16.11.2021

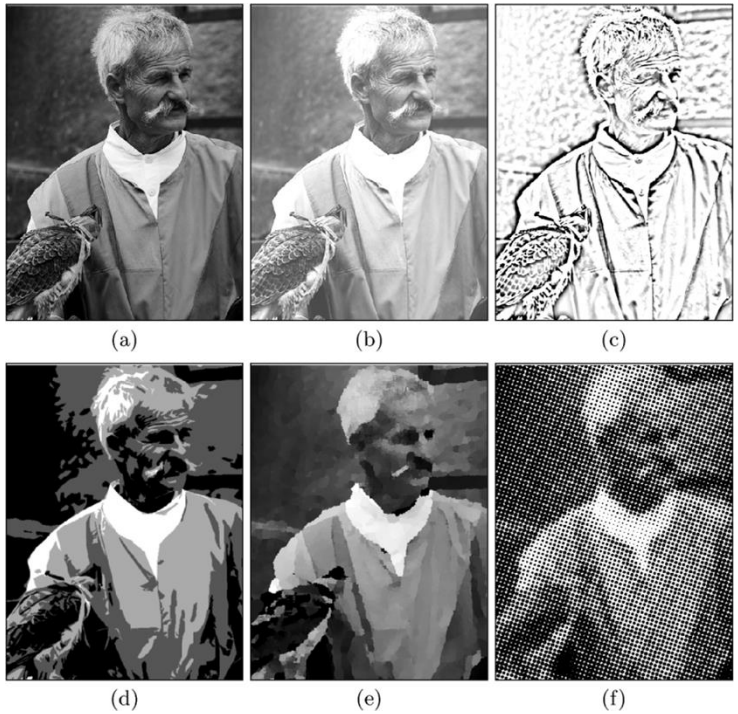
Digital Image Processing (CSE/ECE 478)
Lecture 22: Comparing Images

Ravi Kiran and Sudipta Banerjee

Slides courtesy: Emmanuel Agu, WPI and Robert Collins

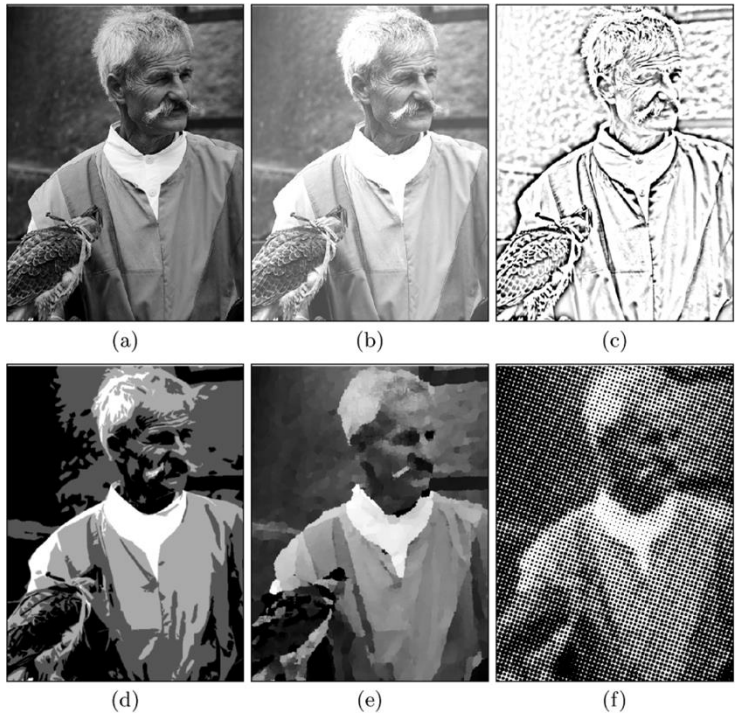
How to tell if 2 Images are same?

- Pixel by pixel comparison?
  - Makes sense only if pictures taken from same angle, same lighting, etc
- Noise, quantization, etc introduces differences
  - Human may say images are same even with numerical differences



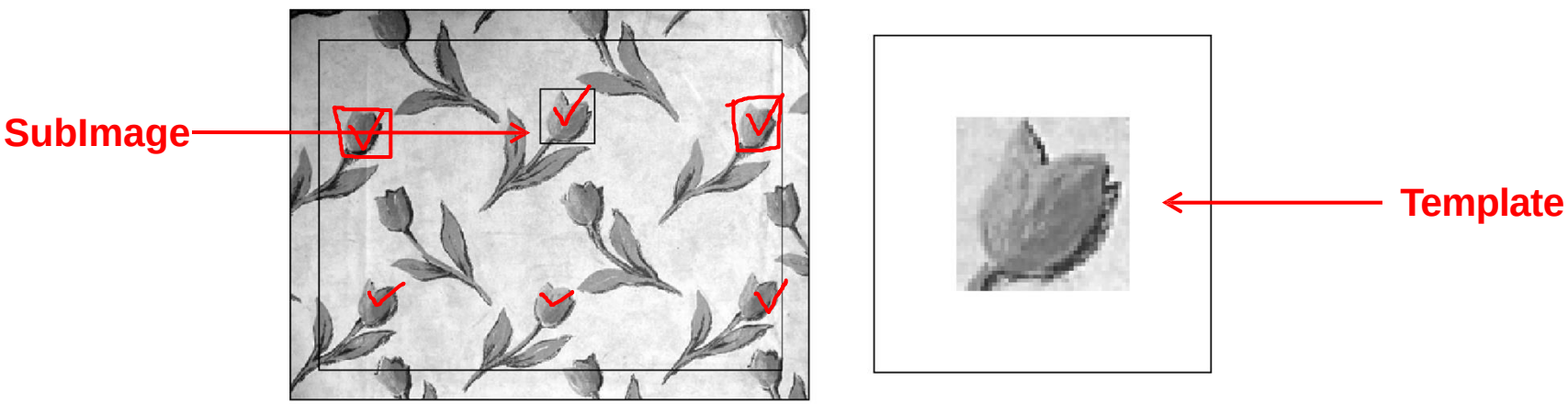
Comparing Images

- Better approach: Template matching
  - Identify similar sub-images (called template) within 2 images
- Applications?
  - Match left and right picture of stereo images
  - Find particular pattern in scene
  - Track moving pattern through image sequence



Template Matching

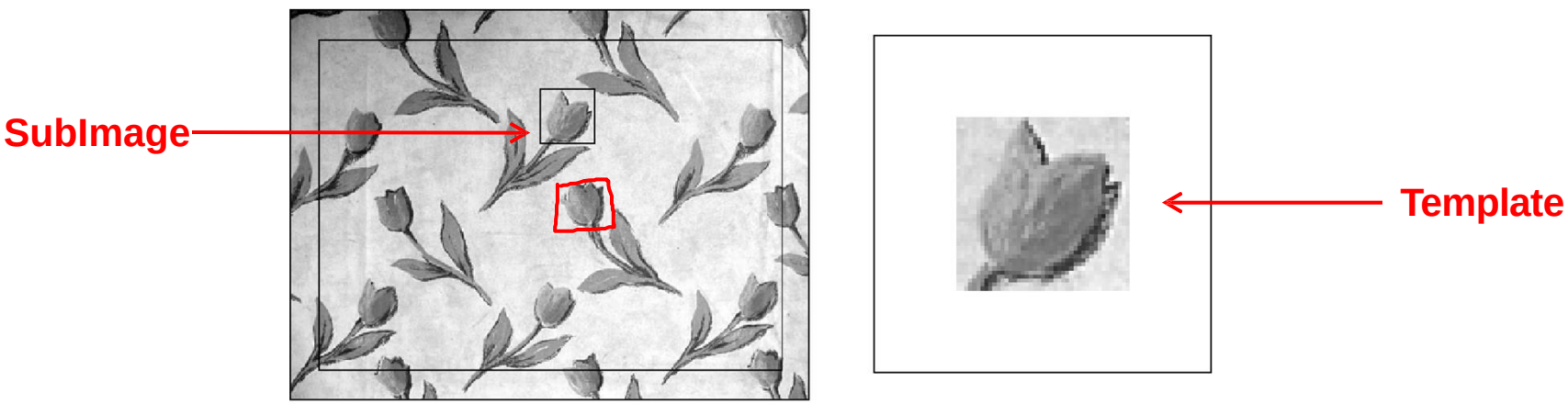
- Basic idea
  - Move given pattern (template) over search image
  - Measure difference between template and sub-images at different positions
  - Record positions where highest similarity is found



(a) original image I (b) reference image R

Template Matching

- Difficult issues
  - What is distance (difference) measure?
  - What levels of difference should be considered a match?
  - How are results affected when brightness or contrast changes?

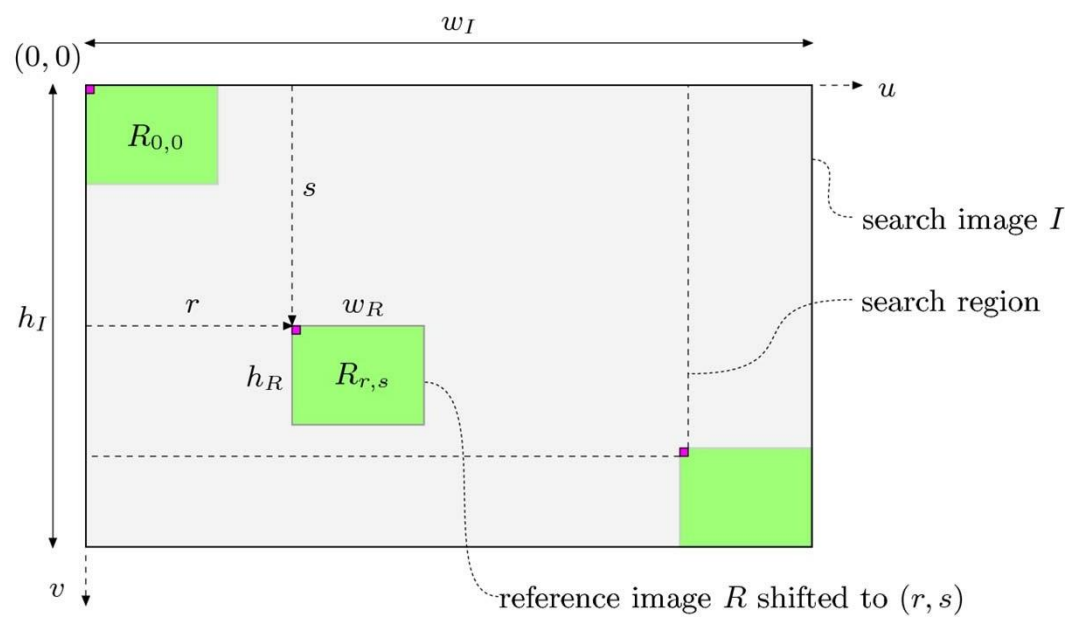


(a) original image I (b) reference image R

Template Matching in Intensity Images

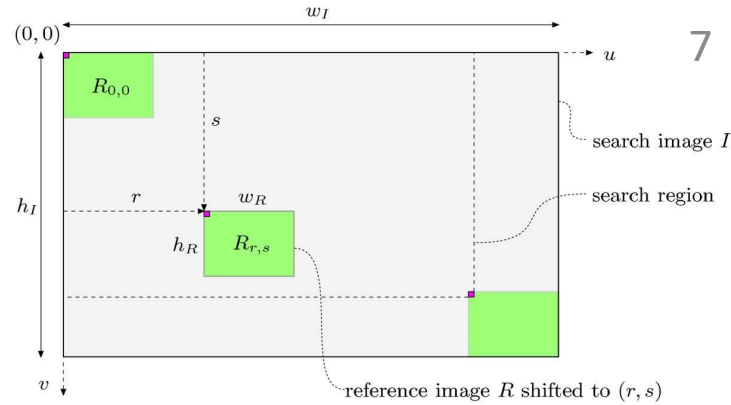
- We can restate template matching problem as:
- Finding the offset *(r,s)* such that the similarity between the shifted reference image $R_{r,s}$ and corresponding subimage I is a maximum

$$R_{r,s}(u,v) = R(u-r, v-s)$$



- Solving this problem involves solving many sub-problems

Distance between Image Patterns



- Many measures proposed to compute distance between the shifted reference image $R_{r,s}$ and corresponding subimage I
- Sum of absolute differences:

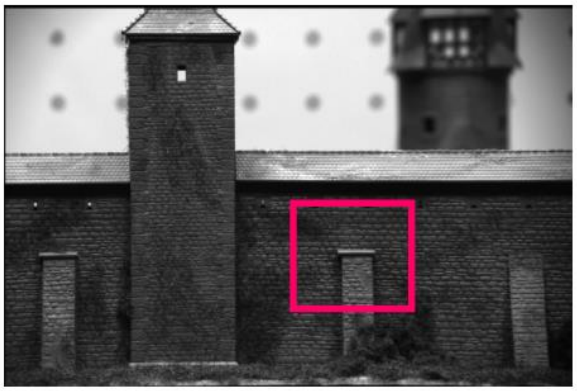
$$\mathrm{d}_A(r,s) = \sum_{(i,j)\in R} |I(r+i, s+j) - R(i,j)|$$

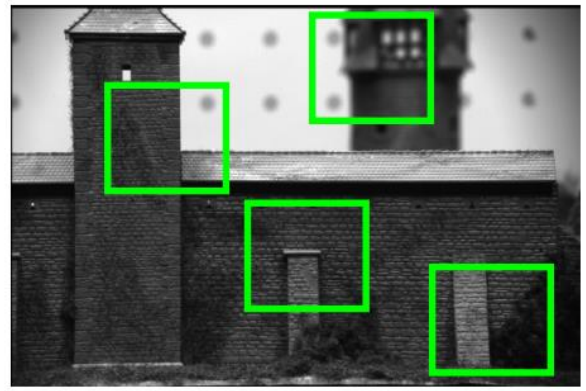
- Maximum difference:

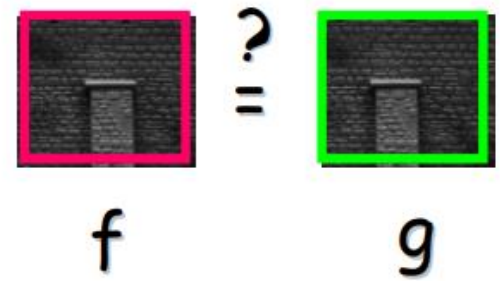
$$\mathrm{d}_M(r,s) = \max_{(i,j)\in R} |I(r+i, s+j) - R(i,j)|$$

- Sum of squared differences (also called N-dimensional Euclidean distance):

$$\mathrm{d}_E(r,s) = \Big[\sum_{(i,j)\in R} \big(I(r+i, s+j) - R(i,j)\big)^2 \Big]^{1/2}$$







Some possible measures:

$$\max_{[i,j]\in R} |f(i,j) - g(i,j)|$$

$$\sum_{[i,j]\in R} |f(i,j) - g(i,j)|$$

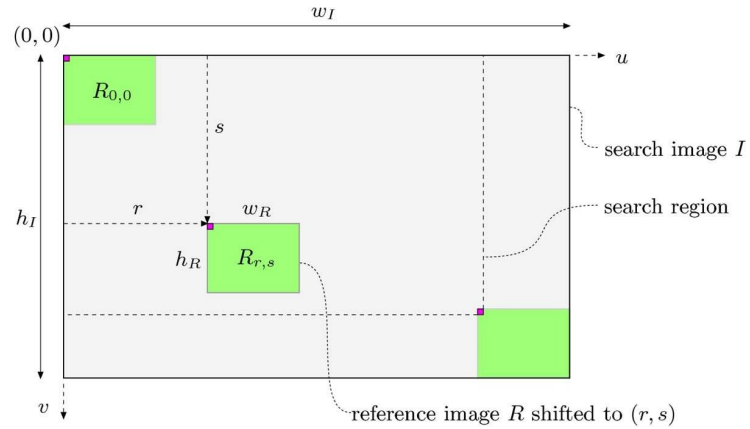
$$SSD = \sum_{[i,j]\in R} (f(i,j) - g(i,j))^2$$

Correlation (if we do exhaustive search, this is cross-correlation)

$$C_{fg} = \sum_{[i,j]\in R} f(i,j)g(i,j)$$

Most popular

Cross-correlation



$$\boxed{(I \circledast R)(r,s) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} I(r+i, s+j) \cdot R(i,j)}$$

Cross-correlation defined between search image I and reference template R

$$\sum_{i=0}^{w_R-1} \sum_{j=0}^{h_R-1} I(r+i, s+j) \cdot R(i,j) = \sum_{(i,j)\in R} I(r+i, s+j) \cdot R(i,j)$$

Since R and I supposed to be zero outside their boundaries

$$\sum_{m=-\infty}^{m=\infty} x[m]y[m-k] = \sum_{n=-\infty}^{n=\infty} y[n]x[n+k]$$

Problem with Cross-Correlation Using Raw Image Templates

Consider correlation of template with an image of constant grey value:

a	b	c
d	e	f
g	h	i

$\otimes$

v	v	v
v	v	v
v	v	v

Result: v*(a+b+c+d+e+f+g+h+i)

Problem with Cross-Correlation Using Raw Image Templates

Now consider correlation with a constant image that is twice as bright.

a	b	c
d	e	f
g	h	i

$\otimes$

2v	2v	2v
2v	2v	2v
2v	2v	2v

Result: 2*v*(a+b+c+d+e+f+g+h+i)
> v*(a+b+c+d+e+f+g+h+i)

Larger score, regardless of what the template is!

Problem with Cross-Correlation Using Raw Image Templates

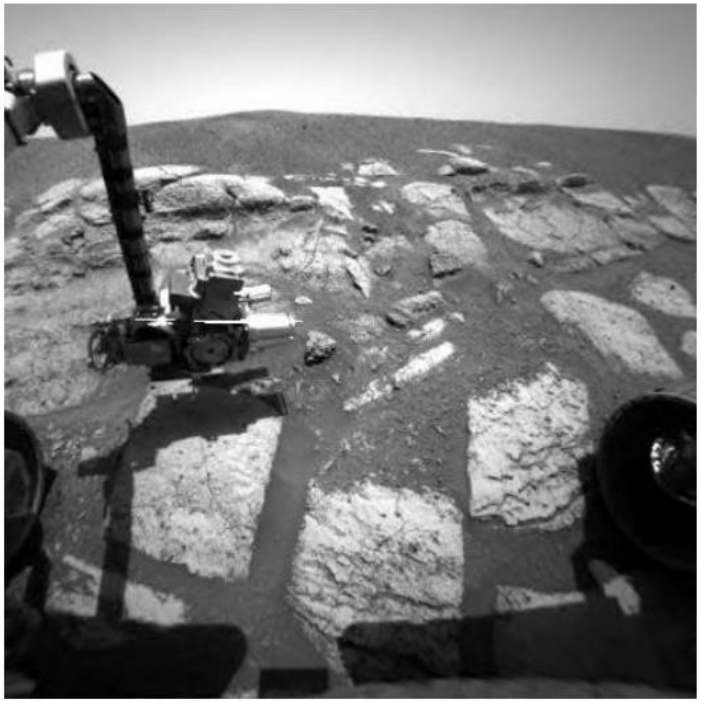


Image 1

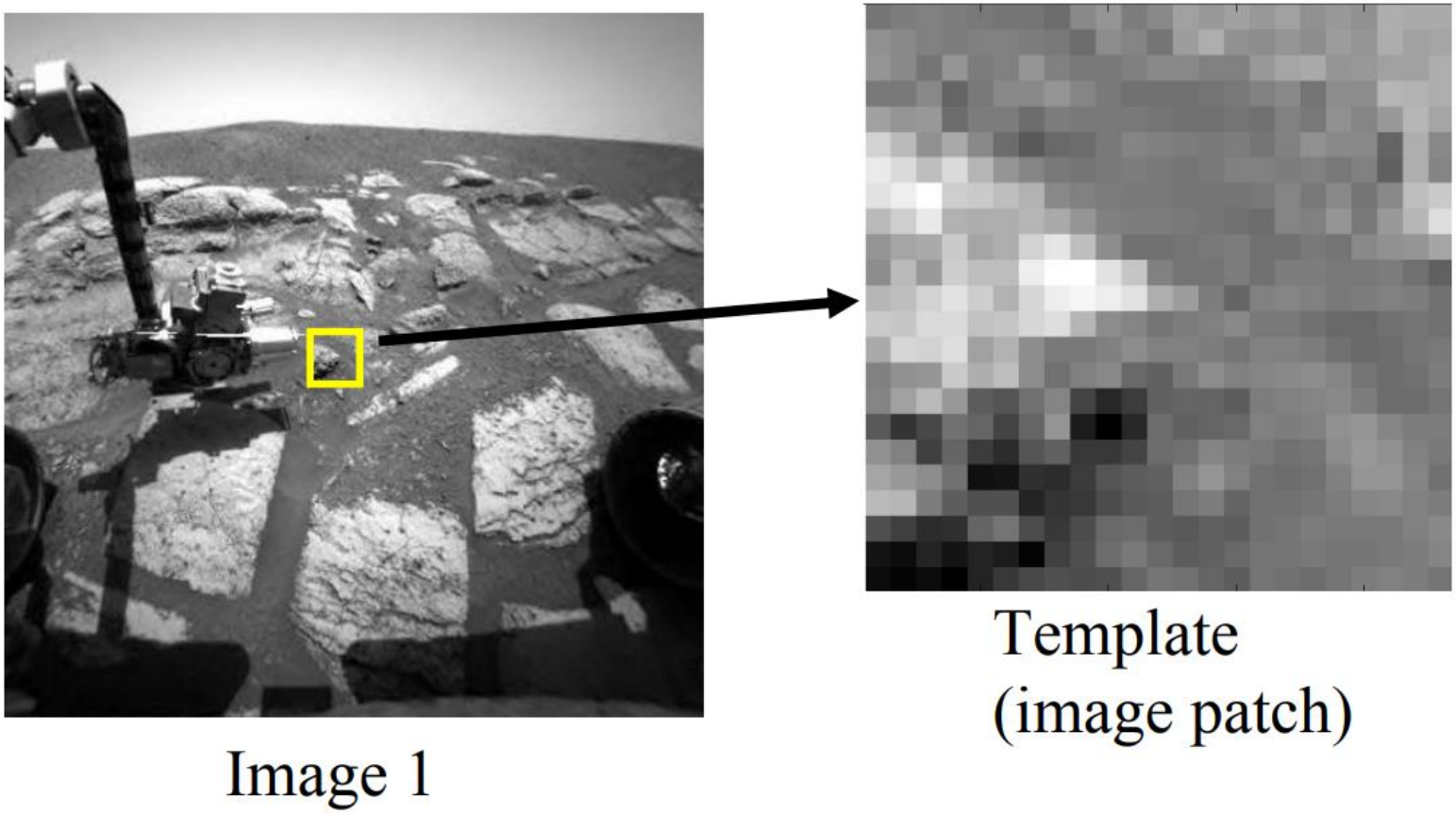
Image 2

Note: this is a stereo pair from the NASA mars rover. The rover is exploring the “El Capitan” formation.

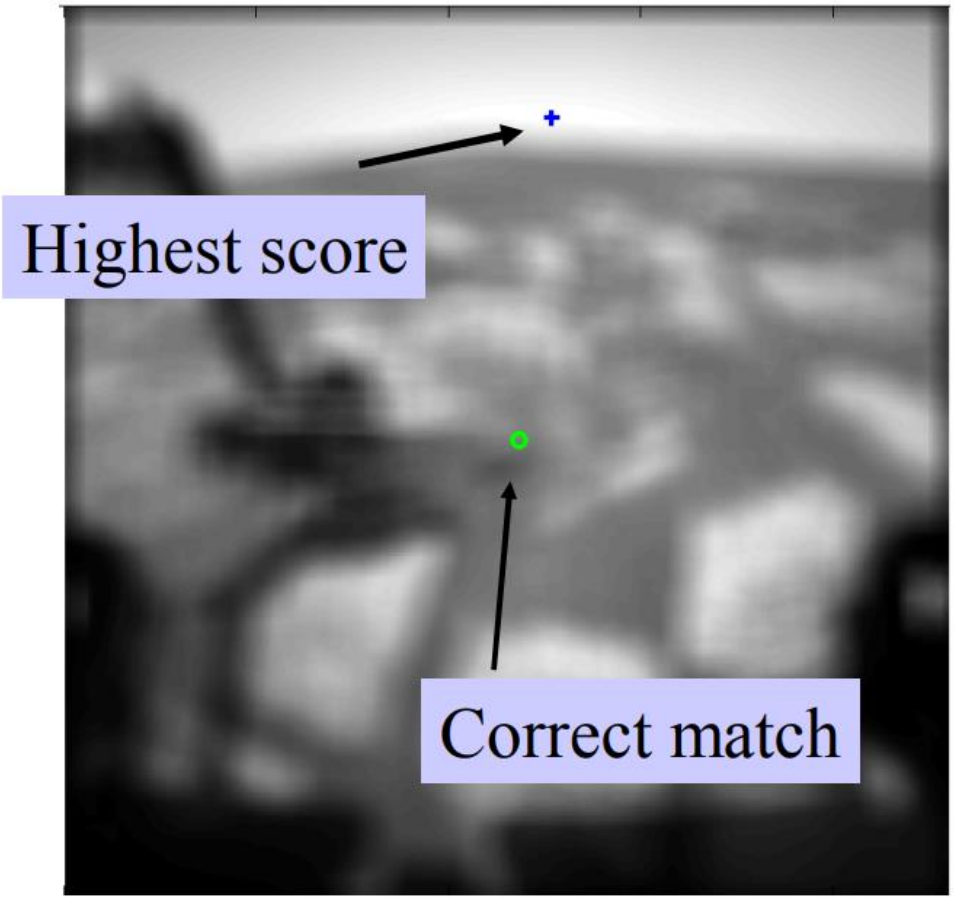
Problem with Cross-Correlation Using Raw Image Templates

Robert Collins
CSE486, Penn State

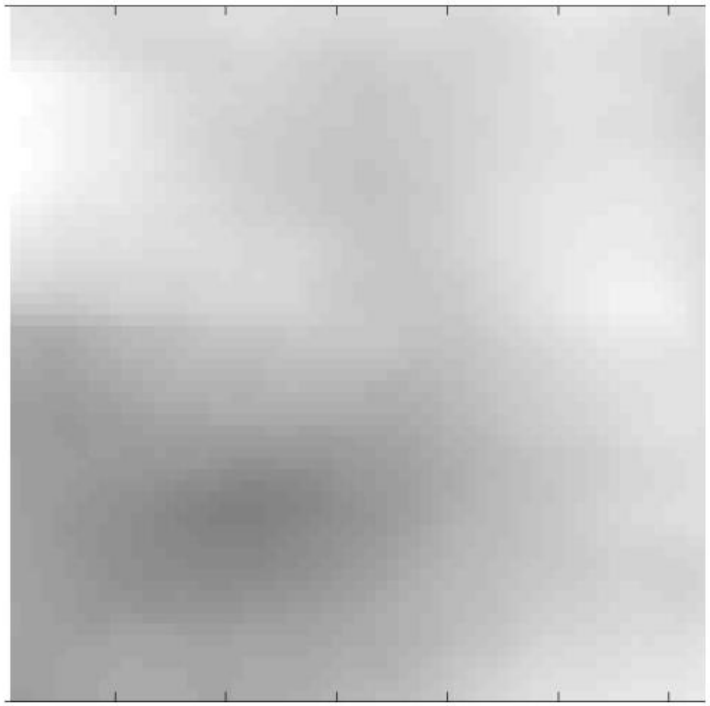
Example



Problem with Cross-Correlation Using Raw Image Templates



Score around correct match



Robert Collins
CSE486, Penn State

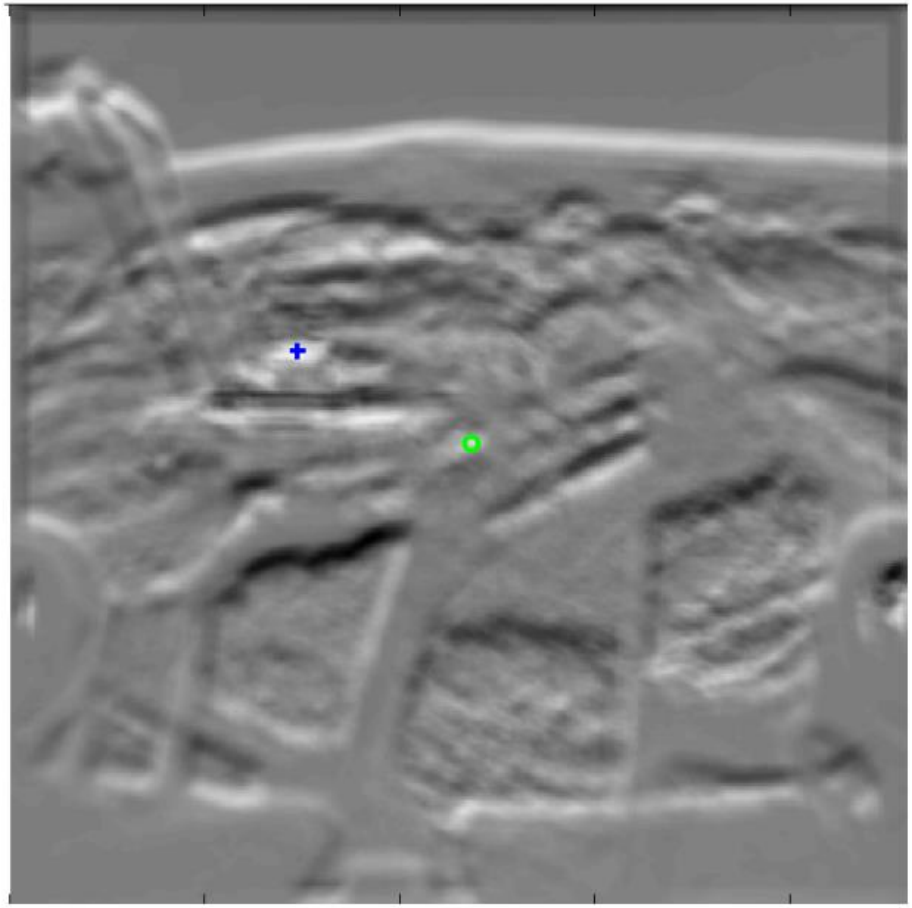
Solution

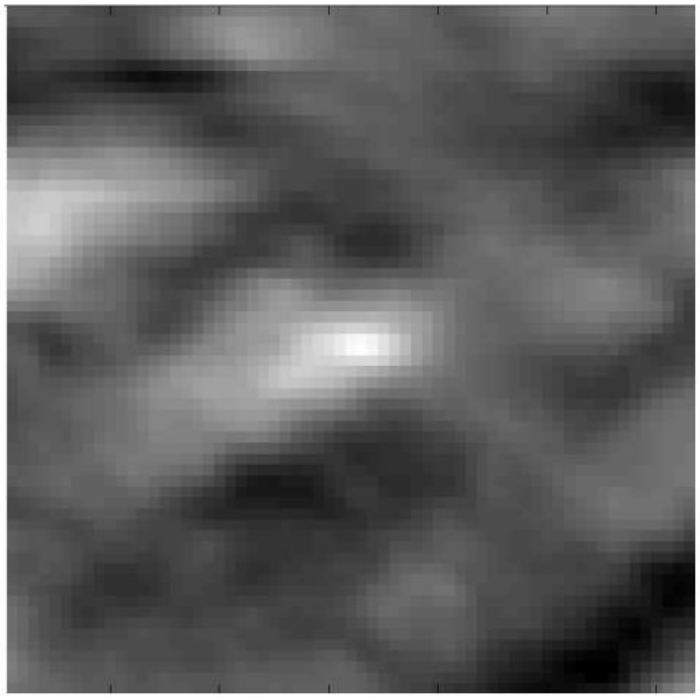
Subtract off the mean value of the template.

In this way, the correlation score is higher only when darker parts of the template overlap darker parts of the image, and brighter parts of the template overlap brighter parts of the image.

Robert Collins
CSE486, Penn State

Correlation, zero-mean template





Better! But highest score is still not the correct match.

Note: highest score IS best within local neighborhood of correct match.

Robert Collins
CSE486, Penn State

"SSD" or "block matching" (Sum of Squared Differences)

$$\sum_{[i,j]\in R} (f(i,j) - g(i,j))^2$$

$2fg$

Minimize

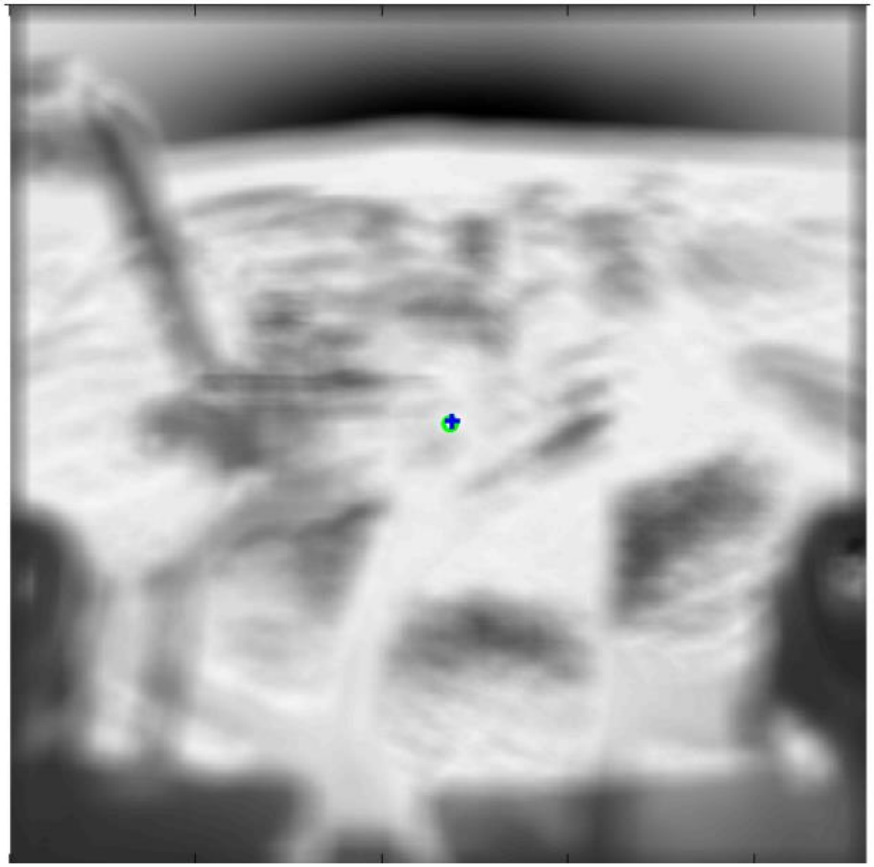
$$\mathrm{d}_E^2(r,s) = \sum_{(i,j)\in R} \big(I(r+i,s+j) - R(i,j)\big)^2$$

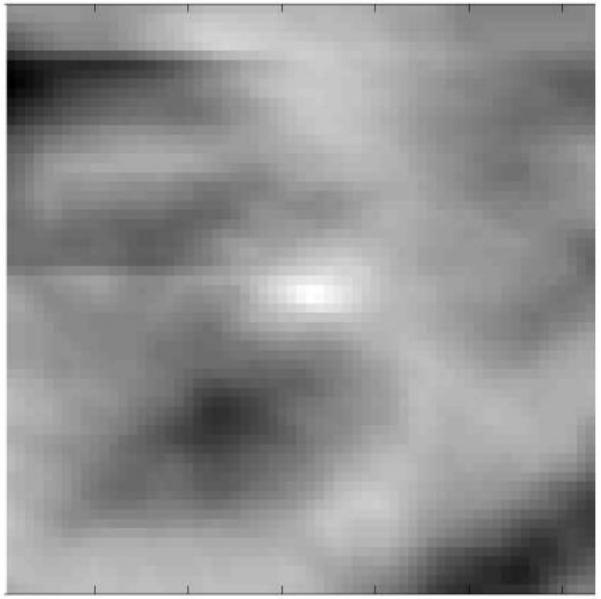
$$= \underbrace{\sum_{(i,j)\in R} I^2(r+i,s+j)}_{A(r,s)} + \underbrace{\sum_{(i,j)\in R} R^2(i,j)}_{B} - \underbrace{2\sum_{(i,j)\in R} I(r+i,s+j)\cdot R(i,j)}_{C(r,s)}$$

Maximize

Robert Collins
CSE486, Penn State

SSD





Best match (highest score) in image coincides with correct match in this case!

Robert Collins
CSE486, Penn State

Handling Intensity Changes

Intensity Changes:

•the camera taking the second image might have different intensity response characteristics than the camera taking the first image

•Illumination in the scene could change

•The camera might have auto-gain control set, so that it's response changes as it moves through the scene.

Robert Collins
CSE486, Penn State

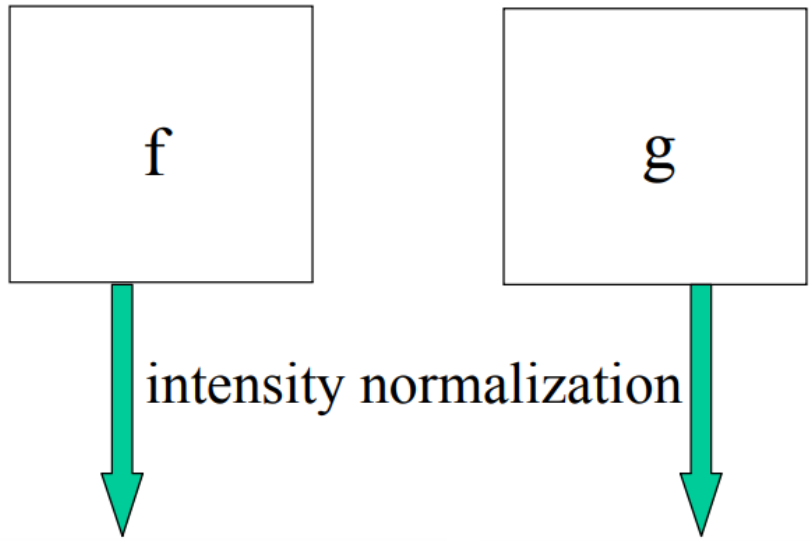
Intensity Normalization

- When a scene is imaged by different sensors, or under different illumination intensities, both the SSD and the C_{fg} can be large for windows representing the same area in the scene!
- A solution is to NORMALIZE the pixels in the windows before comparing them by subtracting the mean of the patch intensities and dividing by the std.dev.

$$\hat{f} = \frac{f - \bar{f}}{\sqrt{\sum (f - \bar{f})^2}} \qquad \hat{g} = \frac{g - \bar{g}}{\sqrt{\sum (g - \bar{g})^2}}$$

Robert Collins
CSE486, Penn State

Normalized Cross Correlation

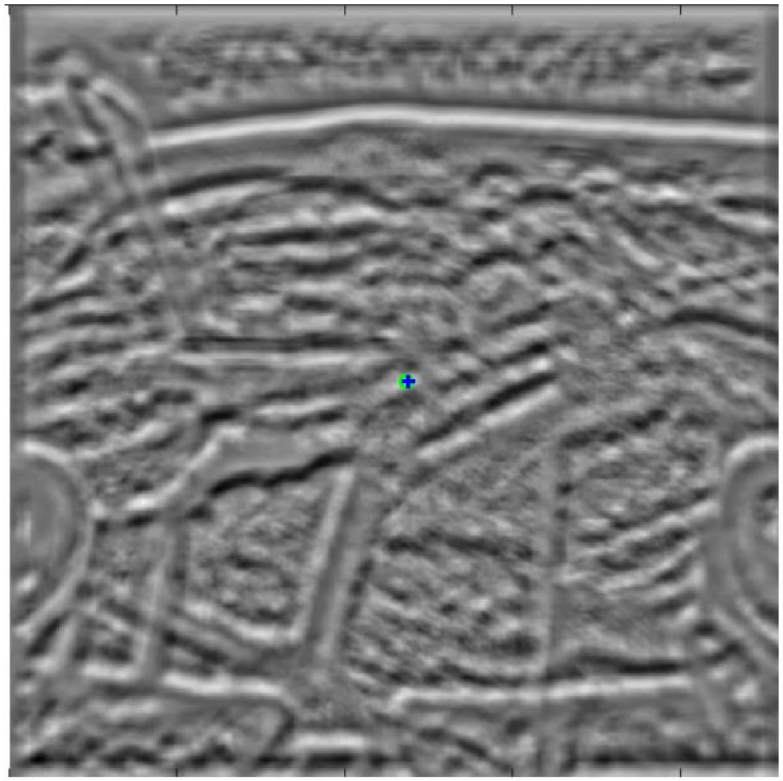


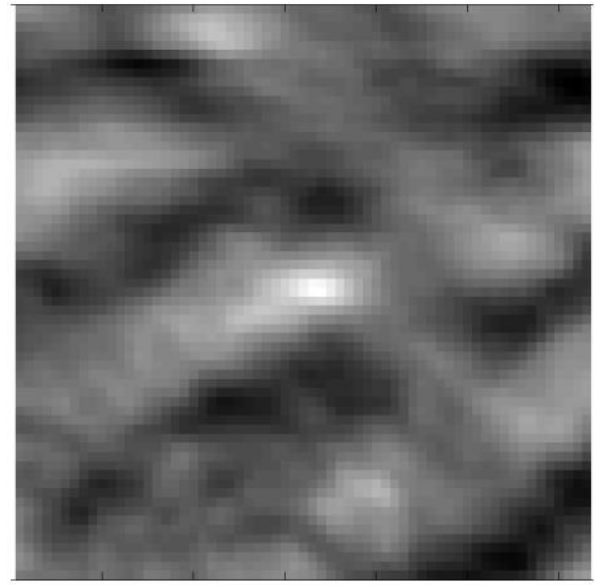
$$\hat{f} = \frac{f - \bar{f}}{\sqrt{\sum (f - \bar{f})^2}} \qquad \hat{g} = \frac{g - \bar{g}}{\sqrt{\sum (g - \bar{g})^2}}$$

$$\text{NCC(f,g)} = C_{fg}(\hat{f}, \hat{g}) = \sum_{[i,j] \in R} \hat{f}(i,j)\hat{g}(i,j)$$

$$NCC(f,g) = \frac{f.g}{\|f\|\|g\|}$$

Normalized Cross Correlation





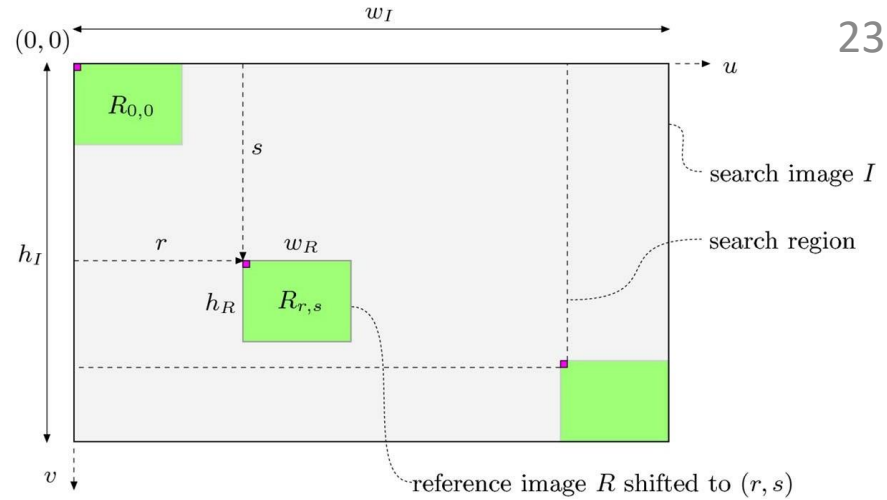
Highest score also coincides with correct match.
Also, looks like less chances of getting a wrong match.

Important point about NCC:
Score values range from 1 (perfect match)
to -1 (completely anti-correlated)

Intuition: treating the normalized patches as vectors, we see they are unit vectors. Therefore, correlation becomes dot product of unit vectors, and thus must range between -1 and 1.

Briechle and Hanbeck, "Template Matching Using Fast Normalized Cross Correlation," SPIE 2001

Correlation Coefficient



- **Correlation coefficient:** Use differences between *I* and *R* and their average values

$$C_L(r,s) = \frac{\sum_{(i,j)\in R} \big(I(r+i,s+j) - \bar{I}(r,s)\big) \cdot \big(R(i,j) - \bar{R}\big)}{\Big[\sum_{(i,j)\in R} \big(I(r+i,s+j) - \bar{I}_{r,s}\big)^2\Big]^{1/2} \cdot \underbrace{\Big[\sum_{(i,j)\in R} \big(R(i,j) - \bar{R}\big)^2\Big]^{1/2}}_{S_R^2 = K \cdot \sigma_R^2}}$$

where the average values are defined as

$$\bar{I}_{r,s} = \frac{1}{K} \cdot \sum_{(i,j)\in R} I(r+i,s+j) \quad \text{and} \quad \bar{R} = \frac{1}{K} \cdot \sum_{(i,j)\in R} R(i,j)$$

- *K* is number of pixels in reference image *R*
- $C_L(r,s)$ can be rewritten as

$$C_L(r,s) = \frac{\sum_{(i,j)\in R} \big(I(r+i,s+j) \cdot R(i,j)\big) \; - \; K \cdot \bar{I}_{r,s} \cdot \bar{R}}{\Big[\sum_{(i,j)\in R} I^2(r+i,s+j) \; - \; K \cdot \bar{I}_{r,s}^2\Big]^{1/2} \cdot S_R}$$

Correlation Coefficient Algorithm

CorrelationCoefficient (I, R)

$I(u,v)$: search image of size $w_I \times h_I$
$R(i,j)$: reference image of size $w_R \times h_R$
Returns $C(r,s)$ containing the values of the correlation coefficient between I and R positioned at (r,s).

Step 1–initialize:

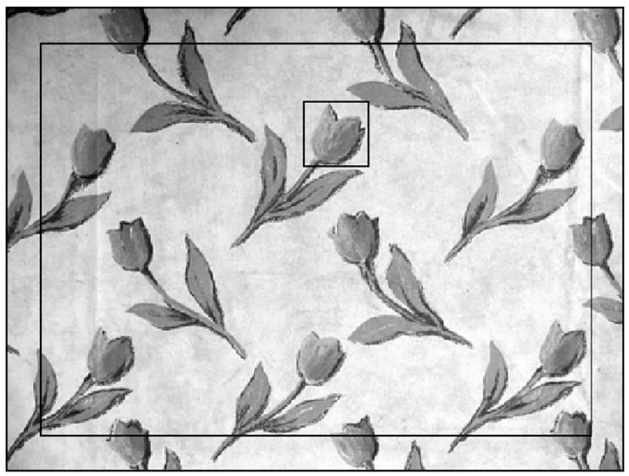
$K \leftarrow w_R \cdot h_R$
$\Sigma_R \leftarrow 0,\ \Sigma_{R2} \leftarrow 0$
for $i \leftarrow 0 \ldots (w_R - 1)$ **do**
  for $j \leftarrow 0 \ldots (h_R - 1)$ **do**
    $\Sigma_R \leftarrow \Sigma_R + R(i,j)$
    $\Sigma_{R2} \leftarrow \Sigma_{R2} + \big(R(i,j)\big)^2$
$\bar{R} \leftarrow \Sigma_R / K$ ▷ Eqn. (17.8)
$S_R \leftarrow \sqrt{\Sigma_{R2} - K \cdot \bar{R}^2} = \sqrt{\Sigma_{R2} - \Sigma_R^2 / K}$ ▷ Eqn. (17.10)

Step 2—compute the correlation map:

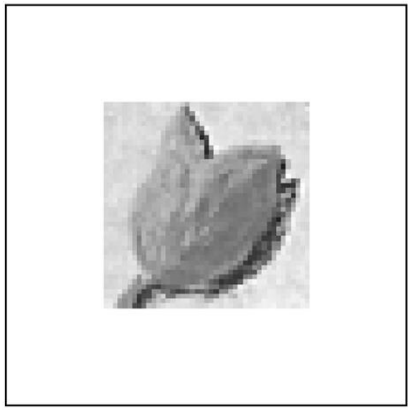
$C \leftarrow$ new map of size $(w_I - w_R + 1) \times (h_I - h_R + 1)$, $C(r,s) \in \mathbb{R}$
for $r \leftarrow 0 \ldots (w_I - w_R)$ **do** ▷ place R at position (r,s)
  for $s \leftarrow 0 \ldots (h_I - h_R)$ **do**
    Compute correlation coefficient for position (r,s):
    $\Sigma_I \leftarrow 0,\ \Sigma_{I2} \leftarrow 0,\ \Sigma_{IR} \leftarrow 0$
    for $i \leftarrow 0 \ldots (w_R - 1)$ **do**
      for $j \leftarrow 0 \ldots (h_R - 1)$ **do**
        $a_I \leftarrow I(r+i, s+j)$
        $a_R \leftarrow R(i,j)$
        $\Sigma_I \leftarrow \Sigma_I + a_I$
        $\Sigma_{I2} \leftarrow \Sigma_{I2} + a_I^2$
        $\Sigma_{IR} \leftarrow \Sigma_{IR} + a_I \cdot a_R$
    $\bar{I}_{r,s} \leftarrow \Sigma_I / K$ ▷ Eqn. (17.8)
    $$C(r,s) \leftarrow \frac{\Sigma_{IR} - K \cdot \bar{I}_{r,s} \cdot \bar{R}}{\sqrt{\Sigma_{I2} - K \cdot \bar{I}_{r,s}^2}\ \cdot S_R} = \frac{\Sigma_{IR} - \Sigma_I \cdot \bar{R}}{\sqrt{\Sigma_{I2} - \Sigma_I^2 / K}\ \cdot S_R}$$
return C. ▷ $C(r,s) \in [-1, 1]$

Examples and Discussion

- We now compare these distance metrics
- **Original image *I*:** Repetitive flower pattern
- **Reference image *R*:** one instance of repetitive pattern extracted from *I*



(a) original image I



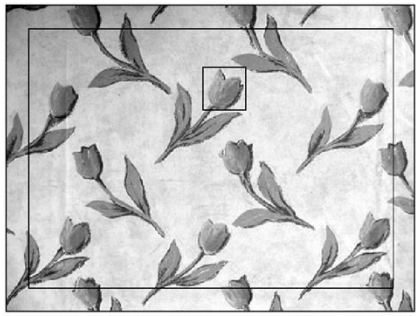
(b) reference image R

- Now compute various distance measures for this *I* and *R*

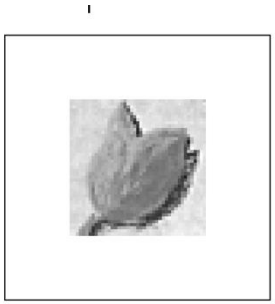
Examples and Discussion

(c) sum of absolute differences

(d) maximum difference



(a) original image I



(b) reference image R

- **Sum of absolute differences:** performs okay but affected by global intensity changes

$$\mathrm{d}_A(r,s) = \sum_{(i,j)\in R} |I(r+i, s+j) - R(i,j)|$$

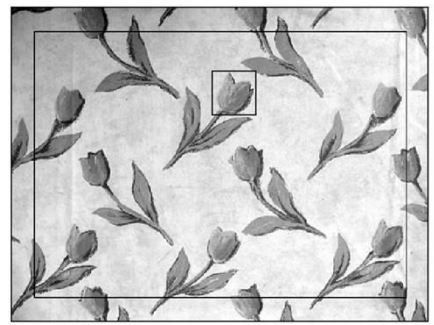
- **Maximum difference:** Responds more to lighting intensity changes than pattern similarity

$$\mathrm{d}_M(r,s) = \max_{(i,j)\in R} |I(r+i, s+j) - R(i,j)|$$

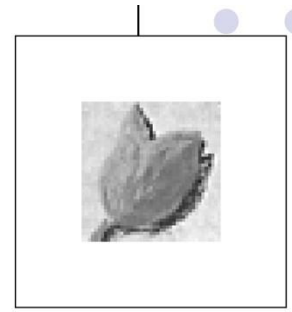
Examples and Discussion

(e) sum of squared distances

(f) global cross correlation



(a) original image I



(b) reference image R

- **Sum of squared (euclidean) distances:** performs okay but affected by global intensity changes

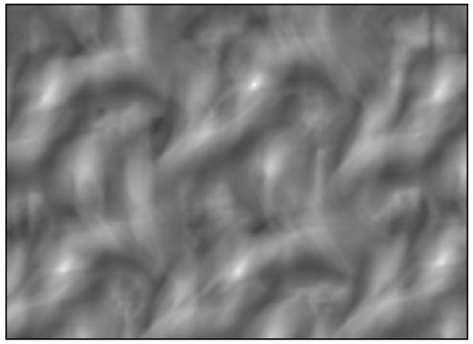
$$\mathrm{d}_E(r,s) = \Big[\sum_{(i,j)\in R} \big(I(r+i,s+j) - R(i,j)\big)^2 \Big]^{1/2}$$

- **Global cross correlation:** Local maxima at true template position, but is dominated by high-intensity responses in brighter image parts

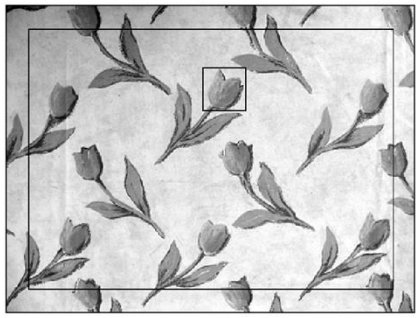
$$(I \circledast R)(r,s) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} I(r+i,s+j) \cdot R(i,j)$$

Examples and Discussion

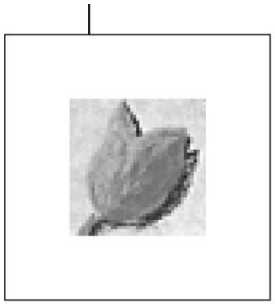
(g) normalized cross correlation



(h) correlation coefficient



(a) original image I



(b) reference image R

- **Normalized cross correlation:** results similar to euclidean distance (affected by global intensity changes)

$$NCC(f,g) = \frac{f \cdot g}{\|f\|\|g\|}$$

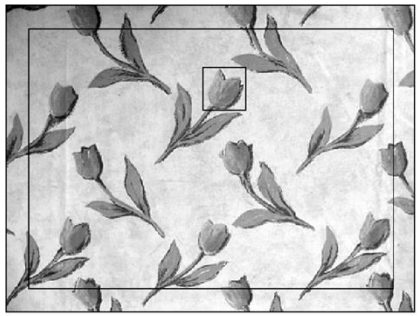
$$\frac{\sum_{(i,j)\in R} I(r+i,s+j)\cdot R(i,j)}{\Big[\sum_{(i,j)\in R} I^2(r+i,s+j)\Big]^{1/2}\cdot\Big[\sum_{(i,j)\in R} R^2(i,j)\Big]^{1/2}}$$

- **Correlation coefficient:** yields best results. Distinct peaks produced for all 6 template instances, unaffected by lighting

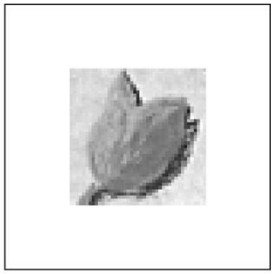
$$C_L(r,s) = \frac{\sum_{(i,j)\in R}\big(I(r+i,s+j)\cdot R(i,j)\big) - K\cdot\bar{I}_{r,s}\cdot\bar{R}}{\Big[\sum_{(i,j)\in R} I^2(r+i,s+j) - K\cdot\bar{I}^2_{r,s}\Big]^{1/2}\cdot S_R}$$

Effects of Changing Intensity

- To explore effects of globally changing intensity, raise intensiy of reference image *R* by 50 units
- Distinct peaks disappear in **Euclidean distance**
- **Correlation coefficient** unchanged, robust measure in realistic lighting conditions



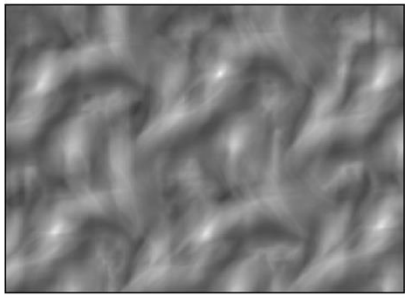
(a) original image I



(b) reference image R

Original reference image: R

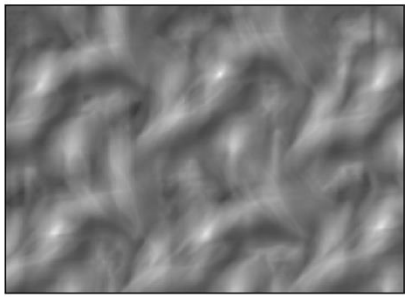
(a) Euclidean distance



(b) correlation coefficient

Modified reference image: $R' = R + 50$

(c) Euclidean distance



(d) correlation coefficient

Shape of Template

- Template does not have to be rectangular
- Some applications use circular, elliptical or custom-shaped templates
- Non-rectangular templates stored in rectangular array, but pixels in template marked using a mask
- More generally, a weighted function can be applied to template elements

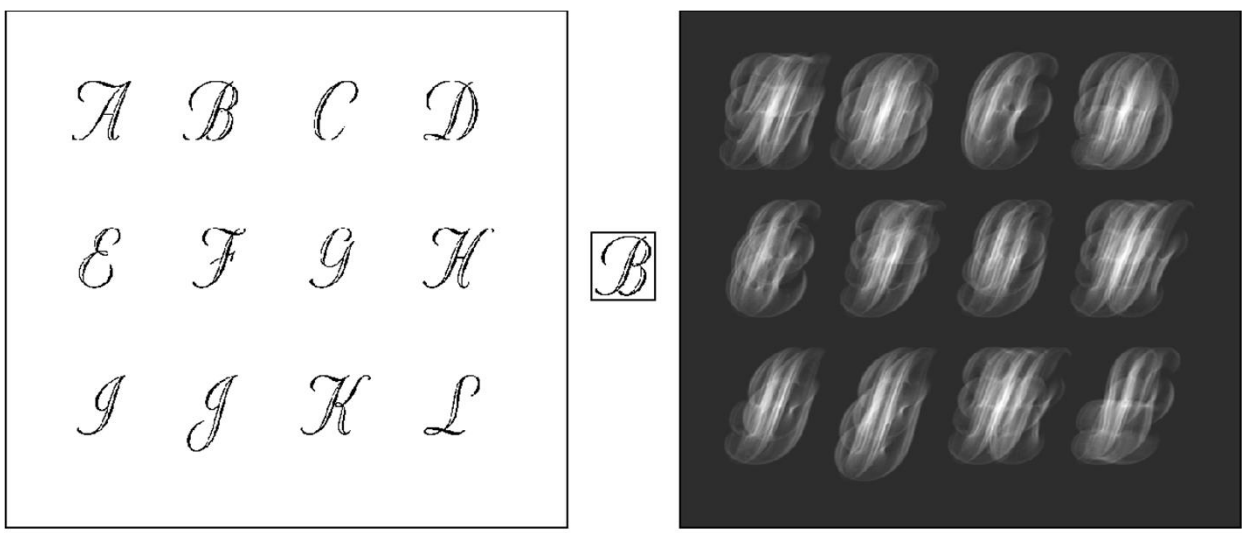
Matching under Rotation and Scaling

- **Simple Approach:**
  - Store multiple rotated and scaled versions of template
  - Computationally prohibitive

- Alternate approaches:
  - Matching in logarithmic-polar space (complicated!)
  - Affine matching: Use local statistical features invariant under affine image transformations (including rotation and scaling)

Matching Binary Images

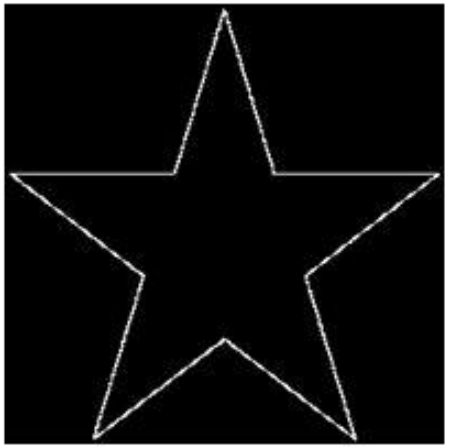
- **Direct Comparison:**
  - Count the number of identical pixels in search image and template
  - Small total difference when most pixels are same
- Problem: Small shift, rotation or distortion of image create high distance
- Need a more tolerant measure



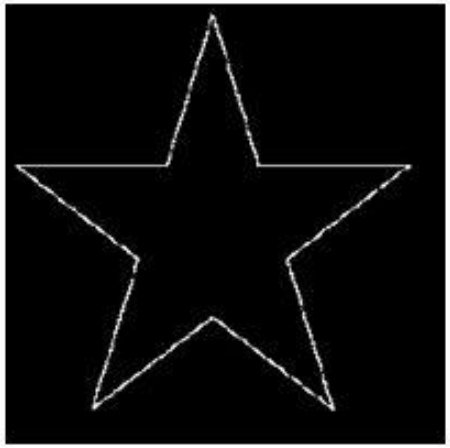
(a) (b) (c)

Detecting Shapes Without Texture

- Normalized correlation does not work well.
- Slight misalignments have a great impact on the correlation score.

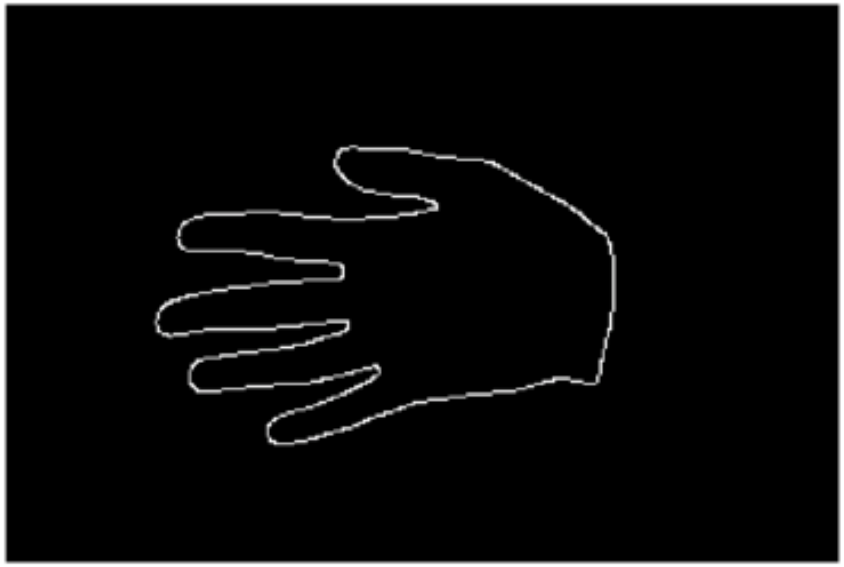


star1



star3

combined



The Distance Transform

- For every position (u,v) in the search image *I*, record distance to closest foreground pixel
- Recall: In binary image, convention is

$$FG(I) = \{\boldsymbol{p} \mid I(\boldsymbol{p}) = 1\}$$
$$BG(I) = \{\boldsymbol{p} \mid I(\boldsymbol{p}) = 0\}$$

- Distance transform is defined as

$$D(\boldsymbol{p}) = \min_{\boldsymbol{p}' \in FG(I)} \mathrm{dist}(\boldsymbol{p}, \boldsymbol{p}')$$

- Examples of distance measures are **Euclidean distance**

$$\mathrm{d}_E(\boldsymbol{p}, \boldsymbol{p}') = \|\boldsymbol{p} - \boldsymbol{p}'\| = \sqrt{(u-u')^2 + (v-v')^2} \quad \in \mathbb{R}^+$$

- Or **Manhattan distance**

$$\mathrm{d}_M(\boldsymbol{p}, \boldsymbol{p}') = |u - u'| + |v - v'| \quad \in \mathbb{N}_0$$

Distance Transform Example

- Example using Manhattan distance

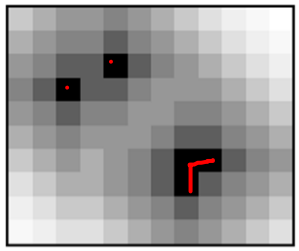
binary image

0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	**1**	0	0	0	0	0	0	0
0	0	**1**	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	**1**	**1**	0	0	0
0	0	0	0	0	0	0	**1**	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0

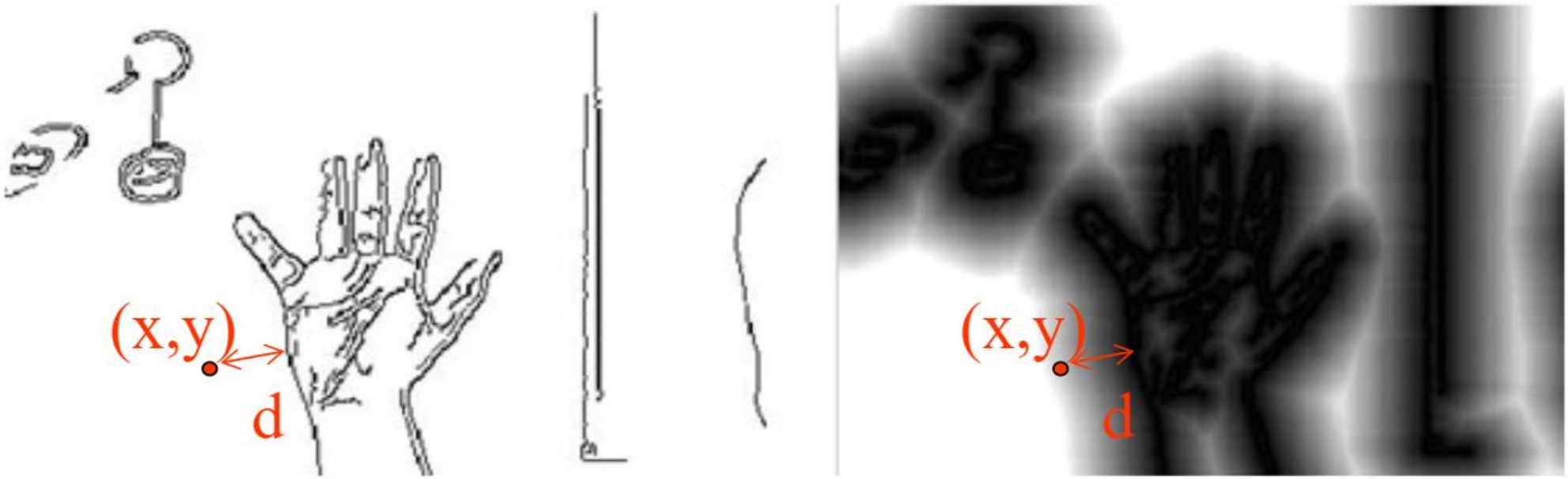
$\rightarrow$

distance transform

5	4	3	3	2	3	4	5	6	7	8	9
4	3	2	2	1	2	3	4	5	6	7	8
3	2	1	1	**0**	1	2	3	4	5	6	7
2	1	**0**	1	1	2	3	3	3	4	5	6
3	2	1	2	2	3	3	2	2	3	4	5
4	3	2	3	3	3	2	1	1	2	3	4
5	4	3	4	3	2	1	**0**	**0**	1	2	3
6	5	4	4	3	2	1	**0**	1	2	3	4
7	6	5	5	4	3	2	1	2	3	4	5
8	7	6	6	5	4	3	2	3	4	5	6



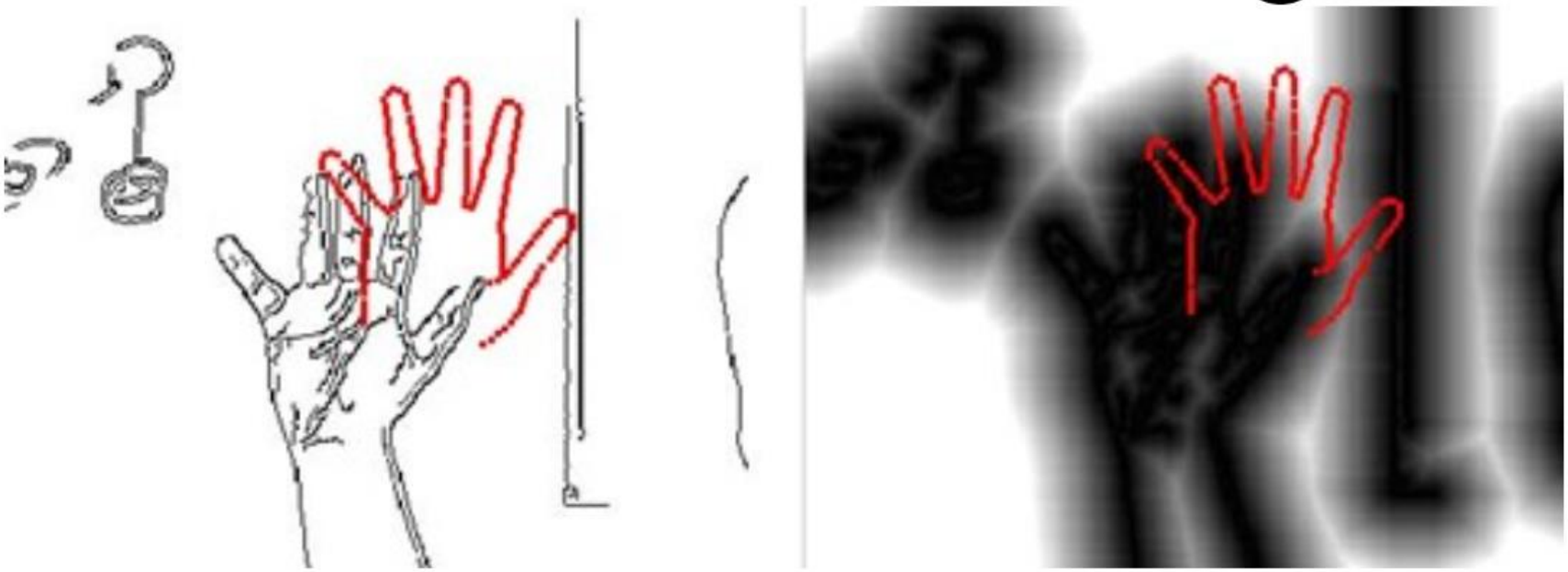
Distance Transform



- Distance image gives the distance to the nearest edge at every pixel in the image
- Calculated only once for each frame

https://slideplayer.com/slide/5097454/

Chamfer Matching



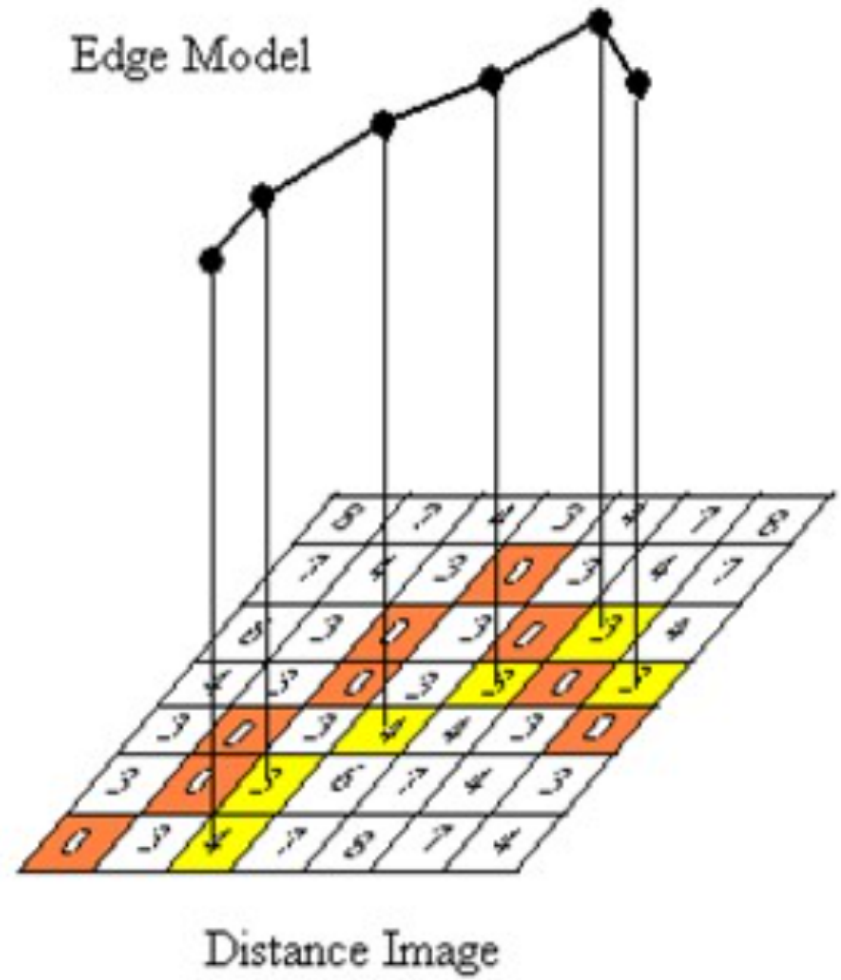
- Chamfer score is average nearest distance from template points to image points
- Nearest distances are readily obtained from the distance image
- Computationally inexpensive

Chamfer Matching

- Edge-model translated over Distance Image.
- At each translation, edge model superimposed on distance image.
- Average of distance values that edge model hits gives Chamfer Distance.

R.M.S. Chamfer Distance =

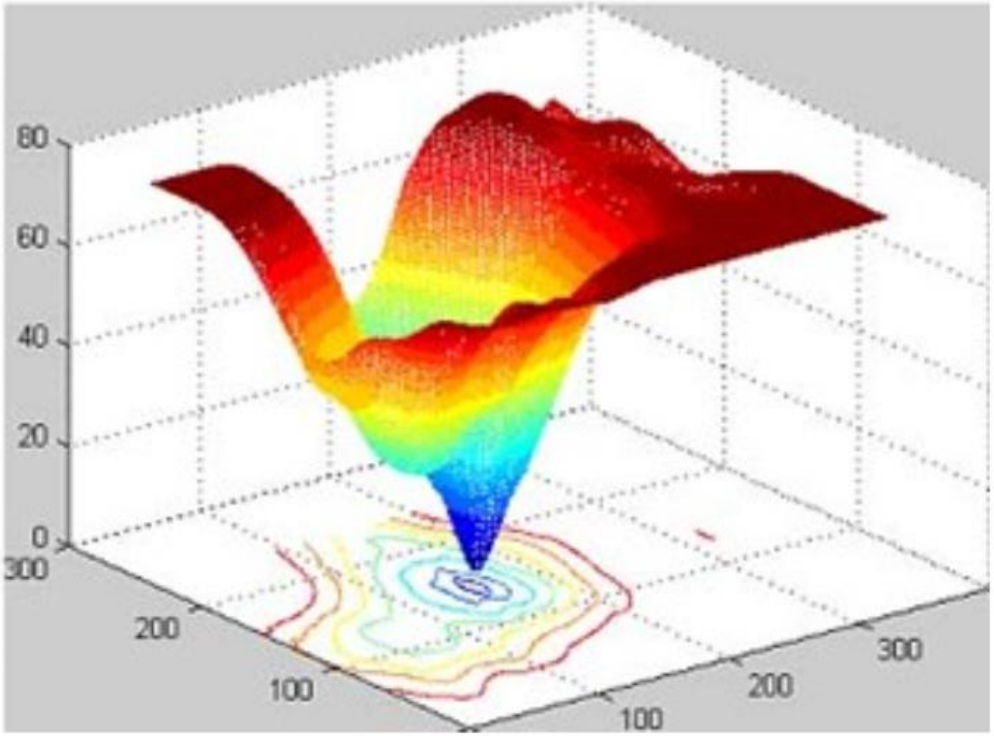
$$\frac{1}{3}\sqrt{\frac{1}{n}\sum_{i=1}^{n} v_i^2}$$



Chamfer Distance = 1.12

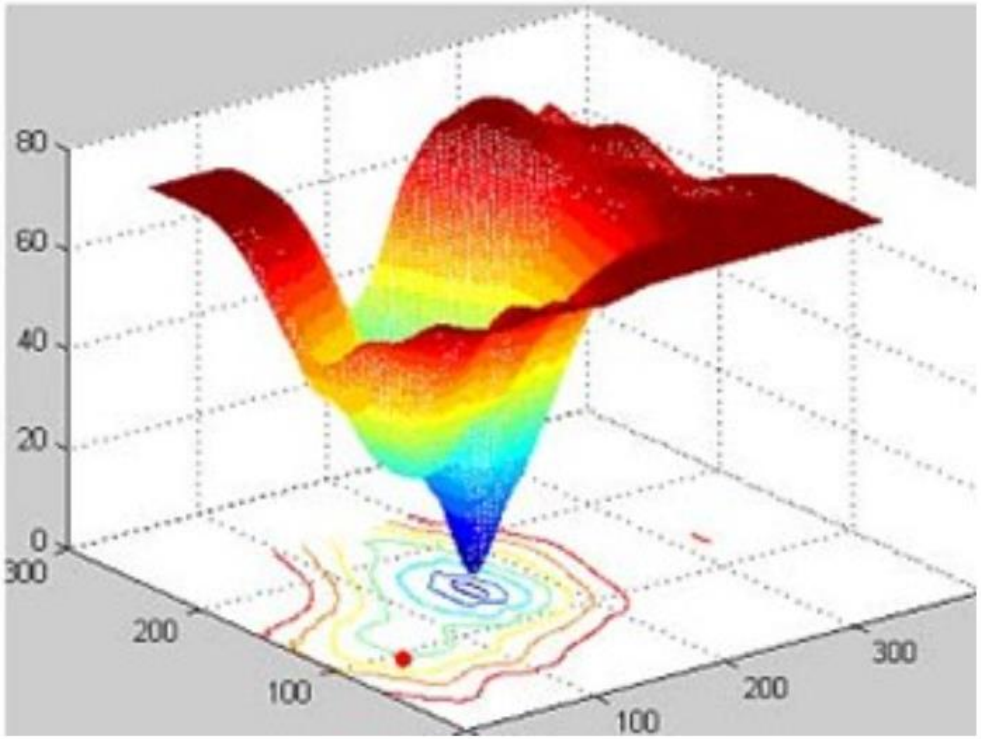
V_i = distance value, n = number of points

Chamfer Matching

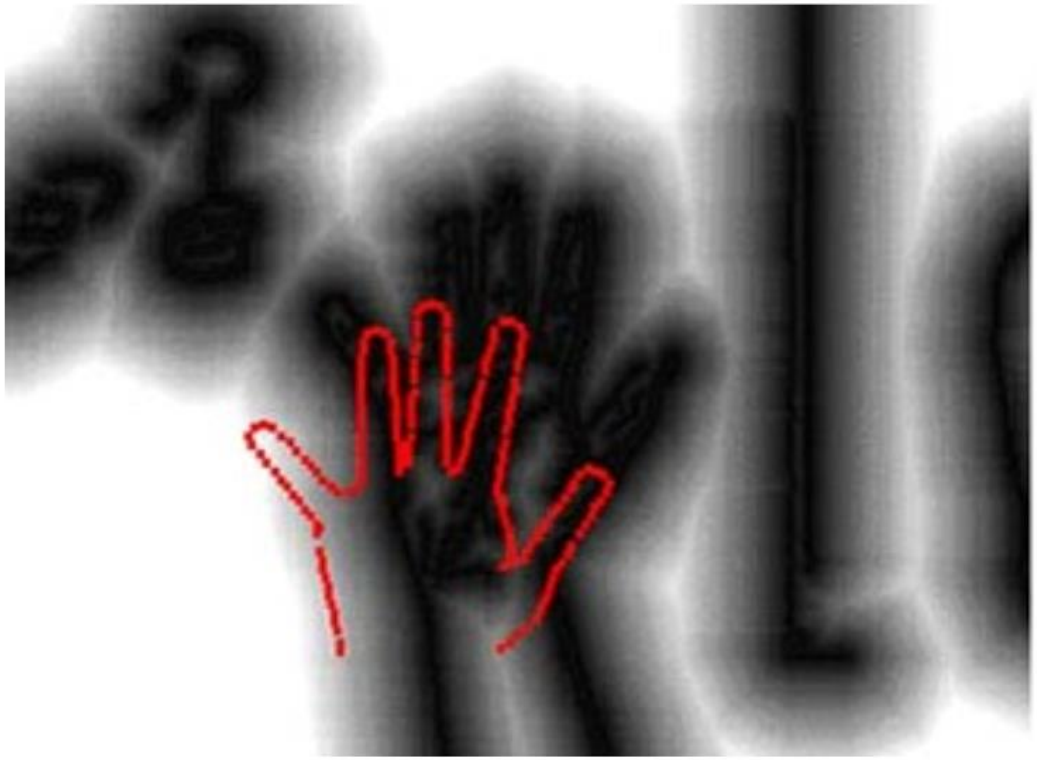


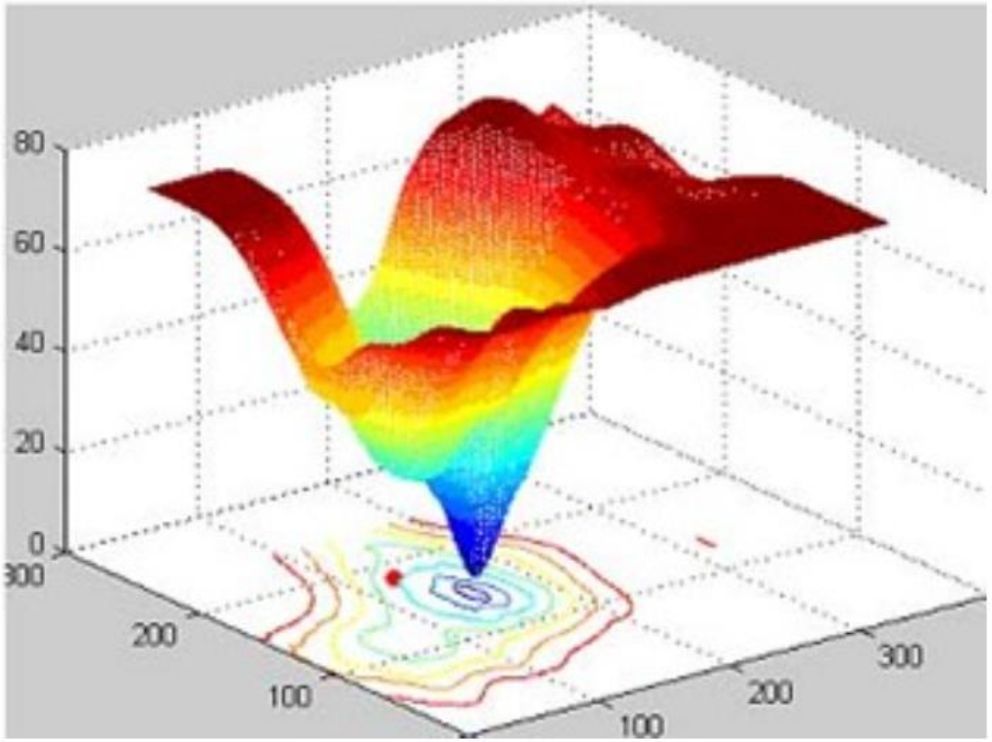
- Distance image provides a smooth cost function
- Efficient searching techniques can be used to find correct template

Chamfer Matching

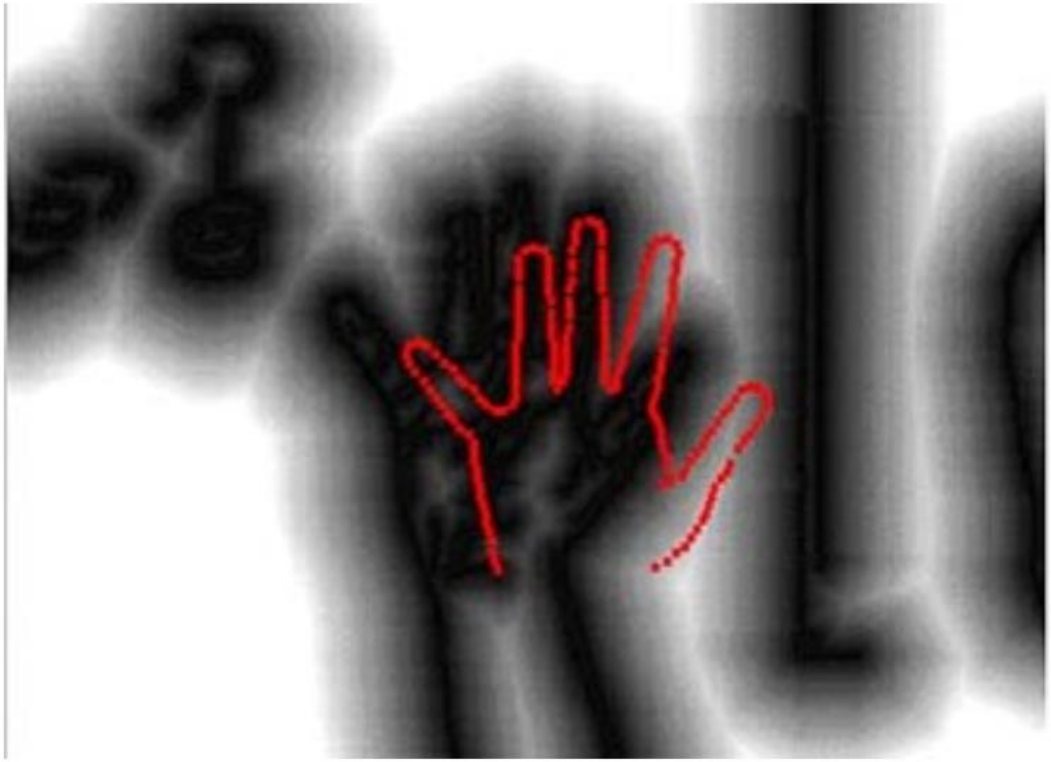


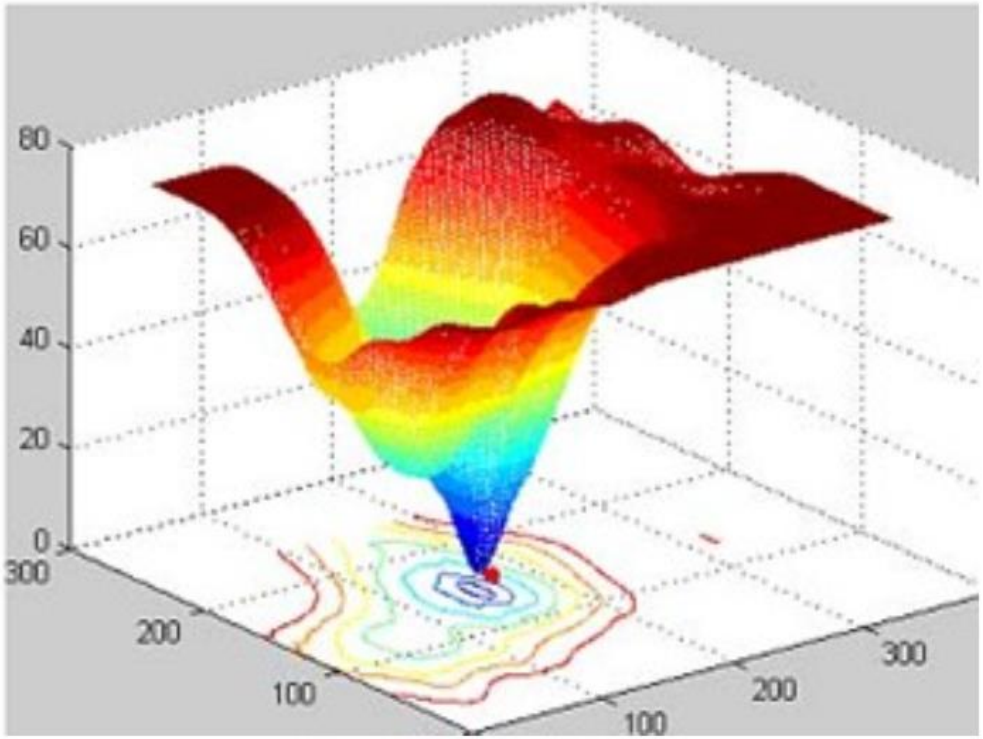
Chamfer Matching



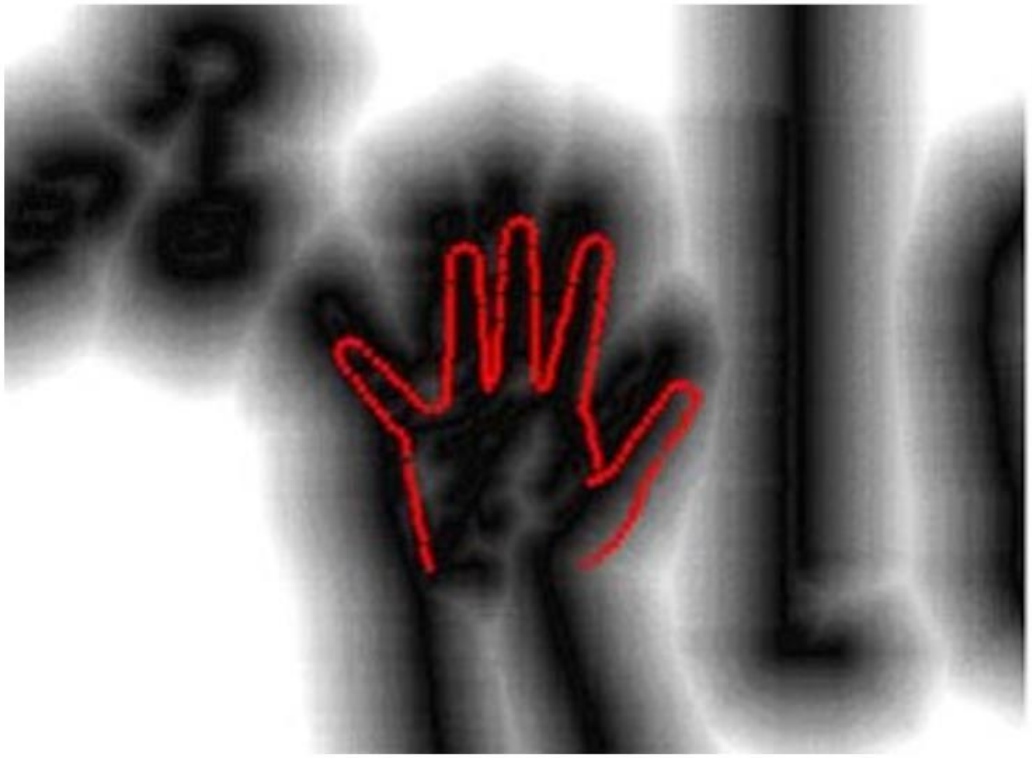


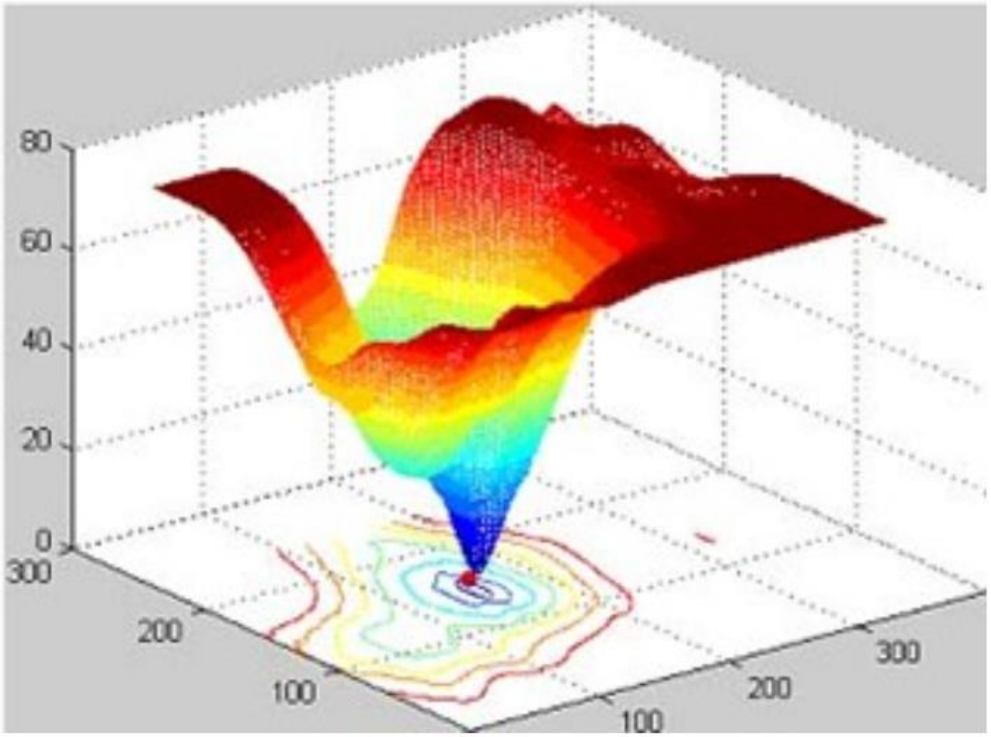
Chamfer Matching



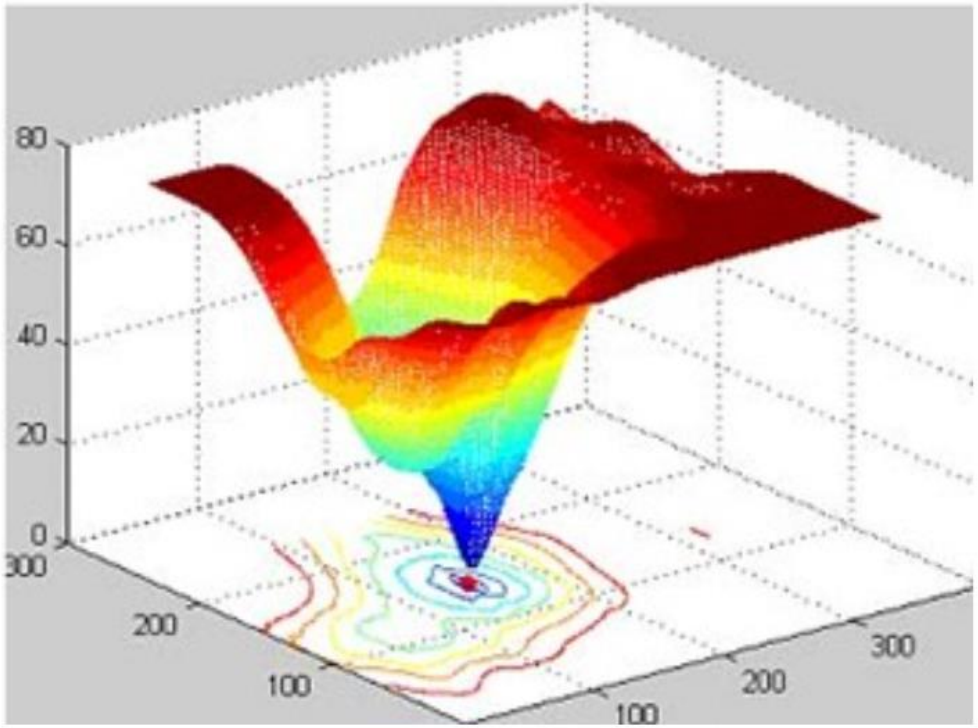


Chamfer Matching





Chamfer Matching



Applications: Hand Detection

- Initializing a hand model for tracking
  - Locate the hand in the image
  - Adapt model parameters
  - No skin color information used
  - Hand is open and roughly fronto-parallel

Results: Hand Detection

Original Shape Context | Shape Context with Continuity Constraint | Chamfer Matching

Courtesy: Chamfer Matching & Hausdorff Distance, Presented by Ankur Datta (Slides Courtesy: Mark Bouts, Arasanathan Thayananthan)

Results: Hand Detection

Original Shape Context

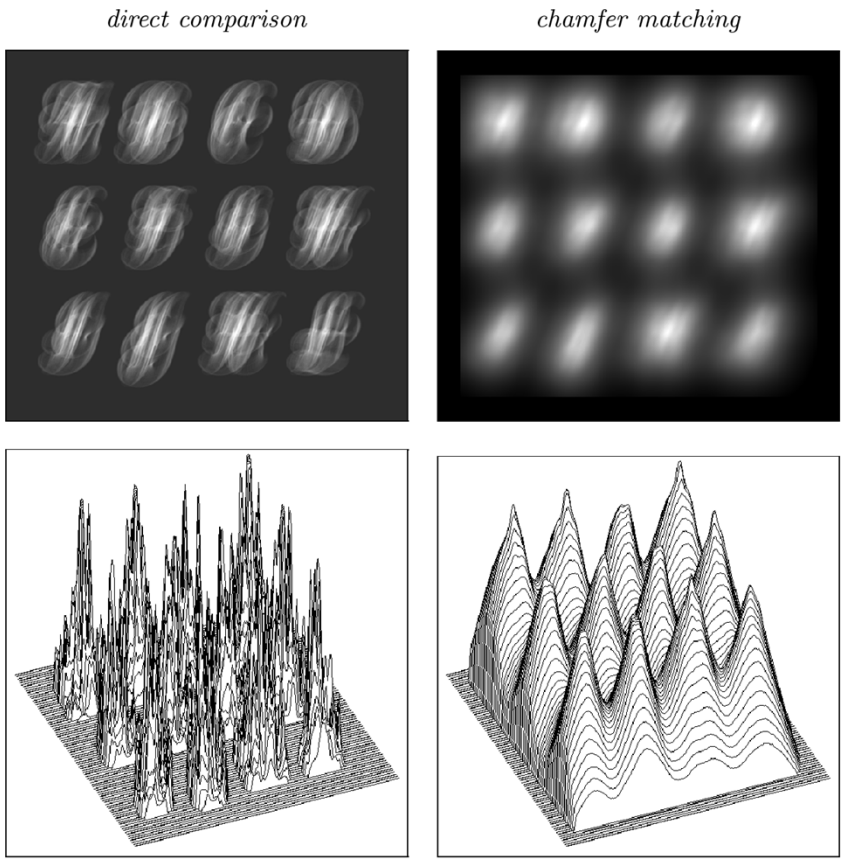
Shape Context with Continuity Constraint

Chamfer Matching

Courtesy: Chamfer Matching & Hausdorff Distance, Presented by Ankur Datta (Slides Courtesy: Mark Bouts, Arasanathan Thayananthan)

Comparing Direct Pixel comparison and Chamfer Matching

- Chamfer match score Q much smoother than direct comparison
  - Distinct peaks in places of high similarity



(a) (b)

- Chamfer Matching
  - Variant to scale and rotation
  - Sensitive to small shape changes
  - Need large number of template shapes

  But
  - Robust to clutter
  - Computationally cheap

References & Resources

- Look up Shape Contexts : http://cs-courses.mines.edu/csci508/schedule/14/ContourFeatures.pdf
- https://www2.cs.duke.edu/courses/fall15/compsci527/notes/convolution-filtering.pdf
- https://docs.opencv.org/master/d4/dc6/tutorial_py_template_matching.html
- http://www.cse.psu.edu/~rtc12/CSE486/lecture07.pdf
- Chamfer Matching & Hausdorff Distance, Presented by Ankur Datta (Slides Courtesy: Mark Bouts, Arasanathan Thayananthan)
- https://slidetodoc.com/chamfer-matching-hausdorff-distance-presented-by-ankur-datta/