

INTERACTIVE OBJECT SEGMENTATION IN HIGH RESOLUTION SATELLITE IMAGES

J. Osman, J. Inglada

CNES
DCT/SI/AP
18, Av. E. Belin,
31401 Toulouse Cedex 9, France

E. Christophe

CRISP
Block SOC-1, Level 2,
Lower Kent Ridge Road
Singapore 119260

ABSTRACT

High resolution remote sensing image segmentation is a great challenge in terms of potential applications, but also because of the difficulty of the task. Fully automatic algorithms are not able to extract all the desired features from complex images but visual image analysis is time consuming and tedious (therefore error prone). In this work we present a simple, yet powerful approach for interactive image segmentation. This approach combines the best of the automatic image processing together with the ability of a human operator to choose the objects of interest for a given application. Results are presented on a wide variety of objects and contexts.

Index Terms— Image Segmentation, Support Vector Machines, Region Growing

1. INTRODUCTION

High resolution remote sensing image segmentation is a great challenge in terms of potential applications, but also because of the difficulty of the task. Indeed, metric and sub-metric resolution images allow to access landscape features which are difficult to extract and understand in a fully automatic way. These features are often related to objects and not to individual pixels as it was the case for metric resolution images. This is why the paradigm of Object-Based Image Analysis has been introduced in the recent years. Nevertheless, existing algorithms still fail to produce pertinent segmentation results in a fully automatic way.

On the other hand, visual image analysis is time consuming and tedious (therefore error prone). In this work we present a simple, yet powerful approach for interactive image segmentation. This approach tries to combine the best of the automatic image processing together with the ability of a human operator to choose the objects of interest for a given application. It is inspired from approaches proposed in natural image processing as for instance in [1].

In terms of domain of application, the main constraint imposed to our system is that it should perform well for different kinds of objects (no object specificities in the algorithm).

Also, since the approach has to be interactive, the processing time has to be very short: less than a few seconds.

2. ALGORITHM DESCRIPTION

The algorithm considers as input a classical 4-band (blue, green, red and near-infrared) optical high resolution image as the ones delivered by Ikonos, Quickbird and the future Pleiades systems. These images are usually pan-sharpened, but this is not a real requirement for the algorithm.

On the input image, the user is invited to select some samples inside the object of interest and some additional samples on its neighborhood. The task of selecting a few pixels inside and outside the object is much faster and much less tedious than precisely delimiting the contour of the object. From these 2 sets of samples, a binary mask is produced in order to represent the selected object. The processing is decomposed into the following steps:

1. feature computation: NDVI, water index, spectral angle with respect to the training samples;
2. unsupervised clustering of the spectral angles extracted from the input samples in order to have a fixed length feature vector;
3. SVM learning using the 2 (inside and outside) training sets;
4. generation of an image of distance to the separating surface: this image gives, for each pixel, the likelihood of belonging to the object of interest;
5. rough detection of shadows to help the next step;
6. region growing segmentation of the likelihood image: the inside samples given by the user are used as seeds for the region growing which uses an Otsu thresholding [2]; shadows are used to stop the growing if they are not selected as an object of interest.

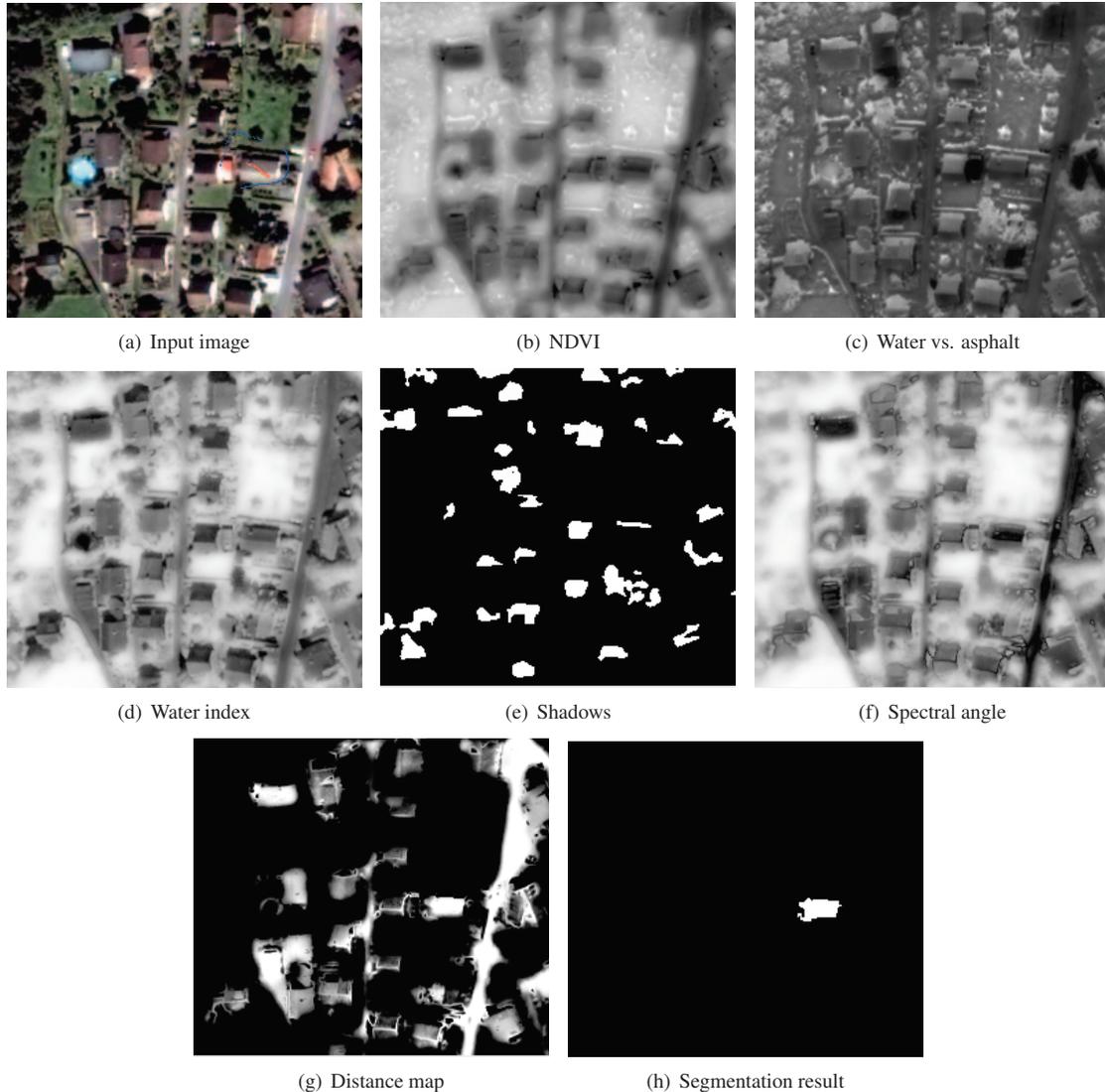


Fig. 1. Building segmentation: overview of the different steps of the processing.

2.1. Features

The features extracted from the input image for the segmentation are the following:

- The well known *Normalized Difference Vegetation Index* (NDVI):

$$NDVI = \frac{(NIR - RED)}{(NIR + RED)} \quad (1)$$

Figure 1(b) shows an example of this feature for the image of figure 1(a).

- An index similar to the NDVI which is mostly useful for the discrimination of water and asphalt:

$$WI = \frac{(GREEN - RED)}{(GREEN + RED)} \quad (2)$$

Figure 1(c) shows an example of this feature for the image of figure 1(a).

- A water index. There exist several water indices available in the literature, as for instance [3], but they need a richer set of spectral bands than those available in commercially available high resolution satellite imagery. In order to produce a water index, an empirical approach was taken: a set of water bodies coming from a large variety of images was used in order to compute a mean water spectrum. The water index is computed using a spectral angle – conveniently normalized – between the reference water spectrum and each pixel of interest. Figure 1(d) shows an example of this feature for the image of figure 1(a). It has to be noted that contrary to the NDVI, as it is defined as a distance to a reference low

value of the index (black) will indicate the presence of water.

- 10 spectral angles defined by the inside samples: the set of samples entered by the user are clustered using a classical K-means algorithm and each of the cluster centroids are used as reference values for the spectral angles. Figure 1(f) shows an example of this feature for the image of figure 1(a).
- A (binary) shadow index. Shadow extraction can be done quite simply on the luminance image looking for areas with a low luminance. The extraction of these regions is done in two steps: a hard very low thresholding whose purpose is to provide at least one pixel per shadow region and a region growing from these seeds with a wider tolerance in terms of radiometry. Figure 1(e) shows an example of this feature for the image of figure 1(a).

2.2. Learning and distance map generation

The features presented in section 2.1 are extracted for every pixel of an area of interest around the user selection. Since we are dealing with a user-in-the-loop approach, the area of interest does not have to be estimated and is set to the area displayed by the user.

The features are arranged into sample vectors which will be used for supervised classification. The first step is the learning phase which will consist in training a binary Support Vector Machine classifier [4]. Once the learning is complete, the second phase will consist in producing a likelihood image of each pixel being inside the object of interest. Indeed, a hard decision on the class label is avoided here. Therefore, for each pixel, a signed distance to the separating hyper-plane (the decision boundary of the supervised classification) is computed. The result is a distance map: an image where the pixel values increase with the likelihood of belonging to the inside class. Figure 1(g) shows an example of distance map for the image of figure 1(a). It is interesting to note that most of the buildings of the image have a high likelihood of belonging to the same class as the one selected by the user.

2.3. Object extraction

Although a thresholding of the distance map might be considered enough for the object extraction step, we can also take advantage of the fact that we aim at segmenting a single region for which we know the location of some of the pixels (the inside samples selected by the user).

Therefore, we use a region growing algorithm which adds connected pixels if they are below a threshold. The inside samples selected by the user are used as the region growing seeds. The threshold which allows to decide whether a pixel belongs to the object or not is computed automatically

as follows. A small number of thresholds – typically 4 – are computed using the approach proposed by Otsu [2] and the threshold to be used is the highest one which is below the value of the distance map for the inside samples.

From the distance map, we can see that the limit of the building is not always clear. An element which is present in most images and quite easy to extract can help to get a better extraction for the object: shadows. The shadow region extracted are used to bound the growing on side of the building where it is available.

3. RESULTS

Figure 1(h) shows the result of applying the algorithm described above to the image of figure 1(a). One can observe that the result is of good quality despite the minor errors on the borders caused by the high similarity in terms of radiometry between the building's roof and the concrete paths around it.

In order to show the difficulty of the task, figure 2 shows the result of a region growing segmentation. A classical region growing algorithm does not obtain any pertinent results. In this case we have used the so-called confidence connected region growing which extracts a connected set of pixels whose pixel intensities are consistent with the pixel statistics of a seed point.

The mean and variance across a neighborhood (8-connected) are calculated for a seed point. Then pixels connected to this seed point whose values are within the confidence interval for the seed point are grouped. The width of the confidence interval is controlled by the a coefficient (1.5 here) variable (the confidence interval is the mean plus or minus this coefficient times the standard deviation).

After this initial segmentation is calculated, the mean and variance are re-calculated. All the pixels in the previous segmentation are used to calculate the mean the standard deviation (as opposed to using the pixels in the neighborhood of the seed point). The segmentation is then recalculated using these refined estimates for the mean and variance of the pixel values. This process is repeated for the specified number of iterations (2 here).

Figure 2(a) shows the result of the segmentation using one seed on the lower part of the roof. As expected, tuning the parameters in order not to get out of the region, the upper part of the building is not segmented. Figure 2(b) shows what happens when a second seed is selected in order to segment the upper part of the roof with the same parameter set: the region growing gets out of the region and the segmentation is wrong.

Figure 3 shows the results obtained on 2 different images of buildings with non homogeneous roofs. The blue lines correspond to the user input samples. One can see the good performances of the algorithm.

This algorithm is available in OTB [5], <http://www.>

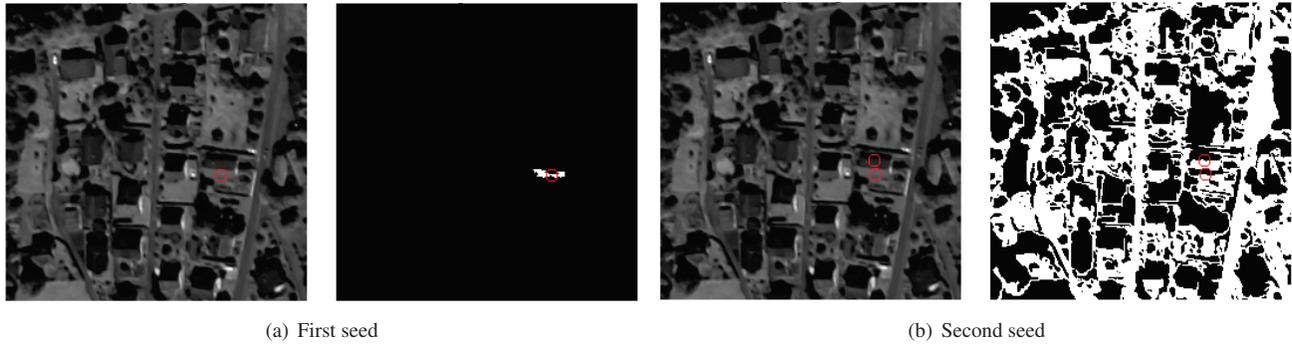


Fig. 2. Illustration of the region growing segmentation: this classical algorithm which is reliable to segment uniform areas fails when the area is not uniform. In (a), the first seed extracts only half of the roof as the other side has a different sun exposition. When a second seed is provided, in (b), the difference between the two seeds is too important and the segmented area goes well beyond the roof-top.

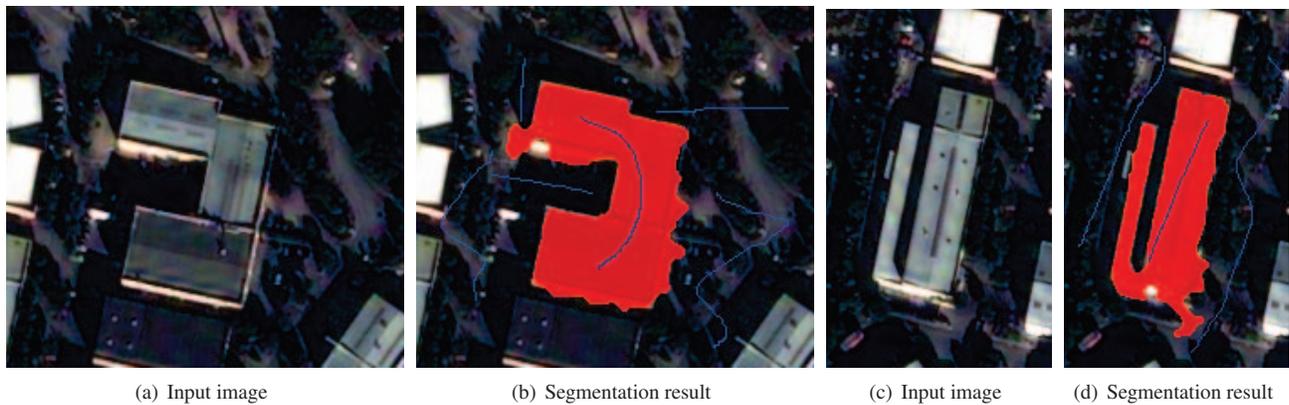


Fig. 3. Building segmentation.

orfeo-toolbox.org, as atomic functionality, but also as a stand-alone application with a graphical user interface (GUI) allowing for simple man-machine interaction.

4. CONCLUSION

An efficient and flexible method for the segmentation of complex object as been presented. A full open-source implementation is provided, both as a library and as a stand-alone application with GUI, allowing experiments on a wide range of images. The processing time, in the order of the second, is compatible with interactive processing.

5. REFERENCES

- [1] G. Friedland, K. Jantz, and R. Rojas, "SIOX: Simple interactive object extraction in still images," *Proceedings of the IEEE Symposium on Multimedia (ISM2005)*, pp. 253–259, 2005.
- [2] N. Otsu, "An automatic threshold selection method based on discriminant and least squares criteria," *IECE Transactions*, vol. 63, pp. 349–356, 1988.
- [3] Bo-Cai Gao, "NDWI – A Normalized Difference Water Index for Remote Sensing of Vegetation Liquid Water From Space," *Remote Sensing of Environment*, vol. 58, no. 3, pp. 257–266, Dec. 1996.
- [4] B.E. Boser, I.M. Guyon, and V.N. Vapnik, "A training algorithm for optimal margin classifiers," *Proceedings of the fifth annual workshop on Computational learning theory*, pp. 144–152, 1992.
- [5] "The ORFEO toolbox software guide," <http://www.orfeo-toolbox.org>, 2009.