

Tarea 2 (versión 2)

Prof. M.C. Rivara 2017/1

Elija una de las dos tareas siguientes:

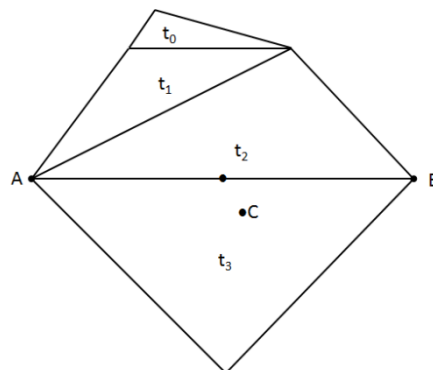
TAREA A

Implemente un algoritmo de simplificación por refinamiento, usando triangulación dependiente de los datos. Input: grillas de datos de terrenos o imágenes a color.

TAREA B

Implemente el algoritmo Lepp Centroide / Flipping de Arista Terminal para construir triangulaciones de buena calidad basado en las siguientes ideas:

- i) Dada una triangulación Delaunay (o no Delaunay) inicial (de mala calidad) de la geometría.
- ii) Encuentre el conjunto S de los triángulos malos con ángulo más pequeño menor o igual que θ_{tol} ($\theta_{tol} \leq 30^\circ$).
- iii) Para cada triángulo t_0 en S encuentre lista ordenada $Lepp(t_0) = \{t_0, t_1, \dots, t_n\}$ donde t_i es vecino de t_{i-1} por la arista más larga de t_{i-1} . Este camino es finito y se detiene cuando t_{n-1}, t_n son triángulos (terminales) que comparten la misma arista más larga (arista terminal).
Caso especial: t_n tiene arista terminal sobre el borde.
- iv) Si los triángulos terminales son localmente Delaunay, seleccione el Centroide C de los triángulos terminales e insértelo en la triangulación uniendo C con los vértices del cuadrilátero terminal. En caso contrario realice flipping de la arista terminal.
- v) Actualice S (la inserción de C puede crear triángulos malos y eliminar otros)
- vi) Repita mientras t_0 permanezca en la triangulación.



AB arista terminal

t_2, t_3 triángulos terminales

A continuación se describe en pseudolenguaje el algoritmo que realiza bisección por la arista más larga en vez de la inserción del centroide

Algorithm 2 Terminal Edge Bisection Flipping algorithm

Input: triangulation τ , set S_{ref} of triangles to be refined
Output: triangulation τ' such that each $t \in S_{ref}$ has been refined

```
for each  $t$  in  $S_{ref}$  do
  while  $t$  remains in  $\tau$  do
    Find  $Lepp(t)$ , terminal triangles  $t_1, t_2$  and terminal edge  $l$ . Triangle  $t_2$  can be null for boundary  $l$ 
    if  $t_1, t_2$  are non locally Delaunay then
      Perform flipping of terminal edge  $l$ 
    else
      Select point  $P$ , midpoint of edge  $l$ 
      Perform bisection by  $P$  of triangles  $t_1, t_2$ 
    end if
    Update  $S_{ref}$ 
  end while
end for
```

Observaciones

1. Elija una operación entre bisección por la arista más larga o inserción del centroide.
2. En vez de usar el test del círculo para determinar si los triángulos terminales son Delaunay, puede usar otra función geométrica.

Realice pruebas al menos con 4 triangulaciones (no triviales) de mala calidad. Detenga su software por número de puntos por si no converge.

Escriba una presentación / informe de su trabajo que incluya:

1. Introducción al problema
2. Solución elegida
3. Discusión de la implementación
4. Trabajo empírico
5. Discusión de dificultades
6. Conclusiones

Entrega: 27 de mayo, presentación 29 de mayo