ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

Automatic Building Detection Using Modified Grab Cut Algorithm from High Resolution Satellite Image

Mahak Khurana[1], Vaishali Wadhwa[2]

M. Tech. Student, Computer Science and Engineering, N.C.College of Engineering, Panipat, India [1]

Associate Professor, Computer Science and Engineering, N.C.College of Engineering, Panipat, India [2]

Abstract: Understanding aerial image / satellite image is highly demanded for generating maps, analyzing disaster scale, detecting changes to manage estate, etc. But it's been usually done by human experts, so that it's both slow and costly. Remote sensing community has been focused on this task but it's still difficult to detect terrestrial objects automatically from high resolution satellite image with high accuracy. An important Object detection problem in remote sensing is the automated detection of buildings from satellite images. In this paper , Modified grab cut partitioning algorithm will be used to detect buildings in image which will take input from the previous objective and rather than min-max evaluation used in grab cut, we will use bio inspired optimization which will find a global optimal solution for maximum energy better than min max algorithm.

Keywords: Satellite Image, Thresholding, Shadow Detection, Grab cut Algorithm, Bacterial Foraging optimisation, Fuzzy Landscape.

I. INTRODUCTION AND MOTIVATION

Nowadays; by virtue of the advances in satellite data acquisition, object detection and classification problems in remote sensing, specifically in very high resolution (VHR) satellite images, have been a popular research topic.
An important object detection problem in remote sensing is the automated detection of buildings from satellite images. Up to now, several building detection methodologies have been proposed in the literature. For the cases where multiple sensors are used for data acquisition, the problem becomes easier.

Using Light Detection and Ranging (LIDAR) or stereo imaging, it is possible to obtain reliable height information of the objects in the terrain. This information can be integrated with the multispectral color information obtained from the optical sensor to detect buildings accurately. Moreover, fusion of optical and synthetic aperture radar (SAR) sensors provides more reliable information than utilizing only one of these sensors.

On the other hand, building detection from monocular images is much more difficult. Unfortunately, it is not possible to generalize the shape, size, color of a building. Therefore, building detection methodologies using classical pattern recognition approaches are not guaranteed to work in every case; the performance depends on the statistical correlation between training and test data (for supervised approaches), and distribution of the features extracted from the image (for unsupervised approaches).
Introduction of too many parameters and the algorithm's sensitivity to these parameters is another issue. For a Robust and generalizable object detection algorithm; obtaining an invariant1 for the object is essential.

To give an example; for detecting balls in images, the circularity can be utilized as an invariant for balls, because every ball in the world has a circular shape. Moreover; in order to detect lemons in an image, yellow color is considered as an invariant for lemon object, since all lemons have a color of yellow.
It is worth noting that the object invariance property is not one-to-one and onto. In other words;

- Every object satisfies the constraints related to its invariant (e.g. all lemons in the world are yellow),
- But an arbitrary object satisfying the invariant constraints is not necessarily the object to be detected (e.g. a yellow object in the image may not be a lemon).

For building detection from monocular satellite images, this invariant can be shadow of a building. All buildings differ from each other in terms of their colors, sizes, shapes, textures; but they all have shadow regions attached to them.

The shadow detection in urban areas is totally different which often exhibit compact areas of buildings as one factor. This factor can affect the actual size and shape of urban objects that are presented by their shadows.
The reason behind this is the shadow line of the object will be shorter in the case of a rise in a horizontal receiving surface and longer where there is a drop in this surface for the other objects within urban areas.
The second factor is the existence of trees and their shadows or even shadows from non-built-up areas, such as vehicles, which can distort the real boundaries of shadows (the geometric form) and give an arbitrary shape to the built-up objects, when they combine with the object's shadows.
Another factor is the shadow regions of objects in images will be larger in the summer season than their counterpart

 DOI 10.17148/IJARCCE.2015.4833

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

in the winter due to the sun's position in the sky vault and its angle and altitude. Spectrally, non-shadow regions, for instance, water, can present the same pixel intensity values or darkness with shadow regions of the objects in images that can cause an error in detecting shadows.
Therefore, shadow detection and its extraction from VHR satellite images are not easy tasks and all image processing techniques for detecting shadows still depend on the estimation of shadow areas but using various developed methods.
In this Paper, a Research methodology is introduced in which; first the vegetation regions are detected from a given satellite image and local directional fuzzy landscapes representing the existence of building are generated from the shadow regions using the direction of illumination obtained from image metadata. For each landscape, foreground (building) and background pixels are automatically determined and a bi partitioning is obtained using a graph-based algorithm, Grab cut, which is further modified by an iterative bio inspired optimization: bacterial foraging optimization. This work towards the optimization k-means clustering used in grab cut method. Finally, local results are merged to obtain the final building detection result. Considering performance evaluation results, this approach can be seen as a proof of concept that the shadow is an invariant for a building object and promising detection results can be obtained when even a single invariant for an object is used.

II. METHODOLOGY

A. Vegetation Mask and Shadow Mask

In this paper, for the detection of the vegetation areas, widely accepted Normalized Difference Vegetation Index (NDVI) metric is utilized. The metric **NDVI** involves a comparison between the values of the near-infrared (NIR) and red image bands (RIB)

$$\textbf{NDVI} = \textbf{(NIR-RIB) / (NIR+RIB)}$$

The NIR and RIB represent the normalized near-infrared And red image bands, respectively. For each test site, we use the automatic thresholding based on the Otsu’s method To find the optimum threshold of the histogram of the NDVI ratio values computed from equation and apply that threshold on the NDVI ratio map to compute a binary vegetation mask.
Next,the false color image is normalized and converted to Hue-Saturation-Intensity (ρ_{HSI}) color space. Then, a ratio map (ρ_{RM}), in which the normalized saturation (ρ_S) and the normalized intensity (ρ_I) values are compared with a ratio, is generated:

$$\rho_{RM} = \frac{\rho S-\rho I}{\rho S+\rho I}$$

To detect the shadow areas, as utilized in the case of vegetation extraction, Otsu’s method is applied to the histogram of the ratio map, ρ_{RM}. Due to the fact that the thresholding scheme detects both shadow and vegetation regions at the same time, the regions that belong to the vegetation are subtracted to obtain a binary shadow mask.

This approach provided successful shadow detection results for various satellite images and the major advantage is that it is independent from manual thresholds. Figure 1.1 and 1.2 shows example of Input image and detected shadow regions.

Figure 1. Input Satellite Image

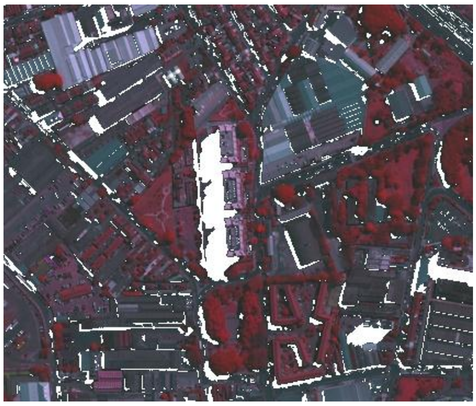

Figure 2. Shadow Detected

B. Generation of Shadow Probability Map

After detecting shadows and determining the illumination direction in the image, the next step is to generate fuzzy landscapes for each shadow blob. The main purpose for landscape generation is to provide a set of local maps which will be utilized for determining the automated self-supervision data in the building detection phase also, with the help of the fuzzy landscapes, the potential non-building objects are eliminated by verification with respect to vegetation and shadow length.
Bresenham‘s line discretization methodology, a new methodology which is a combination of a fuzzy and a binary structuring element has been proposed.
As discussed, buildings are not the only objects to cast shadows in satellite images. All objects with prominent height such as trees, vehicles, bridges, etc. also bring about the existence of shadows in the image. Since there is no relation between the size and shape of the shadow region and the reference object the shadow is cast by, the probability map is pruned by checking whether the shadow region is adjacent to vegetation in ά direction, and by exploiting height information

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

Although vegetated regions have been detected, there may be shadow regions which are cast by the vegetated areas. Since the local landscapes are generated independently for each shadow blob, the vegetated regions may also have relevance values in the final probability map.
In a VHR satellite image, the parameters such as the sensor viewing angle, the solar angles, the seasons and atmospheric conditions are largely beyond the control of the data user, and there is a limited opportunity for the user to dictate specific imaging dates, times and weather conditions for the image acquisition . Thus, the properties of the shadow regions may substantially differ from image to image and it is hard to predict the size and shape of the shadows. Besides, in an urban environment, buildings are not the only objects that cast shadows. Other objects such as trees, vehicles, garden walls, pools, and bridges also cast shadows. For those reasons, in an urban area, it is essential for a building detection task to eliminate the landscapes that may occur due to shadows cast by non-building objects. To eliminate the landscapes generated by distinct vegetation objects, for each shadow region, we investigate the vegetation evidence within the close neighborhoods of the shadow regions. To do that, we define a search region in the immediate vicinity of each shadow object whose extent are outlined after applying a double thresholding (Tlow, Thigh) to the fuzzy landscapes generated. In all our experiments, we used 0.7 and 0.9 fuzzy membership values respectively for Tlow and Thigh in image space, and once the search region is defined, we check for vegetation evidence within the defined region with the help of the pre-computed binary vegetation mask, MV. More specifically, we compute a ratio where the denominator is the total number of pixels defined in the region and the numerator is the total number of pixels labeled as vegetation in the vegetation mask. We reject a fuzzy landscape region generated from a cast shadow if the computed ratio is equal or larger than a ratio threshold (Tveg = 0.7). Fig. 6 illustrates the fuzzy landscapes generated from cast shadows before and after the vegetation analysis. To separate the landscapes of building and other non-building objects, we assess the height difference of the objects compared to the terrain height. Shadow information is also useful to extract the height information from monocular images, and already used in several studies for the estimation of the building heights, our aim is to investigate the length of the shadow objects in the direction of illumination to enforce a pre-defined height threshold value. To do that, for a given solar elevation angle (φ) and a minimum building height threshold (Theight), we compute the minimum shadow length (Lmin) that should be cast on a flat surface $Lmin{=}T\text{height}tan\emptyset$ If none of the perimeter pixels of a shadow object is found to be satisfying the length Lmin, we assume that the shadow is cast from a non-building object, and thus, the generated fuzzy landscape is rejected. In an urban area, most of the vehicles (e.g., cars, trams, and single-decker buses), garden walls and pools, and some of the bridges used rather than vehicular traffic have height differences of less than 3–4 m compared to the terrain height. Therefore, in this study, we utilize a single height threshold (Theight) defined in object space to detect and eliminate the landscapes generated by non-building objects.

C. Grab Cut Algorithm

Grab Cut is a semi-automated foreground/background partitioning algorithm. Given a group of pixels interactively labelled as foreground/background by the user, it partitions the rest of the pixels in an image using a graph-based approach but in proposed work we have removed this graph based approach which further uses min cut algorithm to iterate and reach to a minimum energy level. Now this iteration process is done by BFO optimization .The procedure applied in Grab cut is very similar to our work. The minor differences are:

1. The unary penalties are calculated by using Gaussian mixture models for pixel RGB values instead of histograms for gray-level intensities.
2. The energy minimization routine by using bacterial foraging optimisation.

Using RGB vectors Ip for pixels p labelled as α, the background (where α = B) and foreground (where α = F) are modelled using Gaussian mixture models. In order to obtain the mixture parameters wk, μ and ∑, an initial assignment of pixels p labelled as α = F or α = B to corresponding mixture components is utilized by k-means algorithm. Afterwards, the parameters for each component k are calculated as:

$$W_k^{(\alpha)} = |GMM_k(\alpha)| / \sum_k |GMM_k(\alpha)|$$

$$\mu_k^{(\alpha)} = \sum_{p \in GMM_k(\alpha)} I_P / |GMM_k(\alpha)|$$

$$\sum\nolimits_k(\alpha) = \frac{1}{|GMM_k(\alpha)|} \sum_{p \in GMM_k(\alpha)} (I_P - \mu_k^{(\alpha)})(I_P - \mu_k^{(\alpha)})^T$$

Where$\alpha \in \{F, \beta\}$, $GMM_k(\alpha)$ is the set of pixels assigned to k-th component of the GMM of α label and $|GMM_k(\alpha)|$ denotes the size of set$GMM_k(\alpha)$.
The unary penalty functions Up(α) for a pixel p, where $\alpha \in \{F, \beta\}$, is calculated as negative log-likelihood of Gaussian mixture models:

$$U_p(\alpha) = \sum_{k=1}^{K} U_P'(\alpha, k)$$

Where

$$U_P'(\alpha, k) = -log w_k^{(\alpha)} + \frac{1}{2} log \left|\sum\nolimits_k \alpha\right| + \frac{1}{2}(((I_P - \mu_k^{(\alpha)})^T \sum\nolimits_k \alpha)^{-1} (I_P - \mu_k^{(\alpha)}))$$

Below is the Grab cut algorithm step by step
1. (Initialization)

- With bounding box: Set T_F=0, T_B: outside the bounding box, T_U: the bounding box and its interior region

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

- With explicit trimap: pixels $p \in T_F, T_B, T_U$ are given explicitly with a mask with labels $\alpha_p = F$ if $p \in T_F \cup T_U$; $\alpha_p = 0$ if $p \in T_B$.

2. (Mixture component assignment) Assign each pixel p to a component for foreground or background GMM by an initial EM or K-means algorithm.
3. (Iteration) until convergence of GMMs;

(a) (GMM parameter computation) Calculate mixture parameters for GMMs of F and B.
(b) (Inference by Bacterial Foraging Optimization (BFO)) after defining energy function and unary / boundary penalties, optimize the clusters assigned in k means using BFO.
(c) (Reassignment of mixture components) Update the GMM component assignments to samples for background and foreground, with respect to the obtained result in inference step.

After the non-building regions have been detected and the probability map representing the proximity of pixels to the shadow regions in the opposite of illumination direction has been generated, the remaining problem is to combine this information in a framework to boost the building detection performance. In this work, Grab cut is determined to be a feasible algorithm in which initial foreground and background information can be integrated easily, interaction potentials between pixels with respect to probability map can be computed feasibly and the fast.
The steps of the proposed building detection algorithm are listed below:

- (Iteration over shadow components) For each shadow blob ζ_i detected in Section 4.5;

1. (Initialization: Local bounding box and trimap generation) Generate a bounding box BBi, which defines the boundary of the local image patch to be worked on, using the fuzzy landscape generated in Section 4.3. Simultaneously; for each pixel, determine which pixels are in TF and TB, and define initial value α_p for each pixel p inside BBi.
2. (Assignments to mixture components) For each pixel, assign pixel to a GMM component for either background or foreground
3. (Iterations over energy minimization) Same as explained above in 3rd step of this section.
4. (Refinement)

Simple post-processing operations are applied on the partitioning result, using connected component analysis.

The proposed algorithm runs independent, local 'Grab cut's over each shadow component. This is because of the fact that the most useful information about a building object is obviously obtained within a neighborhood of the object itself. In a satellite image, the assumption that two buildings far from each other share similar color properties can lead to poor results. For instance; there may be a red building covered with a blue environment in a part of an image, whereas a blue building covered with a red environment on another part of the image.

D. Bacterial Foraging Optimization

Bacteria Foraging Optimization Algorithm (BFOA) is a new comer to the family of nature-inspired optimization algorithms. For over the last five decades, optimization algorithms like Genetic Algorithms (GAs), Evolutionary Programming (EP), Evolutionary Strategies (ES), which draw their inspiration from evolution and natural genetics, have been dominating the realm of optimization algorithms. Recently natural swarm inspired algorithms like Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO) have found their way into this domain and proved their effectiveness. Application of group foraging strategy of a swarm of E.coli bacteria in multi-optimal function optimization is the key idea of the new algorithm. Bacteria search for nutrients in a manner to maximize energy obtained per unit time. Individual bacterium also communicates with others by sending signals. A bacterium takes foraging decisions after considering two previous factors. The process, in which a bacterium moves by taking small steps while searching for nutrients, is called chemo taxis and key idea of BFOA is mimicking chemo tactic movement of virtual bacteria in the problem search space.
During foraging of the real bacteria, locomotion is achieved by a set of tensile flagella. Flagella help an E.coli bacterium to tumble or swim, which are two basic operations performed by a bacterium at the time of foraging. When they rotate the flagella in the clockwise direction, each flagellum pulls on the cell. That results in the moving of flagella independently and finally the bacterium tumbles with lesser number of tumbling whereas in a harmful place it tumbles frequently to find a nutrient gradient. Moving the flagella in the counter clockwise direction helps the bacterium to swim at a very fast rate. In the above-mentioned algorithm the bacteria undergoes chemo taxis, where they like to move towards a nutrient gradient and avoid noxious environment. Generally the bacteria move for a longer distance in a friendly environment. Figure 3.3 depicts how clockwise and counter clockwise movement of a bacterium take place in a nutrient solution.

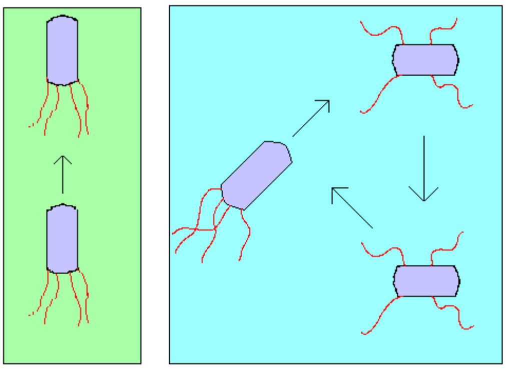

Figure 3. Swim and tumble of a bacterium

When they get food in sufficient, they are increased in length and in presence of suitable temperature they break in the middle to from an exact replica of itself. This phenomenon inspired Passino to introduce an event of reproduction in BFOA. Due to the occurrence of sudden

DOI 10.17148/IJARCCE.2015.4833

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

environmental changes or attack, the chemo tactic progress may be destroyed and a group of bacteria may move to some other places or some other may be introduced in the swarm of concern. This constitutes the event of elimination-dispersal in the real bacterial population, where all the bacteria in a region are killed or a group is dispersed into a new part of the environment.

Thus search for food of E.Coli can be categorised into four steps: Chemo tactic, Swarming, Reproduction and Killing/ Dispersion. Mathematically these can be represented step by step as:

Chemo tactic:

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + c(i)\frac{\Delta(i)}{\sqrt{\Delta^t(i)\Delta(i)}}$$

Where $\theta^i(j,k,l)$ represents ith bacterium at jth chemo tactic, k-th reproductive and l-th elimination-dispersal step. C(i) is the size of the step taken in the random direction specified by the tumble (run length unit).

Swarming:

J (i, j, k, l) = J (i, j, k, l) + J_{cc} (θ (j, k, l), P (j, k, l)

Where J (i, j, k, l) is the fitness function.

BFO Algorithm: steps for BFO algorithm are discussed as follows

Parameters:

[Step 1] Initialize parameters p, S, Nc, Ns, Nre, Ned, Ped, C(i)(i=1,2…S),

Algorithm:

[Step 2] Elimination-dispersal loop: l=l+1

[Step 3] Reproduction loop: k=k+1

[**Step 4**] Chemo taxis loop: j=j+1

[a] For i =1,2…S take a chemo tactic step for bacterium i as follows.

[b] Compute fitness function, J (i, j, k, l).

Let, J (i, j, k, l) = J (i, j, k, l) + J_{cc} (θ(j, k, l),P(j, k, l)) (i.e. add on the cell-to cell attractant–repellant profile to simulate the swarming behaviour)

[c] Let Jlast=J (i, j, k, l) to save this value since we may find a better cost via a run.

[d] Tumble: generate a random vector D(i)ÎÂp with each element (i),m 1,2,..., p, m D = a random number on [-1, 1].

[e] Move: Let

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + c(i)\frac{\Delta(i)}{\sqrt{\Delta^t(i)\Delta(i)}}$$

This results in a step of size C(i) in the direction of the tumble for bacterium i.

[f] Compute J (i, j +1, k, l) and let

J (i, j 1, k, l) = J (i, j, k, l) +J_{cc} (θ(j 1, k, l), P(j 1, k, l))

[g] Swim:

i) Let m=0 (counter for swim length).

ii) While m< s N (if have not climbed down too long).

• Let m=m+1.

• If J (i, j +1, k, l) < Jlast (if doing better), let Jlast = J (i, j +1, k, l) and let

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + c(i)\frac{\Delta(i)}{\sqrt{\Delta^t(i)\Delta(i)}}$$

Use this θ(j +1, k, l) to compute the new J (i, j +1, k, l) as we did in [f]

• Else, let m= s N. This is the end of the while statement.

[h] Go to next bacterium (i+1) if i ≠ S (i.e., go to [b] to process the next bacterium).

[Step 5] If c j < N , go to step 4. In this case continue chemo taxis since the life of the bacteria is not over.

[**Step 6**] Reproduction:

[a] For the given k and l, and for each i = 1,2,..., S , let

$$J_{health} = \sum_{j=1}^{N+1} J(i,j,k,l)$$

be the health of the bacterium i (a measure of how many nutrients it got over its lifetime and how successful it was at avoiding noxious substances). Sort bacteria and chemo tactic parameters C(i) in order of ascending cost health J (higher cost means lower health).

[b] The r S bacteria with the highest J_{health} values die and the remaining r S bacteria with the best values split (this process is performed by the copies that are made are placed at the same location as their parent).

[**Step 7**] If k < N_{re} , go to step 3. In this case, we have not reached the number of specified reproduction steps, so we start the next generation of the chemo tactic loop.

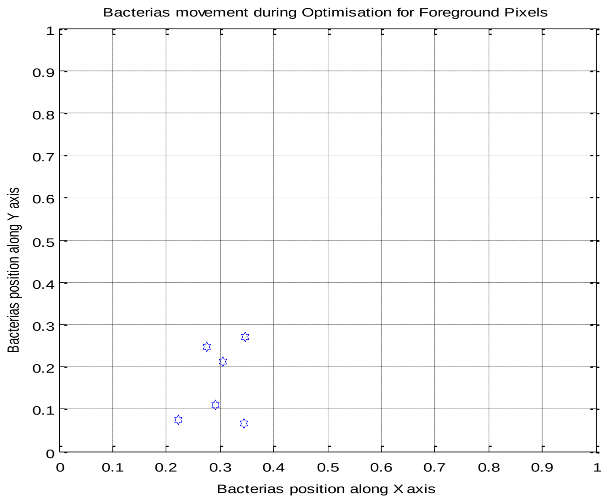

Figure 4. Final positions of bacteria for foreground pixels clustering

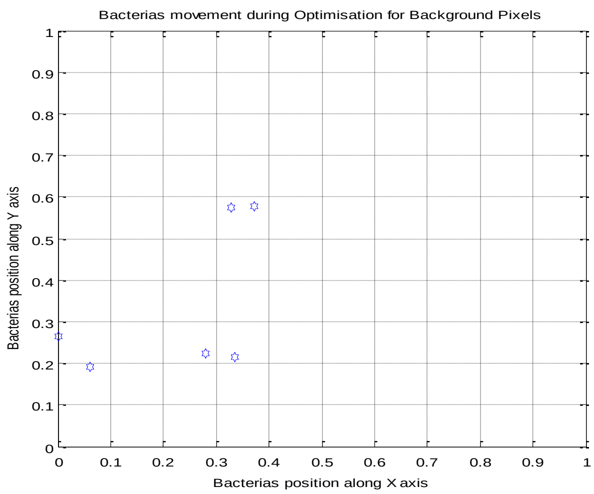

Figure 5. Final positions of bacteria for background pixels clustering

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

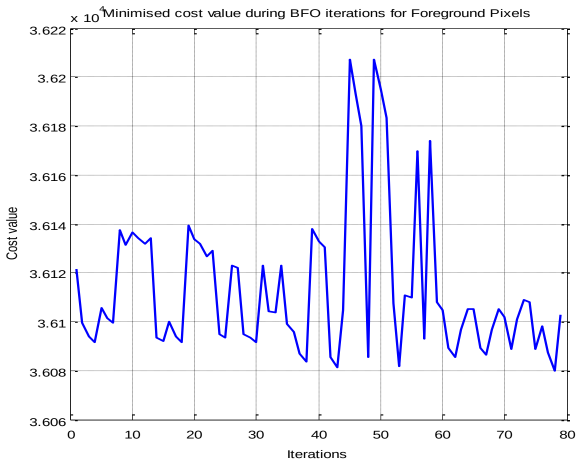

Figure 6. Cost value decreasing for foreground pixels

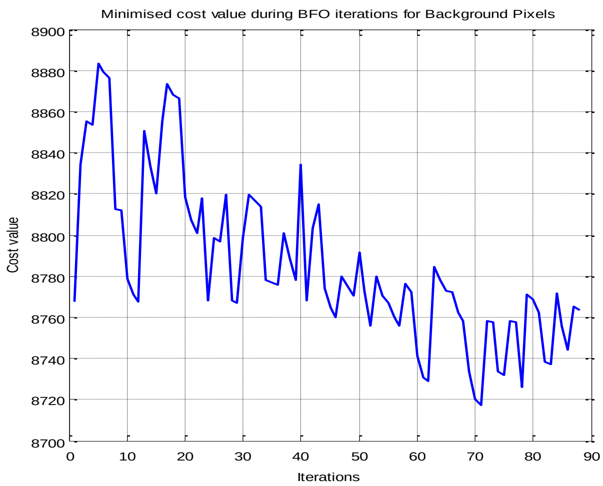

Figure 7. Cost value decreasing for background pixels

Figure 8. Building detected after proposed algorithm

III.CONCLUSION

In this work, a new automated building detection methodology for satellite images is proposed. In this methodology; first the vegetation, water and shadow regions are detected from a given satellite image, and local fuzzy landscapes are generated from the shadow regions using the direction of illumination obtained from image metadata. Afterwards, for each fuzzy landscape, foreground and background pixels are automatically determined and a bi partitioning is obtained using a graph-based algorithm called Grab cut. The grab cut method is not automatic, it need human intervention and plotting of bounding box in the image for the area which need to segment. We have made this process fully automated by fuzzy landscape which made blobs of shadow areas. Further Gibb's energy minimization iteration in grab cut is done using bacterial foraging optimization which is not done in any paper as per our knowledge. In our work no manual supervision of user interaction is required. Instead; the supervision data is automatically generated inside the algorithm, using shadow cues.

IV. FUTURE WORK

The cons of the proposed algorithm are as follows. At first, the algorithm is eminently dependent to shadow detection. If the shadow of a building cannot be detected, then the detection of that building using the proposed methodology is not possible. However; considering the shadow detection performances over varying environments, this is an infrequent issue. Also; the algorithm cannot efficiently detect buildings in regions where the shadow of a building falls onto another building, or where self-occlusion occurs for building rooftops. Moreover; in some test cases, there may exist non-building objects such as road segments and bridges that cannot be eliminated using height verification and mistakenly detected as buildings. The final limitation of the algorithm is that in some residential areas where buildings are located in a dense and crowded manner, multiple separate buildings can be under-detected as a single building.

For detecting buildings in satellite images, more object invariants can be integrated into the architecture. One of these invariants can be considered as the star shape prior, where star shape is described as "for any point p inside the object, all points on the straight line between the centre c and p also lie inside the object". This shape prior can easily be integrated by changing the boundary penalty term of the energy function of MRF. More generally; given a shape prior who's defining curve can be parameterized; this shape prior can be added to the graph cut partitioning framework by only changing the energy function.

Another improvement may be obtained by estimating the initial contour of the building candidate. For the estimation; while the approaches like snakes or level sets have a problem of sticking to the local minimum, the active segmentation algorithm first computes the polar transformation of the image, and then computes the globally optimum contour by utilizing graph cut on the polar edge map. This approach may be useful for a better determination of tramp.

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

REFERENCES

[1]. M. Salih M. Kadhim," Shadow Detection From Very High Resoluton Satellite Image Using Grabcut Segmentation And Ratio-Band Algorithms" The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-3/W2, 2015 PIA15+HRIGI15 – Joint ISPRS conference 2015.

[2]. D. Koc San," Building extraction from high resolution satellite images using hough transform" International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science, Volume XXXVIII, Part 8, Kyoto Japan 2010.

[3]. S.Praveena,"Hybrid Clustering Algorithm and Feed-Forward Neural Network for Satellite Image Classification" International Journal of Engineering Science Invention Volume 3 Issue 1‖ January 2014.

[4]. Beril Sırmacek, "Building Detection Using Local Gabor Features in Very High Resolution Satellite Images" Pattern Recognition Letters, vol. 25, no. 3, 2008.

[5]. Basavaprasad B., "Improved GrabCut Technique for Segmentation of Color Image"International Journal of Computer Applications (0975 – 8887) Recent Advances in Information Technology, 2014.

[6]. Shiyan Pang, "Object-Based Analysis of Airborne LiDAR Data for Building Change Detection" Remote Sens.2014, 6, 10733-10749; doi:10.3390/rs61110733.

[7]. Shabnam Jabari, "Building Detection In Very High Resolution Satellite Image Using IHS Model" ASPRS 2014 Annual Conference Louisville, March 23-28, 2014.

[8]. Ali Ozgun Ok, "Automated Detection of Arbitrarily Shaped Buildings in Complex Environments From Monocular VHR Optical Satellite Imagery" IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, VOL. 51, NO. 3, MARCH 2013.

[9]. HUERTAS, "Detecting Buildings in Aerial Images" COMPUTER VISION, GRAPHICS, AND IMAGE PROCESSING 41,131-152 (1988).

[10]. R. Nevatia, "A System for Building Detection from Aerial Images" IEEE Computer Vision and Pattern Recognition Conference, 2002.

[11]. Nicholas Shorter, "Automatic Vegetation Identification and Building Detection from a Single Nadir Aerial Image" Remote Sens. 2009, 1, 731-757.

[12]. Jiaojiao Tian, "Building Change Detection Based on Satellite Stereo Imagery and Digital Surface Models" IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, VOL. 52, NO. 1, JANUARY 2014.

[13]. Gui-Song Xia, "STRUCTURAL HIGH-RESOLUTION SATELLITE IMAGE INDEXING", IAPRS, Vol. XXXVIII, Part 7A, July 5–7, 2010.

[14]. Madhuri Kalapala, "Estimation of Tree Count from Satellite Imagery through Mathematical Morphology", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 1, January 2014.

[15]. Zhisheng Gao1, "A mutual GrabCut method to solve co-segmentation" Gao et al. EURASIP Journal on Image and Video Processing 2013.

[16]. Yuri Y. Boykov," Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images" Proceedings of "Internation Conference on Computer Vision", Vancouver, Canada, July 2001.

[17]. Carsten Rother , "Grab Cut" — Interactive Foreground Extraction using Iterated Graph Cuts ,In Proc. IEEE Conf. Computer Vision and Pattern Recog, 2009.

[18]. Kunio Kashino, "Tri-Map Self-Validation Based on Least Gibbs Energy for Foreground Segmentation" IEEE Trans. Pattern Anal. Mach. Intell., 2014.

[19]. Victor Lempitsky, "Image Segmentation by Branch-and-Mincut" ECCV, Part IV, LNCS 5305, pp. 15–29, 2008.

[20]. Anders P. Eriksson, "Image Segmentation Using Minimal Graph Cuts" IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 888-905, 2002.

[21]. S. Aksoy and H. Akc¸ay. Image Classification and Object Detection Using Spatial Contextual Constraints. CRC Press, 2012. ISBN 978-1-43985-596-6.