Image alignment

http://www.cs.cmu.edu/~16385/

16-385 Computer Vision
Spring 2018, Lecture 23

Course announcements

- Homework 6 has been posted and is due on April 20th.
  - Any questions about the homework?
  - How many of you have looked at/started/finished homework 6?

- This week Yannis' office hours will be on Wednesday 3-6 pm.
  - Added an extra hour to make up for change.

Overview of today's lecture

- Leftover from last time: Horn-Schunck flow.
- Motion magnification using optical flow.
- Image alignment.
- Lucas-Kanade alignment.
- Baker-Matthews alignment.
- Inverse alignment.

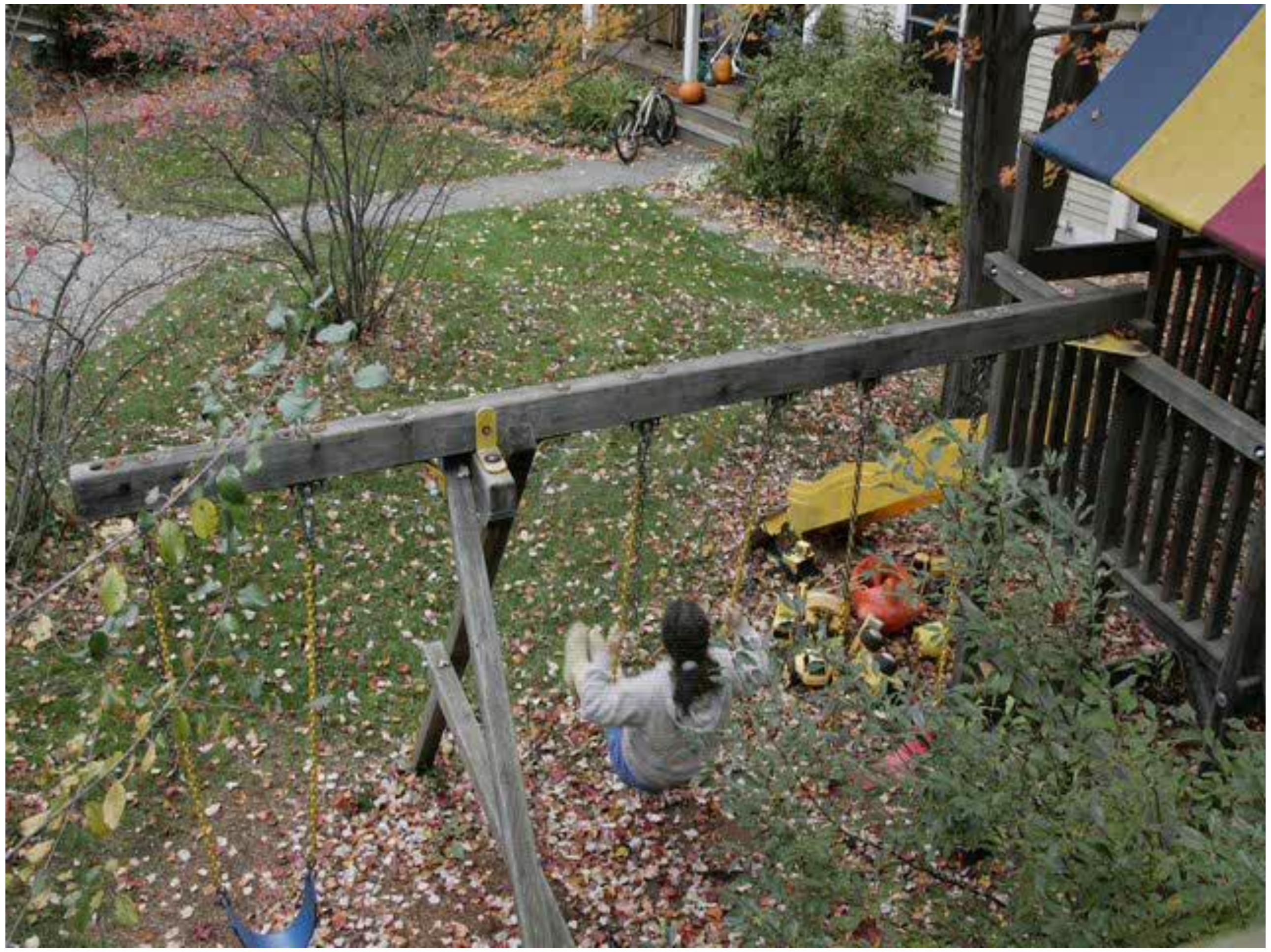
Slide credits

Most of these slides were adapted from:

- Kris Kitani (16-385, Spring 2017).

Motion magnification using optical flow

How would you achieve this effect?



original

motion-magnified

- Compute optical flow from frame to frame.
- Magnify optical flow velocities.
- Appropriately warp image intensities.

How would you achieve this effect?

naïvely motion-magnified

motion-magnified

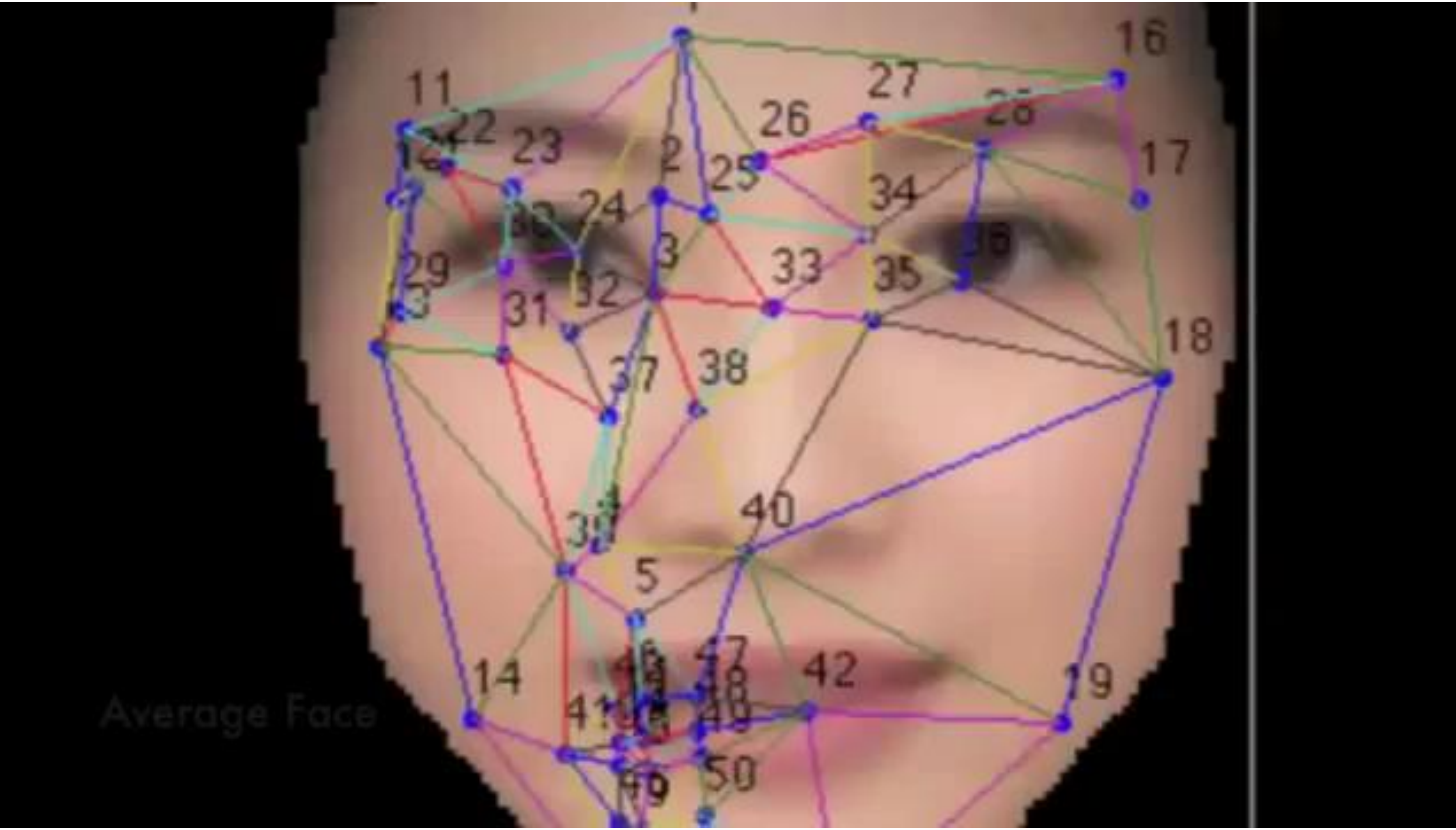
- Compute optical flow from frame to frame.
- Magnify optical flow velocities.
- Appropriately warp image intensities.

In practice, many additional steps are required for a good result.

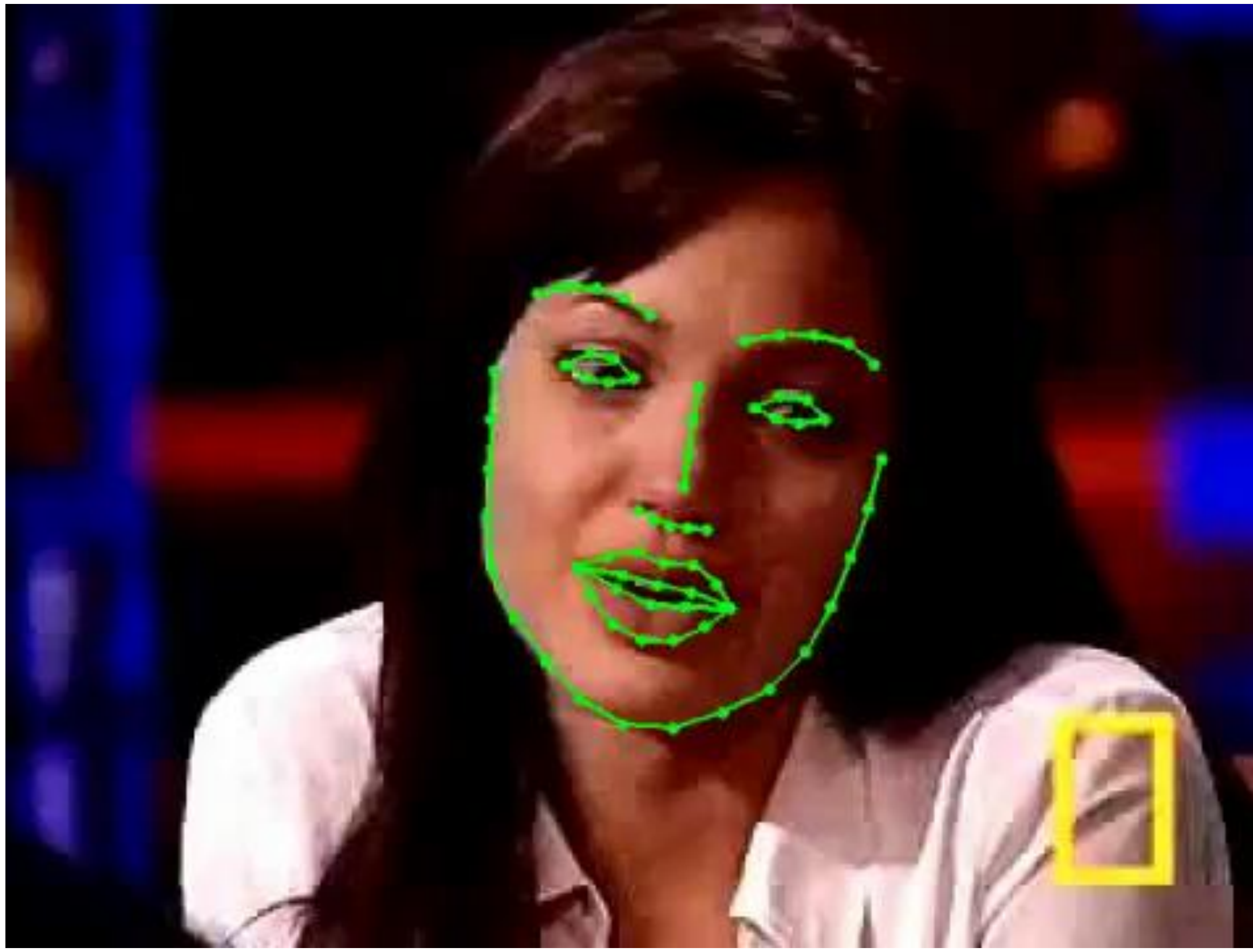
Some more examples

Some more examples

Image alignment



16
11
27
26
28
22
23
17
2
25
34
24
29
3
33
35
31
32
18
37
38
40
5
14
42
19
50
Average Face



IntraFace

http://www.humansensing.cs.cmu.edu/intraface/


31 fps

How can I find  in the image?

Idea #1: Template Matching

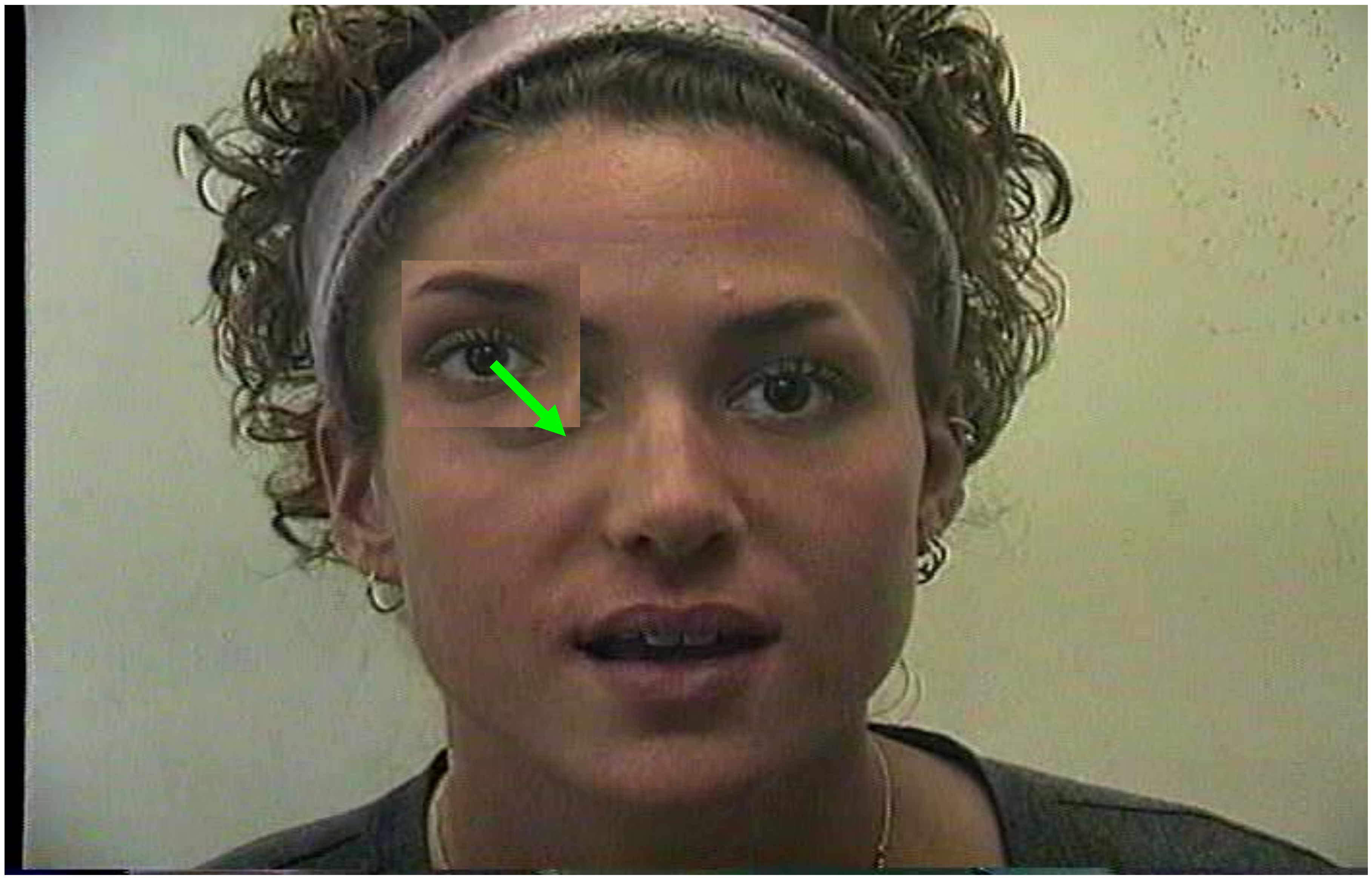
Slow, combinatory, global solution

Idea #2: Pyramid Template Matching

Faster, combinatory, locally optimal

Idea #3: Model refinement

(when you have a good initial solution)



Fastest, locally optimal

Some notation before we get into the math…

2D image transformation

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p})$$

2D image coordinate

$$\boldsymbol{x} = \left[\begin{array}{c} x \\ y \end{array}\right]$$

Parameters of the transformation

$$\boldsymbol{p} = \{p_1, \ldots, p_N\}$$

Warped image

$$I(\boldsymbol{x}') = I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))$$

Pixel value at a coordinate

Translation

Affine

Some notation before we get into the math…

2D image transformation

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p})$$

2D image coordinate

$$\boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$

Parameters of the transformation

$$\boldsymbol{p} = \{p_1, \ldots, p_N\}$$

Warped image

$$I(\boldsymbol{x}') = I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))$$

Pixel value at a coordinate

Translation

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) = \begin{bmatrix} x + p_1 \\ y + p_2 \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} 1 & 0 & p_1 \\ 0 & 1 & p_2 \end{bmatrix}}_{\text{transform}} \underbrace{\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}_{\text{coordinate}}$$

Affine

Some notation before we get into the math…

2D image transformation

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p})$$

2D image coordinate

$$\boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$

Parameters of the transformation

$$\boldsymbol{p} = \{p_1, \ldots, p_N\}$$

Warped image

$$I(\boldsymbol{x}') = I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))$$

Pixel value at a coordinate

Translation

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) = \begin{bmatrix} x + p_1 \\ y + p_2 \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} 1 & 0 & p_1 \\ 0 & 1 & p_2 \end{bmatrix}}_{\text{transform}} \underbrace{\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}_{\text{coordinate}}$$

Affine

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) = \begin{bmatrix} p_1 x + p_2 y + p_3 \\ p_4 x + p_5 y + p_6 \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} p_1 & p_2 & p_3 \\ p_4 & p_5 & p_6 \end{bmatrix}}_{\text{affine transform}} \underbrace{\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}_{\text{coordinate}}$$

can be written in matrix form when linear

affine warp matrix can also be 3x3 when last row is [0 0 1]

$\mathbf{W}(\boldsymbol{x};\boldsymbol{p})$ takes a ________ as input and returns a _______

$\mathbf{W}(\boldsymbol{x};\boldsymbol{p})$ is a function of ____ variables

$\mathbf{W}(\boldsymbol{x};\boldsymbol{p})$ returns a ______ of dimension ___ x ___

$\boldsymbol{p} = \{p_1, \ldots, p_N\}$ where N is _____ for an affine model

$I(\boldsymbol{x}') = I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))$ this warp changes pixel values?

Image alignment
(problem definition)

$$\min_{\boldsymbol{p}} \sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

warped image

template image

Find the warp parameters **p** such that the SSD is minimized

Find the warp parameters **p** such that the SSD is minimized

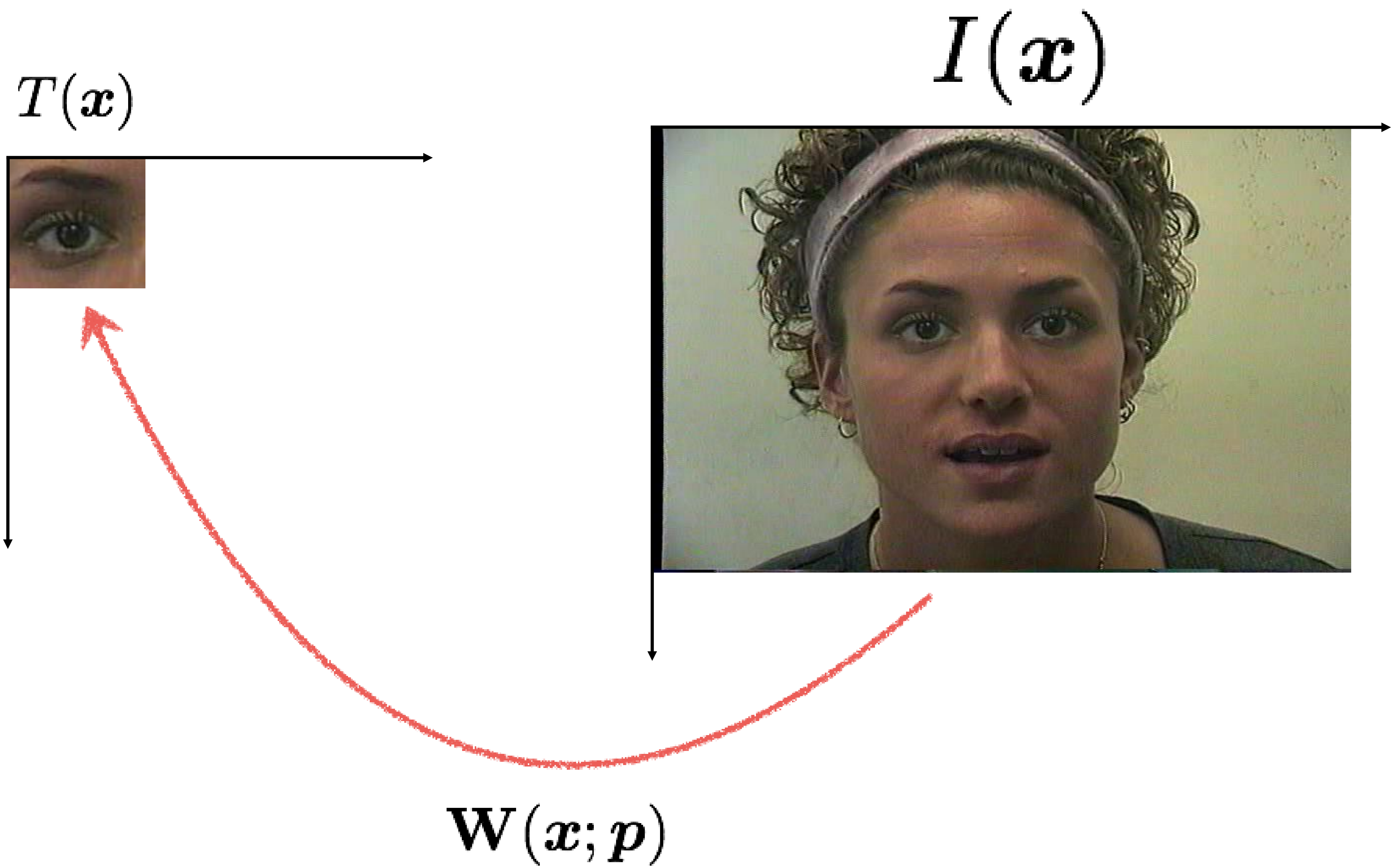


Image alignment
(problem definition)

$$\min_{\boldsymbol{p}} \sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

warped image template image

Find the warp parameters **p** such that the SSD is minimized

How could you find a solution to this problem?

This is a non-linear (quadratic) function of a non-parametric function!

(Function $\boldsymbol{I}$ is non-parametric)

$$\min_{\boldsymbol{p}} \sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

Hard to optimize

What can you do to make it easier to solve?

This is a non-linear (quadratic) function of a non-parametric function!

(Function $\boldsymbol{I}$ is non-parametric)

$$\min_{\boldsymbol{p}} \sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

Hard to optimize

What can you do to make it easier to solve?

assume good initialization,
linearized objective and update incrementally

Lucas-Kanade alignment

(pretty strong assumption)

If you have a good initial guess ***p***…

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

can be written as …

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p} + \Delta\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

(a small incremental adjustment)
(this is what we are solving for now)

This is **still** a non-linear (quadratic) function of a non-parametric function!

(Function $\boldsymbol{I}$ is non-parametric)

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

How can we linearize the function $\boldsymbol{I}$ for a really small perturbation of ***p****?*

This is **still** a non-linear (quadratic) function of a non-parametric function!

(Function $\boldsymbol{I}$ is non-parametric)

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

How can we linearize the function $\boldsymbol{I}$ for a really small perturbation of ***p****?*

Taylor series approximation!

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x}; \boldsymbol{p} + \Delta \boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

Multivariable Taylor Series Expansion
(First order approximation)

$$f(x,y) \approx f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b)$$

Multivariable Taylor Series Expansion
(First order approximation)

$$f(x,y) \approx f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b)$$

Recall: $\boldsymbol{x}' = \mathbf{W}(\boldsymbol{x};\boldsymbol{p})$

$$I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p})) \approx I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \frac{\partial I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))}{\partial \boldsymbol{p}}\Delta\boldsymbol{p}$$

chain rule
$$= I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \frac{\partial I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))}{\partial \boldsymbol{x}'}\frac{\partial \mathbf{W}(\boldsymbol{x};\boldsymbol{p})}{\partial \boldsymbol{p}}\Delta\boldsymbol{p}$$

short-hand
$$= I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I\frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\Delta\boldsymbol{p}$$

short-hand

$$\sum_{\boldsymbol{x}} [\underline{I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p}))} - T(\boldsymbol{x})]^2$$

Multivariable Taylor Series Expansion
(First order approximation)

$$f(x,y) \approx f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b)$$

Linear approximation

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta\boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

What are the unknowns here?

$$\sum_{\boldsymbol{x}} [\underline{I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p}))} - T(\boldsymbol{x})]^2$$

Multivariable Taylor Series Expansion
(First order approximation)

$$f(x,y) \approx f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b)$$

Linear approximation

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

Now, the function is a linear function of the unknowns

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

$\boldsymbol{x}$ is a __________ of dimension ___ x ___

output of $\mathbf{W}$ is a __________ of dimension ___ x ___

$\boldsymbol{p}$ is a ___________ of dimension ___ x ___

$I(\cdot)$ is a function of _____ variables

The Jacobian $\frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}$

(A matrix of partial derivatives)

$$\boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\mathbf{W} = \begin{bmatrix} W_x(x, y) \\ W_y(x, y) \end{bmatrix}$$

$$\frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} = \begin{bmatrix} \frac{\partial W_x}{\partial p_1} & \frac{\partial W_x}{\partial p_2} & \cdots & \frac{\partial W_x}{\partial p_N} \\ \frac{\partial W_y}{\partial p_1} & \frac{\partial W_y}{\partial p_2} & \cdots & \frac{\partial W_y}{\partial p_N} \end{bmatrix}$$

Rate of change of the warp

Affine transform

$$\mathbf{W}(\boldsymbol{x}; \boldsymbol{p}) = \begin{bmatrix} p_1 x + p_3 y + p_5 \\ p_2 x + p_4 y + p_6 \end{bmatrix}$$

$$\frac{\partial W_x}{\partial p_1} = x \qquad \frac{\partial W_x}{\partial p_2} = 0 \qquad \cdots$$

$$\frac{\partial W_y}{\partial p_1} = 0 \qquad \cdots$$

$$\frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} = \begin{bmatrix} x & 0 & y & 0 & 1 & 0 \\ 0 & x & 0 & y & 0 & 1 \end{bmatrix}$$

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

∇I is a __________ of dimension ___ x ___

$\frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}$ is a __________ of dimension ___ x ___

$\Delta \boldsymbol{p}$ is a __________ of dimension ___ x ___

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

pixel coordinate
(2 x 1)

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

pixel coordinate
(2 x 1)

image intensity
(scalar)

warp function
(2 x 1)

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

pixel coordinate
(2 x 1)

image intensity
(scalar)

warp function
(2 x 1)

warp parameters
(6 for affine)

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

pixel coordinate
(2 x 1)

image intensity
(scalar)

warp function
(2 x 1)

warp parameters
(6 for affine)

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

image gradient
(1 x 2)

pixel coordinate
(2 x 1)

image intensity
(scalar)

Partial derivatives of warp function
(2 x 6)

warp function
(2 x 1)

warp parameters
(6 for affine)

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

image gradient
(1 x 2)

pixel coordinate
(2 x 1)

image intensity
(scalar)

Partial derivatives of warp function
(2 x 6)

warp function
(2 x 1)

warp parameters
(6 for affine)

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

image gradient
(1 x 2)

incremental warp
(6 x 1)

pixel coordinate
(2 x 1)

image intensity
(scalar)

Partial derivatives of warp function
(2 x 6)

warp function
(2 x 1)

template image intensity
(scalar)

warp parameters
(6 for affine)

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

image gradient
(1 x 2)

incremental warp
(6 x 1)

pixel coordinate
(2 x 1)

image intensity
(scalar)

When you implement this, you will compute everything in parallel and store as matrix … don't loop over x!

Summary

(of Lucas-Kanade Image Alignment)

Problem:

$$\min_{\boldsymbol{p}} \sum_{\boldsymbol{x}} [I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x})]^2$$

warped image template image

Difficult non-linear optimization problem

Strategy:

$$\sum_{\boldsymbol{x}} [I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p} + \Delta\boldsymbol{p})) - T(\boldsymbol{x})]^2$$

Assume known approximate solution

Solve for increment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta\boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

Taylor series approximation

Linearize

then solve for $\Delta\boldsymbol{p}$

OK, so how do we solve this?

$$\min_{\Delta \boldsymbol{p}} \sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

Another way to look at it…

$$\min_{\Delta \boldsymbol{p}} \sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

(moving terms around)

$$\min_{\Delta \boldsymbol{p}} \sum_{\boldsymbol{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - \{ T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) \} \right]^2$$

vector of constants

vector of variables

constant

Have you seen this form of optimization problem before?

Another way to look at it…

$$\min_{\Delta \boldsymbol{p}} \sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

$$\min_{\Delta \boldsymbol{p}} \sum_{\boldsymbol{x}} \left[\underbrace{\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}}_{\text{constant}} \underbrace{\Delta \boldsymbol{p}}_{\text{variable}} - \underbrace{\{T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))\}}_{\text{constant}} \right]^2$$

Looks like $\mathbf{A}\mathbf{x} - \mathbf{b}$

How do you solve this?

Least squares approximation

$$\hat{x} = \arg\min_{x} ||Ax - b||^2$$ is solved by $$x = (A^\top A)^{-1} A^\top b$$

Applied to our tasks:

$$\min_{\Delta \boldsymbol{p}} \sum_{\boldsymbol{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta \boldsymbol{p} - \{ T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x}; \boldsymbol{p})) \} \right]^2$$

is optimized when

$$\Delta \boldsymbol{p} = H^{-1} \sum_{\boldsymbol{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \right]^\top \left[T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x}; \boldsymbol{p})) \right]$$

after applying

$x = (A^\top A)^{-1} A^\top b$

where $$H = \sum_{\boldsymbol{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \right]^\top \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \right]$$

$A^\top A$

Solve:

$$\min_{\boldsymbol{p}} \sum_{\boldsymbol{x}} [I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x})]^2$$

warped image template image

Difficult non-linear optimization problem

Strategy:

$$\sum_{\boldsymbol{x}} [I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p})) - T(\boldsymbol{x})]^2$$

Assume known approximate solution
Solve for increment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta\boldsymbol{p} - T(\boldsymbol{x})\right]^2$$

Taylor series approximation
Linearize

Solution:

$$\Delta\boldsymbol{p} = H^{-1} \sum_{\boldsymbol{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]^\top [T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]$$

Solution to least squares approximation

$$H = \sum_{\boldsymbol{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]^\top \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]$$

Hessian

This is called…

Gauss-Newton gradient decent non-linear optimization!

Lucas Kanade (Additive alignment)

1. Warp image $I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))$
2. Compute error image $[T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]$
3. Compute gradient $\nabla I(\boldsymbol{x}')$ x'coordinates of the warped image (gradients of the warped image)
4. Evaluate Jacobian $\frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}$
5. Compute Hessian H $\quad H = \sum_{\boldsymbol{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]^{\top} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]$
6. Compute $\Delta \boldsymbol{p}$ $\quad \Delta \boldsymbol{p} = H^{-1} \sum_{\boldsymbol{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]^{\top} [T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]$
7. Update parameters $\boldsymbol{p} \leftarrow \boldsymbol{p} + \Delta \boldsymbol{p}$

Just 8 lines of code!

Baker-Matthews alignment

Image Alignment

(start with an initial solution, match the image and template)

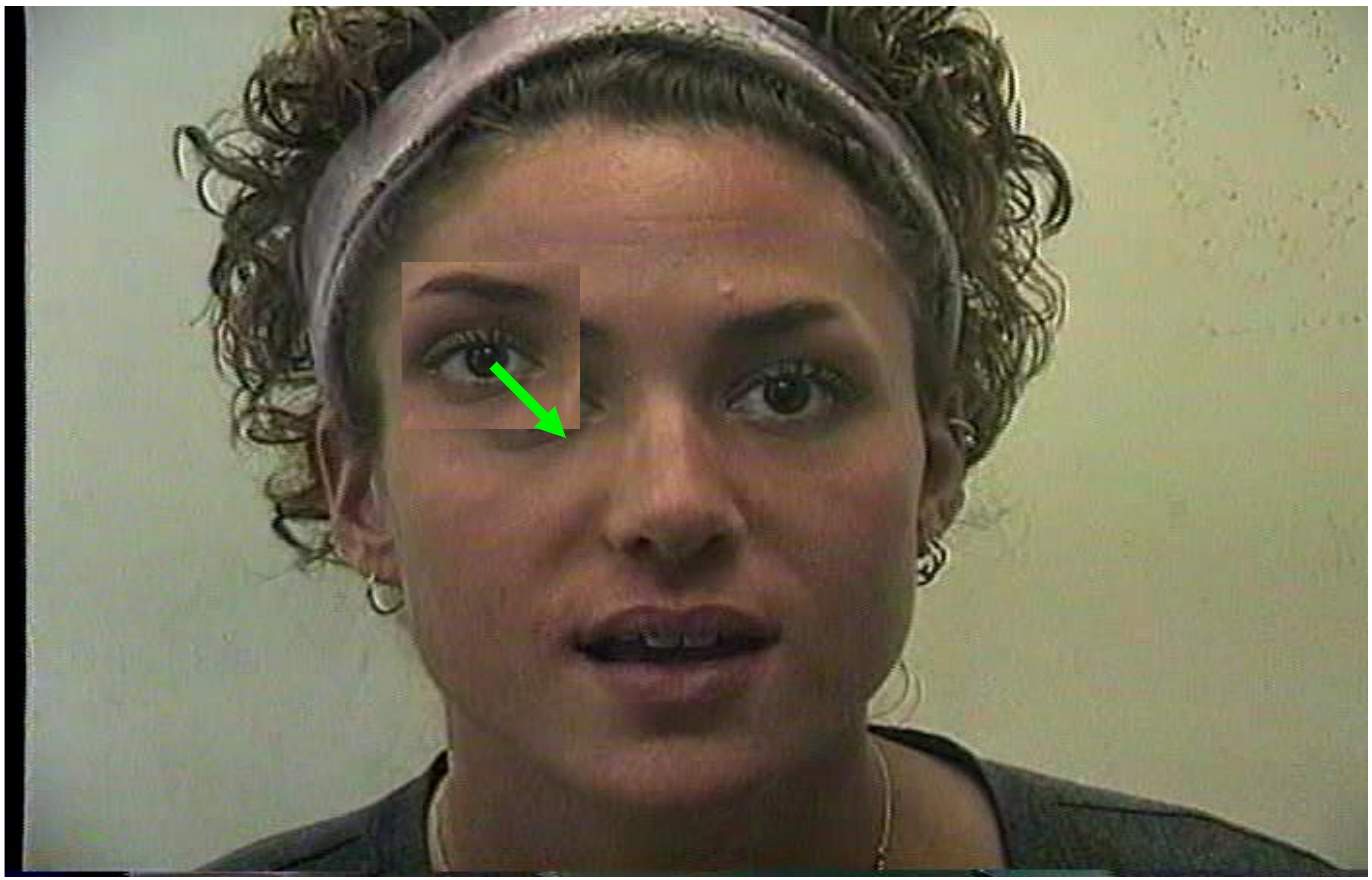


Image Alignment Objective Function

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

Given an initial solution…several possible formulations

Additive Alignment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

incremental perturbation of parameters

Image Alignment Objective Function

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

Given an initial solution…several possible formulations

Additive Alignment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

incremental perturbation of <u>parameters</u>

Compositional Alignment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p});\boldsymbol{p}\ ) - T(\boldsymbol{x}) \right]^2$$

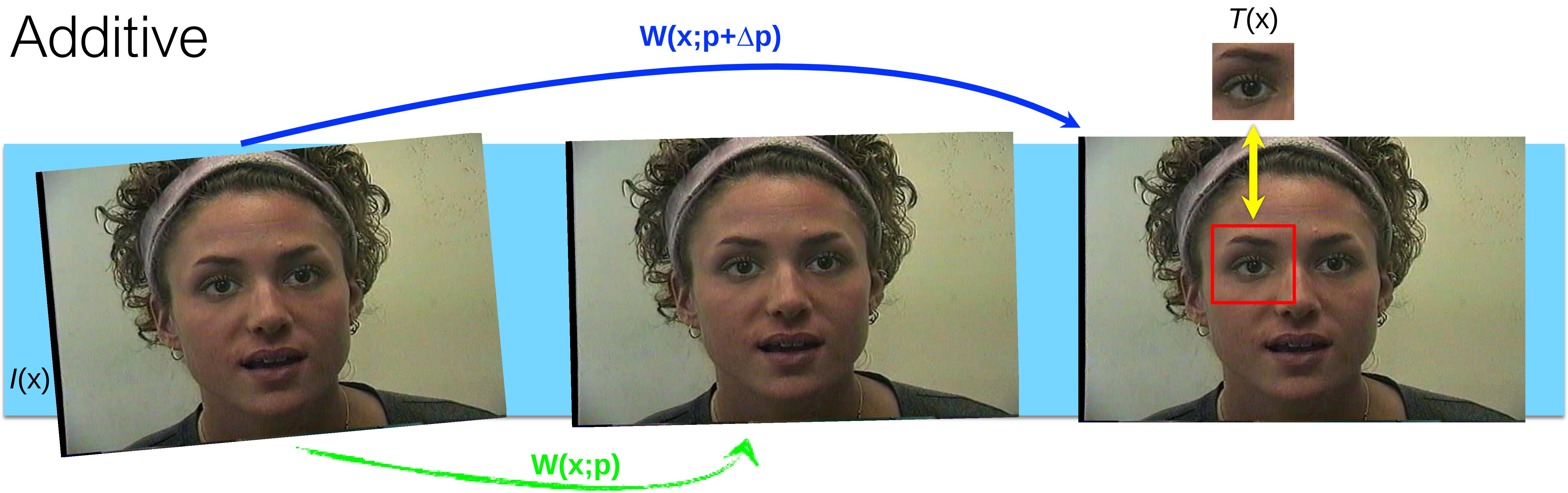
incremental <u>warps</u> of image

Additive strategy

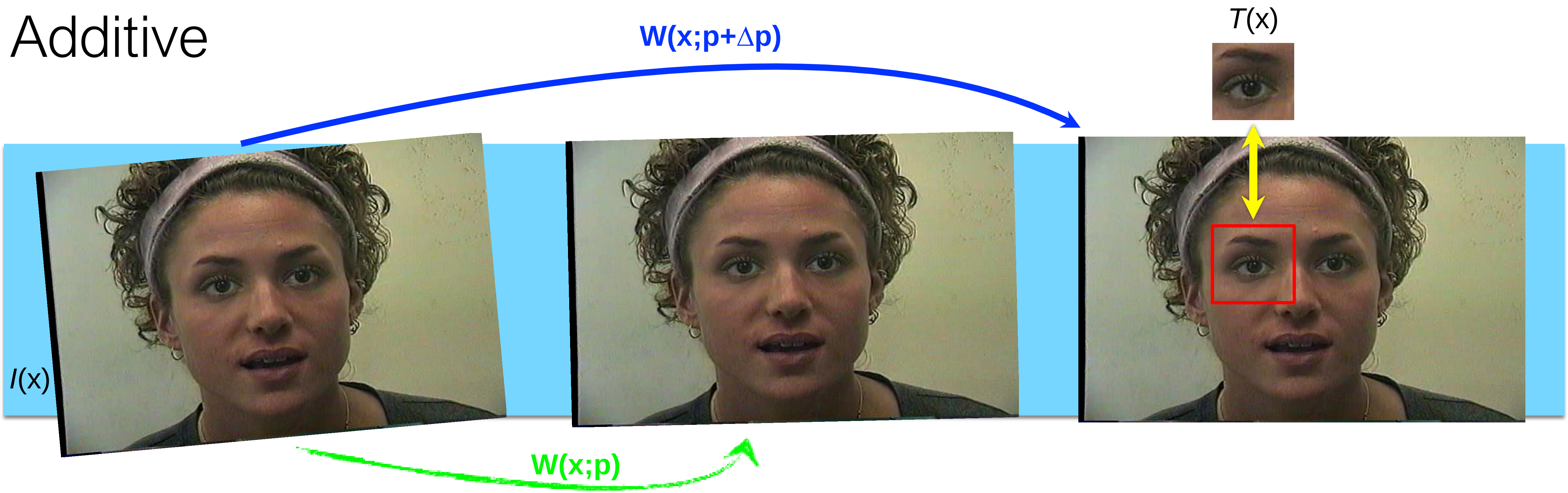
Compositional strategy

Additive

Additive

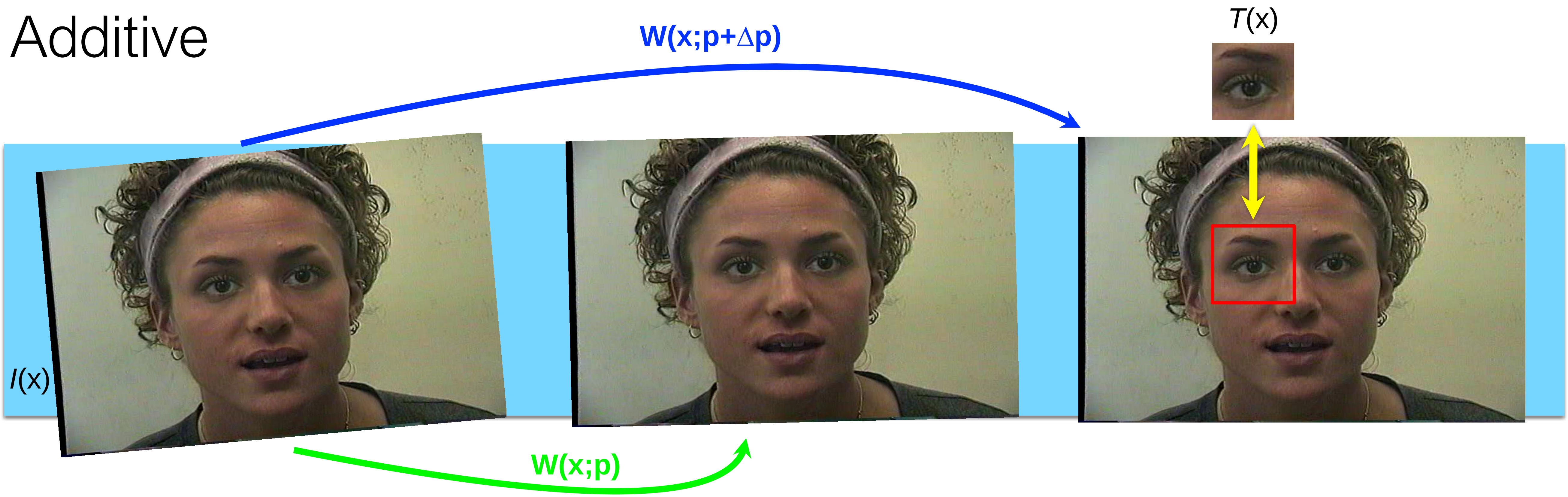


Additive

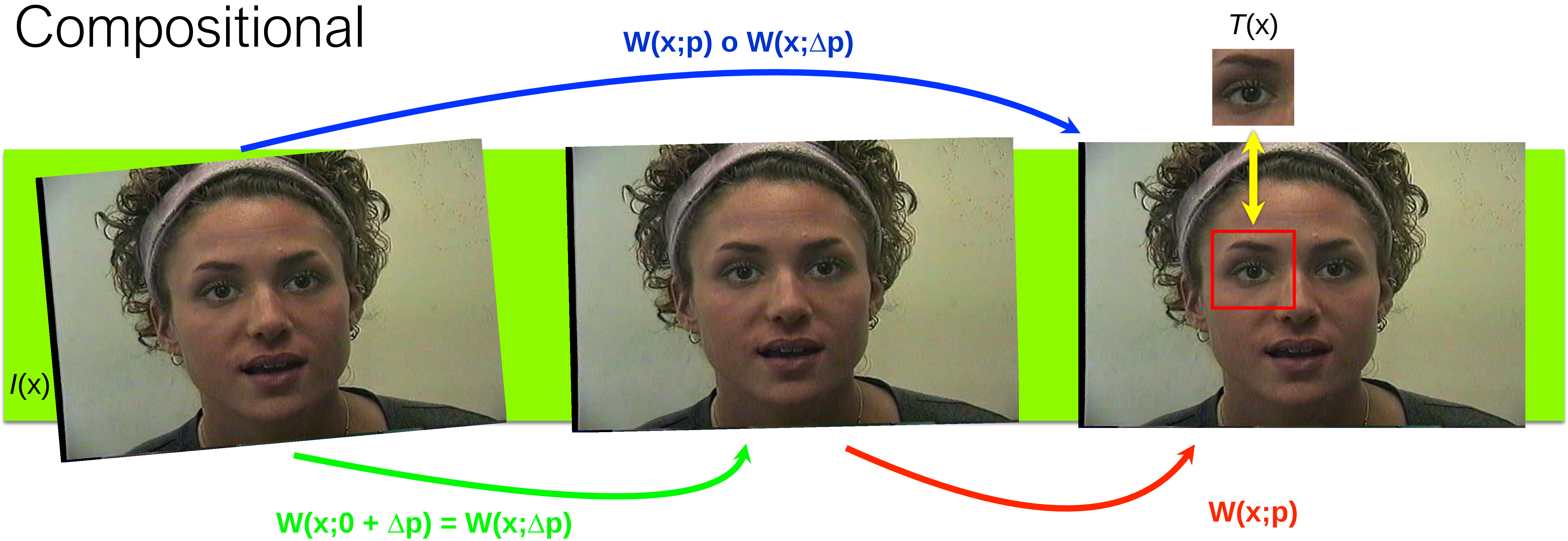


Compositional

Additive



Compositional



Compositional Alignment

Original objective function (SSD)

$$\min_{\boldsymbol{p}} \sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

Assuming an initial solution **p** and a compositional warp increment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p});\boldsymbol{p}\ ) - T(\boldsymbol{x}) \right]^2$$

Compositional Alignment

Original objective function (SSD)

$$\min_{\boldsymbol{p}} \sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

Assuming an initial solution **p** and a compositional warp increment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p});\boldsymbol{p} \ ) - T(\boldsymbol{x}) \right]^2$$

Another way to write the composition

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \circ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p}) \equiv \mathbf{W}(\ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p});\boldsymbol{p} \ )$$

Identity warp

$$\mathbf{W}(\boldsymbol{x};\mathbf{0})$$

Compositional Alignment

Original objective function (SSD)

$$\min_{\boldsymbol{p}} \sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

Assuming an initial solution **p** and a compositional warp increment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p});\boldsymbol{p}\ ) - T(\boldsymbol{x}) \right]^2$$

Another way to write the composition

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \circ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p}) \equiv \mathbf{W}(\ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p});\boldsymbol{p}\ )$$

Identity warp

$$\mathbf{W}(\boldsymbol{x};\mathbf{0})$$

Skipping over the derivation…the new update rule is

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \leftarrow \mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \circ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p})$$

So what’s so great about this compositional form?

Additive Alignment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

linearized form

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I(\boldsymbol{x}') \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta\boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

Compositional Alignment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p});\boldsymbol{p}\ ) - T(\boldsymbol{x}) \right]^2$$

linearized form

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I(\boldsymbol{x}') \frac{\partial \mathbf{W}(\boldsymbol{x};\mathbf{0})}{\partial \boldsymbol{p}} \Delta\boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

Additive Alignment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p})) - T(\boldsymbol{x}) \right]^2$$

Compositional Alignment

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p});\boldsymbol{p}\ ) - T(\boldsymbol{x}) \right]^2$$

linearized form

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I(\boldsymbol{x}')\frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\Delta\boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

Jacobian of W(x;p)

linearized form

$$\sum_{\boldsymbol{x}} \left[I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})) + \nabla I(\boldsymbol{x}')\frac{\partial \mathbf{W}(\boldsymbol{x};\mathbf{0})}{\partial \boldsymbol{p}}\Delta\boldsymbol{p} - T(\boldsymbol{x}) \right]^2$$

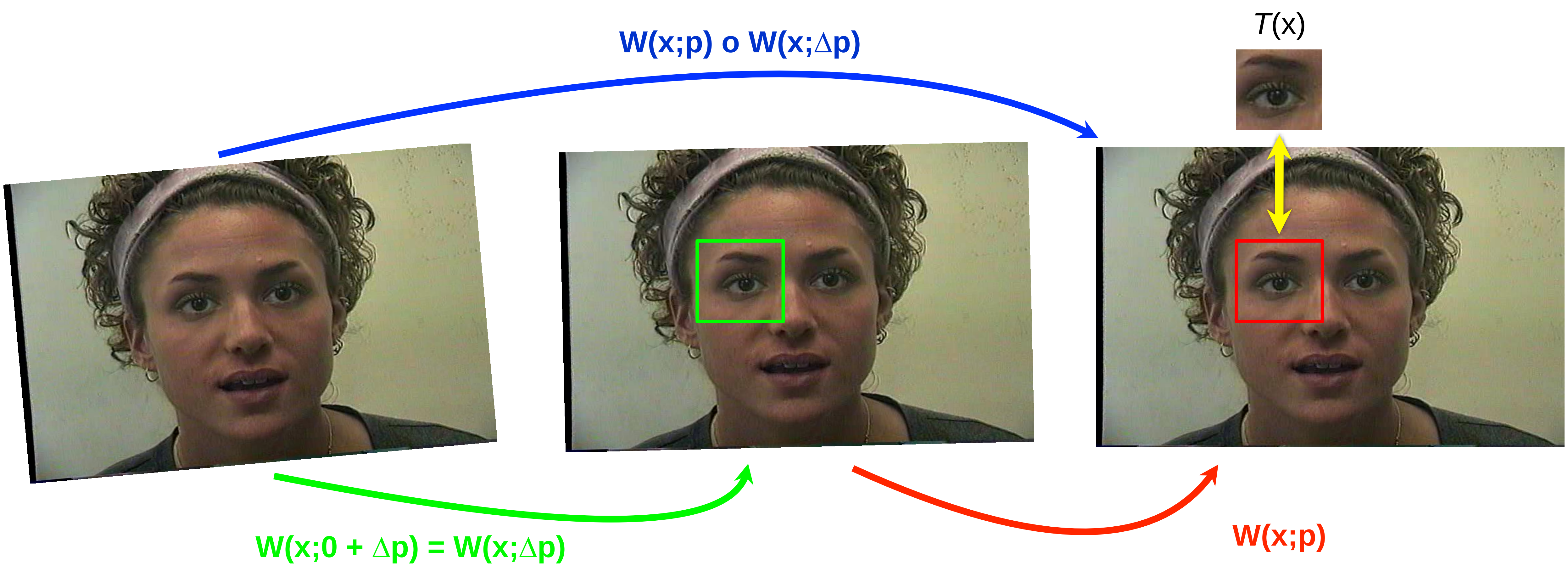
Jacobian of
W(x;0)

The Jacobian is constant.
Jacobian can be precomputed!

Compositional Image Alignment

Minimize

$$\sum_{\mathbf{x}} \left[I(\mathbf{W}(\mathbf{W}(\mathbf{x};\Delta\mathbf{p});\mathbf{p})) - T(\mathbf{x}) \right]^2 \approx \sum_{\mathbf{x}} \left[I(\mathbf{W}(\mathbf{x};\mathbf{p})) + \nabla I(\mathbf{W}) \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta\mathbf{p} - T(\mathbf{x}) \right]^2$$



Jacobian is simple and can be precomputed

Lucas Kanade (Additive alignment)

1. Warp image $I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))$

2. Compute error image $[T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]^2$

3. Compute gradient $\nabla I(\boldsymbol{x}')$

4. Evaluate Jacobian $\frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}$

5. Compute Hessian H

6. Compute $\Delta \boldsymbol{p}$

7. Update parameters $\boldsymbol{p} \leftarrow \boldsymbol{p} + \Delta \boldsymbol{p}$

Shum-Szeliski (Compositional alignment)

1. Warp image $I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))$
2. Compute error image $[T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]^2$
3. Compute gradient $\nabla I(\boldsymbol{x}')$
4. Evaluate Jacobian $\frac{\partial \mathbf{W}(\boldsymbol{x};\mathbf{0})}{\partial \boldsymbol{p}}$
5. Compute Hessian H
6. Compute $\Delta \boldsymbol{p}$
7. Update parameters $\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \leftarrow \mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \circ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p})$

Any other speed up techniques?

Inverse alignment

Why not compute warp updates on the template?

Additive Alignment

$$\sum_{\boldsymbol{x}} [I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p})) - T(\boldsymbol{x})]^2$$

Compositional Alignment

$$\sum_{\boldsymbol{x}} [I(\mathbf{W}(\ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p});\boldsymbol{p}\ ) - T(\boldsymbol{x})]^2$$

Why not compute warp updates on the template?

Additive Alignment

$$\sum_{\boldsymbol{x}} [I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}+\Delta\boldsymbol{p})) - T(\boldsymbol{x})]^2$$

Compositional Alignment

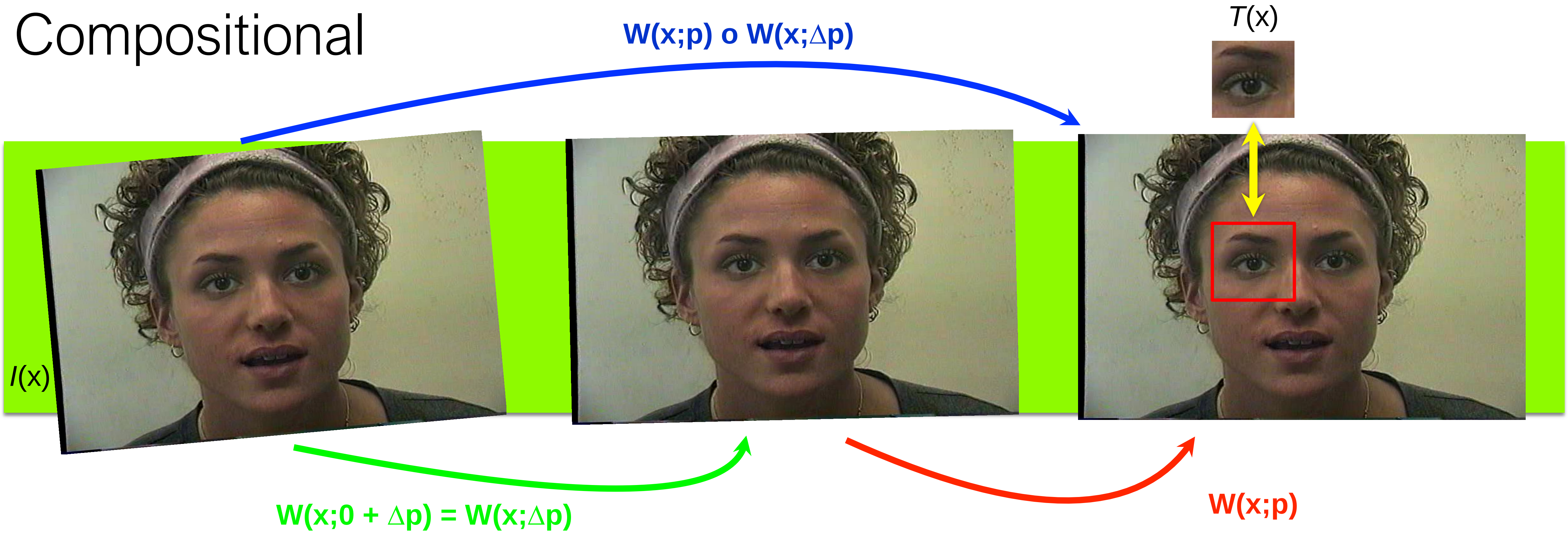
$$\sum_{\boldsymbol{x}} [I(\mathbf{W}(\ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p});\boldsymbol{p}\ ) - T(\boldsymbol{x})]^2$$

What happens if you let the template be warped too?

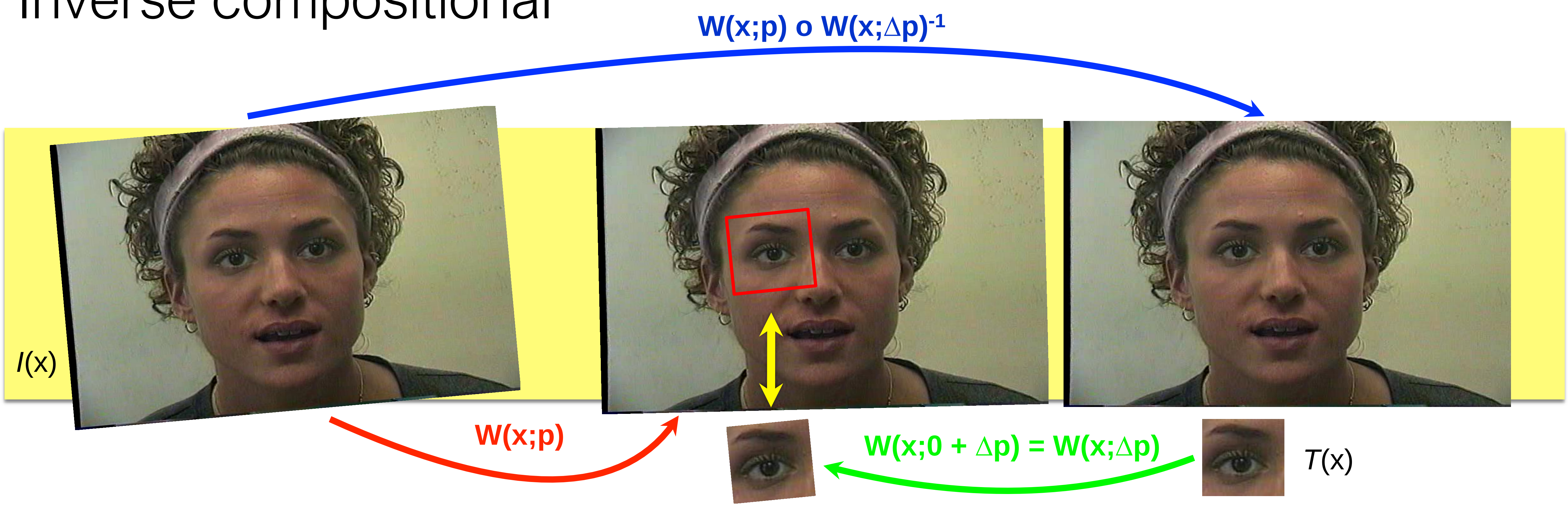
Inverse Compositional Alignment

$$\sum_{\boldsymbol{x}} [T(\mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]^2$$

Compositional



Inverse compositional



Compositional strategy

Inverse Compositional strategy

So what's so great about this inverse compositional form?

Inverse Compositional Alignment

Minimize

$$\sum_{\boldsymbol{x}} [T(\mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]^2 \approx \sum \boldsymbol{x} \left[T(\mathbf{W}(\boldsymbol{x};\mathbf{0})) + \nabla T \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \Delta\boldsymbol{p} - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p})\right]^2$$

Solution

$$H = \sum_{\boldsymbol{x}} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]^\top \left[\nabla T \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]$$

can be precomputed from template!

$$\Delta\boldsymbol{p} = \sum_{\boldsymbol{x}} H^{-1} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]^\top [T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]$$

Update

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \leftarrow \mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \circ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p})^{-1}$$

Properties of inverse compositional alignment

Jacobian can be precomputed

It is constant - evaluated at W(x;0)

Gradient of template can be precomputed

It is constant

Hessian can be precomputed

$$\boxed{H = \sum_{\boldsymbol{x}} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \right]^{\top} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \right]}$$

$$\Delta \boldsymbol{p} = \boxed{\sum_{\boldsymbol{x}} H^{-1} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \right]^{\top}} [T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x}; \boldsymbol{p}))]$$

(main term that needs to be computed)

Warp must be invertible

Lucas Kanade (Additive alignment)

1. Warp image $I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))$
2. Compute error image $[T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]^2$
3. Compute gradient $\nabla I(\mathbf{W})$
4. Evaluate Jacobian $\frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}$
5. Compute Hessian H
6. Compute $\Delta \boldsymbol{p}$
7. Update parameters $\boldsymbol{p} \leftarrow \boldsymbol{p} + \Delta \boldsymbol{p}$

Shum-Szeliski (Compositional alignment)

1. Warp image $I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))$
2. Compute error image $[T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]$
3. Compute gradient $\nabla I(\boldsymbol{x}')$
4. Evaluate Jacobian $\frac{\partial \mathbf{W}(\boldsymbol{x};\mathbf{0})}{\partial \boldsymbol{p}}$
5. Compute Hessian H
6. Compute $\Delta \boldsymbol{p}$
7. Update parameters $\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \leftarrow \mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \circ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p})$

Baker-Matthews (Inverse Compositional alignment)

1. Warp image $I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))$
2. Compute error image $[T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]$
3. Compute gradient $\nabla T(\mathbf{W})$
4. Evaluate Jacobian $\frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}$
5. Compute Hessian H $\quad H = \sum_{\boldsymbol{x}} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]^{\top} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]$
6. Compute $\Delta \boldsymbol{p}$ $\quad \Delta \boldsymbol{p} = \sum_{\boldsymbol{x}} H^{-1} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}}\right]^{\top} [T(\boldsymbol{x}) - I(\mathbf{W}(\boldsymbol{x};\boldsymbol{p}))]$
7. Update parameters $\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \leftarrow \mathbf{W}(\boldsymbol{x};\boldsymbol{p}) \circ \mathbf{W}(\boldsymbol{x};\Delta\boldsymbol{p})^{-1}$

Algorithm	Efficient	Authors
Forwards Additive	No	Lucas, Kanade
Forwards compositional	No	Shum, Szeliski
Inverse Additive	Yes	Hager, Belhumeur
Inverse Compositional	Yes	Baker, Matthews

References

Basic reading:

- Szeliski, Section 8.1.