Motion Models

- Image Transformations to relate two images
- 3D Rigid motion
- Perspective & Orthographic Transformation
- Planar Scene Assumption
- Transformations
  - Translation
  - Rotation
  - Rigid
  - Affine
  - Homography
  - Pseudo Perspective

Global Flow

Lecture-9

Global Motion

- Estimate motion using all pixels in the image.
- Parametric flow gives an equation, which describes optical flow for each pixel.
  - Affine
  - Projective
- Global motion can be used to
  - Remove camera (ego) motion (motion compensation)
  - Object-based segmentation
  - generate mosaics

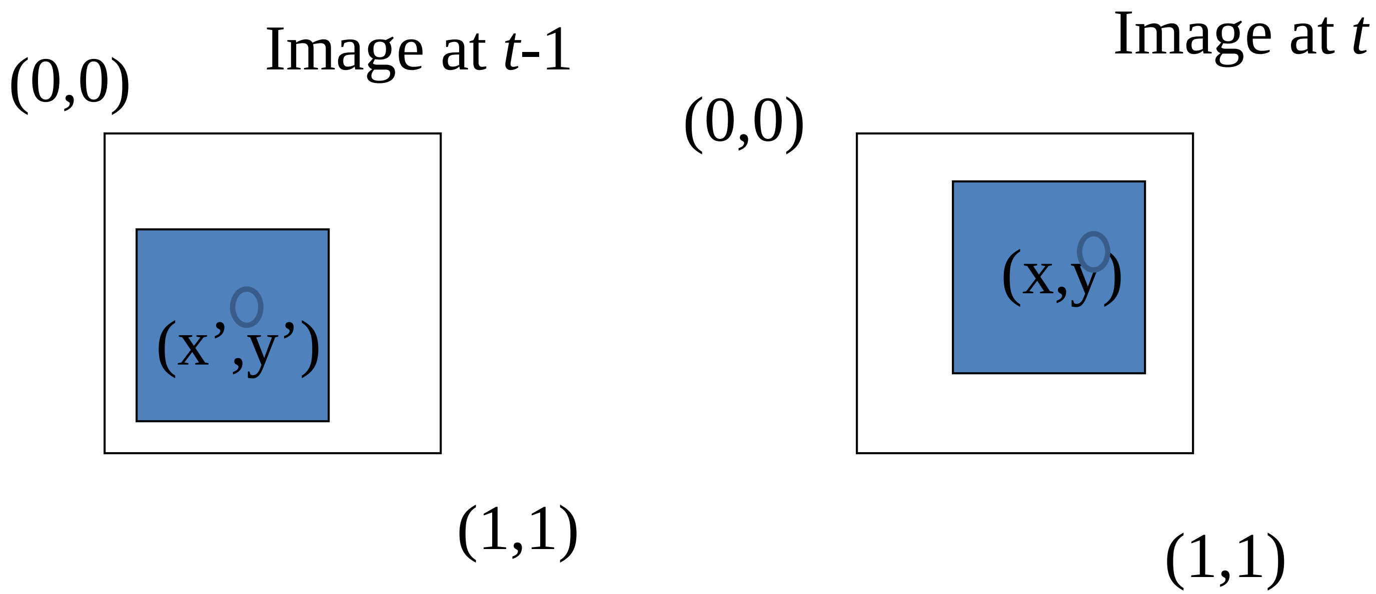
Contents

- Bergen *et al* method
  - Affine transformation
- Mann & Piccard
  - Homography (Projective)
  - Pseudo Perspective
  - Bi-linear
- Image Warping
- Applications
  - Mosaics
  - COCOA system

Bergan et al

Affine

Affine



$$u(x,y) = a_1 x + a_2 y + b_1$$
$$v(x,y) = a_3 x + a_4 y + b_2$$

$$U = X - X'$$
$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} x' \\ y' \end{bmatrix}$$

Affine

$$u(x, y) = a_1 x + a_2 y + b_1$$
$$v(x, y) = a_3 x + a_4 y + b_2$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

Bergan et al

$$u(x, y) = a_1 x + a_2 y + b_1$$

$$v(x, y) = a_3 x + a_4 y + b_2$$

- Affine

$$\begin{bmatrix} u(x, y) \\ v(x, y) \end{bmatrix} = \begin{bmatrix} x & y & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & y & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ b_1 \\ a_3 \\ a_4 \\ b_2 \end{bmatrix}$$

$$\mathbf{u}(\mathbf{x}) = \mathbf{X}(\mathbf{x})\mathbf{a}$$

Bergan et al

$$\mathbf{u}(\mathbf{x}) = \mathbf{X}(\mathbf{x})\mathbf{a}$$

Optical flow constraint eq

$$f_x u + f_y v = -f_t$$

$$E(\mathbf{u}) = \sum_{\forall x \in f(x,y)} (f_t + f_{\mathbf{x}}^T \mathbf{u})^2$$

$$f_{\mathbf{X}} = \begin{bmatrix} f_x \\ f_y \end{bmatrix}$$

$$E(\mathbf{a}) = \sum_{\forall x \in f(x,y)} (f_t + f_{\mathbf{x}}^T \boxed{\mathbf{X}(\mathbf{x})\mathbf{a}})^2$$

$$E(\delta a) = \sum_{\forall x \in f(x,y)} (f_t + f_{\mathbf{x}}^T \mathbf{X} \delta a \ )^2$$

Bergan et al

$$E(\delta a) = \sum_{\forall x \in f(x,y)} (f_t + f_{\mathbf{x}}^T \mathbf{X} \delta a \ )^2$$

min

$$\left[\sum \ X^T (f_X (f_X)^T X\right] \delta a = -\sum X^T f_X f_t$$

Homework

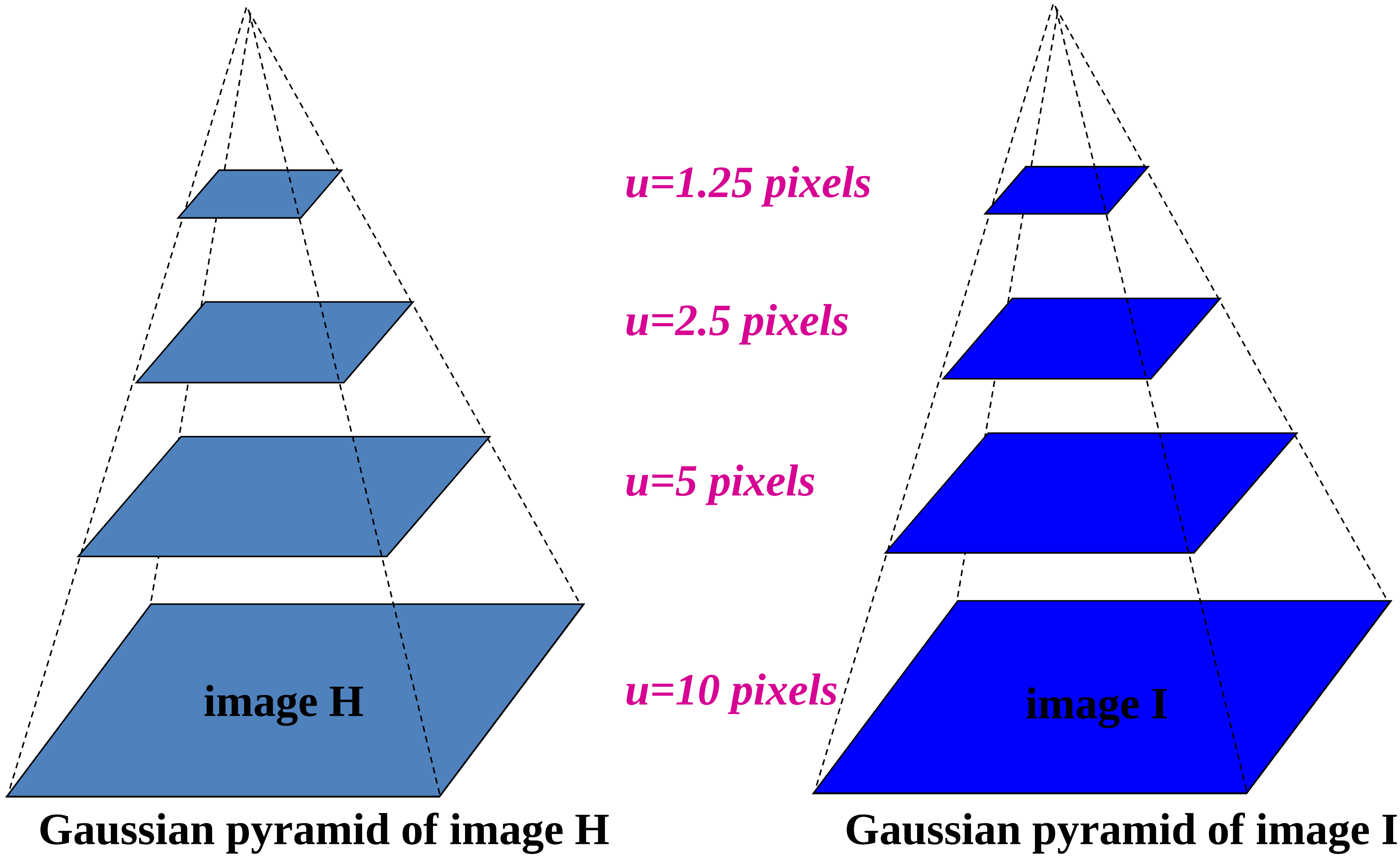
$$Aa = B$$

Linear system

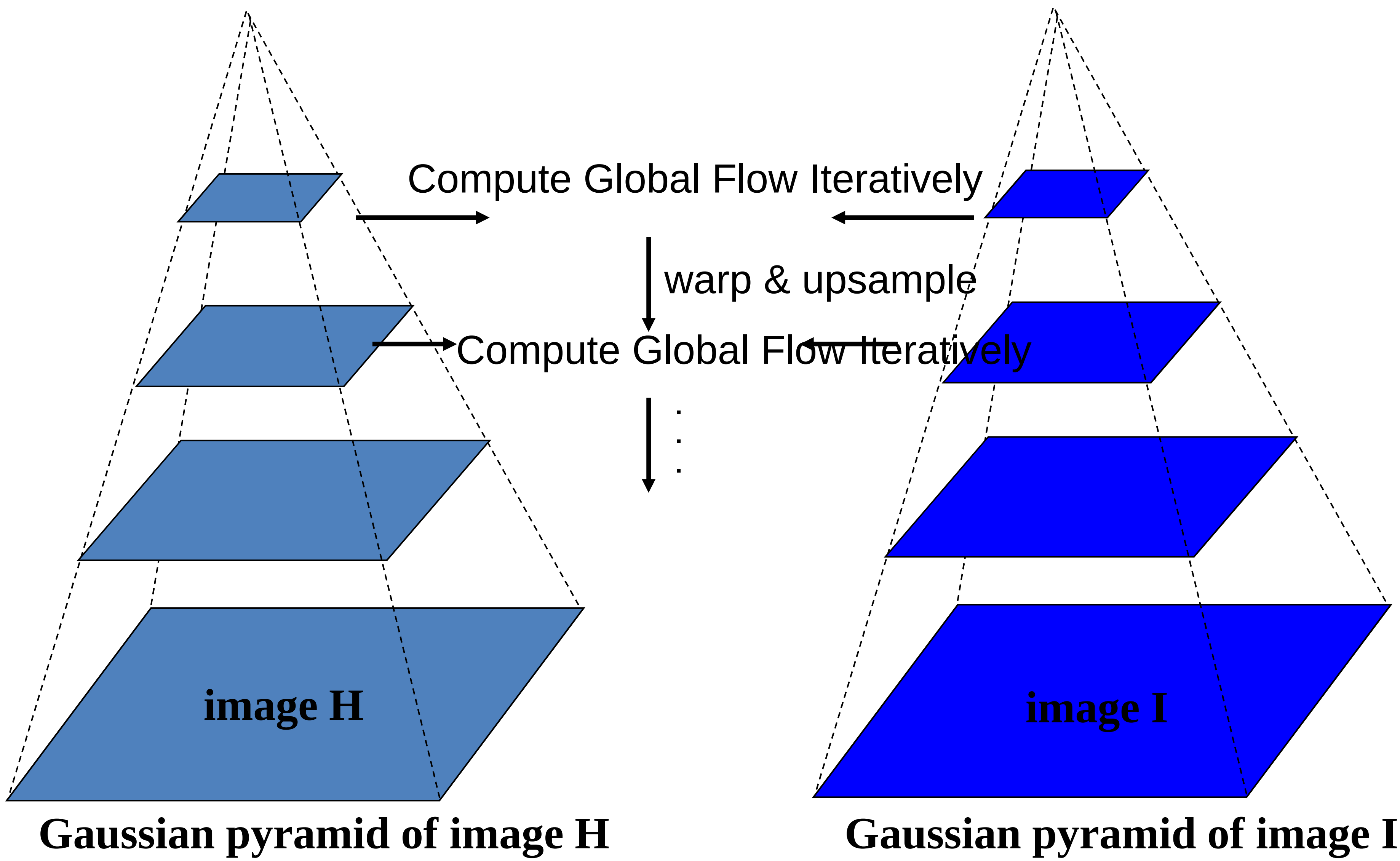
Basic Components

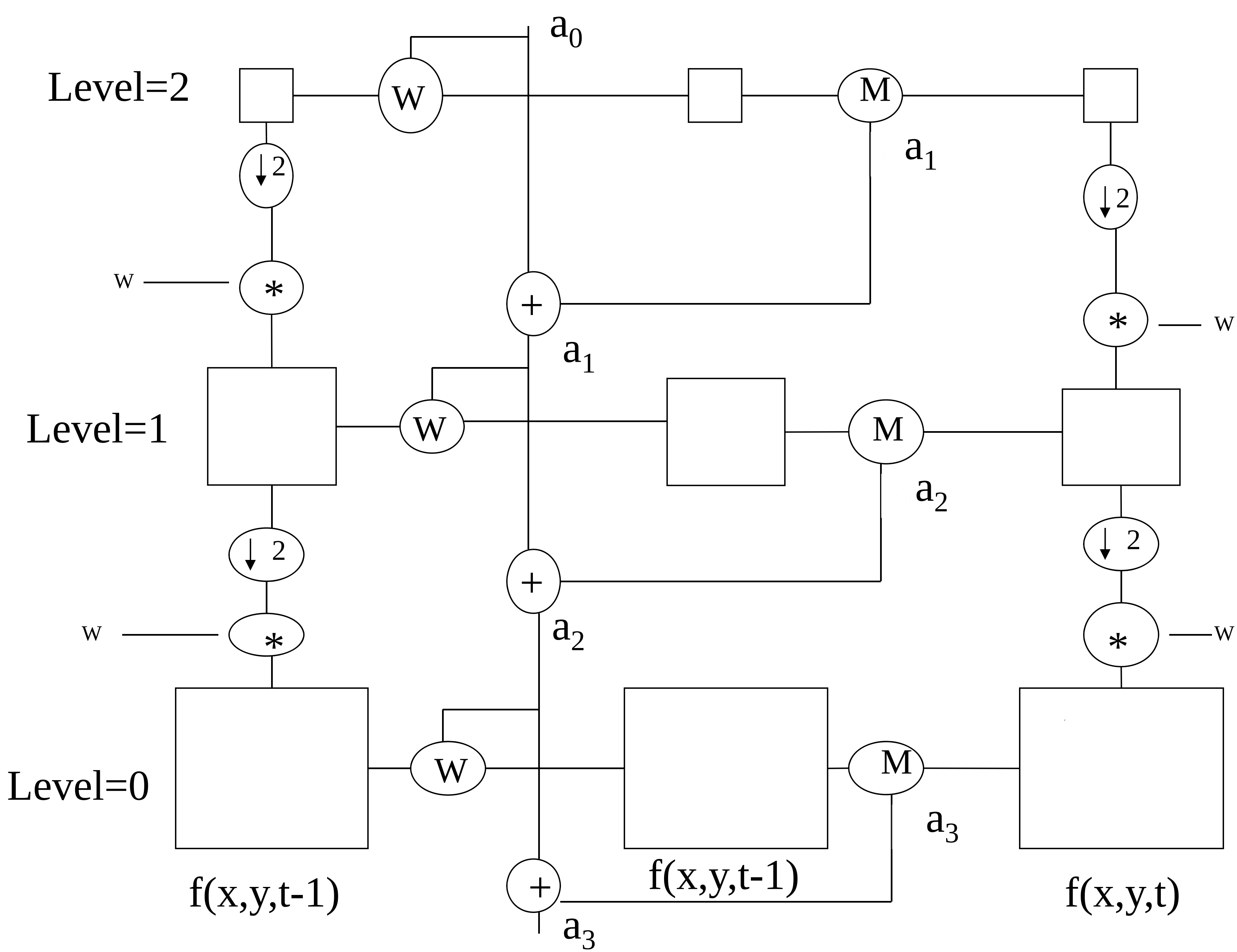
- Pyramid construction
- Motion estimation
- Image warping
- Coarse-to-fine refinement

Coarse-to-fine global flow estimation



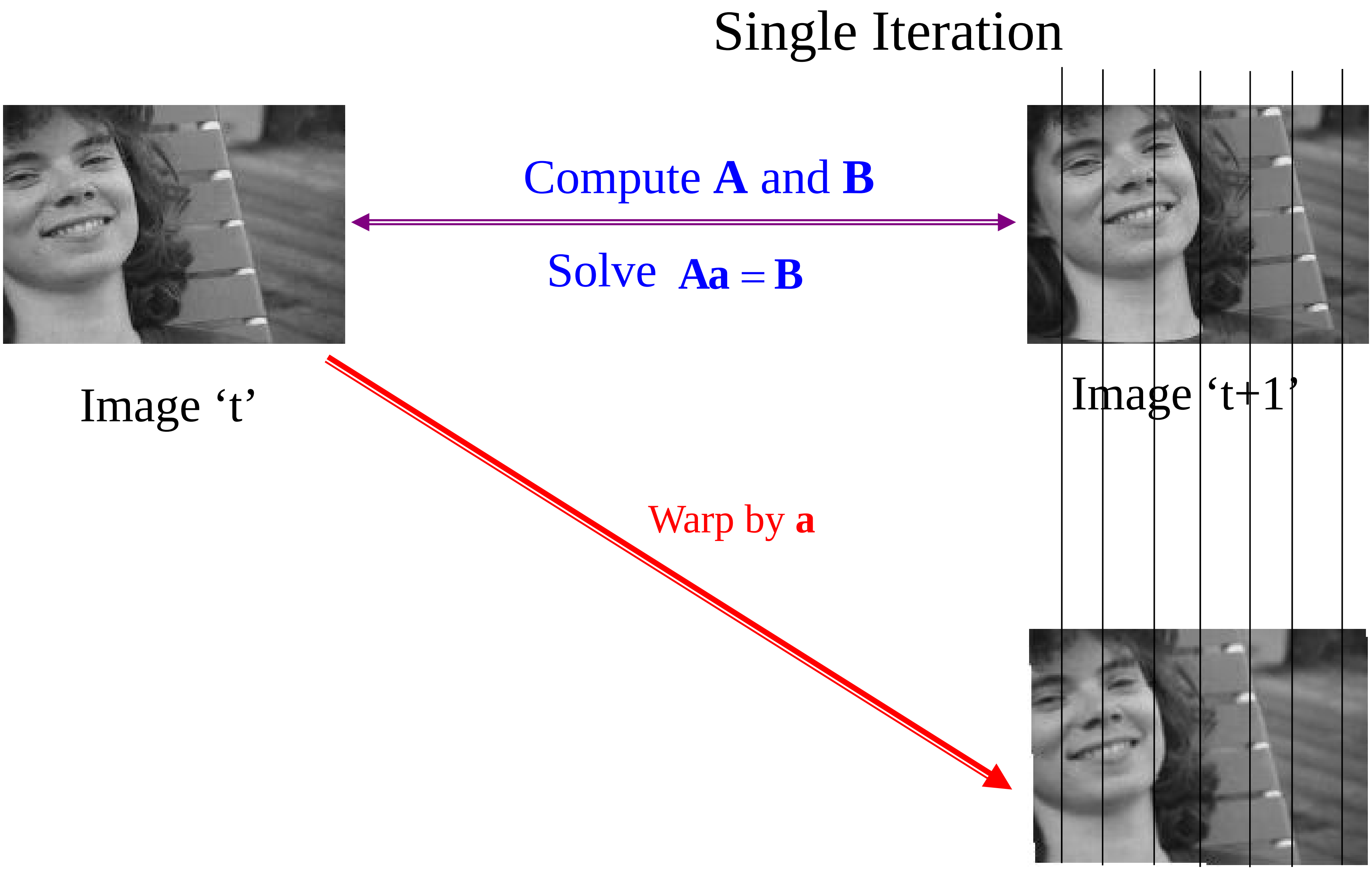
Coarse-to-fine global flow estimation





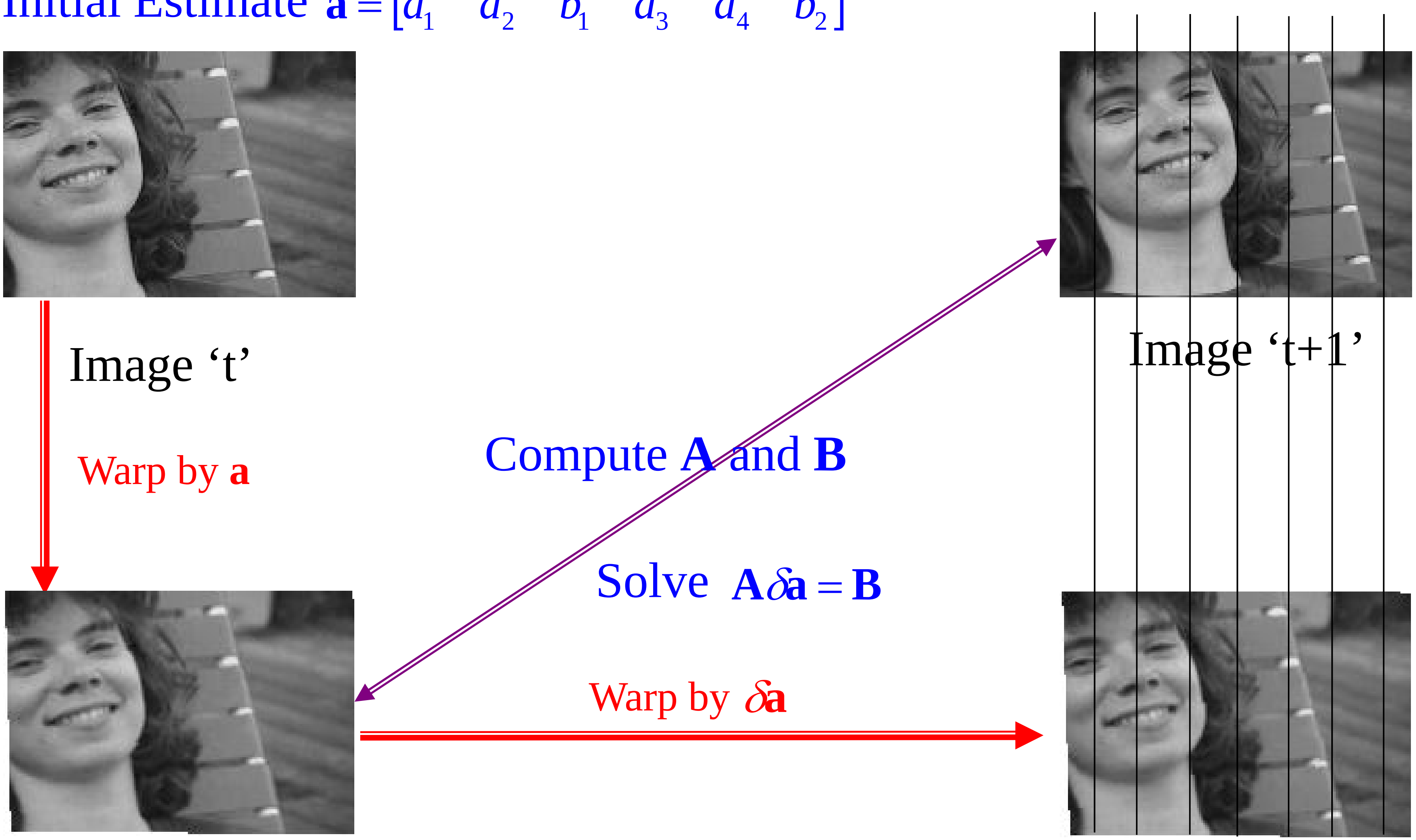
a0
Level=2
W
M
a1
2
w
*
+
a1
Level=1
W
M
a2
2
w
*
+
a2
Level=0
W
M
a3
+
a3
f(x,y,t-1)
f(x,y,t-1)
f(x,y,t)

Estimation of Global Flow



Estimation of Global Flow

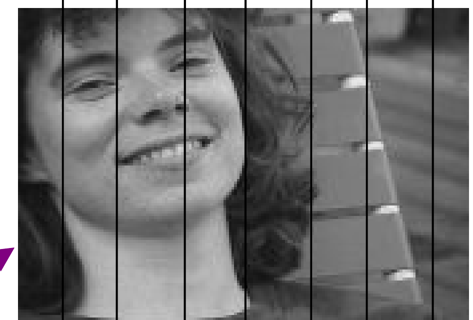
Initial Estimate $\mathbf{a} = [a_1 \quad a_2 \quad b_1 \quad a_3 \quad a_4 \quad b_2]^T$



Estimation of Global Flow

Iterative

Initial Estimate $\mathbf{a} = [a_1 \quad a_2 \quad b_1 \quad a_3 \quad a_4 \quad b_2]^T$



Iterate

Image ‘t’

Image ‘t+1’

Warp by **a**

Compute **A** and **B**

Solve $\mathbf{A}\delta\mathbf{a} = \mathbf{B}$

Update **a**

Image Warping

- Warping an image *f* into image *h* using some transformation *g*,
  - involves mapping intensity at each pixel (x,y) in image *f* to a pixel ($g(x)$,g(y)) in image *h* such that

$$f(x', y') = h(g(x), g(y))$$

In case of affine transformation, $\mathbf{x} = (x, y)$ is transformed to $\mathbf{x}' = (x', y')$ as:

$$\mathbf{x}' = \mathbf{A}\mathbf{x} + \mathbf{b}$$ Displacement model

$$\mathbf{U} = \mathbf{x} - \mathbf{x}' = \mathbf{A}\mathbf{x} + \mathbf{b}$$ Instantaneous model

Image Warping (Bergan et al)

$$X' = X - U = X - (AX + b)$$

$$X' = (I - A)X - b$$

$$X' = A'X - b$$

$$X' + b = A'X$$

$$(A')^{-1}(X' + b) = X$$

↓ warp

$$(A')^{-1}(X' + b) = X''$$

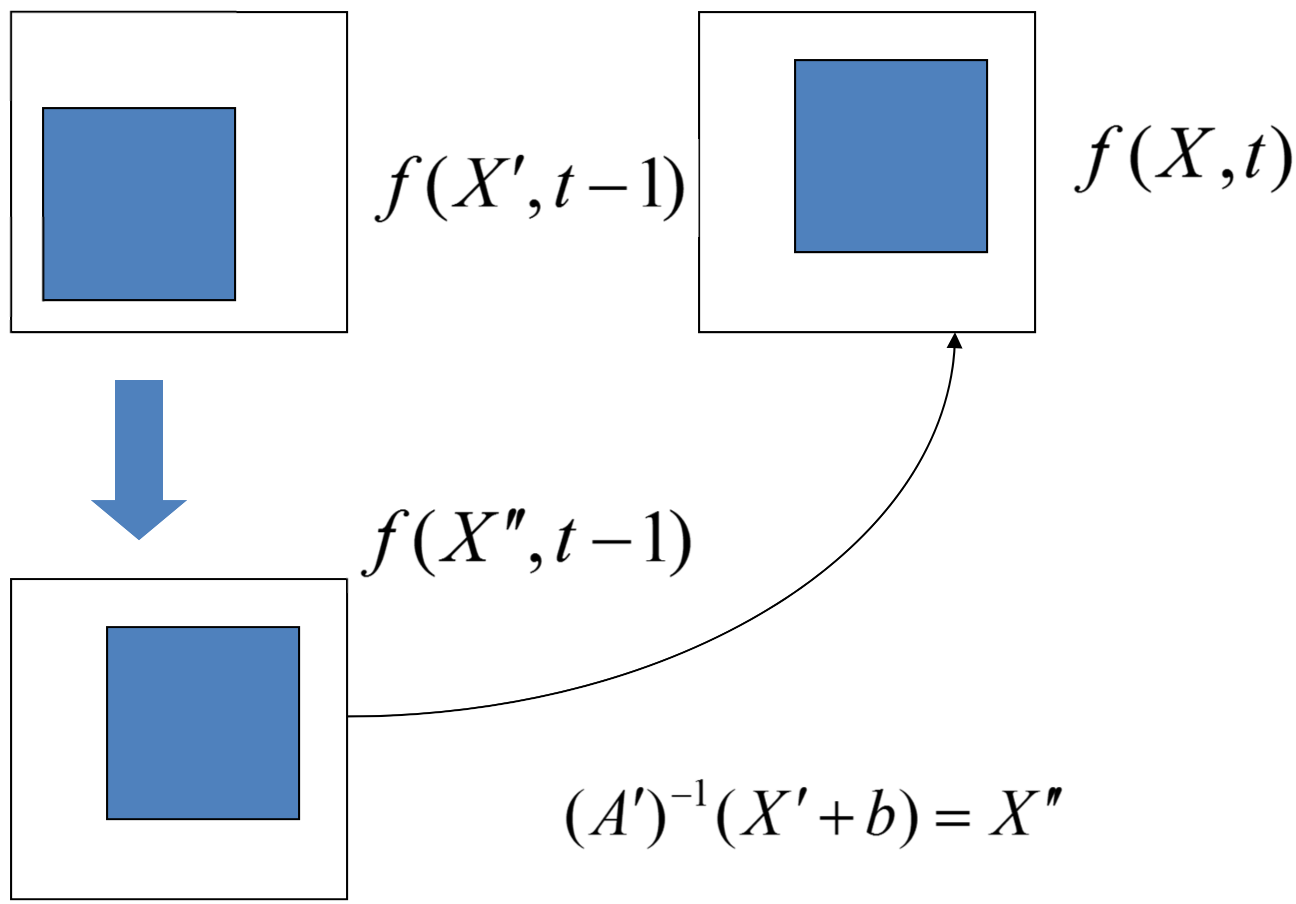


Image Warping

- How about values in $X'' = (x'', y'')$ are not integer.
- But image is sampled only at integer rows and columns
  - Instead of converting X' to X'' and copying $f(X', t-1)$ at $f(X'', t-1)$ we can convert integer values X'' to X' and copy $f(X', t-1)$ at $f(X'', t-1)$

Image Warping

$$X' = X - U$$

$$X' = X - (AX + b)$$

$$X' = X'' - (AX'' + b)$$

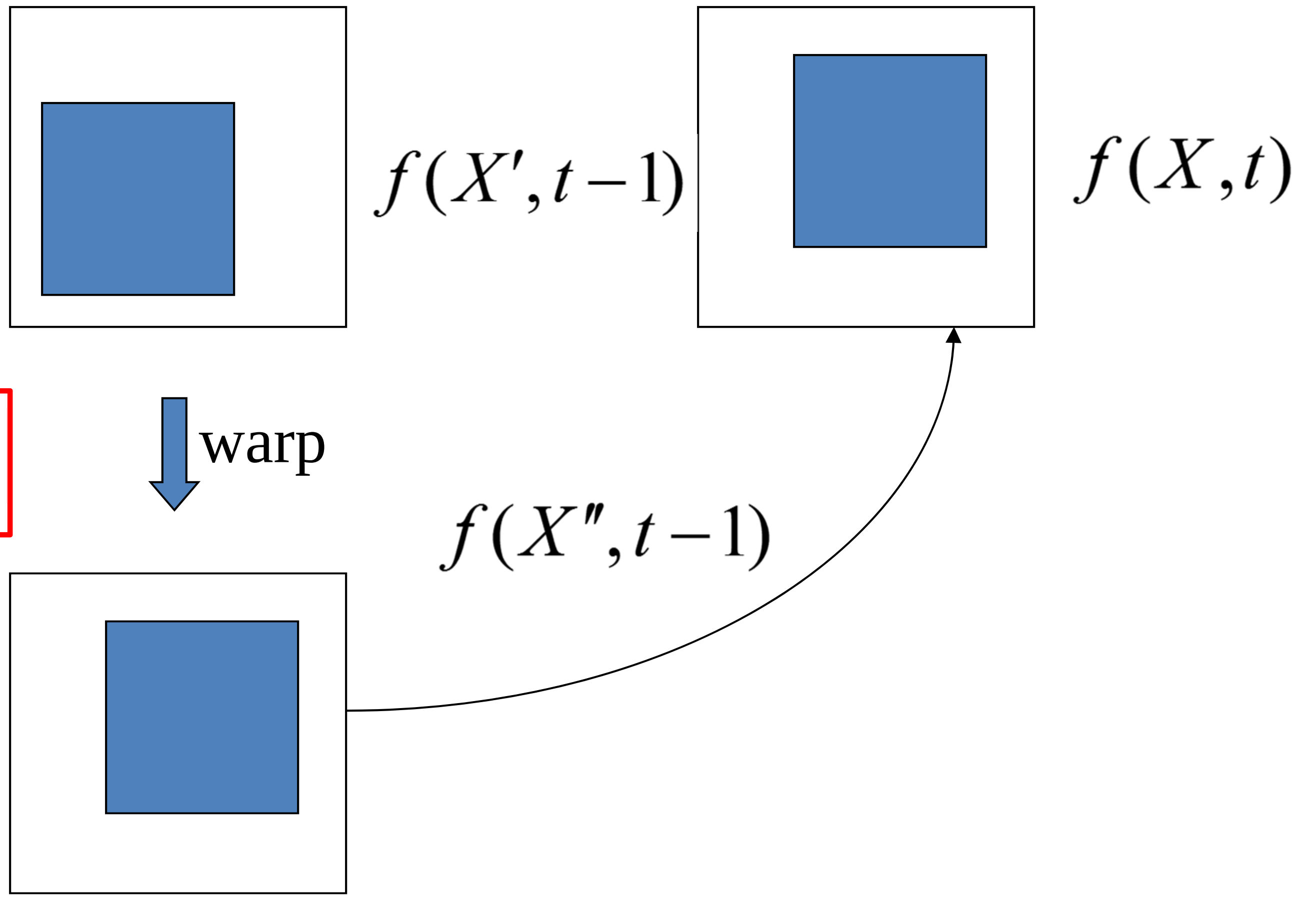


Image Warping

• But how about the values in X' are not integer.

• Perform bilinear interpolation to compute at non-integer values.

Warping

$f(X', t-1)$

$f(X', t-1)$

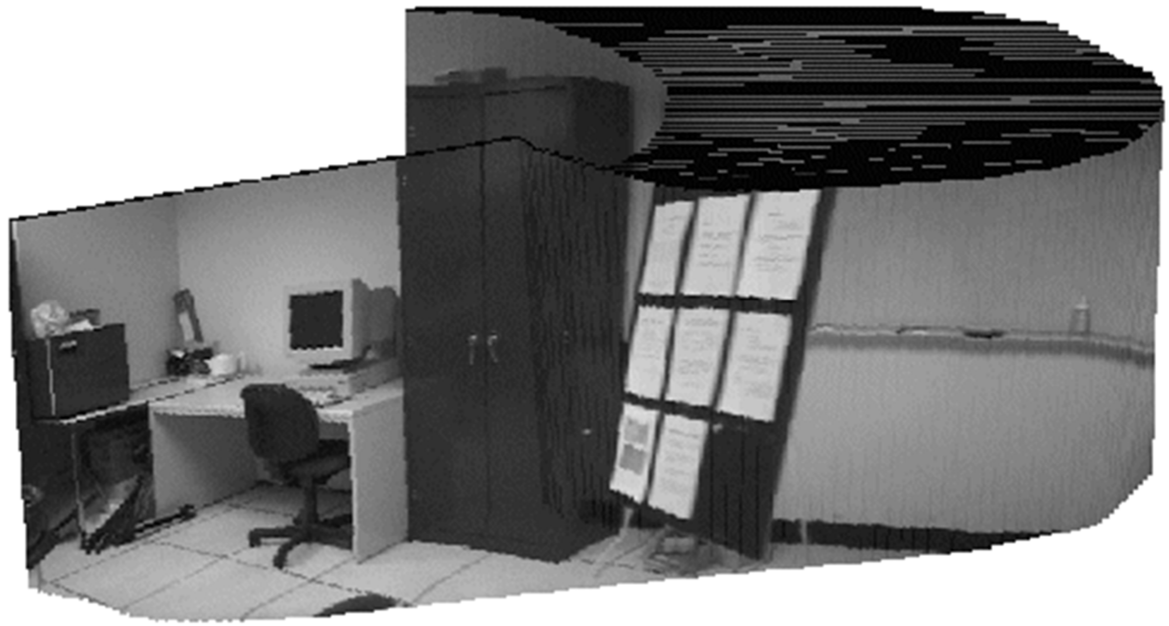
Warped image at *t-1*

Difference image before

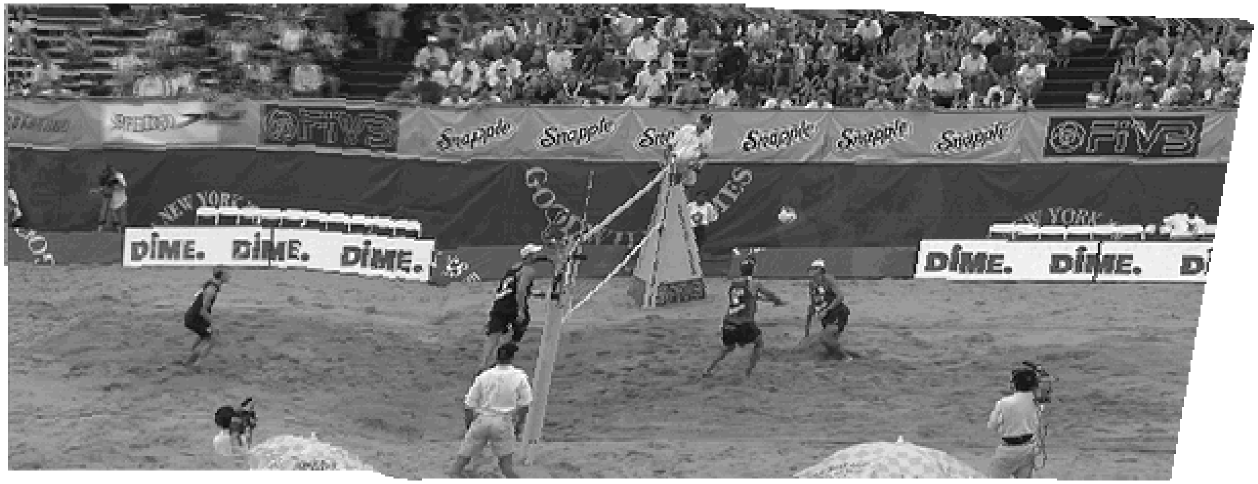
Difference image after

Video Mosaic

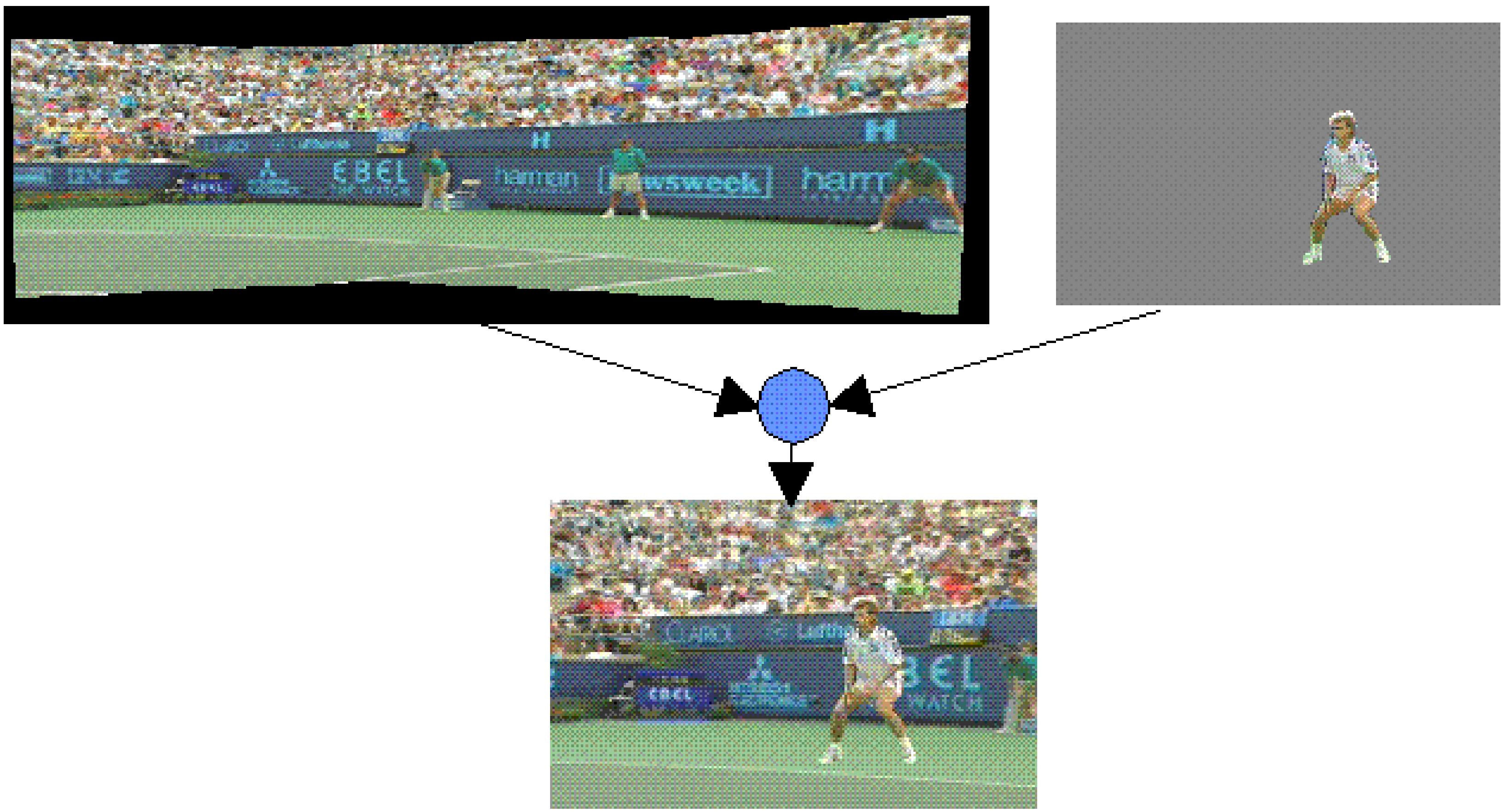
Video Mosaic

Video Mosaic



Sprite



Mann & Picard

Projective

Projective Flow (weighted)

$$u f_x + v f_y + f_t = 0$$

Optical Flow const. equation

$$\mathbf{u}^T \mathbf{f}_{\mathbf{x}} + f_t = 0$$

$$\mathbf{x}' = \frac{A\,\mathbf{x} + \mathbf{b}}{\mathbf{C}^{\mathbf{T}}\mathrm{x} + 1}$$

Projective transform

$$\mathbf{u} = \mathbf{x}' - \mathbf{x} = \frac{A\mathbf{x} + \mathbf{b}}{\mathbf{C}^{\mathbf{T}}\mathrm{x} + 1} - \mathbf{x}$$

Projective Flow (weighted)

$$\varepsilon_{flow} = \sum (\mathbf{u}^{\mathbf{T}}\mathbf{f}_{\mathbf{X}} + f_{\mathbf{t}})^2$$

$$= \sum ((\frac{A\mathbf{x}+\mathbf{b}}{\mathbf{C}\mathbf{x}^{\mathbf{T}}+1} - \mathbf{x})^T \mathbf{f}_{\mathbf{x}} + f_t)^2$$

$$= \sum ((A\mathbf{x}+\mathbf{b} - (\mathbf{C}\mathbf{x}^{\mathbf{T}}+1)\mathbf{x})^T \mathbf{f}_{\mathbf{x}} + (\mathbf{C}\mathbf{x}^{\mathbf{T}}+1) f_t)^2$$

⇩ **minimize**

Homework

Projective Flow (weighted)

$$(\sum \phi\phi^T)\mathbf{a} = \sum (\mathbf{x}^{\mathbf{T}}\mathbf{f}_x - f_t)\phi$$

$$\mathbf{a} = \left[a_1, a_2, b_1, a_3, a_4, b_2, c_1, c_2\right]^T$$

$$\phi^t = [f_x x, f_x y, f_x, f_y x, f_y y, f_y, x f_t - x^2 f_x - xy f_y, y f_t - xy f_x - y^2 f_y]$$

Projective Flow (unweighted)

Pseudo-Perspective

$$\mathbf{x}' = \frac{A\mathbf{x} + \mathbf{b}}{\mathbf{C}^{\mathbf{T}}\mathrm{x} + 1}$$

⇓ **Taylor Series**

$$x + u = a_1 + a_2 x + a_3 y + a_4 x^2 + a_5 xy$$

$$y + v = a_6 + a_7 x + a_8 y + a_4 xy + a_5 y^2$$

Bilinear

$$\mathbf{x}' = \frac{A\mathbf{x} + \mathbf{b}}{\mathbf{C}^{\mathbf{T}}\mathrm{x} + 1}$$

Taylor Series & remove Square terms

$$u + x = a_1 + a_2 x + a_3 y + a_4 xy$$

$$v + y = a_5 + a_6 x + a_7 y + a_8 xy$$

Projective Flow (unweighted)

$$\varepsilon_{flow} = \sum (\mathbf{u}^{\mathrm{T}} \mathbf{f}_{\mathbf{X}} + f_{\mathbf{t}})^2$$

Minimize

Bilinear and Pseudo-Perspective

$$(\sum \Phi\Phi^T)\mathbf{q} = -\sum f_t \Phi$$

$$\Phi^T = \left[f_x(xy, x, y, 1), f_y\left(xy, x, y, 1\right)\right]$$ **bilinear**

$$\Phi^T = \left[f_x(x, y, 1), f_y\left(x, y, 1\right), c_1, c_2\right]$$

$$c_1 = x^2 f_x + xyf_x$$

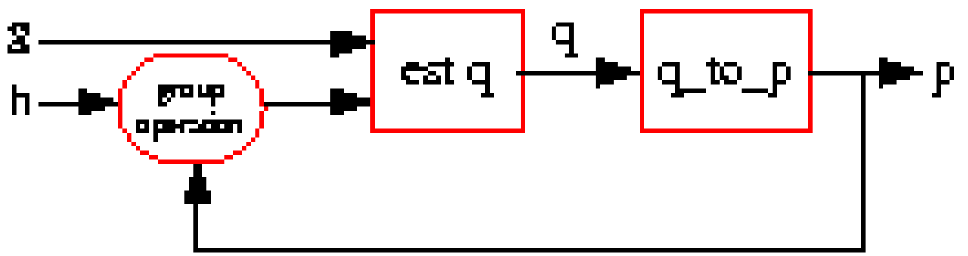
$$c_2 = xyf_x + y^2 f_y$$

Pseudo perspective

Homework

Algorithm-1

- Estimate "q" (using approximate model, e.g. bilinear model).
- Relate "q" to "p"
  - select four points S1, S2, S3, S4
  - apply approximate model using "q" to compute
  - estimate exact "p":


g
h
group operation
est q
q
q_to_p
p

Determining Projective transformation using point correspondences

- If point correspondences (x,y)<-->(x',y') are known
- a's can be determined by least squares fit

$$x' = \frac{a_1 x + a_2 y + a_3}{a_7 x + a_8 y + 1}$$

$$y' = \frac{a_4 x + a_5 y + a_6}{a_7 x + a_8 y + 1}$$

$$a_7 x' x + a_8 x' y + x' = a_1 x + a_2 y + a_3$$

$$a_7 y' x + a_8 y' y + y' = a_7 x + a_8 y + a_6$$

$$x' = a_1 x + a_2 y + a_3 - a_7 x' x - a_8 x' y$$

$$y' = a_7 x + a_8 y + a_6 - a_7 y' x - a_8 y' y$$

$$a_1 x + a_2 y + a_3 - a_7 x' x - a_8 x' y = x'$$

$$a_7 x + a_8 y + a_6 - a_7 y' x - a_8 y' y = y'$$

Two rows for each point i

$$\begin{bmatrix} & & & & \vdots & & & \\ x_i & y_i & 1 & 0 & 0 & 0 & -x_i x_i' & -y_i x_i' \\ 0 & 0 & 0 & x_i & y_i & 1 & -x_i y_i' & -y_i y_i' \\ & & & & \vdots & & & \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{bmatrix} = \begin{bmatrix} \vdots \\ x_i' \\ y_i' \\ \vdots \end{bmatrix}$$

Determining Projective transformation using point correspondences

$$\begin{bmatrix} & & & & & \vdots & & \\ x_i & y_i & 1 & 0 & 0 & 0 & -x_i x_i' & -y_i x_i' \\ 0 & 0 & 0 & x_i & y_i & 1 & -x_i y_i' & -y_i y_i' \\ & & & & & \vdots & & \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{bmatrix} = \begin{bmatrix} \vdots \\ x_i' \\ y_i' \\ \vdots \end{bmatrix}$$

$$Aa = \mathbf{x}'$$

$$a = (A^T A)^{-1} A^T \mathbf{x}'$$

Final Algorithm

- A Gaussian pyramid of three or four levels is constructed for each frame in the sequence.
- The parameters “p” are estimated at the top level of the pyramid, between the two lowest resolution images, “g” and “h”, using algorithm-1.

Final Algorithm

- The estimated “p” is applied to the next higher resolution image in the pyramid, to make images at that level nearly congruent.
- The process continues down the pyramid until the highest resolution image in the pyramid is reached.

Video Mosaics

- Mosaic aligns different pieces of a scene into a larger piece, and seamlessly blend them.
  - High resolution image from low resolution images
  - Increased filed of view

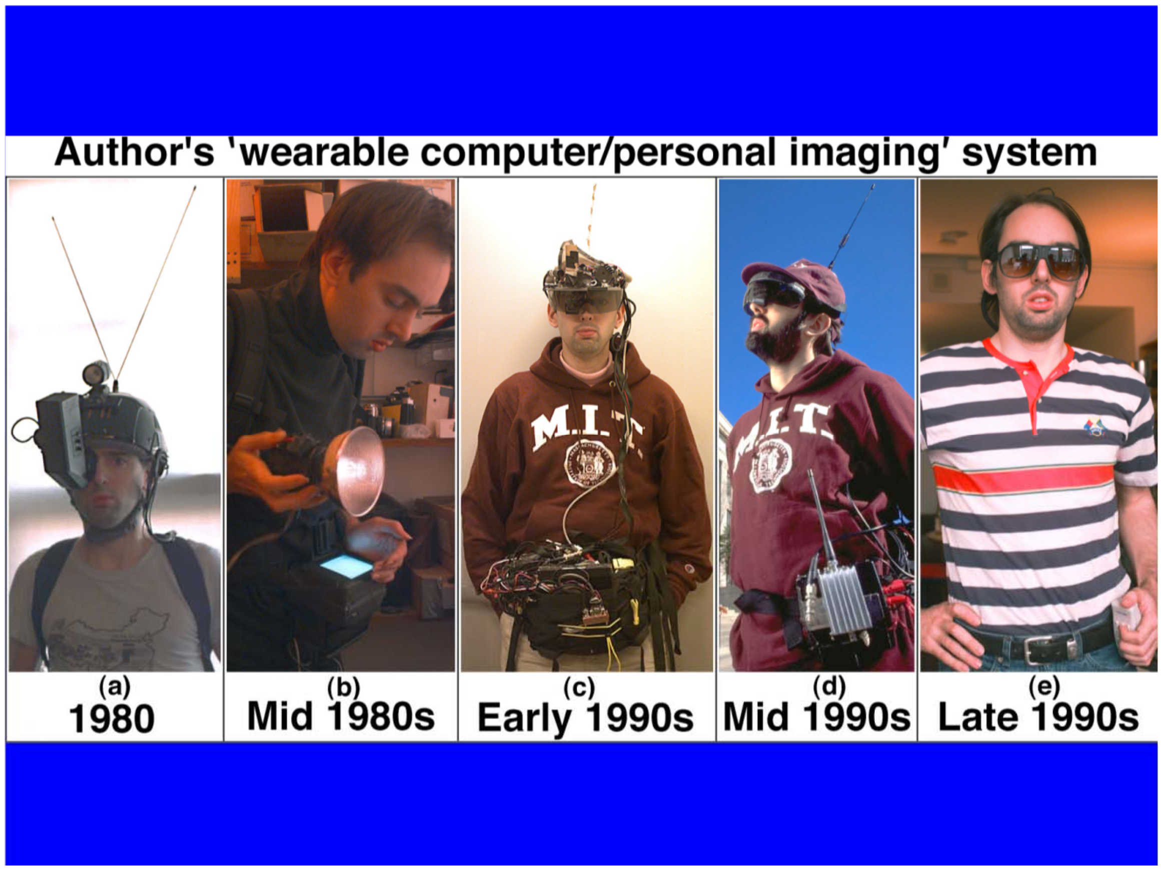
Steps in Generating A Mosaic

- Take pictures
- Pick reference image
- Determine transformation between frames
- Warp all images to the same reference view

Applications of Mosaics

- Virtual Environments
- Computer Games
- Movie Special Effects
- Video Compression

Steve Mann


Author's 'wearable computer/personal imaging' system
M.I.T.
M.I.T.
(a)
1980
(b)
Mid 1980s
(c)
Early 1990s
(d)
Mid 1990s
(e)
Late 1990s

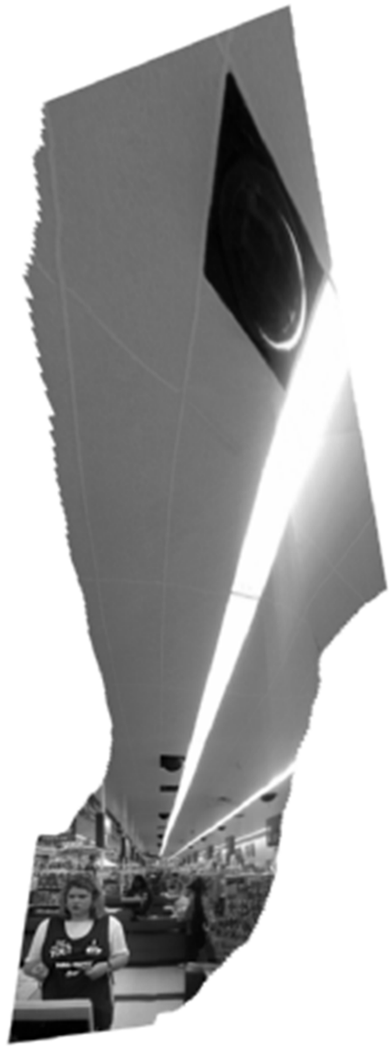
Sequence of Images

Projective Mosaic

Affine Mosaic

Building

Wal-Mart



Scientific American Frontiers

Scientific American Frontiers

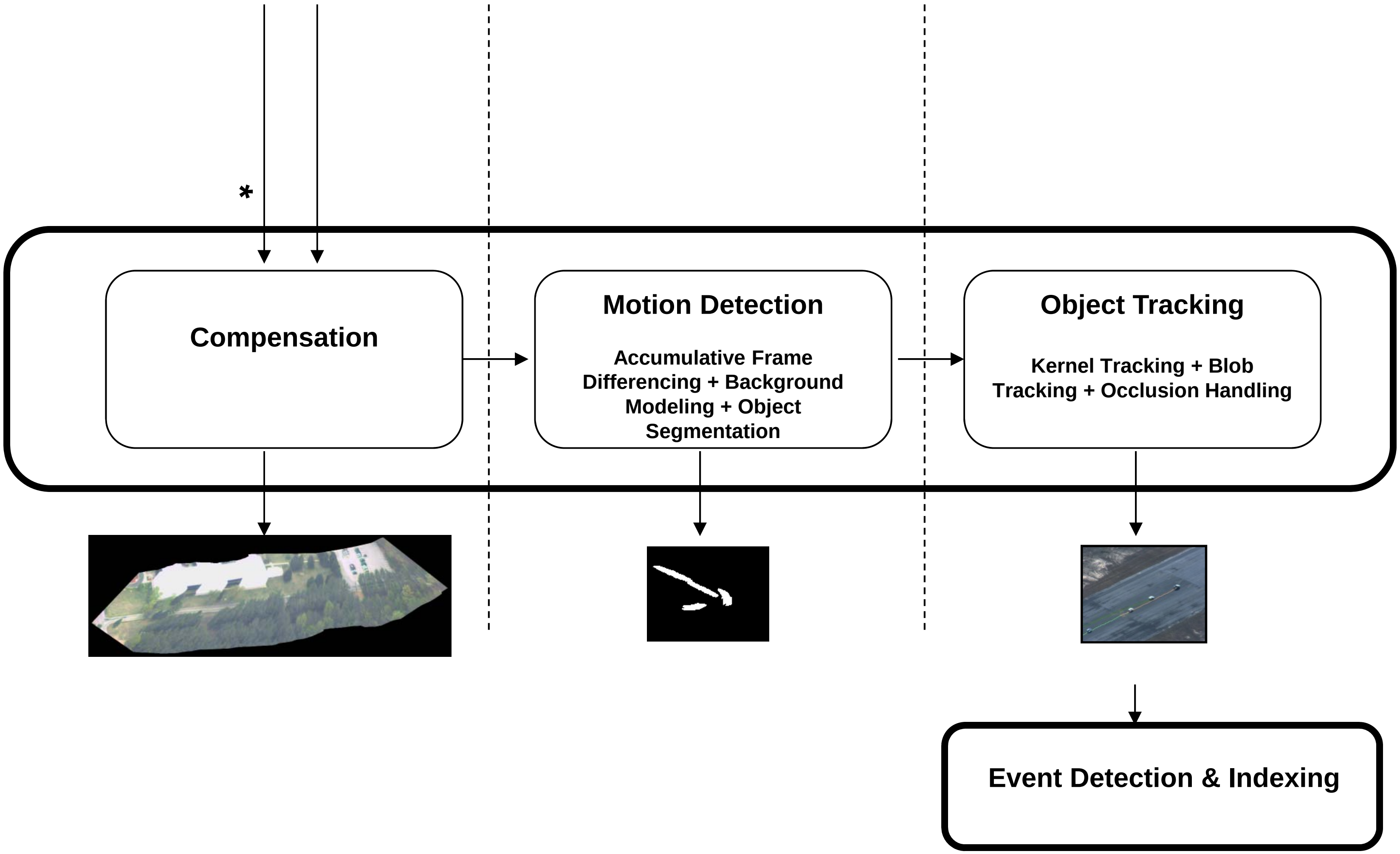
Head-mounted Camera at Restaurant

MIT Media Lab

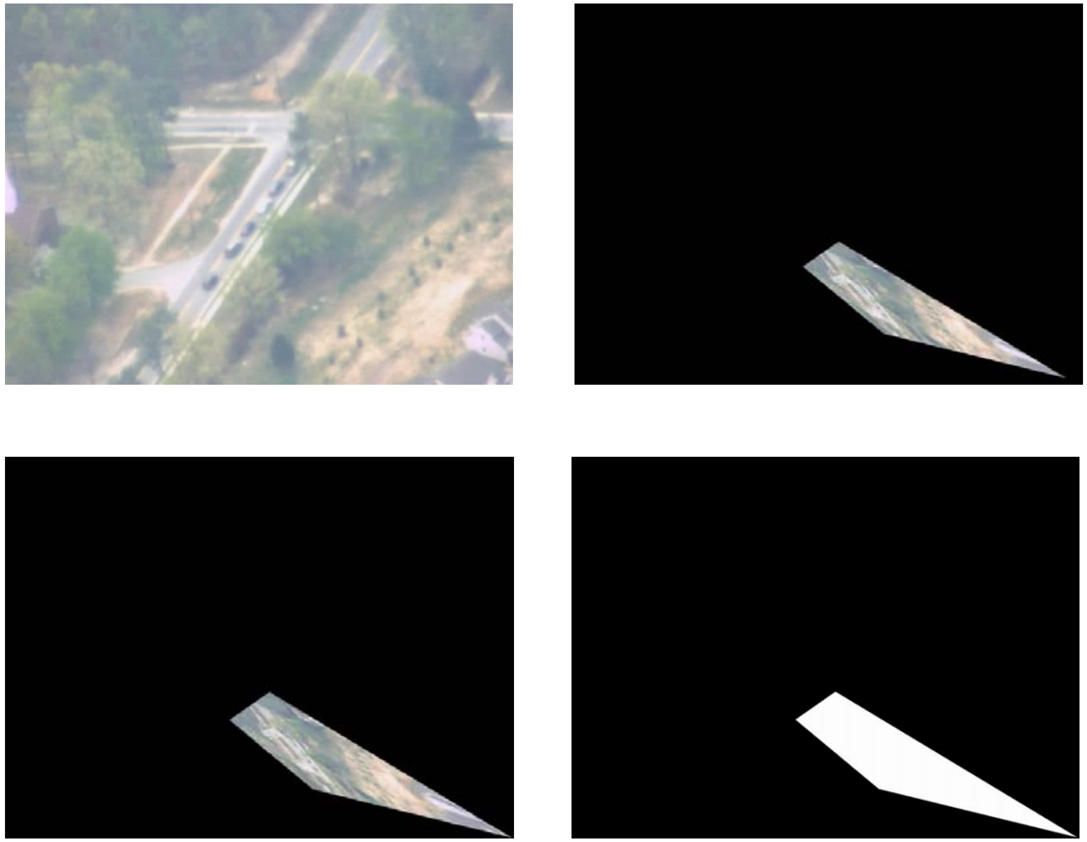
COCOA: A System for Processing of Aerial Videos

Saad Ali and Mubarak Shah, COCOA - Tracking in Aerial Imagery, SPIE Airborne Intelligence, Surveillance, Reconnaissance (ISR) Systems and Applications, Orlando, 2006.

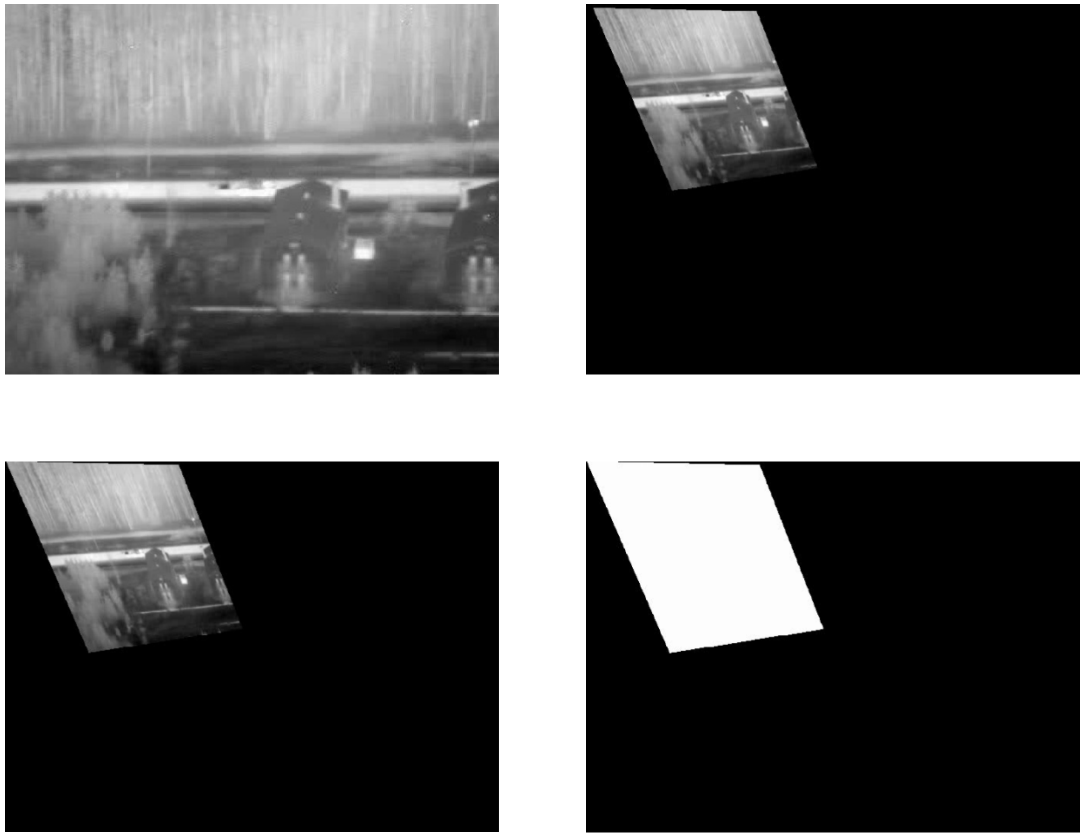
COCOA – System Flow



Registration Result - I



Registration Result - II



Detection Results

Tracking Results

References

- J. Bergen, P. Anandan, K. Hanna, and R. Hingorani, "Hierarchical Model-Based Motion Estimation", ECCV-92, pp 237-22.
- Video orbits of the projective group a simple approach to featureless estimation of parameters S Mann, RW Picard - Image Processing, IEEE Transactions on, 1997
- Saad Ali and Mubarak Shah, COCOA - Tracking in Aerial Imagery, SPIE Airborne Intelligence, Surveillance, Reconnaissance (ISR) Systems and Applications, Orlando, 2006.
- R. Szeliski. "Video mosaics for virtual environments", IEEE Computer Graphics and Applications, pages,22-30, March 1996.