



Animazione Digitale

lezione 7

N. Alberto Borghese

- Animazione parametrica solida.
- Morphing
- Cloth animation

N.B.: Il diritto di scaricare questo file è riservato solamente agli studenti regolarmente iscritti al corso di Animazione Digitale.



Oggetti deformabili

Flessibilità e deformazione.

Morphing (da μορφη – forma), acronimo per metamorphosis.

Morphing dinamico – elementi finiti – calcolo strutturale.

Morphing cinematico – **Key shapes**.

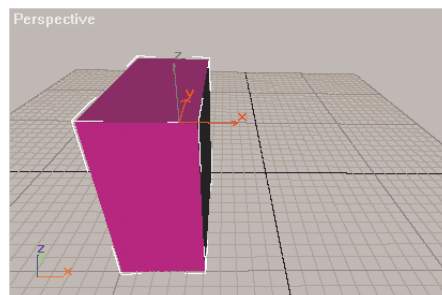
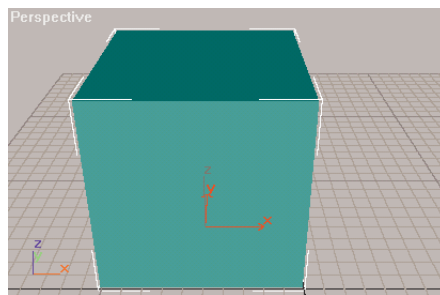
Se la connettività del modello viene mantenuta -> interpolazione dei vertici.



Stessa connettività



Morphing mediante matrici Affini



$$\begin{bmatrix} S_x X_p \\ S_y Y_p \\ S_z Z_p \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_p \\ Y_p \\ Z_p \\ 1 \end{bmatrix}$$

$$S_x(t), S_y(t), S_z(t)$$

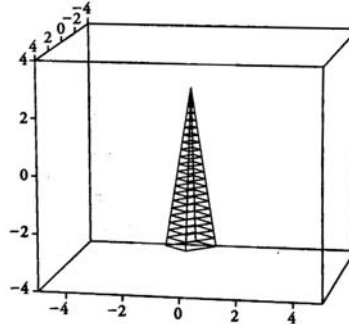
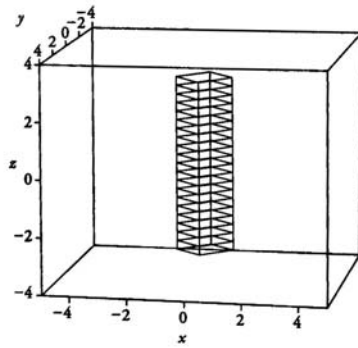
Con le matrici affini, si possono modellare alcuni effetti semplici (e.g. schiacciamento).



Deformazioni non-lineari



$$P' = M(P) \cdot P$$



$$s(z) = \frac{(maxz - z)}{(maxz - minz)}$$

$$x' = s(z) \cdot x$$

$$y' = s(z) \cdot y$$

$$z' = z$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} s(z) & 0 & 0 \\ 0 & s(z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

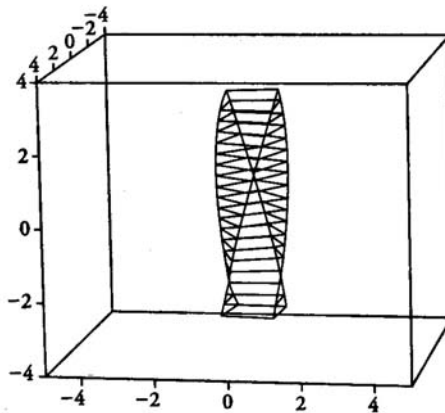
$$P' = M(P) \cdot P$$

Original object

Tapered object



Twist

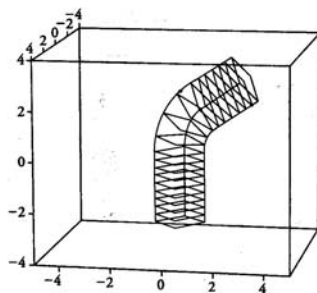


k = twist factor

$$x' = x \cdot \cos(k \cdot z) - y \cdot \sin(k \cdot z)$$

$$y' = x \cdot \sin(k \cdot z) + y \cdot \cos(k \cdot z)$$

$$z' = z$$



(z_{\min}, z_{\max}) —bend region
 (y_0, z_{\min}) —center of bend

$$x' = x$$

$$y' = \begin{cases} y & z < z_{\min} \\ y_0 - (R \cdot C_\theta) & z_{\min} \leq z \leq z_{\max} \\ y_0 - (R \cdot C_\theta) + (z - z_{\max}) \cdot S_\theta & z > z_{\max} \end{cases}$$

$$z' = \begin{cases} z & z < z_{\min} \\ z_{\min} + (R \cdot S_\theta) & z_{\min} \leq z \leq z_{\max} \\ z_{\min} + (R \cdot S_\theta) + (z - z_{\max}) \cdot C_\theta & z > z_{\max} \end{cases}$$

Bend

$$\theta = \begin{cases} z - z_{\min} & z < z_{\max} \\ z_{\max} - z_{\min} & \text{otherwise} \end{cases}$$

$$C_\theta = \cos \theta$$

$$S_\theta = \sin \theta$$

$$R = y_0 - y$$

$$\begin{cases} z < z_{\min} \\ z_{\min} \leq z \leq z_{\max} \\ z > z_{\max} \end{cases}$$

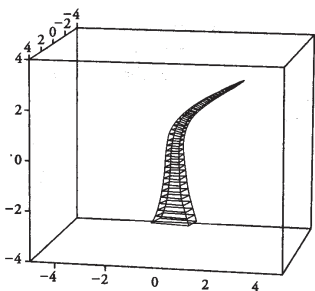
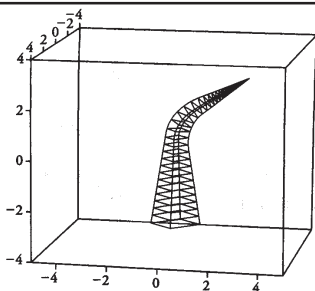
$$\begin{cases} z < z_{\min} \\ z_{\min} \leq z \leq z_{\max} \\ z > z_{\max} \end{cases}$$

Laboratory of

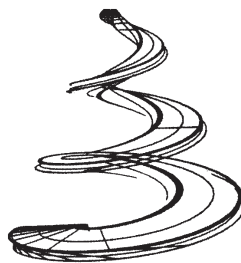
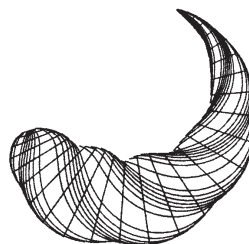
homes.dsi.unimi.it/~borghese/



Deformazioni complesse



Compound global deformations



Examples from Barr [2]

Laboratory of Motion Analysis & Virtual Reality, MAVR

<http://homes.dsi.unimi.it/~borghese/>





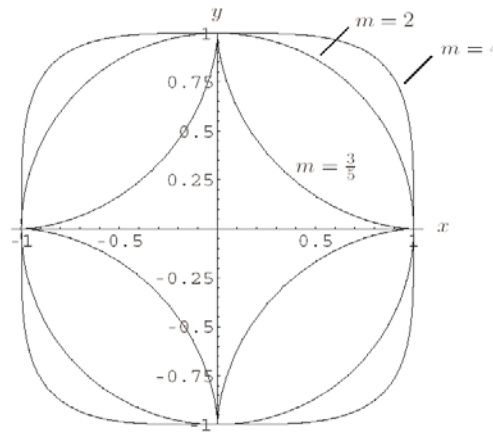
I superellissi



I superellissi sono definiti dalle equazioni (curve di Lamè, XIX° secolo):

$$\left(\frac{x}{a}\right)^m + \left(\frac{y}{b}\right)^m = 1$$

m parametro di forma
 $= 2/\epsilon \quad \epsilon > 0$.



I super-egg (superellissoidi)

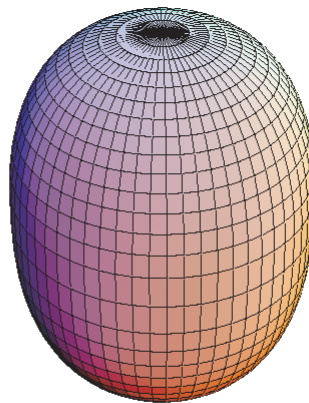


Figure 2.2. A “superegg” (superquadric with $m = 2.5$, height-width ratio = stable in the upright position because the center of gravity is lower than the of curvature (Gardner, 1965).



Costruzione di una sfera



A 3D surface can be obtained by the spherical product of two 2D curves (Barr, 1981). A unit sphere, for example, is produced when a half circle in a plane orthogonal to the (x, y) plane (Fig. 2.4)

$$\mathbf{m}(\eta) = \begin{bmatrix} \cos \eta \\ \sin \eta \end{bmatrix}, \quad -\pi/2 \leq \eta \leq \pi/2 \quad (2.5)$$

is crossed with the full circle in (x, y) plane

$$\mathbf{h}(\omega) = \begin{bmatrix} \cos \omega \\ \sin \omega \end{bmatrix}, \quad -\pi \leq \omega < \pi, \quad (2.6)$$

$$\mathbf{r}(\eta, \omega) = \mathbf{m}(\eta) \otimes \mathbf{h}(\omega) = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos \eta \cos \omega \\ \cos \eta \sin \omega \\ \sin \eta \end{bmatrix}, \quad \begin{matrix} -\pi/2 \leq \eta \leq \pi/2 \\ -\pi \leq \omega < \pi \end{matrix} \quad (2.7)$$

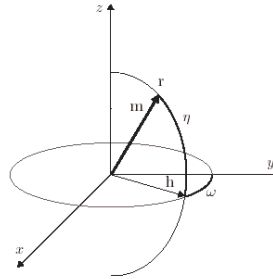


Figure 2.4. A 3D vector \mathbf{r} , which defines a closed 3D surface, can be obtained by a spherical product of two 2D curves.



Matematica dei super-egg



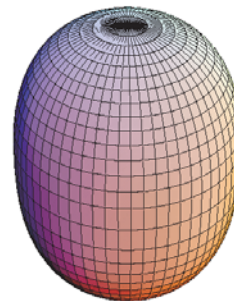
$$\left(\frac{x}{a}\right)^{2/\varepsilon} + \left(\frac{y}{b}\right)^{2/\varepsilon} = 1 \quad \mathbf{s}_2(\vartheta) = \begin{bmatrix} a_1 \cos^\varepsilon \vartheta \\ a_2 \sin^\varepsilon \vartheta \end{bmatrix} \quad -\pi \leq \vartheta \leq \pi$$

$$\cos^\varepsilon \vartheta = \text{sign}(\cos \vartheta) |\cos \vartheta|^\varepsilon$$

$$\mathbf{p}(\eta, \vartheta) = \mathbf{s}_1(\eta) \otimes \mathbf{s}_2(\vartheta) =$$

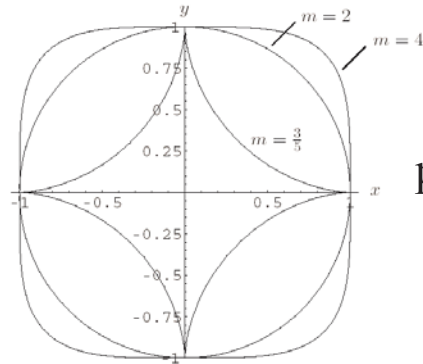
$$\begin{bmatrix} \cos^{\varepsilon_1} \eta \\ a_3 \sin^{\varepsilon_1} \eta \end{bmatrix} \otimes \begin{bmatrix} a_1 \cos^{\varepsilon_2} \vartheta \\ a_2 \sin^{\varepsilon_2} \vartheta \end{bmatrix} =$$

$$\begin{bmatrix} a_1 \cos^{\varepsilon_1} \eta \cos^{\varepsilon_2} \vartheta \\ a_2 \cos^{\varepsilon_1} \eta \sin^{\varepsilon_2} \vartheta \\ a_3 \sin^{\varepsilon_1} \eta \end{bmatrix}$$





I parametri dei super-egg



$$p(\eta, \vartheta) = \begin{bmatrix} a_1 \cos^{\varepsilon_1} \eta \cos^{\varepsilon_2} \theta \\ a_2 \cos^{\varepsilon_1} \eta \sin^{\varepsilon_2} \theta \\ a_3 \sin^{\varepsilon_1} \eta \end{bmatrix}$$

a_1, a_2, a_3 : fattori di scala lungo i 3 assi.

ε_1 : determina la forma del superellissoide su un piano perpendicolare al piano xy.

ε_2 : determina la forma del superellissoide sul piano xy.



Superellissoidi

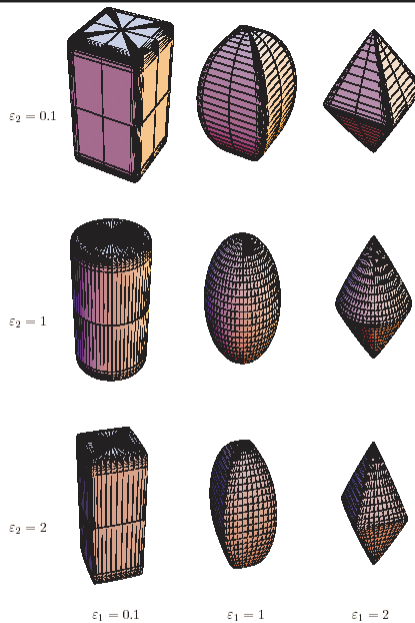


Figure 2.5. Superellipsoids with different values of exponents ε_1 and ε_2 . Size parameters a_1, a_2, a_3 are kept constant. Superquadric-centered coordinate axis z points upwards!



Equazioni parametriche



$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_1 \cos^{\varepsilon_1} \eta \cos^{\varepsilon_2} \theta \\ a_2 \cos^{\varepsilon_1} \eta \sin^{\varepsilon_2} \theta \\ a_3 \sin^{\varepsilon_1} \eta \end{bmatrix} \quad \begin{bmatrix} (x/a_1)^2 \\ (y/a_2)^2 \\ (z/a_3)^2 \end{bmatrix} = \begin{bmatrix} \cos^{2\varepsilon_1} \eta \cos^{2\varepsilon_2} \theta \\ \cos^{2\varepsilon_1} \eta \sin^{2\varepsilon_2} \theta \\ \sin^{2\varepsilon_1} \eta \end{bmatrix}$$

$$\left(\frac{x}{a_1}\right)^{2/\varepsilon_2} + \left(\frac{y}{a_2}\right)^{2/\varepsilon_2} = \cos^{2\varepsilon_1/\varepsilon_2}(\eta)$$

$$\left[\left(\frac{x}{a_1}\right)^{2/\varepsilon_2} + \left(\frac{y}{a_2}\right)^{2/\varepsilon_2}\right]^{\varepsilon_2/\varepsilon_1} + \left(\frac{z}{a_3}\right)^{2/\varepsilon_1} = 1$$

Inside / outside function



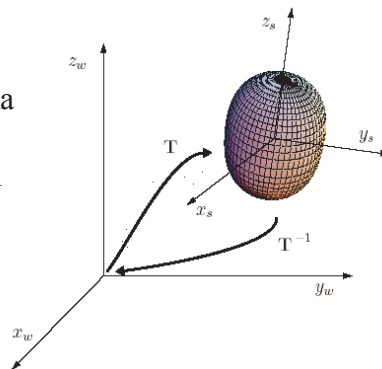
Le superquadriche (Alan Barr, 1981)



Generalizzazione delle quadriche: ellissoidi, iperbolodi e tori.

Modelli parametrici: 3 parametri di dimensioni lungo i 3 assi coordinati + 2 parametri di forma (esponenti).

6 parametri aggiuntivi descrivono la posizione/orientamento della superquadrica rispetto al sistema di riferimento assoluto.

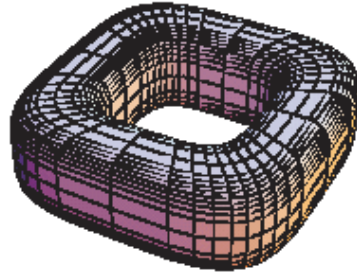




I supertoroidi



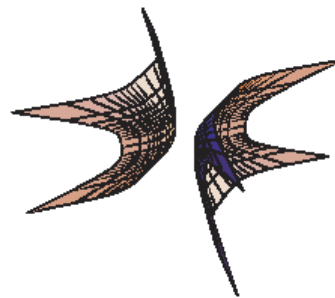
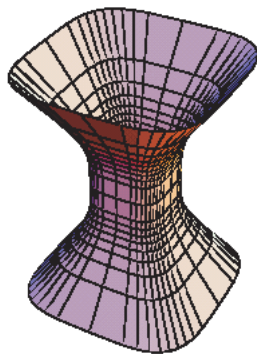
Generalizzazione dei toroidi



$$(r - a)^2 = \left(\frac{z}{a_3}\right)^2 = 1 \quad r = \sqrt{\left(\frac{x}{a_1}\right)^2 + \left(\frac{y}{a_2}\right)^2}$$



Superiperboloidi



$$\left(\frac{x}{a_1}\right)^2 + \left(\frac{y}{a_2}\right)^2 - \left(\frac{z}{a_3}\right)^2 = 1$$

$$\left(\frac{x}{a_1}\right)^2 - \left(\frac{y}{a_2}\right)^2 - \left(\frac{z}{a_3}\right)^2 = 1$$



Deformazione parametrica



I parametri definiscono la forma.

Deformazione tra modelli con la stessa connettività.

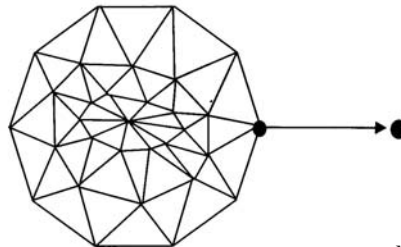
Trasformazioni affini.

Deformazioni non-lineari

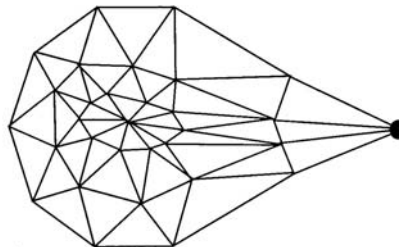
Superquadriche.



Free-form deformation



Displacement of seed vertex



Metriche:

- Distanza Euclidea.
- Distanza sulla superficie.
- Numero di vertici.



Dalla deformazione all'animazione



$$\{P(t)\}$$

Definire l'andamento temporale della posizione del vertice deformato: $P(t) \Rightarrow$ curva di animazione.

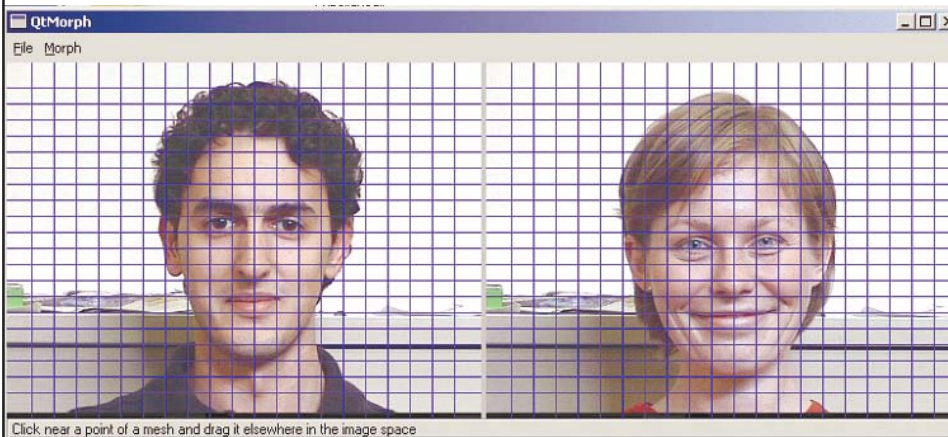
Definire il mix tra due posizioni estreme del vertice: $P1$ e $P2$:
Morphing. $P(t) = u(t)*P1 + (1-u(t))*P2$. Curva di animazione per il parametro $u(t)$.



Grid deformation



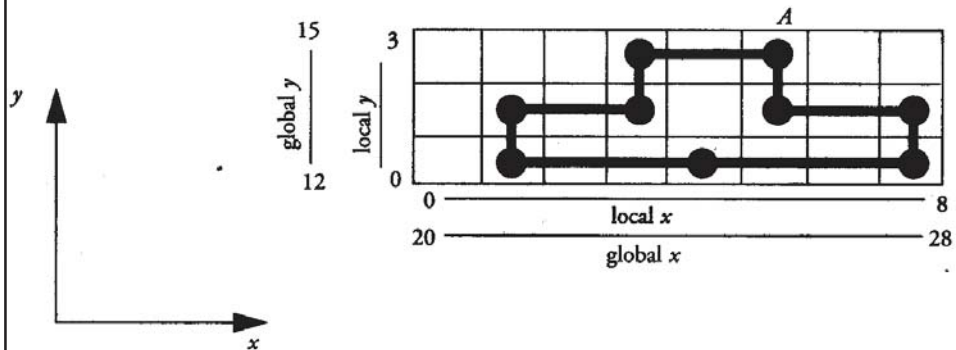
Primo utilizzo: Hunger, 1974.



<http://www.colorado-research.com/~gourlay/software/Graphics/Xmorph/>



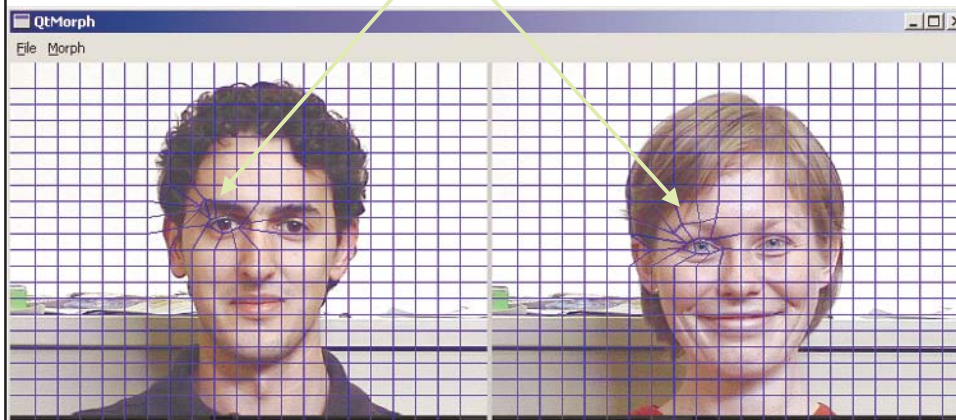
Morphing mediante deformazione di griglia



Offset + scala



Adattamento della griglia



- Identificazione dei punti corrispondenti.
- Definizione del colore dei pixel delle immagini intermedie.



Trasformazione tra 2 griglie



Parametro di morphing: $0 \leq m(t) \leq 1$

$$P(t) = m(t) * P_1 + (1 - m(t)) * P_2$$

Calcolo la posizione dei vertici (e della griglia) per ogni immagine intermedia.



Morphing (I)



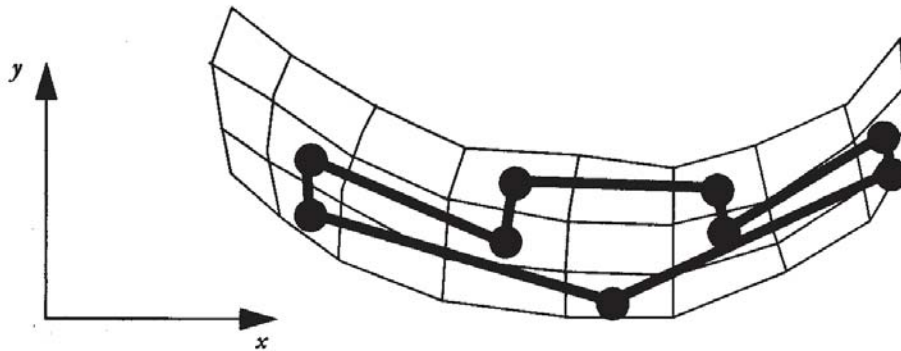
Consideriamo un'immagine intermedia: k.

Per ogni pixel:

- 1) Determiniamo a quale maglia della griglia appartiene.
- 2) Calcoliamo le coordinate implicite del pixel (u,v).
- 3) Riportiamo le coordinate implicite nelle 2 immagini di partenza ed identifichiamo i corrispondenti pixel.
- 4) Determiniamo il colore corrispondente dei pixel.
- 5) Calcoliamo la media pesata dei pixel, che verrà assegnata al pixel dell'immagine intermedia (cross-dissolving).



1) Determinazione della maglia di appartenenza



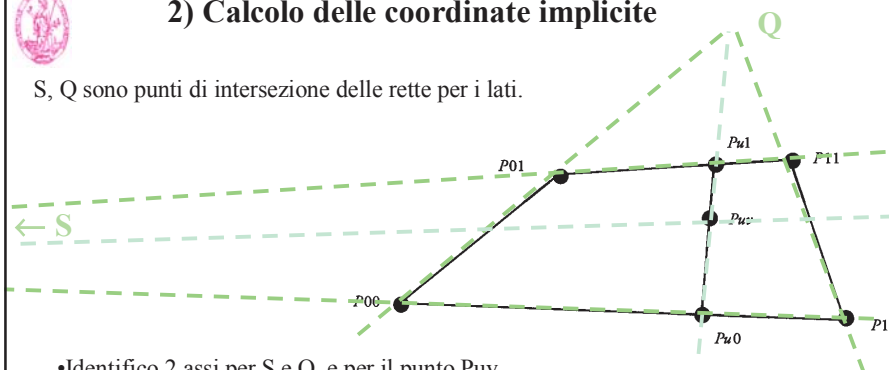
Funzione `in_polygon()` [utilizza prodotti vettoriali].



2) Calcolo delle coordinate implicite



S, Q sono punti di intersezione delle rette per i lati.



- Identifico 2 assi per S e Q, e per il punto Puv.
- Calcolo le coordinate parametrizzate di Puv come:

$$u = |P_{u0} - P_{00}| / |P_{10} - P_{00}|$$

$$v = |P_{uv} - P_{u0}| / |P_{u1} - P_{u0}|$$
- Dati 4 vertici di un quadrilatero (quelle di partenza), è sempre possibile ricavare il punto Puv corrispondente (interpolazione bilineare):

$$P_{u0} = (1-u) * P_{00} + u * P_{10}$$

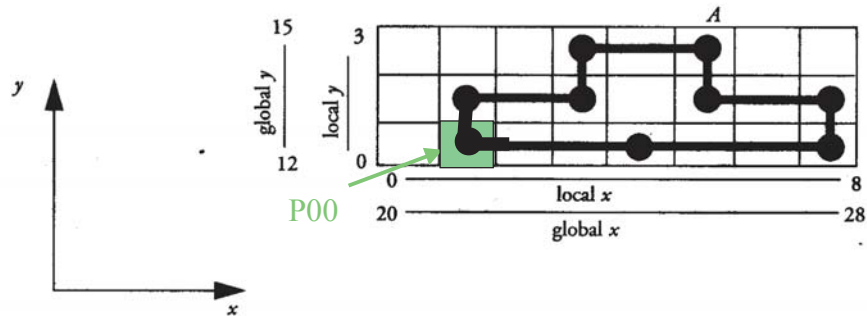
$$P_{u1} = (1-u) * P_{01} + u * P_{11}$$

$$P_{uv} = (1-v) * P_{u0} + v * P_{u1} = (1-u) * (1-v) * P_{00} + (1-u) * v * P_{01} + u * (1-v) * P_{10} + u * v * P_{11}$$

Ho quindi associato ad ogni pixel dell'immagine k, un punto delle due immagini di partenza.



2bis) Calcolo delle coordinate implicite: esempio



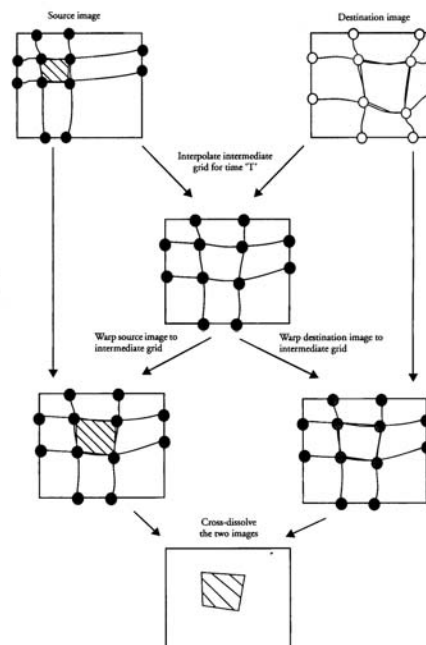
$$\begin{aligned}
 P &= 0.4 \cdot 0.3 \cdot [1 \ 1] + 0.4 \cdot 0.7 \cdot [1 \ 2] + 0.6 \cdot 0.3 \cdot [2 \ 1] + 0.6 \cdot 0.7 \cdot [2 \ 2] = \\
 &= 0.12 \cdot [1 \ 1] + 0.28 \cdot [1 \ 2] + 0.18 \cdot [2 \ 1] + 0.42 \cdot [2 \ 2] = \\
 &= [0.12 \ 0.12] + [0.28 \ 0.56] + [0.36 \ 0.18] + [0.84 \ 0.84] = [1.60 \ 1.70]
 \end{aligned}$$

NB La procedura funziona solamente per quadrilateri convessi!



3) Identificazione del punto sulle 2 immagini di partenza

4) Identificazione del colore associato ai 2 punti





5) Calcolo del colore dell'immagine intermedia



$$[r \ g \ b]_1 \Rightarrow u$$

$$[r \ g \ b]_2 \Rightarrow (1-u)$$

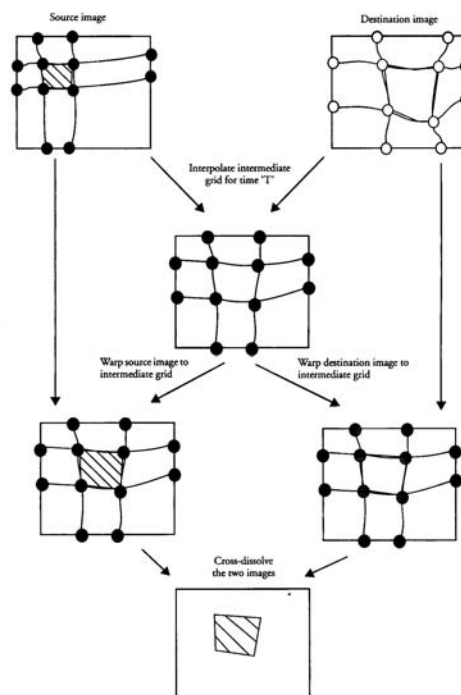
Cross-dissolving le due immagini di partenza:

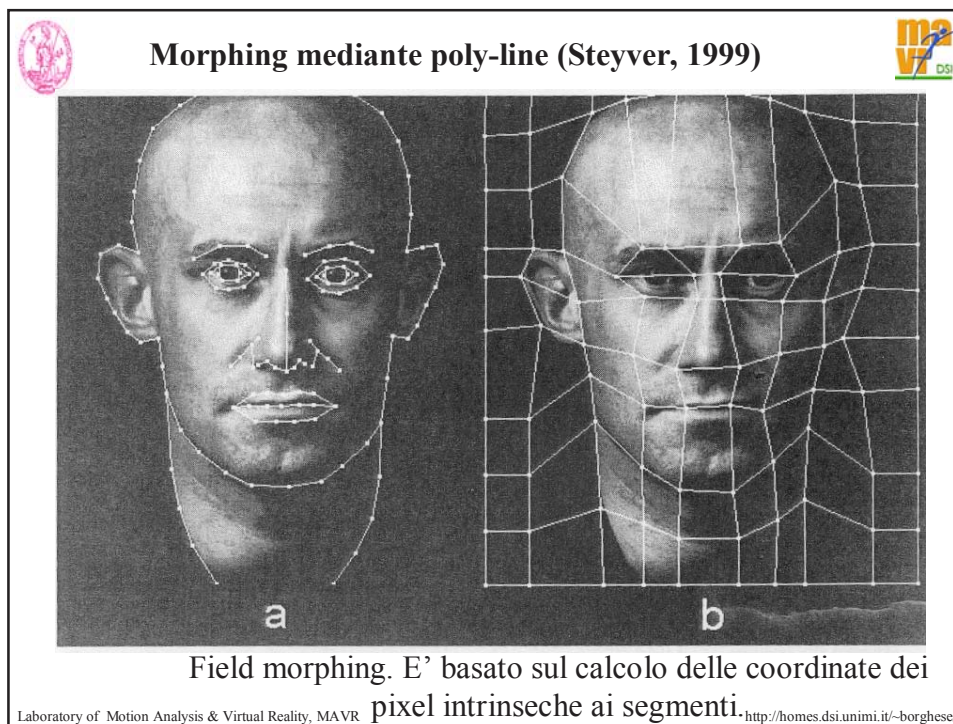
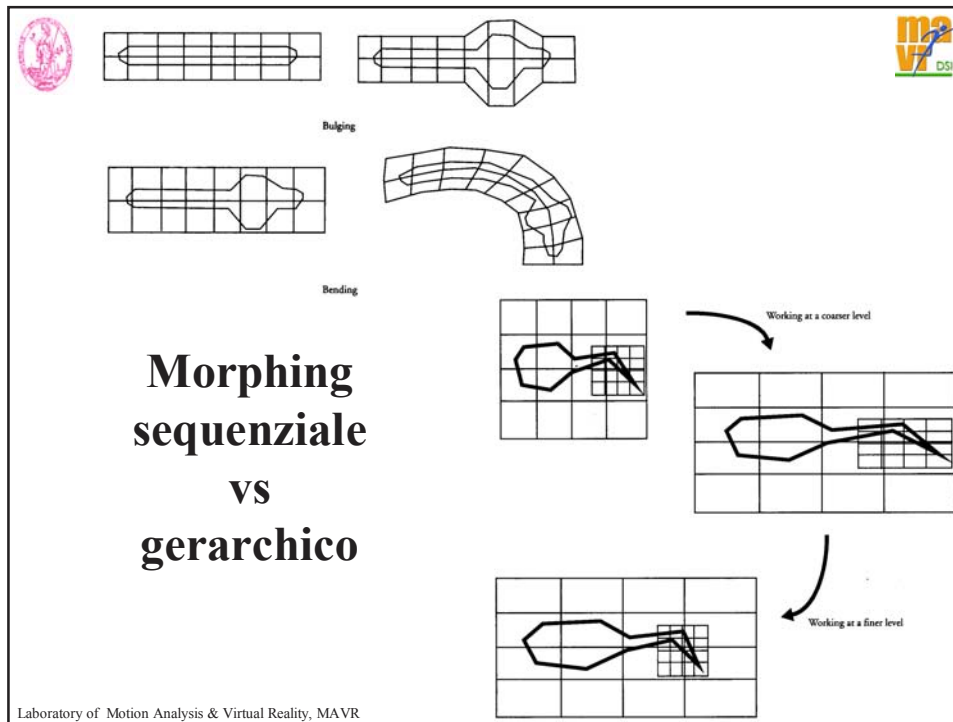
$$[r \ g \ b]_k \Rightarrow u*[r \ g \ b]_1 + (1-u)*[r \ g \ b]_2$$

Si possono utilizzare spazi di colore diversi da RGB.



Creazione di un'immagine intermedia







Morphing 3D



Non esistono soluzioni definitive: morphing di superficie o di volume.

E' relativamente semplice morphare *superfici omomorfe* (che hanno la stessa topologia).

Morphing volumetrico può ridursi ad una cross-dissolvenza in 3D.

In generale, similmente al 2D, il morphing 3D richiede:

- 1) Identificare elementi corrispondenti sulle 2 immagini.
- 2) Interpolare tra gli elementi (in 3D).



Morphing



Consideriamo un'immagine intermedia: k.

Per ogni pixel:

- 1) Determiniamo a quale maglia della griglia appartiene.
- 2) Calcoliamo le coordinate implicite del pixel (u,v).
- 3) Riportiamo le coordinate implicite nelle 2 immagini di partenza ed identifichiamo i corrispondenti pixel.
- 4) Determiniamo il colore corrispondente dei pixel.
- 5) Calcoliamo la media pesata dei pixel, che verrà assegnata al pixel dell'immagine intermedia (cross-dissolving).

Si crea quindi un campo di morphing.



I vestiti (cloth)



Parte integrante nella resa realistica del movimento (e.g. una ragazza con la gonna che cammina).

Ma anche del movimento di oggetti: un fazzoletto che esce dal taschino, una tovaglia che cade su una tavola.



Laboratory of Motion Analysis & Virtual Reality, MAVR

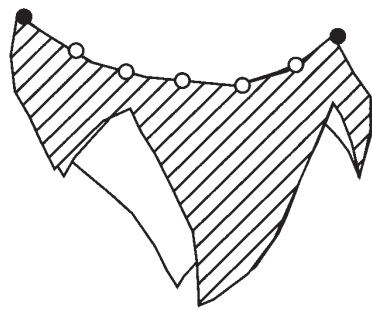
<http://homes.dsi.unimi.it/~borghese/>



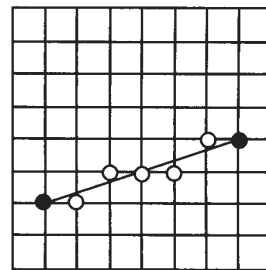
Draping



1. Identifica le regioni di contatto tra tessuto e supporto.
2. Dispone la regione compresa tra i supporti secondo una curva caratteristica (catenaria)
3. Mediante una procedura di relaxation deforma il tessuto circostante.



Cloth supported at two constrained points



Constrained points in grid space

Deformazione locale

$$y = c - \left(a * \cosh \left(\frac{x - 1/2}{a} \right) \right)$$

Laboratory of Motion Analysis & Virtual Reality, MAVR

<http://homes.dsi.unimi.it/~borghese/>



Catenaria tra due punti

$$y_1 = c + a \cdot \cosh\left[\frac{(x_1 - b)}{a}\right]$$

$$y_2 = c + a \cdot \cosh\left[\frac{(x_2 - b)}{a}\right]$$

$$L = a \cdot \sinh\left[\frac{(x_2 - b)}{a}\right] - a \cdot \sinh\left[\frac{(x_1 - b)}{a}\right]$$

$$\sqrt{L^2 - (y_2 - y_1)^2} = 2 \cdot a \cdot \sinh\left[\frac{(x_2 - x_1)}{2 \cdot a}\right] \quad (a \text{ can be solved for numerically at this point})$$

$$M = \sinh\left(\frac{x_2}{a}\right) - \sinh\left(\frac{x_1}{a}\right)$$

$$N = \cosh\left(\frac{x_2}{a}\right) - \cosh\left(\frac{x_1}{a}\right)$$

$$\text{if } N > M \quad \mu = \tanh^{-1}\left(\frac{M}{N}\right)$$

$$Q = \frac{M}{\sinh(\mu)} = \frac{N}{\cosh(\mu)}$$

$$b = a \cdot \left[\mu - \sinh^{-1}\left(\frac{L}{Q \cdot a}\right) \right]$$

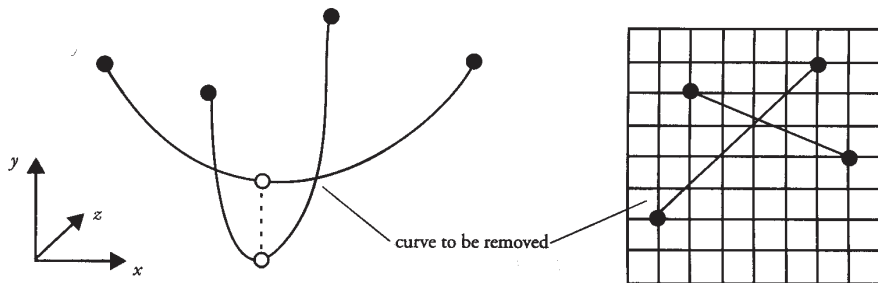
$$\text{if } M > N \quad \mu = \tanh^{-1}\left(\frac{N}{M}\right)$$

$$Q = \frac{N}{\sinh(\mu)} = \frac{M}{\cosh(\mu)}$$

$$b = a \cdot \left[\mu - \cosh^{-1}\left(\frac{L}{Q \cdot a}\right) \right]$$



Più linee di ancoraggio





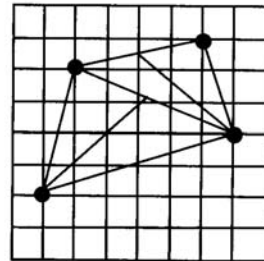
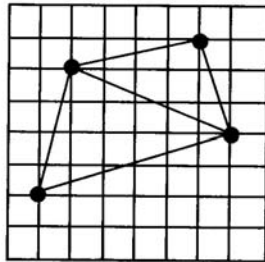
Interpolazione tra punti di ancoraggio



- Connessione dei punti mediante catenarie.
- Vengono mantenute le catenarie più alte.

Analisi dei triangoli interni.

- Connessione dei vertici con il punto medio del lato opposto mediante catenarie.
- Vengono mantenute le catenarie più alte.



Problemi



Animazione mediante inserimento/rimozione di punti di contatto.

I punti di contatto sono in realtà regioni.

Problemi di compressione/stiramento del tessuto.

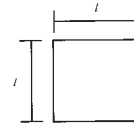
Pieghe sopra e attorno alla superficie di contatto.



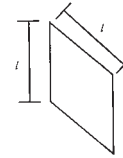
Clothing



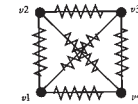
- Modellazione mediante quadrilateri (trama + ordito)
- Definizione di proprietà dinamiche del quadrilatero (tensore sforzi/deformazioni) + risoluzione agli elementi finiti.
- Definizione di forze di attrazione tra i vertici che modellizzano: stiramento, curvatura, deformazione.



(a) Original quadrilateral of mesh



(b) Skew of original quadrilateral without changing the length of edges



(c) Diagonal springs to control skew

Risoluzione di un sistema di equazioni dinamiche soggetto a forze esterne.



Cloth animation



- Definizione delle collisioni puntuali e delle risposte del materiale.
- Propagazione delle forze di reazione al materiale.
- Draping. Definizione iterativa della posizione del tessuto quando appoggiato su supporti.
- Cloth dynamics.
 - Definizione delle proprietà locali del materiale.
 - Soluzione agli elementi finiti o con un sistema di equazioni dinamiche con input esterni.