Journal of Data Analysis and Information Processing, 2014, 2, 117-136
Published Online November 2014 in SciRes. http://www.scirp.org/journal/jdaip
http://dx.doi.org/10.4236/jdaip.2014.24014

Hierarchical Image Segmentation Using a Combined Geometrical and Feature Based Approach

Melissa Cote, Parvaneh Saeedi

School of Engineering Science, Simon Fraser University, Burnaby, Canada
Email: melissa_cote@sfu.ca, psaeedi@sfu.ca

Received 25 August 2014; revised 30 October 2014; accepted 20 November 2014

Abstract

This paper presents a fully automatic segmentation algorithm based on geometrical and local attributes of color images. This method incorporates a hierarchical assessment scheme into any general segmentation algorithm for which the segmentation sensitivity can be changed through parameters. The parameters are varied to create different segmentation levels in the hierarchy. The algorithm examines the consistency of segments based on local features and their relationships with each other, and selects segments at different levels to generate a final segmentation. This adaptive parameter variation scheme provides an automatic way to set segmentation sensitivity parameters locally according to each region's characteristics instead of the entire image. The algorithm does not require any training dataset. The geometrical attributes can be defined by a shape prior for specific applications, *i.e.* targeting objects of interest, or by one or more general constraint(s) such as boundaries between regions for non-specific applications. Using mean shift as the general segmentation algorithm, we show that our hierarchical approach generates segments that satisfy geometrical properties while conforming with local properties. In the case of using a shape prior, the algorithm can cope with partial occlusions. Evaluation is carried out on the Berkeley Segmentation Dataset and Benchmark (BSDS300) (general natural images) and on geo-spatial images (with specific shapes of interest). The F-measure for our proposed algorithm, *i.e.* the harmonic mean between precision and recall rates, is 64.2% on BSDS300, outperforming the same segmentation algorithm in its standard non-hierarchical variant.

Keywords

Image Segmentation, Adaptive Color Analysis, Shape Analysis, Prior Model, Image Processing, Split-and-Merge Segmentation, Perceptual Grouping

How to cite this paper: Cote, M. and Saeedi, P. (2014) Hierarchical Image Segmentation Using a Combined Geometrical and Feature Based Approach. *Journal of Data Analysis and Information Processing*, **2**, 117-136.
http://dx.doi.org/10.4236/jdaip.2014.24014

1. Introduction

Image segmentation is one of the most commonly used processes in applications of computer vision. It is also one of the most challenging problems and has a wide range of computer vision and machine learning applications in areas such as geo-spatial, biomedical, security, surveillance, and inspection. These applications generally require matching, correspondence establishing, and identification of image regions. Segmentation algorithms that deliver robust and consistent results under different conditions such as scene and viewpoint variations, shading, specularities and illumination change have key roles in the success of their machine vision applications. The task of segmentation is often complicated by various factors such as unclear boundaries, color/intensity similarities of adjacent structures, and variations in shape or other characteristics of image segments. In the past few decades numerous algorithms have been developed to address the need for robust and reliable segmentations. Due to issues such as complexity and diversity of data and image sources, the quest for a good segmentation algorithm that performs equally well for different images and applications is somehow still not fully satisfied.

This work presents a computational approach to the problem of image segmentation. Our proposed method can be broadly used in applications for indoor and outdoor scenes as well as for regions with specific shapes. The paper is structured in the following way. Related work regarding image segmentation is reviewed next, followed by the contributions of the proposed approach. Section 2 details the methodology, while Section 3 describes how to modify it to accommodate shape-specific segmentation. Experimental results on varied imageries (state-of-the-art color dataset and benchmark, geo-spatial), as well as their qualitative and quantitative evaluations are provided in Section 4. Finally, Section 5 presents conclusions and future works.

1.1. Related Work

1.1.1. Feature Based Segmentation

Early works in image segmentation were based on featural and low level information such as intensity value, color, texture, edge, etc. Feature based methods cluster image pixels into homogeneous regions that have high probability of arising from a similar physical property. Chen *et al.* [1] proposed a color based image segmentation algorithm using contrast information. They utilized four directional operators to extract directional color contrast information. A single-threshold scheme was used to detect directional boundaries. These directional boundaries were merged together via a verification process to form 2-D boundaries of segments. Felzenszwalb and Huttenlocher [2] utilized the intensity differences across the boundary of regions and the intensity differences between neighboring pixels within a region to make a graph based representation of an image which was used in the segmentation process by finding the minimal spanning tree of the graph. In more recent work, Lo *et al.* [3] introduced a texture-based segmentation algorithm that performed feature extraction using the Dual-Tree Complex Wavelet Transform (DT-CWT). They first transferred all image pixels into a spatial-texture space and then used a clustering method to cluster pixel vectors (created for all pixels). Tan and Isa [4] used histograms for image segmentation. They presented a novel histogram thresholding algorithm also known as fuzzy C-means hybrid (HTFCM). Their method could obtain regions with uniform colors.

Feature based methods are simple and intuitive to implement. However, most of them require tuning of several control parameters, and generally the performance of these methods is subjected to proper setting of such parameters. Also, since these methods rely on texture, region homogeneity, intensity, etc. their performance could be highly affected by noise, illumination condition, variation in the view point and sometimes specific properties of objects.

1.1.2. Model Based Segmentation

In recent years, model based image segmentation algorithms have attracted more attention. Model based image segmentation aims to recover predefined shapes or structures from an image using shape properties and constraints. States of the art on model based segmentation are mostly based on active shape and appearance models, active contours and deformable templates, and level set methods.

Active shape and appearance models incorporate statistical or parametric models of shapes (using training data) in the segmentation process. Cootes *et al.* [5] described point distribution models based on Principal Component Analysis (PCA) to parameterize variations of active contours in a training dataset. They exploited a linear formulation of the shape model (active shape model) to iteratively search for the shape in an image. Staib and Duncan [6] proposed a parametric model based on an elliptic Fourier decomposition of the shape boundaries. The segmentation problem was addressed as an optimization problem that found the best match between the model and the image gradient. Wang and Staib [7] used shape parameters derived from statistical variations of object boundary points in a training set to model an object. A Bayesian formulation, based on this model and the gradient of the image, was employed to identify the object's boundaries in the image.

In general, active shape model based approaches use an off-line learning process. Processing the training dataset is commonly performed via manual segmentation and the shape model cannot be extended as new segmentation results become available [8]. Perhaps the main disadvantage associated with these methods is the lack of robustness in the recognition and training phases. For instance, often there is a poor performance under partial occlusion and imperfect training datasets cannot be utilized.

Combining prior shape information with active contours and deformable templates is also used in model based segmentation. In these approaches, the active contour's energy function is modified such that it incorporates a representation of the reference shape within it. Therefore, the contour converges to the boundaries that are more similar to the prior shape boundary. Tsai *et al.* [9] [10] proposed a model based curve evolution technique using a parametric model of the segmenting curve by applying principal analysis component to a collection of signed distance representations of the training data. Foulonneau *et al.* [11] presented a new way of constraining the evolution of a region based active contour with respect to a reference shape. Minimizing a shape prior was achieved through a distance between shape descriptors based on the Legendre moments of the characteristic function. Using the level set method for finding specific shapes in an image has also been reported by many researchers. Chen *et al.* [12] modified the energy function so that it depends on the image gradient and the prior shape. In this approach the level set formulation could find boundaries that were similar in shape to the prior, even under partial occlusions. Leventon *et al.* [13] incorporated prior shape information into the geodesic active contour method by introducing model representation of deformable shapes and probability distribution over the variances of a set of training shapes. The segmentation process embedded an initial curve as the zero level set of a higher dimensional surface, and evolved the surface such that the zero level set converged on the boundaries of the object to be segmented. The main advantage of the Leventon method was on intrinsically representing active contours by the level set function. This feature made it possible to construct a parameterization free shape model. Rousson and Paragios [14] proposed a level set based approach for shape-driven object extraction using a voxel-wise probabilistic level set formulation. Constraints on the segmentation process were imposed by seeking a projection of the prior model onto the image plane using a similarity transformation. Riklin-Raviv *et al.* [15] introduced a variational approach for model based segmentation, using a single reference object. They proposed a shape-similarity measure by generalizing the Chan-Vese [16] level set framework and embedding the projective homography between the prior shape and the image to segment.

The advantage of methods based on active contours and level sets is that they can handle topological changes of the contour in a natural way. For instance, the contour may split and/or merge during evolution without any special considerations. For shape modeling, this property allows constructing shape similarity measures that can handle variant shape topologies. But like many other level set based methods, these methods can suffer from numerical instabilities and sensitivity to noise. They are also prone to getting caught in local minima.

1.1.3. Hierarchical Segmentation

Recently, many hierarchical approaches have been proposed for image segmentation purposes. Sharon *et al.* [17] presented a fast segmentation by a weighted aggregation (SWA) method that determined salient regions of an image and built them into a hierarchical structure. Their algorithm consisted of an adaptive process in which pixels were recursively aggregated into increasingly large-scale aggregates of coherent properties: intensity, texture and boundary integrity. Paris and Durand [18] presented a hierarchical segmentation approach using mean shift segments [19] for segmenting color images and videos. They used Mores theory to topologically decompose mean shift segments into density modes. Their method is fast and requires no training. They reported an F-measure, *i.e.* the harmonic mean between precision and recall rates, of 0.61 on average on the Berkeley Segmentation Dataset and Benchmark (BSDS300). Alpert *et al.* [20] presented a probabilistic aggregation based method that utilized local intensity, texture of regions along with the length of common boundaries between neighboring regions to segment grayscale images. In their method they merged pixels according to the above cues to generate larger regions as the aggregation advances. They trained their system parameters using a large number of patches and hold those parameters unchanged for the entire tested image subjects. To verify their algorithm they had chosen a set of images with clear distinction between the object in the foreground from background. They compared their results against manually created ground truth and showed an F-measure of 0.87. Such high F-measure was due to the fact that the image test subjects were handpicked with the above criteria making comparison of this method with other methods, where the input data is not biased, impossible. Corso *et al.* [21] introduced an image labeling method based on a graph-shift algorithm (an energy minimization algorithm) for natural image labeling. They decomposed the image into multiple layers and adaptively manipulated (shifted) the parent-child relationship. They efficiently computed all possible shifts (which lead to the change of energy) for all of the nodes in the image graph and assessed the energy function. Using the energy value, they coarsened the image adaptively. While they showed some visual labeling results, they have presented no F-measure for their method.

Here, we propose a fully automatic, non-training-based segmentation method for color images based on geometrical and local features. First, we create a hierarchical composition of all potential segments of the image using a multi-layer approach in which segments can partially or fully overlap with other segments at different layers. In this approach, a general segmentation method (such as mean shift [19], graph-based [2], etc.) is utilized. The segmentation algorithm is run using a range of parameter settings that leads to results from under to over segmentation outcomes. Then, starting both from the finest level (most over-segmented layer) and the coarsest level (most under-segmented layer), the algorithm examines various local properties of the segments and compare those properties between parent and children segments along the hierarchical direction. These properties include color, texture and distinctiveness of the boundaries. The proposed algorithm is a general approach; however we show that with the addition of complementary shape constraints, it can be used to bias the detection of objects with specific shapes.

1.2. Contributions

The main contributions of this paper are as following:

1) A new adaptive parameter manipulation that in general can be incorporated with any algorithm where the quality of results on different input images is dependent on adjusting some control parameters. The adaptive parameter variation scheme provides an automatic way to set control parameters locally according to each sub-region's quality instead of the entire image.

2) A novel segmentation algorithm based on local image features and geometrical properties of regions. Through a hierarchical approach, mean shift segmentation is applied on the image several times using different settings that drive segmentation results from under to over segmentation. By inspecting the relationship between parents-children in two directions of bottom-up and top-down, segments conforming best with local and geometrical constraints are chosen from different levels of the hierarchy.

2. Methodology

This section describes the proposed approach to image segmentation. General segmentation problems are first illustrated, followed by the description of the proposed hierarchical feature-based segmentation method, divided into several processes: general hierarchical segmentation, definition of region comparison constraints, segment tree creation, search and identification of segments, and segment post-processing. Shape-specific geometrical aspects are presented in Section 3.

2.1. General Segmentation Problems

The performance of image segmentation algorithms often depends on various elements such as the skill of the user (if manual or semi-automatic), the compatibility of the method with the nature of the application, and the image quality. There is no algorithm that has established itself as the definitive solution to the image segmentation problem.

Often segmentation algorithms are parameter dependent which adds to the applicability of such algorithms. Examples of such cases can be seen for mean shift segmentation algorithm [19], fuzzy c-means clustering [22], and Gaussian mixture model [23]. For instance, fuzzy c-means clustering requires that three parameters (number of iterations, termination threshold, and number of clusters) be set according to the application. As another example, the quality of segmentation in mean shift is controlled by three parameters: the spatial resolution h_s, the range (or color) resolution h_c, and the minimum segment size M. These parameters are usually set manually and often a single setting will not be sufficient for a complete segmentation. **Figure 1** depicts an example of a segmentation process using the mean shift algorithm where h_c was set to two different values. In these two examples, a variation in h_c has caused the segmentation results to be improved for some regions but deteriorated for others. It is a challenge (if ever possible) to find one value for h_c to segment all building regions correctly, which is why we propose a local adaptive parameter manipulation scheme.

2.2. Proposed Hierarchical Feature-Based Segmentation

In this paper we propose a segmentation algorithm for segmenting color images based on distinctiveness in color and region boundaries. The algorithm includes four main processes as depicted in **Figure 2**. Details of each process along with quality assessment constraints are described in this section.

2.2.1. General Hierarchical Segmentation

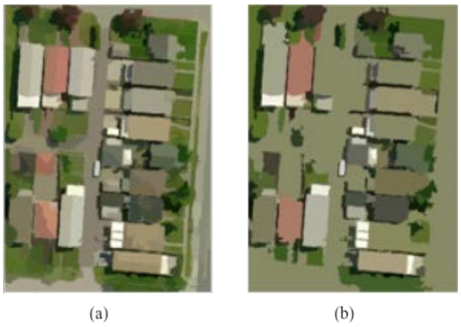
In this section, we define a general hierarchical segmentation process to generate a large number of potential segmentation outcomes. Here, any common segmentation algorithm could be utilized as long as the quality control parameters are identified and changed in between the layers. The results of these segmentations must cover all potential outcomes from under to over segmentation.

In this work, we have used the mean shift segmentation method [19]. The idea behind the mean shift algorithm is, for a set of discrete data, to locate the stationary points of the underlying density function, allowing the detection of the modes of the density. For an image segmentation application, the input space is defined by a joint domain (location and color of pixels), with each pixel iteratively associated with a significant mode of the density, located in its neighborhood, thus delineating clusters in the joint domain. An image region (segment) is therefore defined by all the pixels associated with the same mode in the joint domain. From our experiments, the main control parameter for this method is the range resolution (h_c). The input image (I) is segmented multiple times for the range resolutions of:

$$h_c = \left(T_l, T_l + \Delta T, \cdots, T_l + (K-2)\Delta T, T_h\right)$$

For each value of $i = 1, \cdots, K$ where $K = \left(\left(T_h - T_l + 1\right)/T\right)$, a segmented image S_i is generated that includes regions R_{ij}:

$$S_i = \cup R_{ij} \quad \text{and} \quad j = 1, \cdots, N_i \tag{1}$$


(a)

(b)

Figure 1. Mean shift algorithm on aerial imagery. (a) Low range resolution (h_c) value; (b) High range resolution (h_c) value.

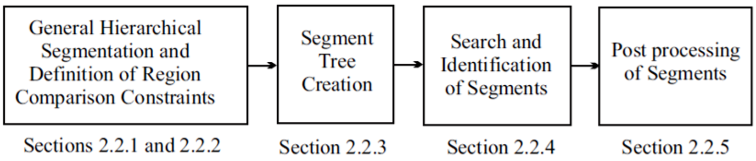

General Hierarchical Segmentation and Definition of Region Comparison Constraints
Segment Tree Creation
Search and Identification of Segments
Post processing of Segments
Sections 2.2.1 and 2.2.2
Section 2.2.3
Section 2.2.4
Section 2.2.5

Figure 2. Algorithm's main components.

Here N_i is the number of regions in S_i. Parameters T_l and T_h have a large difference in value so that all possible regions in the image are covered. The value of T_h should be set in a way that segmentation results at the highest level (S_K) include only coarse segments. Parameter T controls the number of segmentation layers and it affects the precision, computational complexity, as well as quality of the segmentation. The spatial resolution h_s and the minimum segment size M for mean shift segmentation are adjusted according to the application and potential dimensions of the objective regions. For instance for the application of rooftop detection (for aerial images) M is set to 200 and h_s is set to 3. They also could be estimated automatically in a similar manner to h_c by varying them to obtain additional segmentation layers but this is out of the scope of this paper. An additional segmented image S_{K+1} is also required by segmenting the entire image into one single region.

This hierarchical segmentation scheme generates a large number of segments based on a range of various parameter settings. Clearly the collections of these segments include redundant, overlapping, divided and merged segments and only a few of these segments will conform to valid/intended segments. The tendency of each segmented region for splitting or merging based on the color range (in the consecutive layers) are measured using two constraints that are explained next.

2.2.2. Region Comparison Constraints

For non-specific applications (generic segmentation), the proposed algorithm is based on two sets of constraints that measure the quality of potential segments. When segmenting an image based on its pixels' colors, the color resolution can affect the sensitivity of the segmentation results. Therefore, since the hierarchical multi-layer segmentation generates region components by varying the range (color) resolution, the neighboring segments in one level might be merged together in the next level, if their color values fall into one color range interval. A similar reasoning can be done if using another segmentation algorithm.

The color constraint in this section implies selecting regions according to their color stability across hierarchical segmentation layers. Here we define the color stability of a region by its reluctantness in splitting into two or more regions or merging with its neighboring regions. Therefore, we are after the most distinctive regions across all hierarchical segmentation levels. In the hierarchical segmentation process, the highest level is the least sensitive layer (includes the lowest number of segments) and the lowest level (with minimum range resolution value) is the most sensitive layer (includes the highest number of segments). Here we introduce two constraints: *Dividability* and *Distinctiveness*.

1) Dividability: Dividability indicates the tendency of a region to break into two or more parts color-wise. It is a reflection of the entropy computed from the distribution of the regions' colors. Many color based segmentation approaches use a color reduction process in which the amount of the information presented by color is reduced drastically. An example of a reduction process is the color quantization in which only a certain number of colors are used to represent an image. The association of each color with the representative quantized color is usually based on the minimum distance constraint. The HSV color model has been used in many applications. Since the hue component of this model holds the color information, this model is very useful for processing color images. Unfortunately, the hue component is unstable near the chromatic axis. That is, with low saturation, the error within hue measurements is high. As saturation is a radial dimension, it has a linear relationship with the certainty in the hue measurement. Also, when it comes to natural scenes, often regions that are identified by a human as being similar in color indeed have different hue values. For the two above reasons we created a color reduction scheme in which 219 colors are grouped into 22 groups. Each group includes similar colors that cover various shades and tones of that color. The similarity of each pixel is measured with all individual colors of each group. For each pixel, both the reduced color value (the most similar color using Euclidean distance) as well as its group will be maintained. The concept of grouping colors allows us to correctly associate together those neighboring regions that include various shades of the same color usually due to illumination or reflection variation. **Figure 3** represents five groups of the above 22 groups of colors.

The dividability of a region is high when its tendency to break into different colors is high, which is reflected by high entropy. The dividability measure $D(R_{ij})$ of a region R_{ij} is defined by:

$$D(R_{ij}) = -\sum_{\forall t, CG_t \cap R_{ij} \neq \varnothing} \left(P(R_{ij}, CG_t) \times \log\left(P(R_{ij}, CG_t) \right) \right) \quad (2)$$

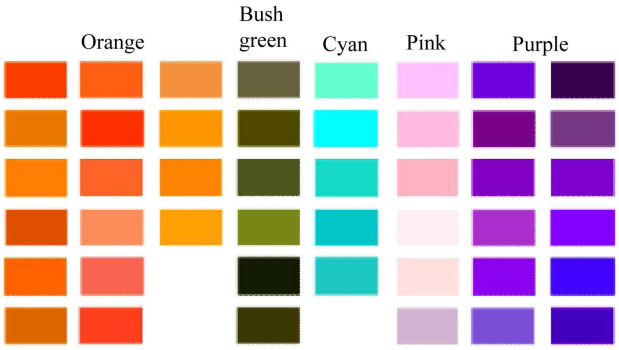


Figure 3. Five representative groups of colors along with their colors.

$$P\left(R_{ij}, CG_t\right) = \frac{\text{Area}\left(R_{ij} \cap CG_t\right)}{\text{Area}\left(R_{ij}\right)} \tag{3}$$

Here, CG_t represents all the pixels on the image that fall into the color group t. A region composed of only one color group will have a minimal dividability of 0, whereas a region composed of all the color groups in equal parts will have a maximal dividability. A region with low dividability is more stable thus more desirable.

2) *Distinctiveness*: The distinctiveness of a region is measured according to the quality of its boundary points. Distinctive regions tend to have clear and solid boundaries. For this, the input image is first edge detected (Canny) and then the edges are linked together, as the Canny edge detector rarely produces closed boundaries. The linking is achieved by analyzing pixels in a 9 × 9 pixel neighborhood at each edge point. For each edge point two measures are calculated:

- Magnitude of the gradient vector $\left(\left|\nabla I(x,y)\right|\right)$, and
- Direction of the gradient vector $\left(\alpha(x,y)\right)$.

Using the two above measures, points with similarities in both magnitude and direction will be added to the edge point and the algorithm progresses until there is no point in the neighborhood that can be added to the current edge point. The similarity constraints are defined by:

a) An edge pixel with coordinates $\left(x_0, y_0\right)$ is similar in magnitude to point (x, y) in its neighborhood if: $\left|\nabla I(x,y) - \nabla I\left(x_0, y_0\right)\right| < 10$.

b) An edge pixel with coordinates $\left(x_0, y_0\right)$ is similar in direction to point (x, y) in its neighborhood if: $\left|\alpha(x,y) - \alpha\left(x_0, y_0\right)\right| < 45^\circ$.

Once all edge points are linked, a minimum length constraint of 20 pixels is applied and any edge with a length smaller than that will be removed from the edge map. The final edge map is referred by E_M. For each region of every layer the following measure is computed:

$$E_D\left(R_{ij}\right) = \frac{E_M \cap \text{Boundary}\left(R_{ij}\right)}{\text{Boundary}\left(R_{ij}\right)} \times \frac{\text{Area}\left(R_{ij}\right)}{\text{Area}(I)} \tag{4}$$

where $\text{Area}\left(R_{ij}\right) = \int\int_{x,y \in R_{ij}} \mathrm{d}x\mathrm{d}y$, $\text{Area}(I)$ represents the image area, and $\text{Boundary}\left(R_{ij}\right)$ includes all the boundary pixels of region R_{ij}. The second fraction is designed to give priority to larger segments with similar boundary edge coverages. The two above constraints are equally weighted when examining a segment region. These constraints are used to assess the different hierarchical segmentation levels (Section 2.2.4).

2.2.3. Segment Tree Creation

Tree $\langle V, E, Root\rangle$ is defined from regions in segmented images as follows:

$$V = \left\{ R_{ij} \middle| \forall R_{ij} \in S_i \right\} \quad \text{for} \quad i = 1, \cdots, K+1 \quad \text{and} \quad j = 1, \cdots, N_i \tag{5}$$

$$E = \left\{ \left(R_{ij}, R_{i+1,x} \right) \middle| \forall R_{ij} \in S_i,\ \exists R_{i+1,x} \in S_{i+1,x} = \underbrace{\max}_{u} \left(R_{ij} \cap R_{i+1,u} \right) \right\}$$
$$\text{for } i = 1, \cdots, K+1 \quad \text{and} \quad j = 1, \cdots, N_i \text{ and } Root = R_{K+1,1} \tag{6}$$

In this tree, a node (region) R_{ij} at level i is the child of node $R_{i+1,x}$ at level $i+1$, only if $R_{i+1,x}$ has the highest area overlap (among all neighboring nodes) with R_{ij}. **Figure 4(c)** shows a visual representation of the segment tree with four levels (K to $K-3$) plus the root $(K+1)$.

2.2.4. Tree Search (Segment Selection)

To extract all regions that best conform to the geometrical and color constraints, an iterative algorithm is designed. Two different searches are performed: a bottom-up search and a top-down search.

The bottom-up search starts at the lowest level of the tree (leaves of the tree) and goes up iteratively, always assessing two consecutive levels against each other (**Figure 4(b)**). The assessment is invoked by comparing each child with its parent. If a child is found to be better than its parent, the selected child is added to the set R_{BU} and the area it covers is removed from the higher level for the next iteration. Function RS is defined by both color and geometrical properties through:

$$RS\left(R_{ij}, R_{i+1,x}\right) = \begin{cases} \text{True}, & \left(D\left(R_{ij}\right) < D\left(R_{i+1,x}\right)\right) \text{ and } \left(E_D\left(R_{ij}\right) > E_D\left(R_{i+1,x}\right)\right) \\ \text{False}, & \text{otherwise} \end{cases} \tag{7}$$

A child R_{ij} is better than its parent $R_{i+1,x}$ $(RS = \text{true})$ if its dividability is lower and its edge distinctiveness is higher. In other words, a child is better if it is less prone to be divided into several regions of different colors than its parent (more stability) and if its boundaries conform better to discontinuities in the image than its parent's. The operators < and > are defined in a way that if the difference for one of the conditions is minimal between the child and its parent (for instance less than 5%), then it is too close to call a winner and that condition is discarded from the assessment.

In **Figure 4**, the lowest level (S_{K-3}) comprises seven children, which are all compared against their respective parent on the level above (S_{K-2}). Once all children have been assessed, the areas covered by the three selected children (shown with star pattern in **Figure 4(b)**) are subtracted from the parent level (shown as dark regions in **Figure 4(b)**), and the parent segments are updated and relabeled (S'_{K-2}) to reflect the most recent changes. At the next iteration, the children come from the updated segments (S'_{K-2}) and the parents from the original segmentation (S_{K-1}). Parents are always derived from the original mean shift segments, whereas children are always derived from the updated mean shift segments except at the leaf level $(S'_{K-3} = S_{K-3})$. The search is stopped when the root level is reached $(K+1)$ or when the selected segments cover the entire image. The union of all segments in the set R_{BU} creates the segmented version of the input image (**Figure 4(d)**) for this direction of traversal.

The top-down approach is similar to the bottom-up one in the sense that two consecutive levels are assessed against each other, comparing children and parent. However it differs twofold, in the order in which the levels are visited (from the root of the tree going down to the leaves), and in the segments that are selected to put in the set R_{TD} (coming from the parent level instead of the child level). **Figure 4(c)** illustrates the top-down search scheme using the same mean shift segments as for the bottom-up (**Figure 4(a)**). The parents are compared against all their children using RS and are selected if better than their children. If a parent has several children and some of them are assessed as better and some as worse, every child that is better $(RS = \text{true})$ is removed from the parent and the parent's features are recomputed considering its new boundaries. For instance, in **Figure 4(c)**, when assessing the parent level S'_{K-1} against the children level S_{K-2}, the parent $R'_{K-1,3}$ is compared against $R_{K-2,3}$ and $R_{K-2,4}$. Because the child $R_{K-2,4}$ is found as better, it is removed from the parent $R'_{K-1,3}$ and the parent is updated to $R''_{k-1,3}$ (shown as hatched in **Figure 4(c)**). The children that were found as worse are reassessed against the updated parent ($R_{K-2,3}$ against $R''_{k-1,3}$), and removed from the parent if they are found as better. These evaluation/update steps are carried out until no more children are found as better than the parent. After all parents have been assessed on a given level, the areas of their updated versions (only the parts that were better than all their children) are selected (shown as regions with a star pattern in **Figure 4(c)**), put in

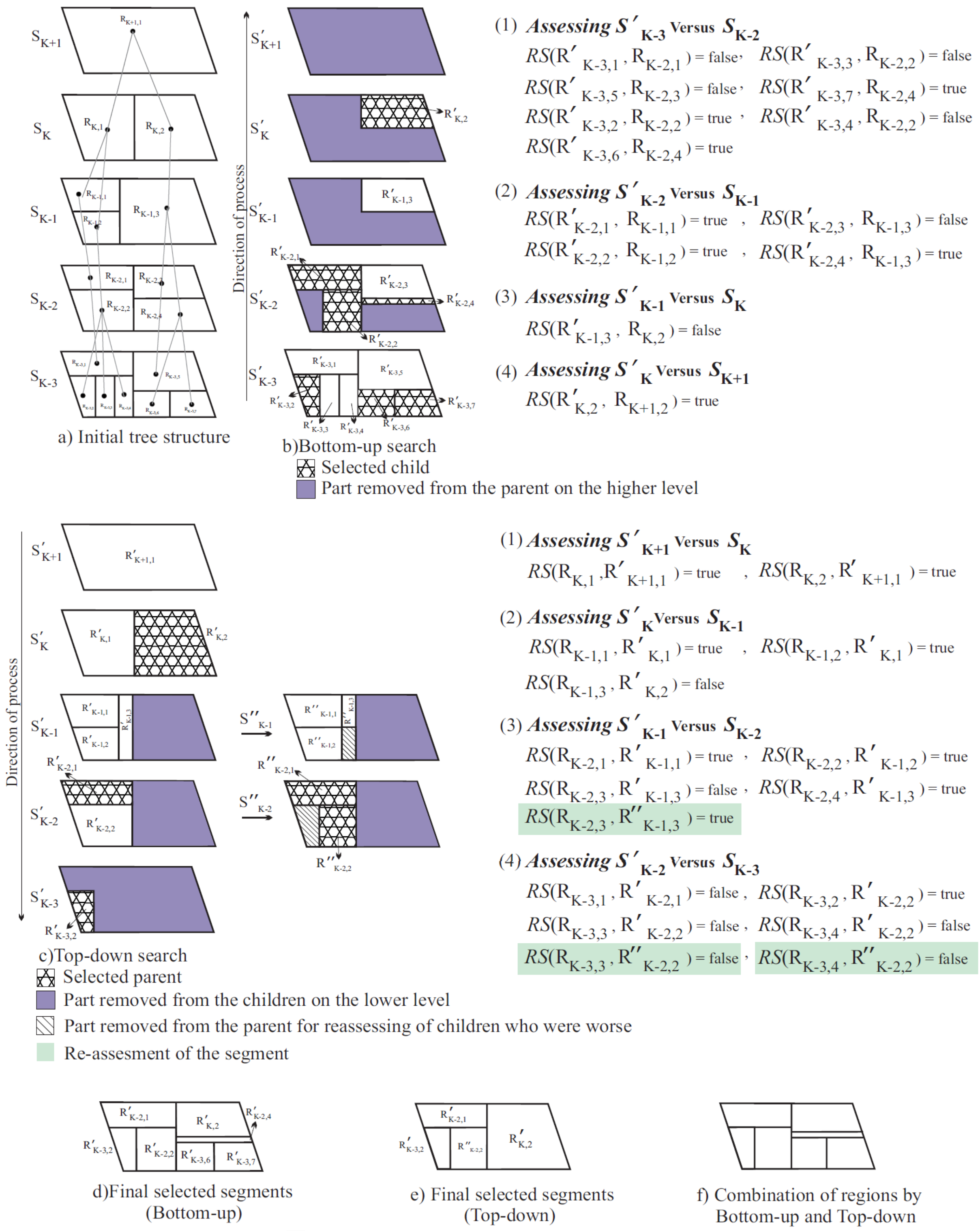

a) Initial tree structure
b)Bottom-up search
Direction of process
▩ Selected child
▆ Part removed from the parent on the higher level
(1) Assessing S'_{K-3} Versus S_{K-2}
$RS(R'_{K-3,1}, R_{K-2,1}) = false$, $RS(R'_{K-3,3}, R_{K-2,2}) = false$
$RS(R'_{K-3,5}, R_{K-2,3}) = false$, $RS(R'_{K-3,7}, R_{K-2,4}) = true$
$RS(R'_{K-3,2}, R_{K-2,2}) = true$, $RS(R'_{K-3,4}, R_{K-2,2}) = false$
$RS(R'_{K-3,6}, R_{K-2,4}) = true$
(2) Assessing S'_{K-2} Versus S_{K-1}
$RS(R'_{K-2,1}, R_{K-1,1}) = true$, $RS(R'_{K-2,3}, R_{K-1,3}) = false$
$RS(R'_{K-2,2}, R_{K-1,2}) = true$, $RS(R'_{K-2,4}, R_{K-1,3}) = true$
(3) Assessing S'_{K-1} Versus S_K
$RS(R'_{K-1,3}, R_{K,2}) = false$
(4) Assessing S'_K Versus S_{K+1}
$RS(R'_{K,2}, R_{K+1,2}) = true$
c)Top-down search
▩ Selected parent
▆ Part removed from the children on the lower level
▨ Part removed from the parent for reassessing of children who were worse
🟩 Re-assesment of the segment
(1) Assessing S'_{K+1} Versus S_K
$RS(R_{K,1}, R'_{K+1,1}) = true$, $RS(R_{K,2}, R'_{K+1,1}) = true$
(2) Assessing S'_K Versus S_{K-1}
$RS(R_{K-1,1}, R'_{K,1}) = true$, $RS(R_{K-1,2}, R'_{K,1}) = true$
$RS(R_{K-1,3}, R'_{K,2}) = false$
(3) Assessing S'_{K-1} Versus S_{K-2}
$RS(R_{K-2,1}, R'_{K-1,1}) = true$, $RS(R_{K-2,2}, R'_{K-1,2}) = true$
$RS(R_{K-2,3}, R'_{K-1,3}) = false$, $RS(R_{K-2,4}, R'_{K-1,3}) = true$
$RS(R_{K-2,3}, R''_{K-1,3}) = true$
(4) Assessing S'_{K-2} Versus S_{K-3}
$RS(R_{K-3,1}, R'_{K-2,1}) = false$, $RS(R_{K-3,2}, R'_{K-2,2}) = true$
$RS(R_{K-3,3}, R'_{K-2,2}) = false$, $RS(R_{K-3,4}, R'_{K-2,2}) = false$
$RS(R_{K-3,3}, R''_{K-2,2}) = false$, $RS(R_{K-3,4}, R''_{K-2,2}) = false$
d)Final selected segments (Bottom-up)
e) Final selected segments (Top-down)
f) Combination of regions by Bottom-up and Top-down

Figure 4. Graphical presentation of the search segmentation tree.

the set R_{TD} and removed from the lower level for the next iteration (shown as dark regions in **Figure 4(c)**). Children are always derived from the original mean shift segments, whereas parents are always derived from the updated mean shift segments except at the root level $(S'_{K+1} = S_{K+1})$. The search is stopped when the lowest level is reached or when the selected segments cover the entire image. The union of all segments in the set R_{TD} creates the segmented version of the input image (**Figure 4(e)**) for this direction of traversal.

Results from the bottom-up and the top-down searches typically differ from each other, thus sets R_{BU} and R_{TD} are later combined $(R_{\text{BU-TD}})$ to prevent missing real boundaries that might have been discarded in one of the two searches. In **Figure 4(f)**, even though the segments from R_{BU} and R_{TD} were different, the resulting boundaries are the same, which strengthens the segmentation.

2.2.5. Post-Processing of Segments

A post-processing merging algorithm may be applied on the segments of sets R_{BU} and R_{TD} that attempts to further merge neighboring segments according to their color similarity, texture similarity and presence of edges at their common boundaries. As the mean shift algorithm is based on color distinctiveness, it finds segments that may or may not conform to real edges in an image, which might also be a problem for other general segmentation methods that could be used in the hierarchical scheme. This post-processing merging step prevents from announcing boundaries in the middle of a smooth and rather uniform region that could have escaped the segment selection constraints because of non-ideal hierarchical segmentation layers.

The idea is to merge in passes, starting with the most similar pairs of neighboring segments and updating the segments before the next pass, stopping when no more pairs can be merged. The algorithm works as follows:

1) Put all the segments in the initial set Regions (Regions = R_{BU} or R_{TD}).

2) Create a new empty set of segments $(\text{New Regions} = \varnothing)$.

3) For every segment R_i in the set Regions, compute its features and find the labels of its immediate touching neighbors. The features are:

a) Color: normalized color group histogram (NCGHist_i) using the definitions presented in Section 2.2.2;

b) Texture: gradient magnitude mean and standard deviation values.

$$\left(\left[\text{GradMag}_i(1) = \text{mean, GradMag}_i(2) = \text{standard deviation}\right]\right)$$

4) For every segment R_i, compute its merging score (MScore_{ij}) with every neighbor R_j, which depends on color similarity and texture similariy:

$$\text{MScore}_{ij} = (\text{ColSim}_{ij} + \text{TS}_{ij}) \times \text{NoEdge} \quad (8)$$

$$\text{ColSim}_{ij} = 2 \times ((\text{HS}_{ij}) > 0.85) \quad (9)$$

$$\text{HS}_{ij} = \sum_{k=1}^{n} \sqrt{\text{NCGHist}_i(k) \times \text{NCGHist}_j(k)} \quad (10)$$

The color similarity ColSim_{ij} of neighboring segments is computed from the Bhattacharyya coefficient (HS_{ij}), which determines the relative closeness of two statistical samples, here the normalized histograms of the n color groups of each segment. HS_{ij} is a summation over the n color groups and ranges from 0 to 1, 1 corresponding to the most similarity. The threshold of 0.85 has been selected as to generally prevent the false merging of neighboring segments derived from the mean shift algorithm that are in fact different objects. This value was found using 100 test images of natural scenes.

$$\text{TS}_{ij} = \left(\left|\text{GM}_i(1) - \text{GM}_j(1)\right| < 2.5\%\right) + \left(\left|\text{GM}_i(2) - \text{GM}_j(2)\right| < 2.5\%\right) \quad (11)$$

The texture similarity TS_{ij} of neighboring segments relies on the difference of the mean and standard deviation of the gradient magnitude values over each segment. The gradient is first computed horizontally and vertically for each pixel of the image to find its dominant angle. The magnitude is then computed at that angle, which is approximated by one of the eight following values: 0, π/4, π/2, 3π/4, π, 5π/4, 3π/2, 7π/4. It is finally normalized between 0 and 1 over the entire image to facilitate its use as a comparison tool. The mean and standard deviation values of the gradient magnitude provide us with an approximation of the textureness of a segment relative to the content of the image, allowing the distinction between two neighbors of different materials. The thresholds of 2.5% have been set to prevent the false merging of the majority of neighboring segments derived from the mean shift algorithm that are in fact different in texture, over 100 test images of natural scenes.

$$\text{NoEdge} = \begin{cases} 1, & \dfrac{\text{Area}\left(\text{CB}_{ij} \cap \text{EM}\right)}{\text{Area}\left(\text{CB}_{ij}\right)} < 0.75 \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

The flag NoEdge is utilized in a way that if the common boundary between two neighboring segments $\left(\text{CB}_{ij}\right)$ covers at least 75% of the edge map EM, no merging should occur. The edge map used here is derived from the linked Canny edges presented in Section 2.2.2, dilated by a small structural element (square 3 × 3) to allow for small discontinuities and distortions. The merging score MScore_{ij} has a value ranging from 0 to 4, with 4 signifying the highest similarity. The color similarity is multiplied by a factor 2 to give it an equal weight to the texture similarity.

5) For every segment R_i, find the most similar neighbor R_j (highest score) and merge them if $\text{MScore}_{ij} \geqq 3$, *i.e.* a majority of the similarity criteria has to be reached. If there is a tie among two or more neighbors, then the neighbor having the highest values for the individual criteria is selected as best candidate for merging. Create a new segment from the union of the two neighbors and update the new set:

$$\text{New Regions} = \text{New Regions} \cup \left(R_i \cup R_j\right)$$

6) Once all segments have been assessed for merging, replace the working set Regions by the updated one:

$$\text{Regions} = \text{New Regions}$$

7) Repeat steps 3 to 6 until no pair of neighboring segments can be merged (the set New Regions remains equal to the set Regions).

8) Replace all the segments from New Regions in the initial set (R_{BU} *or* $R_{\text{TD}} = \text{New Regions}$). This post-processing merging algorithm can be repeated once more on the combined segments $\left(R_{\text{BU-TD}}\right)$.

3. Shape Specific Segmentation

For specific applications, a geometrical constraint via a shape prior can be added to the system to bias the segmentation towards a particular shape. This section shows how to integrate such a constraint into the proposed methodology (see Section 2) with minimal modifications.

Here we assume that the knowledge of the prior shape is provided via vector V_P. This vector is computed for a binary mask of the prior model. V_P can be defined (in general) by any scale and rotation invariant shape descriptor.

$$V_P = f_{\text{desc}}\left(S\right) \tag{13}$$

f_{desc} is a scale and rotation invariant shape descriptor function and S is the prior shape region for which vector V_P is computed. In general f_{desc} can be chosen according to the application and the complexity of the prior shape. In this work, we use a set of seven normalized central moments introduced by Hu [24]. Hu defines order three normalized central moments by:

$$M_1 = \eta_{20} + \eta_{02} \tag{14}$$

$$M_2 = \left(\eta_{20} - \eta_{02}\right)^2 + 4\eta_{11}^2 \tag{15}$$

$$M_3 = \left(\eta_{30} - 3\eta_{12}\right)^2 + \left(3\eta_{21} - \eta_{03}\right)^2 \tag{16}$$

$$M_4 = \left(\eta_{30} + \eta_{12}\right)^2 + \left(\eta_{21} + \eta_{03}\right)^2 \tag{17}$$

$$\begin{aligned} M_5 = {} & \left(\eta_{30} - 3\eta_{12}\right)\left(\eta_{30} + \eta_{12}\right)\left[\left(\eta_{30} + \eta_{12}\right)^2 - 3\left(\eta_{21} + \eta_{03}\right)^2\right] \\ & + \left(3\eta_{21} - \eta_{03}\right)\left(\eta_{21} + \eta_{03}\right)\left[3\left(\eta_{30} + \eta_{12}\right)^2 - 3\left(\eta_{21} + \eta_{03}\right)^2\right] \end{aligned} \tag{18}$$

$$M_6 = \left(\eta_{20} - \eta_{02}\right)\left[\left(\eta_{30} + \eta_{12}\right)^2 - \left(\eta_{21} + \eta_{03}\right)^2\right] + 4\eta_{11}\left(\eta_{30} + \eta_{12}\right)\left(\eta_{21} + \eta_{03}\right) \tag{19}$$

$$\begin{aligned}M_7 &= (3\eta_{21}-\eta_{30})(\eta_{30}+\eta_{12})\left[(\eta_{30}+\eta_{12})^2-3(\eta_{21}+\eta_{03})^2\right] \\ &+(3\eta_{12}-\eta_{30})(\eta_{21}+\eta_{03})\left[3(\eta_{30}+\eta_{12})^2-(\eta_{21}+\eta_{03})^2\right]\end{aligned} \tag{20}$$

where

$$\eta_{pq}=\frac{\mu_{pq}}{\mu_{00}^{\gamma}}, \qquad \gamma=\frac{p+q}{2}+1$$

and,

$$\mu_{pq}=\sum_x\sum_y (x-\overline{x})^p (y-\overline{y})^q f(x,y)$$

$f(x,y)$ represents the binary values of each image region (in the form of a mask). At each shape matching attempt, function f_{desc} returns the values of M_1 to M_7 for region R_{ij}.

$$V_R = f_{\text{desc}}(R_{ij}) = [M_1, M_2, \cdots, M_7]' \tag{21}$$

The shape similarity of a region R_{ij} with that of the prior shape (S) is measured by:

$$Sh(R_{ij}, S) = V_R - V_P \tag{22}$$

Clearly the shape similarity of the regions R_{ij} and S is maximum when the value of Sh is minimum. Function RS is redefined as the following to take the shape similarity into account:

$$RS(R_{ij}, R_{i+1,x}) = \begin{cases}\text{True}, D(R_{ij}) < D(R_{i+1,x}) \text{ and } E_D(R_{ij}) > E_D(R_{i+1,x}) \text{ and } Sh(R_{ij}, S) < Sh(R_{i+1,x}, S) \\ \text{False}, \quad \text{otherwise}\end{cases} \tag{23}$$

We will present the results of the addition of the shape constraint in the next section for geo-spatial aerial images.

4. Experimental Results

In this section, experimental results for various image types are presented: firstly for the general segmentation problem using the Berkeley Segmentation Dataset and Benchmark (BSDS300), and secondly for the specific application of geo-spatial aerial imagery segmentation and rooftop detection. We also demonstrate that our hierarchical segmentation scheme is applicable to other general segmentation methods.

4.1. Berkeley Dataset (BSDS300)

We tested our proposed algorithm on the publicly available Berkeley Segmentation Dataset and Benchmark (BSDS300) [25], which consists of human segmented natural images and a benchmark that assesses the quality of a boundary detection or segmentation algorithm based on the F-measure (harmonic mean between precision and recall rates).

4.1.1. Parameter Setting

The range resolution (h_c) for the entire dataset was defined as $T_l = 4$, $T_h = 16$, and $\Delta T = 2$. These values were selected in a way that the mean shift segments would cover from under to over segmentation for 90% of the dataset. **Figure 5** shows the mean shift segments of a typical image for all values of h_c.

4.1.2. Benchmark Results

Figure 6 presents typical results for the BSDS300 dataset. Our method successfully finds most of the objects in a given image. However, using the mean shift algorithm for the initial hierarchical segmentation has some shortcomings when it comes to natural scenes. Because it relies on color information and the distance relationship of pixels of similar color only, the segments may not correspond to what humans would define as different objects, different textures, or parts of the same object or same texture. Incorporating cues from edge distinctiveness improves the segment selection but cannot make up for real missed edges by the mean shift algorithm.

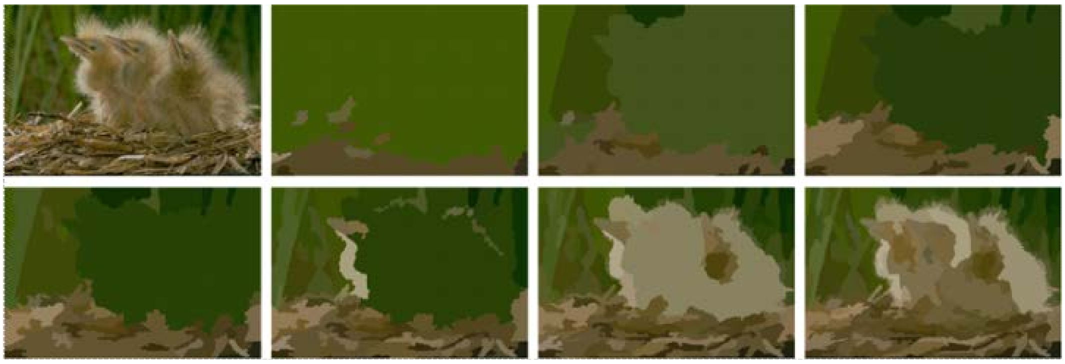


Figure 5. Mean shift segments of a typical BSDS300 image for a range resolution (h_c) of $T_l = 4$, $T_h = 16$ and $\Delta T = 2$. Original image shown top left.

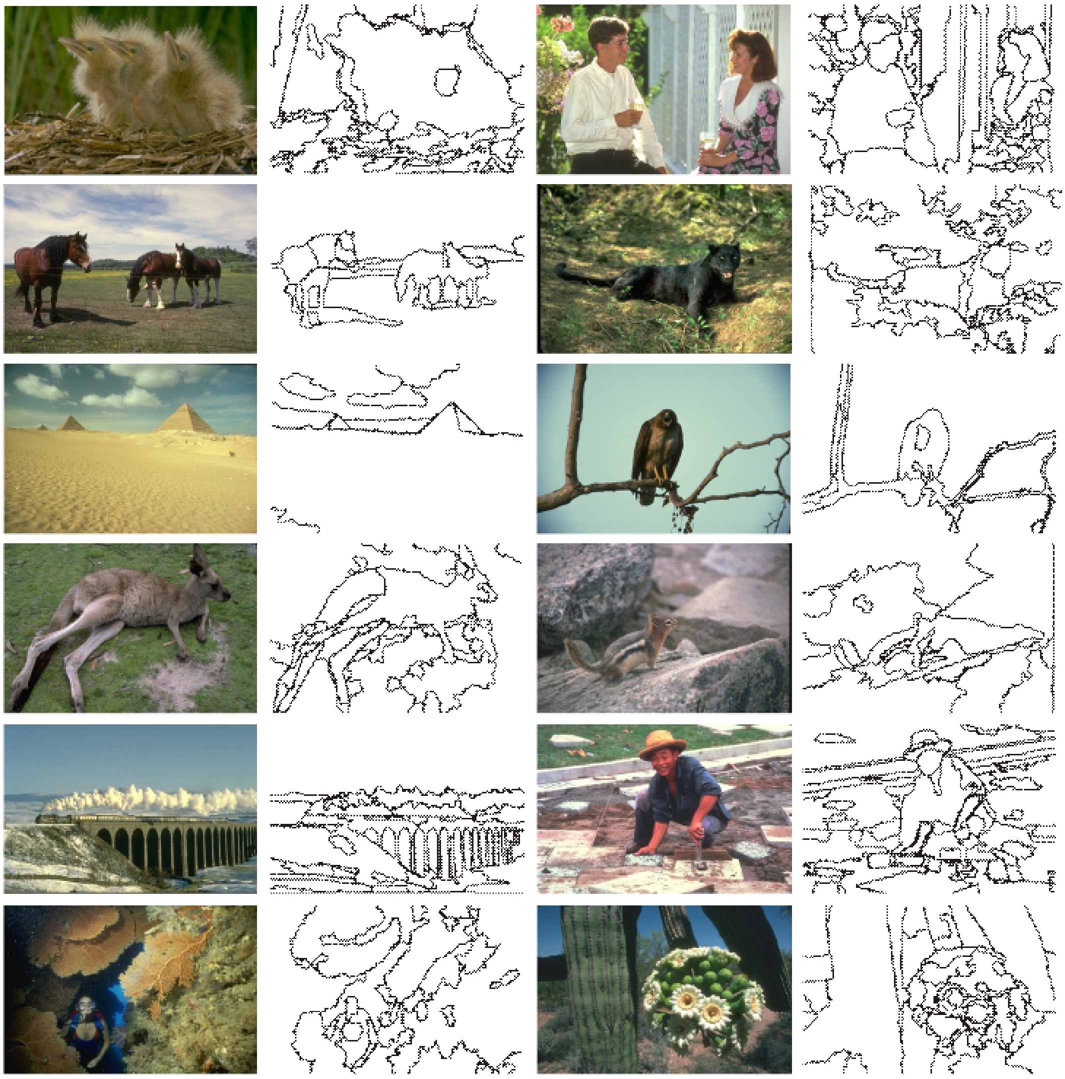


Figure 6. Sample images from BSDS300 dataset along with segmentation results of the proposed algorithm.

The algorithm has an overall F-measure of 0.642, which is similar to other recent algorithms as reported on the website of the benchmark (between 0.57 and 0.70). Methods that have achieved a higher score include an approximation of the weighted min-cover problem that extracts salient smooth curves [26], a method based on local changes in brightness, color, and texture cues [27], and an improvement of [27] coupling the local cues with global information obtained from spectral partitioning [28]. However, these algorithms are in fact boundary detection algorithms, and as such do not necessarily find closed segments representing objects. Our method has the advantages of being automatic, producing closed segments (true segmentation), and not requiring any parameter tuning nor training phase. Moreover, the benchmark is designed to accommodate soft boundary maps, with higher values signifying greater confidence in the existence of the boundary. Our method is penalized for producing closed binary boundaries, which may be more common for computer vision applications. Another aspect affecting the performance of any algorithm on the benchmark is related to the lack of consistency in human segmentations: the reported level of details varies from one image to another. An example would be for distinctive clouds in a sky that are sometimes segmented as different objects, sometimes completely merged as one large object with the blue sky.

4.1.3. Effect of the Different Stages

In this section, we illustrate the effect of the different stages of the hierarchical segmentation algorithm. **Figure 7** shows, for a typical image, selected segments from the bottom-up and the top-down searches both before and after post-processing (R_{BU} and R_{TD}), as well as their combination before and after post-processing $\left(R_{\text{BU-TD}}\right)$.

The selected segments differ according to the direction of traversal of the tree and complement each other. For instance, the bottom-up search failed to find the proper boundaries of the animal’s tail end, which were better defined through the top-down search. Combining the results allowed for a more accurate segmentation of the tail. The post-processing merging scheme was successful in removing some of the inaccurate boundaries, for instance on the animal’s body. For the bottom-up search, the post-processing reduced the number of segments from 106 to 44, and from 68 to 37 for the top-down search (there were 136 mean shift segments at $h_c = T_l$ and 17 at $h_c = T_h$).

Table 1 presents the overall F-measure, recall and precision rates of the entire dataset for the different stages. The best results come from the combination of bottom-up and top-down searches after post-processing (F-measure of 0.642), which justifies the usefulness of combining the two approaches. The bottom-up search has a higher recall rate than the top-down search, but a lower precision. This observation is consistent with the expectations as the bottom-up search, starting from the most sensitive level, has a tendency to find more segments, *i.e.* more true boundary pixels which increases the recall rate, but also more false boundary pixels at the same time, which decreases the precision rate. The top-down search, starting from the least sensitive level, has a tendency to find less segments, *i.e.* less true boundary pixels, which decreases the recall rate, but less false boundary pixels, which increases the precision rate.

4.1.4. Comparison with Non-Hierarchical Version

We also have compared our hierarchical segmentation approach, using mean shift as the general segmentation method, with the standard non-hierarchical mean shift segmentation algorithm. **Table 2** presents the overall F-measure, recall and precision rates of the entire dataset for values of h_c ranging from T_l (4) to T_h (16) with a ΔT of 4. Our hierarchical segmentation approach outperforms the same segmentation algorithm run with a non-hierarchical approach over the same range of parameter values, having a higher F-measure.

4.1.5. Applicability to Another General Segmentation Algorithm

In order to show that our hierarchical segmentation scheme can be integrated with other general segmentation algorithms that can produce segmentation outcomes at different granularities (from under to over segmentation) by varying their parameters’ values, we substituted the mean shift segmentation algorithm [19] with the graph-based image segmentation algorithm [2]. This algorithm segments images based on a predicate that measures the evidence of a boundary between two regions using a graph-based representation of the image. Its main control parameter *k* sets the scale of observation. It was defined as T_l = 100, T_h = 1300 and ΔT = 200 for the entire dataset. These values were selected in a way that the graph-based segments would cover from under to over segmentation for 90% of the dataset. **Figure 8** shows the graph-based segments of a typical image (same as in **Figure 5**) for all values of *k*.

Table 3 presents the overall F-measure, recall and precision rates of the entire dataset for the different stages, with the best results again coming from the combination of bottom-up and top-down searches after post-processing (F-measure of 0.628).

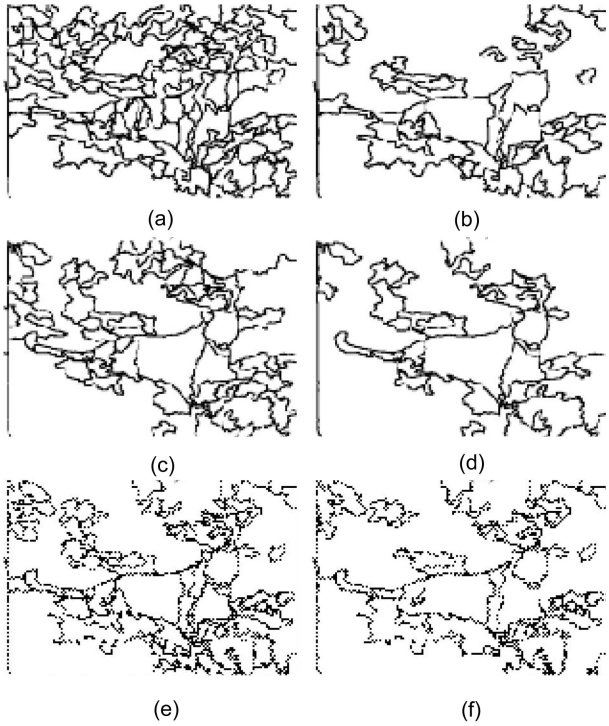


Figure 7. Results after different stages of the proposed algorithm for a typical BSDS300 image. Segments from bottom-up search (R_{BU}) before (a) and after (b) post-processing, segments from top-down search (R_{TD}) before (c) and after (d) post-processing, combination of the bottom-up and top-down results ($R_{BU\text{-}TD}$) before (e) and after (f) post-processing.

Table 1. F-measure, recall and precision rates for the different stages of the proposed algorithm.

Metric	Bottom-up		Top-down		Combination	
	Before P-P	After P-P	Before P-P	After P-P	Before P-P	After P-P
F-measure	0.623	0.636	0.566	0.569	0.637	0.624
Recall	0.734	0.689	0.521	0.508	0.717	0.682
Precision	0.540	0.591	0.620	0.646	0.573	0.607

Table 2. F-measure, recall and precision rates for non-hierarchical mean shift segmentations for different h_c values and our proposed method.

Metric	Mean shift, value of HC				Our method
	4	8	12	16	
F-measure	0.579	0.628	0.581	0.471	0.642
Recall	0.837	0.685	0.508	0.345	0.682
Precision	0.442	0.580	0.679	0.745	0.607

Table 3. F-measure, recall and precision rates for the different algorithm's stages using graph-based segmentation [2].

Metric	Bottom-up		Top-down		Combination	
	Before P-P	After P-P	Before P-P	After P-P	Before P-P	After P-P
F-measure	0.601	0.610	0.503	0.500	0.623	0.628
Recall	0.723	0.650	0.404	0.393	0.669	0.660
Precision	0.515	0.574	0.666	0.681	0.562	0.600

4.2. Geo-Spatial Aerial Images

To test the proposed algorithm in a practical and specific application, we utilized the proposed algorithm for the segmentation of aerial geo-spatial images (Pictometry Int. Corp.'s, resolution of 0.15 meter/pixel). The test images are from suburban regions of Vancouver, BC, Canada. The parent application for this case is a rooftop detection system. Since we are interested in selecting segments from the hierarchical tree that conform better to a rooftop profile, a square model was added as the shape prior to the geometrical constraint (Equation (23)).

As for the BSDS300, the range resolution (h_c) for all aerial images was defined in a way that the mean shift segments would cover from under to over segmentation for 90% of the dataset images. The values used are $T_l = 2$, $T_h = 12$ and $\Delta T = 2$. Three typical output results are shown in **Figure 9**, which comprise several types of buildings (detached houses, condominium buildings, commercial buildings). On the second row are the segmentation results and on the third row are the final results after applying a simple filtering process, aimed at discarding non-rooftop regions, consisting of the following steps:

1) Light morphological cleaning (opening with a squared structural element of 9 × 9 pixels) is carried out on each segment to remove small protrusions and separate objects that could be linked by a small isthmus.

2) All segments connected to the image border are discarded since we are looking only for rooftops that are fully contained within the image.

3) Segments corresponding to vegetation are removed. An analysis of several hundred objects (rooftops and non-rooftops) has shown that green vegetation (be it grass or trees) can be distinguished from rooftops based on the saturation component of the HSV color space. A vegetation mask (M_v) is created from pixels that have a saturation value over 0.3. To strengthen the approach due to the nonetheless existing overlap in saturation value, a second constraint related to hue is added: a tolerance spanning from yellow $(H = 0.1)$ to cyan $(H = 0.5)$ is allowed:

$$M_v(x,y) = \begin{cases} 1, S(x,y) > 0.3 \text{ and } 0.1 < H(x,y) < 0.5 \\ 0, \qquad\qquad\qquad \text{otherwise} \end{cases} \tag{24}$$

here, x and y denote pixel coordinates, and S and H the saturation and hue bands, respectively.

4) Segments smaller than a specific size are removed. For aerial images the minimum size of a rooftop was defined at 1000 pixels (22.5 m^2).

Quantitative assessment of the results is done through four metrics which have become standard for the evaluation of man-made structure extraction [29]: shape accuracy, correctness, completeness and overall quality. The results are compared with manually delimitated ground truth data. The shape accuracy [30], based on the overlap between the segmented rooftop and its ground truth, is estimated by:

$$\text{Shape Accuracy} = 1 - \left\| A_{\text{GT}} - A_{\text{S}} \right\| / A_{\text{GT}} \tag{25}$$

$$\text{Completeness} = \text{TP} / (\text{TP} + \text{FN}) \tag{26}$$

$$\text{Correctness} = \text{TP} / (\text{TP} + \text{FP}) \tag{27}$$

$$\text{Quality} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \tag{28}$$

here A_{GT} and A_{S} denote a rooftop's area from the ground truth and the segmentation process, respectively. As errors of pixel labeling are not taken into account in the shape accuracy, the metrics of correctness, completeness, and overall quality are computed as well. Correctness measures the degree to which detected building pixels are indeed real building pixels, whereas completeness measures the degree to which real building pixels are detected by the system. Overall quality is the normalization between the previous two metrics.

Here, TP represents true positives (correctly extracted building pixels), FP false positives (incorrectly extracted building pixels), and FN false negatives (missed building pixels). When computing these results, all missing and falsely detected rooftops in **Figure 9** are accounted for. When one rooftop is detected as several segments, the system is penalized accordingly, considering only the largest segment as true positives. Optimal values for the four metrics are 1 (or 100%), and the overall quality cannot be higher than neither correctness nor completeness. The mean shape accuracy is computed rooftop-wise whereas the mean values of the other three metrics are computed pixel-wise across all images.

The proposed method (**Table 4**) holds a shape accuracy of 84.6%, correctness of 84.2%, completeness of

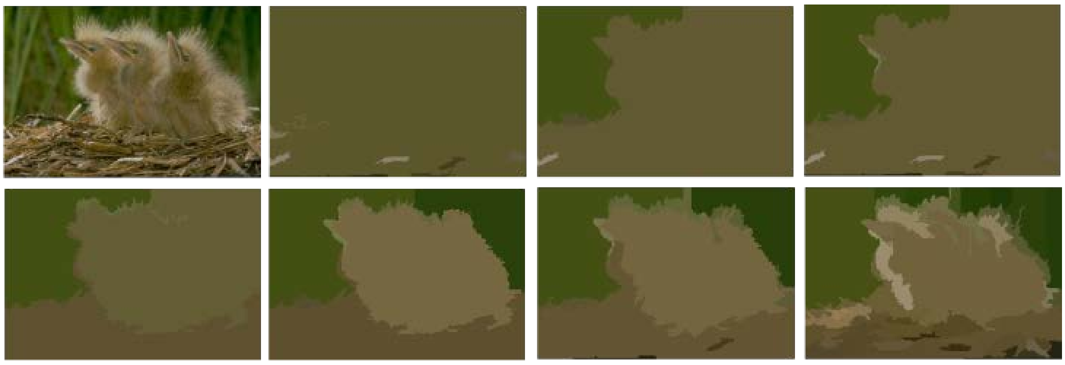


Figure 8. Graph-based segments of a typical BSDS300 image for a scale of observation (k) of $T_l = 100$, $T_h = 1300$ and $\Delta T = 200$. Original image shown on the top left.

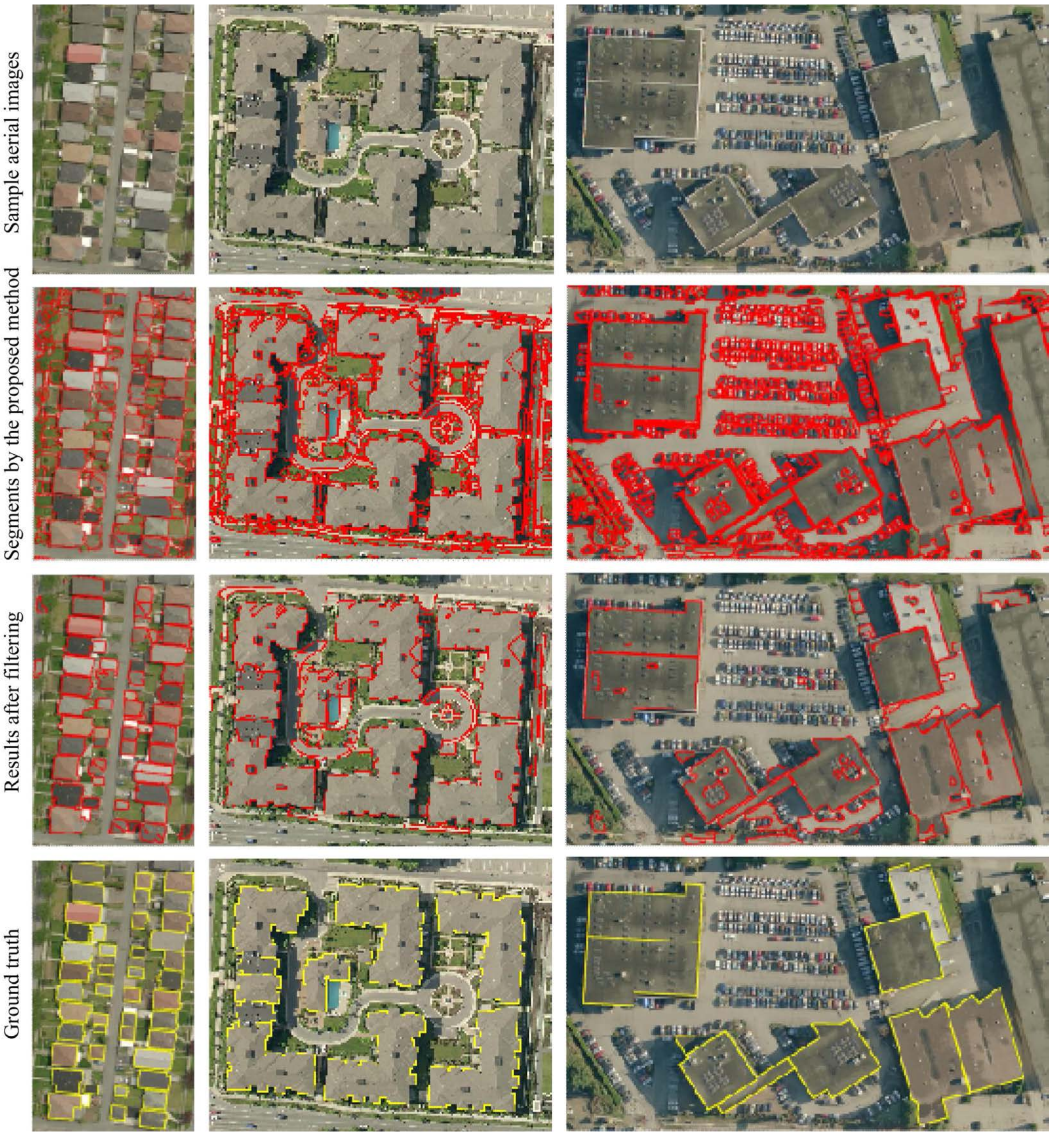


Figure 9. Examples of aerial images that are processed by the proposed algorithm to identify rooftops. Segmentation results are shown on the second row, and results after filtering on the third and the ground truth on the last row.

Table 4. Quantitative results for geo-spatial images.

	Shape accuracy [%]	Correctness [%]	Completeness [%]	Overall quality [%]
Without shape constraint	80.3	77.5	89.7	71.1
With shape constraint	84.6	84.2	92.1	78.5

92.1% and overall quality of 78.5%. Its shape accuracy is comparable to that reported in other works (81% in [31], 83.6% in [32]). It performs better with the shape constraint (Equation (23)) than without it. The number of buildings in the ground truth data for the first image (**Figure 9**, first column) was 44 out of which 43 were detected by the proposed system. 10 false positives were also detected in this case. The second image (**Figure 9**, second column) included 11 rooftops, which were all detected. 33 false positives were also detected. The third image (**Figure 9**, third column) had 10 rooftops and all of them were identified correctly but the system also identified 15 false positives. The high completeness rate reflects that most of the rooftop pixels are detected as such. Because the method was developed mainly as a general segmentation approach, in its current state it detects a substantial number of non-rooftop objects.

A simple filtering scheme was applied to discard some of the background segments. One way to increase the performance and get higher correctness and overall quality would be to filter out segments with no evident shadows. By using the image acquisition parameters in estimating the direction of the sun and the expected shadows of each segment (rooftop candidate), flat ground structures such as roads and fields could be removed, impacting the quantitative assessment metrics significantly.

5. Conclusion

An automatic segmentation approach was introduced in this paper. The proposed approach uses a hierarchical multi-layer segmentation scheme with a tree-based search mechanism to inspect the consistency of segmented regions and their relationship with each other, providing automatic adaptive parameter selection. Performance of the algorithm was showcased by applying it on different types of images including natural scenes through the Berkeley Segmentation Dataset and Benchmark (BSDS300) and geo-spatial imageries. The proposed hierarchical scheme displayed comparable performance with recent algorithms on the BSDS300 dataset, and outperformed the original non-hierarchical segmentation algorithm for the same range of parameter values. While the mean shift algorithm was used as a general segmentation algorithm to demonstrate the feasibility and usefulness of our hierarchical segmentation scheme for most results, we have also shown the applicability of the approach to other general segmentation algorithms that can produce segmentations at several granularities, such as the graph-based image segmentation algorithm. Future works include the analysis of the applicability to soft and/or stochastic segmentation environments.

Acknowledgements

The authors would like to acknowledge with gratitude NSERC Canada and MacDonald Dettwiler and Associates Ltd. for support through the NSERC Discovery Grant Program.

References

[1] Chen, H., Chien, W. and Wang, S. (2004) Contrast-Based Color Image Segmentation. *IEEE Signal Processing Letters*, **11**, 641-644. http://dx.doi.org/10.1109/LSP.2004.830116

[2] Felzenszwalb, P. and Huttenlocher, D. (2004) Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, **59**, 167-181. http://dx.doi.org/10.1023/B:VISI.0000022288.19776.77

[3] Lo, E.H., Pickering, M.R., Frater, M.R. and Arnold, J.F. (2011) Image Segmentation from Scale and Rotation Invariant Texture Features from the Double Dyadic Dual-Tree Complex Wavelet Transform. *Image and Vision Computing*, **29**, 15-28. http://dx.doi.org/10.1016/j.imavis.2010.08.004

[4] Tan, K.S. and Isa N.A.M. (2011) Color Image Segmentation Using Histogram Thresholding—Fuzzy C-Means Hybrid Approach. *Pattern Recognition*, **44**, 1-15. http://dx.doi.org/10.1016/j.patcog.2010.07.013

[5] Cootes, T.F., Taylor, C.J., Cooper, D.H. and Graham, J. (1995) Active Shape Models—Their Training and Application.

Computer Vision and Image Understanding, **61**, 38-59. http://dx.doi.org/10.1006/cviu.1995.1004

[6] Staib, L.H. and Duncan, J.S. (1992) Boundary Finding with Parametrically Deformable Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**, 1061-1075. http://dx.doi.org/10.1109/34.166621

[7] Wang, Y. and Staib, L.H. (1998) Boundary Finding with Correspondence Using Statistical Shape Models. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Santa Barbara, 23-25 June 1998, 338-345.

[8] Fussenegger, M., Roth, P.M., Bischof, H. and Pinz, A. (2006) On-Line, Incremental Learning of a Robust Active Shape Model. *DAGM-Symposium*, 122-131.

[9] Tsai, A., Yezzi, A., Wells, W., Tempany, C., Tucker, D., Fan, A., Grimson, W. and Willsky, A. (2001) Model-Based Curve Evolution Techniques for Image Segmentation. *Proceedings of the* 2001 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Kauai, 8-14 December 2001, I-463-I-468.

[10] Tsai, A., Yezzi, A., Wells, W., Tempany, C., Tucker, D., Fan, A., Grimson, W. and Willsky, A. (2003) A Shape-Based Approach to the Segmentation of Medical Imagery Using Level Sets. *IEEE Transactions on Medical Imaging*, **22**, 137-154. http://dx.doi.org/10.1109/TMI.2002.808355

[11] Foulonneau, A., Charbonnier, P. and Heitz, F. (2006) Affine-Invariant Geometric Shape Priors for Region-Based Active Contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**, 1352-1357. http://dx.doi.org/10.1109/TPAMI.2006.154

[12] Chen, Y., Tagare, H., Thiruvenkadam, S., Huang, F., Wilson, D., Gopinath, K.S., Briggs, R.W. and Geiser, E. (2002) Using Shape Priors in Geometric Active Contours in a Variational Framework. *International Journal of Computer Vision*, **50**, 315-328. http://dx.doi.org/10.1023/A:1020878408985

[13] Leventon, M.E., Grimson, W.E.L. and Faugeras, O. (2000) Statistical Shape Influence in Geodesic Active Contours. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Hilton Head Island, 13-15 June 2000, 316-323.

[14] Rousson, M. and Paragios, N. (2008) Prior Knowledge, Level Set Representations & Visual Grouping. *International Journal of Computer Vision*, **76**, 231-243. http://dx.doi.org/10.1007/s11263-007-0054-z

[15] Riklin-Raviv, T., Kiryati, N. and Sochen, N. (2007) Prior-Based Segmentation and Shape Registration in the Presence of Perspective Distortion. *International Journal of Computer Vision*, **72**, 309-328. http://dx.doi.org/10.1007/s11263-006-9042-y

[16] Chan, T.F. and Vese, L.A. (2001) Active Contours without Edges. *IEEE Transactions on Image Processing*, **10**, 266-277. http://dx.doi.org/10.1109/83.902291

[17] Sharon, E., Galun, M., Sharon, D., Basri, R. and Brandt, A. (2006) Hierarchy and Adaptivity in Segmenting Visual Scenes. *Nature*, **442**, 810-813. http://dx.doi.org/10.1038/nature04977

[18] Paris, S. and Durand, F. (2007) A Topological Approach to Hierarchical Segmentation Using Mean Shift. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, 17-22 June 2007, 1-8.

[19] Comaniciu, D. and Meer, P. (2002) Mean Shift: A Robust Approach toward Feature Space Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**, 603-619. http://dx.doi.org/10.1109/34.1000236

[20] Alpert, S., Galun, M., Brandt, A. and Basri, R. (2012) Image Segmentation by Probabilistic Bottom-Up Aggregation and Cue Integration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **34**, 315-327. http://dx.doi.org/10.1109/TPAMI.2011.130

[21] Corso, J.J., Yuille, A. and Tu, Z. (2008) Graph-Shifts: Natural Image Labeling by Dynamic Hierarchical Computing. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, 23-28 June 2008, 1-8.

[22] Bezdek, J.C. (1981) Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York. http://dx.doi.org/10.1007/978-1-4757-0450-1

[23] Duda, R., Hart, P. and Stork, D. (2001) Pattern Classification. John Wiley & Sons, New York.

[24] Hu, M.K. (1962) Visual Pattern Recognition by Moment Invariants. *IRE Transactions on Information Theory*, **8**, 179-187. http://dx.doi.org/10.1109/TIT.1962.1057692

[25] Martin, D.R., Fowlkes, C., Tal, D. and Malik, J. (2001) A Database of Human Segmented Natural Images and Its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics. Tech. Rep. UCB/CSD-01-1133, EECS Department, University of California, Berkeley.

[26] Felzenszwalb, P. and McAllester, D. (2006) A Min-Cover Approach for Finding Salient Curves. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*, New York, 17-22 June 2006, 185.

[27] Martin, D.R., Fowlkes, C.C. and Malik, J. (2004) Learning to Detect Natural Image Boundaries Using Local Brightness, Color, and Texture Cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**, 530-549.

http://dx.doi.org/10.1109/TPAMI.2004.1273918

[28] Maire, M., Arbeláez, P., Fowlkes, C. and Malik, J. (2008) Using Contours to Detect and Localize Junctions in Natural Images. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, 23-28 June 2008, 1-8.

[29] Agouris, P., Doucette, P. and Stefanidis, A. (2004) Automation and Digital Photogrammetric Workstations. In: *Manual of Photogrammetry*, 5th Edition, American Society for Photogrammetry and Remote Sensing, Bethesda, 949-981.

[30] McKeown, D., Bulwinkle, T., Cochran, S., Harvey, W., McGlone, C. and Shufelt, J. (2000) Performance Evaluation for Automatic Feature Extraction. *International Archives of Photogrammetry and Remote Sensing*, **33**, 379-394.

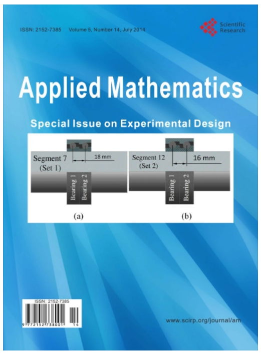
[31] Ruther, H., Martine, H.M. and Mtalo, E.G. (2002) Application of Snakes and Dynamic Programming Optimization Technique in Modeling of Buildings in Informal Settlement Areas. *ISPRS Journal of Photogrammetry and Remote Sensing*, **56**, 269-282. http://dx.doi.org/10.1016/S0924-2716(02)00062-X

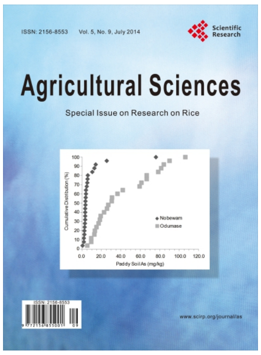
[32] Peng, J. and Liu, Y.C. (2005) Model and Context-Driven Building Extraction in Dense Urban Aerial Images. *International Journal of Remote Sensing*, **26**, 1289-1307. http://dx.doi.org/10.1080/01431160512331326675

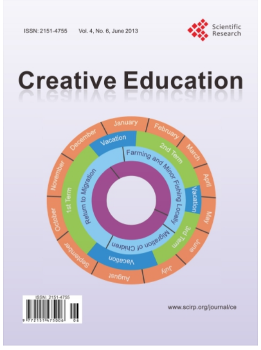
Advances in
Bioscience and Biotechnology

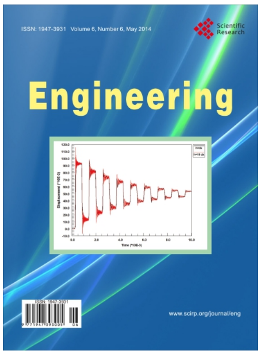
American Journal of
Analytical Chemistry

American Journal of
Plant Sciences
Special Issue on Photosynthesis

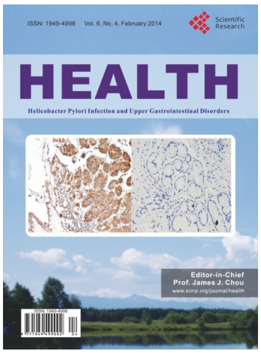

Applied Mathematics
Special Issue on Experimental Design


Agricultural Sciences
Special Issue on Research on Rice


Creative Education

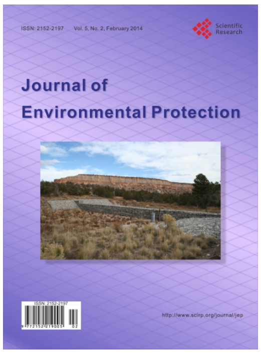

Engineering

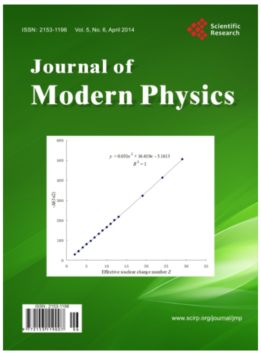
Food and Nutrition Sciences


HEALTH
Helicobacter Pylori Infection and Upper Gastrointestinal Disorders
Editor-in-Chief
Prof. James J. Chou

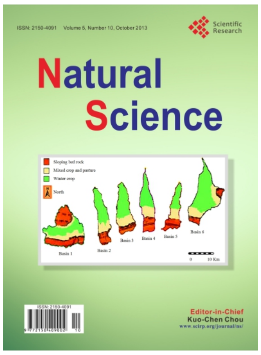
Journal of Computer
and Communications

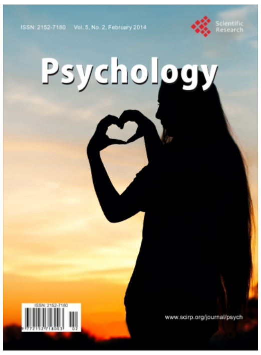
Journal of Cancer Therapy
Special Issue on Bone Tumor Research


Journal of
Environmental Protection


Journal of
Modern Physics

Modern Economy


Natural
Science
Editor-in-Chief
Kuo-Chen Chou


Psychology