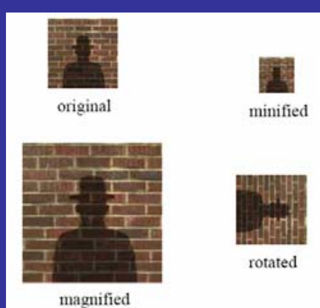
Image Warping

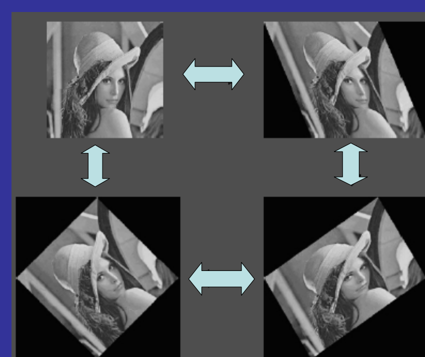
Image Warping

- Warping: The mapping from signal $F(\mathbf{x})$ to signal $G(\boldsymbol{\zeta})$ by some transformation $\boldsymbol{\zeta}=m(\mathbf{x})$ satisfying $G(m(\mathbf{x})) = F(\mathbf{x})$.
- Image warping:
  - $\mathbf{x}=(x,y)$, $\boldsymbol{\zeta}=(\zeta, \eta)$, $x, y, \zeta, \eta \in \mathrm{I}$
  - $F(\mathbf{x})$, $G(\boldsymbol{\zeta}) \in [0, 255]$
  - m():
    - Linear operation: $\boldsymbol{\zeta}=M\mathbf{x}$
    - Piecewise linear operations: $\boldsymbol{\zeta}=M(\mathbf{x})\,\mathbf{x}$
    - RBF approximation: $\zeta = \sum_{i=1}^{N} a_i g_i(\|x - x_i\|)$ [4]

Warping Example



Warping Example



Warping Example

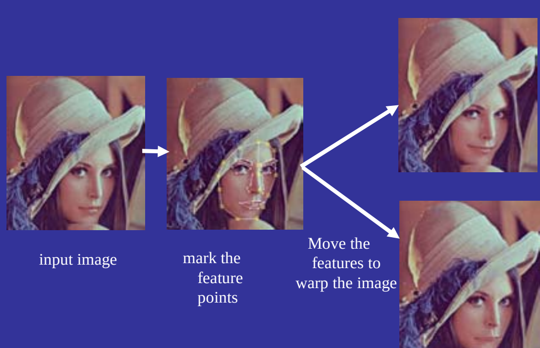


Image warping

- Two warping procedures:
  - For each $\mathbf{x}$, $\boldsymbol{\zeta}=M\mathbf{x}$, $F(\mathbf{x}) = G(\boldsymbol{\zeta})$
  - For each $\boldsymbol{\zeta}$, $\mathbf{x}=M^{-1}\boldsymbol{\zeta}$, $F(\mathbf{x}) = G(\boldsymbol{\zeta})$. ***This one is bad, because x can be out of the target image region.***

Affine Warping – definition

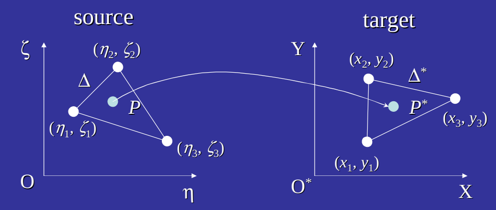
•Assume the mapping from (x, y) to (ζ , η) is linear:

$\zeta = a_1x + a_2y + a_3$
$\eta = b_1x + b_2y + b_3$

$$\begin{bmatrix}\zeta\\ \eta\\ 1\end{bmatrix} = \begin{bmatrix}a_1 & a_2 & a_3\\ b_1 & b_2 & b_3\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x\\ y\\ 1\end{bmatrix}$$

6 Unknowns (a, b) need to be solved by 6 equations, e.g. need 3 pair of correspondences

Affine Warping



Correspondences: $(\eta_1, \zeta_1) \to (x_1, y_1)$,
$(\eta_2, \zeta_2) \to (x_2, y_2)$,
$(\eta_3, \zeta_3) \to (x_3, y_3)$,
$\Delta \to \Delta^*$

Affine Warping – better representation

For any (ζ, η) in the triangle, we have

$$\zeta = \lambda_1\zeta_1 + \lambda_2\zeta_2 + \lambda_3\zeta_3$$
$$\eta = \lambda_1\eta_1 + \lambda_2\eta_2 + \lambda_3\eta_3 \qquad (2)$$
$$1 = \lambda_1 + \lambda_2 + \lambda_3 \qquad (\lambda_i \geq 0,\ i = 1, 2, 3)$$

or

$$\begin{bmatrix}\zeta\\ \eta\\ 1\end{bmatrix} = \lambda_1\begin{bmatrix}\zeta_1\\ \eta_1\\ 1\end{bmatrix} + \lambda_2\begin{bmatrix}\zeta_2\\ \eta_2\\ 1\end{bmatrix} + \lambda_3\begin{bmatrix}\zeta_3\\ \eta_3\\ 1\end{bmatrix} = \begin{bmatrix}\zeta_1 & \zeta_2 & \zeta_3\\ \eta_1 & \eta_2 & \eta_3\\ 1 & 1 & 1\end{bmatrix}\begin{bmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\end{bmatrix}$$

Affine Warping – better representation

• This representation is invariant to affine transformation, thus P's corresponding point $P^*(x, y)$ can be represented as

$$x = \lambda_1x_1 + \lambda_2x_2 + \lambda_3x_3$$
$$y = \lambda_1y_1 + \lambda_2y_2 + \lambda_3y_3. \qquad (3)$$

with the same λ_i

$$\begin{bmatrix}\zeta\\ \eta\\ 1\end{bmatrix} = \begin{bmatrix}\zeta_1 & \zeta_2 & \zeta_3\\ \eta_1 & \eta_2 & \eta_3\\ 1 & 1 & 1\end{bmatrix}\begin{bmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\end{bmatrix} = M\begin{bmatrix}x_1 & x_2 & x_3\\ y_1 & y_2 & y_3\\ 1 & 1 & 1\end{bmatrix}\begin{bmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\end{bmatrix} = M\begin{bmatrix}x\\ y\\ 1\end{bmatrix}$$

Affine Warping

Affine Warping in Matrix format

Let $\mathbf{V} = (\zeta, \eta, 1)^T$; $\mathbf{V}' = (x, y, 1)^T$;

$\mathbf{A} = [\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3]$; $\mathbf{B} = [\mathbf{V}_1', \mathbf{V}_2', \mathbf{V}_3']$; $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3)^T$;

We then have

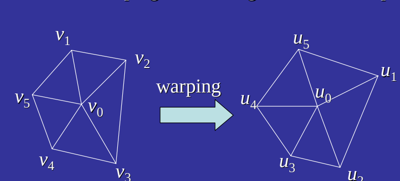
$\mathbf{A}\boldsymbol{\lambda} = \mathbf{V}$; $\mathbf{B}\boldsymbol{\lambda} = \mathbf{V}'$;

Thus the affine warping is:

$$\mathbf{V} = \mathbf{M}\mathbf{V}' = \mathbf{A}\mathbf{B}^{-1}\mathbf{V}';$$

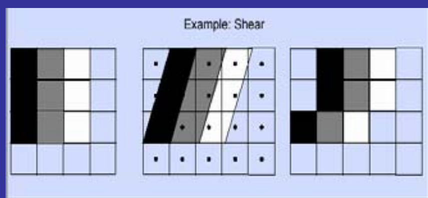
Affine Warping

Piecewise warping: each triangle defines a warping.



Problem: possible distortion along the boundaries of the triangles

Resampling is a problem



Ideal warping

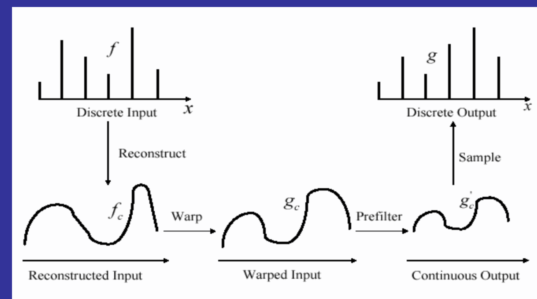
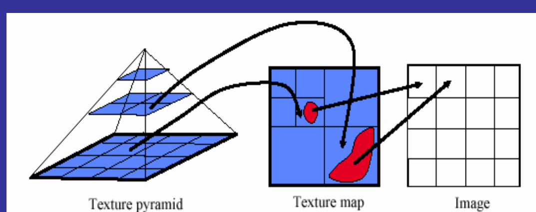
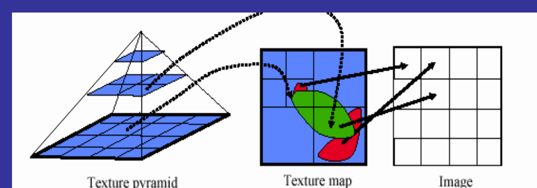


Image pyramids



Trilinear pyramid interpolation



Reference

[1] F. I. Parke, “Parameterized models for facial animation,” *IEEE Computer Graphics and Applications*, vol. 2, no. 9, pp.61-68, Nov. 1982.

[2] Y. Lee, D. Terzopoulos, and K. Waters, “Realistic modeling for facial animation,” in *Proc. SIGGRAPH 95*, pp. 55-62, 1995.

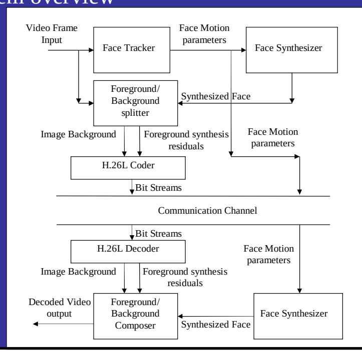
[3] Kalra, A. Mangili, N. M. Thalmann, and D. Thalmann, “Simulation of facial muscle actions based on rational free form deformations,” in *Proc. EUROGRAPHICS'92, pp.* 59-69, Sep. 1992.

[4] N. Arad, N. Dyn, *et al.*, “Image Warping by Radial Basis Functions: Application to Facial Expressions,” *Graphical Models and Image Processing, vol.* 56, *no.* 2, March 1994, pp. 161-172.

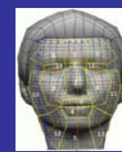
Application: Model based Video Coding based on visual tracking

Basic Idea

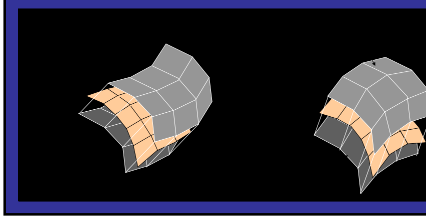
- System overview



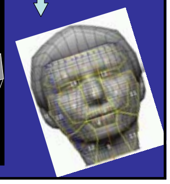
Basic Idea



- Face Tracker
  - Piece-wise Bezier Volume Deformation Face Model
    - Purpose: To design FACS motion units

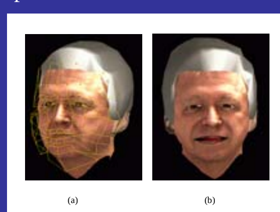


$\hat{R}(\mathbf{V}_0 + L\hat{P}) + \hat{T}$



Basic Idea

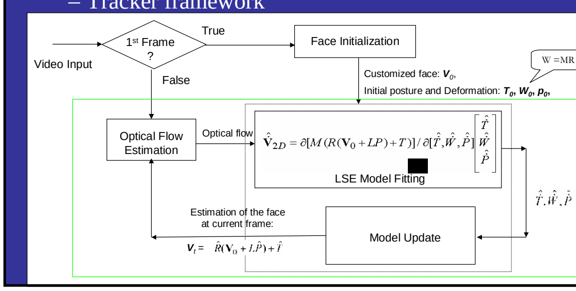
- Face Tracker
  - A example of PBVD Face Model



(a) Bézier controlling mesh (b) An expression "smile".

Basic Idea

- Face Tracker
  - Tracker framework



Basic Idea

- Embedding synthesized face in H.26L Codec
  - Let H.26L take care the background efficiently.
  - The foreground reconstruction residuals coded with Intra_16X16 mode.

Experiment results

- Tracker: robust tracking at 25fps in rigid tracking mode and 14 fps in non-rigid mode on a PC with CPU of 2GHz.
- H.26L(JM42): IMTC ftp website: http://ftp.imtc-files.org.

	Coding with facial tracker	H.26L
Video Input	352X240	
Bit rate	18-19kbits/second	
PSNR around facial area	29.28	27.35
Coding speeding	1.4 sec/frame	5 sec/frame

Experiment results

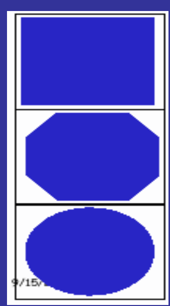
(a) The synthesized face motion

(b) The reconstructed video frame with synthesized face motion

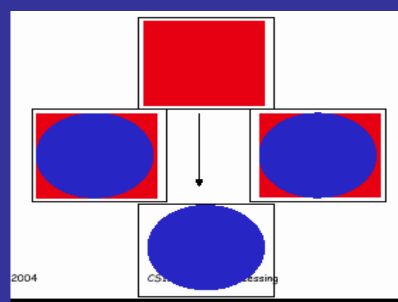
(c) The reconstructed video frame using H.26L codec!

Experiment results

Warping



Blending

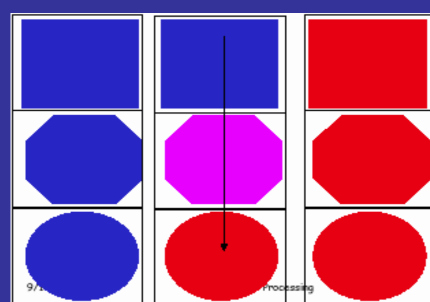


Blending

- Blending is linear interpolation:

$$\alpha(t)I_0 + (1-\alpha(t))I_1$$

Morphing:warping+blending



Morphing example