Index Generation for Satellite Image Retrieval

Pattrawuth Phuthong∗†, Charay Lerdsudwuchai† and Panu Srestasathierm*
∗Geo-Informatics and Space Technology Development Agency (Public Organization) Bangkok, Thailand
†Kasetsart University, Bangkok, Thailand
Email: phuthong@gistda.or.th; jan@ku.ac.th; panu@gistda.or.th

Abstract—Index generation for satellite image is an important task for retrieving satellite image in data warehouse. The goal of index generation is to find a polygon that represents the geographic image area. Using too many vertices for satellite images index can decrease retrieval speed. Therefore, the aim of this research is to develop an algorithm for generating index for satellite images using small amount of vertices and having high accuracy. The key idea of the algorithm is to decrease the number of vertices representing image shapes by mean of polygon simplification. The obtained polygon is expanded in order to make its cover the image area as much as possible. The experimental results showed that the proposed algorithm generated image index with high accuracy using small number of vertices comparing with other softwares i.e., ArcGIS 10.1, Global Mapper 15 and DIQB. In addition, the result is useful and quick for retrieving data from the spatial database.

I. INTRODUCTION

Satellite image is an important data for monitoring large-scale geographical area. To retrieve images of a specific area, satellite image index is used. The index of a satellite image can be simply defined as the polygon representing the geographical boundary of image area. Therefore, an image is retrieved if its index intersect with the given geographical area. An example of the satellite image index is illustrated in Fig. 1 where the red line is satellite image index.

Generally, satellite images must be geometrically corrected in order to remove geometric distortion caused by sensor and earth rotation, see Fig. 2. Moreover, the satellite images are generated by mosaicking and clipping in order to show only the desired geographical are, see Fig. 3. As a consequence, the shapes or boundaries of the satellite images are transformed to be arbitrary shapes and can cause a problem in index generation because a large amount of polygon’s vertices is required to define the image area.

Since the shapes of the image areas can be complicated, the amount of points used to represent the shapes can be very large. As a consequence, the spatial query can be very slow and a big storage is required [1]. To attack such problems, the objective of this paper is to propose a satellite image indexing generation method using the boundary pixels of the image area. Particularly, the concept of the proposed method is to sub-sample the boundary pixels such that polygonal area of the subsampled pixels is maximally covering the image area. The indexing process can work automatically by using morphological processing for finding edge of image area boundary. The obtained edge is then simplified by polygon approximation algorithm. The obtained polygon is expanded in order to make it covers the image area.

Fig. 1. An example of satellite image index which is shown in red lines.

Fig. 2. Satellite image before and after geometric correction

Fig. 3. Image after mosaicking and clipping where the pixels outside the image area has no-data values.

The rest of this paper is organized as following. In Section II, we reviewed some of existing works of generating index from satellite images. The methods introduce in this paper will be presented in Section III. We tested our method with several datasets. The results are reported in Section IV and conclusions are drawn in the last section.
II. RELATED WORKS

In [2], a system called Hadoop-GIS was proposed for supporting the queries on large volumes of spatial data. The indexing process begins with dividing the rectangular circumscribing the image into \( N \times N \) sub-regions and then providing satellite image an index as demonstrated in Fig. 4. The red dash line is the rectangle circumscribing the image area and the green dash line is satellite index generated by the system. That is, the image index is the geo-location of the green dash squares. It is evident that some of the index (green square) are false positives because they partially contain image area. Therefore, this is the weakness of this approach.

Fig. 4. Satellite image indexes received from Hadoop-GIS

Phuthong et al. proposed a system called DIQB in [3]. Satellite image index can be automatically generated by first thresholding the image in order to obtain a binary image. The majority filter is then applied to fix no-data pixel on the image. The index is then obtained by using the raw edge of the binary image as illustrated in Fig. 5. The red line is the satellite image index from DIQB and the yellow line is satellite image index drawn by operator. Although the obtained index is highly accurate, too many polygon vertices are used. As a result, image retrieval can be very slow.

Fig. 5. An example of index generated by DIQB (red line) compared with manually drawn index (red line)

III. METHODOLOGY

The proposed algorithm begins with image thresholding for binary image generation. The morphological image processing is then used to find boundary of the image area. In order to reduce the number of polygon vertices representing the image boundary, the polygon approximation algorithm is utilized. The polygon after vertex reduction is then expanded to cover the whole image area as much as possible using Quadratic programming technique. Lastly, the expanded polygon is then geo-referenced and used as satellite image index. The flow of the proposed algorithm is illustrated in Fig. 6.

Fig. 6. The workflow of the proposed algorithm

A. Image Thresholding

The first step is to apply image thresholding to generate binary image which will be used to extract the edge of the image in the next step. Image pixel can have no-data value for representing the absence of data. Assuming that the RGB values of no data pixels are zero, the image thresholding can be performed using the following formula:

\[
I_{th}(x,y) = \begin{cases} 
0, & \sum_{i=1}^{n} I(x,y) = 0 \\ 
1, & \sum_{i=1}^{n} I(x,y) \neq 0 
\end{cases},
\]

(1)

Where \( n \) is the number of bands of the image \( I \) and \( I_{th} \) the thresholded image. An example of image after applying image thresholding is demonstrated in Fig. 7.

B. Morphological image processing

In next step, we use the binary image after zero padding to perform morphological processing to find edge. It can be computed as following:

\[
E = I_{th} - (I_{th} \Theta K),
\]

(2)

where \( I_{th} \) is the binary image, \( K \) 3 by 3 circular kernel, \( \Theta \) the erosion operator and \( E \) edge image. A result after performing of morphological processing is illustrated in Fig. 8(a). It can be observed that the resulting image contain noise because of pixel with no-data value. The noise problem is fixed in next step.
C. Edge tracing

To extract image boundary from noisy edge image, we apply connected component i.e., chain code using algorithms presented in [4]. In the case of obtaining more than one connected components i.e., noisy case, we choose the longest one because we hypothesize that the image noises have small connected components and the boundary edge has the longest one. An example of noise removal is demonstrated in Fig. 8(b).

D. Vertex reduction and expansion

The obtained edge pixels are then refined using a heuristic method called the Douglas-Peucker algorithm [5] to approximate the image area boundary by a polygon. The concept of this algorithm is to find a piecewise linear curve that is similar to the original curve or polygon. The Douglas-Peucker algorithm uses error measures [6] for choosing which vertices must be removed. That is, a vertex is removed if its error is greater than a threshold value which is set to be 1.5 pixel.

Although the coordinates of vertices are in (quantized) pixel coordinate system which has no physical meaning or dimensionless, each pixel of the satellite image has spatial resolution. That is, each corner of the pixel has its own coordinate on the ground or earth surface. Therefore, a corner of each pixel must be carefully selected to give coordinates to polygon vertices. In Fig. 9, the concept of pixel corner selection is illustrated. Red pixels are image area (texture), gray pixels the no-data value pixel, yellow dot the selected pixel corner and the green polygon the generated satellite image index. If only upper left corner is used, satellite image index cannot cover the geographical image area. We therefore hypothesize that the satellite image index should be at the pixel corners maximizing polygon area. In other word, the polygon is expanded to cover the image area as much as possible.

Given that the polygon has \( n \) vertices i.e., \( p_i = (x_i, y_i) \) where \( i = 1, 2, \ldots, n \). Therefore, the area of the polygon can be computed using Shoelace formula:

\[
Area(x_1, y_1, x_2, y_2, \ldots, x_n, y_n) = \frac{1}{2}((x_1y_2 - x_2y_1) + (x_2y_3 - x_3y_2) + \ldots + (x_ny_1 - x_1y_n)).
\]  

(3)

Let the polygon vertice \( p_i \) is at upper left pixel corner. Each four corner of a pixel can be represented by shift \( (x_i + s_i, y_i + t_i) \) where \( s_i, t_i = 0 \) or \( 1 \) e.g., \( (s_i, t_i) = (1, 1) \) means lower right corner. After some algebraic manipulation, the polygon area with shifting can be formulated as:

\[
Area(x_1 + s_1, y_1 + t_1, \ldots, x_n + s_n, y_n + t_n) = c^\top p + \frac{1}{2}p^\top G p + Area(x_1, y_1, x_2, y_2, \ldots, x_n, y_n),
\]  

(4)

where \( p = [s_1 \ t_1 \ldots s_n \ t_n]^\top \). Since the polygon area is in the form quadratic function of the shift \( p \), the shift maximizing the polygon area can then be obtained by quadratic programming. The refined polygon vertices are then transformed from Cartesian coordinate system to geo-coordinate system [7].
IV. RESULTS AND DISCUSSION

The performance of the proposed algorithm was tested against DIQB [3] and commercial softwares including ArcGIS 10.1, Global Mapper 15, QGIS 2.0. In this experiment, 1489 satellite images from Thaichote, WorldView1, WorldView2, GeoEye1, Ikonos2, QuickBird2, Spot5, Spot6, Landsat5, Landsat7 and Landsat8 were used. These images were grouped into three datasets:

1) Dataset 1: Image area boundary is convex polygon (456 images).
2) Dataset 2: Image area boundary is concave polygon (466 images).
3) Dataset 3: Image area boundary is the combination of straight lines and curves (567 images).

A. Performance evaluation

In information retrieval applications, Precision and Recall is used to measure the performance of the retrieval algorithm. Such two indices are computed using True Positive (TP), False Positive (FP) and False Negative (FN). In this context of satellite image index generation, TP is defined as the intersection area of generated index and true image area (ground truth), FP the area of generated index which not intersect with ground truth and FN the area of ground truth which does not intersect with generated index. Precision and Recall can be computed as following:

\[
\text{Precision} = \frac{TP}{TP + FP},
\]

\[
\text{Recall} = \frac{TP}{TP + FN}.
\]

Precision is then the fraction of retrieved instances that is relevant to the ground truth and Recall the fraction of ground truth that is retrieved. When Recall is 1, the generated index covers the whole area of the ground truth. The precision and recall can be combined to obtain a single performance measure called F-measure:

\[
F_\beta = (1 + \beta^2) \times \frac{\text{Precision} \times \text{Recall}}{(\beta^2 \times \text{Precision}) + \text{Recall}}.
\]

The value of $\beta$ is selected based on applications. $F_2$ gives weight on recall higher than precision, and $F_{0.5}$ gives weight precision higher than recall.

In this experiment, ground truths were generated by image thresholding and tracing inner boundary by 8-connectivity condition. The center of pixel is then used to find the geo-coordinates of polygon vertices. The accuracy of index is then measured by $F_2$ measure because the aim of is to find the index covering the image area as much as possible.

Another performance metric is the number of vertices because it affects the amount of storage and retrieving speed. If too many vertices are used, retrieving speed is then slow [8].

In order to visualize the performance testing result, the covariance ellipse [9] of the performance metrics is used. Since a good index should use small amount of vertices and also having high $F_2$ measure, a good index generation algorithm is expected to have the covariance ellipse close to the lower right of the graph.

B. Comparison using datasets 1 and 2

The shapes of the images in dataset 1 and 2 are not complex because their shape are convex and concave polygons. The expected number of vertices for image indices should be small. The experimental results are shown in Fig. 10 and 11. It is shown that proposed algorithm can generate satellite image index using small number of vertices with high accuracy of boundary as shown by covariance ellipse at lower right of graph. Comparing with DIQB, the proposed algorithm can obtain similar accuracy level using much lower number of vertices. For ArcGIS 10.1 and Global Mapper 15, their number of vertices and $F_2$ measure standard deviations are wider than the proposed algorithm. In other word, the proposed algorithm can obtain better $F_2$ measure with lower number of vertices.

C. Comparing with several algorithm by dataset 3

The shapes of the images in dataset 3 are more complex dataset 1 and 2 because they are the combinations of piecewise
straight lines and curve. The experimental result is illustrated in Fig. 12. It can be observed that although ArcGIS 10.1 has less vertices than the proposed algorithm, the accuracy of the proposed method is better because the shape of image boundary is complex. Namely, the default number of vertices used by ArcGIS 10.1 is not enough to represent complex shapes, see Fig. 13.

![Fig. 12. Covariance ellipse between accuracy of boundary and number of vertices of dataset 3](image)

D. Comparison using all images

The experimental result using all images is shown in Fig. 14. It can be observed that the proposed algorithm can generate satellite image index with high accuracy using small number of vertices and high accuracy. The concept of the proposed method is to use the polygon approximation algorithm to reduce the number of vertices. The obtain polygon is then expanded in order to make it covers the image area as much as possible. The proposed method is tested against existing method and many commercial softwares. Comparing with the softwares, the experimental results shown that the proposed algorithm can generate index having small number of vertices with high accuracy. The advantage of the proposed algorithm is that any ortho-rectified image can be used including satellite, aerial or UAV imagery.

![Fig. 14. Covariance ellipse between accuracy of boundary and number of vertices using all images](image)

V. CONCLUSION AND FUTURE DIRECTION

In this paper, a novel algorithm was proposed. The ultimate goal of this research is to develop an algorithm that can generate satellite image index having small number of vertices and high accuracy. The concept of the proposed method is to use the polygon approximation algorithm to reduce the number of vertices. The obtain polygon is then expanded in order to make it covers the image area as much as possible. The proposed method is tested against existing method and many commercial softwares. Comparing with the softwares, the experimental results shown that the proposed algorithm can generate index having small number of vertices with high accuracy. The advantage of the proposed algorithm is that any ortho-rectified image can be used including satellite, aerial or UAV imagery.

REFERENCES


