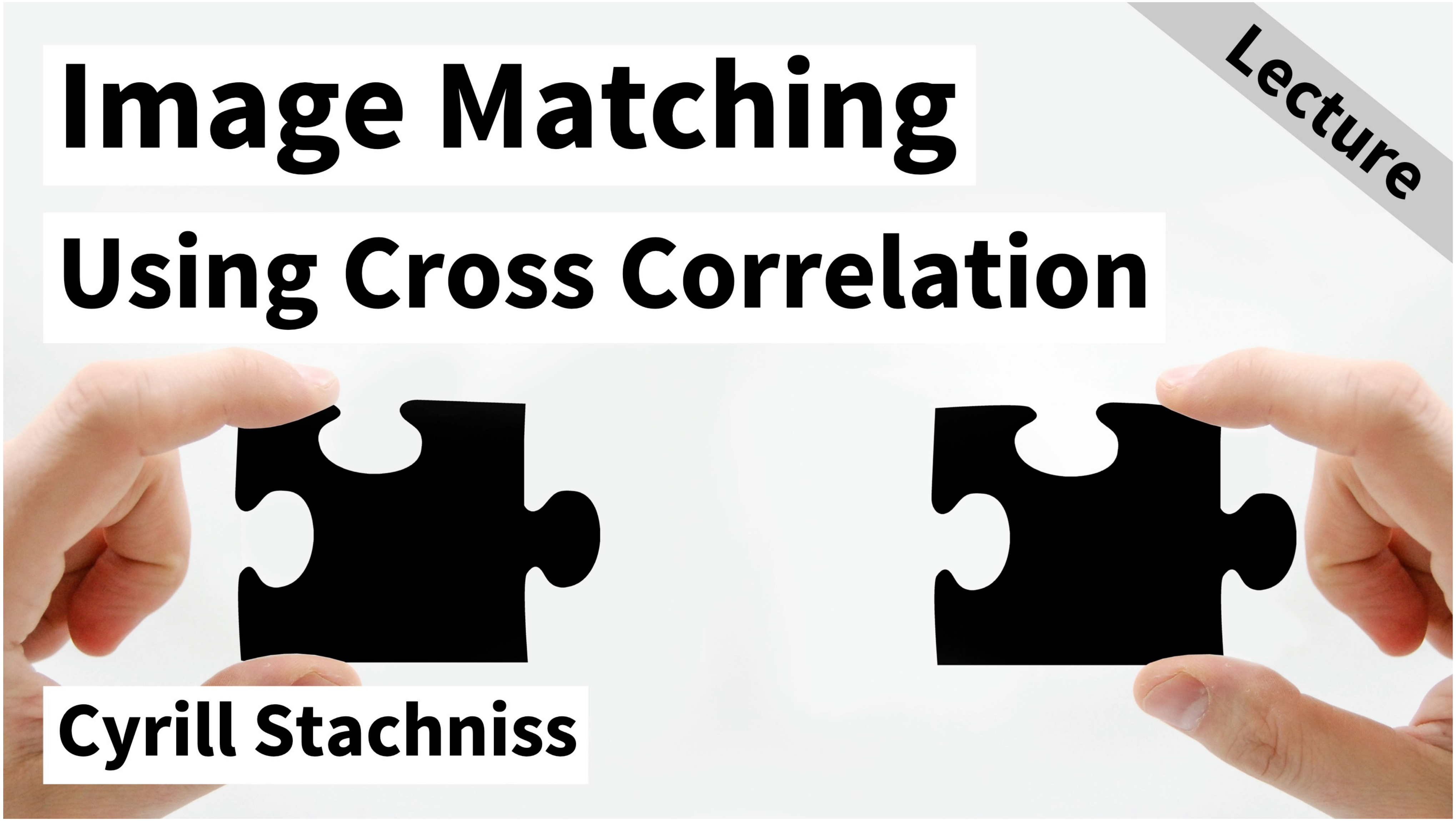


Summer term 2024 – Cyrill Stachniss

Photogrammetry & Robotics Lab

Image Template Matching Using Cross Correlation

Cyrill Stachniss

The slides have been created by Cyrill Stachniss.

Example: Image Alignment Using Corresponding Points

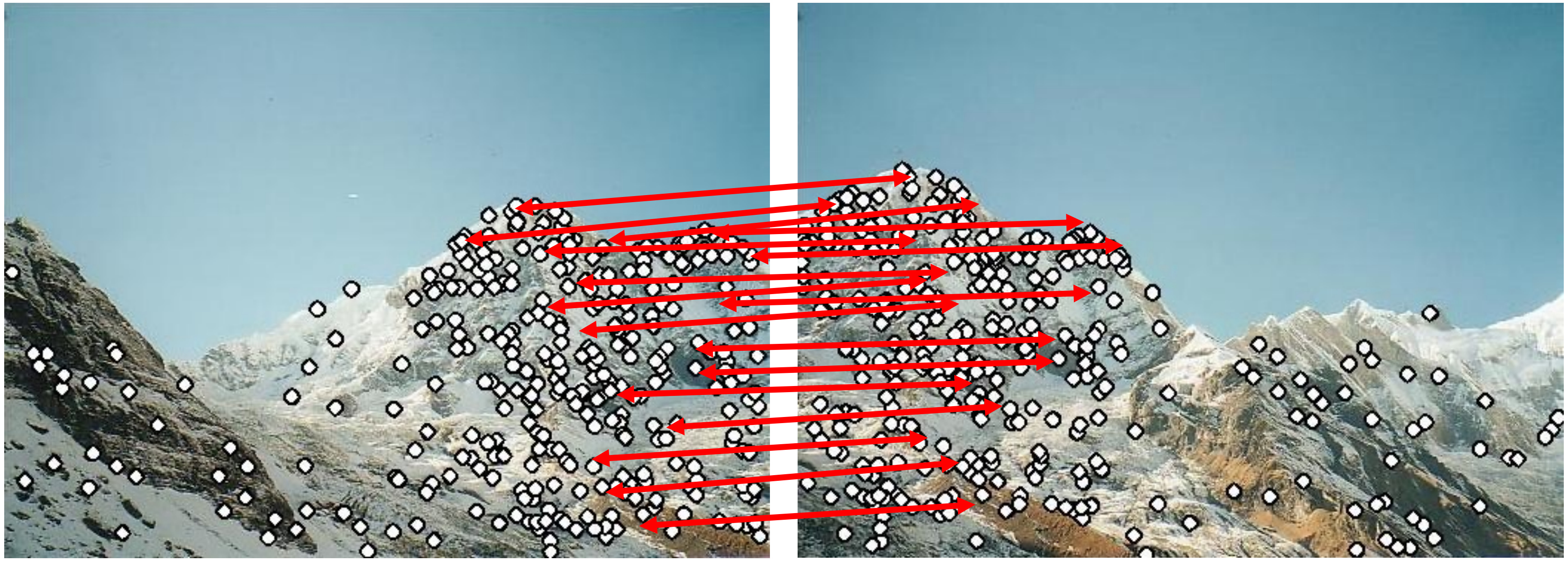


Image courtesy: Efros

Example: Image Alignment Using Corresponding Points

Image courtesy: Efros

Estimating 3D Information

Given corresponding points and the orientation of the cameras, we can compute the point locations in 3D

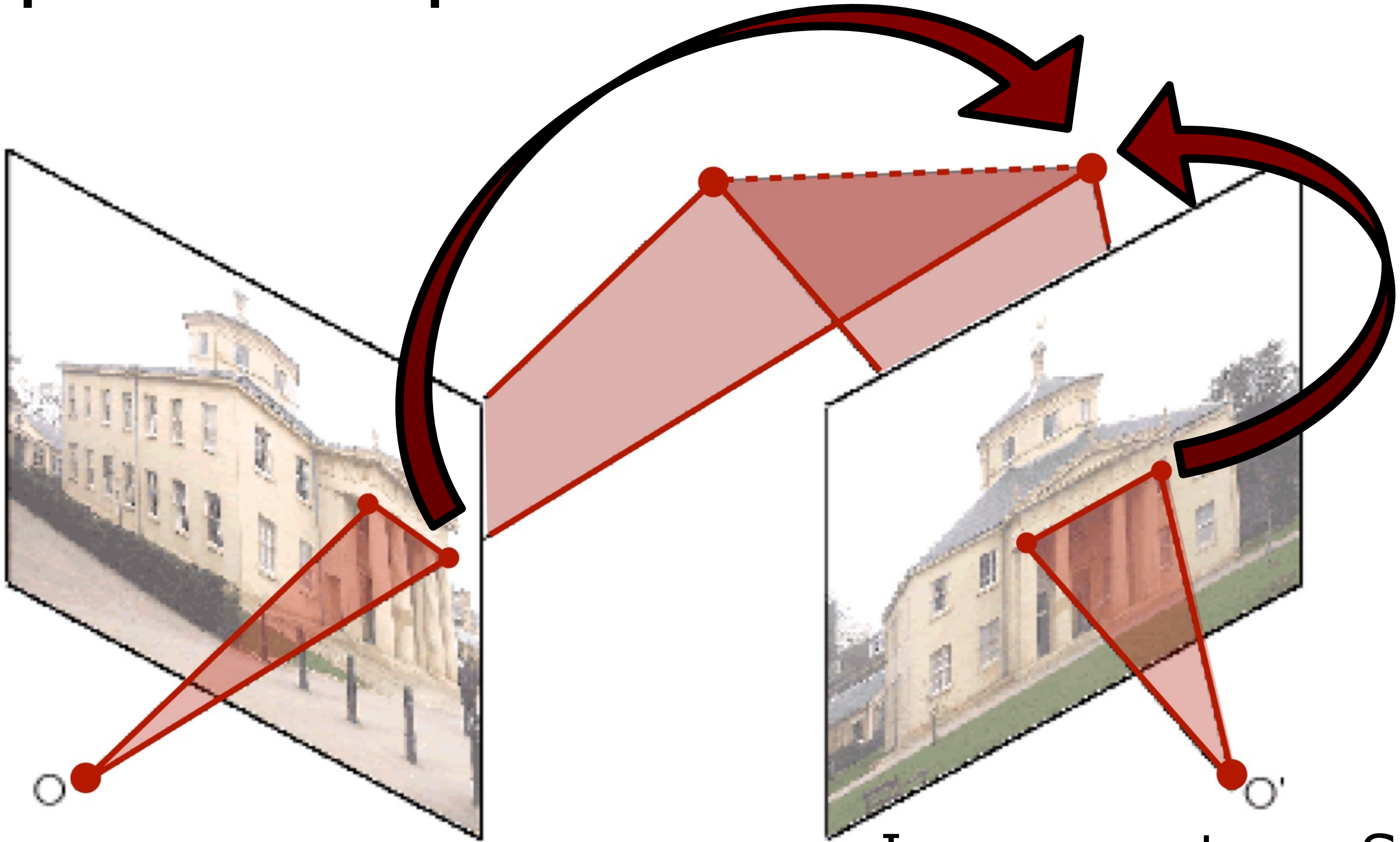


Image courtesy: Schindler

Data Association

- If we know the corresponding points, (and the orientation of cameras), we can perform a 3D reconstruction
- For stereo image matching, images are taken under similar conditions

Question: Can we localize a local image patch in another image?

Data Association

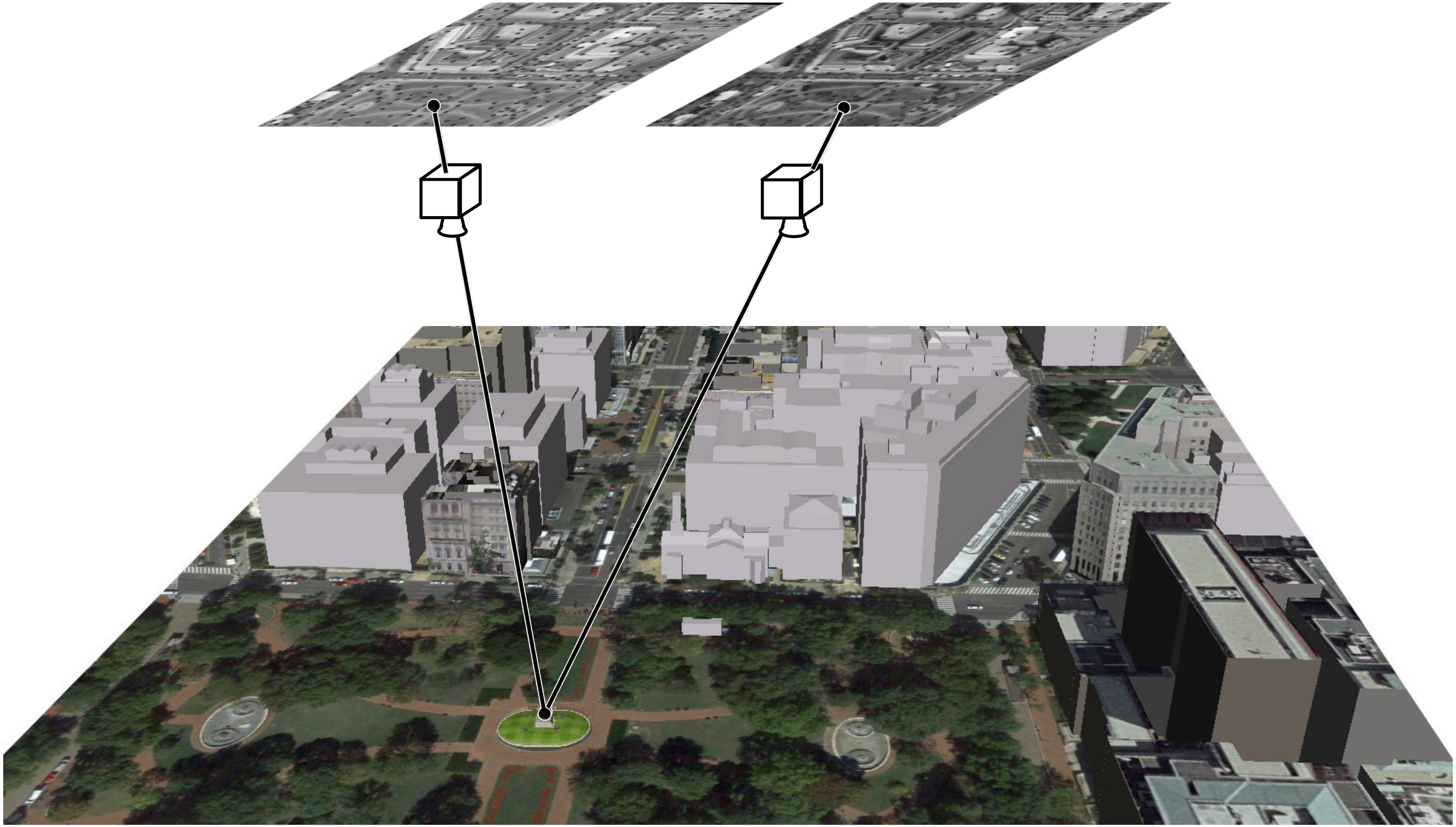
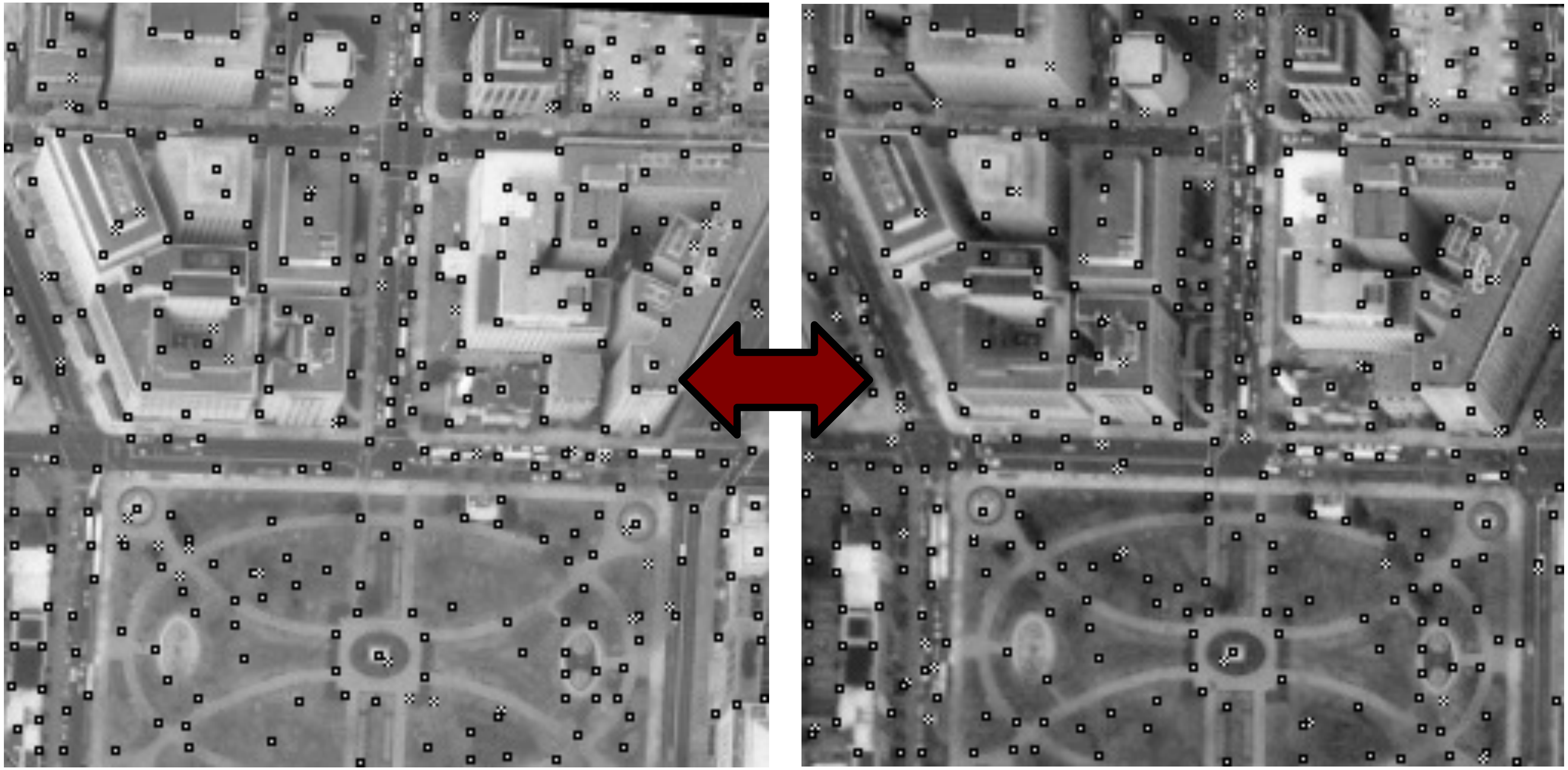


Image courtesy: Förstner, Wenzel

Data Association



How to know which parts of both images correspond to each other?

Image courtesy: Förstner

Cross Correlation

DE: “Kreuzkorrelation”

Cross Correlation (CC)

Cross correlation is a powerful tool to:

- Find certain image content in an image
- Determine its location in the image

Key assumption: Images differ only by

- Translation
- Brightness
- Contrast

Template Matching

- Find the location of a small template image within a (larger) image
- Usually: size of template << size of image

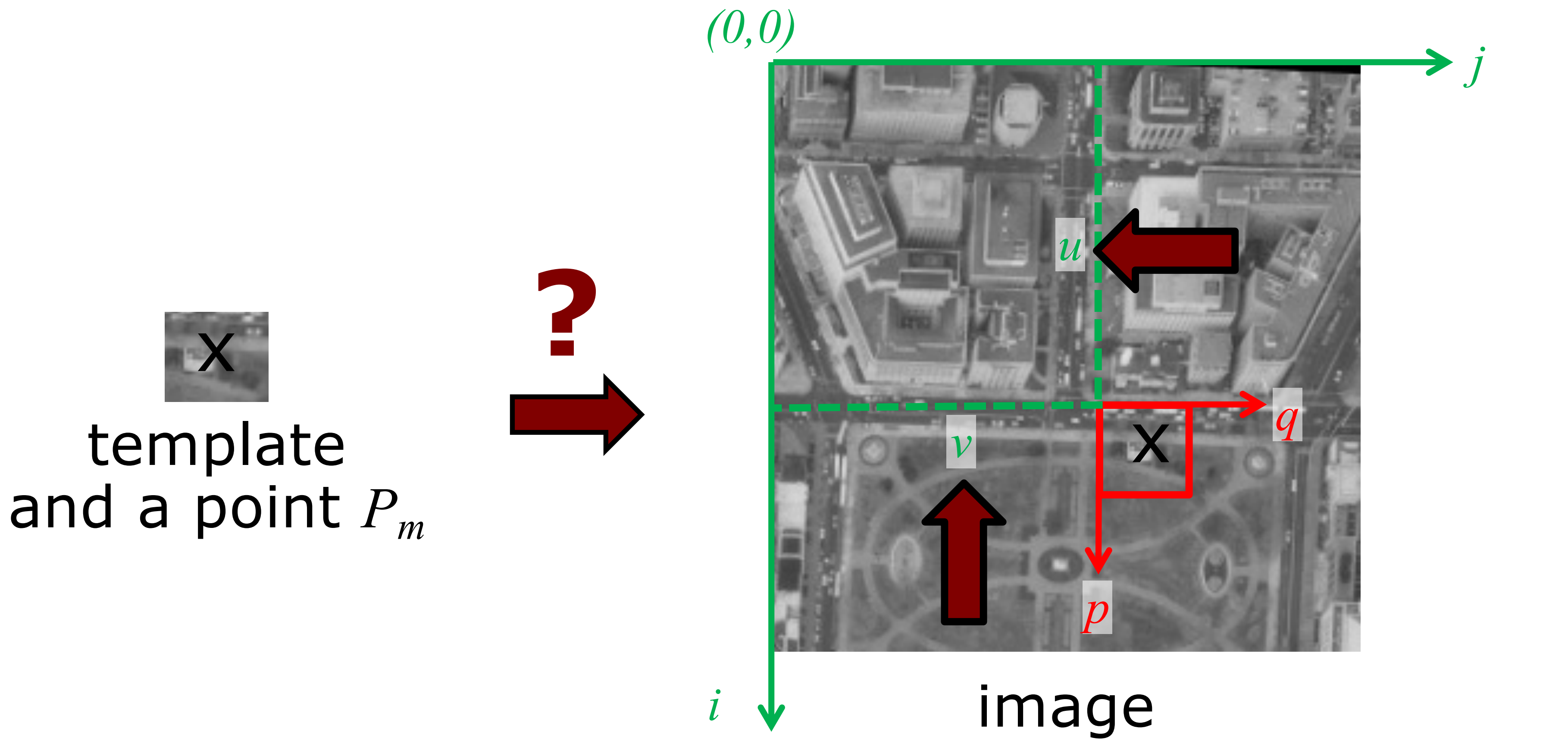
template

?

image

Template Matching

- Find the location of a small template image within a (larger) image
- Usually: size of template << size of image



Principle

- Given image $g_1(i, j)$ and template $g_2(p, q)$
- Find offset $[\hat{u}, \hat{v}]$ between g_1 and g_2

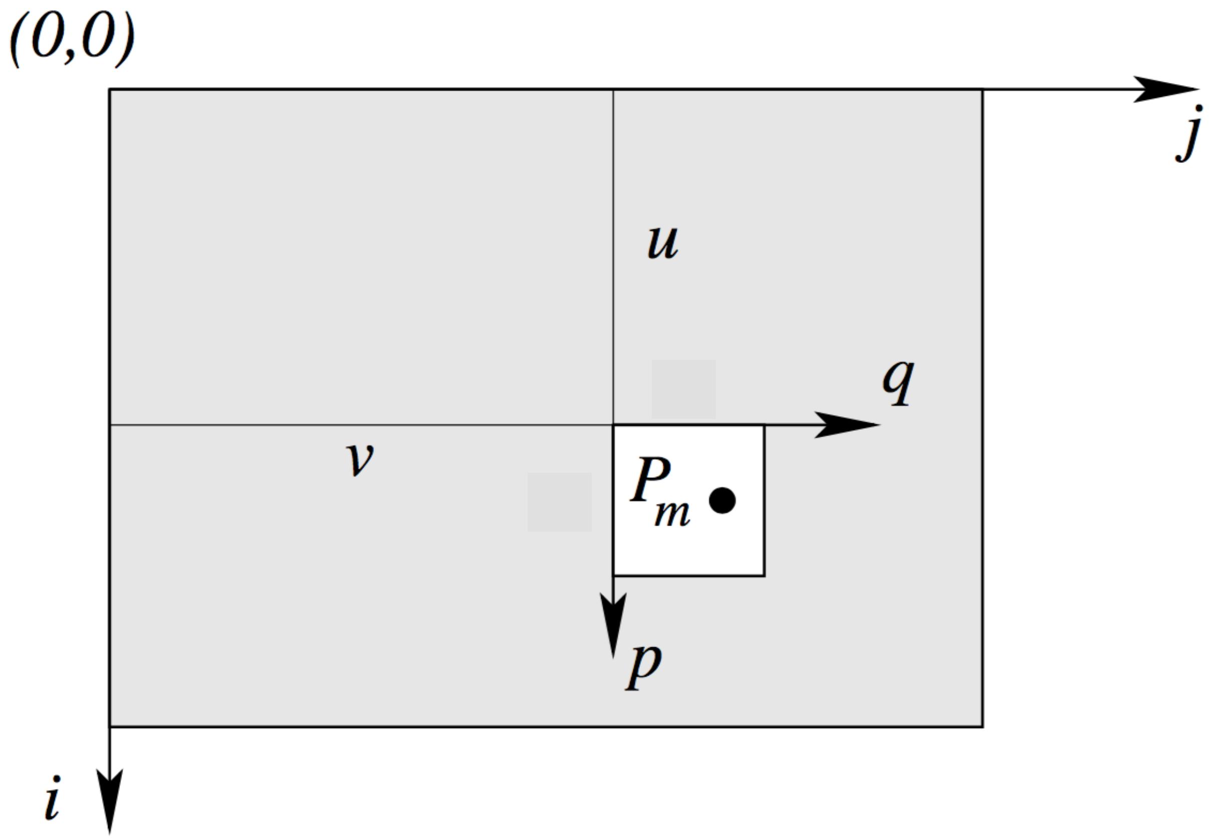


Image courtesy: Förstner

Assumptions

- Geometric transformation

$$T_G : \begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} i \\ j \end{bmatrix} - \begin{bmatrix} u \\ v \end{bmatrix} \longleftarrow$$ **translation only**

Two unknowns $p_G = [u, v]^\mathsf{T}$

Assumptions

- Geometric transformation

$$T_G : \begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} i \\ j \end{bmatrix} - \begin{bmatrix} u \\ v \end{bmatrix}$$

translation only

Two unknowns $p_G = [u, v]^\mathsf{T}$

- Radiometric transformation

brightness

$$T_I : g_2(p, q) = a + b\, g_1(i, j)$$

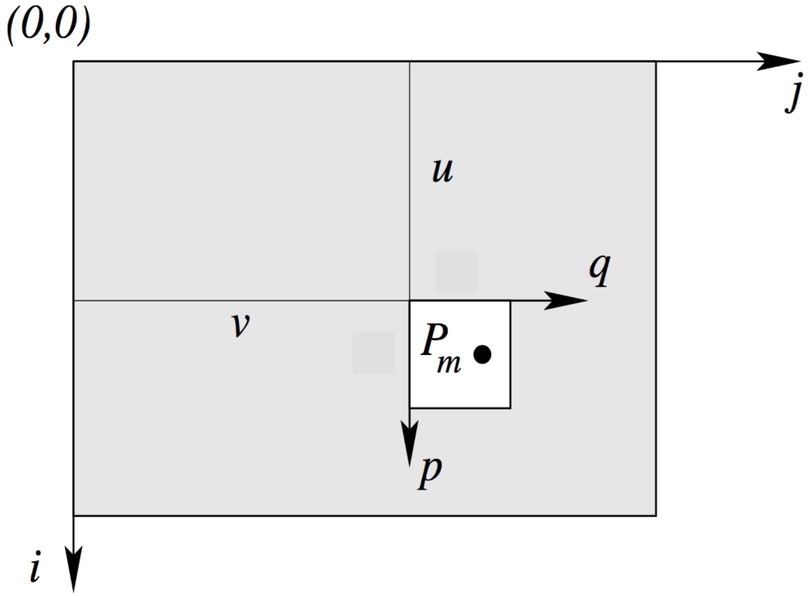
contrast

  - Intensities of each pixel in g_2 are linearly dependent of those of g_1
  - Two additional unknowns $p_R = [a, b]^\mathsf{T}$

Problem Definition

$$\begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} i \\ j \end{bmatrix} - \begin{bmatrix} u \\ v \end{bmatrix}$$

$$g_2(p, q) = a + b\, g_1(i, j)$$



Task: Find the offset $[\hat{u}, \hat{v}]$ that maximizes the similarities of the corresponding intensity values

How to quantify "similarity"?

Typical Measures of Similarity

- Sum of squared differences (SSD)

$$\mathrm{SSD} = \sum_m \left(g_2(m) - g_1(m)\right)^2$$

- Sum of absolute differences (SAD)

$$\mathrm{SAD} = \sum_m \left|g_2(m) - g_1(m)\right|$$

- Maximum of differences

$$\mathrm{Max} = \max_m \left|g_2(m) - g_1(m)\right|$$

No invariance against changes in brightness and contrast!

Cross Correlation Function

Best estimate of the offset $[\hat{u}, \hat{v}]$ is given by maximizing the **cross correlation coefficient** over all possible locations

$$[\hat{u}, \hat{v}] = \mathrm{argmax}_{u,v} \rho_{12}(u, v)$$

$$\rho_{12}(u, v) = \frac{\sigma_{g_1 g_2}(u, v)}{\sigma_{g_1}(u, v) \sigma_{g_2}}$$

Normalized Cross Correlation

Product of the variations of intensities from mean in template and image

$$\rho_{12}(u,v) = \frac{\sigma_{g_1 g_2}(u,v)}{\sigma_{g_1}(u,v)\sigma_{g_2}}$$

Standard deviation of intensity values of the image in the area overlayed by template

Standard deviation of intensity values of the template

Normalized Cross Correlation

$$\rho_{12}(u,v) = \frac{\sigma_{g_1 g_2}(u,v)}{\sigma_{g_1}(u,v)\boxed{\sigma_{g_2}}}$$

Standard deviation of intensity values of template g_2

$$\sigma_{g_2}^2 = \underbrace{\frac{1}{M-1}}_{\text{Number of pixels in } g_2} \underbrace{\sum_{m=1}^{M}}_{\text{Sum over all rows and columns of the template } g_2} \left(g_2(p_m, q_m) - \underbrace{\frac{1}{M} \sum_{m=1}^{M} g_2(p_m, q_m)}_{\text{Mean of intensities in } g_2} \right)^2$$

Normalized Cross Correlation

$$\rho_{12}(u,v) = \frac{\sigma_{g_1 g_2}(u,v)}{\boxed{\sigma_{g_1}(u,v)}\sigma_{g_2}}$$

Standard deviation of intensity values of g_1 in the area overlapping with template g_2 given offset $[u,v]$

$$\sigma^2_{g_1}(u,v) = \frac{1}{M-1} \sum_{m=1}^{M} \left(g_1(p_m+u, q_m+v) - \frac{1}{M} \sum_{m=1}^{M} g_1(p_m+u, q_m+v) \right)^2$$

Number of pixels in g_2 !!

Sum over all rows and columns in overlap area with template g_2

Mean of intensity values in overlap area

Normalized Cross Correlation

$$\rho_{12}(u,v) = \frac{\boxed{\sigma_{g_1 g_2}(u,v)}}{\sigma_{g_1}(u,v)\sigma_{g_2}}$$

Covariance between intensity values of g_1 and in the overlap area with template g_2 given offset $[u,v]$

$$\begin{aligned}\sigma_{g_1 g_2}(u,v) \quad = \quad & \frac{1}{M-1}\sum_{m=1}^{M}\Bigg[\left(g_2(p_m,q_m) - \frac{1}{M}\sum_{m=1}^{M} g_2(p_m,q_m)\right)\cdot \\ & \left(g_1(p_m+u,q_m+v) - \frac{1}{M}\sum_{m=1}^{M} g_1(p_m+u,q_m+v)\right)\Bigg]\end{aligned}$$

Search Strategies

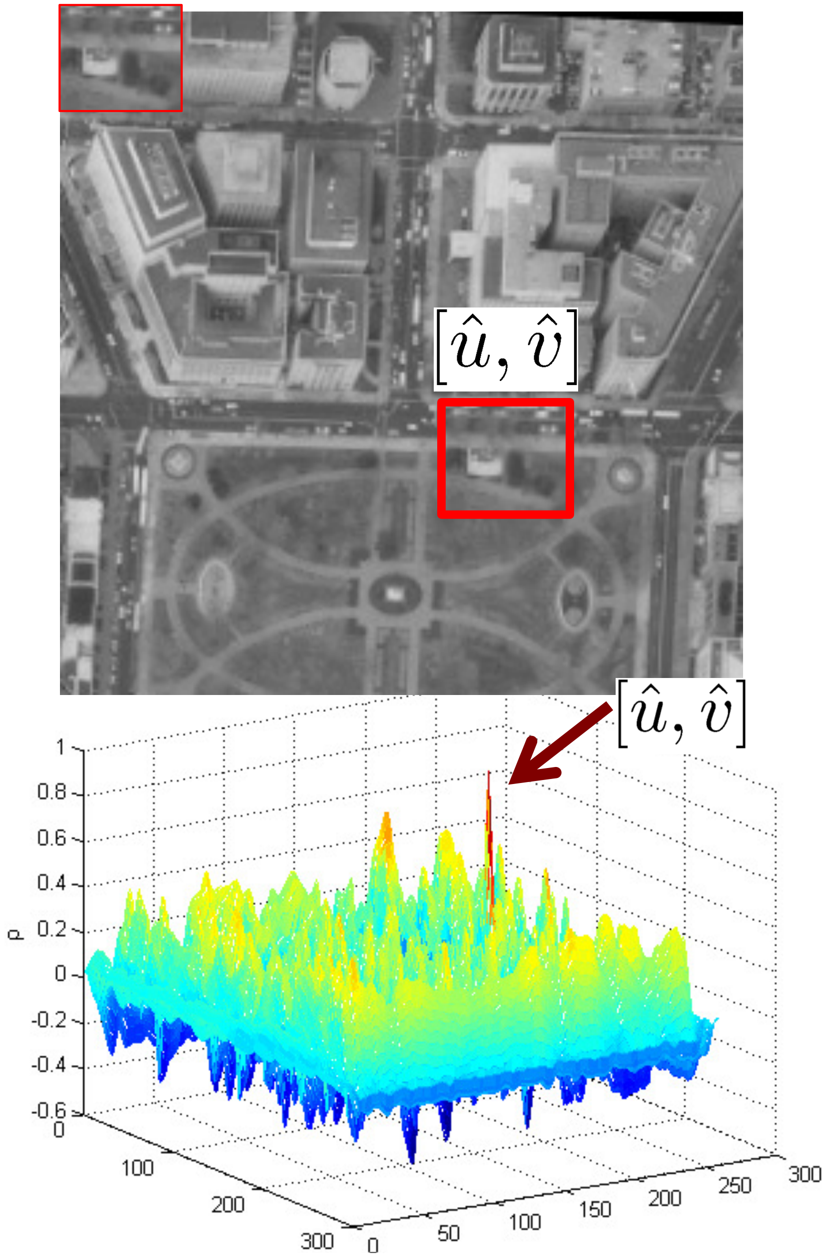
How to search the best position?

Image courtesy: Förstner

Exhaustive Search

- For all offsets $[u, v]$ compute $\rho(u, v)$

- Select offset $[u, v]$ for which $\rho(u, v)$ is maximized



Complexity

- Full search in the 2D translation parameters
- In theory we can also search for rotation and other parameters
- Complexity increases exponentially with the dimension of the search space

Dimension of search space

$$O(\prod_{d=1}^{D} r_d)$$

Number of possible locations in dimension d

Coarse-To-Fine Strategy Using an Image Pyramid

- Iteratively use resized image from large to small
- Start on top of the pyramid
- Match gives initialization for next level

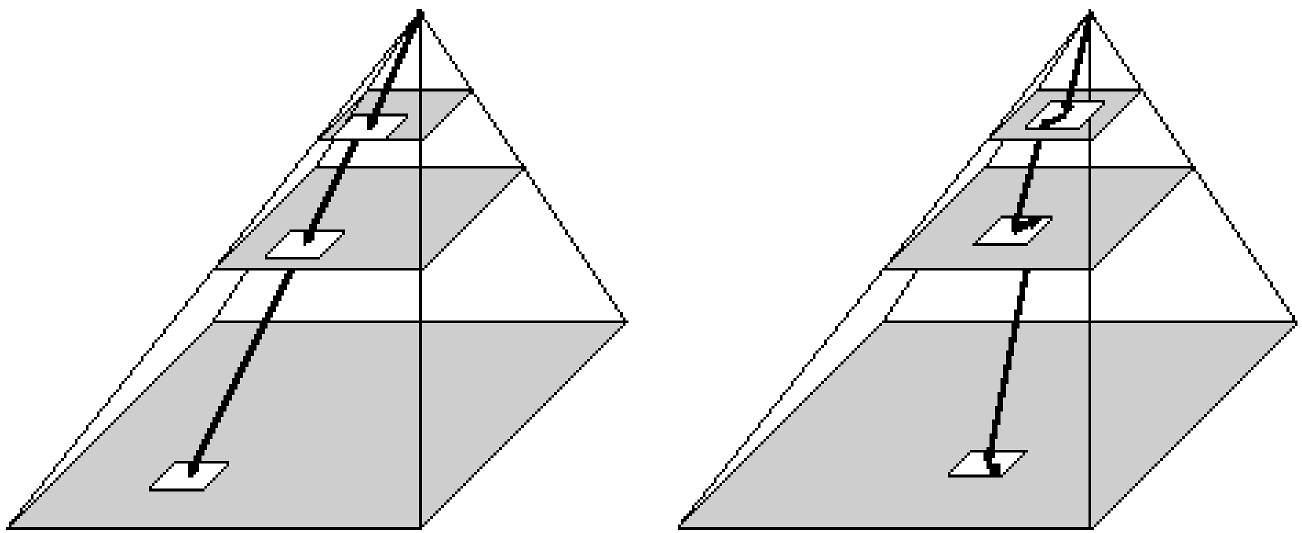
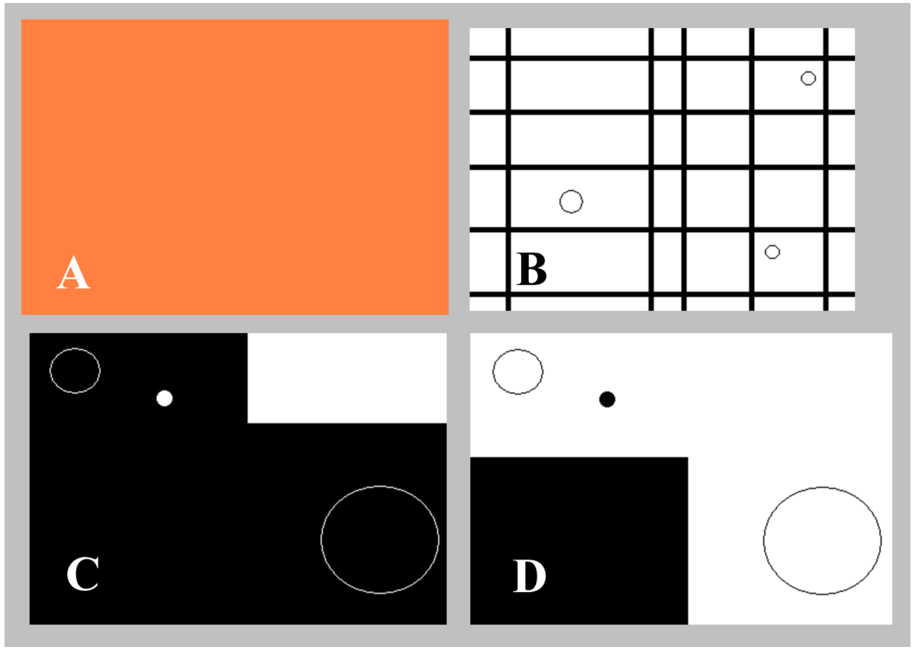
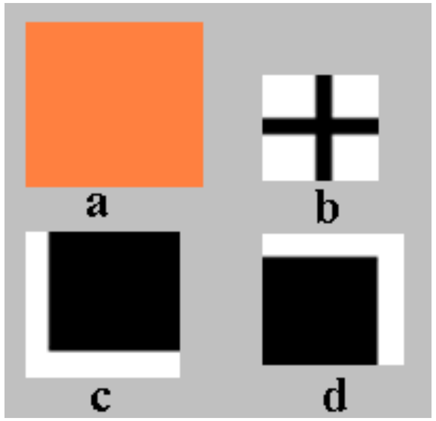


Image courtesy: Förstner

Example

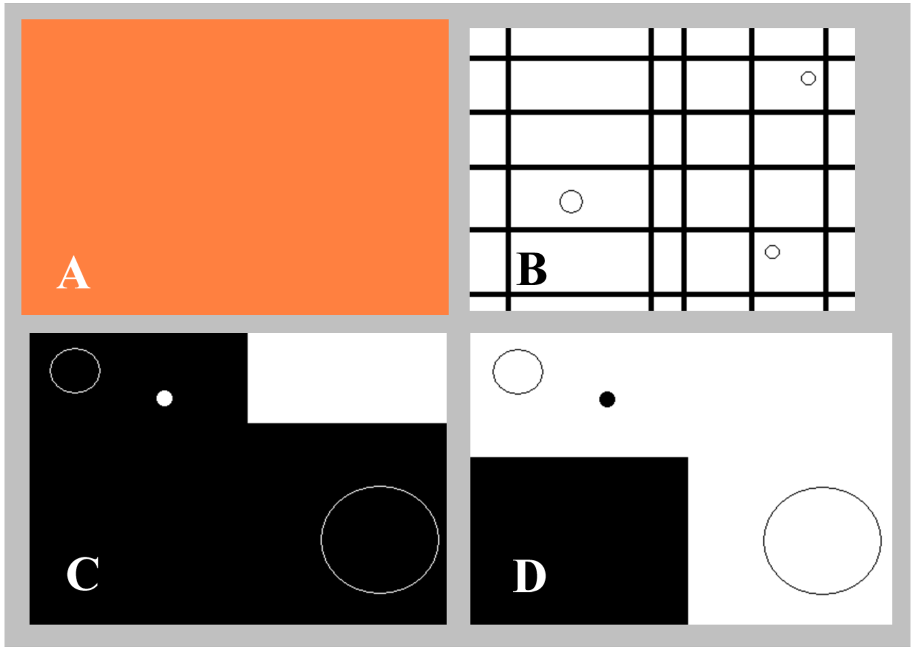


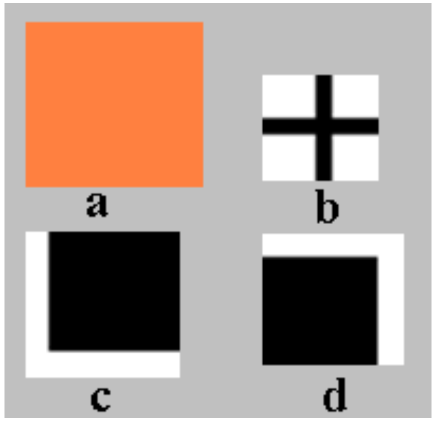


Cross correlation of

- a and A: ?
- b and B: ?
- c and C: ?
- d and D: ?

Example

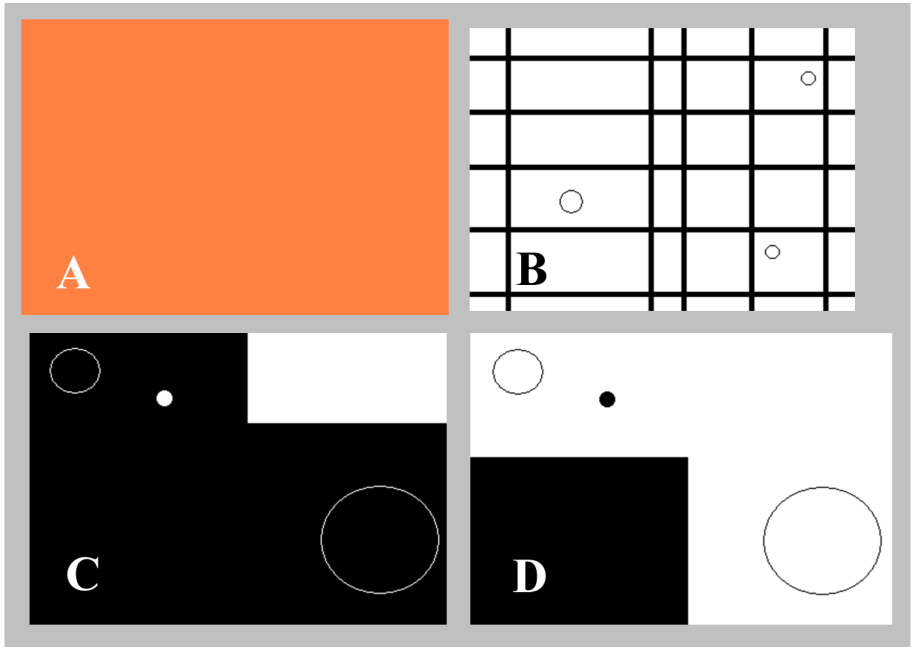


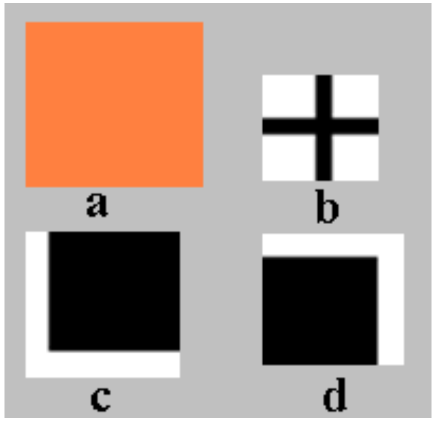


Cross correlation of

- **a and A: no match, ρ = 1 everywhere**
- b and B: ?
- c and C: ?
- d and D: ?

Example

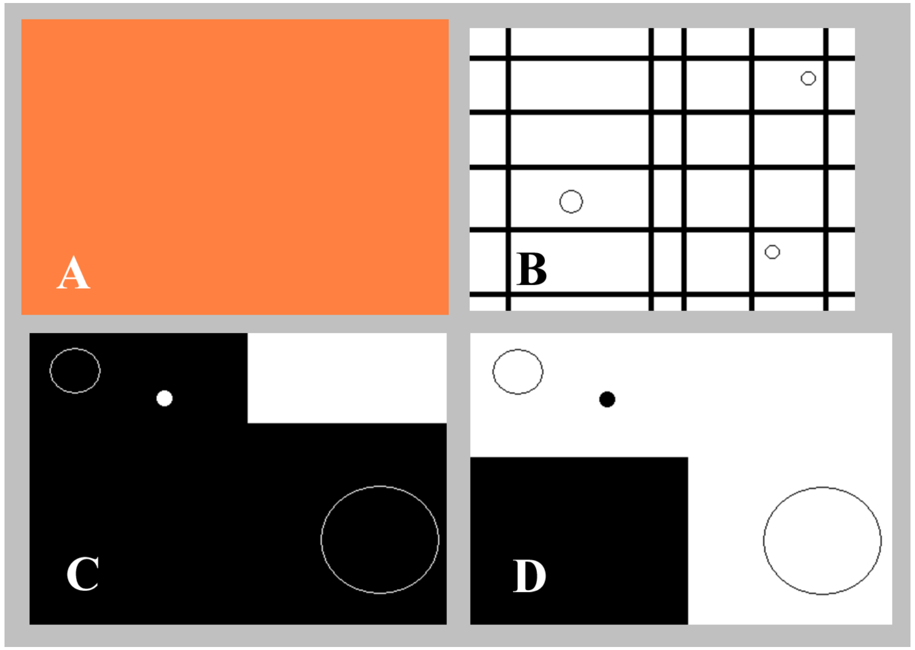


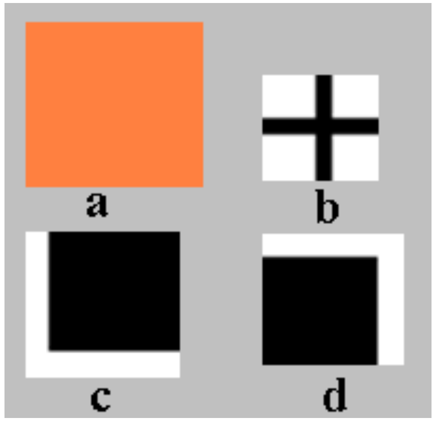


Cross correlation of

- a and A: no match, ρ = 1 everywhere
- **b and B: several matches, ρ = 1 at every cross**
- c and C: ?
- d and D: ?

Example

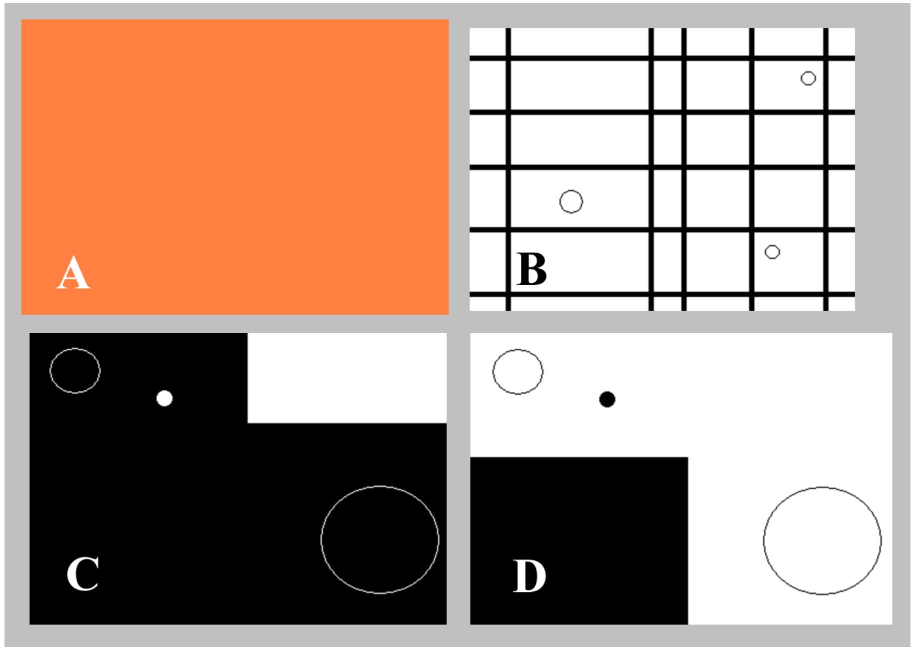


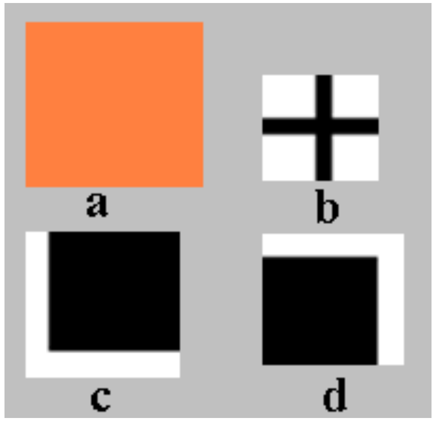


Cross correlation of

- a and A: no match, ρ = 1 everywhere
- b and B: several matches, ρ = 1 at every cross
- **c and C: no match**
- d and D: ?

Example





Cross correlation of

- a and A: no match, ρ = 1 everywhere
- b and B: several matches, ρ = 1 at every cross
- c and C: no match
- **d and D: exactly one match**

Basic Cross Correlation

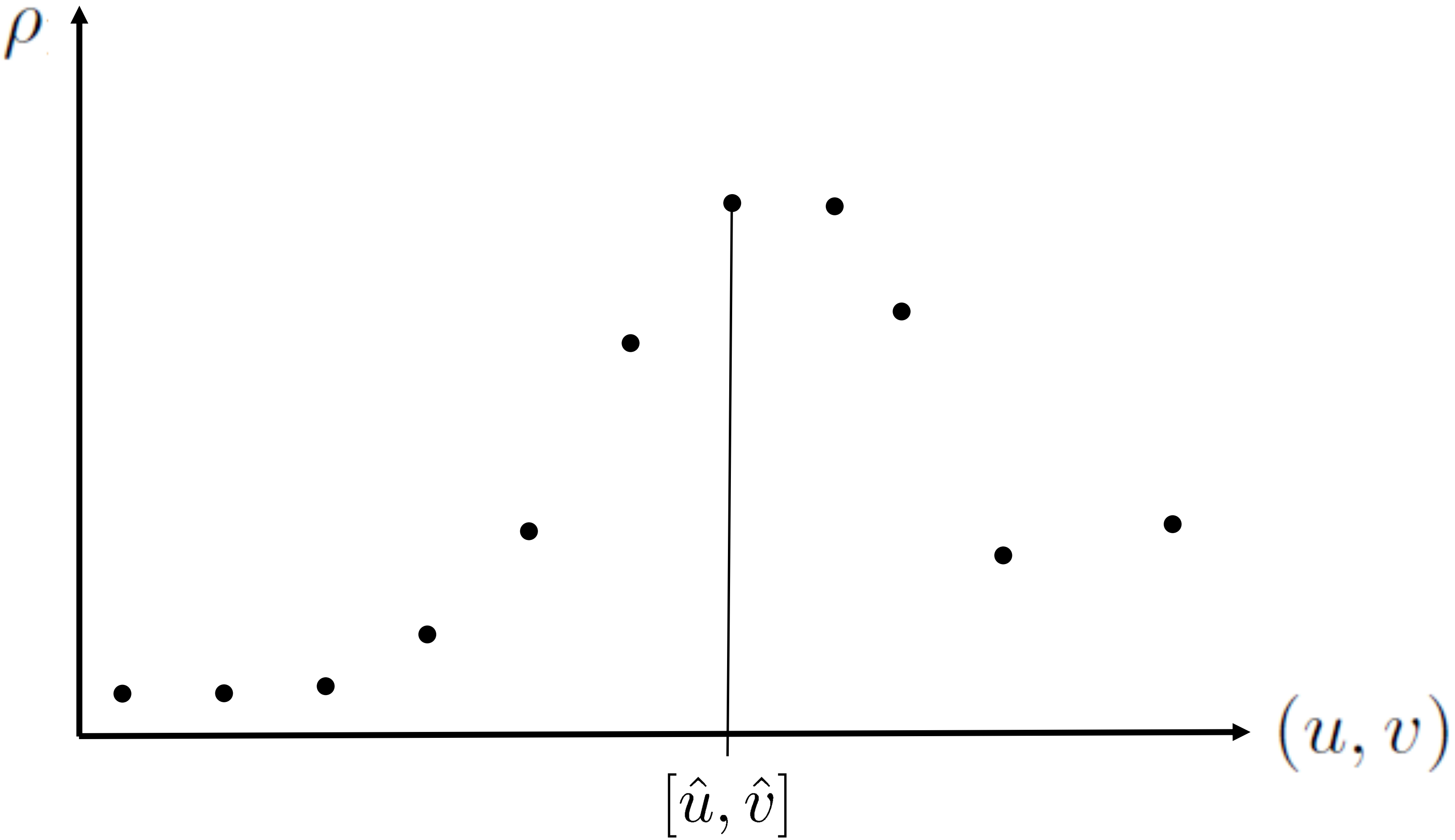
- Searches for a template image in another image
- CC is fast and easy to compute
- CC allows for variations in translation, brightness, contrast
- Changes in brightness and contrast though cross correlation function
- Search space defined by the translation parameters

Subpixel Estimation for Cross Correlation

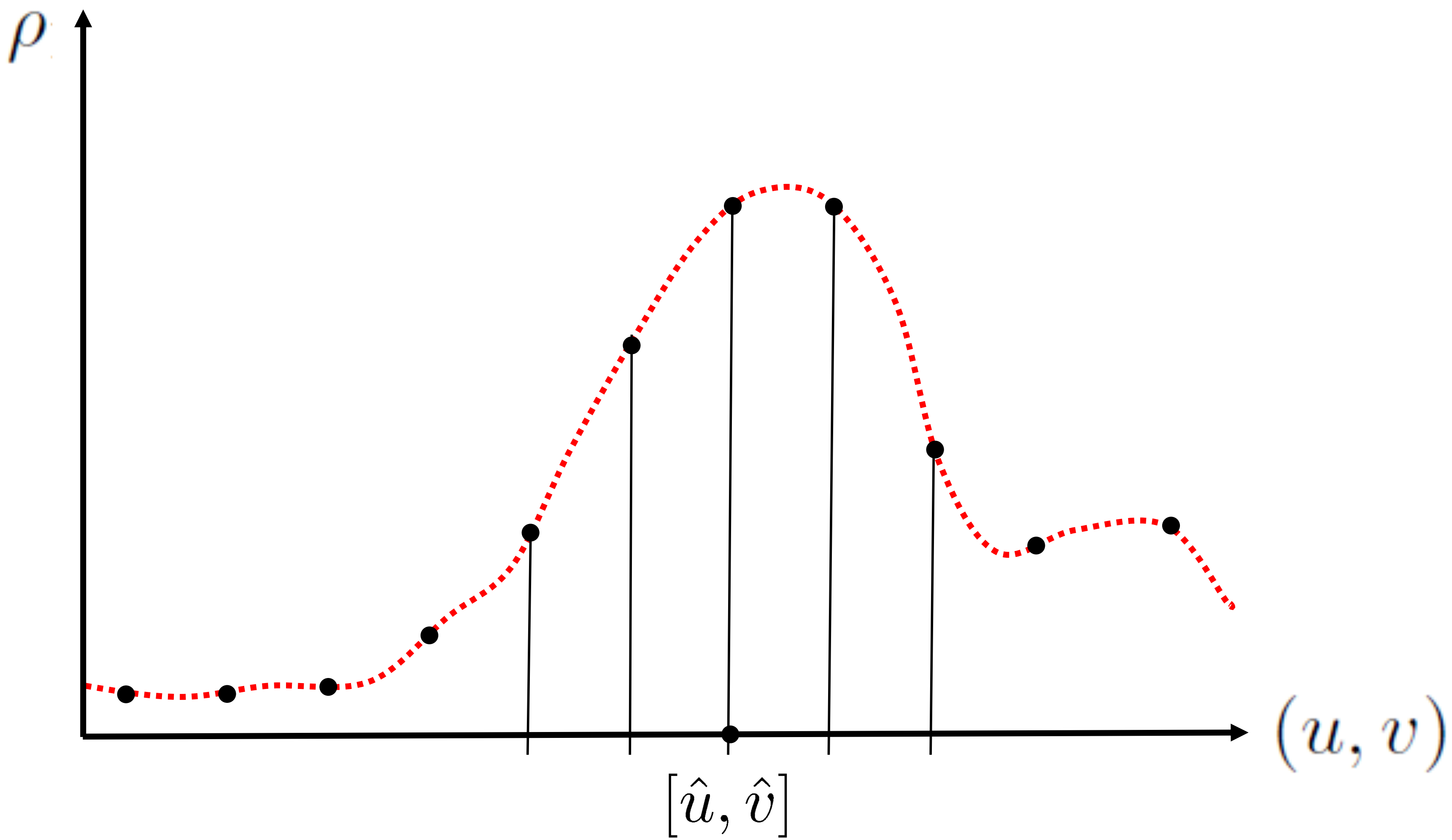
Subpixel Estimation

- Result of template matching by cross correlation is integer-valued
- More precise estimate can be obtained through subpixel estimation

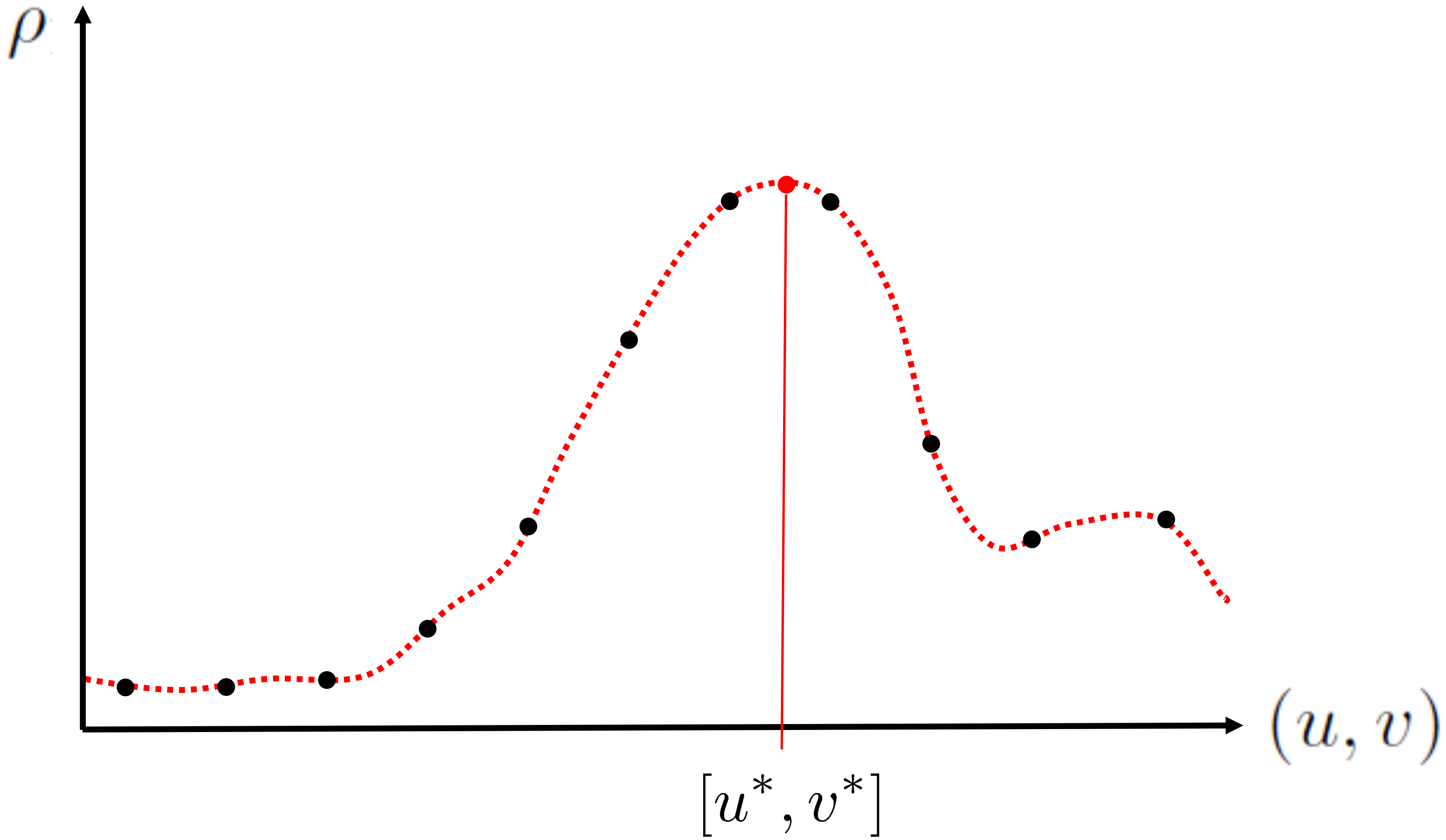
Subpixel Estimation



Subpixel Estimation



Subpixel Estimation



Subpixel Estimation

- Result of template matching by cross correlation is integer valued
- More precise estimate can be obtained through subpixel estimation

Procedure

- Fit a locally smooth surface through $\rho_{12}(u, v)$ around the initial position $[\hat{u}, \hat{v}]$
- Estimate its local maximum

Subpixel Estimation

- Fit a quadratic function around $[\hat{u}, \hat{v}]$

$$\rho(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{x}^*)^\mathsf{T} A(\boldsymbol{x} - \boldsymbol{x}^*) + a$$

NCC function in $\boldsymbol{x} = [u, v]^\mathsf{T}$

maximum at unknown $\boldsymbol{x}^*$

Subpixel Estimation

- Fit a quadratic function around $[\hat{u}, \hat{v}]$

$$\rho(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{x}^*)^\mathsf{T} A(\boldsymbol{x} - \boldsymbol{x}^*) + a$$

NCC function in $\boldsymbol{x} = [u, v]^\mathsf{T}$

maximum at unknown $\boldsymbol{x}^*$

- Compute first derivative

$$\nabla\rho(\boldsymbol{x}) = \frac{\mathrm{d}\rho(\boldsymbol{x})}{\mathrm{d}\boldsymbol{x}} = 2A(\boldsymbol{x} - \boldsymbol{x}^*)$$

- At maximum: $\nabla\rho(\boldsymbol{x}^*) = 0$

Subpixel Estimation

- First derivative $\nabla \rho(\boldsymbol{x}) = 2A(\boldsymbol{x} - \boldsymbol{x}^*)$
- Hessian $H_\rho(\boldsymbol{x}) = 2A$
- We can rewrite this to

$$\nabla \rho = H_\rho(\boldsymbol{x} - \boldsymbol{x}^*)$$

- which leads to

$$H_\rho^{-1} \nabla \rho = (\boldsymbol{x} - \boldsymbol{x}^*)$$

- and finally to

$$\boldsymbol{x}^* = \boldsymbol{x} - H_\rho\big|_{\boldsymbol{x}}^{-1} \nabla \rho\big|_{\boldsymbol{x}}$$

Subpixel Estimation

- For an image, $\boldsymbol{x}^* = \boldsymbol{x} - \left. H_\rho \right|_{\boldsymbol{x}}^{-1} \left. \nabla \rho \right|_{\boldsymbol{x}}$ consists of

$$\boldsymbol{x} = \begin{bmatrix} \hat{u} \\ \hat{v} \end{bmatrix}$$

$$\left. \nabla \rho \right|_{\boldsymbol{x}} = \begin{bmatrix} \rho_i \\ \rho_j \end{bmatrix}_{\boldsymbol{x}}$$

Sobel

$$\left. H_\rho \right|_{\boldsymbol{x}} = \begin{bmatrix} \rho_{ii} & \rho_{ij} \\ \rho_{ji} & \rho_{jj} \end{bmatrix}_{\boldsymbol{x}}$$

Operators for 2nd derivatives

Subpixel Estimation

$$\boldsymbol{x}^* = \boldsymbol{x} - \left. H_\rho \right|_{\boldsymbol{x}}^{-1} \left. \nabla \rho \right|_{\boldsymbol{x}} \qquad \boldsymbol{x} = \begin{bmatrix} \hat{u} \\ \hat{v} \end{bmatrix}$$

$$\left. \nabla \rho \right|_{\boldsymbol{x}} = \begin{bmatrix} \rho_i \\ \rho_j \end{bmatrix}_{\boldsymbol{x}} \qquad \left. H_\rho \right|_{\boldsymbol{x}} = \begin{bmatrix} \rho_{ii} & \rho_{ij} \\ \rho_{ji} & \rho_{jj} \end{bmatrix}_{\boldsymbol{x}}$$

Operators from the chapter “local operators”

$$\rho_i = \frac{\partial \rho}{\partial u} = \frac{1}{8} \begin{bmatrix} 1 & 2 & 1 \\ 0 & \underline{0} & 0 \\ -1 & -2 & -1 \end{bmatrix} * \rho$$

$$\rho_j = \frac{\partial \rho}{\partial v} = \frac{1}{8} \begin{bmatrix} 1 & 0 & -1 \\ 2 & \underline{0} & -2 \\ 1 & 0 & -1 \end{bmatrix} * \rho$$

Sobel

$$\rho_{ii} = \frac{\partial^2 \rho}{\partial u^2} = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ -2 & \underline{-4} & -2 \\ 1 & 2 & 1 \end{bmatrix} * \rho$$

$$\rho_{ij} = \frac{\partial^2 \rho}{\partial u \, \partial v} = \frac{1}{4} \begin{bmatrix} 1 & 0 & -1 \\ 0 & \underline{0} & 0 \\ -1 & 0 & 1 \end{bmatrix} * \rho$$

$$\rho_{jj} = \frac{\partial^2 \rho}{\partial v^2} = \frac{1}{4} \begin{bmatrix} 1 & -2 & 1 \\ 2 & \underline{-4} & 2 \\ 1 & -2 & 1 \end{bmatrix} * \rho$$

2^{nd} derivatives

Discussion

- CC provides the **optimal solution** when considering only translations
- Using subpixel estimation, we can obtain a **1/10 pixel precision**
- CC assumes equal and uncorrelated noise in both images
- CC cannot deal with occlusions
- Optimizations for certain situation (zero mean signals, const. variance)

Discussion

- Quality drops considerably when violating the model assumptions
  - rotation >20°
  - scale difference >30%

Summary

- Cross correlation is a standard approach for localizing a template image patch in another image
- CC is fast and easy to compute
- CC allows for variations in translation, brightness, contrast
- Subpixel estimation up to 1/10 pixel

Literature

- Szeliski, Computer Vision: Algorithms and Applications, Chapter 4
- Förstner, Scriptum Photogrammetrie I, Chapter “Matching / Kreuzkorrelation”