SIMULATION OF TRAFFIC CONGESTION AT THE TOURIST ATTRACTION SPOT OF KUCHING WATERFRONT, SARAWAK

JONATHAN SIDI, LIEW WEI FA, SYAHRUL N. JUNAINI

Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak (UNIMAS)
94300 Kota Samarahan, Sarawak, Malaysia

This paper presents reports a design and development of an interactive map using Scalable Vector Graphics (SVG). In this paper we used this map to simulate traffic congestion at the tourist attraction spot of Kuching Waterfront, Sarawak. We argue that with our map application, the tourists are able to plan their way by finding and selecting the shortest path along the intended route.

1. Introduction

An interactive map normally provides navigation function such as move the map, zoom-in and zoom-out function. When the user wants to zoom-in and zoom-out the map, they can draw a rectangular inside the map and the area inside the rectangular will be zoom-in or zoom-out. Besides, this system also provides print function for user to print the map. An interactive map was implemented using Dijkstra's algorithm to help tourist to find second shortest path in Kuching Waterfront area. If there is traffic congestion occurred along the path, the map will suggest the tourist with the alternative route.

The main objective of our interactive map is to enable tourists to select and plan their route along Kuching Waterfront. Besides that, this interactive SVG Map will guide them to avoid the traffic congestion. Besides that, our proposed interactive SVG map is able to find the shortest path of a route.

The rest of this paper is organized as follows. Section 2 discusses related application of SVG in map and several related algorithm. Section 3 reports the design and development process of our interactive map. Finally, section 4 concludes the paper.

2. Related Work

The problem of finding a path linking two vertices in a graph is generally referred to as pathfinding [5]. In pathfinding, there are several existing algorithm that can find shortest path. The shortest path problem is vital to various applications including routing [1][4]. However, the problem of computing the shortest paths data has acquired little attention in the literature [2]. We have studied various algorithms such as A* Pathfinding Algorithm, Euclidean shortest path algorithm and Single source shortest path algorithm (Dijkstra's algorithm) are studied. The most suitable algorithm was used in our interactive SVG map.

Some pathfinding techniques such as A* involve searching through a graph of nodes for a path [6]. A* Pathfinding algorithm is more suitable if implement in game application. Euclidean shortest path algorithm does have the pseudo code and it is ready to implement in any programming language. However, it only focuses on solving shortest path problem in a simple polygon with m polygonal holes [3]. For Single Source Source Shortest Path algorithm or Dijkstra's algorithm, it is normally used to find shortest path in a graph and find the shortest path more faster as it search the shortest path in parallel. Lastly, for the Calculate Shortest distance algorithm, it compares the pre-drawn paths from one place to another place then finds the paths which have the shortest length. However, it is not as flexible as Single Source Shortest Path algorithm or Dijkstra's algorithm as it needs to pre-drawn all the paths from one place to all other places.

Based from the comparison of both algorithms at previous section, we found that Single Source Shortest Path algorithm or Dijkstra's algorithm is more suitable in mapping application because it is able to determine and calculate distance from start position to destination and fast in searching. By implement this algorithm, the nodes inside the algorithm will be representing the places which can be selected by user as origin and destination.

978-0-7695-3892-1/09 $26.00 © 2009 IEEE
DOI 10.1109/ICCTD.2009.225