AN ACCURACY ASSESSMENT MEASURE FOR OBJECT BASED IMAGE SEGMENTATION

Nicholas Clinton[a, *], Ashley Holt[a], Li Yan[b], Peng Gong[ac].

[a]Department of Environmental Science, Policy and Management. University of California, Berkeley CA 94720-3114
[b]International Institute for Earth System Science. Nanjing University, China. 210093 – nclinton@nature.berkeley.edu
[c]State Key Laboratory of Remote Sensing Science, Jointly Sponsored by Institute of Remote Sensing Applications, Chinese Academy of Sciences, and Beijing Normal University, 3 Datun Road, Chaoyang District, Beijing 100101

KEY WORDS: Segmentation, ASTRO, BerkeleyImageSeg, Ecognition, Definiens

ABSTRACT:

Traditional approaches to accuracy assessment are inadequate for object oriented image processing. We tested some measures to assess the accuracy of object based image segmentation in a supervised context. The measures quantify the extent to which objects in the segmentation match training objects in terms of over-segmentation, under-segmentation, and distance to a perfect match. Using high resolution digital aerial photographs over an urban setup, we obtained segmentation results for a variety of parameter combinations using two software packages: eCognition and ASTRO. We compute the accuracy measures using three types of objects: vehicles, trees and buildings. The measures were used to compare the software, identify ideal parameter combinations, and identify objects that each software is better at extracting from the images. The measures are shown to be an intuitive, useful technique for consistency checking different segmentation results and assessing segmentation accuracies among a large set of disparate segmentation results.

INTRODUCTION

In object based image processing, the first step is generally a segmentation of the image of interest. A wide variety of segmentation results may be obtained through different parameter combinations or different segmentation software. Prior to classification or even to training of a suitable classifier, one of the segmentation results must be chosen. In this paper, we describe well defined measures that can be used in the identification of a "best" segmentation and the "best" objects within that segmentation for training a classifier. These measures are applicable in the supervised setting only, and "bestness" is therefore relative to a set of pre-defined training objects (assumed polygons) over the image of interest.

Assume first that the landscape of interest is a finite population of objects (Bian 2007). The spatial information about these objects is useful in the ultimate classification of the object (Gong and Howarth 1990). It is obvious that exact representation of the objects in the segmentation is important, since this shape information will eventually be presented to a classifier for the identification of a pattern. The accuracy of the classification is thus dependent on the accuracy of the shape information submitted to the classifier. Measures of the segmentation result are therefore relevant to the interpretation and optimization of ultimate classification accuracy. The measures we tested are *not* measures of classification accuracy, but are related. Assuming that accuracy assessment is conducted with statistical rigor, a probability sample will be obtained on the population of objects (Stehman and Czaplewski 1998, Stehman 1999). If the population is assumed to be represented by the segmentation result and a simple random sample is used to generate accuracy statistics, then the accuracy of the shapes has been completely ignored! On the other hand, if a sample is taken from the landscape directly (*e.g.* human delineated training polygons are used) and compared to the segments, then areas of intersection between mapped classes and reference classes affect the resultant accuracy. The accuracy of the segmentation will thus directly influence the classification accuracy, unless classification is performed on object primitives, a different problem discussed below.

There are a large number of methods with which to judge segmentations (Zhang 1996). This study is focused on the scenario in which a set of training objects is available for a static image and segmentation results are to be compared to these pre-defined training objects. Unlike unsupervised evaluation of segmentation results (Levine and Nazif 1985, Ng and Lee 1996, Borsotti et al. 1998, Chabrier 2006), spectral aspects (such as homogeneity within segment or within class) of the resultant segments are not considered and the quality of segments is evaluated solely in respect to the shape of training objects. In this context, a segmentation result should contain segments that match the training objects. For automatically checking this, a simple, intuitive measure of polygon matching can be computed. This measure relies on the observation that there should be a one-to-one correspondence (in area) between human identified objects (training objects) and segments. A measure of this correspondence was first proposed by Levine and Nazif (1982) and demonstrated by Yang et al. (1995). Moller et al. (2007) describe a similar measure called Relative Area (RA) which relies on the ratio of intersected area to segment and reference object area. We present the intuition behind this measure, a refinement of its computation, and a case study using high resolution urban imagery.

SEGMENTATION GOODNESS MEASURES

In a supervised interpretation of the segmentation result, let $X = \{x_i: i=1...n\}$ be the set of n training objects, assumed polygons, relative to which the segmentation is to be judged. Let $Y = \{y_j: j=1...m\}$ be the set of all segments in the segmentation. Let $\tilde{Y}_i$ be a subset of Y such that:

$$\tilde{Y}_i = \{y_j : area(x_i \cap y_j) \neq 0\}.$$

For convenience, let area($x_i \cap y_j$) = the area of the geographic intersection of training object x_i and segment y_j and area(·) be the geographic area of ·. For each training object x_i, the following subsets of Y exist:

$$Y_a = \{\text{all } y_j \text{ where the centroid of } x_i \text{ is in } y_j\}$$
$$Y_b = \{\text{all } y_j \text{ where the centroid of } y_j \text{ is in } x_i\}$$
$$Y_c = \{\text{all } y_j \text{ where area}(x_i \cap y_j) / \text{area}(y_j) > 0.5\}$$
$$Y_d = \{\text{all } y_j \text{ where area}(x_i \cap y_j) / \text{area}(x_i) > 0.5\}$$

The union of these subsets is the subset $Y_i^* = Y_a \cup Y_b \cup Y_c \cup Y_d$ where Y_i^* is assumed to be the subset of segments that are *relevant* to training object x_i. Processing over Y^* is designed to minimize if not eliminate the effects of spurious intersections with very small parts of very large segments. Define #(Y_i^*) = p_i and $\sum_{i=1...n} p_i = P$. Thus, for each training object, there are p_i segments deemed relevant to it.

Define the following properties of the segments in Y_i^*:

$$\text{OverSegmentation}_{ij} = 1 - \text{area}(x_i \cap y_j) / \text{area}(x_i).$$
$$\text{UnderSegmentation}_{ij} = 1 - \text{area}(x_i \cap y_j) / \text{area}(y_j).$$

Here, we have simply rescaled Moller et al. (2007) *RAsub* (as OverSegmentation) and *RAsuper* (as UnderSegmentation) in order to facilitate their combination and minimization on a [0,1] scale. We have also defined them on the Y^* subset of intersecxted segments. Observe that OverSegmentation and UnderSegmentation are properties of the segments, but can be averaged over the p_i segments associated with each training object, and in turn averaged over the n training objects. Alternatively, OverSegmentation and UnderSegmentation can be averaged over the P segmentation objects that interact with the set of all training objects, X. The difference is related to whether these measures should be weighted by the training objects, larger or more extensive training polygons being likely to interact with more segments than smaller ones. The un-weighted version first averages OverSegmentation and UnderSegmentation for each training object, then averages over all the training objects. Both the weighted and un-weighted averages can be used as indicators of overall segmentation quality relative to the training set X.

The range of OverSegmentation and UnderSegmentation is in [0,1], where OverSegmentation=0 and UnderSegmentation=0 define a *perfect* segmentation, where the segments match the training objects exactly. Obviously, imperfect segmentations, as defined here, could result from poor delineation of training objects, in combination with poor segmentation. Assuming that the training objects in X are exact, OverSegmentation and UnderSegmentation also have the nice property of identifying segments that match the training objects more or less perfectly. Combining the measures could result in a method to sort or rank the segments (for classification purposes) in terms of agreement with the furnished training objects.

The two dimensional space defined by OverSegmentation and UnderSegmentation is the unit square S. As a result of the fact that the ideal segmentation result is a point at the origin in this space, the Euclidean norm of a vector with coordinates (OverSegmentation, UnderSegmentation) is a measure for the quality of a segmentation (Levine and Nazif (1982) first propose this and an absolute value based combination of metrics). Let the "distance" index D be as follows:

$$D = \sqrt{OverSegmentation^2 + UnderSegmentation^2}$$

This index D should be interpreted as the "closeness" in the space defined above to an ideal segmentation result, in the context of a pre-defined training set. In this context, D is in $[0, 2^{1/2}]$. The distance index can be defined for each segment y_j in Y^*, averaged over each training object x_i, or averaged over the set of all training objects X to produce a composite index for the entire segmentation result.

METHODS

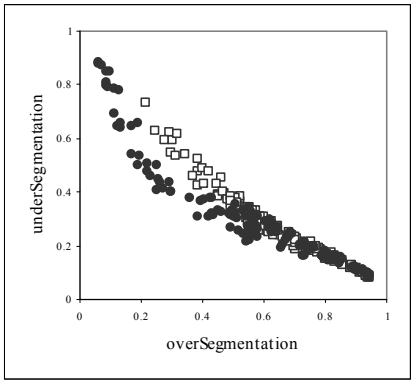
The imagery we used is a 3 band (RGB) aerial image of an urban area in San Francisco, California, USA. Resolution is approximately 0.174 meters. Using the imagery and parameter combinations described by Holt et al. (under review), we obtained segmentation results for two different software packages: eCognition (http://www.definiens.com) and ASTRO (http://berkenviro.com/berkeleyimgseg/). Both of these programs use a region merging technique to obtain a complete spatial partition of the input image pixels. ASTRO is developed based on the region merging algorithms described in Benz et al. (2003).

Both software packages perform segmentation and export the results as polygons in the ESRI shapefile format. In total, 150 parameter combinations were examined for scale, smoothness and color according to {10, 20, 30, 40, 50}x{0.1, 0.3, 0.5, 0.7, 0.9}x{0.1, 0.3, 0.5, 0.7, 0.9}, respectively. Using the resultant shapefile from each parameter combination, we computed the measures in the Java environment using JTS (http://www.vividsolutions.com/jts/jtshome.htm) and GeoTools (http://geotools.codehaus.org/).

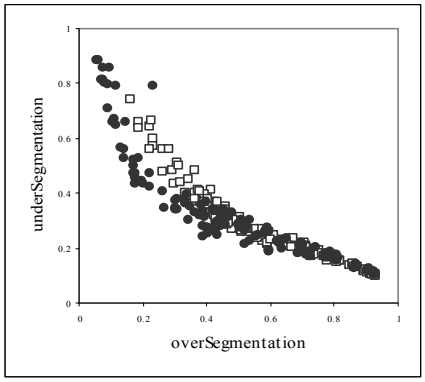
For training sets, we digitized 119 vehicles (cars and trucks) as simple rectangles, 48 tree crowns, and 36 building rooftops for a total of 203 training shapes. Relative to these training object sets (vehicles, trees, buildings and combined), we computed OverSegmentation and UnderSegmentation for each combination of parameters in each software package and examined the goodness D when averaged over the n training objects in X and averaged over $y_j \in Y_i^*$, $\forall$ i,j. Resultant segmentation results were visually examined and interpreted. The results are reported below.

RESULTS

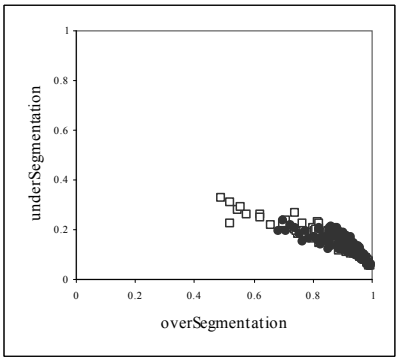
Figure 1 shows the overall segmentation results when OverSegmentation and UnderSegmentation are averaged over $y_j \in Y_i^*$, $\forall$ i,j (left) and when OverSegmentation and UnderSegmentation are first averaged for each training object, then averaged over all training objects (right). The behavior of eCognition and ASTRO in response to parameter variation is illustrated in Figure 1.



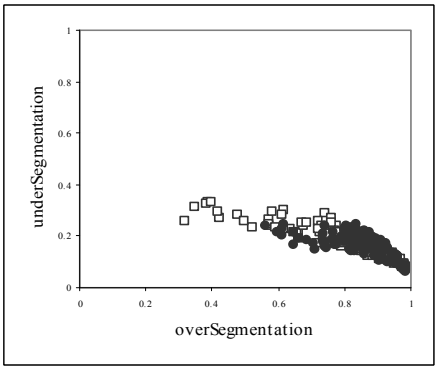
a. Vehicles, average of all segments in Y^*



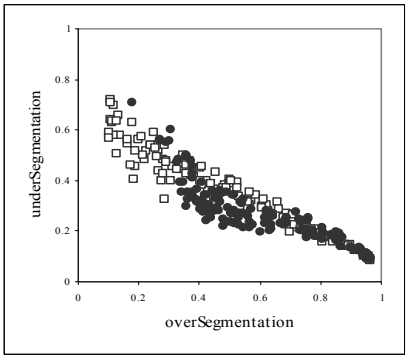
b. Vehicles, average of training objects



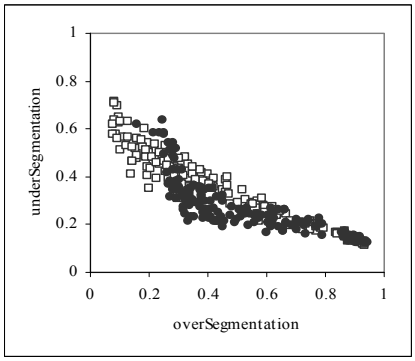
c. Buildings, average of all segments in Y^*



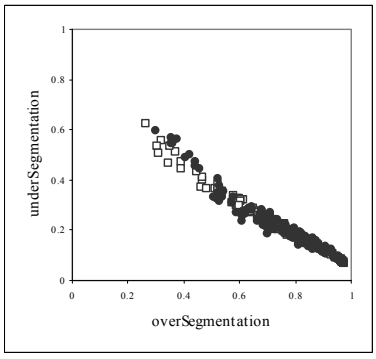
d. Buildings, average of training objects



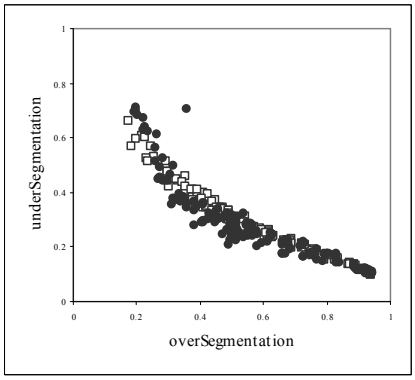
e. Trees, average of all segments in Y^*



f. Trees, average of training objects



g. Average of all segments in Y^* for all training objects



h. Average over all training objects

Figure 1. The segmentation results when averaged over $y_j \in Y_i^*$, $\forall\, i,j$. ASTRO = □, eCognition = ♦.

Figure 1. The segmentation results when averaged over x_i, $\forall$ i. ASTRO = □, eCognition = ♦.

For vehicles (Figure 1 a and b), eCognition is more responsive to parameter settings, judging from the more pronounced curvature in the distribution of parameter combinations over *S*. It is also readily apparent that eCognition produces results closer to an optimal segmentation near the origin. For larger training objects, such as buildings (Figure 1. c and d), the opposite is true, with ASTRO producing a result closer to the origin (smaller *D*). The software is less distinguishable from the other training object sets (trees, Figure 1 e and f; and combined vehicles, buildings and trees, Figure 1 g and h).

For both ASTRO and eCognition (relative to combined vehicles, buildings and trees), the parameter combinations with the lowest *D* values differ from the combinations with the lowest *D* when averaged over training objects. For ASTRO, the scale=50, color=0.1, smoothness=0.5 combination minimizes *D* while the scale=40, color=0.1, smoothness=0.5 minimizes *D* when it is computed by averaging over training objects. For eCognition, the scale=60, color=0.3, smoothness=0.3 combination minimizes *D* while the scale=40, color=0.3, smoothness=0.1 minimizes *D* when it is computed by averaging over training objects. The segmentations that result from parameter combinations that minimize *D* are shown in Figures 2 and 3 for a subset of the image we used. Figure 4 shows the training objects in the same subset. The results are quite obviously qualitatively different, suggesting that visual interpretation of the segmentation is relevant to the ultimate selection of a particular parameter combination or segmentation software.

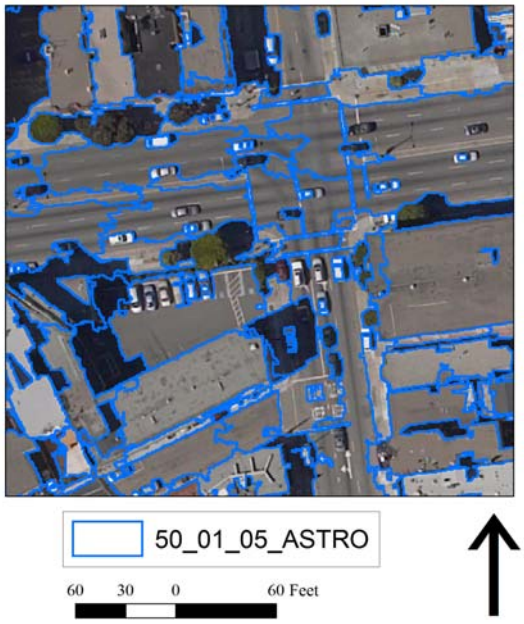


Figure 2. The ASTRO result that minimizes *D*: scale=50, color=0.1, smoothness=0.5.

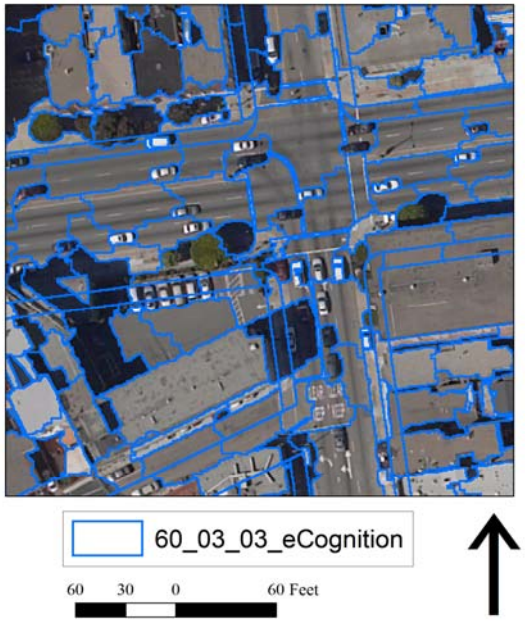


Figure 3. The eCognition result that minimizes *D*: scale=60, color=0.3, smoothness=0.3.

Figure 4. The training shapes corresponding to the area in Figures 2 and 3.

DISCUSSION

The problem of finding an optimal configuration of parameter settings has been addressed by Holt et al. (accepted). The index *D* can be used for this purpose, though it not been attempted for this study. The procedure involves fitting a convex function of scale,

smoothness and color to the observed *D*, then minimizing the function.

The evaluation of segmentation relative to a training set is simply a quantitative measure of the goodness of polygon matching. It does not necessarily imply a good classification result. This is particularly true in the event that a classification of primitives can be used as a preliminary step to the ultimate assembly of objects (see, for instance, Pichel et al. 2006). For example, consider an evaluation of segmentation results relative to the ultimate classification of an entire vehicle. This discounts the prospect of first classifying vehicle parts such as windshield, hood, roof, etc., then assembling these parts into complete cars through dissolve operations or other adjacency rules. However, the method described here could easily be applied to such a scenario through the provision of training sets for the individual car parts, then evaluating the goodness of match between the segmentation and the supplied primitives. These hierarchical relationships between objects at different spatial scales could be more easily exploited using OverSegmentation and UnderSegmentation. With any software that produces nested segmentations at different scales (as both ASTRO and eCognition do), the *D* measure could be harnessed to compare predefined object primitives to a wide variety of segmentations at different scales. In this way, optimal scales for analysis could be identified by comparing the training objects to different levels of the hierarchy.

The advantage of the measures we describe is that a quantitative index can be generated relative to any set of training objects of interest. The measures will also provide useful diagnostic information for the efficacy of the segmentation relative to the different object types. This characteristic of *D* is illustrated by Figure 1, in which the performance of the different software is shown to be very different when supplied with different kinds of training objects.

In the event that two segmentation results have similar values of *D*, the setup described here can be extended to incorporate additional indices. However, the indices should be scaled to [0,1] and increase the dimension of *S*, with the Euclidean norm *D* calculated accordingly. The distribution of *D* in *S* is of great interest and should be defined in order to determine the significance of differences between segmentation results. Simulation studies are needed to identify this distribution.

CONCLUSION

We have presented and demonstrated measures that facilitate the identification of optimal segmentation results relative to a training set. We propose that these measures are not only useful for the selection of segmentations from an array of choices, but also have utility in reporting the overall accuracy of segmentation, again relative to the set of supplied training objects. This setup is useful in the case where pre-defined objects are to be located and extracted (through a classification algorithm) from an image of interest. The objective selection of a segmentation result (*i.e.* not based on "expert opinion," "visual interpretation" and the like) necessitates such an approach. Additionally, the growing supply of segmentation software means that inter-comparisons such as that presented here could benefit from a set of quantitative, well defined measures that communicate the effectiveness of the software to find objects of interest. This paper presents an approach that provides an initial basis for the consistent comparison of segmentations resulting from varying parameters and software.

REFERENCES

Antani, S., D.J. Lee, L.R. Long, G.R. Thoma. 2004. Evaluation of Shape Similarity Measurement Methods for Spine X-ray Images. *Journal of Visual Communication and Image Representation*. Special issue: Multimedia Database Management Systems 15(3): 285-302.

Benz, Ursula C., Peter Hofmann, Gregor Willhauck, Iris Lingenfelder, Markus Heynen. 2004. Multi-resolution, object-oriented fuzzy analysis of remote sensing data for GIS-ready information. *ISPRS Journal of Photogrammetry & Remote Sensing*. 58: 239–258.

Bian, Ling. 2007. Object-Oriented Representation of Environmental Phenomena: Is Everything Best Represented as an Object? *Annals of the Association of American Geographers*. 97(2)267-281.

Borsotti, M., P. Campadelli, R. Schettini. 1998. Quantitative evaluation of color image segmentation results. *Pattern Recognition Letters*. 19: 741–747

Chabrier, Sebastien, Bruno Emile, Christophe Rosenberger, Helene Laurent. 2006. Unsupervised Performance Evaluation of Image Segmentation. *EURASIP Journal on Applied Signal Processing*. (2006): 1-12

Congalton, Russell G. 1991. A Review of Assessing the Accuracy of Classifications of Remotely Sensed Data. *Remote Sensing of Environmment*. 37: 35-46

Gong, P., and P.J. Howarth, 1990. Land-cover to land-use conversion: A knowledge-based approach, ACSM-ASPRS Annual Convention Proceedings, 18–23 March, Denver, Colorado, 4:pp. 447–456.

Holt, Ashley, E.Y.W Seto, Q. Yu, T. Rivard, P. Gong. *Accepeted*. Optimization of segmentation parameters for object-oriented target detection in remotely sensed imagery. Photogrammetric Endgineering & Remote Sensing.

Krolupper, Filip and Jan Flusser. 2007. Polygonal shape description for recognition of partially occluded objects. *Pattern Recognition Letters*. 28: 1002–1011

Lee, E. T. 1974. The Shape-Oriented Dissimilarity of Polygons and its Application to The Classification of Chromosome Images. *Pattern Recognition*. 6: 47-60

Levine, M.D. and A.M. Nazif. 1982. "An experimental rule based system for testing low level segmentation strategies." *In Multicomputers and Image Processing: Algorithms and Programs*. K. Preston and L. Uhr, Eds. New York: Academic. Pp. 149-160.

Levine, M.D. and A.M. Nazif. 1985. Dynamic measurement of computer generated image segmentation. *IEEE Transactions on Pattern Recognition and Machine Intelligence*. 7:652-655.

Lu, Cheng-Chang, James George Dunham. 1993. Shape matching using polygon approximation and dynamic alignment. *Pattern Recognition Letters*. 14: 945-949

Moller, M., L. Lymburner, M. Volk. 2007. The comparison index: A tool for assessing the accuracy of image segmentation. *International Journal of Applied Earth Observation and Geoinformation.* 9:311-321.

Ng, W.S., C.K. Lee. 1996. Comment on Using the Uniformity Measure for Performance Measure in Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 18(9): 933-934.

Pichel, Juan C., David E. Singh, Francisco F. Rivera. 2006. Image segmentation based on merging of sub-optimal segmentations. *Pattern Recognition Letters* 27: 1105–1116

Stehman, Stephen V. 1999. Basic probability sampling designs for thematic map accuracy assessment. *International Journal of Remote Sensing*. 20(12): 2423-2441.

Stehman, Stephen V. and Raymond L. Czaplewski. 1998. Design and Analysis for Thematic Map Accuracy Assessment: Fundamental Principles. *Remote Sensing of Environment*. 64: 331-344.

Yang, Luren, Fritz Albregtsen, Tor Lønnestad, Per Grøttum. 1995. A Supervised Approach to the Evaluation of Image Segmentation Methods, Proceedings of 6th International Conference: *Computer Analysis of Images and Patterns*, Prague, Czech Republic, September 6–8, 1995. PP. 759-765.

Zhang, Y.J. 1996. A survey on evaluation methods for image segmentation. *Pattern Recognition*, 29(8): 1335 – 1346.