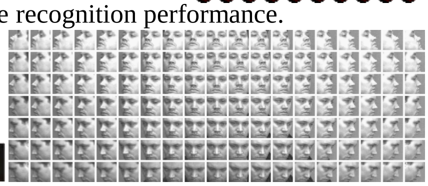
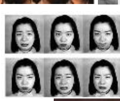
Biometric Systems
Lesson 7: Face recognition – 3D

Maria De Marsico
demarsico@di.uniroma1.it

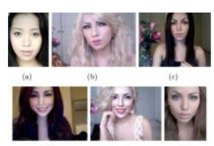
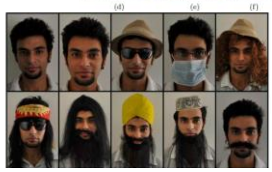
SAPIENZA
UNIVERSITÀ DI ROMA
Dipartimento di Informatica

Face recognition: problems

- Let's recall possible factors affecting face recognition performance.

A-PIE
- PIE
  - Pose 
  - Illumination 
  - Expression 
- Ageing 

Real images: ALL TOGETHER!

(Malicious) Disguise
- Makeup 
- Plastic surgery 
- Glasses, scarfs, etc. 

Face recognition: 2D versus 3D

- Face recognition in 2D is affected by **ALL** factors
- What about using 3D?

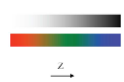
Variation	2D	3D
Pose	Affected	NOT Affected
Illumination	Affected	NOT Affected
Expression	Affected	Affected
Ageing	Affected	Affected
Makeup	Affected	NOT Affected
Plastic surgery	Affected	NOT Affected
Glasses, scarfes, etc.	Affected	Affected

Unfortuntely, no face representation is robust enough to all kinds of variations/distortions

- **3D pros**: more information, robustness to some distortions, possibility to synthetize (approximate) 2D images from virtual 3D poses and expressions computed from a 3D model
- **3D cons**: cost of devices, computational cost of procedures, possible risk (laser scanner is dangerous for the eye
- **In the middle**: techniques to approximate a 3D model from 2D image(s)

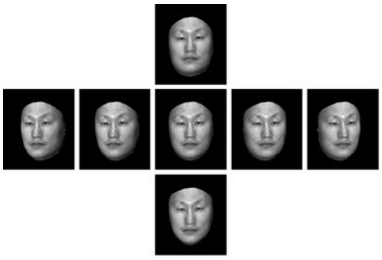
Possible face spaces/representations

- **2D**
  - **Intensity Image**: the usual way we think about a face image, a 2D grid where the value of each pixel is given by the intensity of the illumination reflexed in that point, which depends on the kind of surface and on the kind of illumination, and which can be expressed as a grey value or in colors according to a chosen color space (e.g., RGB).
  - **Images from other sensors** : e.g., thermal iages or NIR images
  - **Processed Image**: the result of processing an intensity image with some filter or transformation (e.g., wavelets or LBP).

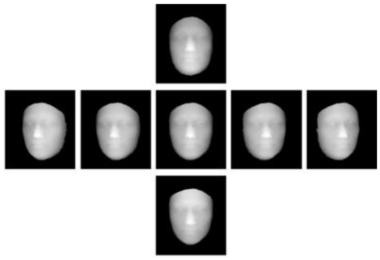


- **2.5D**
  - **Range Image**: a 2D grid where the values of each pixel represents the distance between the point and the light source; it is usually expressed in grey levels, but cal also be expressed in some color space (typically RGB), and the name derives from the fact that values represent an information in the 3D space without building a real 3D model.
- **3D**
  - **Shaded Model**: a structure made of points and polygons connected in the 3D space; each polygon represents a very small face patch, and the smaller the patched, the better the 3D reconstruction.

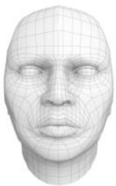
Possible face spaces/representations



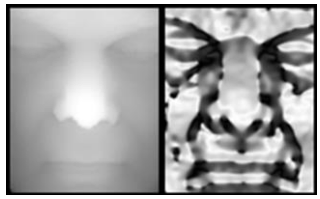
2D



2.5D



3D

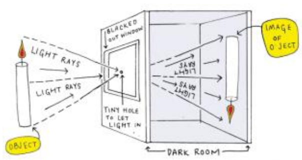


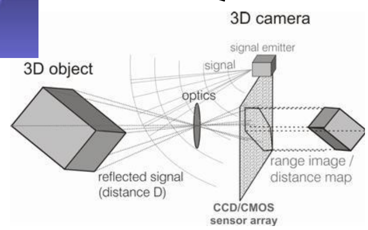
Fusion of range image with thermal image

Possible face spaces/representations

Very broadly speaking, we can say that:

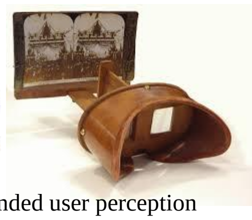
- 2D images encode the result of the interaction between the light coming from a source and the objects in the scene, according to their reflectance properties
- 2.5D images and 3D models encode the result of the interaction between the (structured) light or other beam coming from a source and the objects in the scene, according to their shape properties



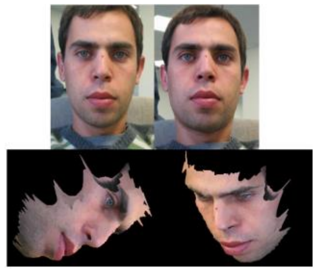


3D acquisition

- **Stereoscopic cameras**:
  - The object (face) is captured by one or more stereoscopic cameras with different points of view
  - A set of relevant features is located on on image and the corresponding sets of features are searched for in the other images
  - The position of each relevant point on the depth axix is given by the geometrical relation among homologous points.
  - Low cost, medium accuracy, low robustness to illumination, real time capture

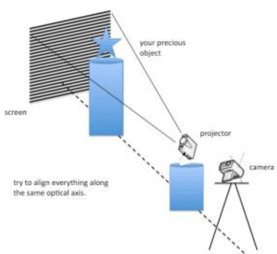
Past application: expanded user perception



Present application: digital 3D models

3D acquisition

- **Structured light scanner**:
  - A single light pattern is projected along the surface (face)
  - The pattern is deformd by the 3D structure of the face and the deformation gives the measure of the depth for each point
  - The capture must be repeated from different points of view to obtain a full 3D model
  - Medium-high cost, medium-high accuracy, medium-high robustness to illumination, 3-8 seconds per scan, NOT DANGEROUS FOR THE EYES



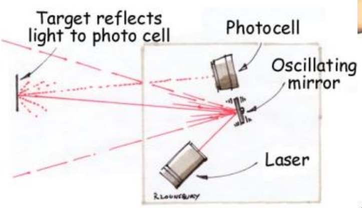
Actually, the result is a 2.5D image

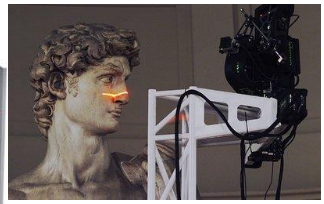
From: http://fab.cba.mit.edu/content/processes/structured_light/

Note: Microsoft Kinect uses a pattern of projected infrared points to generate a dense 3D image.

3D acquisition

- **Laser scanner**:
  - A single laser beam is projected along the surface (face)
  - The beam is deformd by the 3D structure of the face and the deformation gives the measure of the depth for each point
  - The capture must be repeated from different points of view to obtain a full 3D model
  - Medium-high cost, high accuracy, high robustness to illumination, 6-30 seconds per scan, DANGEROUS FOR THE EYES





Preprocessing of 3D scans

- Ideal results …

- … but 3D scans (2.5D images) present some problems
- **Noise removal**: spikes (filters), clutter(manually), noise(median filter)

- **Holes filling** (Gaussian smoothing, linear interpolation, symmetrical interpolation, morphological operators)
- **Smoothing + alignment**

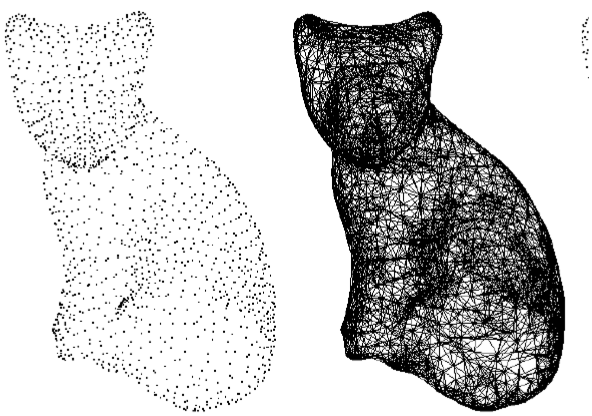
From 2.5D representation to 3D model

- A 3D model is constructed by integrating several 2.5D face scans which are captured from different views:
  - **1**: for each 2.5D image a cloud of 3D points is generated, with x and y coordinates equally spaced and z (depth) coordinate derived from the value in the 2.5D image
  - **2**: the cloud of points is triangularized (a mesh of adjacent triangles is derived)
    - Example from http://pointclouds.org/documentation/tutorials/greedy_projection.php:

      «The method works by maintaining a list of points from which the mesh can be grown ("fringe" points) and extending it until all possible points are connected. It can deal with unorganized points, coming from one or multiple scans, and having multiple connected parts. It works best if the surface is locally smooth and there are smooth transitions between areas with different point densities.

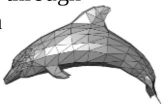
      Triangulation is performed locally, by projecting the local neighborhood of a point along the point's normal, and connecting unconnected points."

From 2.5D representation to 3D model

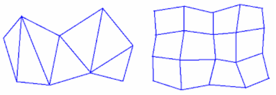


3D model

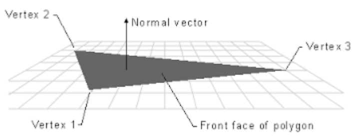
- A polygon is a sequence of coplanar points connected by segments
- Coplanarity is what causes the approximation when we represent a surface through a mesh of polygons: the smaller the polygons, the better the approximation

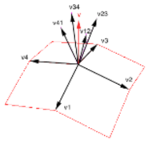


- A side is a segment connecting two vertices
- A polygon is composed of a set of sides (the first vertex of the first side corresponds to the second vertex of the last side)
- Each vertex in the mesh is adjacent to two sides at least
- Each side belongs to one polygon at least



- How can we determine the orientation of a surface in the 3D space? We use the normals (to vertices or to polygons).



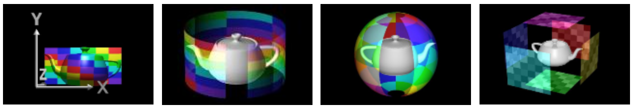


The normal to a polygon is a vector perpendicular to the plane where the polygon lies: it identifies the orientation of the polygon in the space and is computed by the cross product of two vectors in the plane

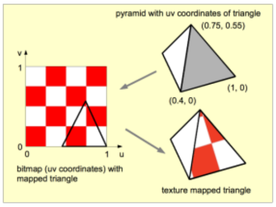
The normal to a vertex is the (normalized) sum of the (unit lenght) normals to its adjacent polygons: it identifies the orientation of the vertex in the space

3D model

- After building the polygonal mesh (geometrical structure) it is necessary to assign colors to vertex and/or polygons (perceptive structure).
- It is possible to assign a color to a vertex
- It is possible to assign a color to a polygon according to the colors of its vertices:
  - a uniform colore derived from the sum of the colors of the vertices
  - a varying color depending from the colors of the vertices and from the distances of each point of the polygon from the different vertices
- Texture mapping: used to compute position and orientation of a texture on a surface
  - Problem: how to map each texture pixel on the correct surface point?
  - It is possible to use different kinds of projections

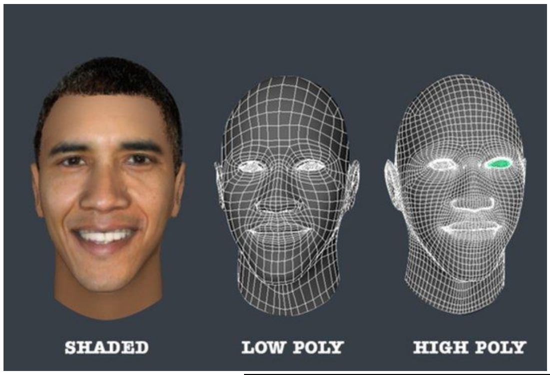


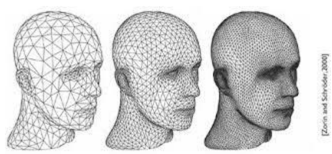
planar | cylindrical | spherical | box

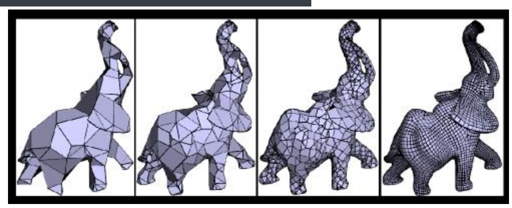


https://www.siggraph.org/education/materials/HyperGraph/mapping/r_wolfe/r_wolfe_mapping_1.htm

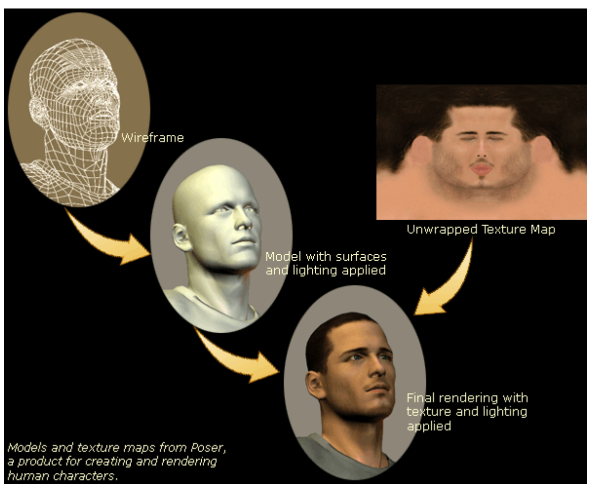
3D model



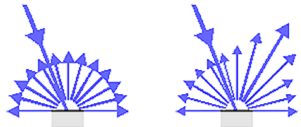




3D model



From 2D representation to 3D model: shape from shading



Ideal diffuse reflection (Lambertian surface) — Diffuse reflection with directional component

- Shading as a cue for shape reconstruction
- What is the relation between intensity and shape?
  - Reflectance Map
- The computation of the map is performed assuming a Lambertian surface
  - Lambertian reflectance is the property that defines an ideal "matte" or diffusely reflecting surface. The apparent brightness of such a surface to an observer is the same regardless of the observer's angle of view. More technically, the surface's luminance is isotropic (isotropic radiation has the same intensity regardless of the direction of measurement), and the luminous intensity obeys Lambert's cosine law: the radiant intensity or luminous intensity observed from an ideal diffusely reflecting surface or ideal diffuse radiator is directly proportional to the cosine of the angle θ between the observer's line of sight and the surface normal
- In general it is not possible to recover shape from a single image → add information about shading

From 2D representation to 3D model: shape from shading

- The procedure starts with an ensemble of shapes of related 3D objects (faces), and uses standard statistical techniques such as PCA to derive a dimensionally reduced representation for shapes in the same class.
- Atick, Griffin & Redlich in 1996 used a database of several hundred laser-scanned heads' to run the procedure for the class of 3D human heads.
- They showed that principal components provide an excellent low-dimensional parameterization of head shape that maintains facial detail and identity of the person.
- They used this representation to solve the shape-from-shading problem for any human head
- They can recover an accurate 3D surface of the head/face of any person from a single 2D image of the face.

From 2D representation to 3D model: shape from shading

- A recent approach from Kemelmacher-Shlizerman and Basri in 2011 avoids representing input faces as combinations (of hundreds) of stored 3D models, and uses only the input image as a guide to ”mold” a single reference model to reach a reconstruction of the sought 3D shape.

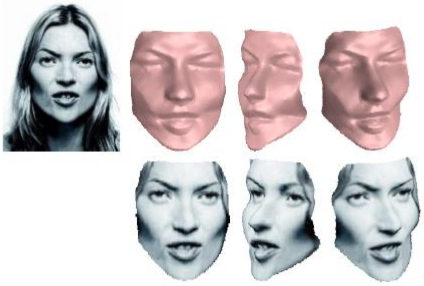


Fig. 1. 3D reconstruction of a face from an image downloaded from the internet using our algorithm. We present the input image (top left), the reconstructed shape viewed from three viewpoints (top right), and the image overlay of the reconstructed shape (bottom right).

From 2D representation to 3D model: morphable models

- This technique uses morphing (typical of image graphics)
- The procedure starts from a generic initial 3D model (morphable model) and arrives to a 3D model for a specific subject
- The initial approach by Blanz and Vetter proposed in 1999 uses a single image, but to improve the results also 2 or 3 images of the face can be used (e.g., frontal, profile, 45°)
- Shape and texture of the generic model are manipulated to adapt to the captured images
- Morphable models also allow to synthetize face expressions approximating the possible expressions of a specific subject

From 2D representation to 3D model: morphable models

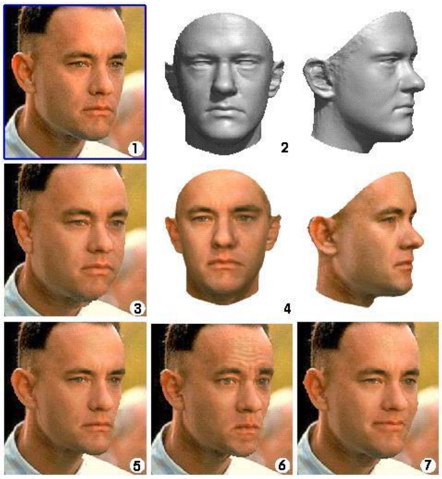


Figure
The figure shows an application of our approach. Matching a morphable model atomatically to a single sample image (1) of a face results in a 3D shape (2) and a texture map estimate. The texture estimate can be improved by additional texture extraction (4). The 3D model is rendered back into the image after changing facial attributes, such as gaining (3) and loosing weight (5), frowning (6), or being forced to smile (7).
(from: http://gravis.cs.unibas.ch/Sigg99.html)

From 2D representation to 3D model: morphable models

- The morphable model is based on a data set of 3D faces. Morphing between faces requires full correspondence (alignment) between all of the faces.
- The geometry of a face is represented by a shape vector composed of the set of X, Y, Z coordinated of the *n* 3D vertices $S=(X_1, Y_1, Z_1, \ldots, X_n, Y_n, Z_n)^T \in \Re^{3n}$
- A similar vector represent the texture of the face through RGB values.
- A morphable face model is constructed using a data set of *m* exemplar faces, each represented by its S_i and T_i vectors
- New shapes and textures can be obtained through a linear combination of the shapes and textures of the *m* exemplars

$$\mathbf{S}_{mod} = \sum_{i=1}^{m} a_i \mathbf{S}_i\,, \quad \mathbf{T}_{mod} = \sum_{i=1}^{m} b_i \mathbf{T}_i\,, \quad \sum_{i=1}^{m} a_i = \sum_{i=1}^{m} b_i = 1.$$

From 2D representation to 3D model: morphable models

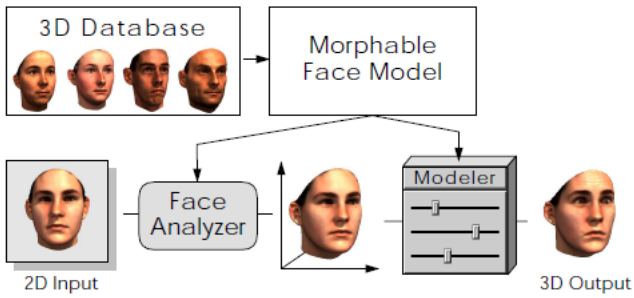


Figure 1: Derived from a dataset of prototypical 3D scans of faces, the morphable face model contributes to two main steps in face manipulation: (1) deriving a 3D face model from a novel image, and (2) modifying shape and texture in a natural way.

From: Blanz and Vetter. 1999

From 2D representation to 3D model: morphable models

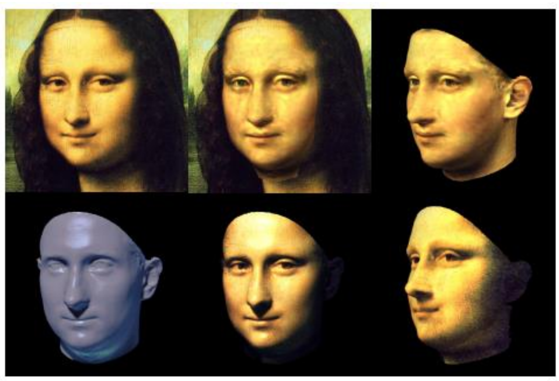
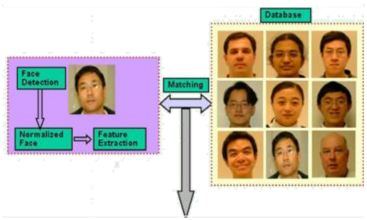
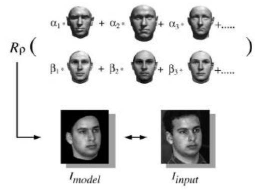
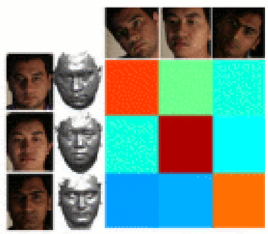


Figure 8: Reconstructed 3D face of Mona Lisa (top center and right). For modifying the illumination, relative changes in color (bottom left) are computed on the 3D face, and then multiplied by the color values in the painting (bottom center). Additional warping generates new orientations (bottom right, see text), while details of the painting, such as brush strokes or cracks, are retained.

From: Blanz and Vetter. 1999

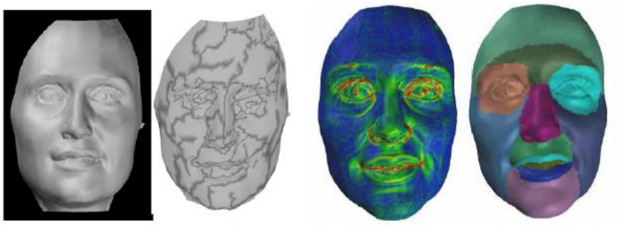
Possible configurations

		PROBE	
		2D	3D
GALLERY	2D		 (Non exactly what we would expect … 3D is an intermediate step)… From: Blanz e Vetter, 2013
	3D	 From: http://web.mit.edu/gevang/www/research.html	 From: http://www.cc.gatech.edu/conferences/3DPVT08/Program/Session-Recognition.html

From the lecture by Gabriele Sabatino «3D face recognition»
in the course «Sistemi Biometrici» by Professor Michele Nappi – University of Salerno

3D features

- 3D face recognition algorithms often work on local and global curvature of the face model.
- In particular, it is possible to extract the information concerning the shape of a 3D face by analyzing the local curvature of the surface.
- Examples:
  Crest Lines: selecting areas with the greatest curvature
  Local Curvature: representing the local curvature with a color
  Local Features: segmenting the face into regions of interest



(a) Crest lines (b) local curvature (c) local features

3D alignement: coarse

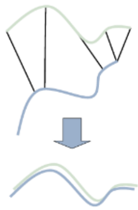
- Alignment can be performed using landmarks

1) Find a finite number of characteristic points of the face (eye corners, nose tip, center of the mouth, etc.).
2) Align faces (rotation, translation, scaling) by minimizing the distance between corresponding points

Points can be located on the 2D image or directly on the 3D model.

3D alignment: fine

- Alignment between models is often required: the ICP (Iterative Closest Point) algorithm is widely used for geometric alignment of three dimensional models when an initial estimate of the relative pose is known.
- The ICP algorithm (Iterative ClosestPoint) is based on the calculation of "volume difference" between two 3D surfaces.
  - Given two 3D surfaces:
  1. Find an initial match between the two surfaces (mapping of points, surfaces, lines, curves)
  2. Compute the distance between the two surfaces by the least squares method
  3. Compute the transformation which minimizes such distance
  4. Perform the transformation and reiterate the procedure until the distance is less than a threshold.



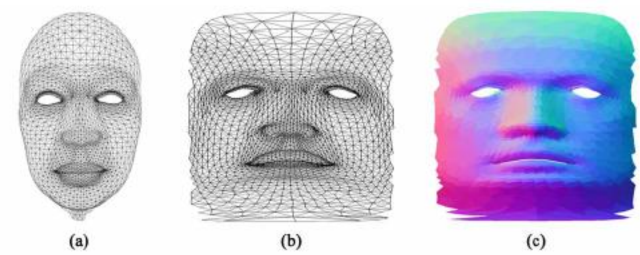
ICP is very accurate but computationally expensive and does not always converge to the optimal solution

3D face recognition with normal maps

a) Acquisition and generation of the face.

b) Projection of geometry from 3D space to a 2D space.

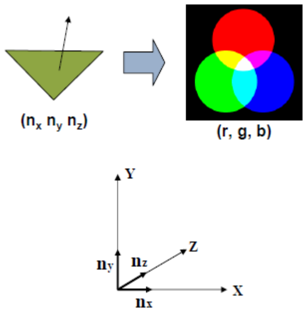
c) Generation of the normal map.



Normal maps are commonly stored as regular RGB images where the RGB components correspond to the X, Y, and Z coordinates, respectively, of the surface normal.

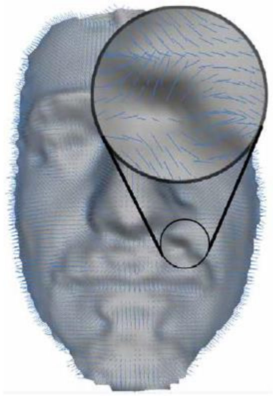
3D face recognition with normal maps

- The components of the normal (unit vectors) are sampled with a triple RGB (Red / Green / Blue) .
- La length of the vector component is represented by the intensity of the color (e.g., if a unit vector n_x is very long then the pixel will have a very high intensity of red)



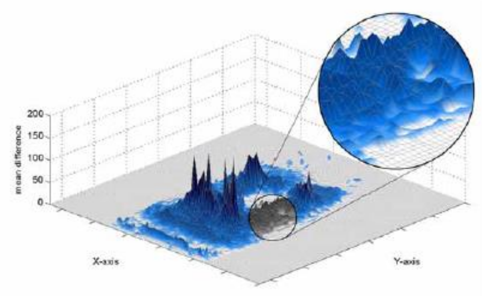
3D face recognition with normal maps

- Using the normal map we obtain a two-dimensional representation of three-dimensional information.
- The information about the curvature of a model is represented by the set of the surface normals
- Reading and processing a 2D image is much faster than reading and processing a 3Dmodel .



3D face recognition with normal maps

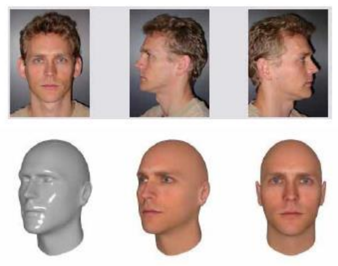
- The difference map is the difference image between two normal maps.
- Each pixel of the difference map represents the angular distance between two normal maps in that point.



3D face recognition via morphable models: FaceGen modeller

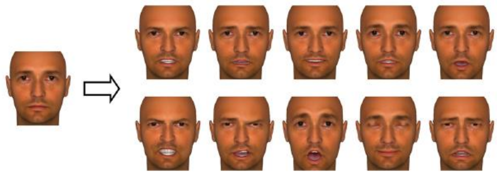
- Facegen Modeller is a commercial tool able to generate the 3D model of the face of an individual “instrumented” with an underlying structure capable of simulating the "dynamics" of the face.
- As we have shown, the algorithm is based on fitting a model "generic" (morphable-model) to the shape and color of the final model of the subject to be captured.
- The adaptation of the morphable model to the final model is driven by a set of facial features extracted directly from the photos.

FaceGen
Shaping faces

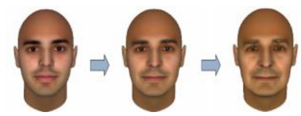


3D face recognition via morphable models: FaceGen modeller

- Morphable-models are a powerful tool, allowing you to generate synthetic facial expressions
- Synthetic facial expressions can be used to obtain a good approximation or real facial expressions that an individual may have during the process of acquisition, and can be used in some cases to add samples to the gallery or to reproduce probe expression on the fly …



- … but also to simulate aging …



3D Face Recognition using iso-Geodesic Stripes (2010)

- Method proposed by Berretti, Del Bimbo and Pala in 2010

- Stuff for next slides taken from:

  S. Berretti, A. Del Bimbo, P. Pala. "3D Face Recognition using iso-Geodesic Stripes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.32, no.12, pp.2162-2177, December 2010.

3D Face Recognition using iso-Geodesic Stripes (2010)

Preliminary considerations:

- 3D face models are insensitive to illumination and pose changes but …
- … they are even more sensitive than 2D images to face expressions
- in presence of an expression change, point positions on the 3D face surface can exhibit large variations.

Important observations:

- geodesic distances (we can assume that the geodesic is the shortest path between two points in a curved space) are slightly affected by changes of facial expressions:
- the geometry of convex face regions is less affected by changes of facial expressions than concave regions.

3D Face Recognition using iso-Geodesic Stripes (2010)

- 3D face recognition is computationally expensive.
- Some studies have demonstrated that the Euclidean and geodesic distances and the angles between as many as 47 fiducial points may suffice to capture most of relevant facial information…
- … however in practice automatic detection of fiducial points is difficult …
- … and the full 3D face surface information must be exploited for matching.
- The conversion of 3D scans to efficient and meaningful descriptors of face structure is crucial to perform fast processing and particularly to permit indexing over large datasets for identification.

3D Face Recognition using iso-Geodesic Stripes (2010)

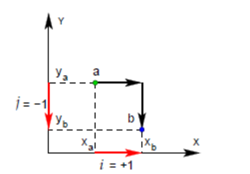
- In this approach, all the points of the face are taken into account so that the complete geometrical information of the 3D face model is exploited.
- The relevant information is encoded into a compact representation in the form of a graph and face recognition is finally reduced to matching the graphs.
- Face graphs have a fixed number of nodes, that respectively represent iso-geodesic facial stripes of equal width and increasing distance from the nose tip.
- Arcs between pairs of nodes are annotated with descriptors referred to as *3D Weighted Walkthroughs* (3DWWs), that capture the mutual spatial displacement between all the pairs of points of the corresponding stripes and show smooth changes of their values as the positions of face points change.
- This representation has the great advantage of a very efficient computation of face descriptors and a very efficient matching operation for face recognition.
- The method obtained the best ranking at the SHREC 2008 contest

Weighted Walkthroughs in 2D and 3D

- In a two-dimensional Cartesian reference system with *X*, *Y* coordinate axes, given two generic points $a = (x_a; y_a)$ and $b = (x_b; y_b)$, their projections on each of the two axes can take three different orders (*b* with respect to *a*): *before*, *coincident*, or *after*, for a total number of nine possible cases of two-dimensional displacements between *a* and *b*.
- Each displacement can be encoded by a pair of indexes $\langle i, j \rangle$, with *i* and *j* taking values in {-1, 0, +1}:

$$i = \begin{cases} -1 & x_b < x_a \\ 0 & x_b = x_a \\ +1 & x_b > x_a \end{cases} \qquad j = \begin{cases} -1 & y_b < y_a \\ 0 & y_b = y_a \\ +1 & y_b > y_a . \end{cases}$$

- A displacement can be regarded as a *walkthrough* from *a* to *b*



2D walkthrough $\langle i, j \rangle = \langle +1, -1 \rangle$

- Given two continuous regions *A* and *B*, points of *A* can be connected to points of *B* by walkthroughs.
- The number of unique pairs (*a*, *b*) with $a \in A$ and $b \in B$ that are connected by the same walkthrough $\langle i, j \rangle$ can be measured and is denoted as $w_{i,j}(A,B)$
- The 3×3 matrix $w(A,B)$ of the weights is referred to as *2D Weighted Walkthrough (2DWW)*, and can be used to model the relative displacement between the two sets.
- Intuitively, 2D Weighted Walkthroughs can measure the mutual spatial distributions of the masses of two 2D regions.
- Extension to 3D of the above takes to the definition of 3DWW

Directional indexes

- *Directional indexes are* aggregate measures that take values in [0; 1] and account for directional information between two 3D spatial entities.
- Directional Indexes can be directly derived from the weights of the 3DWW matrix.

- six directional indexes based on the *corner weights* of $w(A, B)$:

$$\begin{aligned} w_H &= w_{1,1,1} + w_{1,-1,1} + w_{1,1,-1} + w_{1,-1,-1} \\ w_V &= w_{-1,1,1} + w_{1,1,1} + w_{-1,1,-1} + w_{1,1,-1} \\ w_D &= w_{1,1,1} + w_{1,-1,1} + w_{-1,1,1} + w_{-1,-1,1} \\ w_{XY} &= w_{-1,-1,1} + w_{1,1,1} + w_{-1,-1,-1} + w_{1,1,-1} \\ w_{XZ} &= w_{-1,-1,-1} + w_{-1,1,-1} + w_{1,1,1} + w_{1,-1,1} \\ w_{YZ} &= w_{-1,1,1} + w_{1,1,1} + w_{-1,-1,-1} + w_{1,-1,-1} \ , \end{aligned} \tag{3}$$

that express the percentage of pairs of points (a, b), such that b is on the right of a, b is above a and b is in front of a (respectively w_H, w_V, and w_D), and the percentage of pairs of points (a, b), that are distributed along the XY, XZ and YZ diagonal planes of the 3D space (respectively w_{XY}, w_{XZ}, and w_{YZ});

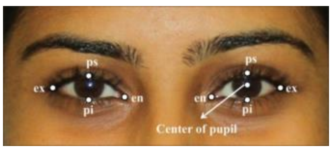
- six directional indexes based on *inner weights* of $w(A, B)$:

$$\begin{aligned} w_{H_0} &= w_{0,1,1} + w_{0,-1,1} + w_{0,1,-1} + w_{0,-1,-1} \\ w_{V_0} &= w_{1,0,1} + w_{-1,0,1} + w_{-1,0,-1} + w_{1,0,-1} \\ w_{D_0} &= w_{1,1,0} + w_{1,-1,0} + w_{-1,1,0} + w_{-1,-1,0} \\ w_{HV_0} &= w_{0,0,1} + w_{0,0,-1} \\ w_{HD_0} &= w_{0,1,0} + w_{0,-1,0} \\ w_{VD_0} &= w_{1,0,0} + w_{-1,0,0} \ , \end{aligned} \tag{4}$$

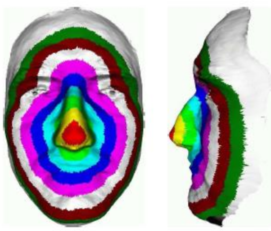
that express the percentage of pairs of points (a, b), that are aligned along the X, Y and Z axes (respectively w_{H_0}, w_{V_0}, and w_{D_0}), and the percentage of pairs of points (a, b), that are aligned along the XY, XZ and YZ axes (respectively w_{HV_0}, w_{HD_0}, and w_{VD_0}).

Iso-geodesic stripes

- Each face is partitioned into a fixed number of iso-geodesic stripes of equal width, pseudo-concentric and centered on the nose tip fiducial point.
- The mutual displacement between each pair of stripes is measured by computing the 3DWWs between all the pairs of points of the two stripes.
- Face stripes are obtained by:
  - computing the normalized geodesic distance $\bar{\gamma}$ between each face point and the nose tip (using the Dijkstra's algorithm applied to the points of the surface).
  - quantizing $\bar{\gamma}$ values into N intervals $c_1, \ldots, c_N$, so that the i-th stripe collects all the face points that have values of $\bar{\gamma}$ within the interval c_i
  - the normalization factor is the Euclidean eyes-to-nose distance (i.e., sum of distances between the nose tip and the two endocanthions): this guarantees invariance with respect to scaling and expression changes.

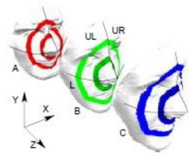


Anthropometric landmarks: Exocanthion (ex), endocanthion (en), palpebrale superius (ps), inferius (pi), and center of pupil

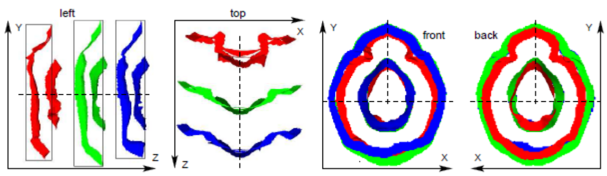


3Dww features

Under appropriate design choices, face partitioning into iso-geodesic stripes of equal width and calculation of 3DWWs between stripes provides an effective representation of 3D faces allowing to distinguish facial differences due to different facial traits of different individuals from differences induced by facial expressions of the same individual.



(a)



(b)

Index	A neutral	B neutral	C smile
w_H	0.45	0.49	0.49
w_V	0.82	0.77	0.77
w_D	0.02	0.12	0.13

(c)

(a) Sample face models (stripes 4 and 7 evidenced). B, C: Models of the same individual with neutral and smiling expression; A: Model of a different individual with neutral expression. (b) Projections of the pairs of face stripes on the coordinate planes. (c) Measures of 3DWW directional indexes w_H, w_V, w_D for the lower part of the pair of stripes of each model. w_H, w_V, w_D are three aggregate measures that can be extracted from w(A, B) that express the percentage of pairs of points (a; b), such that b is on the right of a, b is above a and b is in front of a respectively.

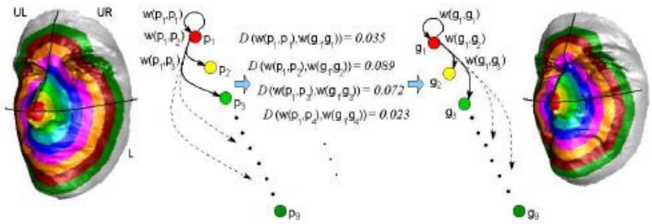
3DWW features

1. Geodesic distances between two facial points keep sufficiently stable under expression changes → **the large majority of the points of each stripe still remain within the same stripe, even under facial expression changes**.
2. Only few parts of the face are affected by deformations due to expressions. If P_1 and P_2 are two generic points on the 3D face surface with neutral expression, and $P'_1 = P_1 + \Delta P_1$ and $P'_2 = P_2 + \Delta P_2$ are the new points under a generic expression, the displacement between P'_1 and P'_2 is: $(P'_2 - P'_1) = (P_2 - P_1) + (\Delta P_2 - \Delta P_1)$
   – the two terms account for the displacement $(P_2 - P_1)$ on the neutral face—that depends on the original face geometry—and the deformation of the face geometry due to the expression $(\Delta P_2 - \Delta P_1)$.
   – due to the constrained elasticity of the skin, neighbor points can be assumed to feature very similar motion vectors for moderate facial expressions in most parts of the face; according to this, for all these points the term $(\Delta P2 - \Delta P1)$ is negligible and **the mutual displacement between the two points is mainly determined by the geometry of the neutral face.**
   – **this property is preserved by 3DWWs that provide an integral measure of displacements between pairs of points.**
3. Due to the property of continuity of 3DWWs, **small variations of the mutual displacement between a limited number of points yield small variations of the 3DWWs descriptor**.

3DWW features

- Reasonable choices for the stripe width, the number of stripes, face partitioning and relative weighting of the different parts can make the described properties hold for the whole face representation.
- Experiments show that use of 9 stripes of width 1*cm* yields good invariance to facial expressions of the same individual and discrimination between different individuals.
- To account for the larger deformation of the mouth area with respect to the nose area, each stripe is partitiones into three parts *lower* (L), *upper-left* (UL) and *upper-right* (UR) with respect to the coordinates of the nose tip.
- Iso-geodesic stripes and 3DWW computed between pairs of stripes (*inter-stripe* 3DWW) and between each stripe and itself (*intra-stripe* 3DWW), are collected into a a graph representation where stripes are used to label the graph nodes and 3DWWs to label the $N(N+1)/2$ resulting graph edges.
- The similarity between two 3D faces reduces to matching their corresponding graphs. Actually, since there is an unambiguous ordering of stripes from the nose tip to the border of the face model, the graph matching problem is reduced to the computation of a distance between multidimensional vectors, by comparing homologous stripe pairs.

3DWW matching



Matching of two 3D face graph models for the *upper-left* part of the face. Matching distances between pairs of stripes in the two models are reported.

For homologous pairs of stripes in two faces, a distance D that accounts for how much the spatial distributions between points of the two stripes differ from each other, can be defined directly from the values of the differences between corresponding directional indexes:

$$\begin{aligned}\mathcal{D}(w,w') = \; & \lambda_H|w_H - w'_H| + \lambda_V|w_V - w'_V| + \lambda_D|w_D - w'_D| \\ & + \lambda_{XY}|w_{XY} - w'_{XY}| + \lambda_{XZ}|w_{XZ} - w'_{XZ}| \\ & + \lambda_{YZ}|w_{YZ} - w'_{YZ}| + \lambda_{H_0}|w_{H_0} - w'_{H_0}| \\ & + \lambda_{V_0}|w_{V_0} - w'_{V_0}| + \lambda_{D_0}|w_{D_0} - w'_{D_0}| \\ & + \lambda_{HV_0}|w_{HV_0} - w'_{HV_0}| + \lambda_{HD_0}|w_{HD_0} - w'_{HD_0}| \\ & + \lambda_{VD_0}|w_{VD_0} - w'_{VD_0}| \,.\end{aligned}$$

being λ_H, λ_V, λ_D, λ_{XY}, λ_{XZ}, λ_{YZ}, λ_{H_0}, λ_{V_0}, λ_{D_0}, λ_{HV_0}, λ_{HD_0}, λ_{VD_0}, a combination of non-negative numbers with sum equal to 1. Values of λ-parameters have been set so as to give the maximum relevance to directional indexes that encode Z-displacements (w_D, w_{YZ}), and minimum relevance to the alignment indexes.

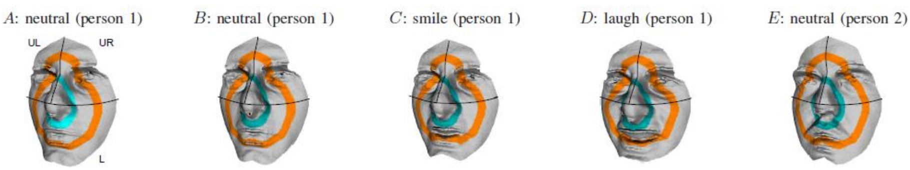
3DWW matching

- The measure of the similarity between two face models represented through the graphs P and G with nodes p_k and g_k is derived as :

$$\mu(P,G) = \frac{\alpha}{N_P} \cdot \sum_{k=1}^{N_P} \mathcal{D}(w(p_k,p_k), w(g_k,g_k)) + \frac{2(1-\alpha)}{N_P(N_P-1)} \cdot \sum_{k=1}^{N_P} \sum_{h=1}^{k-1} \mathcal{D}(w(p_k,p_h), w(g_k,g_h)),$$

- Differences between faces of two individuals are measured by computing the 3DWW for each pair of stripes, separately in the three face parts for each face, and then comparing the 3DWWs of homologous pairs of the two faces. The final dissimilarity measure is obtained by averaging distances in the three parts.

3DWW matching



UL	B	C	D	E
A	0.019	0.031	0.039	0.193
B	-	0.016	0.054	0.207
C	-	-	0.049	0.216
D	-	-	-	0.172

UR	B	C	D	E
A	0.026	0.040	0.052	0.208
B	-	0.031	0.047	0.222
C	-	-	0.024	0.234
D	-	-	-	0.231

L	B	C	D	E
A	0.022	0.023	0.037	0.160
B	-	0.043	0.058	0.147
C	-	-	0.015	0.182
D	-	-	-	0.193

- Face scans and distances between stripes 4 and 7 (for *UL*, *UR* and *L* parts): scans *A*, *B*, *C* and *D* show the same person under different facial expressions (from neutral to laugh); scan *E* shows a different person with neutral expression. It can be noticed that different expressions of the same person do not alter significantly the values of the distances. Larger differences are found instead between different persons.

- In particular, stripes 4 and 7 are considered in that stripe 4 typically covers a part of the face that remains almost unchanged under expression changes, and stripe 7 instead includes points that typically change their position with facial expressions.

Some references

- R. Polikar. The Wavelet Tutorial - Part I. http://users.rowan.edu/~polikar/WAVELETS/WTpart1.html (http://users.rowan.edu/~polikar/WAVELETS/WTtutorial.html)
- Wiskott, L., Fellous, J. M., Kuiger, N., & Von Der Malsburg, C. (1997). Face recognition by elastic bunch graph matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *19*(7), 775-779.
- Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *24*(7), 971-987.
- Ahonen, T., Hadid, A., & Pietikainen, M. (2006). Face description with local binary patterns: Application to face recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *28*(12), 2037-2041.
- R. Wolfe. Teaching Texture Mapping Visually. https://www.siggraph.org/education/materials/HyperGraph/mapping/r_wolfe_mapping.pdf
- Marton, Z. C., Rusu, R. B., & Beetz, M. (2009, May). On fast surface reconstruction methods for large and noisy point clouds. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on* (pp. 3218-3223). IEEE.
- R. Schettini, A. Colombo, C. Cusano. Multimodal face recognition. http://www.ivl.disco.unimib.it/Seminari/IBM-volti.pdf
- Atick, J. J., Griffin, P. A., & Redlich, A. N. (1996). Statistical approach to shape from shading: Reconstruction of three-dimensional face surfaces from single two-dimensional images. *Neural Computation*, *8*(6), 1321-1340.

Some references

- V. Blanz and T. Vetter. A Morphable Model for the Synthesis of 3D Faces SIGGRAPH'99 Conference Proceedings (pp 187-194). http://gravis.cs.unibas.ch/Sigg99.html
- Kemelmacher-Shlizerman, I., & Basri, R. (2011). 3d face reconstruction from a single image using a single reference face shape. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *33*(2), 394-405.
- Toderici G., G. Evangelopoulos, T. Fang, T. Theoharis and I. Kakadiaris, UHDB11 Database for 3D-2D Face Recognition, 6th Pacific Rim Symposium on Image and Video Technology (PSIVT), October 2013, Guanajuato, Mexico.
- Blanz, V., & Vetter, T. (2003). Face recognition based on fitting a 3D morphable model. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *25*(9), 1063-1074.
- X. Lu, D. Colbry, A.K. Jain, “Three-Dimensional Model Based Face Recognition, *In Proc. International Conference on Pattern Recognition*,
- 2004.
- X. Lu, A.K. Jain, D. Colbry, “Matching 2.5D Face Scans to 3D Models”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.
- 28, no. 1, pp. 31-43, 2006.
- S. Berretti, A. Del Bimbo, P. Pala. "3D Face Recognition using iso-Geodesic Stripes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.32, no.12, pp.2162-2177, December 2010.