Rooftop Detection using a Corner-Leaping based Contour Propagation Model

Masoud S. Nosrati and Parvaneh Saeedi
School of Engineering Science, Simon Fraser University
e-mail: smn6@sfu.ca, psaeedi@sfu.ca

***Abstract*—Extracting building rooftops in satellite/aerial images is one of the most challenging problems in the application of computer vision for remote sensing. In this paper a new contour propagation model for rooftop boundary detection is proposed. It includes developing contour models that evolve by leaping on image corners and edge points while minimizing an energy function based on image corner responses, image color invariants and edge points. The proposed method is capable for coping with the complications associated with the gabled rooftop using Gaussian color invariance modeling. Experimental results for aerial/satellite images show that the average shape accuracy is above 90% for the test images of sub-urban areas.**

***Keywords*—Rooftop extraction, object detection, remote sensing, Gaussian invariant model**

I. Introduction

Creating 3D maps of cities is of high interest by many industries and organizations. Applications of 3D city maps ranges from civilian purposes such as insurance risk assessment and disaster management to security based ones such as site monitoring and military tactic planning. 2D rooftop extraction plays a critical role in the 3D map generation.

In recent years various methods and systems have been developed for rooftop extraction. Some researchers restricted their methods to simple models such as quadrilateral shapes [1] while others try to deal with more complicated shapes by incorporating additional information such as Digital Elevation Maps (DEM) or Digital Surface Models (DSM), Lidar height data and stereo images. For instance, Ruther et al. [2] proposed a semi-automatic approach using DSM to generate the initial raised structure hypotheses. These hypotheses are refined later using active contour model. Florent Lafarge et al. [3] also used DEMs and proposed an object-based approach for building extraction.

Some other approaches have focused on image features and rooftop surface characteristics to segment rooftops or identify their boundaries using curve evolution methods such as active contours, snake and deformable models. In the method by Woo et al. [4], primary hypotheses are selected by extracting useful building profile information from generated disparity map. The algorithm is followed by forming a graph and graph-based search techniques for rooftop extraction. Li et al. [5] combined different techniques such as image segmentation, region growing, and morphological methods for detecting building rooftop with bright intensity values. Wei et al. [6] used shadows and their directions as clues to verify the presence of building structures. An unsupervised clustering algorithm was proposed to separate shadows from other parts of a scene. The building's boundaries were refined using Canny edge detector and Hough transform. Jin and Davis [7] proposed an automatic system that utilized structural, contextual and spectral information to detect buildings in satellite imagery. Peng et al. [8] introduced an improved snake model based on radiometric and geometric behaviors of buildings and Mayunga et al [9] proposed a semi automatic approach using radial casting algorithm to initialize snake based contours.

While all the above methods present some good results for specific type of buildings (rooftops) they are not capable of addressing general complications that exist in this problem. For example the use of external data (DEMs or DSMs) which generally has low resolutions (within tens of meters) is useful for very large size building structures. Lidar height-based methods could be an expensive solution with a low update rate. Assuming simple rooftop profiles for buildings clearly is not sufficient for detecting buildings with complicated footprints. Assumptions such as light color and flat rooftops are also too restrictive to be applicable on the huge resource of satellite imageries. The attempt in this paper is to address the issue of building detection with minimum number of assumptions and restrictions about rooftops.

This paper presents a new deformable method for building rooftop extraction. In this method the deformable model of the rooftop boundaries evolves by leaping on corners while minimizing an energy function. In addition to getting better results the proposed method is capable of dealing with gabled rooftops.

II. Methodology

The proposed work in this paper is based on a deformable model that evolves by leaping on corners while minimizing an energy function until the model fits to the rooftop boundary. The dataflow of the proposed method is depicted on Figure 1.

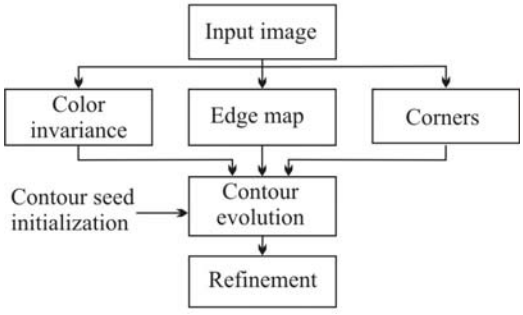


Fig. 1. Dataflow of the proposed algorithm.

Detail of each process is presented next.

A. Image features

Three types of image features are incorporated in this work: Color invariant image, Edge map image and Corners. Each one of these features has a critical role in a proper identification of the scene's rooftops.

1) Color invariants:

Detection of gabled rooftops is a challenging problem due to the color/intensity variation of different components of the rooftop that often occurs when the sun/camera is not perpendicular to the earth or when the slops of the rooftop pieces are too steep. For this reason, a color enhancing process is utilized to reduce such effects. Geusebroek et al. [11] proposed a Gaussian color invariance model for various conditions (including both illumination and reflectance) for creating an illumination/geometrical invariant model of scene images. One of the cases that they studied was for the condition under which the scene that included matte/dull surfaces was illuminated by a source with equal energy but unevenly. This model seems reasonable for this work since building rooftops are made of matte/dull materials. Moreover the sun light energy is equal for all objects in the scene but the gabled rooftop pieces do not reflect the light evenly. The above model is implemented in this work. Based on the model proposed in [11] the reflected spectrum in the viewing direction is given by:

$$E(\lambda, x) = i(x) R_{\infty}(\lambda, x) \tag{1}$$

Here x denotes the position at the imaging plane and λ is the wavelength. $i(x)$ is the intensity variation and $R_{\infty}(\lambda,x)$ is the material reflectivity. Using the above reflected spectrum the two following parameters are defined:

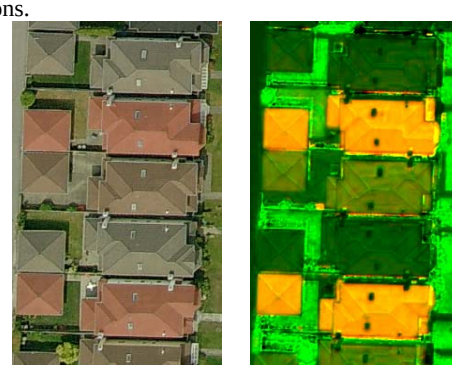
$$C_{\lambda} = \frac{E_{\lambda}}{E},\ C_{\lambda\lambda} = \frac{E_{\lambda\lambda}}{E} \tag{2}$$

Here E_{λ} and $E_{\lambda\lambda}$ are the first and second derivatives with respect to the wavelength. C_{λ} describes object's color regardless of its intensity and $C_{\lambda\lambda}$ is the higher order spectral derivative of C_{λ} which is the object reflectance property under an equal energy illumination.

The Gaussian color model can be computed using the RGB image by the following transformation:

$$\begin{bmatrix} E \\ E_{\lambda} \\ E_{\lambda\lambda} \end{bmatrix} = \begin{pmatrix} 0.3 & 0.58 & 0.11 \\ 0.25 & 0.25 & -0.5 \\ 0.5 & -0.5 & 0 \end{pmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3}$$

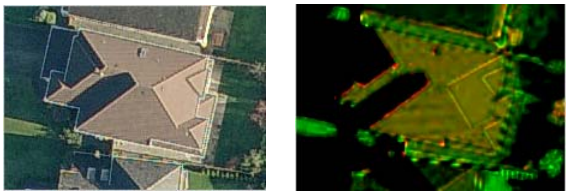
To display the Gaussian color model, C_{λ} and $C_{\lambda\lambda}$ are placed in G and R channels and the value of B channel is set to zero. Figure 2 shows an aerial scene (a) and its corresponding Gaussian color model (b). From this image it can be seen that darker (shaded) regions of the gabled rooftop are highly neutralized by the illumination invariant Gaussian model. It is important to note that while the above color invariant model performs well for the presented image, there is a limit to the amount of shading that could be neutralized. If the sun is at or near horizon or if for any reason the shading of rooftop pieces are strong, the model is simply incapable of coping with such variations.



(a) (b)

Fig. 2. (a) RGB image. (b) The illumination invariant Gaussian model of image (a).

For instance Figure 3 (a) represents a case in which the sun is located near horizon. As displayed some of the components of this rooftop are darker than others. While the presented model is capable of dealing with moderate color changes (in the middle of the rooftop), it is completely incapable of coping with the color variations on the left side components of the rooftop. The results of the color invariant model are presented on the Figure 3 (b).



(a) (b)

Fig. 3. Limitation of the color invariance model in correcting intensity/color distortions for gabled rooftops.

In this work, C_{λ} and $C_{\lambda\lambda}$ are utilized to measure the similarity of color regions inside rooftop definitions. The following equation is incorporated for measuring color variation (φ) inside any region of interest c:

$$\varphi = \oint_{C} \left(C_{\lambda}(i,j) - m_{C\lambda}\right)^2 + \oint_{C} \left(C_{\lambda\lambda}(i,j) - m_{C\lambda\lambda}\right)^2 \quad (i,j) \in \textit{inside the region } c \tag{4}$$

Here $m_{C\lambda}$ and $m_{C\lambda\lambda}$ are means of C_{λ} and $C_{\lambda\lambda}$ inside region c.

2) Edge:

Edge data is an important piece of information in the proposed algorithm as it generally creates a distinguishable boundary between the rooftop and its surroundings. Canny edge detector with high and low thresholds of 0.1 and 0.04 is applied on the input image. The edges are first linked together and then the edge segments are converted into straight lines to create a clean *Edge map* image:

3) Corner:

In order to detect image corners Harris corner detector [10] is implemented. To make sure that no corner is missed, the sensitivity of the Harris corner detector is increased. In this work, the standard deviation of smoothing Gaussian is set to 1 and the corner response threshold is set to 600. This setting would cause corners to be over called. Once extracted, corners are evaluated and assessed according to their Harris corner response and their color variation.

$$Corner\ response\ (o) = Harris(o) + \varphi_{3x3}(o) \quad (5)$$

Here *Harris (o)* is the Harris response for the corner *o*. The value of $\varphi_{3\times3}$ is calculated using equation (4) in which *c* is a *3×3* image window centered at corner *o*.

Figure 4 shows detected corners for a sample image. For display purposes the corners are shown with three colors. The top 30% strongest corners are shown in red while the bottom 30% weakest corners are shown in blue. The remaining corners are shown in yellow. The radius of the circle for each corner is proportional to the corner's strength. Only the top 70% of the corners are maintained as corner features and the remaining are eliminated.

B. Contour evolution

Before detailing the proposed algorithm, some definitions are presented.

- *Control point*: A contour formed by a number of control points and the lines connecting every two consecutive control points.

-*Edge point*: Any edge pixel in the *Edge map* image.

- *Anterior Area*: A sector situated in the front of a control point and outside the contour. The anterior area is defined by two parameters: radius (ρ) and the deviation angle from the external bisector (θ). In this work ρ is set to 20 pixels and θ is set to 45 degrees.

- *Control point patch*: A small sector (radius of 8 pixels) between the control point's two sides inside the contour definition.

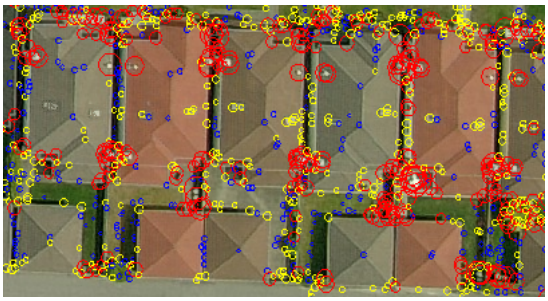


Fig. 4. Detected Corners ($r_{Red} > r_{Yellow} > r_{Blue}$)

Visual presentations of above terms are shown in Figure 5. Since the proposed algorithm is a semi-automatic one, an initialization step is required in which an initial seed contour is placed on the rooftop. This initial contour is a very small contour usually spans about 4 to 8 pixels in width and length and can have any number of control points starting from 3.

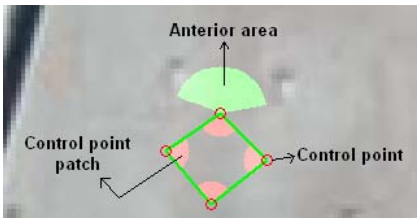


Fig. 5. Visual descriptions of the used terminologies.

Once this contour is initialized it evolved iteratively until the final contour is reached.

1. In every iteration of the contour evolution process and for each control point a search is performed which finds all corners and edge points within the control point's anterior area. If however no corner or edge point is found, a point on the border of the anterior area (located at a distance of ρ) is selected as a candidate position. This allows the contour to grow when it is too small or the area outside the contour is too smooth.

2. After finding all candidate positions, the control point will leap onto candidate positions and at each position the following energy function is computed:

$$E_n = \alpha \cdot \frac{d\varphi_n}{dn} + \beta \cdot A_n \quad (6)$$

The position that minimizes the above energy function is considered as the new control point. In this equation E_n is the energy in the n^{th} iteration. φ_n is the color variation inside the contour and is calculated using equation (4). $\frac{d\varphi_n}{dn}$ is the difference between the current color variation and that of the previous iteration. A_n is the area of the contour. The first term in equation 6 is the main minimizing term while the second one is a regularizing term. In this work α and β are constants and are set (empirically) to 0.4 and -0.2 respectively.

3. When evolving, if a control point touches an edge point, that control point is split into two control points along the touched edge. For this the algorithm searches for the two best possible edge point positions (potentially corners at the edge two ends) by calculating the energy function in two opposite directions along the touched edge (Figure 6). This strategy allows addition of control points to fit the profile of building rooftops with complicated shapes and high number of vertices.

4. The control points are updated one by one until no more update can be made.

5. After each iteration, the contour is checked to find if any connecting line between two consecutive control points intersects with any other lines in the contour definition. If such condition is detected, the control points for the intersecting lines are re-ordered so that no two contour

lines intersect inside the rooftop profile.

6. Steps 1 to 5 are repeated until the contour cannot evolve any further.

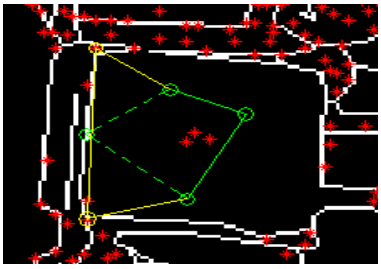


Fig. 6. A control point (green) is split into two control points (yellow) if touching an edge.

C. Refinement

After the contour is evolved, some control points might not be located on any corner or edge point. This is due to the fact that if the algorithm does not find any corner or edge point in anterior area, it would allocate a point on the anterior border to force the contour to grow. A refinement process is implemented here that asses and improves the locations of control points. The main idea here is to search in both anterior and posterior of each control point for potential corner/edge point substitutes. Following steps describe the refinement process:

1. Increase the number of control points in the contour definition. Add a control point(s) whenever the distance between the two consecutive control points is larger than 10 pixels.
2. Compute the mean values for all C_λ and $C_{\lambda\lambda}$ channels inside the contour (μ).
3. Compute a color distance (D) between image pixels inside the contour from the mean value (μ).

$$D_{Contour}(l) = \left\| I(l) - \mu \right\| \tag{7}$$

I is the color invariant image inside the contour and l is the l^{th} pixel within I. $D_{Contour}$ is the vector of distances. The standard deviation of $D_{Contour}$ provides a measure of color distance inside the rooftop candidate contour.

4. Find all corners/edge points in the anterior and the posterior of each control point (Figure 7). The parameters of posterior and anterior areas at this stage are set to ρ = 10 pixels and θ = 20 degrees. The candidate point set is called $X=\{x_1, x_2,\ldots\}$.
5. Move the current control point to each candidate position (x_k) and compute the color distance for the corresponding *Control point patch* (equation (8)) Remove candidates with large distances (equation (9)).

$$D_{x_k}(i,j) = \frac{\sum_{i,j} \left\| Patch_{x_k}(i,j) - \mu \right\|}{Area\ of\ the\ control\ patch}$$
$$i, j : pixel's\ coordinates\ inside\ control\ patch\ x_k \tag{8}$$

$$Best\ Candidates = \left\{ x \mid D_x \le (4 \times std(D_{Contour})) \right\} \tag{9}$$

6. Among all the chosen candidates the one that minimizes the following energy function will be chosen and the control point's position will be updated accordingly.

$$E_{refinement} = \lambda_1 \cdot E_{Edge} + \lambda_2 \cdot A$$
$$E_{Edge} = \sum_{i,j} \varepsilon(i,j),$$
$$i, j \in Contour\ boundary\ pixels \tag{10}$$
$$Inverse\ Edgemap = \varepsilon = 1 - Edgemap$$

Here λ_1 is set to 1 and λ_2 is set to 0.5. E_{Edge} is the edge energy of the contour which is defined by summation of all pixels in the Inverse *Edge map image* (ε) at the corresponding contour boundary locations.

7. Steps 4 to 6 are repeated until the contour cannot evolve any further.

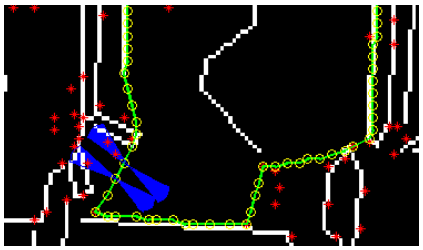


Fig. 7. Searching for corners and edge points in anterior and posterior regions (shown in blue) of each control point (red points: corners, yellow circles: control points).

III. Experimental Results

The proposed method is tested on 20 different satellite (0.6 m/p) and aerial (0.16 m/p) images. The code was developed in C++ and MATLAB and run on a 2Quad Intel CPU 2.4 GHz machine. To assess the performance quality of the proposed algorithm Shape Accuracy (SA) [13], Area Overlap Error (AOE) and Relative Absolute area Difference (RAD) measures are computed for each extracted rooftop. Table 1 represents these results. In this table the first 15 test images are aerial and the last 5 are satellite images. A brief description of each measure is presented next.

Shape Accuracy [12] (SA) [%]: In the Shape Accuracy formula the area of each building in the manually found ground truth is compared against the area of the detected rooftop by the proposed method. A_{GT} and A_{DB} are areas of a building in the ground truth and the detected hypothesis respectively.

$$Shape\ Accuracy = 1 - \frac{\left| A_{GT} - A_{DB} \right|}{A_{GT}} \times 100 \tag{11}$$

Area Overlap Error [12] (AOE) [%]: It is the percentage of the number of pixels in the intersection of detected contour and the ground truth contour divided by the number of pixels in the union of the two. This value is 0 for a perfect detection and

100 for the worst case scenario. Here the term mask is used for the image area inside each contour.

$$AEO = \left(1 - \frac{\sum_{i,j} (mask_{GT} \cap mask_{DB})}{\sum_{i,j} (mask_{GT} \cup mask_{DB})} \right) \times 100 \qquad (12)$$

Relative Absolute area Difference [12] (RAD) [%]: The total area difference between the detected contour and the ground truth is divided by the total area of the ground truth. The negative value of this measure shows under-segmentations and the positive values show over-segmentation. To compute the mean, the absolute values are taken. Note that the perfect value of 0 can also be obtained for non-perfect results since this score is based on the total areas.

$$RAD = \left(\frac{\sum_{i,j} (mask_{DB} - mask_{GT})}{\sum_{i,j} mask_{GT}} \right) \times 100 \qquad (13)$$

Table 1: Quantitative assessment of the proposed method

Scene No.	No. of Bldgs.	Min SA [%]	Max SA [%]	Mean SA [%]	Mean AOE [%]	Mean RAD [%]
1	10	86.94	99.97	95.61	5.61	0.65
2	17	81.80	99.59	89.68	9.56	6.37
3	4	95.29	99.72	98.05	6.14	-3.381
4	15	82.33	96.94	89.62	9.94	7.84
5	6	89.39	96.03	92.75	7.42	-0.63
6	6	88.60	98.00	94.77	7.42	-1.24
7	11	71.16	95.59	84.86	9.72	8.37
8	4	84.10	92.15	88.94	8.29	-3.76
9	4	91.48	98.41	95.04	9.89	-6.04
10	13	81.14	98.94	89.90	12.81	7.03
11	14	80.30	99.79	87.41	11.54	7.67
12	13	54.86	99.72	85.87	11.07	5.94
13	18	46.85	98.83	88.75	13.29	2.12
14	7	89.08	99.68	96.36	7.91	-4.15
15	14	68.11	98.39	86.51	18.35	-1.30
16	6	75.60	98.73	89.26	18.8	-2.83
17	4	93.85	99.92	97.08	12.36	-1.11
18	4	86.21	95.52	92.38	14.62	5.67
19	6	87.06	97.52	92.32	14.08	3.45
20	8	76.51	99.54	91.44	11.57	6.83
Mean	-	-	-	**91.33**	**11.02**	**4.31**

Figures 8 to 14 show some of the detected results. From Table 1, the mean accuracy of the proposed method is 91.3% which is better that the values reported by [2] (80%) and [8] (83.6%).

Fig. 8. Output contours for scene No.1 (Aerial image).

Fig. 9. Output profiles for scene No. 4 (Aerial image).

Fig. 10. Output profiles for scene No. 7 (Aerial image).

Fig. 11. Output profiles for scene No. 10 (Aerial image).

Figu. 12. Output profiles for scene No. 12 (Aerial image).

Fig. 13. Output profiles for scene No. 19 (Satellite image).

Fig. 14. Output profiles for scene No. 20 (Satellite image).

A problem with this approach is in its sensitivity to the location/attribute of the initial seed. For instance, for a rooftop with non-uniform reflective surface the seed at a location with maximum amount of reflection variation could results in a good contour. While choosing an initial seed at a location with a uniform surface could result in an incomplete contour.

IV. Conclusion and Future Works

A new deformable model for rooftop extraction is proposed in this paper. The model is based on corners, color image invariants, edge points and an energy minimization scheme. The system requires an initialization step in which a small contour with a minimum 3 control points is initialized on the building of interest. A color invariant model is utilized to neutralize the color of slopped surface pieces of each gabled rooftop. Harris corners and Canny edges are detected for each image. The initialized contour grows by leaping on corners and edge points while an energy function is minimized. The energy minimization is performed by taking into account the similarity of image areas in the color invariant image, corners strength and the edge response. The algorithm is applied on satellite and aerial images and experimental results validate the quality of the proposed algorithm.

The future direction for this work includes automation of the contour initialization step and improvement of the accuracy.

Acknowledgment

The authors extend their thanks to the R&D department of MacDonald, Dettwiler and Associates Ltd. for providing access to the satellite and aerial imageries for this work. Authors would also like to acknowledge with gratitude NSERC Canada for support through the NSERC Strategic Grant Program.

References

[1] P. Saeedi, and H. Zwick, "Automatic building detection in aerial and satellite images," *ICARCV, proceedings: 623-629, 2008.*

[2] H. Ruther, H.M. Martine and E.G. Mtalo, "Application of snakes and dynamic programming optimization technique in modeling of buildings in informal settlement areas," *ISPRS J. 56:269–282, 2002.*

[3] F. Lafarge, X. Descombes, J.B. Zerubia, and M. Pierrot Deseilligny, "Automatic building extraction from DEMs using an object approach and application to the 3d-city modeling," *JPRS, 63(3):365–381, 2008.*

[4] D.M. Woo, Q.D. Nguyen, Q.D.N. Tran, D.C. Park, and Y.K. Jung, "Building detection and reconstruction from aerial images," In *ISPRS Congress,, XXXVII (B3b), 2008.*

[5] H.Y. Li, H.Q. Wang, and C.B. Ding, "A new solution of automatic building extraction in remote sensing images," *IGARSS, proceedings: 3790–3793, 2006.*

[6] Y. Wei, Z. Zhao, and J. Song, "Urban building extraction from high-resolution satellite panchromatic image using clustering and edge detection," *IGARSS, proceedings:. 2008–2010, 2004.*

[7] X. Jin and C.H. Davis, "Automated building extraction from high-resolution satellite imagery in urban areas using structural, contextual, and spectral information," *EURASIP JASP, 2006: 2196–2206, 2005.*

[8] J. Peng, D. Zhang, and Y. Liu, "An Improved snake model for building detection from urban aerial images," *Pattern Recognition Letter, 26(5): 587-595, 2005.*

[9] S.D. Mayunga, D.J. Coleman, and Y. Zhang, "Semi-Automatic System for Building Extraction in Dense Urban Settlement Areas From High-Resolution Satellite Imagery," *ISPRS, XXXVI (8/W27) , 2005.*

[10] C. Harris and M. Stephens, "A combined corner and edge detector," *Proceeding 4'th Alvey Vision Conference, proceedings: 147–151, 1988.*

[11] Jan-Mark Geusebroek, Rein van den Boomgaard, Arnold W.M. Smeulders and Hugo Geerts, "Color Invariance," *IEEE Trans. on PAMI, 23(12), 2001.*

[12] B. van Ginneken, T. Heimann, and M. Styner, "3D Segmentation in the Clinic: A Grand Challenge", MICCAI Wshp. 3D Segmentation in the Clinic: A Grand Challenge, 2007.

[13] D.M. McKeown, , G.E. Bulwindle, S.D. Cochran, W. A. Harvey, J.C. McGlone and J.A. Shufelt, "Performance evaluation for automatic feature extraction *ISPRS, 33 (Part B2):379-394, 2000.*