Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 13, Number 7 (2017), pp. 2975–2982

http://www.ripublication.com/gjpam.htm

The Uniqueness of Image Segmentation Generated by Different Minimum Spanning Tree

Efron Manik
Department of Mathematics, University of Sumatera Utara,
and Department of Mathematics Education,
Nommensen HKBP University, Medan-Indonesia.

Saib Suwilo and Tulus
Department of Mathematics,
University of Sumatera Utara, Medan-Indonesia.

Opim Salim Sitompul
Department of Computer Science,
University of Sumatera Utara, Medan-Indonesia.

Abstract

Image segmentation is an important topic in computer vision and image can be viewed as a connected graph. One method for segmenting the image is the use of minimum spanning tree for a graph. The minimum spanning tree of the connected graph can always be built. If the edge that is a weight greater than a threshold of minimum spanning tree is removed, it will form some of the connected components. One of the connected components will form a segment of an image. We know that the minimum spanning tree for a weight graph is not unique. The problem in this study is what the different minimum spanning tree does not produce the different segments of the image. We will discuss the components obtained, namely: if the edge that is a weight greater than a threshold of minimum spanning tree is removed. Let $G(V, E)$ is a weight graph. Let $\alpha \in \mathbf{R}$, and S, T is a minimum spanning tree of the graph G. Suppose that all the edges with weights greater than or equal to α are removed from S, and T. Then the connected components $S_1, S_2, \ldots, S_p$ of the tree S and $T_1, T_2, \ldots, T_q$ of the tree T will be formed. Then $p = q$ and if $V(S_i) \cap V(T_j) \neq \emptyset$ then $V(S_i) = V(T_j)$. So although S, and T minimum spanning tree of a graph G is different, but each set of points of a connected component of S has the same pair of a set of points of a connected component of T. If that graph is viewed as an image, it can be concluded that image segmentation generated by different minimum spanning tree is unique.

AMS subject classification: 05C (Graph theory).
Keywords: Segmentation, minimum spanning tree, connected component.

1. Introduction

Image segmentation is an important topic in computer vision. Region-growing approach, the boundary approach, graph approach is a variety of approaches to segmenting the image. Set of vector sequence that converges to a point will form one segment. Such a method is called a region-growing meanshift method (see [1]). The area for each segment is strongly influenced by the value of bandwidth. If the value of bandwidth to be smaller then the area of each segment to be great. We actually expect a collection of several small segments will form a large segment to a value smaller bandwidth. But meanshift method does not work as intended. This is a weakness meanshift method (see [2]). The border approach [3] employs edge function in MatLab to segment the image. The use of minimum spanning tree (MST) for a graph is another method for segmenting. If the edge whose weight is greater than the threshold value is removed from the minimum spanning tree then some connected components will be formed that comes from the tree. Each connected component is seen as one segment (see [4]).

Peter [4] explains that the image of mining is more than just an extension of data mining for domain image. The technique used to extract the direct knowledge of the image is a part of data mining. The first step in mining image is image segmentation. Minimum Spanning Tree-based Structural Similarity Clustering for Image Mining with Local Region Outliers (MSTSSCIMLRO) algorithm is used for segmenting the image. MSTSSCIMLRO algorithm able to detect outlier data. MSTSSCIMLRO algorithm uses Euclidean distance weighted to the side (edge), the which is a key element in building a graph of the image. Image segmentation based MST is a fast and efficient method to produce a set of segments of an image. This algorithm uses a new cluster validation criteria based on geometric properties partitioning of the data sets to find the right number of segments. The algorithm works in two stages. The first stage of the algorithm creates a number of optimal segments, where as the second phase of the algorithm further segment the optimal number of segments and detecting outliers local area.

Minimum Spanning Tree (MST), denoted EMST1, is obtained by using Kurskal algorithm. Furthermore, $\bar{w}$ and σ is calculated, where the $\bar{w}$ is the average of all weights contained in EMST1 and σ is the standard deviation. Each edge e_i with $w(e_i) > \bar{w} + \sigma$ removed from EMST1 and this has resulted in a set of disjoint sub-tree, for example: $S_T = \{T_1, T_2, \ldots\}$. Every T_i is seen as a segments.

MSTSSCIMLRO algorithm has several advantages. Segmentation is not influenced by the selection of the initial value. The benefits of this algorithm is to find common ground structure (segment) in the cluster. Data outliers have little or no influence in order to determine a final conclusion. This algorithm is able to determine the number of clusters based only on data that is processed. We do think that this is more natural way to segment the image. All of reviews these look nice from the theoretical point of view, there is still some room for improvement for the running time of the clustering algorithm (see [4]).

From the above description, we know that if the edge of MST, whose weight is greater than the threshold, is removed from the tree then some connected components will be formed. One connected component will form one segment in the image. But we know that the MST for a weight graph is not unique. The problem in this study is what the different minimum spanning tree does not produce the different segments of the image.

In this paper, we will discuss about the segmentation of images obtained from different MST. Section 2 discusses the graph and trees that will be needed for proving theorems in the next section. Section 3 contains evidence that different MST will still produce the same image segmentation. Finally, this paper ends with Section of conclusions and suggestions.

2. Graf and Trees

Before discussing spanning tree, we will first discuss the terms of a graph. Notions of graph and trees, which are required, will be written next. Definitions, theorems, and proofs in this section is taken from [5].

Definition 2.1. A **graph** G is an object consisting of

- A finite set and not empty V whose elements are called points of the graph G,
- Together with a set E which is a subset of unordered pairs of elements in the set V. Elements of the set E is called the edge of the graph G.

A graph G with a set of points V and the set of edges E is denoted by $G(V, E)$. Sometimes to clarify the context of the conversation, the set of points on a graph G is denoted by $V(G)$ and the set of edges in a graph G is denoted by $E(G)$. In the same way for the H subgraph of a graph G, the notation $V(H)$ is the set of points at subgraph H and $E(H)$ is the set of edges in the subgraph H. Graf $G(V, E)$ is called a **weight graph** if a graph $G(V, E)$ with a real-valued function $w : E \to \mathbf{R}$. For every edge $\{u, v\}$ in $E(G)$, the value of $w(\{u, v\})$ is called the weight of the edge $\{u, v\}$. Total weight of all edges of the graph G denoted

$$w(G) = \sum_{e \in E(G)} w(e).$$

Many problems in life can be modeled in a graph. The problem can be solved by solving problems like in graph problems. Image in computer vision can be viewed as a graph. Pixels is a point in the graph and the weight of edges connecting these points is the difference in intensity between adjacent points. We can make the nets as grid graph (each point is connected with the surrounding four points, namely: the point of the left, right, top, and bottom) or as supergrid graph (each point is connected with eight points surrounding) (see [6]).

We started the concept of connectedness in the graph by introducing some important terms. Suppose G is a graph. A **walk** with t connecting points u and v is a sequence consisting of t edges in the form

$$\{u = u_0, u_1\}, \{u_1, u_2\}, \{u_2, u_3\}, \ldots, \{u_{t-1}, u_t = v\}.$$

Note that in the above sequence for each $i = 1, 2, \ldots, t - 1$ edge to i have the same end point with the starting point of the edge to $i + 1$. A walk is said to be **open** when $u \neq v$ and is said to be **closed** when $u = v$. It is possible that in a walk might be a repetition of edge. A walk without repetition of an edge is called **path**. A closed path is called a **circuit**. Further, in a path it is possible there is also a point repeated use. A path without repetition point, except perhaps the end points, is called a **simple path**. A simple path can also be defined as a path without repetition point, except perhaps the endpoints. A **circle** is a simple closed path.

Two points u and v are said to be connected if there is a walk in the graph G connecting a point u to point v. Equivalence class of connectedness relation for the two points is referred to as a connected component of the graph G. A graph G is said to be connected if G has exactly one connected component. In other words, a graph G is said to be connected if for every pair of points u and v there is a walk that connects a point u to point v.

Next we will discuss a special graph, which is a tree. We will also observe the special properties of the tree that are useful in this paper.

Definition 2.2. A **tree** is a connected graph, which does not contain circles.

First Properties of the tree, that will be discussed, is the relationship between the number of vertices and edges of a tree. The following theorem states that the number of edges is smaller than the number of points.

Theorem 2.3. A graph T with n vertices and m edges is a tree if and only if T is connected and $n = m + 1$.

Furthermore, a simple path connecting the two points will be the basis of a statement saying that the addition of one edge to the tree will form a circle. It will be written in the next Corollary.

Theorem 2.4. Theorem 2.4. A graph T is a tree if and only if any two different points is connected by exactly one simple path.

Corollary 2.5. If two points are not neighbors in a tree are connected, it will be formed exactly one circle.

Proof. Let u and v are not adjacent points on the tree. According to Theorem 4 then there is exactly one simple path P from point u to the point v. So $\langle P, \{v, u\}\rangle$ a simple trajectory and closed. So exactly one circle is formed. ■

The uniqueness of the circle in the corollary above will be used in the proof of the theorems later.

Theorem 2.6. Every connected graph G has a spanning tree.

This theorem states the existence of spanning tree. This means that we can construct a spanning tree of a graph if the graph is connected.

Definition 2.7. An **incidence matrix** of a digraph D on n points $\{1, 2, \ldots, n\}$ and m edge $\{s_1, s_2, \ldots, s_m\}$ is a matrix $P(G) = (p_{ij})$ with order $n \times m$ where entry is defined as

$$p_{ij} = \begin{cases} 1, & \text{if } i \text{ insidence to } s_j, \\ -1, & \text{if } i \text{ insidence of } s_j, \\ 0, & \text{if } i \text{ do not insidence with } s_j. \end{cases}$$

Suppose G is a graph and digraph D is the orientation of G. Let P be the incidence matrix of a digraph D and J is a square matrix with all entries 1.

The following theorem says that the spanning tree graph is not unique. It is not unique to make us want to know the effect on the image segmentation or graph.

Theorem 2.8. The number of spanning tree of a graph G with n point is $\kappa(G) = det(J + PP^t)/n^2$.

The spanning tree with the minimum sum of all weights is important. Spanning tree with the minimum sum of all weights is called Minimum Spanning Tree (MST). Many algorithms were developed to determine the MST, for example: Kruskal's algorithm.

Algorithm: Kruskal

Input : A weight undirected graph $G(V, E, w)$!

Output : A minimum spanning tree T.

1. Sort the edges in E in $w(e_1) \leq w(e_2) \leq \cdots \leq w(e_m)$.
2. $T \leftarrow (V(G), \emptyset)$.
3. **for** $i := 1$ to m **do**
   if $T + e_i$ not load circles **then**
   $T \leftarrow T + e_i$

Is the graph produced by Kruskal algorithm a minimum spanning tree? This question will be answered on the following theorem. Proof of this theorem will be written, because the steps on the proof of this theorem is similar to what will be done in the proof of the theorems in Section 3.

Theorem 2.9. Kruskal's algorithm produces a minimum spanning tree of a weight connected graph.

Proof. Suppose G is a connected weight graph with n points, and let T is a subgraph generated by Kruskal algorithm. Due to the replacement of $T := T + e_i$ in step 3 of the algorithm if the edge e_i do not form a cycle with the edges before. Then T is a spanning tree of G. According to Theorem 3 the edges can be written

$$E(T) = \{e_1, e_2, \ldots, e_{n-1}\},$$

where $w(e1) \leq w(e2) \leq \cdots \leq w(e_{n-1})$. So that the weight of the tree T is

$$w(G) = \sum_{i=1}^{n-1} w(e_i).$$

Next we will show that T is a minimum spanning tree of G. Suppose instead that T is not a minimum spanning tree. So among all the minimum spanning tree of G, there is a minimum spanning tree, say H, which has the maximum number of edges that equal to T. Now edge spanning tree T and H are not identical, then T has at least one edge that is not in H. Suppose e_j, $j = 1, 2, \ldots, n-1$ is a first edge of the T is not present in H. Let $H_1 = H + e_j$, then according to Corollary 5, H_1 has exactly one circle, say C. Since T does not have the circle, there are edge e_0 in C not be in T. Since e_0 is an edge of the circle C, then $T_1 = H_1 - e_0$ is a spanning tree and

$$w(T_1) = w(H) + w(e_j) - w(e_0).$$

Because H is a minimum spanning tree, $w(H) \leq w(T_1)$. Consequently $w(e_0) \leq w(e_j)$. By Kruskal algorithm, e_j edge is a edge with a minimum weight so subgraph

$$\langle\{e_1, e_2, \ldots, e_{j-1}\}\rangle \cup \{ej\}$$

not load circles. But $\langle\{e_1, e_2, \ldots, e_{j-1}, e_0\}\rangle$ is subgraph of H therefore not loading circle, so $w(e_j) \leq w(e_0)$. So $w(T_1) = w(H)$, which means that the T_1 is a minimum spanning tree of G. But the T_1 has a number of edges equal to H more than the number of edges between T and H. The contradiction with the assumption that H is minimum spanning tree with the number of edges equal to T is maximum. So T is a minimum spanning tree. ■

According to Mayr [7], every minimum spanning tree can be generated using Kruskal algorithm. So that all minimum spanning tree can be seen as the minimum spanning tree generated by Kruskal algorithm.

3. Results and Discussion

We will start theorem that discusses a unique circle on a tree if the neighbor points are connected. This circle has the edge which have special properties.

Theorem 3.1. Let $G(V, E)$ a weight graph and S, T is a minimum spanning tree of G. If $e \in E(S) \setminus E(T)$, then $T + e$ contains exactly one circle C such that C contains edge $e\prime \in E(T) \setminus E(S)$ where $w(e) = w(e\prime)$, $E(C - e) \subseteq E(T)$, $E(C - e\prime) \subseteq E(S)$, and for each $e_i \in C$ applies $w(e_i) \leq w(e) = w(e\prime)$.

Proof. Let $e_1, e_2, \ldots, e_{n-1}$ edges of S, where $w(e_1) \leq w(e_2) \leq \cdots \leq w(e_{n-1})$. Suppose that the first member $e_j = e \in E(S) \setminus E(T)$ is a member of S but not a member of T. Since T is minimum spanning tree of G then $T_1 = T + e$ contains exactly one cycle C. Thus $E(C - e) \subseteq E(T)$. Since S also minimum spanning tree of G that does not load the circle then the circle C contains edge $e\prime \in E(T) \setminus E(S)$ with the greatest weight. Because according to Corollary 5, the circle C is unique then $E(C - e\prime) \subseteq E(S)$. Note $S_1 = T_1 - e\prime$ is a spanning tree of G and $w(S_1) = w(T) + w(e) - w(e\prime)$. Because T is a minimum spanning tree of a graph G then $w(S_1) - w(T) \geq 0$ and $w(e\prime) \leq w(e)$. Therefore $e_1, e_2, \ldots, e_{j-1}, e_j$, where $e_j = e$, is the first edges of S and $e_1, e_2, \ldots, e_{j-1}, e\prime$ is the subgraph of T that are not loading circle then by Kruskal algorithm on S we obtain $w(e) = w(e_j) \leq w(e\prime)$. So $w(e) = w(e\prime)$. Because $e\prime$ is the edge with the greatest weight on the circle C then for each $e_i \in C$ apply $w(e_i) \leq w(e) = w(e\prime)$. In the same way we can do the same thing for each edge of next member of $E(S) \setminus E(T)$. ■

Corollary 3.2. Different minimum spanning tree of a weight graph $G(V, E)$ may only differ in the edge with the same weight.

Proof. Let S, T minimum spanning tree which is different from a weight graph $G(V, E)$. Suppose $e \in E(S) \setminus E(T)$, then by Theorem 1 there is $e\prime \in E(T) \setminus E(S)$, where $w(e) = w(e\prime)$. ■

Based on the above theorem, we will write theorems and proofs that are important in this paper. This theorem is the answer to the problems posed in Section 1.

Theorem 3.3. Let $G(V, E)$ is a weight graph. Let $\alpha \in \mathbf{R}$, and S, T is a minimum spanning tree of the graph G. Suppose that all the edges with weights greater than or equal to α are removed from S, and T. Then the connected components $S_1, S_2, \ldots, S_p$ of the tree S and $T_1, T_2, \ldots, T_q$ of the tree T will be formed. Then $p = q$ and if $V(S_i) \cap V(T_j) \neq \oslash$ then $V(S_i) = V(T_j)$.

Proof. In the case of $E(S_i) = E(T_j)$, $V(S_i) = V(T_j)$ is proved. So we just have to prove to the case $E(S_i) \neq E(T_j)$. Because $V(S_i) \cap V(T_j) \neq \oslash$ then, no less generality, suppose there is only one $e \in E(S_i) \setminus E(T_j)$. By Theorem 10, there is exactly one circle C in $T_j + e$ containing edge e such that C contains edge $e\prime \in E(T_j) \setminus E(S_i)$ where $w(e) = w(e\prime)$, $E(C - e) \subseteq E(T_j)$, $E(C - e\prime) \subseteq E(S_i)$, and each $e_i \in C$ satisfy $w(e_i) \leq w(e) = w(e\prime) < \alpha$. So $E(T_j - e\prime) = E(S_i - e)$ and $V(T_j - e) = V(S_i - e\prime)$. Because $V(C - e) = V(C - e\prime) = V(C)$ so that $V(e) \subseteq V(C) = V(C - e) \subseteq V(T_j)$ and $V(e\prime) \subseteq V(C) = V(C - e\prime) \subseteq V(S_i)$. Then $V(S_i) = V(T_j)$. Furthermore, since $\cup_{i=1}^{q} V(S_i) = \cup_{i=1}^{p} V(T_j)$, number of connected component of S and T are finite, and there is bijective mapping of S to T. So the number of connected components of S equals T. Then $p = q$. ■

Theorem 12 says that although S, and T minimum spanning tree of a graph G is different, but each set of points from a connected component of S had the same pair of a set of points from a connected component of T. If that graph is viewed as an image, one of the connected components will form a segment of an image. It can be concluded that Image segmentation Generated by Different Minimum spanning tree is unique.

4. Conclusion

Having regard to all of the above discussion, we can draw some conclusions. Image can be viewed as a connected graph. Spanning tree of a connected graph can always be built and not always single. So the minimum spanning tree of the connected graph can always be built and not always unique. If the edge that is a weight greater than a threshold of minimum spanning tree is removed, it will form some of the connected components. One of the connected components will form a segment of an image. Theorem 12 concludes that the image segmentation generated by different minimum spanning tree is unique.

References

[1] D. Freedman, and P. Kisilev , KDE Paring and a Faster Mean Shift Algorithm, *SIAM J. Imaging Sci.*, **3(4)**, (2010), 878–903.

[2] E. Manik, 2010, Pengaruh Bandwidth Terhadap Segmentasi Citra Digital Dengan Menggunakan Mean Shift, *VISI*, **18(1)**, (2010), 43–49.

[3] T. Uemura, G. Koutaki, and K. Uchimura, Image Segmentation Based on Edge Detection Using Boundary Code' *Proc. ICIC International*, (2011).

[4] S.J. Peter, Minimum Spanning Tree-based Structural Similarity Clustering for Image Mining with Local Region Outliers, *IJCA*, **8(6)**, (2010), 0975–8887.

[5] B. Korte, and J. Vygen, *Combinatorial Optimization: Theory and Algorithms, Third Edition*. Springer-Verlag Berlin (2006).

[6] R.W. Hung, "Hamiltonian cycles in linear-convex supergrid graphs," *Discrete Applied Mathematics*, **211**, (2016), 99–112.

[7] E.W. Mayr, and C.G. Plaxton, On the spanning trees of weight graphs, *Combinatorica*, **12(4)**, (1992), 433–447.