**Tema d’esame**

**Sviluppare un’applicazione 3D basata su OpenGL e GLUT per la visualizzazione di   
colate laviche su topografie reali**

Il dato topografico è fornito attraverso un file della forma

ncols 10

nrows 10

xllcorner 2487289.5023187

yllcorner 4519797.0771414

cellsize 10.0

NODATA\_value -9999

9.00 8.00 7.00 6.00 5.00 4.00 3.00 2.00 1.00 0.00

8.00 7.00 6.00 5.00 4.00 3.00 2.00 1.00 0.00 1.00

7.00 6.00 5.00 4.00 3.00 2.00 1.00 0.00 1.00 2.00

6.00 5.00 4.00 3.00 2.00 1.00 0.00 1.00 2.00 3.00

5.00 4.00 3.00 2.00 1.00 0.00 1.00 2.00 3.00 4.00

4.00 3.00 2.00 1.00 0.00 1.00 2.00 3.00 4.00 5.00

3.00 2.00 1.00 0.00 1.00 2.00 3.00 4.00 5.00 6.00

2.00 1.00 0.00 1.00 2.00 3.00 4.00 5.00 6.00 7.00

1.00 0.00 1.00 2.00 3.00 4.00 5.00 6.00 7.00 8.00

0.00 1.00 2.00 3.00 4.00 5.00 6.00 7.00 8.00 9.00

dove l’iintestazione ha il seguente significato:

ncols 10

nrows 10

indicano che i dati sono organizzati in una matrice (di quote) 10x10

xllcorner 2487289.5023187

yllcorner 4519797.0771414

specificano le coordinate geografiche dell’angolo in basso a sinistra del suolo (ignorarle per ora)

cellsize 10.0

indica che la discretizzazione della superfice del suolo è caratterizzata da celle (quadrate) di 10 m

NODATA\_value -9999

Indica l’assenza di un dato significativo

I dati che seguono l’intestazione rappresentano la matrice delle quote, ognuna delle quali rappresenta la quota media della cella (di 10 m di lato nell’esempio specifico).

In Fig. 1a è rappresentato il dato topografico dell’esempio precedente mentre in Fig. 1b un dato alternativo di dimensioni maggiori. Si usi inizialmente la tecnica del *color mapping* assegnando il colore nero alla cella con quota più bassa, colore bianco a quella con quota più alta e gradazioni di grigio alle celle con quote intermedie.

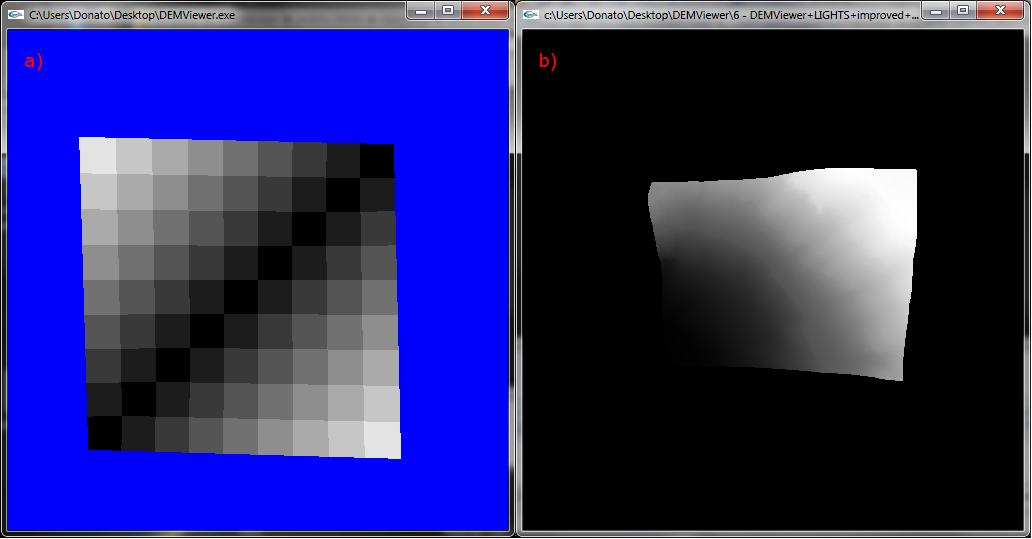
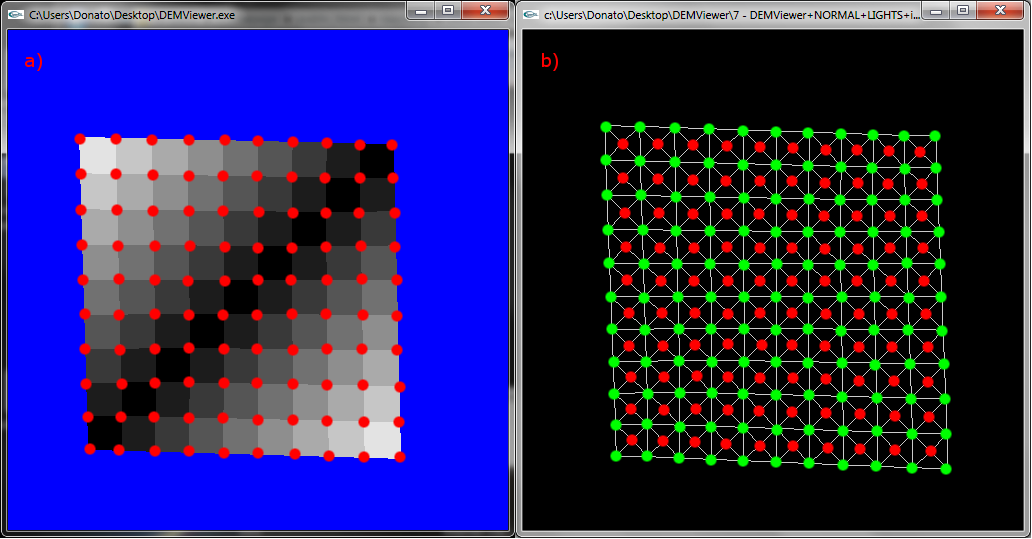


Fig. 1. a) Rappresentazione del dato topografico utilizzato nell’esempio precedente. b) Rappresentazione di un dato topografico di grande dimensione.

Inizialmente si realizzi un modello in cui i punti quotati sono congiunti attraverso triangle strips, una triangle strip per ogni riga della matrice (rappresentazione a griglia). Il colore delle celle è calcolato in relazione al punto in alto a sinistra. In tal modo, una matrice di n\*m punti quotati produce (n-1)\*(m-1) celle, come illustrato in Fig. 2a. Si faccia attenzione alla corretta gestione dei NODATA\_value, che non devono essere visualizzati.

A questa prima tassellazione, affiancare una tassellazione basata su celle in modo che il punto quotato sia il centro di una cella costruita intorno a esso, come illutrato in Fig 2b. In tal modo il colore della cella è relativo al centro della cella stessa che ne rappresenta, per definizione, la quota media. Inoltre, una matrice di n\*m punti quotati produce esattamente n\*m celle. Si faccia, anche in questo caso, attenzione alla corretta gestione dei NODATA\_value, che non devono essere visualizzati.

La rappresentazione basata su griglia è più efficiente, ma rappresenta in modo grossolano il dataset. La rappresentazione basata su celle, invece, rappresenta meglio il dataset, ma è notevolmente più complessa. Infatti, il numero di poligoni necessari aumenta considerevolmente. Inoltre, il rendering è ulteriormente appesantito dal fatto che è necessario calcolare i vertici non noti, in verde in Fig 2b. Per migliorare le performance dell’applicazione, ricorrere all’uso delle display list.

  
Fig. 2. a) Rappresentazione a griglia dei punti quotati (in rosso). Si noti che una griglia di 10\*10 punti è rappresentata con una matrice di 9\*9 celle. b) Rappresentazione a celle dei punti quotati (in rosso). Intorno al generico punto quotato è costruita una cella a partire dai vertici derivati (in verde). Così, una griglia di 10\*10 punti è rappresentata con una matrice di 10\*10 celle.

Lo studente è tenuto a realizzare un’applicazione generale che sia in grado di adattarsi alle dimensioni del dataset da visualizzare parametrizzando opportunamente il volume di vista. Lo studente è inoltre tenuto a definire le seguenti trasformazioni di modeling in modo che sia possibile visualizzare il dataset da ogni punto di vista:

* Rotazioni intorno agli assi x, y e z con centro di rotazione nel centro del dataset
* Traslazioni nelle direzioni x, y e z

È preferibile che durante le traslazioni il mouse insista sempre sulla stessa cella in modo da ottenere un effetto ancoraggio. In caso contrario si potrebbe avere l’impressione che il dataset scappi via nel caso si muova più velocemente del mouse o che reagisca troppo lentamente all’azione dell’utente nel caso si muova troppo lentamente.

Una volta realizzata l’applicazione base, lo studente è invitato a introdurre le luci nella scena in modo da rappresentare il dataset in maniera più realistica. È sufficiente definire una luce posizionale (cioè con una componente diffusa), GL\_LIGHT0, nella posizione della camera. Aggiungere facoltativamente le componenti ambiente e speculare (la componente speculare rende la luce direzionale). Le componenti di GL\_LIGHT0 devono essere definite come segue:

GLfloat ambientLight[] = {0.2, 0.2, 0.2}; //OBBLIGATORIA  
GLfloat diffuseLight[] = {0.8, 0.8, 0.8}; //FACOLTATIVA  
GLfloat specularLight[] = {0.1, 0.1, 0.1}; //FACOLTATIVA

Con l’introduzione di GL\_LIGHT0 è possibile optare per un ombreggiatura uniforme del dataset. A fianco al metodo del color mapping a scala di grigi, lo studente implementi anche un color mapping costante dove le celle del dataset sono ombreggiate col colore bianco. Definire una normale per ogni cella e, facoltativamente, una normale per ogni vertice. L’effetto risultante dovrebbe essere simile a quello in Fig. 3.

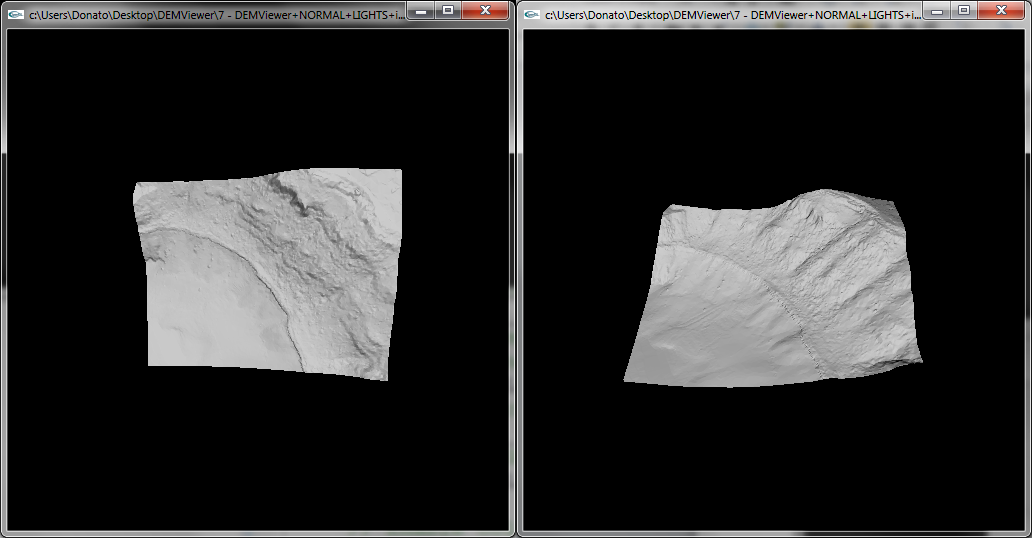


Fig. 3. Rappresentazione del dato topografico con un modello d’illuminazione basato su una luce posizionale posta nella stessa posizione della camera. Le celle sono colorate uniformemente di bianco. Le differenti tonalità sono dovuta alla presenza di una normale per cella.

A questo punto, lo studente deve estendere il visualizzatore per renderlo in grado di applicare una texture e di rappresentare un flusso, ad esempio un flusso lavico o detritico, che si sviluppa sulla superfice topografica. A tal proposito, negli allegati è possibile trovare il dataset “dataset\_completo\_lava” composto da:

* topografia (topografia.dat)
* colata lavica (lava.dat)
* temperatura della lava (temperature.dat)
* foto aerea della topografia (texture.tif)

Gli elementi del dataset, a eccezione della texture, sono nello stesso formato della topografia.

Per concludere, lo studente è invitato a visualizzare il sistema di riferimento nell’applicazione. A tal proposito, per ottenere un risultato soddisfacente, lo studente può far riferimento alle GLU Quadrics, benchè non siano state trattate durante il coro. Lo studente, se ritiene, può far riferimento alla OpenGL Supe Bible. Facoltativamente, lo studente può etichettare gli assi.

Ancora, lo studente deve definire un bounding box, un box 3D in wireframe, che contenga perfettamente il dataset visualizzato.

Infine, lo studente deve definire una legenda per evidenziare il significato dei colori utilizzati per rappresentare la colata. In Fig. 4 è mostrata una legenda che rappresenta le diverse temperature che caratterizzano la colata.

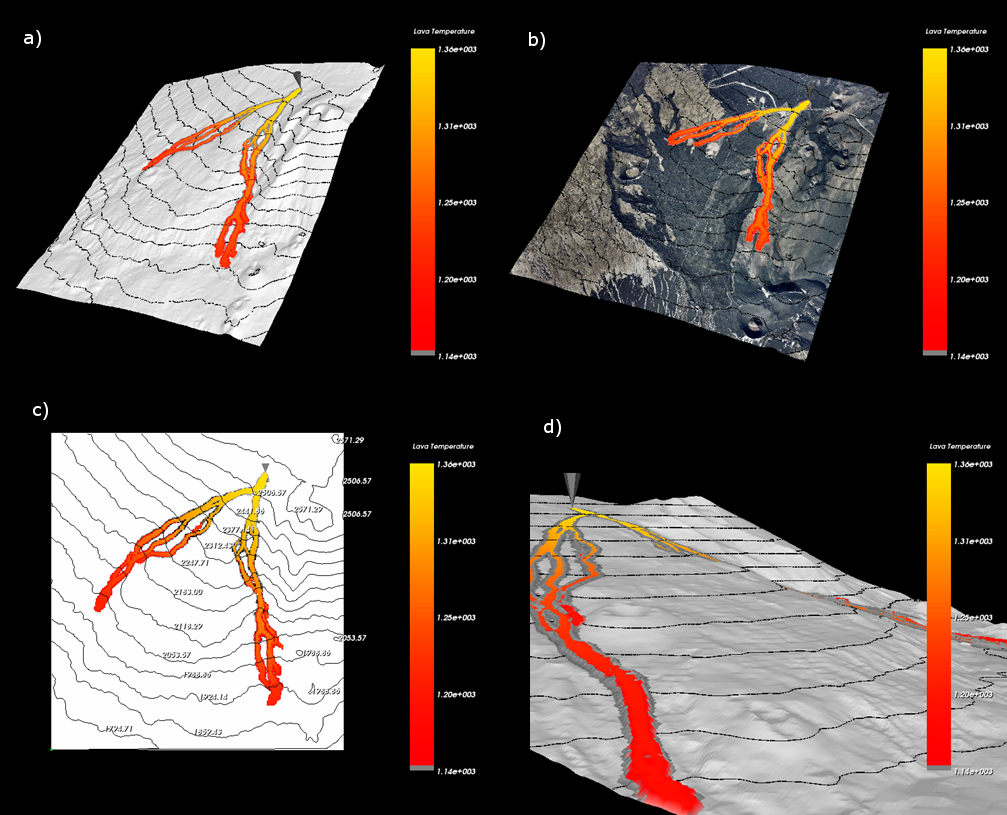


Fig. 4. Risultato finale dell’applicazione. a) Vista 3D del dataset “dataset\_completo\_lava” con vusualizzate la topografia, la colata lavica, la legenda e le curve di livello sulla topografia (queste ultime non richieste per l’esame). b) Vista 3D con in più l’applicazione della texture. c) Vista 2D del dataset (non richiesto per l’esame). d) Vista 3D in cui si vedono porzioni di lava solidificata, in grigio (non richiesto per l’esame).