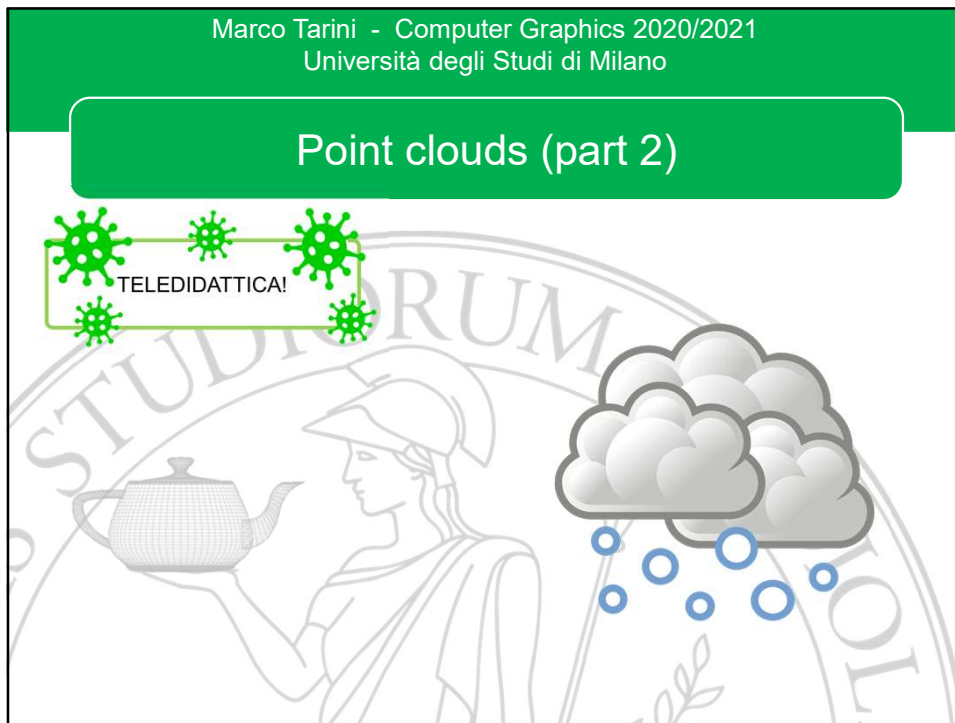


Marco Tarini - Computer Graphics 2020/2021
Università degli Studi di Milano



Point clouds (part 2)

TELEDIDATTICA!




24

Libraries and tools to process point clouds

<p>Point Cloud Library</p> <ul style="list-style-type: none">✓ C++ Lib 	<p>MeshLab</p> <ul style="list-style-type: none">✓ see under Filters → Point Set 
---	--

(actually, more focused on range scans, see later)



25

Geometry Processing on point clouds

Examples of typical tasks: **creation!**

- ✓ Extract point cloud from images
 - ⇒ Shape From Motion, studied in Computer Vision
 - ⇒ see lecture on 3D acquisition
- ✓ Define normals (whenever they are missing)

The availability of easy method to create point clouds (from standard images) is the reason why they are increasingly popular.



26

Geometry Processing on point clouds: surface offsetting

Let's try a simple task first:
offsetting a surface with a point cloud.

- ✓ Surface offsetting:
add a small «offset» to the surface,
e.g. simulating as if a small layer of paint over the surface
- ✓ How to: displace every point
in the cloud P by a small offset k in the normal direction

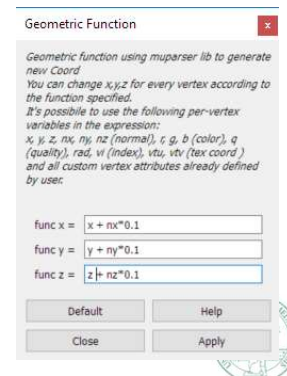
$$\forall \mathbf{p}_i \in P : \mathbf{p}_i \leftarrow \mathbf{p}_i + k \cdot \hat{\mathbf{n}}$$



27

Esperimento in aula: surface offsetting

- ✓ Saricare una point cloud dal sito del corso
 - ⇒ (da ariel)
- ✓ Scaricare, installare ed eseguire MeshLab (sito del corso per il link)
 - ⇒ File => import mesh...
 - ⇒ => selezionare la pointcloud
 - ⇒ Filters => Smoothing, fairing and defor.
 - ⇒ => Geometry Function
 - ⇒ Immettere i valori per applicare la formula
 - il punto \mathbf{p}_i ha coordinate (x,y,z)
 - la normale \hat{n} ha coordinate (nx,ny,nz)
 - come offset usiamo, in questo esempio, 0.1



28

Geometry Processing on point clouds


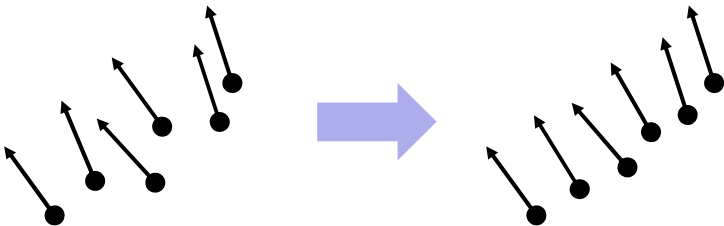
- ✓ Most common sub-task:
find spatial neighborhood of a point in the cloud
 - ⇒ es, find its **k nearest neighbors** (k-NN)
 - ⇒ can be expensive (linear with resolution!)
 - ⇒ remember: each point independent from all others
 - ⇒ see: spatial indexing structures, later
- ✓ A preliminary needed by any other task!



30

Geometry Processing on point clouds: denoising


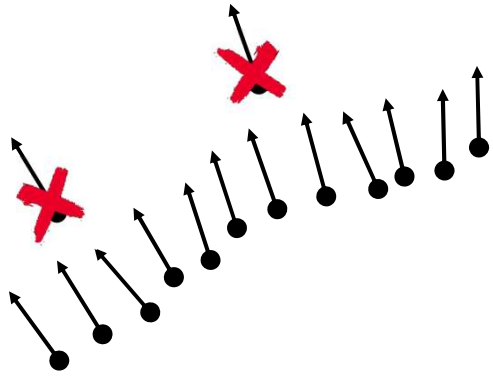
✓ Denoise – Noise reduction – o «fairing»
⇒ on position and or normals



31

Geometry Processing on point clouds: outlier removal

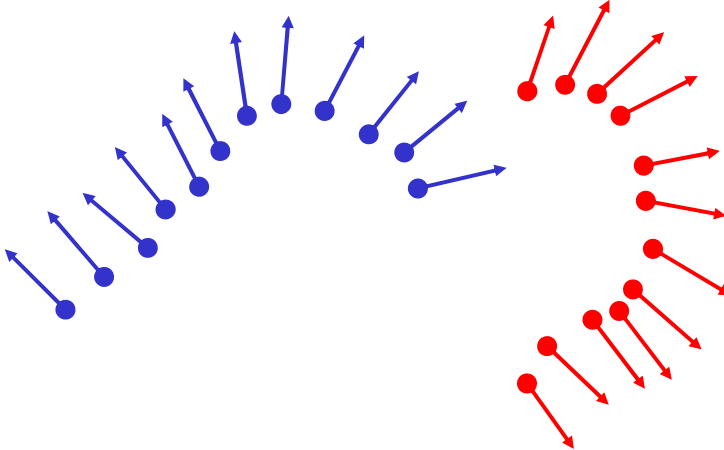
✓ Rimozione degli «outlier»



32

Geometry Processing on point clouds: registration

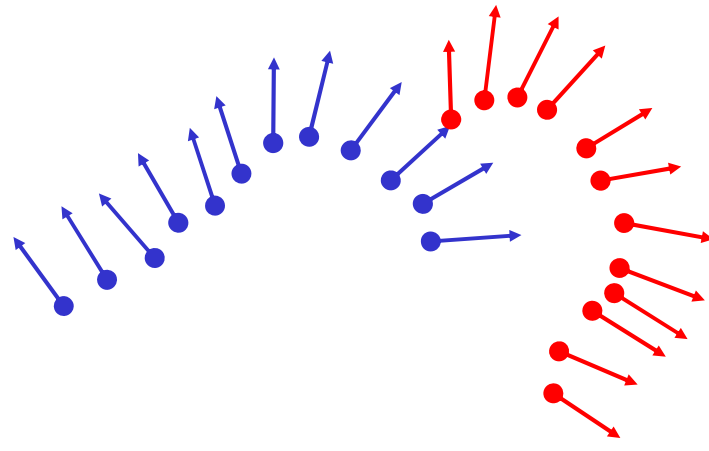
✓ «Registrazione» o «allineamento»
(con un'altra point cloud)



33

Geometry Processing on point clouds: registration


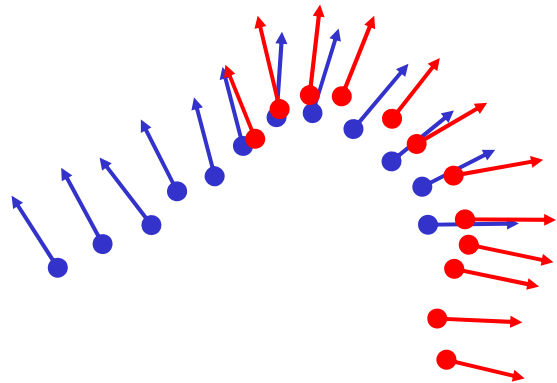
✓ «Registrazione» o «allineamento»
(con un'altra point cloud)



34

Geometry Processing on point clouds: registration


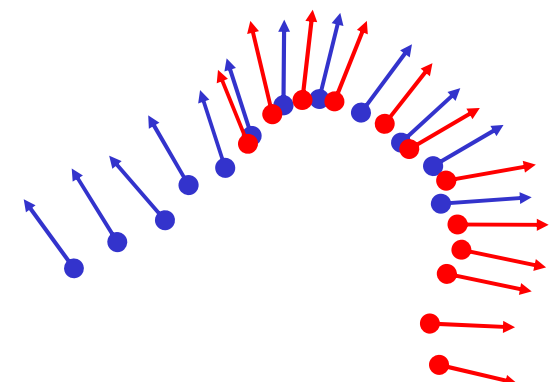
✓ «Registrazione» o «allineamento»
(con un'altra point cloud)



35

Geometry Processing on point clouds: registration

✓ «Registrazione» o «allineamento»
(con un'altra point cloud)



36

Geometry Processing on point clouds: registration

Registering one point cloud over another

- ✓ Step 1: «Coarse Registration»: difficile (es.: a mano)
- ✓ Step 2: «Fine registration»: automatica.
Algoritmo ICP («Iterative Closest Points»)

init 1st iteration 2nd iteration 3rd iteration
(and converged)

37

Geometry Processing on point clouds

Examples of typical tasks: processing

- ✓ Most typical task: convert to ... a mesh
- ✓ see next lecture!

39