page using a greedy policy.

This paper is organized as follows. Section 2 describes the algorithm in detail. Section 3 contains details of our experiments and their results. Section 4 provides our conclusions.

## 2. The Cluster Ensemble-based Image Segmentation Algorithm

### *2.1 The Phases of the Algorithm*

The phases of our segmentation algorithm are shown in Fig. 1. This includes the following three phases.

1. Phase 1. Given an image, this algorithm begins with an initial over-segmentation algorithm, which partitions it into several homogeneous regions. To ensure that the pixels in a region belong to the same object and to avoid obtaining regions larger than the objects, we over-segment the image using the Normalized Cut (N-Cut) algorithm [17] initially. Actually, any over-segmentation algorithm [19, 20] could be used for this purpose as long as it can provide good over-segmentation results.
2. Phase 2. There are three parallel subsegmentation tasks during this phase because we select three types of features, i.e., colour, texture and SIFT. If we add more features, we only need to add more tasks. During each subsegmentation task, the feature will be extracted from each region. A linking graph is built where the regions are nodes in the graph. A link is added when the similarity between two adjacent regions is greater than a threshold, where the direction of the link is from the small region to the large region. Based on the PageRank algorithm, the importance of each region is computed according to the semantic similarity between a region and its linked regions. The linked regions will be clustered into clusters according to the linking relations and importance of the nodes. During each task, the subsegmentation results will be produced in parallel using each type of feature.
3. Phase 3. This is the cluster ensemble phase for the three subsegmentation results. As shown in Fig. 1, the subsegmentation results are represented as a hypergraph. The initial regions produced by N-Cut are the nodes while each cluster from the subsegmentation tasks is a hyperedge on the hypergraph. We achieve the final segmentation result by applying the spectral clustering algorithm to this hypergraph.

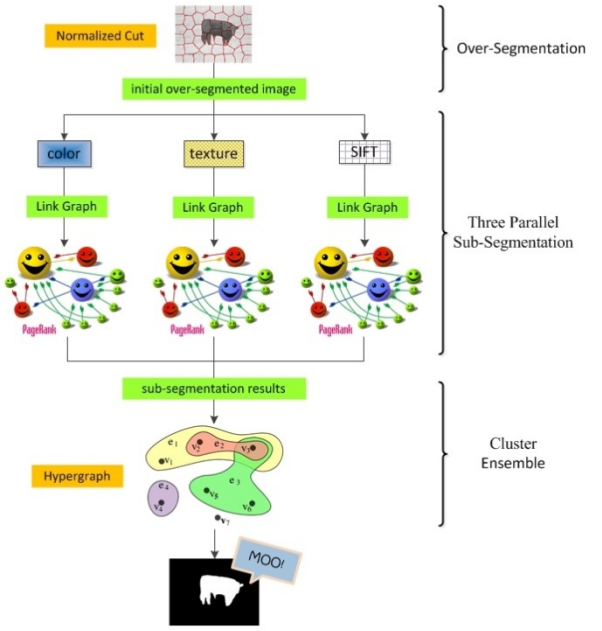


**Figure 1.** Cluster ensemble-based image segmentation algorithm

### *2.2 Feature-spatial Semantic-based Subsegmentation*

After applying the over-segmentation algorithm in [17], the image is separated into N regions. In this way, the problem of segmenting the image is cast into merging the regions into objects. Each region has multiple connected neighbours. It has been shown that it is better to merge according to the semantic similarity of regions. But how do we merge the regions based on their semantic similarity? We mimic the web page links used on the Internet. The regions can be viewed as pages on the Internet. When two connected regions have similar semantics, there will be a link between them. Otherwise, they will not have a link even when they are neighbours. Thus, we transform the spatial neighbouring relationships of the regions into a linking relationship based on their semantic similarity. Each subsegmentation process is performed as follows.

1. Extract each type of visual features for each region, e.g., colour, texture or SIFT feature.
2. Set a similarity threshold and compute the feature similarity between each pair of neighbouring regions.
3. Iterate through each region and based on the feature similarity of each with its neighbours:
   a) a link is added when the feature similarity between two neighbours is greater than the threshold.
   b) the direction of the link points from the small region to the large region. Thus, if we compare the areas of two regions $p_i$ and $p_j$, if $area(p_i) < area(p_j)$, the direction of the link is from $p_i$ to $p_j$.

After these steps, an image is represented as a linking graph with N nodes. The merging process can be viewed as the jumping probability from one page to another on the Internet. We use the jumping probability $P_R(p_i \to p_j)$ in the PageRank algorithm for this purpose. For page $p_i$ the criteria for it to pick the next page are as follows: it picks page $p_j$ with the highest $P_R$ probability as the next hop. The method is the same for our region merging process. Region $p_i$ may have several linked regions, which are the candidates for merging. Region $p_i$ will pick the linked $p_j$ with the highest semantic similarity for merging, as shown in Eq. (1).

$$P_R(p_i \to p_j) \propto \sigma_s(p_i, p_j) \tag{1}$$

Where $\sigma_s(p_i, p_j)$ is the semantic similarity between region $p_i$ and $p_j$. The probability of $p_i$ with $p_j$ is linearly proportional to the semantic similarity between $p_i$ and $p_j$. Like the PageRank algorithm for web pages, if region $p_i$ is linked to $p_j$, the semantic similarity assigned to $p_j$ by $p_i$ is the ratio of the semantic similarity between $p_i$ and $p_j$ to the sum of the similarities with all linked neighbours of $p_i$.

$$\sigma_s(p_i, p_j) \Big/ \sum_{k=N(p_i)} \sigma_s(p_k, p_i) \tag{2}$$

The semantic similarity of linked regions cannot be computed directly. Thus, we use the visual feature similarity between the linked regions to simulate the semantic similarity.

$$\sigma_s(p_i, p_j) \equiv E\left(sim(p_i, p_j)\right) \tag{3}$$

Where $sim(p_i, p_j)$ is the visual feature similarity of two linked regions $p_i$ and $p_j$. Function $E$ is a similarity evaluation function that tries to overcome the semantic gap problem. The semantics may not be the same when the visual features of two linked regions are similar. However, their semantics may be similar even when the visual features are not consistent. Thus, the semantic similarity and visual similarity are not equivalent. Therefore, we need to use a similarity evaluation function to compensate. Function $E$ can be a normal distribution function or a polygonal function.

Based on the above description, the equation for the PageRank algorithm has been modified to Eq. (4).

$$P_R(p_j) = \frac{(1-\varepsilon)}{n} + \varepsilon \times \sum_{p_i = N(p_j)} \frac{P_R(p_i) \times E\left(sim(p_i, p_j)\right)}{\sum_{p_k = N(p_i)} E\left(sim(p_k, p_i)\right)} \tag{4}$$

Where $P_R$ is the merging weight for region $p_j$ and $\varepsilon$ is a factor constant. The merging weight of one region is determined by its linked neighbours. When $p_j$ is similar to every linked neighbour, the weight for $p_j$ will be very high.

When we change the neighbour relationship to a linking relationship, we always assume that the direction of the link is from the small region to the large region. This assumption is for spatial semantics. The merging weights for the larger region will be higher than for small regions. Thus, the large region has a higher possibility of being merged with surrounding similar neighbours to form an object. This improves the accuracy of segmentation. We also use greedy policy by always picking the region with the maximum weight for merging, which speeds up the process.

Leveraging the linking relationship and the merging weight for each region ensures that image regions will be clustered into different clusters in a distributed manner and will produce several initial subsegmentation results.

### *2.3 Cluster Ensemble-based Subsegmentation Integration*

Cluster ensembles [21] combine multiple clustering results obtained from different sets of features to produce the final result. We use a cluster ensemble policy to combine the initial subsegmentation results, which are based on different visual features of the image, into the final segmentation result.

During this phase, the process is as follows.

1. The subsegmentation results are represented as a hypergraph model. As stated in Section 2.2, we use the colour, texture and SIFT features and perform the subsegmentation tasks in parallel. How do we map the subsegmentation results into a hypergraph? As shown in Fig. 2, the label vectors $C$, $T$ and $S$ represent the subsegmentation results based on colour, texture and SIFT, respectively. For example, the label vector $C[3, 1, 2, \ldots]^T$ represents the cluster label of each segmented region of the image. If the labels of regions $r_i$ and $r_j$ are the same, they will be assigned to the same cluster (merged into one region) by the colour-based subsegmentation task. For a hypergraph $G(V, E)$, the vertices are the regions merged, i.e., $r_1$, $r_2$, and $r_i \in V$. Set $E$ contains the set of hyperedges and $E = \{\{C^{(p)}\}, \{T^{(q)}\}, \{S^{(r)}\}\}$. Each label vector, such as C, has $P$ clusters and each cluster is represented as a $C^{(p)}$, $p = 1, 2, \ldots, P$. We can construct the binary membership indicator matrix $\{C^{(p)}\}$ where each cluster $C^{(p)}$ obtained by subsegmentation is represented as a hyperedge (column). Each column vector, such as $C^{(p)}$, $T^{(q)}$ and $S^{(r)}$, specifies a hyperedge, where 1 indicates that the vertex corresponding to the row is part of that hyperedge while 0 indicates that it is not. All entries in a row of the matrix $\{C^{(p)}\}$ add up to one. Thus, each cluster is mapped onto a hyperedge and the set of clusters is added to a hypergraph.

2. Spectral clustering based on hypergraph integration. When three sub segmentation results are represented as a hypergraph, we use the spectral clustering algorithm to combine the results of subsegmentation into the final segmentation. According to spectral clustering theory, the assignment of two regions to one cluster means that these two regions are similar so they can be merged during several subsegmentation tasks based on different features. Therefore, the integration result will merge them into one region. By contrast, if the two regions are only merged in a few subsegmentation tasks, this means that they may belong to different objects so they should not be merged. For example, two green regions may be grass or bushes. These regions will be merged during a colour-based subsegmentation task, but they will not be merged by texture and SIFT-based subsegmentation tasks. Thus, they will not be merged after the integration.

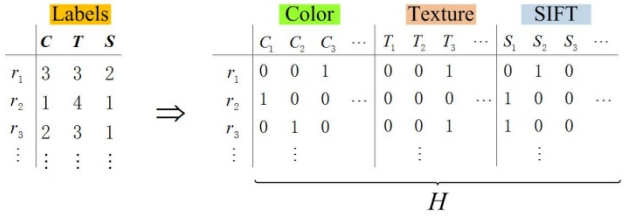


**Figure 2.** Hypergraph modelling of three subsegmentation results for cluster ensembles

*2.4 The Cluster Number of a Cluster Ensemble*

Optimal combined clustering should share the most information with the original clustering of subsegmentations. During segmentation, the sum of the differences for the regions in the object should be lowest when the linked regions with similar features are clustered into one object. Good segmentation requires that all regions are clustered into several objects correctly, so the sum of the differences for all clusters should be lowest. This also applies to the subsegmentations. The best result for all the subsegmentations is the one with the lowest sum of the differences. The number of clusters for this subsegmentation may be used as the cluster number for the final result. This process is performed as follows.

1. For each image, calculate the sum of differences of each subsegmentation result as $S_c$, $S_t$ and $S_s$. The three subsegmentations use different features, so the differences are computed using different scales. Thus, we need to normalize them using a Gaussian function.
2. After normalization, we find the subsegmentation with the smallest sum of differences to get the cluster number, $K_m$.
3. The final cluster number is set as $K_m$ in the cluster ensemble process.

## 3. Experiments

To evaluate the proposed approach, extensive experiments have been conducted on four different data sets and comparisons with state-of-the-art approaches have also been performed. Segmentation results from diverse images are presented for intuitive and perceptual judgments, while F-measure and amount of fragmentation are adopted for quantitative evaluation.

*3.1 Data sets*

Four publicly available data sets are exploited in our experiments, which contains a great diversity of images, so that the assessment can be more objective and convincing.

1. Berkeley Segmentation Data Set (BSDS500) [22]. This data set is probably the most used one and is very challenging. It includes 500 natural images of all categories, each of which is accompanied with five ground truth segmentations.
2. Weizmann Segmentation Evaluation Database [23]. There are 200 images in this data set and there are also three manual segmentations for each image. Half of the images contain only a single object in the foreground, the size of which varies from image to image, while the other half contain two objects that are also in different sizes. These objects differ from their surroundings in some or at least one type of low-level features, e.g., texture, colour, intensity, etc. Therefore, segmentation algorithms based on a single type of feature will have great difficulty achieving stable performances on this data set.
3. Weizmann Horse Database [24]. The data set consists of 328 images of horses that vary in poses, sizes and backgrounds. All the images are manually segmented.
4. Microsoft Research Cambridge Object Recognition Image Database (MSRC) [25]. In this database, a variety of digital photographs are grouped into categories, including trees, cows, sheep, cars, flowers, etc. The sizes of the images are generally 640 × 480 and they are downsized by half for processing efficiency in our experiments.

*3.2 Evaluation Scheme*

To demonstrate the effectiveness of the proposed approach, abundant segmentation results from all four data sets are presented for intuitive and perceptual judgments along with comparisons with several state-of-the-art algorithms, i.e., mean shift [26], normalized cuts [17], Gpb [27] and spatial-LTM [15]. Segmentation evaluation can be subjective because people usually have different understandings towards the same image and such distinctions in semantics lead to inconsistent segmentation evaluation. To avoid such divergence, we only focus on the most salient objects in each image.

Quantitative measures are also adopted for evaluation. F-measure [23] is used to assess the consistency between segmentation results and ground truth segmentations. By denoting the precision and recalling the values of segmentation by P and R respectively, the corresponding F-measure is defined as

$$F = \frac{2RP}{P+R} \tag{5}$$

In addition, the amount of fragmentation, which is defined as the number of segments needed to cover a single object, is computed as well.

*3.3 Features and Settings*

As previously mentioned, we need to partition each image into over-segmented regions first. There are several methods that can be used to obtain an over-segmentation, such as those from [19, 20]. Here we use N-Cut [17]. The number of over-segmented regions for each image is set to 50 in our experiments.

With regards to the subsegmentation tasks, three features are employed in the experiments, colour, texture and SIFT. Note that these features can be simply replaced by others or more can be added since the subsegmentation scheme does not depend on a specific type of feature and all the subsegmentation results are integrated using spectral clustering on the constructed hypergraphs. The proposed approach enjoys great flexibility and extensibility.

A colour histogram represents the number of pixels that have colours in each of a fixed list of colour ranges. It can be built for any kind of colour space such as HSV or RGB. In our experiments, the HSV colour histogram is computed for each over-segmented region, resulting in 72-dimensional feature vectors. The texture features are based on grey-level co-occurrence matrices and eight statistics, including mean and variance of energy, entropy, inertia and correlation, are used to describe a region. For the SIFT descriptor, first a visual word dictionary with 1000 entries is built according to the Bag of Words (BoW) model. Then a visual word histogram is constructed by mapping the descriptors to the dictionary.

Similarity between over-segmented regions is simply based on the Euclidian distances between corresponding feature vectors and the Gaussian similarity function with a fixed parameter of 0.6. It is set to the mean of the similarity values for the similarity threshold during the construction of the linking graph. And the area threshold is set to 0.05. For a few images that contain very small objects, it is adjusted to 0.01 instead. In addition, the constant factor $\varepsilon$ in Eq. (4) is empirically set to 0.85.

*3.4 Segmentation and Comparison Results*

*3.4.1 Results on Weizmann Segmentation Evaluation Database*

The proposed approach is compared with three state-of-the-art algorithms for this data set and F-measure and amount of fragmentation are calculated for quantitative comparisons.

1) Mean Shift [26]. The algorithm measures similarity in both spatial and range domains based on a computed attraction force field. Only intensity cues are used for segmentation. For the majority of the experiments, the parameters for mean shift, i.e., $h_s$, $h_r$ and minimum region size M, are set as 8, 7 and 1000 respectively. M shrinks to 500 for images containing very small objects. (Source code and precompiled binary are available at http://coewww.rutgers.edu/riul/research/code/EDISON).
2) Normalized Cuts (N-Cut) [17]. The segmentation problem is also formulated as graph partitioning and brightness values as well as spatial locations are used for calculation of edge weights. Note that N-Cut segmentation starts from pixels while our approach is based on over-segmented results. In the experiments, the number of segments for N-Cut is set to five. (Matlab implementation is available at http://www.cis.upenn.edu/~jshi/software).
3) Contour Detection and Hierarchical Image Segmentation (Gpb) [27]. After the contours have been detected, sequences of threshold values in the range from zero to one are tried for segmentation until the optimal results are met. (Matlab implementation is available at http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html).

F-measure and the amount of fragmentation are shown in Table 1 and segmentation results are illustrated in Fig. 3 and Fig. 4. By comparison, it can be seen that the proposed approach is superior to the others, with a better F-measure and amount of fragmentation. Our algorithm can segment out the salient object in its entirety, particularly for more textured and complex ones. N-Cut and mean shift are significantly outperformed, as indicated in Table 1, since only one type of feature is used in them, which cannot be adapted to a wide range of images. Note that the N-Cut segmentation results are displayed with only segmented region boundaries because the object is sometimes torn apart and covered by several segments. This mainly results from its tendency to partition the "graph" into more balanced clusters. Gpb is relatively more powerful but it suffers from over-segmenting due to strong intra-region variations, which is a common issue with contour-based approaches. Besides, weak boundaries can lead to over-merging and make it hard to determine the threshold for segmentation.

| Algorithm | Averaged F-measure Score | Average number of fragments |
|---|---|---|
| Our Method | 0.85 | 1.55 |
| Gpb | 0.74 | 2.27 |
| Mean Shift | 0.65 | 3.18 |
| N-Cut | 0.62 | 2.75 |

**Table 1.** Salient Objects Segment Coverage Test Results on the Weizmann Segmentation Evaluation Database

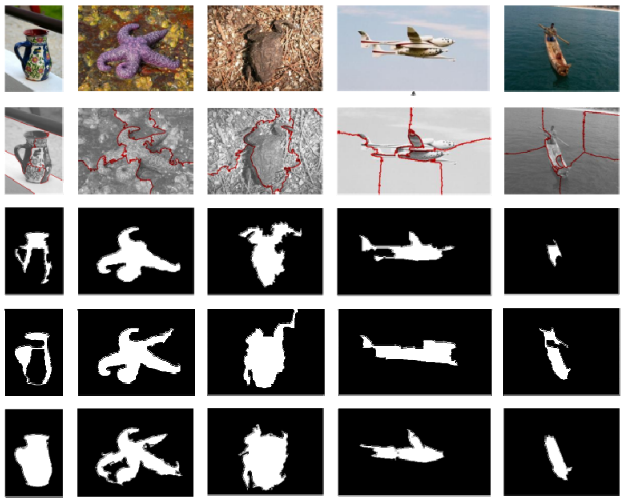

**Figure 3.** A sample of the results obtained by applying our algorithm to images compared to other algorithms. From top to bottom: Original images, N-Cut, mean shift, Gpb and our method

Unlike the algorithms analysed above, the proposed approach takes advantage of multiple types of features and then fuses the subsegmentation results by clustering over the constructed hypergraph. An object may be segmented into several pieces in one subsegmentation, but as long as these pieces are consistent in terms of at least some features, they can be merged in other subsegmentations and form a better result after final integration (see Section 3.4.3 for further discussions).

As stated in the previous section, because of the semantic gap problem, two regions with similar visual features may have different semantics and belong to different objects. In the meantime, two regions with different visual features can have the same semantics and belong to the same object. Take the image of a vase in Fig. 3 for example, many regions within the vase have completely different visual features, like colour, texture, SIFT or contour. Therefore, the methods of N-Cut, mean shift and Gpb can't deliver the whole vase in the final segmentation result. Our algorithm uses the PageRank scheme in each subsegmentation task. The merging of two regions is not simply based on the similarity of visual features. We used linking relationship and merging weight ($P_R$, defined in Eq. 4) instead, which measure more semantic similarity, so we can achieve better segmentation results in such cases.

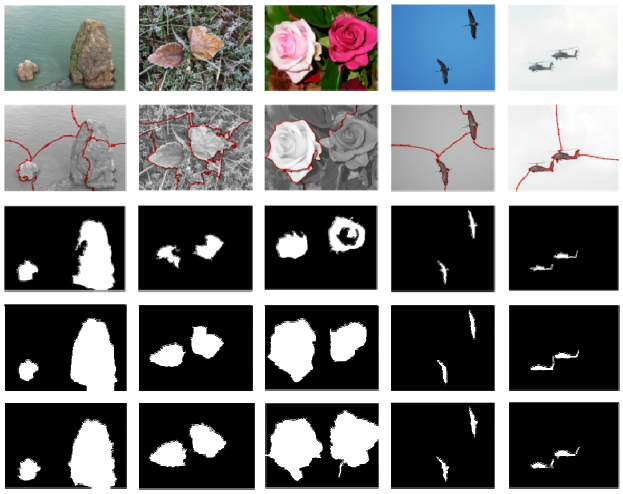

**Figure 4.** A sample of the results obtained by applying our algorithm to images compared to other algorithms. From top to bottom: Original images, N-Cut, mean shift, Gpb and our method

*3.4.2 Results on BSDS500*

Comparisons with Gpb are also conducted on BSDS500 [22], with some results presented in Fig. 5 and table 2.

Table 2 shows our algorithm is slightly better than Gpb. As shown in row 1-3 of Fig. 5, when the internal contour of salient objects is weak, the segmentation using Gpb is better than our algorithm on the completeness of the objects. Our algorithm uses N-cut in the beginning for over segmentation. At this stage, if some pieces of the objects are segmented into other objects and do not form a single region, it cannot be corrected in the final results. For example, in Fig. 5, the legs of the horse were in the same region as the grass after N-cut. The horse object will miss some legs in the final result of our algorithm. However, when the internal texture or contour of the salient object is very complex, the Gpb algorithm has the problem of over segmentation, as shown in row 4-8 of Fig. 5. In these cases, our algorithm can avoid this over segmentation problem and achieve better results, as explained in Section 3.3.2.

| Algorithm | Averaged F-measure Score | Average number of fragments |
|---|---|---|
| Our Method | 0.71 | 2.95 |
| Gpb | 0.66 | 3.97 |

**Table 2.** Salient Objects Segment Coverage Test Results on BSDS500

| Algorithm | Averaged F-measure Score | Average number of fragments |
|---|---|---|
| Our Method | 0.84 | 1.93 |
| Spatial- LTM | 0.65 | 3.34 |

**Table 3.** Salient Objects Segment Coverage Test Results on Weizmann Horse Database

**Figure 5.** A sample of the results obtained by applying our algorithm to images compared to Gpb. From left to right: Original images, Gpb and our method

*3.4.3 Results on Weizmann Horse Database*

Using the Weizmann Horse Database [24], the proposed approach is compared with Spatial-LTM [15], which combines multiple types of features based on a graphical model similar to LDA and enforces spatial coherency by sharing the same topic label within a region. As with the proposed approach, Spatial-LTM also starts from over-segmentation. Comparison results are presented in Fig. 6 and Table 3.

It can be clearly seen that our approach outperforms Spatial-LTM, even though it makes use of multiple types of features as well. Spatial-LTM estimates the topic label for each region by maximizing the likelihood, which is the product of all the factors corresponding to the features within this region. The problem is that for one region there is only a single appearance feature (average value of pixel colour and texture) but quite a few visual words (SIFT descriptors). Such an imbalance weakens the influences of the appearance feature and most of the contributions to the final results come from visual words. Consequently, the benefits from multiple types of features are significantly constrained.

By contrast, the proposed approach provides great flexibility for all the features employed and they can work independently and thus more effectively. It is the subsegmentation results instead of the features that are fused, elegantly settling the problem of imbalances between multiple types of features. Fig. 7 presents the three subsegmentation and the final results for three images, illustrating how different features are consolidated and jointly produce a better segmentation. It is not known beforehand which features are more suitable for a certain image, so we use all of them and then fuse all the results.

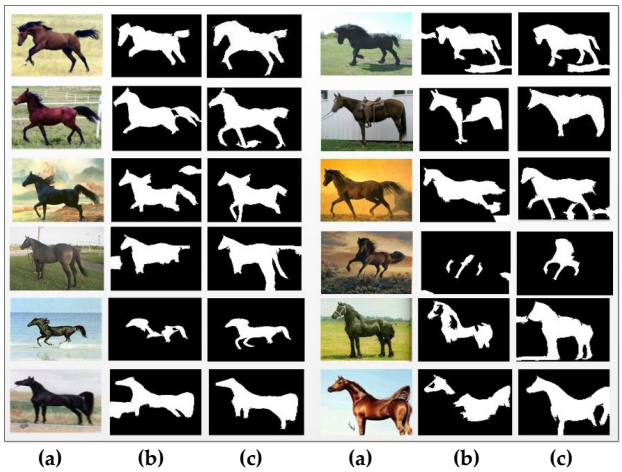


**Figure 6.** A sample of the results obtained by applying our algorithm to images compared to Spatial-LTM [15]. (a) Original image, (b) Spatial-LTM, (c) Our method

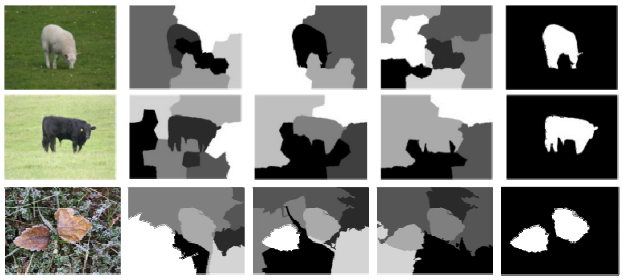

**Figure 7.** A sample of the results obtained by subsegmentation. From left to right: Original images, colour-spatial semantic-based subsegmentation, texture-spatial semantic-based subsegmentation, SIFT-spatial semantic-based subsegmentation and the final result for the cluster ensemble

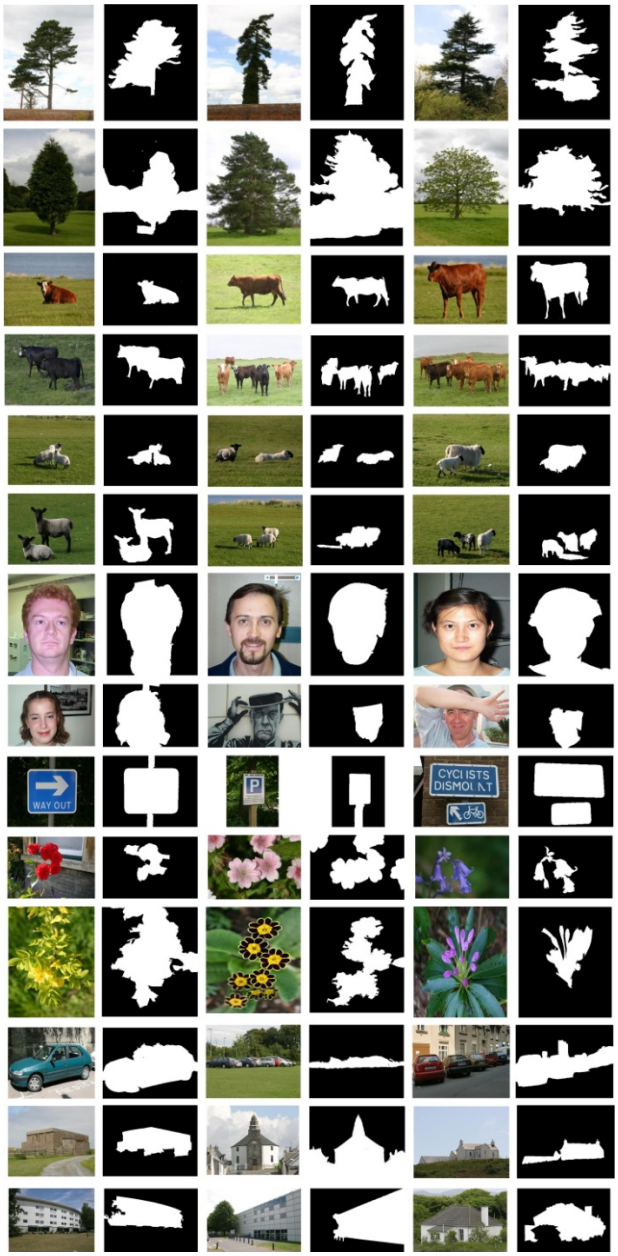


**Figure 8.** A sample of the results obtained by applying our algorithm to images on MSRC and Caltech-101

| Category of Images | Averaged F-measure Score of our Method | Averaged F-measure Score of Spatial-LTM |
|---|---|---|
| Tree | 0.83 | 0.70 |
| Cow | 0.82 | 0.62 |
| Sheep | 0.69 | 0.61 |
| Sign | 0.75 | 0.73 |
| Car | 0.69 | 0.61 |
| Face | 0.81 | 0.68 |
| Flower | 0.64 | 0.55 |
| Building | 0.67 | 0.47 |
| Average | 0.74 | 0.62 |

**Table 4.** Salient Objects Segment Coverage Test Results on MSRC and Caltech-101

*3.4.4 Results on MSRC and Caltech-101*

Comparisons with Spatial-LTM [15] are also conducted on MSRC [25] and Caltech-101 [28]. In the MRSC dataset, we selected 210 pictures from seven categories for comparison. In order to be different from previous experiments, the images selected in this experiment have more complex contours, like trees and flowers, etc. Some have multiple objects of different sizes, for example flower, sheep, cow and car. The others have more complex backgrounds, for example buildings and signs. From the Caltech-101 data set, we randomly selected 30 face images for this experiment.

The results for each category are shown in Table 4. Some of the experiment images are showed in Figure 8. In the results of this experiment, it can be clearly seen that the average F-measure score of our approach outperforms Spatial-LTM. In addition, we achieve a better performance than Spatial-LTM in every category we tested. The results demonstrate the effectiveness of our algorithm, which fuses several sub-segmentation results based on different features.

In the results of our approach, the best segmentation results are achieved in the categories of trees, cows and faces. Although the sheep images have similar background and contours to the cow images, the visual features of sheep heads and legs are very different from those of sheep bodies. Therefore, during the initial N-cut over-segmentation, the sheep heads and especially the sheep legs, stay in the regions containing large areas of grass. This impacts the segmentation performance for the category of sheep and leads to missing legs or heads in the final segmentation results. The results of this experiment also show that among all eight categories of images, our algorithm acts worst with flower images. We analysed this case and found two reasons. First, as shown in Figure 8, flower images can be very complex containing many flowers and the objects are quite small. In addition, because of the complicated background, it is challenging to segment the salient object from the background. Second, since our algorithm starts from the over-segmentation result of N-cut, some small objects will be missing in the final segmentation results as well, which will impact the overall segmentation performance. In summary, we achieve the object integrity in most of the images even for complex ones.

## 4. Conclusions

This paper proposes a novel image segmentation algorithm, which creatively leverages the cluster ensemble method to effectively fuse the segmentation results based on different visual features. Also, the idea of PageRank is exploited to incorporate the spatial information of regions, providing better semantic

similarity measures. Our algorithm is capable of adapting to various kinds of images since it has a more comprehensive view of images in multiple perspectives. The segmentation naturally benefits from those appropriate features with the effects of inappropriate ones suppressed. The spatial information integrated successfully addresses the problem of partitioning a complex object into multiple pieces and exhibits a better performance at preserving the object integrity.

Extensive experiments have been performed on four data sets with a large number and a wide diversity of images, and comprehensive comparisons have been made with the state-of-the-art approaches. The results demonstrate the effectiveness and superiority of our method and fusing the sub-segmentation results based on multiple features can produce more stable segmentations.

In the next stage, we will conduct more testing with more visual cues in more challenging situations and seek a different technique to produce better over-segmentations. We are also considering exploiting this algorithm in the research on automatic image annotation.

## 5. Acknowledgments

This research study was supported by the National Basic Research Program of China (973 Program) (2012CB821206), the National Natural Science Foundation of China (No. 91024001, No.61070142) and the Beijing Natural Science Foundation (No. 4111002), Chinese Universities Scientific Fund (No. 2013RC0306).

## 6. References

[1] J. N. Kapur, P. K. Sahoo, A. K. C. Wong (1985). A new method for gray-level picture thresholding using the entropy of the histogram. Computer Vision, Graphics, and Image Processing, 29(3): 273-285.

[2] J. Malik, S. Belongie, T. Leung, J. Shi (2001). Contour and texture analysis for image segmentation. International Journal of Computer Vision, 43(1): 7-27.

[3] T. Pavlidis, Y. T. Liow (1990). Integrating region growing and edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(3): 225-233.

[4] Y. Deng, B. Manjunath (2001). Unsupervised segmentation of color-texture regions in images and video. IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(8): 800-810.

[5] Y. Boykov, G. Funka-Lea (2006). Graph cuts and efficient N-D image segmentation. International Journal of Computer Vision, 70(2): 109-131.

[6] T. D. Pham (2001). Image segmentation using probabilistic fuzzy c-means clustering. Proceedings of 2011 International Conference on Image Processing, 1: 722-725.

[7] X. Zhang, L. Jiao, F. Liu, L. Bo, M. Gong (2008). Spectral clustering ensemble applied to SAR image segmentation. IEEE Transactions on Geoscience and Remote Sensing, 46(7): 2126-2136.

[8] J. Fan, D. K. Y. Yau, A. K. Elmagarmid, W. G. Aref (2001). Automatic image segmentation by integrating color-edge extraction and seeded region growing. IEEE Transactions on Image Processing, 10(10): 1454-1466.

[9] S. Belongie, C. Carson, H. Greenspan, J. Malik (1998). Color- and texture-based image segmentation using EM and its application to content-based image retrieval, Proceedings of 6th International Conference on Computer Vision, 675-682.

[10] K. Barnard, P. Duygulu, D. Forsyth, N. de Freitas, D. M. Blei, M. I. Jordan (2003). Matching words and pictures. The Journal of Machine Learning Research, 3: 1107-1135.

[11] T. Malisiewicz, A. A. Efros (2008). Recognition by association via learning per-exemplar distances. Proceeding of IEEE Conference on Computer Vision and Pattern Recognition, 1-8.

[12] L. Fei-Fei, P. Perona (2005). A Bayesian hierarchical model for learning natural scene categories. Proceeding of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2: 524-531.

[13] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, T. Tuytelaars, L. Van Gool (2005). Modeling scenes with local descriptors and latent aspects. Proceedings of 10th IEEE International Conference on Computer Vision, 1: 883-890.

[14] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, C. Bray (2004). Visual categorization with bags of keypoints. Workshop on Statistical Learning in Computer Vision, 1-22.

[15] L. Cao, L. Fei-Fei (2007). Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes, Proceedings of IEEE 11th International Conference on Computer Vision, 1-8.

[16] F. Perronnin, C. Dance, G. Csurka, M. Bressan (2006). Adapted vocabularies for generic visual categorization. Proceedings of 9th European Conference on Computer Vision, 3954: 464-475.

[17] J. Shi, J. Malik (2000). Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8): 888-905.

[18] L. Page, S. Brin, R. Motwani, T. Winograd (1999). The PageRank citation ranking: Bringing order to the web. Stanford InfoLab.

[19] X. Ren, C. Fowlkes, C. Malik (2005). Scale-invariant contour completion using conditional random fields. Proceeding of 10th IEEE International Conference on Computer Vision, 1214-1221.

[20] P. Felzenszwalb, D. Huttenlocher (2004). Efficient graph-based image segmentation. International Journal of Computer Vision 59(2): 167-181

[21] A. Strehl, J. Ghosh (2003). Cluster ensembles - a knowledge reuse framework for combining multiple partitions. The Journal of Machine Learning Research, 3: 583-617.

[22] D. Martin, C. Fowlkes, D. Tal, J. Malik (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. Proceeding of 8th International Conference on Computer Vision, 416-423.

[23] S. Alpert, M. Galun, A. Brandt, R. Basri (2012). Image segmentation by probabilistic bottom-up aggregation and cue integration. IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(2): 315-327.

[24] E. Borenstein, S. Ullman. Learning to segment. Proceedings of 8th European Conference on Computer Vision, 3023 : 315-328.

[25] J. Winn, A. Criminisi, T. Minka (2005). Object categorization by learned universal visual dictionary. Proceedings of 10th IEEE International Conference on Computer Vision, 2: 1800-1807.

[26] D. Comaniciu, P. Meer (2002). Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(5): 603-619.

[27] P. Arbelaez, M. Maire, C. Fowlkes, J. Malik (2011). Contour detection and hierarchical image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(5): 898-916.

[28] L. Fei-Fei, P. Perona (2006). One-shot learning of object categories. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(4): 594-611.