Accuracy Assessment Measures for Object-based Image Segmentation Goodness

Nicholas Clinton, Ashley Holt, James Scarborough, Li Yan, and Peng Gong

Abstract

To select an image segmentation from sets of segmentation results, measures for ranking the segmentations relative to a set of reference objects are needed. We review selected vector-based measures designed to compare the results of object-based image segmentation with sets of training objects extracted from the image of interest. We describe and compare area-based and location-based measures that measure the shape similarity between segments and training objects. By implementing the measures in two object-based image processing software packages, we illustrate their use in terms of automatically identifying parsimonious parameter combinations from arbitrarily large sets of segmentation results. The results show that the measures have divergent performance in terms of the identification of parameter combinations. Clustering of the results in measure space narrows the search. We illustrate combination schemes for the measures for generating rankings of segmentation results. The ranked segmentation results are illustrated and described.

Introduction

In object-based image processing, the first step is generally to segment 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 compare well defined measures that can be used in the identification of a particular segmentation result and objects within that segmentation that are suitable for training a classifier. These measures are applicable in the supervised setting only, and the choice of a segmentation is therefore relative to a set of predefined training objects (assumed polygons) over the image of interest. The supervised approach has both advantages and disadvantages. The advantage is that accuracy is determined according to what a human has determined to be of interest. This enables the determination of segmentation accuracy relative to any set of objects that are deemed important. The disadvantage is that different humans ascribe different importance levels to different sets of objects (Martin *et al.*, 2001). For this reason, we do not feel it is appropriate to describe a segmentation result as correct or incorrect. However, it is important to describe the segmentation in terms of how well it extracts sets of objects of interest, a property we call *goodness.*

Traditional pixel-counting-based approaches to accuracy assessment (e.g., Congalton, 1991) are insufficient for the object-based image processing paradigm. In many segmentation-based studies, relatively little attention has been given to the accuracy with which image segmentation extracts the shapes of real objects (Fortin *et al.*, 2000; Radoux and Defourny, 2007). Under the assumption 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). Representation of the objects in the segmentation is important, since this shape information will eventually be presented to a classifier to identify a pattern used for object labeling. The accuracy of the classification is thus dependent (in part) 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 compare are *not* measures of classification accuracy, but are related. If a probability sample is obtained on the population of objects (Stehman and Czaplewski, 1998; Stehman, 1999) and is used to generate accuracy statistics such as a confusion matrix, then the accuracy of the shapes has been completely ignored. On the other hand, if a sample is taken directly from the landscape (e.g., human delineated training polygons are used) and compared to the segments, then the 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 the classification is performed on object primitives, a different problem discussed in the *Object Hierarchies* sub-section.

Nicholas Clinton and Peng Gong are with the Department of Environmental Science, Policy and Management, University of California, Berkeley, CA 94720, the 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, and Berkeley Environmental Technology International, LLC, 3015 Holyrood Dr., Oakland, CA 94611 (nicholas.clinton@gmail.com).

Ashley Holt is with the Department of Environmental Science, Policy and Management, University of California, Berkeley, CA 94720.

James Scarborough is with Berkeley Environmental Technology International, LLC, 3015 Holyrood Dr., Oakland, CA 94611.

Li Yan is with the International Institute for Earth System Science, Nanjing University, China. 210093.

Photogrammetric Engineering & Remote Sensing
Vol. 76, No. 3, March 2010, pp. 289–299.

0099-1112/10/7603–0289/$3.00/0

The objectives of this paper are to implement and compare a variety of segmentation goodness measures, and assess the efficacy of these measures for choosing from a large set of candidate segmentations. The candidate segmentations may result from different parameter combinations, algorithms, or software. First, in methods, we give a precise definition of each segmentation goodness measure we used, followed by a description of the input imagery and the segmentation software. In the next sub-section, we describe how the measures are computed on the segmentation results, followed by a description of a clustering method on the segmentation results and goodness measures. The Results section describes the values of the measures when applied to test segmentations, followed by the Results and some Conclusions.

Methods

Segmentation Goodness Measures

There are a large number of methods used 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 predefined 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 with respect to the shape of training objects. In this context, a segmentation result should contain segments that match the training objects. The problem is, therefore, one of computing the similarity between polygons. A large body of research exists in the computer vision and medical imaging fields for comparing shapes (Lee, 1974; Arkin *et al.*, 1991, Lu and Dunham, 1993; Antani *et al.*, 2004; Krolupper and Flusser, 2007). In these studies, general forms of objects are assumed known and segments are compared to the training objects assuming that the segments could represent any affine transform of the training object shapes. If invariance to affine transformations is not necessary (when each training object is specified individually), there are several intuitive and easy-to-compute measures of polygon matching.

The measures of polygon matching we consider were chosen based on criteria involving vector representation, ease of implementation, and capacity to identify the segments that match the training shapes well. For the vector representation criterion, we make the requirement that the measure can be computed without using any pixel information. One motivation for performing image segmentation is that the pixel does not suitably represent the phenomena under study; therefore, it makes little sense to continue to use the pixel as a unit of analysis. Additionally, shape metrics can be developed independent of pixel scale. The ease of implementation criterion is simply that the metric can be implemented according to OpenGIS standards (**http://www.opengeospatial.org/standards**) without relying on shape representations that are less accessible due to complexity, proprietary software, or other computational requirements. The last criterion is important for choosing accuracy metrics that aid in the identification of segments useful for training a classifier, rather than merely reporting the global performance on a set of reference objects.

Various accuracy metrics are proposed by Levine and Nazif (1982), Delves *et al.* (1992), Yang *et al* (1995), Lucieer and Stein (2002), Prieto and Allen (2003), Zhan *et al.* (2005), Möller *et al.* (2007), Unnikrishnan *et al.* (2007), and Weidner (2008). The metrics described in Delves *et al.* (1992), Prieto and Allen (2003), and Unnikrishnan *et al.* (2007) made use of pixel information or post-classification labeling and were therefore not chosen for this re-implementation and comparison. Despite the prodigious use of new notation in these studies, they are all describing similar aspects of the correspondence between reference objects and segments. The first aspect is the difference in area between reference objects and the segments they intersect. The second is the positional difference between reference objects and segments. Before delving into specifics, we assume that ideally, there should be a one-to-one correspondence between human identified objects (*reference* or *training* objects) and segments. That is, the difference in area and the distance between a reference object and a segment should both be zero.

For the purposes of describing the metrics, some notation is necessary. Let $X = \{x_i : i = 1 \ldots n\}$ be the set of n training objects, assumed polygons, relative to which the segmentation is to be judged. Let $Y = \{y_j : j = 1 \ldots m\}$ be the set of all segments in the segmentation of an image having p pixels. 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 '.'; let $\tilde{Y}_i$ be a subset of Y such that:

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

Thus, $\tilde{Y}_i$ is the set of all y_j that intersect reference object x_i. For each training object x_i, the following subsets of $\tilde{Y}_i$ exist:

$$Ya_i = \{y_j : \text{the centroid of } x_i \text{ is in } y_j\}$$
$$Yb_i = \{y_j : \text{the centroid of } y_j \text{ is in } x_i\}$$
$$Yc_i = \{y_j : area(x_i \cap y_j) / area(y_j) > 0.5\}$$
$$Yd_i = \{y_j : area(x_i \cap y_j) / area(x_i) > 0.5\}.$$

The union of these subsets is the subset $Y_i^* = Ya_i \cup Yb_i \cup Yc_i \cup Yd_i$ where we assume Y_i^* to be the subset of segments that are *relevant* to training object x_i. With the exception mentioned in the following section, we evaluate all the measures on the Y_i^* set. This refinement was designed to eliminate spurious effects caused by intersections that represent a small proportion of the reference object or the segment. Zhan *et al.* (2005) have a similar "matching" requirement, though other studies do not specify (unambiguously) the subset of the segmentation to be used for comparison to a given reference object.

Area-based Measures

Levine and Nazif (1982) propose a measure of area correspondence. In their setup, X is a complete partition of the input image. Due to the fact that different analysts will partition an image differently (Martin *et al.*, 2001), we do not feel this is a useful setup when the reference objects comprise a small subset of a complete partition of the image. Yang *et al.* (1995), modify the Levine and Nazif (1982) metrics to function with a proper subset of a complete reference partition (rather than using a complete reference segmentation of the image of interest). Yang *et al.* (1995) define the following two metrics:

$$underMerging_{ij} = \frac{(area(x_i) - area(x_i \cap y_j))}{area(x_i)}, y_j \in Y_i^*.$$

$$overMerging_{ij} = \frac{(area(y_j) - area(x_i \cap y_j))}{area(x_i)}, y_j \in Y_i^*.$$

Observe that these metrics are ideally zero. These metrics were summed over the $y_j \in Y_i^*$, in keeping with the procedure originally outlined by Levine and Nazif (1982).

Lucieer and Stein (2002) define another area-based metric, the Area Fit Index (AFI), as:

$$AFI_i = \frac{area(x_i) - area(y_{iMax})}{area(x_i)},$$

where y_{iMax} is the $y_j \in \widetilde{Y}_i$ with the largest area. AFI is also ideally zero.

Zhan *et al.* (2005) define SimSize as:

$$SimSize_{ij} = \frac{\min(area(x_i), area(y_j))}{\max(area(x_i), area(y_j))}, y_j \in Y_i^*$$

and propose that the mean and standard deviation of *SimSize* are area-based metrics of segmentation goodness. The average of SimSize is in [0,1] with one being ideal.

Möller *et al.* (2007) describe the Relative Area (RA) metric:

$$RA\text{sub}_{ij} = \frac{area(x_i \cap y_j)}{area(x_i)}, y_j \in \widetilde{Y}_i$$

$$RA\text{super}_{ij} = \frac{area(x_i \cap y_j)}{area(y_j)}, y_j \in \widetilde{Y}_i.$$

Möller *et al.* (2007) note that *RA*sub is a measure of *over-segmentation* and *RA*super is a measure of *under-segmentation.* Observe that these metrics are continuous in [0,1] with 1 being an ideal segmentation. These are the only metrics we evaluated on the $\widetilde{Y}_i$ set.

Weidner (2008) describes the *quality rate* (*qr*) as:

$$qr_{ij} = 1 - \frac{area(x_i \cap y_j)}{area(x_i \cup y_j)}, y_j \in Y_i^*.$$

The *qr* is also in [0,1], and is the only goodness measure we evaluated that takes the rate of false positive into consideration, that is evaluating the amount of the "miss," in addition to the "hit."

We evaluated the following modification of the RA metrics:

$$OverSegmentation_{ij} = 1 - \frac{area(x_i \cap y_j)}{area(x_i)}, y_j \in Y_i^*$$

$$UnderSegmentation_{ij} = 1 - \frac{area(x_i \cap y_j)}{area(x_i)}, y_j \in Y_i^*.$$

OverSegmentation and *UnderSegmentation* are in [0,1], where *OverSegmentation* = 0 and *UnderSegmentation* = 0 define a *perfect* segmentation, where the segments match the training objects exactly. The purpose of this modification of the RA measures was merely to evaluate the metrics over the Y^* set (comparing to the original RA measures) and to rescale the measures.

Location-based Measures

Lucieer and Stein (2002) propose a distance-based measure of segmentation goodness called *D(b).* This is a planimetric measure, based on the distance between boundary pixels in the reference object and in the segments. To compute this measure using vectors, we defined the mod$D(b)_i$ as the mean distance between each vertex in the reference polygon and the closest vertex in every Y^* segment. In general, a lower mod$D(b)_i$ should indicate segmentation goodness.

Zhan *et al.* (2005) define the following distance metric:

$$qLoc_{ij} = \text{dist}(\text{centroid}(x_i), \text{centroid}(y_j)), y_j \in Y_i^*$$

where dist() denotes Euclidean distance in the *xy* plane. Zhan *et al.* (2005) propose the mean and standard deviation of *qLoc* as distance-based measures of segmentation goodness. The range of *qLoc* depends on the input image, but low values are generally preferable.

Möller *et al.* (2007) propose a similar metric called Relative Position (RP). They define two versions, one of which is identical to *qLoc*:

$$RPsub_{ij} = qLoc_{ij}, y_j \in \widetilde{Y}_i$$

$$RPsuper_{ij} = \frac{\text{dist}(\text{centroid}(x_i),\text{centroid}(y_j))}{\text{dist}_{\max}},$$

where $\text{dist}_{\max} = \max_j(RPsub_{ij})$, $y_j \in Y_i^*$. *RPsuper* is therefore in [0,1] with lower values being preferable.

Observe that metrics indexed by *i* and *j* are properties of the segments. When computing averages of these measures for a given segmentation and set of reference objects, the mean can be computed over all *i* and all *j* such that $y_j \in Y_i^*$. 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 (indexed by *i* only) first averages 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.*

Combined Measures

It would be nice to find a way to utilize the information provided by all the measures by combining the scores into a single ranking. Möller *et al.* (2007) describe a "ranking" that is indicated by low values of RP and high values of RA. This ranking can be used to generate a "Comparison Index" that can be used to identify parsimonious parameter combinations. Similarly, Lucieer and Stein (2002) claim that: "A reference object is over-segmented if the overlap is less than 100 % and $AFI > 0$. A reference object is under-segmented if the overlap is 100% and $AFI < 0$," where overlap has been defined relative to the largest intersecting segment. We defined Boolean properties of the training objects according to this definition and counted the number of training objects that were over- and under-segmented ($countOver_i$ and $countUnder_i$, respectively) at different parameter combinations, with the minimum assumed to be superior.

There is a wide variety of other methods by which to combine the measures, the most simple of which is perhaps the root mean square (or RMS, suggested by Levine and Nazif (1982) and Weidner (2008)):

$$D_{ij} = \sqrt{\frac{OverSegmentation_{ij}^2 + UnderSegmentation_{ij}^2}{2}}.$$

This index *D* should be interpreted as the "closeness" to an ideal segmentation result, in relation to a predefined training set. In this context, *D* is in [0, 1]. RMS measures are appropriate in this circumstance, but many of the metrics do not have well defined ranges, or ranges that differ substantially from one another. In these cases, some normalization or standardization is required prior to combining them for the purposes of sorting parameter combinations by goodness. In addition to D_{ij}, for each parameter set, we evaluated the following combination

schemes on various subsets of the measures, subsets inspired by the articles in which the measures were proposed (measures indexed by *i* and *j* are weighted, measures indexed by *i* are un-weighted and measures prefixed by the *n* subscript are normalized to [0,1], by dividing by the maximum):

$$D_i = \sqrt{\frac{OverSegmentation_i^2 + UnderSegmentation_i^2}{2}}$$

$$LS_i = \sqrt{\frac{{}_n|AFI_i|^2 + {}_n\mathrm{mod}Db_i^2}{2}}$$

$$OverUnder_i = \sum(countOver_i + countUnder_i)$$

$$mergeSum_i = \sum(overMerging_i + underMerging_i).$$

The *M* combinations are inspired by Moller *et al.* (2007). The base measures that make up the combined measures are normalized and/or the scale reversed in order make the measures on a [0,1] scale with 0 optimal:

$$M_{ij} = \sqrt{\frac{(1 - RA\mathrm{sub}_{ij})^2 + (1 - RA\mathrm{super}_{ij})^2 + {}_nRP\mathrm{sub}_{ij}^2 + {}_nRP\mathrm{super}_{ij}^2}{4}}.$$

$$M_i = \sqrt{\frac{(1 - RA\mathrm{sub}_i)^2 + (1 - RA\mathrm{super}_i)^2 + {}_nRP\mathrm{sub}_i^2 + {}_nRP\mathrm{super}_i^2}{4}}.$$

The *ZH* combinations are inspired by Zhan *et al.* (2005). As with the *M* combinations, the base measures have been normalized and/or subtracted from 1 as necessary.

$$ZH1_{ij} = \sqrt{\frac{(1 - SimSize_{ij})^2 + {}_n\mathrm{SD}\_SimSize_{ij}^2 + {}_nqLoc_{ij}^2 + {}_n\mathrm{SD}\_qLoc_{ij}^2}{4}}$$

$$ZH1_i = \sqrt{\frac{(1 - SimSize_i)^2 + {}_n\mathrm{SD}\_SimSize_i^2 + {}_nqLoc_i^2 + {}_n\mathrm{SD}\_qLoc_i^2}{4}}$$

$$ZH2_{ij} = \sqrt{\frac{(1 - SimSize_{ij})^2 + {}_nqLoc_{ij}^2}{2}}$$

$$ZH2_i = \sqrt{\frac{(1 - SimSize_i)^2 + {}_nqLoc_i^2}{2}}.$$

Additionally, all the measures were normalized to [0,1] (with 0 optimal) as necessary, then the RMS computed. This aggregate measure is denoted *Combo*. Both weighted and unweighted *qr* measures were included independently. Note that, with the exception of *Combo*, the combination schemes presented here are based on authorship. This setup was envisioned as a sort of proposed measure shootout.

Table 1 summarizes each measure by its practical range and optimum. We determined the practical range according to two degenerative scenarios. The first is total over-segmentation, where there is one training object that corresponds to

TABLE 1. SUMMARY OF THE MEASURES DESCRIBED IN THE METHODS SECTION. THE MINIMA AND MAXIMA ARE BASED ON TOTAL OVER-SEGMENTATION AND TOTAL UNDER-SEGMENTATION FOR AN IMAGE OF *P* PIXELS AND A TRAINING SET OF *M* POLYGONS.

Measure	Minimum	Maximum	Optimum
underMerging	0	p-1	0. The minimum indicates a perfect match or overMerging. Summed over training objects.
overMerging	0	p(p-1)	0. The minimum indicates a perfect match or underMerging. Summed over training objects.
AFI	1-p	(p-1)/p	0. AFI<0 indicates undersegmentation and AFI>0 is oversegmentation.
SimSize	1/p	1	1.0 Approaches 1/p for underSegmentation and overSegmentation
RAsub	1/p	1	1.0 Approaches 1/p for overSegmentation.
RAsuper	1/p	1	1.0 Approaches 1/p for underSegmentation.
QualityRate	1/p	1	1.0 Approaches 1/p for underSegmentation and overSegmentation
overSegmentation	0	(p-1)/p	0. Minimum indicates a perfect match or underSegmentation.
underSegmentation	0	(p-1)/p	0. Minimum indicates a perfect match or overSegmentation.
modDb	0	O(P)	0. Increasing distance indicates worse matching. Arbitrarily large with increasing resolution.
qLoc	0	mean distance to nadir	0. Increasing distance indicates worse matching
RPsub	0	mean distance to nadir	0. Increasing distance indicates worse matching
RPsuper	0	1	0. Increasing distance indicates worse matching
D	0	(p-1)/p($2^{1/2}$)	0. Minimum indicates a perfect match.
LS	0	O(P)	0. Minimum indicates a perfect match. Can get arbitrarily large with increasing resolution.
OverUnder	0	2m	0. Minimum indicates a perfect match.
mergeSum	0	p^2-1	0. Minimum indicates a perfect match.
M	0	1	0. Minimum indicates a perfect match.
ZH1	0	1	0. Minimum indicates a perfect match.
ZH2	0	1	0. Minimum indicates a perfect match.
Combo	0	1	0. Minimum indicates a perfect match.

the image area and each pixel is a segment. The second is total under-segmentation, where each pixel is a training object, and there is a single segment that corresponds to the image area.

Imagery and Segmentation Software

We used the measures to evaluate sets of segmentation results from an urban image. The imagery we used is a 3-band (RGB) aerial image of a portion of the City of San Francisco, California. The image was re-sampled (using nearest neighbor) from slightly rectangular pixels to have a resolution of approximately 0.17 meters, square. The input image is shown in Plate 1.

We obtained segmentations from two different software packages: eCognition® (**http://www.definiens.com**) and BerkeleyImageSeg (BIS; **http://www.imageseg.com**). These programs use a region merging technique to obtain a complete spatial partition of the input image pixels. Both BIS and eCognition® are developed based on the region merging algorithms described in Benz *et al.* (2004). Differences in results between BIS and eCognition® are likely due to propriety implementation details that are impossible to evaluate from closed source software. Following is a brief summary of the region merging technique.

Initially, every pixel is an object and merging proceeds iteratively. For any object, consider its contiguous neighbors. Let any pair of contiguous objects be described as object a, object b, and their possible union ab as the merged object.

Let the difference in spectral heterogeneity h_p for the merged object by defined as:

$$\Delta h_p = \sum_I w_i(n_{ab}\sigma_{i,ab} - (n_a\sigma_{i,a} + n_b\sigma_{i,b})),$$

$0 < w_i \leq 1, \sum_I w_i = 1$ are the weights for $i = 0, 1 \ldots I$ image bands, n denotes the area of an object in pixels, and σ_i is the standard deviation in band i for an object.

Let the difference of compactness and smoothness, Δh_c and Δh_s, respectively, of the objects be defined as:

$$\Delta h_c = \frac{n_{ab}l_{ab}}{\sqrt{n_{ab}}} - \frac{n_a l_a}{\sqrt{n_a}} - \frac{n_b l_b}{\sqrt{n_b}}$$

$$\Delta h_s = \frac{n_{ab}l_{ab}}{b_{ab}} - \frac{n_a l_a}{b_a} - \frac{n_b l_b}{b_b}$$

where l is object perimeter length, and b is the perimeter of the object's bounding box length.

Let the difference in shape heterogeneity be defined as:

$$\Delta h_t = w_c\Delta h_c + w_s\Delta h_s,$$

$0 < w_c, w_s \leq 1, w_c + w_s = 1$, and w_c is the user selected compactness parameter (w_s is the smoothness parameter).

Define the scale rate r as:

$$r = w_p\Delta h_p + w_t\Delta h_t,$$

$0 < w_p, w_t \leq 1, w_p + w_t = 1$, and w_t is the user selected shape parameter.

The program will iterate through the objects, merging contiguous objects if the estimated scale rate r is below

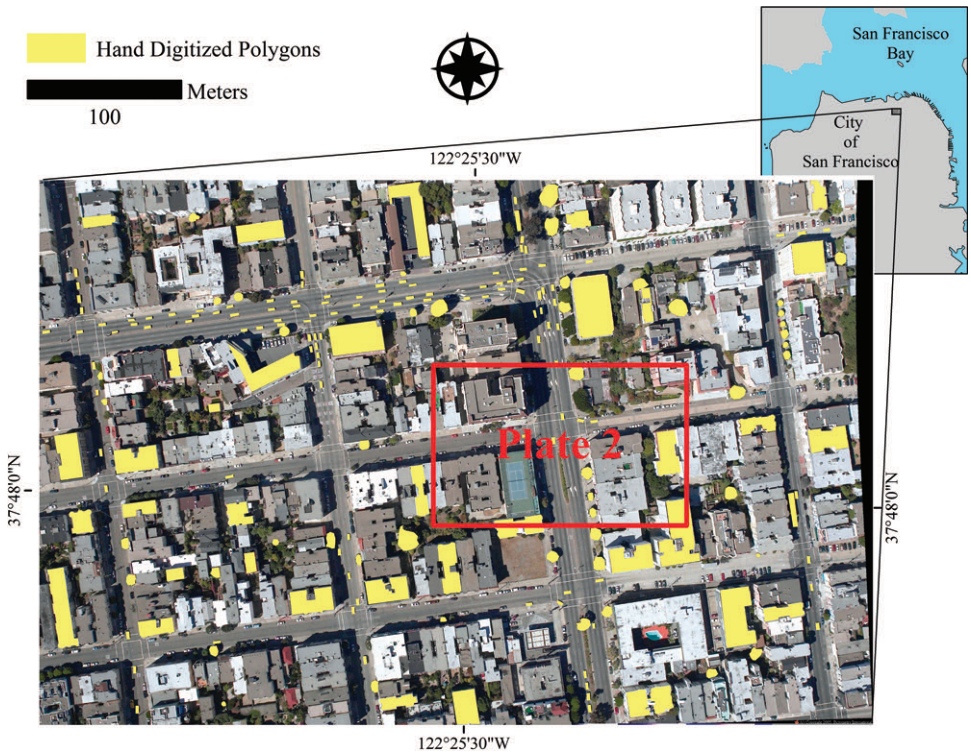

Plate 1. The input image overlaid by the 203 hand digitized training shapes of cars, trees, and buildings. The inset shows the location of the image relative to the City of San Francisco; the extent of the subset image shown in Plate 1 appears in red and is the base image for Plate 2.

$t = 0,1,2 \ldots T$. The user selected scale threshold parameter T determines the number of iterations through the objects or the number of merging cycles. So the higher the scale, the more merging will occur and the larger the objects in the segmentation will be.

Computation of Measures

The software packages perform segmentation and export the results as polygons in the ESRI shapefile format. In total, 150 parameter combinations were examined for scale threshold, shape, and compactness according to {10, 20, 30, 40, 50, 60} × {0.1, 0.3, 0.5, 0.7, 0.9} × {0.1, 0.3, 0.5, 0.7, 0.9}, respectively. 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. The digitized training shapes are shown in Plate 1.

Using the resultant shapefile from each parameter combination, we implemented the measures described above in the Java environment using open source libraries JTS (Java Topology Suite; **http://www.vividsolutions.com/jts/jtshome.htm**) and GeoTools (**http://geotools.codehaus.org/**). We computed the various goodness measures described above for each parameter combination relative to each training object set (vehicles, trees, buildings) and the merged training objects (union of vehicles, trees, buildings).

Clustering

An alternative to mathematically combining the measures (as previously described) is to search for clusters in the measure space (Moller *et al.*, 2007). Under this set up, a segmentation (resulting from a specific parameter combination) is generated and goodness measures computed in relation to some set of training objects. That segmentation represents a point in the space of the goodness measures. We expected segmentations that optimize one or more of the goodness measures to be clustered in this space.

We tested clustering using the open source data mining software Weka's (**http://www.cs.waikato.ac.nz/ml/weka/**) expectation maximization algorithm (Witten and Frank, 2005). Only the base measures were used as features in the clustering, each instance representing a segmentation (parameter combination). Once cluster membership was assigned to each segmentation, we examined each cluster to see how many segmentations it contained that optimized one of the base measures or one of the combined measures. The complete process is diagrammed in Figure 1.

Whichever cluster has the most number of segmentations that are optimal relative to one or more of the combined measures is a "winning" cluster. Ideally, a winning cluster is obvious (i.e., there is not a tie with other clusters for the most number of optimized measures), and all the measures are optimized by the same parameter combination. Alternatively, the measures may be optimized by disparate parameter combinations, but the winning cluster may indicate useful patterns in the parameters (such as nesting at different scales). This method has the nice property of being able to identify the mean of scale, shape, and compactness in winning clusters.

Results

Counter to what we expected, and perhaps in reflection of an unjustified optimism, a simple summary of the results is that the different measures can indicate wildly different parameter combinations as "best." However, this should not be surprising, since, as Weidner (2008) noted, "All quantities evaluate just one aspect of segmentation properties at once." This observation is clearly illustrated in the results, shown in Tables 2 and 3, according to BIS and eCognition® segmentations, respectively. Tables 2 and 3 show the optimal values for each measure, the parameter combination (segmentation) that produced the optimal

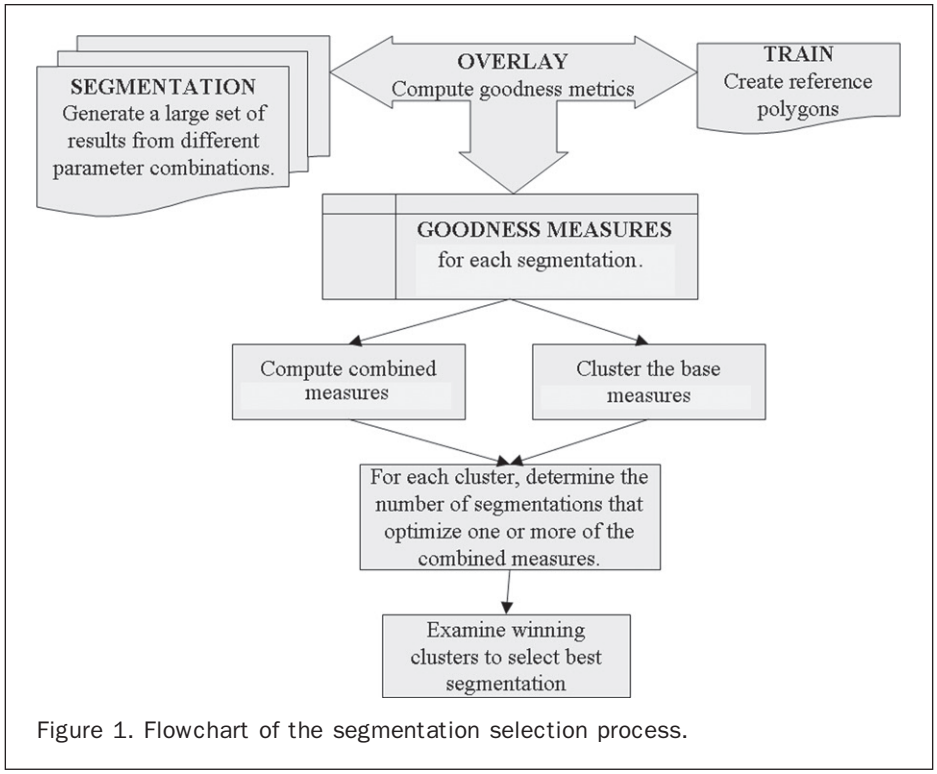

Figure 1. Flowchart of the segmentation selection process.

TABLE 2. BIS GOODNESS MEASURES FOR (A) MERGED TRAINING OBJECTS, (B) CARS, (C) BUILDINGS, AND (D) TREES ARRANGED BY CLUSTER MEMBERSHIP (COLUMN 1). COLUMNS 2 AND 3 SHOW THE MEASURE AND THE OPTIMAL VALUE, RESPECTIVELY, FOR SEGMENTATION CREATED WITH THE PARAMETERS IN COLUMNS 3, 4, AND 5 (ABBREVIATED BY SC. = SCALE, SH. = SHAPE, CPT. = COMPACTNESS). THE CLUSTERS CONTAINING THE LARGEST NUMBER OF MEASURES WITH OPTIMAL VALUES ARE IDENTIFIED BY A "□" BOXED OUTLINE

a. Merged

clus.	measure	value	sc.	sh.	cpt.
0	D_{ij}	0.437	40	0.9	0.1
0	*ModDb*	4.501	60	0.9	0.1
0	*OverUnder*	64	50	0.9	0.5
0	$SimSize_{ij}$	0.167	40	0.9	0.1
0	$SimSizeSD_i$	0.041	60	0.9	0.1
0	$QLocSD_i$	3.266	60	0.9	0.7
0	$QlocSD_{ij}$	14.413	50	0.9	0.7
1	*abs(AFI)*	0.008	20	0.3	0.5
2	D_i	0.378	30	0.9	0.1
2	RP_{ij}	5.337	50	0.7	0.9
2	$QLoq_{ij}$	7.540	50	0.7	0.9
2	qr_i	0.623	30	0.9	0.1
2	qr_{ij}	0.318	60	0.7	0.5
2	$ZH2_{ij}$	0.281	50	0.7	0.9
3	M_{ij}	0.449	50	0.5	0.7
3	*Combo*	0.440	30	0.9	0.5
4	RP_i	5.921	20	0.7	0.7
4	$Qloq_i$	8.346	20	0.7	0.7
4	M_i	0.428	30	0.1	0.9
5	RA_i	0.456	10	0.1	0.7
5	RA_{ij}	0.490	10	0.1	0.7
5	$SimSizeSD_{ij}$	0.054	10	0.1	0.7
5	*LS*	0.007	10	0.1	0.3
5	$ZH1_i$	0.299	10	0.1	0.7
5	$ZH1_{ij}$	0.430	10	0.1	0.1
5	$ZH2_i$	0.153	10	0.1	0.3
7	$SimSize_i$	0.482	20	0.9	0.1
7	*mergeSum*	973.763	30	0.7	0.9

b. Cars

clus.	measure	value	sc.	sh.	cpt.
0	RA_i	0.459	60	0.9	0.1
0	$SimSizeSD_i$	0.022	60	0.9	0.1
2	D_{ij}	0.418	30	0.9	0.1
2	*OverUnder*	33	40	0.9	0.5
3	D_i	0.370	60	0.5	0.1
3	*ModDb*	2.311	60	0.5	0.3
3	$SimSize_i$	0.519	20	0.9	0.1
3	$SimSize_{ij}$	0.208	60	0.5	0.5
3	qr_i	0.596	60	0.1	0.3
3	*Combo*	0.405	30	0.9	0.5
4	*abs(AFI)*	0.004	10	0.9	0.3
4	RP_i	2.804	20	0.7	0.3
4	$Qloq_i$	3.904	20	0.7	0.3
4	M_i	0.412	20	0.5	0.9
5	RA_{ij}	0.451	10	0.1	0.1
5	$SimSizeSD_{ij}$	0.080	10	0.1	0.7
5	*LS*	0.004	10	0.1	0.7
5	$ZH1_i$	0.219	10	0.1	0.7
5	$ZH1_{ij}$	0.142	10	0.1	0.7
5	$ZH2_i$	0.083	10	0.1	0.3
5	$ZH2_{ij}$	0.080	10	0.1	0.9
6	RP_{ij}	1.453	20	0.9	0.7
6	$QLoq_{ij}$	2.030	20	0.9	0.7
6	$QLocSD_i$	1.749	20	0.9	0.7
6	$QLocSD_{ij}$	3.514	20	0.9	0.7
6	*mergeSum*	332.340	20	0.9	0.7
6	qr_{ij}	0.307	20	0.9	0.7
6	M_{ij}	0.345	20	0.9	0.7

c. Buildings

clus.	measure	value	sc.	sh.	cpt.
0	D_i	0.336	60	0.9	0.1
0	D_{ij}	0.410	60	0.9	0.1
0	*ModDb*	9.239	60	0.9	0.1
0	*OverUnder*	9	60	0.9	0.1
0	RP_i	12.602	50	0.9	0.1
0	RP_{ij}	6.006	60	0.9	0.1
0	$SimSize_i$	0.575	60	0.9	0.1
0	$SimSize_{ij}$	0.197	60	0.9	0.1
0	$Qloq_i$	17.807	50	0.9	0.1
0	$QLoq_{ij}$	8.489	60	0.9	0.1
0	$QLocSD_i$	7.402	60	0.9	0.7
0	$QLocSD_{ij}$	16.510	60	0.9	0.5
0	*mergeSum*	74.096	60	0.9	0.1
0	qr_i	0.523	60	0.9	0.1
0	qr_{ij}	0.247	60	0.9	0.1
0	M_{ij}	0.399	60	0.9	0.1
0	$ZH2_{ij}$	0.356	60	0.9	0.1
2	*abs(AFI)*	0.004	50	0.9	0.7
2	M_i	0.574	30	0.9	0.1
2	$ZH1_{ij}$	0.589	50	0.9	0.7
2	$ZH2_i$	0.580	30	0.9	0.1
2	*Combo*	0.535	50	0.9	0.7
4	RA_i	0.535	10	0.9	0.7
4	RA_{ij}	0.536	10	0.9	0.7
4	$SimSizeSD_i$	0.023	10	0.9	0.7
4	$ZH1_i$	0.616	10	0.9	0.9
5	$SimSizeSD_{ij}$	0.020	10	0.1	0.7
6	*LS*	0.202	50	0.1	0.1

d. Trees

clus.	measure	value	sc.	sh.	cpt.
0	*ModDb*	1.629	50	0.7	0.7
0	*OverUnder*	4	60	0.3	0.5
2	$SimSizeSD_i$	0.000	60	0.9	0.1
2	$QLocSD_i$	0.000	60	0.9	0.1
3	M_i	0.403	30	0.1	0.9
4	D_i	0.314	30	0.9	0.9
4	RP_i	4.052	20	0.9	0.3
4	RP_{ij}	1.384	30	0.7	0.7
4	$SimSize_i$	0.613	30	0.9	0.9
4	$SimSize_{ij}$	0.172	30	0.9	0.9
4	$Qloq_i$	5.666	20	0.9	0.3
4	$QLoq_{ij}$	1.946	30	0.7	0.7
4	$QLocSD_{ij}$	4.315	30	0.7	0.7
4	*mergeSum*	71.068	30	0.7	0.7
4	M_{ij}	0.292	30	0.7	0.7
4	$ZH2_{ij}$	0.146	30	0.7	0.7
4	*Combo*	0.406	20	0.9	0.7
5	*abs(AFI)*	0.015	30	0.3	0.5
6	D_{ij}	0.339	30	0.9	0.3
6	qr_i	0.523	30	0.9	0.3
6	qr_{ij}	0.150	40	0.7	0.1
7	RA_i	0.430	10	0.1	0.1
7	RA_{ij}	0.439	10	0.1	0.1
7	$SimSizeSD_{ij}$	0.065	10	0.1	0.7
7	*LS*	0.023	10	0.1	0.9
7	$ZH1_i$	0.244	10	0.1	0.3
7	$ZH1_{ij}$	0.298	10	0.1	0.3
7	$ZH2_i$	0.164	10	0.1	0.5

value, and the cluster membership of the parameter combination. The results have been sorted by cluster membership, with the clusters that contain a majority or plurality of optimal segmentations highlighted. These clusters are useful for noticing the association of particular measures with each other, and also for observing the relationship between segmentation parameters within clusters. In terms of the former (the relationship between measures), it should be noticeable in either the BIS or the eCognition® results that some measures are more frequently represented in the winning cluster (in the case of the merged objects and the cars, for BIS, there was a tie between two clusters for the win). We do not feel it reasonable to make any further observation, aside from this qualitative statement, because it would be unfair to the other measures, which may perform better under other circumstances.

As for the relationship between parameter combinations, the most notable feature of the results is the frequent nesting structure of the parameters within the winning clusters. This result suggests that the measures are converging to parameter sets that are suitable for extracting the objects of interest. For example, in relation to the merged training shapes, the measures identify parameter sets (40, 0.9, 0.1) and (60, 0.9, 0.1) for BIS, and (50, 0.3, 0.9) and (60, 0.3, 0.9) for eCognition®. While the search we evaluate here is fairly coarse, this result indicates particular combinations of shape and compactness that should be subjected to a finer search within ranges of the scale parameter.

Some representative results are displayed in Plate 2. These results were selected from the parameters identified in reference to the cars, trees, and buildings training shapes. They therefore represent *some* of the best results for extracting these objects from the image. We say *some* because, despite our best efforts, there is still some subjectivity left in the process. However, we have chosen from the winning clusters (as shown in Tables 2 and 3) the most representative parameter combinations, meaning most commonly occurring or one of the most commonly occurring. The clustering technique enabled the choice between several segmentations from the winning cluster(s), rather than having to choose from the set of all 150 segmentation results.

Discussion

Classification Accuracy

This paper addresses the first step of object-based image processing which is choosing from a set of candidate segmentations. Using measures such as those we describe is helpful to minimize subjectivity in this choice. Perhaps because of the hitherto subjective nature of this process, the accuracy with which a segmentation represents actual objects in imagery has been largely under-reported in studies that employ object-based image analysis. We claim that the examination of segmentation results relative to training objects is a critical step in this analysis. The first reason is that the spectral and shape characteristics presented to a classifier should be generated from segments that match the objects of interest. If they do not, resultant classification accuracy could be affected. The second reason is to report the accuracy with which the segmentation has captured the objects. It is essential to be able to determine whether inaccuracy in classification is due to a poor classifier or a poor segmentation (or both). Reporting classification accuracy without reporting shape accuracy is ambiguous in this regard.

Parameter Selection

As our results indicate, the complete automation of the parameter selection process is possible, though still subject to some expert judgment, since a measure or set of measures must be chosen and will influence the ultimate selection. In addition, the analyst (or analysts) must still subjectively choose and delineate training objects. However, the process we describe makes it possible to sort a very large collection of segmentation results, and visually interpret a small subset that represents the best outcomes. This type of exercise would be very, very tedious to the point of being infeasible for a human observer. We feel a quantitative approach is greatly needed for the remote sensing community to more objectively choose segmentation results.

The problem of finding an optimal configuration of parameter settings has been addressed by Holt *et al.* (in press) and Möller *et al.* (2007). In our approach, we have conceptualized the issue of finding a parameter combination as more of a search problem than an optimization problem. As such, we would recommend a "grid search" as described by Hsu *et al.* (2008) where a coarse search is initially performed and a fine search is subsequently performed on the parameter combinations identified in winning clusters. Viewed in this way, the measures we describe are basically performance indices that are used to rank or sort a set of results (analogous to a cross-validated accuracy). Our results indicate that, regardless of the measure used, it will probably differ from some other measure published in the literature. For this reason, we recommend the implementation of as many measures as feasible, and the ultimate choice of segmentation justified relative to one or more of these measures.

Choice of Training Objects

Obviously, the choice of training objects influences the results. Different observers will likely choose different objects and manually segment them differently (Martin *et al.*, 2001). It is worth mentioning that some measures take this human variability into consideration explicitly (Unnikrishnan *et al.*, 2007; Martin *et al.*, 2004). We suggest that methods and measures we describe here are equally applicable to multiple observers. This is a strength of the process since segmentation results can be ranked relative to whatever set of objects is considered most important to whichever observer is most important. For example, consider the natural resource application of using image segmentation to partition an image into vegetation types. A wide variety of such vegetation maps are possible from different photo-interpreters or typing rules. The goodness measures can be used to identify desirable segmentation results by comparing the segmentations to vegetation maps resulting from different photo-interpretations or different typing rules.

Object Hierarchies

The evaluation of segmentation relative to a set of training objects 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 set of cars. The individual car objects could be extracted by first classifying car parts such as windshield, hood, roof, etc., then assembling these parts into complete cars through dissolve operations or other adjacency rules. The same approach could be taken for ecosystems composed of objects such as trees, shrubs, meadows, and so forth. The

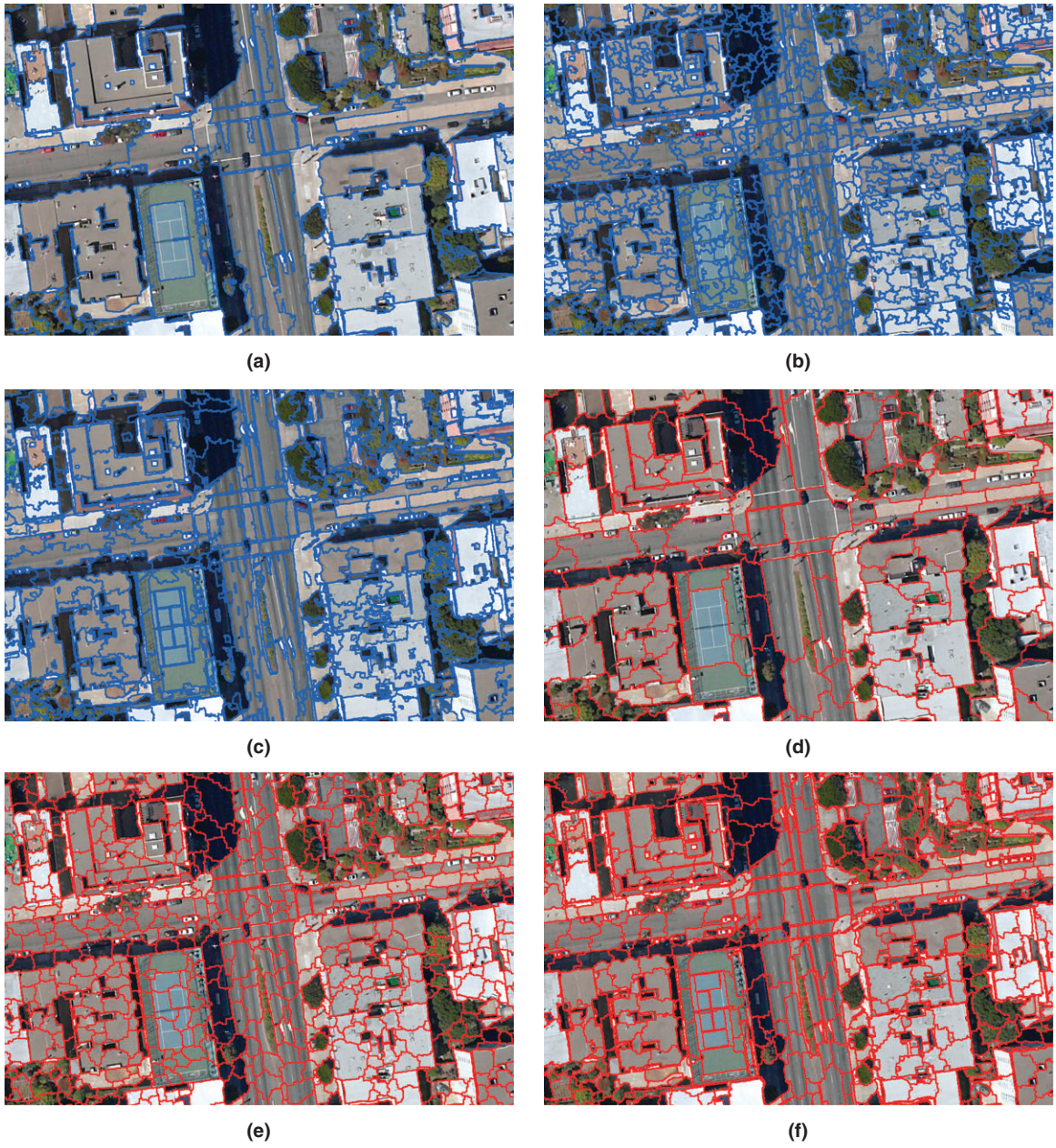

Plate 2. Some illustrations of the segmentation results. Each example was chosen based on frequently occurring parameter combinations in the best cluster for each object type. Each image is identified by the software used to generate it, the reference object set and the parameter combination, respectively. The location of this subset is shown in Plate 1.

method described here could easily be applied to such a scenario through the provision of training sets for the individual parts (windshield, roof, etc.), 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 the measures we propose. With any software that produces nested segmentations at different scales (as both BIS and eCognition® do), the measures 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.

Conclusions

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 predefined

TABLE 3. eCOGNITION® GOODNESS MEASURES FOR (A) MERGED TRAINING OBJECTS, (B) CARS, (C) BUILDINGS, AND (D) TREES ARRANGED BY CLUSTER MEMBERSHIP (COLUMN 1). COLUMNS 2 AND 3 SHOW THE MEASURE AND THE OPTIMAL VALUE, RESPECTIVELY, FOR SEGMENTATION CREATED WITH THE PARAMETERS IN COLUMNS 3, 4, AND 5 (ABBREVIATED BY SC. = SCALE, SH. = SHAPE, CPT. = COMPACTNESS). THE CLUSTERS CONTAINING THE LARGEST NUMBER OF MEASURES WITH OPTIMAL VALUES ARE IDENTIFIED BY A "□" BOXED OUTLINE

a. Merged

clus.	measure	value	sc.	sh.	cpt.
0	D_{ij}	0.405	60	0.3	0.9
0	*ModDb*	5.214	60	0.3	0.1
0	RP_{ij}	4.392	60	0.3	0.5
0	$QLoq_{ij}$	6.204	60	0.3	0.5
0	*mergeSum*	481.226	50	0.3	0.9
0	qr_{ij}	0.249	60	0.3	0.9
0	M_{ij}	0.412	50	0.3	0.9
0	$ZH2_{ij}$	0.426	60	0.3	0.5
0	*Combo*	0.476	50	0.3	0.9
1	D_i	0.311	40	0.3	0.5
1	RP_i	5.119	50	0.5	0.9
1	$SimSize_i$	0.598	50	0.5	0.9
1	$Qloq_i$	7.189	50	0.5	0.9
1	qr_i	0.515	40	0.3	0.5
2	*abs(AFI)*	0	40	0.9	0.5
3	$SimSize_{ij}$	0.154	60	0.9	0.7
6	*OverUnder*	71.000	50	0.1	0.7
6	$SimSizeSD_i$	0.019	60	0.1	0.5
6	$QLocSD_i$	1.335	60	0.1	0.5
6	$QLocSD_{ij}$	12.055	60	0.1	0.5
6	$ZH1_i$	0.386	50	0.1	0.5
6	$ZH1_{ij}$	0.633	60	0.1	0.5
7	RA_i	0.450	10	0.9	0.5
7	RA_{ij}	0.483	10	0.9	0.7
7	$SimSizeSD_{ij}$	0.044	10	0.7	0.9
7	$ZH2_i$	0.299	10	0.5	0.1
9	M_i	0.446	30	0.9	0.5
10	*LS*	0.042	10	0.3	0.9

b. Cars

clus.	measure	value	sc.	sh.	cpt.
0	M_i	0.400	20	0.7	0.5
0	$ZH2_{ij}$	0.132	10	0.1	0.9
1	D_{ij}	0	30	0.1	0.3
1	*ModDb*	1.569	60	0.3	0.1
1	*OverUnder*	26.000	40	0.1	0.9
1	RP_{ij}	0.930	30	0.1	0.7
1	$QLoq_{ij}$	1.288	30	0.1	0.7
1	qr_{ij}	0.164	30	0.1	0.7
1	$ZH1_i$	0.227	40	0.1	0.9
1	*Combo*	0.399	30	0.1	0.9
2	RA_i	0.475	60	0.1	0.3
3	*abs(AFI)*	0.012	30	0.3	0.3
3	RP_i	2.176	30	0.3	0.7
3	$Qloq_i$	2.959	30	0.3	0.5
3	$QLocSD_{ij}$	2.960	20	0.1	0.7
3	*mergeSum*	152.968	20	0.1	0.7
3	M_{ij}	0.272	20	0.1	0.7
4	D_i	0.284	40	0.3	0.5
4	$SimSize_i$	0.655	40	0.3	0.5
4	$SimSize_{ij}$	0.241	50	0.5	0.9
4	qr_i	0.472	40	0.3	0.5
5	$SimSizeSD_i$	0.000	60	0.1	0.1
5	$QLocSD_i$	0.000	60	0.1	0.1
9	RA_{ij}	0.439	10	0.9	0.7
9	$SimSizeSD_{ij}$	0.064	10	0.7	0.9
9	*LS*	0.025	10	0.9	0.7
9	$ZH1_{ij}$	0.197	10	0.7	0.5
9	$ZH2_i$	0.134	10	0.9	0.1

c. Buildings

clus.	measure	value	sc.	sh.	cpt.
0	*Combo*	0.555	50	0.1	0.5
2	M_i	0.587	60	0.9	0.1
2	$ZH2_i$	0.635	60	0.9	0.1
4	$ZH1_{ij}$	0.633	30	0.1	0.7
6	D_i	0.370	60	0.1	0.5
6	D_{ij}	0.462	60	0.1	0.5
6	*ModDb*	16.965	60	0.1	0.1
6	*abs(AF1)*	0.080	60	0.1	0.5
6	*OverUnder*	24.000	60	0.1	0.5
6	RP_i	14.769	60	0.1	0.5
6	RP_{ij}	6.259	60	0.1	0.5
6	$SimSize_i$	0.502	60	0.1	0.3
6	$SimSize_{ij}$	0	60	0.1	0.1
6	$Qloq_i$	20.870	60	0.1	0.5
6	$QLoq_{ij}$	8.848	60	0.1	0.5
6	$QLocSD_i$	7.132	60	0.1	0.5
6	$QLocSD_{ij}$	15.113	60	0.1	0.3
6	*mergeSum*	72.669	60	0.1	0.3
6	qr_i	0.582	60	0.1	0.5
6	qr_{ij}	0.229	60	0.1	0.5
6	*LS*	0.375	60	0.1	0.9
6	M_{ij}	0.377	60	0.1	0.5
6	$ZH1_i$	0.595	60	0.1	0.5
6	$ZH2_{ij}$	0.349	60	0.1	0.5
7	RA_i	0.534	10	0.1	0.9
7	RA_{ij}	0.534	10	0.1	0.9
7	$SimSizeSD_i$	0.010	10	0.1	0.9
7	$SimSizeSD_{ij}$	0.010	10	0.1	0.9

d. Trees

clus.	measure	value	sc.	sh.	cpt.
0	*abs (AF1)*	0.002	40	0.9	0.3
1	RA_i	0.418	10	0.9	0.5
1	RA_{ij}	0.428	10	0.9	0.9
1	$SimSizeSD_{ij}$	0.048	10	0.9	0.3
2	$ZH1_i$	0.415	50	0.3	0.9
3	*OverUnder*	4	60	0.1	0.1
3	$SimSizeSD_i$	0.000	60	0.1	0.5
3	$QLocSD_i$	0.070	60	0.1	0.3
5	$ZH2_i$	0.527	20	0.9	0.3
6	M_i	0.481	30	0.7	0.5
7	D_i	0.204	40	0.5	0.9
7	D_{ij}	0.236	50	0.5	0.9
7	*ModDb*	1.930	60	0.7	0.9
7	RP_i	2.389	40	0.5	0.9
7	RP_{ij}	0.674	50	0.5	0.5
7	$SimSize_i$	0.766	40	0.5	0.9
7	$Qloq_i$	3.248	40	0.5	0.9
7	$QLoq_{ij}$	0.935	50	0.5	0.5
7	$QLocSD_{ij}$	2.750	50	0.5	0.9
7	*mergeSum*	29.534	50	0.5	0.5
7	qr_i	0.338	40	0.5	0.9
7	qr_{ij}	0.080	50	0.5	0.5
7	M_{ij}	0.253	40	0.5	0.5
7	$ZH1_{ij}$	0.543	50	0.5	0.9
7	$ZH2_{ij}$	0.216	50	0.5	0.5
7	*Combo*	0.446	40	0.3	0.7
8	$SimSize_{ij}$	0.183	60	0.9	0.5
9	*LS*	0.157	10	0.3	0.7

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 variety of segmentation methods 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 algorithms. We are hopeful that segmentation software will apply such an approach to assist users in objective parameter selection. Alternatively, there is a need to establish a library of code that can be used by the community at large for judging segmentation results. To this end, we are happy to distribute our test code to any and all interested parties.

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.

Arkin, E.M., L.P. Chew, D.P. Huttenlocher, K. Kedem, and J.S.B. Mitchell, 1991. An efficiently computable metric for comparing polygonal shapes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3).

Benz, U.C., P. Hofmann, G. Willhauck, I. Lingenfelder, and M. 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, L., 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, and R. Schettini, 1998. Quantitative evaluation of color image segmentation results, *Pattern Recognition Letters*, 19:741–747

Chabrier, S., B. Emile, C. Rosenberger, and H. Laurent, 2006. Unsupervised performance evaluation of image segmentation, *EURASIP Journal on Applied Signal Processing*, (2006):1–12

Congalton, R.G., 1991. A review of assessing the accuracy of classifications of remotely sensed data, *Remote Sensing of Environment*, 37:35–46

Delves, L.M., R. Wilkinson, C.J. Oliver, and R.G. White, 1992. Comparing the performance of SAR image segmentation algorithms, *International Journal of Remote Sensing*, 13(11):2121–2149

Fortin, M.J., R.J. Olson, S. Ferson, L. Iverson, C. Hunsaker, G. Edwards, D. Levine, K. Butera, and V. Klemas, 2000. *Landscape Ecology*, 15:453–466

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

Holt, A.C., E.Y.W Seto, Q. Yu, T. Rivard, and P. Gong, In press. Object-based detection and classification of vehicles from high-resolution aerial photography, *Photogrammetric Engineering & Remote Sensing,* 75(7): 871–880.

Hsu, Chih-Wei, Chih-Chung Chang, and Chih-Jen Lin, 2008. A Practical Guide to Support Vector Classication, URL: *http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf* (last date accessed: 08 December 2009).

Krolupper, F., and J. 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, *Multicomputers and Image Processing: Algorithms and Programs* (K. Preston and L. Uhr, editors), New York: Academic Press, 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, C.-C., and J.G. Dunham, 1993. Shape matching using polygon approximation and dynamic alignment, *Pattern Recognition Letters*, 14:945–949

Lucieer, A., and A. Stein, 2002. Existential uncertainty of spatial objects segmented from satellite sensor imagery, *IEEE Transactions on Geoscience and Remote Sensing*, 40(11).

Martin, D., C. Fowlkes, D. Tal, and J. Malik, 2001. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, *Proceedings of the Eighth International Conference on Computer Vision (ICCV'01)*, Volume 2.

Martin, D.R., C.C. Fowlkes, and J. Malik, 2004. Learning to detect natural image boundaries using local brightness, color and texture cues, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 26(5)

Möller, M., L. Lymburner, and 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., and 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, J.C., D.E. Singh, and F.F. Rivera, 2006. Image segmentation based on merging of sub-optimal segmentations, *Pattern Recognition Letters*, 27:1105–1116

Prieto, M.S., and A.R. Allen, 2003. A similarity metric for edge images, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10)

Radoux, J., and P. Defourny, 2007. A quantitative assessment of boundaries in automated forest stand delineation using very high resolution imagery, *Remote Sensing of Environment*, 110:468–475

Stehman, S.V., and R.L. Czaplewski, 1998. Design and analysis for thematic map accuracy assessment: Fundamental principles, *Remote Sensing of Environment*, 64:331–344.

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

Unnikrishnan, R., Caroline P., and M. Hebert, 2007. Toward objective evaluation of image segmentation algorithms, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6).

Weidner, U. 2008. Contribution to the assessment of segmentation quality for remote sensing applications, *Proceedings of the 21st Congress for the International Society for Photogrammetry and Remote Sensing*, 03–11 July, Beijing, China.

Witten, I.H., and E. Frank, 2005. *Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann*, San Francisco, California.

Yang, L., F. Albregtsen, T. Lønnestad, and P. 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, 06–08 September, pp. 759–765.

Zhan, Q., M. Molenaar, K. Tempfli, and W. Shi, 2005. Quality assessment for geo-spatial objects derived from remotely sensed data, *International Journal of Remote Sensing*, 26(14):2953–2974

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

(Received 19 August 2008; accepted 09 March 2009; final version 16 June 2009)