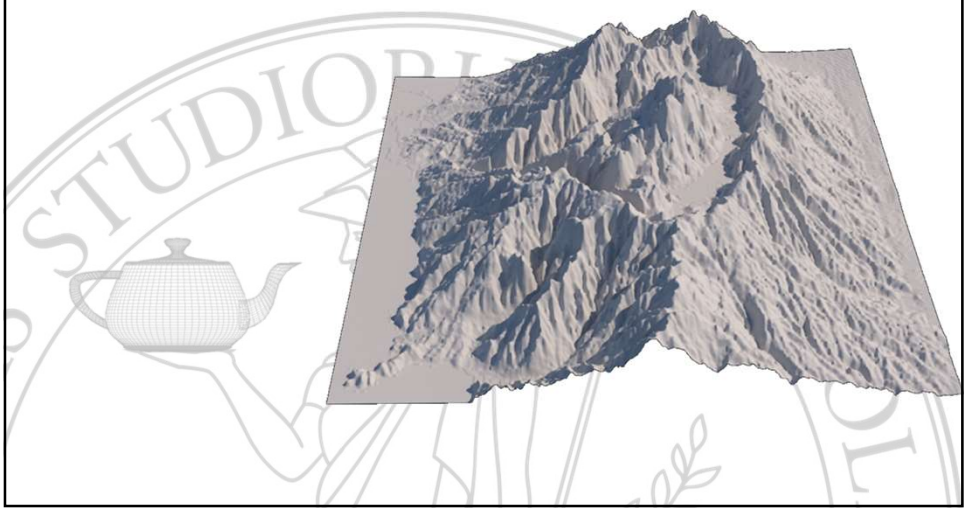
Marco Tarini - Computer Graphics 2023/2024
Università degli Studi di Milano

Dataset superficiali regolari



2

Una (imprefetta) categorizzazione dei tipi di modelli digitali 3D

		ELEMENTI DISCRETI			CONTINUI
		regolari *«a griglia»*	semi-regolari o irregolari: elementi simpliciali	semi-regolari o irregolari: elementi non simpliciali	
SUPERFICIALI	2-manifold *«rappresenta una vera superficie»*	**Height Field**, **Range Scan**	**Triangle Mesh**	**Polygonal Mesh**, **Quad Mesh**, **Quad dominant Mesh**	**Subdivision surfaces**; **Parametric Surfaces** (es. B-splines)
	non-manifold *«non rappresenta una sup»*	**Set di Range Scan**	**Point Cloud**		
VOLUMETRICI	(3-manifold)	**Voxelized Volume**, **Volumetric Textures**	**Tetra Mesh**	**Hexa Mesh**	**Implicit models** (es. CSG)

3

Height fields & Range scans

- ✓ Sono due strutture dati **superficiali** simili che si basano sullo stesso concetto:
  - ⇒ Campionamento della superficie
  - ⇒ I campioni sono memorizzati su una **griglia regolare 2D**
  - ⇒ Ogni campione è memorizzato da un solo scalare
  - ⇒ **Connettività implicita:** ogni campione è *implicitamente* connesso ai propri vicini sulla griglia

(Un altro caso di questo tipo, che non vedremo, è costituito dalle «Geometry Images»)

5

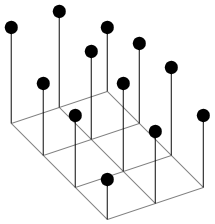
Height fields & Range scans

- ✓ Vantaggi comuni di queste strutture:
  - ⇒ La connettività è **implicita** (non deve essere memorizzata)
  - ⇒ Anche le relazioni di adiacenza fra elementi sono **implicite**
  - ⇒ **Multi-risoluzione** semplice da ottenere (come per le immagini 2D rasterizzate – cioè di pixel)
  - ⇒ Geometry Processing può essere **parallerizzato**: molto facile da ottenere (usando la GPU)
  - ⇒ **Machine learning** su forme 3D: reso semplice dalla regolarità (è come Machine Learning su immagini)
  - ⇒ Analogia con **immagini 2D** (molti vantaggi – es compressione, editing, filtering)
- ✓ Alcuni svantaggi:
  - ⇒ La risoluzione **non è adattiva**
  - ⇒ L'**espressività** è limitata (solo alcune forme possono essere rappresentate)

6

Campo di altezze

- ✓Campo di altezze (height-field)
  - ⇒o mappa di altezze (height map)
  - ⇒o immagine di profondità (depth image)
  - ⇒o mappa di profondità (depth-map)
  - ⇒o modello «2.5D» (2.5D model)



```
float[][] height = new float[resX][resY];
```

7

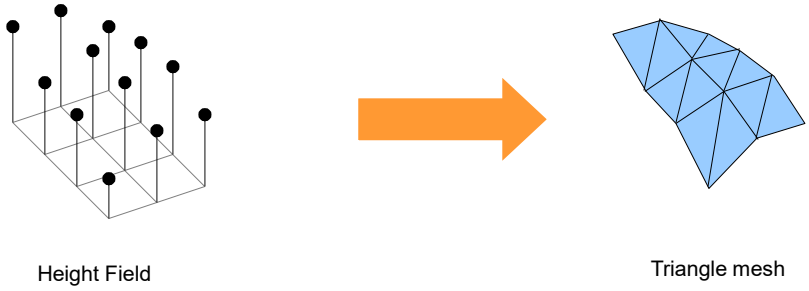
Campo di altezza

```
var height[resX][resY] float;
```

- ✓Array 2D di valori *scalari*
  - ⇒Il valore ad indice x, y rappresenta il punto 3D di coordinate $(x, y, \text{height}[x][y])$
  - ⇒Vantaggio: 2 su 3 delle coordinate non devono essere memorizzate! Solo la z
  - ⇒Molto utilizzato per terreni (per es, in videogames)
  - ⇒E' in pratica una funzione (tabellata) da x,y a z
    - Quindi non sono rappresentabili: sottopassaggi, tunnel, ponti, caverne, burroni spioventi…

8

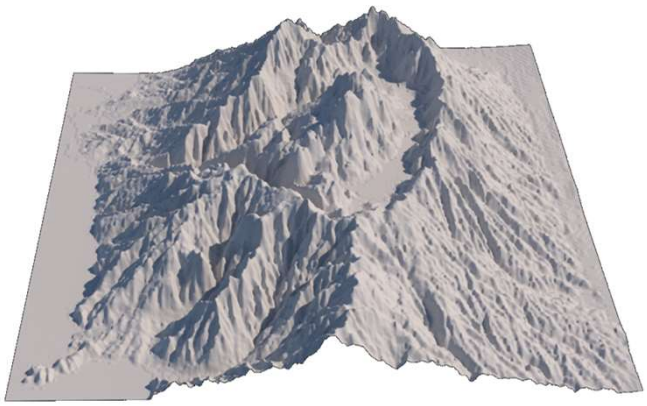
Quasi una mesh (regolare, connettività implicita)



9

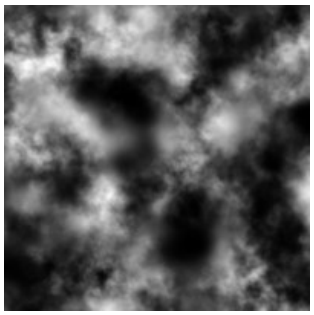
Tipico uso: modellare un terreno

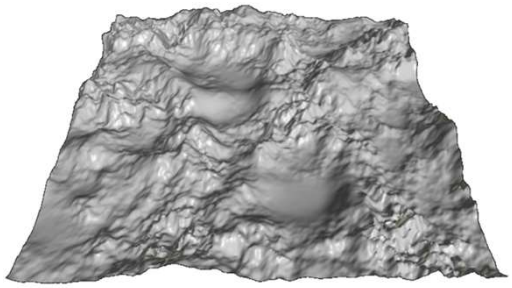
- ✓ un esempio tipico:
  - ⇒ campo d'altezza per modellare un terreno...



10

Campo di altezza: come immagine a scala di grigi

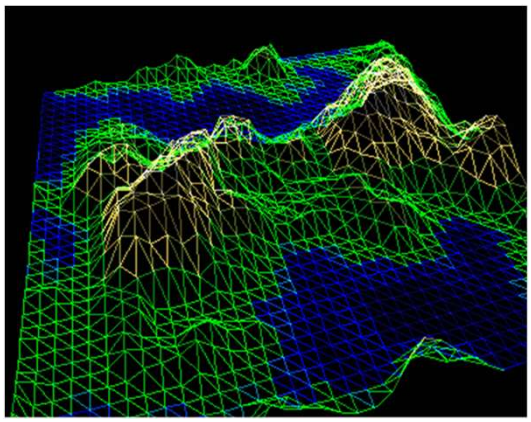




Un'immagine a scala di grigi Gray-scale image,
interpretata come height map

11

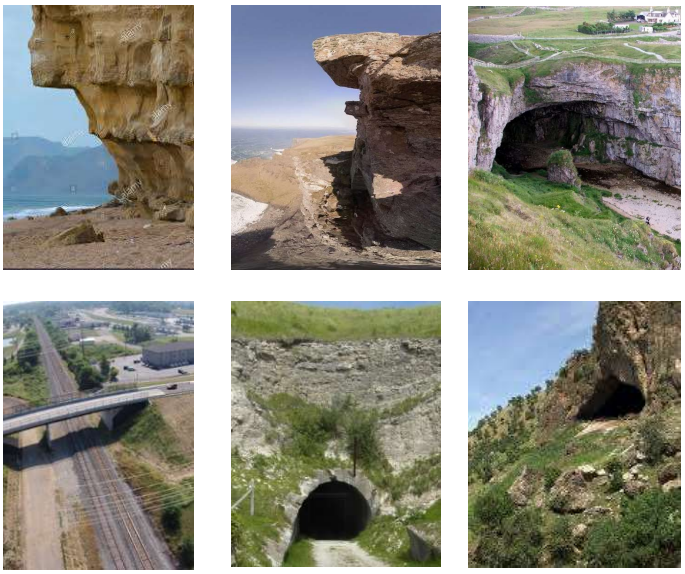
Tipico uso: modellare un terreno



Vedere il primo lucido per un altro esempio, più high res

12

Non tutti i terreni possono essere modellati da campi di altezza…

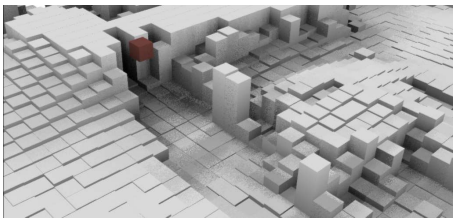


13

Curisosità: height maps come dispositivi

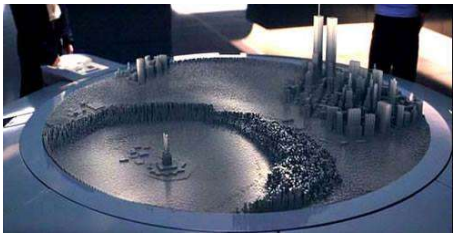
Immaginati dalla fantascienza

o reali

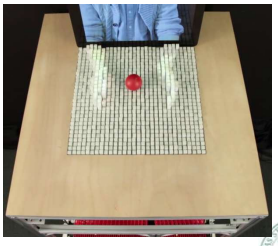


(CGI)

(real)



(CGI)



(real)

18