Wildfire Perimeter Detection via Iterative Trimming Method*

Li Tan[1], Yangsheng Hu[1], Shuo Tan[1], Raymond A. de Callafon[1], and Ilkay Altıntaş[2]

[1] Dept. of Mechanical and Aerospace Engineering
University of California San Diego, La Jolla, CA, U.S.A.
{ltan,yah071,sht001,callafon}@eng.ucsd.edu
[2] San Diego Supercomputer Center
University of California San Diego, La Jolla, CA, U.S.A.
{ialtintas}@ucsd.edu

Abstract. The perimeter of a wildfire is essential for prediction of the spread of a wildfire. Real-time information on an active wildfire can be obtained with Thermal InfraRed (TIR) data collected via aerial surveys or satellite imaging, but often lack the actual numerical parametrization of the wildfire perimeter. As such, additional image processing is needed to formulate closed polygons that provide the numerical parametrization of wildfire perimeters. Although a traditional image segmentation method (ISM) that relies on image gradient or image continuity can be used to process a TIR image, these methods may fail to accurately represent a perimeter or boundary of an object when pixels representing high infrared values are sparse and not connected. An ISM processed TIR image with sparse high infrared pixels often results in multiple disconnected sub-objects rather than a complete object. This paper solves the problem of detecting wildfire perimeters from TIR images in three distinct image processing steps. First, Delaunay triangulation is used to connect the sparse and disconnected high-value infrared pixels. Subsequently, a closed (convex) polygon is created by joining adjacent triangles. The final step consists of an iterative trimming method that removes redundant triangles to find the closed (non-convex) polygon that parametrizes the wildfire perimeter. The method is illustrated on a typical satellite TIR image of a wildfire, and the result is compared to those obtained by traditional ISMs. The illustration shows that the three image processing steps summarized in this paper yield an accurate result for representation of the wildfire perimeter.

Keywords: Wildfire Perimeter · Thermal Infrared Image · Delaunay Triangulation · Iterative Algorithm

* Work is supported by WIFIRE Commons and funded by NSF 2040676 and NSF 2134904 under the Convergence Accelerator program.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

1 Introduction

As unprecedented wildfire activity has occurred in recent years, the reliable prediction of the spread of an active wildfire has proven to be challenging task. Ensemble Kalman filtering [9] is often used as the data assimilation technique for wildfire spread prediction [15]. Many applications have combined a data-driven fire model such as FARSITE [11] with ensemble Kalman filtering to improve the prediction accuracy [10, 20, 21]. Such data assimilation techniques do rely on the availability of (past) fire perimeter measurements to predict (future) wildfire perimeters for characterizing wildfire progression.

For the moment, manual delineation is still frequently used to obtain the wildfire perimeter. Due to the significant advances in sensor technology for wildfire monitoring, new real-time data sources such as Thermal InfraRed (TIR) imaging can be used for a data-driven approach to predict wildfire progression. For example, wildfires can be monitored via MODIS data [6], satellite heat images [19] or thermal infrared imaging (TIR) on aerial flight systems [23] that characterize ground temperature. In case of TIR or heat images, image pixels represent temperature information or strength of the infrared band. In particular for an RGB image, the R value is chosen as the data for the image pixels.

Many contributions can be found that provided automated extraction of wildfire perimeters by applying classic edge detection algorithm, such as global intensity thresholding algorithm, Sobel gradient operator, and Canny edge detector [17, 24]. Among them, Canny method has the ability to outperform the others [24]. Such edge detection methods might be good tools to automatically delineate the wildfire perimeter, and the application of other edge detection methods, such as graph-cut method and level set method, are beneficial for the detection of wildfire perimeter. However, due to limited resolution of the image, discontinuity of the two-dimensional image data, and possibly partial activity of a wildfire along its boundary, identification of the most recent fire perimeter remains a challenge. Furthermore, different image pixels may be independent and subjected to noise or temporary fire inactivity. As a result, the burn area in the wildfire image can be disconnected or even sparse.

Although combining Machine Learning (ML) methodologies with real-time image data has been recognized to advance in wildfire science and management [13], there is an important restriction that a huge amount of training sets are required by ML techniques to learn the characteristics of the heat map of the burn area. Although computer vision is applied more and more on wildfire detection and measurement [22], few contributions can been found that provide a closed polygon for the parametrization of the wildfire perimeter on the basis of TIR or heat images. In [24], unsupervised edge detectors were applied to obtain the wildfire perimeter automatically, but performance in case of sparse TIR data had not been demonstrated.

The main contribution of the paper is to provided a novel TIR image processing technique to characterize a closed polygon for the wildfire perimeter. To solve the problem of discontinuity of a heat image, the basic concept of Delaunay triangulation is used to obtain a convex polygon of a burn area. Similar

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

ideas are explored in [6] where the so-called α-shape algorithm is used to determine the wildfire perimeter using information on hot spots. Unfortunately, the α-algorithm can only adjust the detected wildfire perimeter globally by changing the value of α, and is barely able to distinguish spot fires from the main burn area in an TIR image. To solve this problem, the novelty of the TIR image processing lies in the iterative trimming of triangular objects created by the Delaunay triangulation to obtain a closer match of the wildfire perimeter.

Both rough trimming and fine trimming are included in the iterative trimming method. For the rough trimming, after obtaining the convex polygon covering all pixels of the burn area by applying the Delaunay triangulation, two threshold values are created for this step. One is related to the longest side of the triangle created by Delaunay triangulation, and the other one is related to the relative burn area surrounding a vertex in a chosen domain. Based on these two threshold values, the iterative trimming method will first delete the redundant abnormally large triangle created by the vertex of the polygon, and then delete the isolated pixels of burn area caused by spot fire. As a result, a new convex polygon can be obtained relying on the remaining pixels of the burn area, and this process will be repeated iteratively until all the pixels of spot fires are removed. When the rough trimming is finished, another two threshold values related to the longest side of triangles connected to the vertex of the polygon and the relative burn area surrounding a vertex are created for the fine trimming. The wildfire perimeter can be finally obtained by tuning the threshold values. The performance of the iterative trimming method is illustrated by comparing the wildfire perimeter created by the iterative trimming method to those created by some classical edge detection methods, such as Canny edge detection, graph-cut method, and level set method.

2 Thermal Infrared Image of a Wildfire

One typical discontinuous RGB TIR image of wildfire is presented in Figure 1, and the corresponding R-value TIR image is presented in Figure 2. It can be observed from Figure 1 and Figure 2 that the bright area alternates with the dark area. Although the different bright areas are close, they are not connected. Delaunay triangulation is applied in this paper to link up those bright areas.

To prepare for the Delaunay triangulation, the active pixels and inactive pixels are defined as

$$y_{i,j} = f([i,j], b) = \begin{cases} 0, & \text{if } b < b_t, \\ 1, & \text{if } b \geq b_t, \end{cases} \tag{1}$$

where i, j describe the location of the pixel, b is the R-value of the TIR image, and b_t is the threshold value to distinguish between active pixel and inactive pixel. $y_{i,j} = 1$ means the pixel at i, j is identified as an active pixel. It is part of burn area and can be used to detect the wildfire perimeter; $y_{i,j} = 0$ means the pixel at i, j is considered to be an inactive pixel. It represents either inactive

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

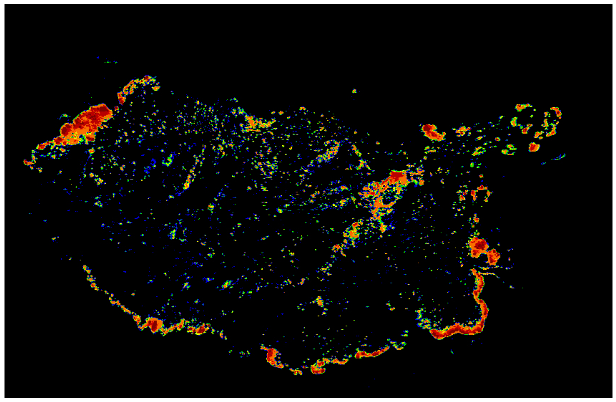


Fig. 1. Example of an RGB TIR image, courtesy of DigitalGlobe WorldView-3 satellite data of the Happy Camp Complex fire. The size of the image is 850×550 pixels.

wildfire or unburned area and should not be used for the detection of wildfire perimeter. In this paper, b_t is chosen to be 50. The reason is that the pixels with the R-value smaller than 50 belong to nearly completely dark area as shown in Figure 2.

3 Delaunay Triangulation and Iterative Trimming

3.1 Delaunay Triangulation

In 1934, Boris Delaunay introduces the Delaunay triangulation [7] that is well known for maximizing the minimum of all the angles of the resulting triangles. The Delaunay triangulation has been used in many applications due to the property of providing connectivity information for a given set of points [12]. This property is also adopted in this paper to solve the problem of missing connection between the burn area.

By applying the Delaunay triangulation on the active pixels determined from Figure 2 in Section 2, the connecting triangles (red) can be established as shown in Figure 3(a). The union set of all the resulting triangles created by Delaunay triangulation is a convex hull depicted in Figure 3(b). It can be observed that the convex hull contains many redundant (sliver) triangles that hide the shape of the wildfire perimeter. These redundant large triangles are mostly caused by the vertices of the polygon characterizing the wildfire perimeter. Delaunay Triangulation connects these vertices as it did for the internal disconnected burn area. In addition, some active pixels are caused by spot fires and should also be removed to discover the main perimeter of the main wildfire. Therefore, it is important to distinguish the triangles outside the main wildfire perimeter from

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

Fig. 2. R-value of TIR image depicted in Figure 1.

those inside the wildfire perimeter. To this end, an iterative trimming method is set up to trim the convex hull and reveal the wildfire perimeter.

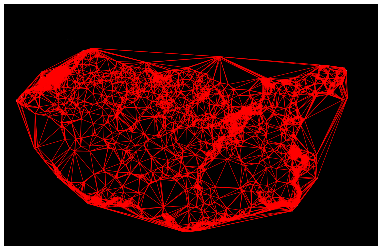
3.2 Iterative Trimming Method

To cut out the redundant triangles established by Delaunay triangulation and the active pixels caused by spot fires, two iterative trimming steps are used: rough trimming and fine trimming. In the process of rough trimming, the abnormally large triangles and the pixels of spot fire will be removed iteratively. After that, the remaining polygon is further trimmed in the fine trimming process. Details on both trimming processes are as follows.

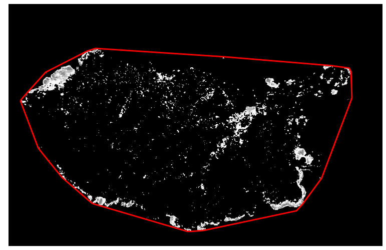
Rough Trimming As mentioned, the goal of rough trimming is to remove the abnormally large triangles created by Delaunay triangulation and some small groups of active pixels caused by spot fire to obtain a coarse shape of the main fire. The abnormally large triangles are selected by using the histogram of the longest side of all triangles. If the longest side of a triangle appears just once in a bin of the histogram and larger than the upper boundary of the last bin with count larger than two, then the triangle is categorized into the abnormally large triangle. The detailed procedure to recognize the abnormally large triangle is summarized in Algorithm 1.

Figure 4 shows the histogram calculated on the basis of the triangles depicted earlier in Figure 3(a). From the histogram in Figure 4, it can be observed that most of the triangles have a longest side smaller than 50, and abnormally large triangles can be recognized by choosing the last few triangles with one count in the histogram.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22



(a) Delaunay triangulation on active pixels of Figure 2.



(b) Polygon (red) by the union set of the triangles

Fig. 3. Delaunay triangulation and convex hull.

Algorithm 1 Identifying abnormally large triangle

Input: Longest side of each triangle, $\mathbf{s}$

Output: Abnormally large triangle

1: Construct a histogram about the longest side of each triangles, h=histogram($\mathbf{s}$).
2: Find the index of the last triangle with count larger than one, k=find(h.count > 1, 'last').
3: Abnormally large triangles are identified as the triangles with longest side larger than $l_r = k \times h$.binwidth.

Due to the fact that spot fire is outside the main body of the wildfire, it is only necessary to check whether the vertices of the currently established polygon belong to the main fire or a spot fire. Spot fires are defined as a tiny connected burn area that is isolated from the main fire. Therefore, the active pixels of spot fires can be distinguished by the relative surface area within a chosen domain. The chosen domain can be a square patch of the TIR image centered by the vertex of the current polygon, and the relative surface area can be calculated by the ratio of the number of the connected active pixels containing the vertex of the current polygon and the number of pixels inside the whole square patch. The approach to select the active pixels of spot fires is summarized in Algorithm 2.

Algorithm 2 Identifying active pixels of spot fire

Input: Vertices of currently established polygon, n; size of square image patch, $m \times m$; threshold value, d_u.

Output: Active pixels of spot fires.

1: For each vertex, calculate the summation, $p_q,\ q = 1, 2, \ldots, n$, of the number of all connected active pixels starting from the vertex inside the $m \times m$ square patch.
2: Compute the ratio $\frac{p_q}{m \times m}$, for $q = 1, 2, \ldots, n$
3: If $\frac{p_q}{m \times m} \leq d_u$, all the connected active pixels with respect to the q_{th} vertex of the polygon are regarded as part of the active pixels of spot fires.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

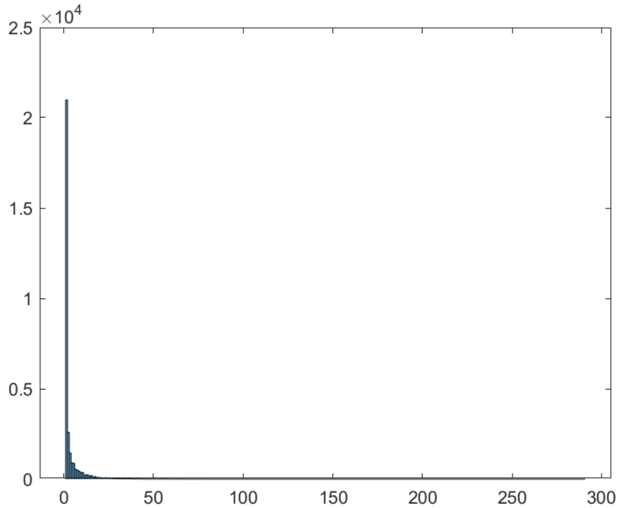


Fig. 4. Histogram of the longest side of the triangles in Figure 3(a). The unit of the side length is (one) pixel length.

As a consequence of removing the active pixels of spot fires, the set of points for Delaunay triangulation are affected. Hence, an iterative process of Delaunay triangulation, removing the abnormally large triangles, and removing the active pixels of spot fires is operated until all the active pixels of spot fires are eliminated. During the iteration, the number n of the vertices of the polygon will also change accordingly. After the rough trimming, a coarse shape of the main wildfire can be acquired, and the fine trimming will further remove the redundant triangles produced by Delaunay triangulation.

Fine Trimming Instead of just removing the abnormally large triangles coarsely as done in rough trimming, fine trimming is more rigorous. With the goal of obtaining the perimeter of the main wildfire, the fine trimming is focused on the triangles connected with the vertices of the current polygon to avoid making the polygon disconnected. The redundant triangles can also be determined by the histogram of the longest side, or specific requirement on the wildfire perimeter.

Furthermore, considering the fact that the TIR image is sparse, the density of the active pixels around a vertex should also be taken into account. Heavier trimming can be done on the triangles connected to a vertex having a denser neighbouring active pixels, and those triangles connected to a vertex that has a relatively sparse neighbouring active pixels should be discarded more carefully. As a result, two threshold values are established for the fine trimming. The first one is the threshold value l_f, set for the longest side of the triangles that should be removed, and the second one is the threshold value d_l, set for the density of

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

the neighbouring active pixels around a vertex. The value of d_l is chosen so that the triangles connected to a vertex with a sparse neighbouring active pixels will be protected from being removed even when the longest sides of the triangles are longer than l_f.

The density of the active pixels around a vertex can be measured similarly by the relative surface area $\frac{p_q}{m \times m}$ used in the Algorithm 2, where p_q is the number of all the connected pixels starting from the q_{th} vertex of the current polygon, and $m \times m$ is the size of the chosen square patch. All the triangles with a longest side larger than l_f and connected with a vertex that meet the requirement $\frac{p_q}{m \times m} > d_l$ will be removed in the fine trimming. During the process of trimming, new vertex will appear as the triangles are eliminated. Therefore, the trimming process should be operated iteratively until the number of the vertices of the polygon stays the same.

One of the main differences between the rough trimming and the fine trimming is that no active pixels are removed in the fine trimming. For this reason, the value of d_l can be less than the value of d_u. In other words, new vertex with $\frac{p_q}{m \times m} \leq d_u$ may appear during the fine trimming, and d_l can be chosen as $d_l < \frac{p_q}{m \times m} \leq d_u$ to trim the triangles connected to this newly created vertex. The complete procedure of the iterative trimming method including the step of Delaunay triangulation is summarized in Algorithm 3.

Algorithm 3 Iterative trimming method

Input: TIR image of wildfire.

Output: Wildfire perimeter.

1: Determine the set of active pixels.
2: Find the locations of the active pixels, and apply the Delaunay triangulation on the active pixels.
3: Remove the abnormally large triangles and the active pixels caused by spot fires summarized in Algorithm 1 and Algorithm 2 respectively.
4: If the number of active pixels changes, go back to step 2.
5: Remove the triangle connected to the the q_{th} vertex with $\frac{p_q}{m \times m} > d_l$, if the longest side of this triangle is larger than l_f.
6: If the number of vertices of the polygon changes in step 5, repeat step 5.

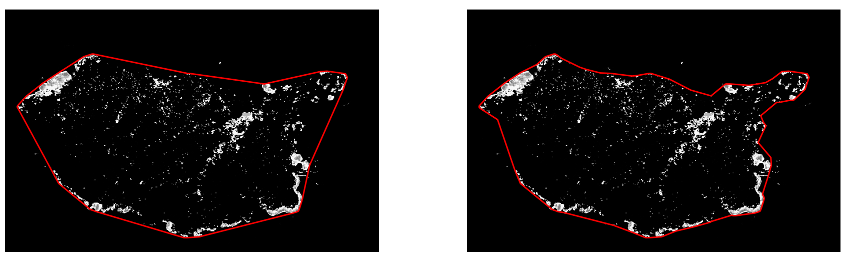
The performance of the iterative trimming method based on the TIR data given earlier in Figure 1 is illustrated in the next section. In addition, the established polygon of the main wildfire perimeter obtained by the iterative trimming method on the basis of the Delaunay triangulation is compared to those obtained by the Canny edge detector, the graph-cut method, and the level set method.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

4 Results and Discussion

4.1 Iterative Trimming Method

Considering the distribution of active pixels in Figure 2, the computed longest side of the triangles in Figure 3(a) and the resulting histogram depicted in Figure 4, the values of l_f, m, d_u, d_l are chosen as $50, 11, 0.16, 0.08$ respectively. The polygons obtained after the rough trimming and the fine trimming are shown in Figure 5(a) and Figure 5(b). It is worthwhile to note that, although the high-value infrared pixels are sparse and disconnected, the iterative trimming method can obtain a closed polygon of the wildfire automatically. It can also be noticed that some isolated active pixels outside the red polygon are regarded as the spot fires, and are not used to establish the polygon of the main wildfire. The computation time is around one second. As reference for the computation time, all calculations were performed on an Intel Core i7-7500U CPU with 16 GB RAM.



(a) Polygon (red) obtained by rough trimming.

(b) Polygon (red) obtained by fine trimming.

Fig. 5. Results of iterative trimming method.

4.2 Canny Edge Detector

The Canny edge detector is developed by John F. Canny in 1986 [5]. Although Canny edge detection is a traditional edge detection method, it has been widely applied and improved in more recent researches [1,8,18]. It was also utilized in the study of wildfire monitoring [24] and was shown to be one of the most effective unsupervised detection algorithms. Three performance criteria: good detection, good localization, and unique response to a single edge, form the basis of Canny edge detector.

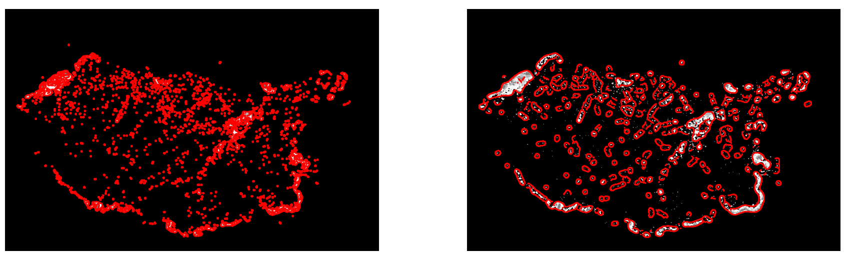
A Canny edge detection has five steps including smoothing the image by Gaussian filter, calculating the gradient of the image, deleting spurious response to true edges, using double threshold to find out prospective edge, and tracking edge by preserving strong edges and weak edges that are connected to strong

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

edges. The image processing toolbox of MATLAB provides the standard edge detection algorithm, and Canny edge detector can be applied by calling the function `edge(I,'canny',threshold,sigma)`, where `I` is the image, `threshold` is used to ignore the unnecessary edges, and `sigma` decides the standard deviation of the Gaussian filter.

It is clear from the procedure of Canny edge detector that the image gradient is the basis for this edge detection method. For the TIR image given earlier in Figure 2, the Canny edge detector is more likely to work on detecting the boundaries of all isolated clusters of pixels instead of the main wildfire. To blur the image to a greater extent so that the effect of the isolation is reduced, a larger standard deviation of the Gaussian filter can be applied.

The results of Canny edge detection with default value for the `threshold` and two different standard deviations of the Gaussian filter are shown in Figure 6. Although increasing the standard deviation of Gaussian filter leads to a slightly better result, the Canny edge detector is still not able to detect the main boundary of the main wildfire.



(a) `sigma` = 1. (b) `sigma` = 5.

Fig. 6. Boundaries (red) of the wildfire generated by Canny edge detector.

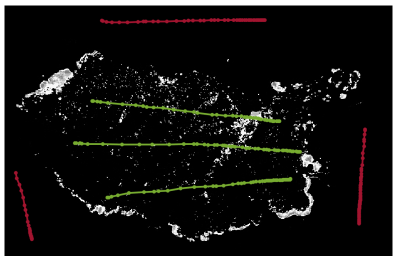
4.3 Graph-Cut Method

The graph-cut method is a widely used method in the field of computer vision and details can, for example, be found in [2,4,25]. Here a short summary of the graph-cut method is given. An image is first transformed to a graph consisting of nodes and edges, where edges are used to connect every two neighbour nodes, and nodes are composed of two terminal nodes, source node and sink node, and all pixels. Each edge is assigned with a weight or cost, and the goal of the graph-cut method is to find a minimum cut of the graph by using the max-flow min-cut theorem [3]. With the minimum cut, the image is divided into a foreground and a background.

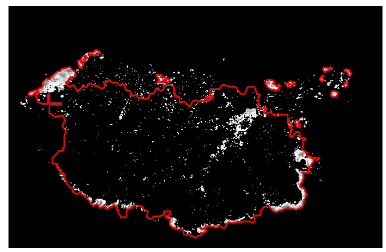
The graph-cut method has a good ability to produce an optimal solution to the image segmentation of a binary problem, which is similar to distinguishing

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

between the burn area and unburned area in a TIR image of a wildfire. Therefore, the graph-cut method is applied on Figure 2 to obtain the boundary of the wildfire via the image segmenter application in MATLAB. The seeds required by graph-cut method are highlighted as Figure 7(a), and the result of the graph-cut method is shown in Figure 7(b).



(a) Seeds for foreground (green) and background (red).



(b) Boundaries (red) generated by graph-cut methods..

Fig. 7. Results of graph-cut method.

It can be observed that graph-cut method only works well on identifying part of the wildfire boundary near the provided seeds. Although a good wildfire perimeter can be created by graph-cut method when more detailed and complicated seeds are provided, too much human interaction is required and the work involved is almost the same as a manual delineation, limiting the application in automatic wildfire perimeter detection.

4.4 Level Set Method

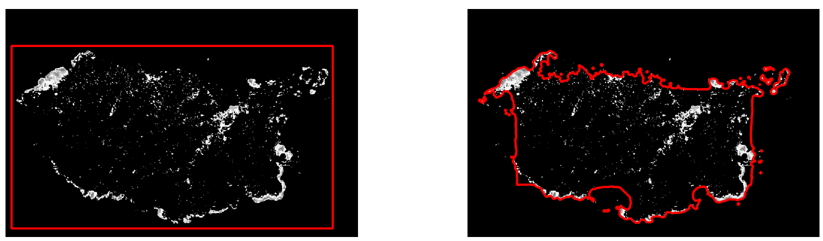
The level set method is an impressive tool for image segmentation by exploiting the information of regions and boundaries of the object [16]. It applies level sets for numerical analysis of surfaces, and the application of level sets makes it beneficial to track the change of the topology, such as the development of a hole. In addition, the level set method provides an implicit description of the object without the need of parameterizing the object. Due to the fact that a level set method has a good ability to separate two regions, it might be a solution to the binary problem of detecting the burn area from the unburned area.

Unfortunately, for a disconnected TIR image of a wildfire as depicted in Figure 2, it is infeasible to decide whether a hole exists, or what the size of the hole is. Moreover, continuous image gradient and intensity of the pixels are important components of level set method to determine the speed of the evolving and the shape described by each level set. Another potential problem of the level set method is that the final result is dependent on the choice of

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

the initial contour, but the shape of the wildfire can be arbitrary. Hence, an algorithm of automatically generating an initial zero-level contour is required, or a fire expert needs to provide an initial contour for each image of a wildfire to apply the level set method.

To test the performance of the level set method on Figure 2, the level set method introduced by [14] is adopted. By choosing the parameters $\alpha = 2$, $\lambda = 5$, $\mu = 0.01$, the result can be achieved after 200 iterations as Figure 8(b), with the initial contour established as Figure 8(a). It can be observed that the boundary captured by level set method passes through the disconnected part of the burn area of the TIR image. The reason for that is level set method also relies on the image gradient or pixel intensity to calculate the level sets, and there is a huge change in the image gradient between the disconnected areas.



(a) Initial zero level contour. (b) Final zero level contour

Fig. 8. Results of level set method.

5 Conclusions

This paper introduces an iterative trimming method (ITM) based on Delaunay triangulation with a goal to establish a closed polygon of the main wildfire perimeter automatically for a TIR image of wildfire. Although the burn area caused by spot fire is deleted in the ITM, they can be captured respectively by treating each burn area of a spot fire as a main wildfire perimeter. The performance of the iterative trimming method is validated by providing a study that compares the result of the ITM with those of the various edge detection methods based on a satellite generated TIR image. The comparison study shows that the various edge detection methods fail to provide a single closed polygon that parametrizes the main wildfire perimeter. Often, disconnected burn areas will be detected separately. The proposed ITM shows good performance with a single closed polygon for the wildfire perimeter. For further studies, more

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

information of the wildfire can be used in the iterative trimming method. For example, the wind direction and wind speed can be used to predict the location of the spot fire, and a priori knowledge of the spot fire can be included in the iterative trimming method to better capture and remove the active pixels caused by spot fires.

References

1. Bao, P., Zhang, L., Wu, X.: Canny edge detection enhancement by scale multiplication. IEEE transactions on pattern analysis and machine intelligence **27**(9), 1485–1490 (2005)
2. Boykov, Y., Funka-Lea, G.: Graph cuts and efficient nd image segmentation. International journal of computer vision **70**(2), 109–131 (2006)
3. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE transactions on pattern analysis and machine intelligence **26**(9), 1124–1137 (2004)
4. Boykov, Y.Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In: Proceedings eighth IEEE international conference on computer vision. ICCV 2001. vol. 1, pp. 105–112. IEEE (2001)
5. Canny, J.: A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence (6), 679–698 (1986)
6. Chiaraviglio, N., Artés, T., Bocca, R., López, J., Gentile, A., Ayanz, J.S.M., Cortés, A., Margalef, T.: Automatic fire perimeter determination using MODIS hotspots information. In: 2016 IEEE 12th International Conference on e-Science (e-Science). pp. 414–423 (2016)
7. Delaunay, B., et al.: Sur la sphere vide. Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk **7**(793-800), 1–2 (1934)
8. Ding, L., Goshtasby, A.: On the canny edge detector. Pattern recognition **34**(3), 721–725 (2001)
9. Evensen, G.: The ensemble Kalman filter: Theoretical formulation and practical implementation. Ocean dynamics **53**(4), 343–367 (2003)
10. Fang, H., Srivas, T., de Callafon, R.A., Haile, M.A.: Ensemble-based simultaneous input and state estimation for nonlinear dynamic systems with application to wildfire data assimilation. Control Engineering Practice **63**, 104–115 (2017)
11. Finney, M.A.: FARSITE, Fire Area Simulator–model development and evaluation. No. 4, US Department of Agriculture, Forest Service, Rocky Mountain Research Station (1998)
12. Ito, Y.: Delaunay triangulation. Encyclopedia of Applied and Computational Mathematics; Springer: Berlin/Heidelberg, Germany pp. 332–334 (2015)
13. Jain, P., Coogan, S.C., Subramanian, S.G., Crowley, M., Taylor, S., Flannigan, M.D.: A review of machine learning applications in wildfire science and management. arXiv preprint arXiv:2003.00646 (2020)
14. Li, C., Xu, C., Gui, C., Fox, M.D.: Distance regularized level set evolution and its application to image segmentation. IEEE transactions on image processing **19**(12), 3243–3254 (2010)
15. Mandel, J., Bennethum, L.S., Chen, M., Coen, J.L., Douglas, C.C., Franca, L.P., Johns, C.J., Kim, M., Knyazev, A.V., Kremens, R., et al.: Towards a dynamic data driven application system for wildfire simulation. In: International Conference on Computational Science. pp. 632–639. Springer (2005)

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22

16. Mumford, D., Shah, J.: Boundary detection by minimizing functionals. In: IEEE Conference on Computer Vision and Pattern Recognition. vol. 17, pp. 137–154. San Francisco (1985)
17. Ononye, A.E., Vodacek, A., Saber, E.: Automated extraction of fire line parameters from multispectral infrared images. Remote Sensing of Environment **108**(2), 179–188 (2007)
18. Rong, W., Li, Z., Zhang, W., Sun, L.: An improved canny edge detection algorithm. In: 2014 IEEE international conference on mechatronics and automation. pp. 577–582. IEEE (2014)
19. Schroeder, W., Oliva, P., Giglio, L., Quayle, B., Lorenz, E., Morelli, F.: Active fire detection using landsat-8/oli data. Remote Sensing of Environment **185**, 210 – 220 (2016), landsat 8 Science Results
20. Srivas, T., Artés, T., de Callafon, R.A., Altintas, I.: Wildfire spread prediction and assimilation for FARSITE using ensemble kalman filtering. Procedia Computer Science **80**, 897–908 (2016)
21. Subramanian, A., Tan, L., de Callafon, R.A., Crawl, D., Altintas, I.: Recursive updates of wildfire perimeters using barrier points and ensemble Kalman filtering. In: International Conference on Computational Science. pp. 225–236. Springer (2020)
22. Toulouse, T., Rossi, L., Campana, A., Celik, T., Akhloufi, M.A.: Computer vision for wildfire research: An evolving image dataset for processing and analysis. Fire Safety Journal **92**, 188–194 (2017)
23. Valero, M.M., Rios, O., Mata, C., Pastor, E., Planas, E.: An integrated approach for tactical monitoring and data-driven spread forecasting of wildfires. Fire Safety Journal **91**, 835 – 844 (2017), fire Safety Science: Proceedings of the 12th International Symposium
24. Valero, M., Rios, O., Pastor, E., Planas, E.: Automated location of active fire perimeters in aerial infrared imaging using unsupervised edge detectors. International journal of wildland fire **27**(4), 241–256 (2018)
25. Yi, F., Moon, I.: Image segmentation: A survey of graph-cut methods. In: 2012 international conference on systems and informatics (ICSAI2012). pp. 1936–1941. IEEE (2012)

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_22