DEVELOPMENT OF A SUPERVISED SOFTWARE TOOL FOR AUTOMATED DETERMINATION OF OPTIMAL SEGMENTATION PARAMETERS FOR ECOGNITION

Y. Zhang*[a], T. Maxwell, H. Tong, V. Dey

[a] University of New Brunswick, Geodesy & Geomatics Eng., 15 Dineen Dr., E3B 5A3, Fredericton, Canada

Technical Commission VII Symposium 2010

KEY WORDS: Land Cover, Classification, Abstraction, Image, Software, High resolution, Advancement

ABSTRACT:

Image segmentation is one of the most important steps in object-based classification. The commercial software eCognition has been proven to be the most advanced software tool for object-based classification of high resolution remote sensing imagery. However, its segmentation process still relies on trial and error to find proper segmentation parameters. The segmentation process is very time consuming and the segmentation quality directly depends on the experience of the operator. To overcome this problem, a supervised software tool—Fuzzy-based Segmentation Parameter optimizer (FbSP optimizer)—was developed to determine the optimal segmentation parameters through a training process and a fuzzy logic analysis. The optimal segmentation parameters are then used in eCognition to segment the entire image, achieving an optimal segmentation result. The FbSP optimizer can radically increases the efficiency of segmentation parameter selection, and achieve improved segmentation results. It also reduces the influence of the operator's experience on the quality of segmentation results.

1. INTRODUCTION

Since the successful launch of the very high resolution (VHR) Ikonon satellite in 1999, object-based classification has quickly become the mainstream technology for land cover classification of VHR remote sensing images, such as Ikonos, QuickBird, GeoEye-1, WorldView-2 and airborne digital imagery (Smith and Morton, 2010; Blaschke, 2010). In object-based classification, image segmentation is a crucial process which directly influences the efficiency of the classification process and quality of the classification result. To date, eCognition software developed by Definiens has proven to be the most effective technique for object-based classification among a variety of object-based classification techniques (Lavigne et al., 2006).

However, trial and error is still a standard approach of eCognition to finding proper segmentation parameters for achieving a proper segmentation of objects of interest. In the segmentation, operator's knowledge of the image and experience of the segmentation process play an important role for the success of the segmentation. In addition, the segmentation process is time consuming. These drawbacks have significantly limited the potential of eCognition for a broad range of practical applications.

To overcome the limitation of eCognition in finding proper segmentation parameters for image segmentation, a software tool has been developed in the CRC-AGIP Lab (Canada Research Chair Laboratory in Advanced Geomatics Image Processing) at the University of New Brunswick, based on previous work done in the lab (Maxwell, 2005; Zhang and Maxwell, 2006). The software tool, named Fuzzy-based Segmentation Parameter optimizer (FbSP optimizer), can automatically determine optimal segmentation parameters for eCognition through a supervised training process and fuzzy logic analysis. Using the FbSP optimizer in combination with eCognition, the segmentation of an object of interest can be achieved within minutes, instead of hours by solely using eCognition. In addition, the segmentation result can be significantly improved.

This paper will first introduce the general concept of image segmentation used in eCognition and the role of segmentation parameters. It will then introduce the concept and process of the developed of the FbSP optimizer for identifying optimal segmentation parameters for eCognition. The experiment results and the comparison between the segmentation qualities and time used in the segmentation processes will also be given to allow readers to judge the improvement made by the supervised software tool— FbSP optimizer.

2. SEGMENTATION TECHNIQUE OF ECOGNITION

2.1. Region Merging

To find the boundary of an image object or segment an object, eCognition implemented a region merging approach to segmentation called "Fractal Net Evolution" approach (Baatz and Schape, 1999). This technique starts with individual adjacent pixels as initial objects, and then measures (1) the spectral heterogeneity change, $h_{spectral}$, and (2) the shape heterogeneity change, h_{shape}, between the two neighbor pixels (objects) to determine whether they need to be merged together, or not. Once the two pixels are merged into one object, the region of the object grows one step. This measurement and merging process continues iteratively until a user defined threshold is reached. Then, the region of the object stops growing; resulting in one image segment. The region merging and region growing process was designed with the view to meeting six aims including the (Baatz and Schape, 2000):

a. Production of homogeneous image object-primitives;
b. Adaptability to different scales;
c. Production of similar segment sizes for a chosen scale;
d. Applicability to a variety of data sets;
e. Reproducibility of segmentation results; and
f. Requirement for reasonably fast performance.

2.2. Role of Segmentation Parameters

Figure 1 illustrates the relationship between spectral heterogeneity change, $h_{spectral}$, the shape heterogeneity change, h_{shape}, and the corresponding segmentation parameters. Where

- $h^c_{spectral}$ is spectral heterogeneity change of individual spectral bands,

- $h_{compact}$ is compactness heterogeneity change,
- h_{smooth} is smoothness heterogeneity change,
- w_c are the weights associated with each layer,
- $w_{compact}$ is compactness weight (parameter),
- $1\text{-}w_{compact}$ is smoothness weight (parameter),
- w is weight (parameter) for overall spectral heterogeneity change, and
- $1\text{-}w$ is weight (parameter) for shape heterogeneity change.

The $h^c_{spectral}$, $h_{compact}$, and h_{smooth} are calculated according to the image grey values within the two neighbor objects in each spectral bands, whereas the weighs (parameters) w_c, $w_{compact}$ (or $1\text{-}w_{compact}$), and w (or $1\text{-}w$) must be given by the user. The user must also give a scale value (s) as a threshold to stop the merging. Figure 2 shows the interface of eCognition to allow users to input the parameters to guide and control the segmentation process.

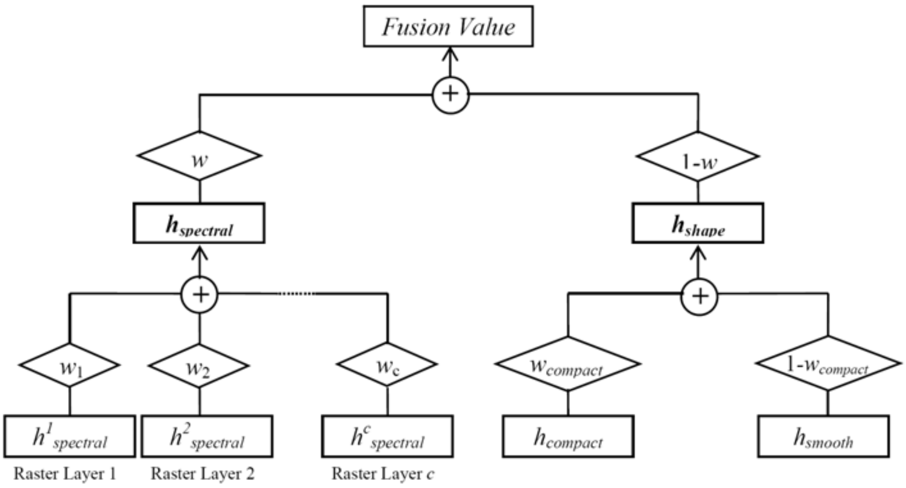

Figure 1. Relationship between the segmentation parameters (user determined weights) in eCognition. Usually, the weights for individual spectral layers (bands) (w_1, w_2, … w_c) are set to 1. Users need to give the value for Smoothness weight ($1\text{-}w_{compact}$) (or Compactness weight ($w_{compact}$)) and Shape weight ($1\text{-}w$). The weights ($1\text{-}w_{compact}$) and ($1\text{-}w$) are used to calculate the Fusion Value (f). The value f is then compared with a user specified Scale value (s) to estimate whether the two adjacent objects need to be merged, or not (if $f < s^2$, merge the two objects; if $f \geq s^2$, stop the merging).

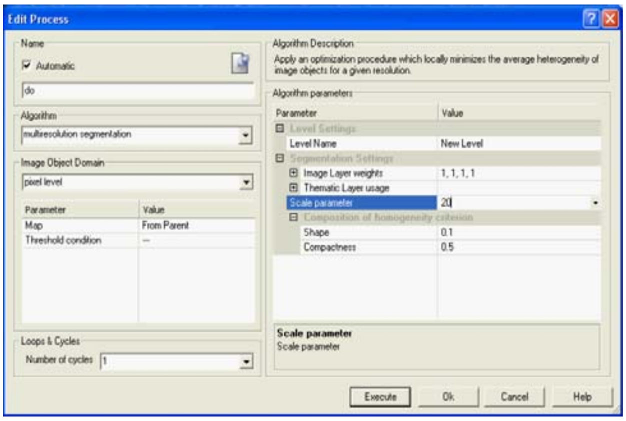

Figure 2. Interface of eCognition to allow users to input user defined segmentation parameters (Image Layer weights (w_1, w_2, … w_c), Scale parameter (s), Shape ($1\text{-}w$), and Compactness ($w_{compact}$))

The purposes of the segmentation parameters are (Hofmann, 2001):

(1) Scale parameter: influence the average object size. It determines the maximal allowed heterogeneity of the objects. The larger the scale parameter, the larger the objects become.

(2) Shape/Color: adjust the influence of shape vs. color homogeneity on the object generation. The higher the shape value, the less spectral homogeneity influences the object generation.

(3) Smoothness/Compactness: determine the compactness or smoothness of the resulting object. With a selected shape value, the user can influence the compactness or smoothness of the final object.

(4) Image Layer weights: determine the weight of each spectral band in the segmentation. It is used to control the influence of each band on the object generation.

(5) Level settings: determine whether a newly generated image level will either overwrite a current level or whether the generated objects shall contain sub- or super-objects of a existing level. The order of the level generation affects the objects' shape (top-down vs. bottom-up segmentation).

2.3. Difficulty of Segmentation Parameter Selection

The segmentation parameters to be selected by the user are interrelated to each other. It is impossible to directly find a set of proper segmentation parameters at one time. Users have to repeatedly select a set of segmentation parameters and test them through a trial-and-error process, until a reasonable segmentation result is achieved or the user does not want to continue the trial and error any more. The change of any of the parameters affects the influences of other parameters on the segmentation, so that it is a tedious and time-consuming process. The segmentation results directly depend on the knowledge and experience of the user. The segmentation process is considered by users as a "black art" (Smith and Morton, 2010). Normally, only those users who are familiar with the spectral characteristics of the land-cover objects of interest and understand the segmentation procedure can select proper segmentation parameters in a relatively efficient way. But, this is not always available in practice (Flanders et at., 2003; Hay et al., 2003; Maxwell, 2005; Li et al., 2009; Smith and Morton, 2010).

3. SUPERVISED DETERMINATION OF SEGMENTATION PARAMETERS

3.1. Design of the Segmentation Tool

To address the issues existing in the trial-and-error selection of segmentation parameters, a tool for supervised segmentation parameter determination should meet the following requirements (Maxwell, 2005):

a. Each execution of the tool is aimed at extracting one land cover type and results in one level of the object hierarchy;
b. Segmentation must be controlled and refined in an iterative manner based on an object model;
c. The tool must rely on an initial segmentation as a start state;
d. Scale, shape, and smoothness parameters must be determined;
e. Parameter selection must be reproducible; and
f. The tool must demonstrate reasonably fast and efficient performance.

The segmentation of an input image is performed on a number of different levels to permit objects of different scales to be extracted on their own level. By using this approach, objects can be classified on the level where the segments are the most meaningful and best represent the object of interest. This infers that the user must have a specific land cover class in mind when segmenting the image so that the parameters can be best estimated and then refined through iteration. As a result, the tool must aim to extract one particular land cover type each time it is executed. By running the tool a number of times, a hierarchy of object levels can then be developed.

3.2. Workflow of FbSP Optimizer

To meet the design requirements of the software tool, the workflow for the supervised fuzzy-based determination of segmentation parameters, i.e. the FbSP optimizer, was developed as shown in Figure 3. To train the FbSP optimizer, the input image needs to be initially segmented achieving an over segmentation, i.e. the segments are smaller than the objects of interest (see Figure 5.a). The small segments, also called sub objects, can be selected to form a meaningful target segment/object. The information of the target object and its sub objects is then used to train the FbSP optimizer to determine the optimal segmentation parameters for the target object (Figure 4).

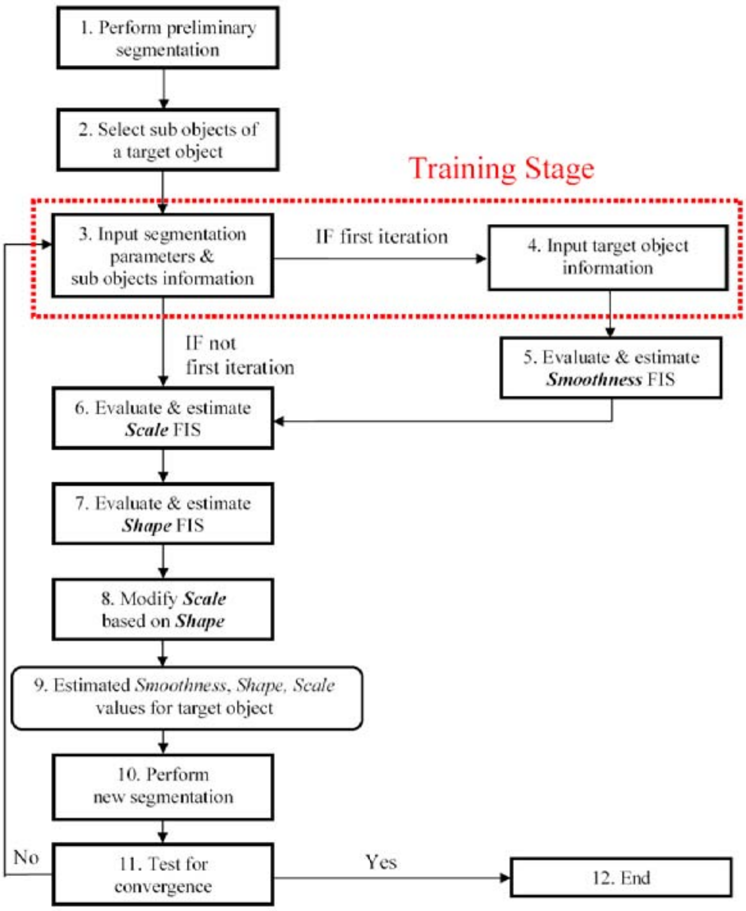

Figure 3. Workflow of the proposed FbSP optimizer. The values of the current Segmentation Parameters (*Smoothness* (1-$w_{compact}$), *Shape* (1-w) and *Scale* (s)), Sub objects information (*Texture*, *Stability*, *Brightness*, and *Area*) and Target Object information (*Texture*, *Stability*, *Brightness*, *Area*, *Rectangular Fit*, and *Compactness*) are inputted into FbSP optimizer to train the FISs (Fuzzy Inference Systems) to estimate the optimal Segmentation Parameters (1-$w_{compact}$, 1-w, and s) for the Target Object in an iterative process.

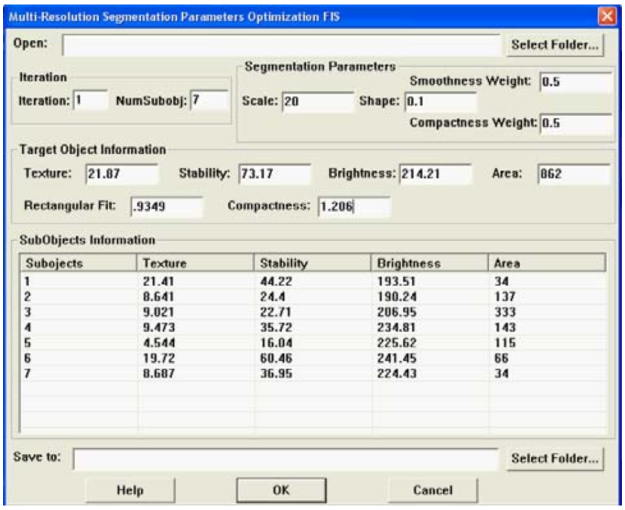

Figure 4. Interface of FbSP optimizer. The Segmentation Parameters, Target Object Information and Sub Object Information are inputted into the system to train FbSP optimizer to estimate the optimal Segmentation Parameters for the Target Object through fuzzy logic analyses.

In Figure 4, the Segmentation Parameters (Scale, Shape, and Smoothness weight) for performing a preliminary over segmentation are initially selected and inputted by user. They are then updated by the FbSP optimizer in the next iteration according to the information of the target object ant its sub objects. The Texture, Stability, Brightness and Area for the target object and sub objects are calculated according to the pixel grey values within the target object and individual sub objects, respectively. The Compactness and Rectangular Fit are calculated according to the shape of the target object.

4. EXPERIMENT AND RESULTS COMPARISION

4.1. Data Sets

Pan-sharpened QuickBird MS image and pan-sharpened Ikonos MS image over Fredericton, Canada, and pan-sharpened QuickBird MS image of Oromocto, Canada, were used to test the FbSP optimizer. The UNB-Pansharp was used to fuse the Pan and MS images. The four pan-sharpened multispectral bands were used as input bands.

4.2. Segmentation Process and Results

For small objects, the FbSP optimizer can find optimal segmentation parameters through one or two iteration(s) of segmentation parameter estimation (Figure 3 and Figure 5). For example, for small buildings shown in Figure 5, the FbSP optimizer used just one iteration to find the optimal segmentation parameters for the object of interest—small buildings. If the initial parameter selection by the user for the initial over segmentation is counted as one iteration, two iterations of parameter selection in total were needed, one by the user and one by the FbSP optimizer (Figure 5.a and 5.c, and Table 1).

Table 1 shows the segmentation parameters selected by the user for initial over segmentation (Iteration 1), and the parameters estimated by the FbSP optimizer in the first loop (Iteration 2). Table 2 shows the feature information of the sub objects (Figure 5.a, red) selected to form a target object. Table 3 lists the feature information of the target object (Figure 5.b, red).

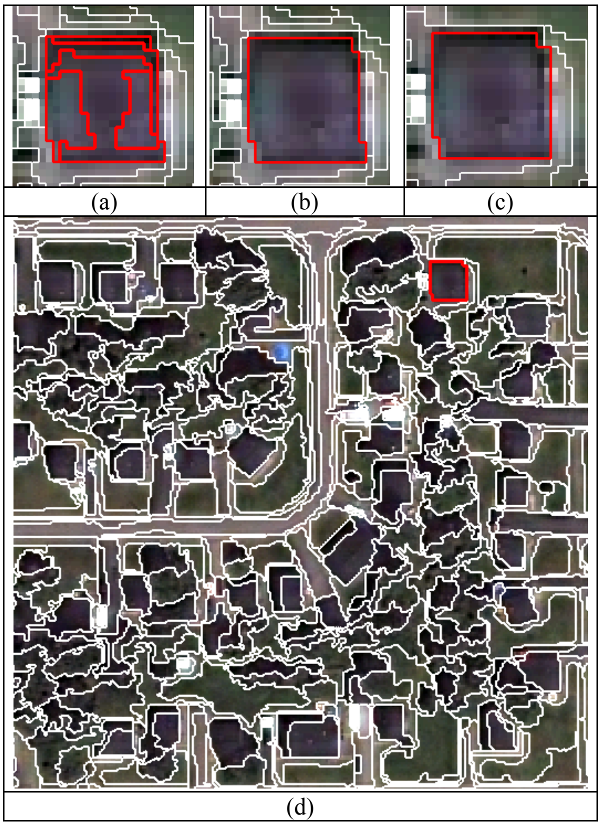

Figure 5. Segmentation process and final segmentation result using the FbSP optimizer for small buildings (pan-sharpened QuichBird MS, Oromocto). (a) Initial over segmentation using user selected initial segmentation parameters (first iteration) and the sub objects (red) selected to form a target object, (b) Target object formed by the sub objects for training the FbSP optimizer, (c) Segmentation result achieved using the parameters estimated by FbSP optimizer in the first loop (second iteration in total), (d) Final segmentation result using the parameters estimated in (c).

According to the feature information in Table 2 and Table 3, the FbSP optimizer estimated the optimal segmentation parameters (Table 1, Iteration 2) for the target object. Using the parameters in Iteration 2 of Table 1 to segment the input image, the segmentation result shown in Figure 5.c was achieved. The feature information (Table 4) of the resulting segment (Figure 5.c, red) is almost identical to that of the target object (Table 3 and Figure 5.b, red), i.e. the resulting segment converges with the target object (Figure 3, step 11), so that the segmentation parameters estimated by FbSP optimizer in the first loop (Iteration 2 of Table 1) was accepted as the optimal segmentation parameters for small buildings.

For large objects, more sub objects need to be selected to form a target object, so that more iterations are usually needed to reach the convergence between the target object and the resulting segment. For example, four iterations were needed to reach the convergence for large buildings shown in Figure 6.

4.3. Result Evaluation

Figure 7 shows the segmentation result of large buildings using trial-and-error approach for segmentation parameter selection. The result was achieved by a very experienced operator through approximately two hours of parameter selection and test. Comparing the result from FbSP optimizer (Figure 6) and that from trial and error (Figure 7), we can see that the FbSP optimizer achieved better segmentation result, especially for complex buildings (compare the circled buildings in Figure 6 and 7).

In terms of time used for the segmentation, the FbSP optimizer presented a much more significant improvement. To achieve the segmentation shown in Figure 6, only 30 minutes were needed under the current software implementation condition, i.e. the operator needs to generate feature information for sub objects and target object and then input and output between eCognition and FbSP optimizer manually and iteratively. The manual, iterative input and output between the two systems occupied more than 90% of the time in the parameter determination process of FbSP optimizer. If the FbSP optimizer can be integrated into eCognition through a API (Application Programming Interface), the 90% of time can be saved. Then, the FdSP optimizer just needs a few minutes to obtain the optimal segmentation parameters for the building segmentation in Figure 6, demonstrating a radical improvement in terms of speed.

Table 1. Segmentation parameters for small buildings

Parameter	**Iteration**	
	1	**2**
Scale	20	35.1809
Shape	0.1	0.551
Smoothness	0.5	0.5

Table 2. Feature information of the sub objects (Figure 5.a, red) forming a target object

Sub object	Texture	Stability	Brightness	Area
1	25.35	55.51	189.56	60
2	14.21	51.6	179.3	123
3	15.63	25.9	194.59	44
4	24.64	50.91	164.07	44
5	16.06	72.52	151.61	15

Table 3. Feature information of the target object (Figure 5.b, red) formed by the sub objects (Figure 5.a, red)

Texture	Stability	Brightness	Area	Rectangle Fit	Compactness
35.81	143.8	183.92	299	0.9792	4.048

Table 4. Feature information of the sub object (Figure 5.c, red) obtained using the parameters estimated by FbSP optimizer (Table 1, Iteration 2)

Sub object	Texture	Stability	Brightness	Area
1	35.81	142.34	183.92	299

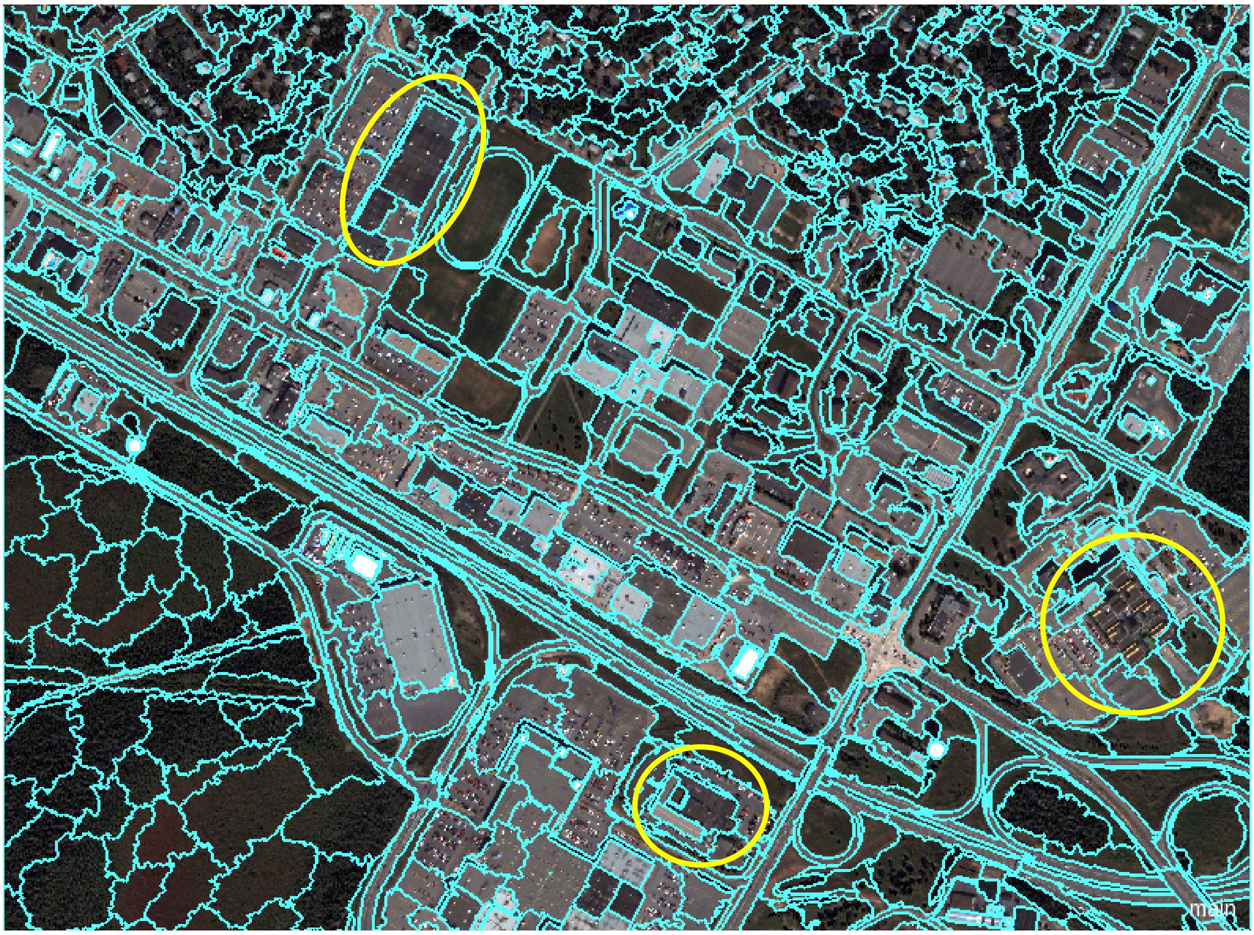

Figure 6. Segmentation result of large buildings obtained using the segmentation parameters of FbSP optimizer, operating time: 30 minutes (90% of the time was used for manual and iterative input and output of the object feature information between eCognition and FbSP optimizer, which can be reduced once the two systems are integrated) (pan-sharpened QuichBird MS, Fredericton)

Figure 7. Segmentation result of large buildings obtained through trial-and-error parameter selection, operating time: 2 hours (pan-sharpened QuichBird MS, Fredericton)

Segmentation of other objects was also tested. The trial-and-error approach needs 2 to 6 hours to reach acceptable segmentation results, whereas FdSP optimizer just needs 30 to 60 minutes. If the 90% of the time for manual input and output is reduced through software integration, FdSP optimizer will just need less than 10 minutes to select optimal segmentation parameters for large complex objects.

5. CONCLUSIONS

A Fuzzy-based Segmentation Parameter (FbSP) optimizer was developed to improve the efficiency of segmentation parameter selection and accuracy of object segmentation for eCognition. The FbSP optimizer can be trained using initially segmented sub segments and the corresponding targeted object of interest. The FbSP optimizer can then find the optimal segmentation parameters for the target object, through fuzzy logical analyses of the target object and its sub objects.

Experiments with QuickBird and Ikonos pan-sharpened MS images demonstrated that the FbSP optimizer can effectively find optimal segmentation parameters for objects of interest within 30 to 60 minutes under the current software implementation condition. If the current manual and iterative input and output of feature information between FbSP optimizer and eCognition is reduced through software integration, FbSP optimizer will just need a few minutes to find optimal segmentation parameters for an object of interest. In contrary, 2 to 6 hours are usually needed for an experienced operator to find proper segmentation parameters through trial and error. The proposed supervised approach to automated determination of optimal segmentation parameters has demonstrated its superior advantage in speeding up the segmentation parameter selection and improving the segmentation quality. It exhibits the potential to boost segmentation techniques from current trial-and-error stage into the next stage—semi-automated or automated process.

Further tests with other remote sensing images will be conducted. The FbSP optimizer will be further improved.

6. ACKNOWLEDGMENTS

This research was supported by the Discovery Research Grants Program of NSERC (Natural Science and Engineering Research Council of Canada), Canada Research Chairs Program, and the Department of National Defence (Canada). The original QuickBird and Ikonos satellite images were provided by CFB Gagetown and the City of Fredericton, NB, Canada.

7. REFERENCES

Baatz, M. and A. Schäpe, 2000, Multiresolution Segmentation - An Optimization Approach for High Quality Multi-Scale Image Segmentation. *Angewandte Geographische Informationsverarbeitung XII*, Ed. J. Strobl et al. AGIT Symposium, Salzburg, Germany, 2000. pp. 12-23.

Baatz, M., Schäpe, A., 1999. Object-Oriented and Multi-Scale Image Analysis in Semantic Networks. In: Proc. of the 2nd International Symposium on Operationalization of Remote Sensing, August 16-20, 1999. Enschede. ITC.

Blaschke, T., 2010. Object based image analysis for remote sensing, ISPRS Journal of Photogrammetry and Remote Sensing, Vol. 65(1), pp. 2-16..

Flanders, D., M.Hall-Beyer, and J. Pereverzoff, 2003. Preliminary Evaluation of eCognition Object-Based software for Cut Block Delinieation and Feature Extraction. *Canadian Journal of Remote Sensing*, Vol. **29**, No. 4, pp. 441-452.

Hay, G., T. Blaschke, D. Marceau, and A. Bouchard, 2003. A Comparison of Three Image-Object Methods for the Multiscale

Hofmann, P , 2001. Detecting Urban Features from IKONOS Data using an Object-Oriented Approach. *Proceedings of the First Annual Conference of the Remote Sensing & Photogrammetry Society*. RSPS, Munich, Germany, 12-14 September 2001, pp. 79-91.

Lavigne, D.A., G. Hong and Y. Zhang, 2006. Performance Assessment of Automatic Feature Extraction Tools on High Resolution Imagery, *MAPPS/ASPRS 2006 Fall Conference*, November 6-10, 2006, San Antonio, Texas.

Li, H., Gu, H., Han, Y. and Yang, J., 2009. An efficient multiscale SRMMHR (Statistical Region Merging and Minimum Heterogeneity Rule) segmentation method for high-resolution remote sensing imagery. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2(2), pp. 67-73.

Maxwell, T., 2005. Object-Oriented Classification: Classification of Pan-Sharpened QuickBird Imagery and a Fuzzy Approach to Improving Image Segmentation Efficiency. M.Sc.E. thesis, Department of Geodesy and Geomatics Engineering Technical report No. 233, University of New Brunswick, Fredericton, New Brunswick, 157 pp.

Smith, G.M., and R.D. Morton, 2010. Real World Objects in GEOBIA through the Exploitation of Existing Digital Cartography and Image Segmentation, Photogrammetric Engineering & Remote Sensing, Vol. 76, No. 2, , pp. 163–171.

Zhang, Y., and T. Maxwell, 2006. A trained segmentation technique for optimization of object-oriented classification, ISPRS Commission VII Mid-term Symposium “Remote Sensing: From Pixels to Processes, Enschede, the Netherlands, 8-11 May 2006.