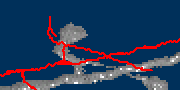
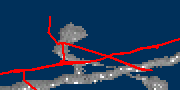
**S.3 Polygonal Path Approximation**

1. (b)

Figure S2: A zoom-in view of the traced result. (a) Red lines show the tracing result; (b) The *ϵ*-approximation of the tracing result, where *ϵ* =.

Given a polygonal path *S* =*< v*0*, . . . , vm >* and an error bound *ϵ*, we look for a polygonal path, ,that is an *ϵ*-approximation of *S*. =*< u*0*, . . . , um >* optimally *ϵ*-approximates *S* if meets the following criteria.

1. Vertex set of is a subset of *S*.
2. Let *ui* = *vj* and *ui*+1 = *vk, i* = 1*, . . . ,m -*1, the distance between any vertex on the polygonal path *< vj . . . , vk >* and the line segment *< ui, ui*+1 *>* is less than *ϵ*.
3. The number of the vertices on is the smallest possible.

This problem can be solved using the dynamic programming technique. We define the number of edges on to be its cost. The lowest cost among all the *ϵ*-approximations for *S* is the optimal cost, denoted *c*(*i, j*). For the boundary condition that *i* = *j*, we let *c*(*i, j*) = 0. If *j > i*, there are two cases for establishing the optimal *ϵ*-approximation path.

* Case 1: is the line segment (*vi*, *vj*)

This case occurs when all the distances between vertices *vk*, *i* ≤ *k* ≤ *j*, to   
(*vi*, *vj*) are less than *ϵ*. (*vi*, *vj*) *ϵ*-approximates *< vi, vi+*1*, . . . , vj >*, and thus   
*c*(*i*, *j*) = 1

* Case 2: consists of two or more line segments.

I In this case, can be divided into two sub-paths, and , where *vk* is a vertex on *< vi, . . . , vj >*. Note that both and *ϵ*-approximate polygonal paths *< vi, . . . , vk >* and *< vk, . . . , vj >*. The cost of optimal *ϵ*-approximation *c*(*i, j*) is .

Based on the above discussion, the optimal cost can be written in the recurrence

Solving the dynamic programming, we compute the optimal *ϵ*-approximation of a polygonal path.