한양대학교
HANYANG UNIVERSITY

Feature Detection

automatic 점 매칭 방법

Tae Hyun Kim

Image Matching

by Diva Sian

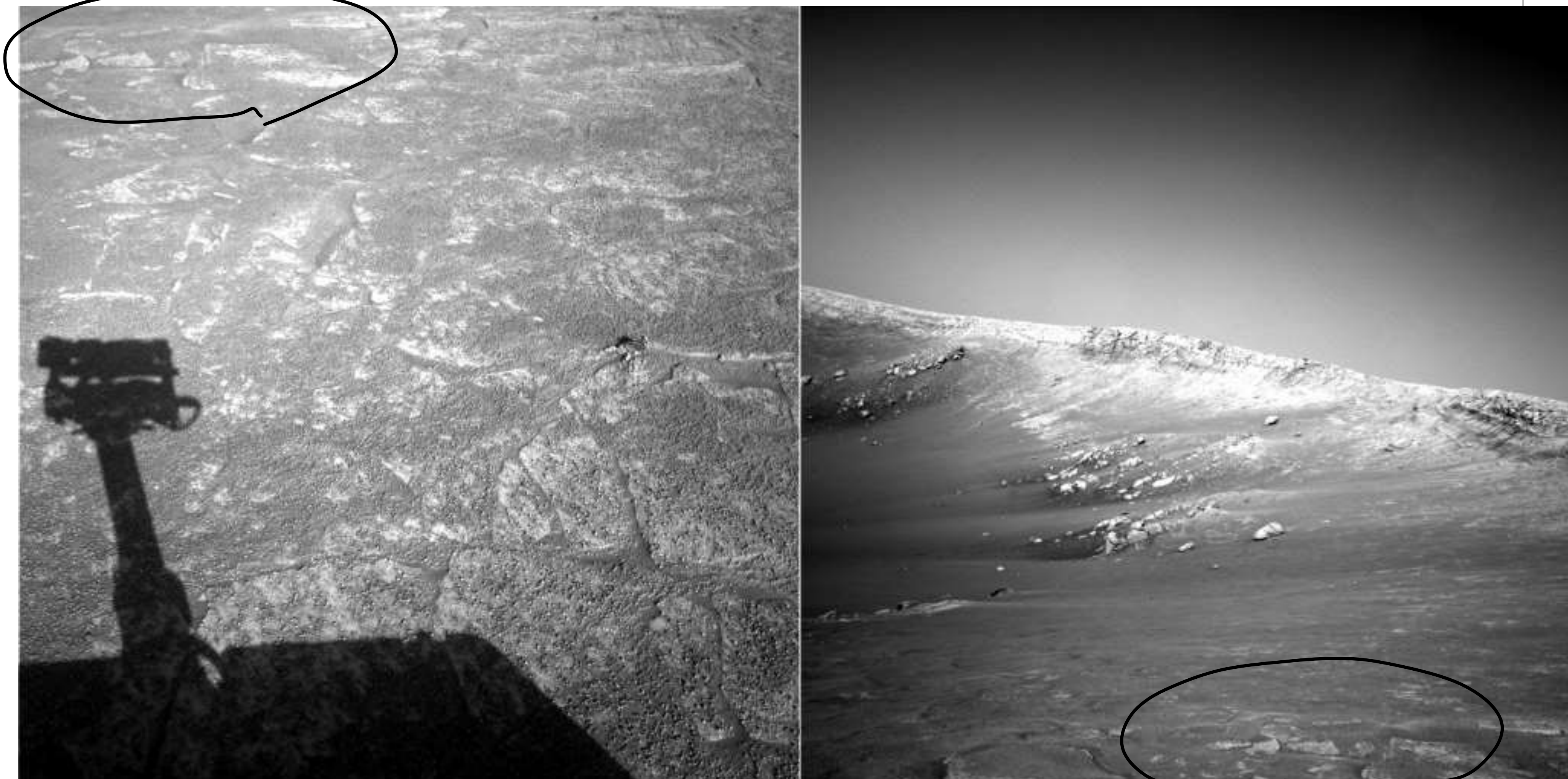
by swashford

Image Matching

by Diva Sian

by scgbt

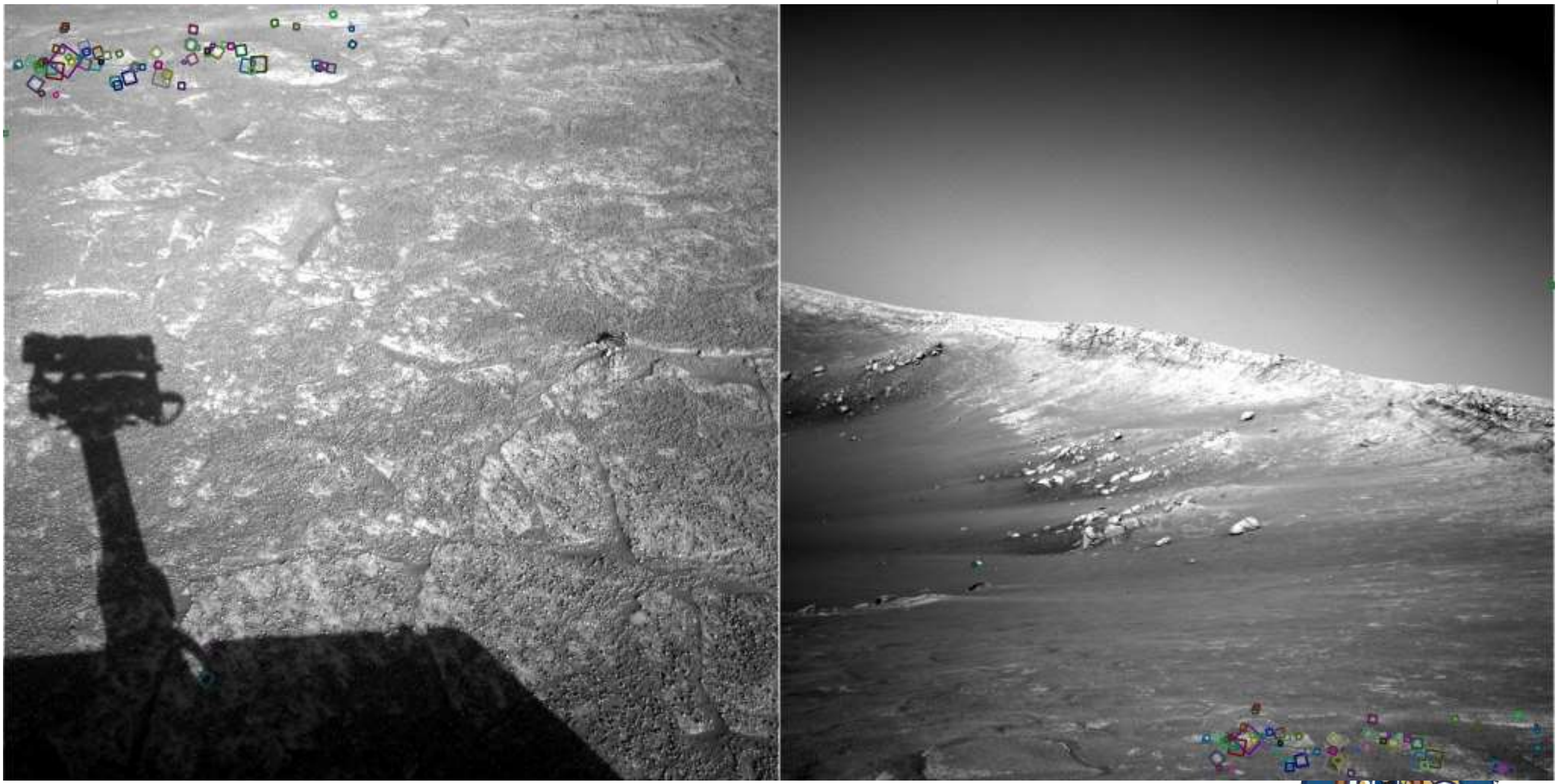
Image Matching



NASA Mars Rover images

overlapping 영역이 있다면 어느 점이 같은 점인지 찾아내기

Image Matching



NASA Mars Rover images with SIFT feature matches

Figure by Noah Snavely

How to Match?

- Human eye movements

What catches your interest?

이런 점들을
feature 라고 함
특징점 point

Yarbus eye tracking

How to Match?

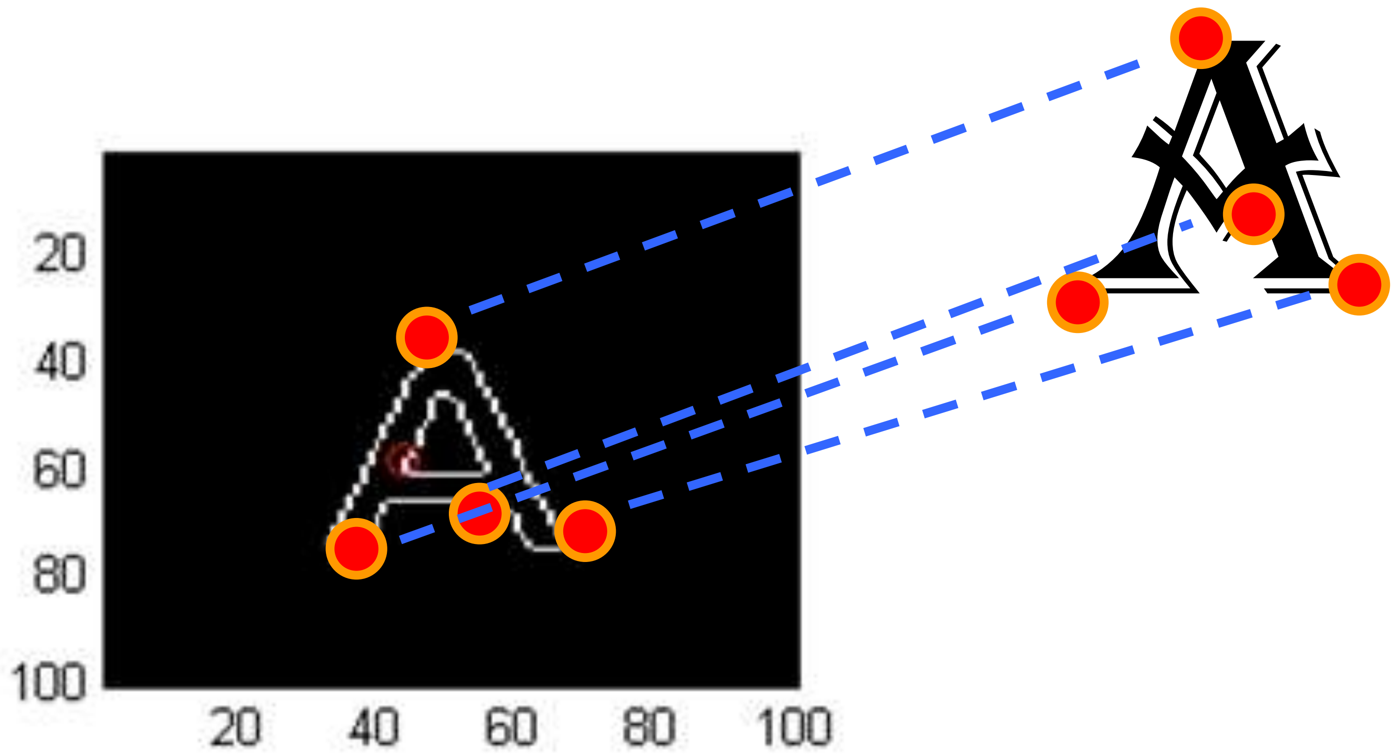
- Corresponding points

Example: automatic panoramas

스티칭
Image stitching

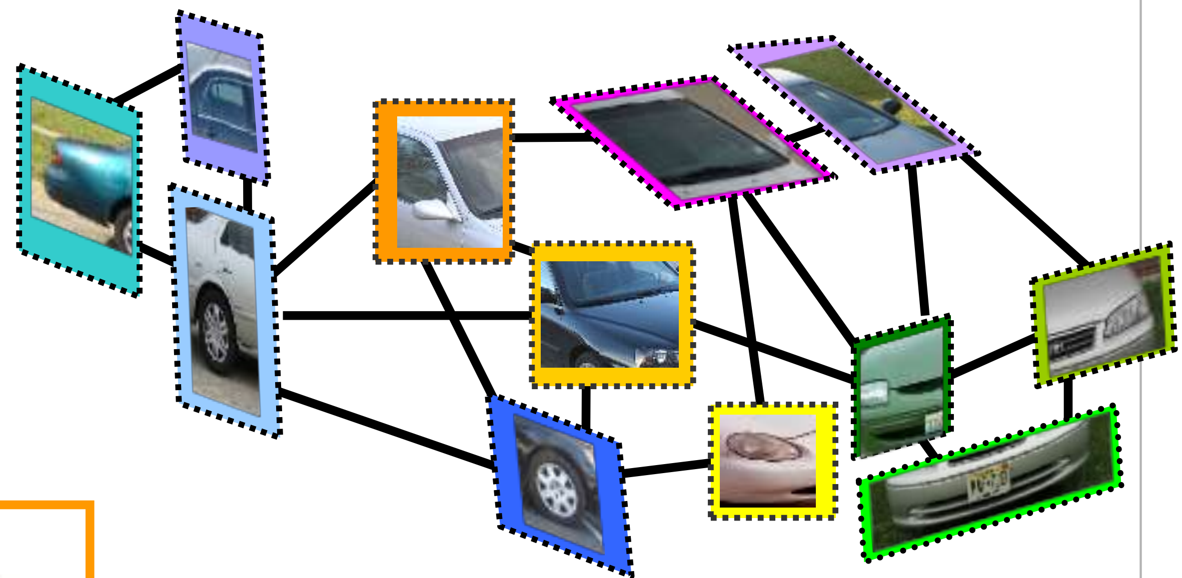
Credit: Matt Brown

Example: fitting an 2D shape template



Slide from Silvio Savarese

Example: fitting a 3D object model

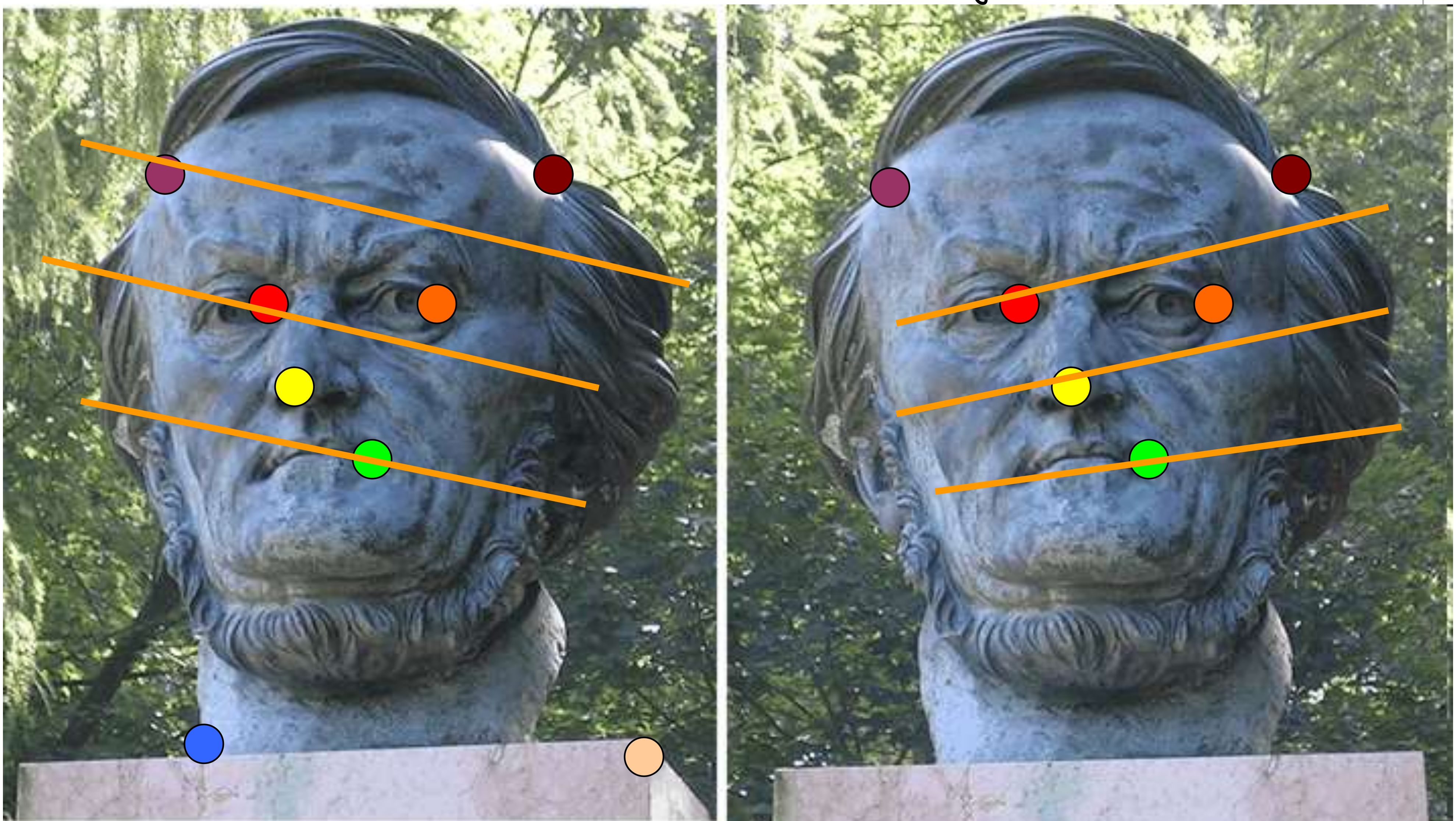


3차원 캐드 모델

Slide from Silvio Savarese

Example: estimating homography/fundamental matrix that corresponds two views

매치가 된 피처 포인트들이 주어지면
이미지의 3차원 Geometry 찾아낼수있음



Example: visual tracking

Tracking over consecutive video frames

← 사전지식에 기반

좋은 feature 찾기 (컬러, shape..)

요즘은 → 중요한 피처를 NN이 찾음
automatic하게

Other Applications

- Key (feature) points are used for:
  - Image alignment
  - 3D reconstruction
  - Motion tracking
  - Robot navigation
  - Indexing and database retrieval
  - Object recognition

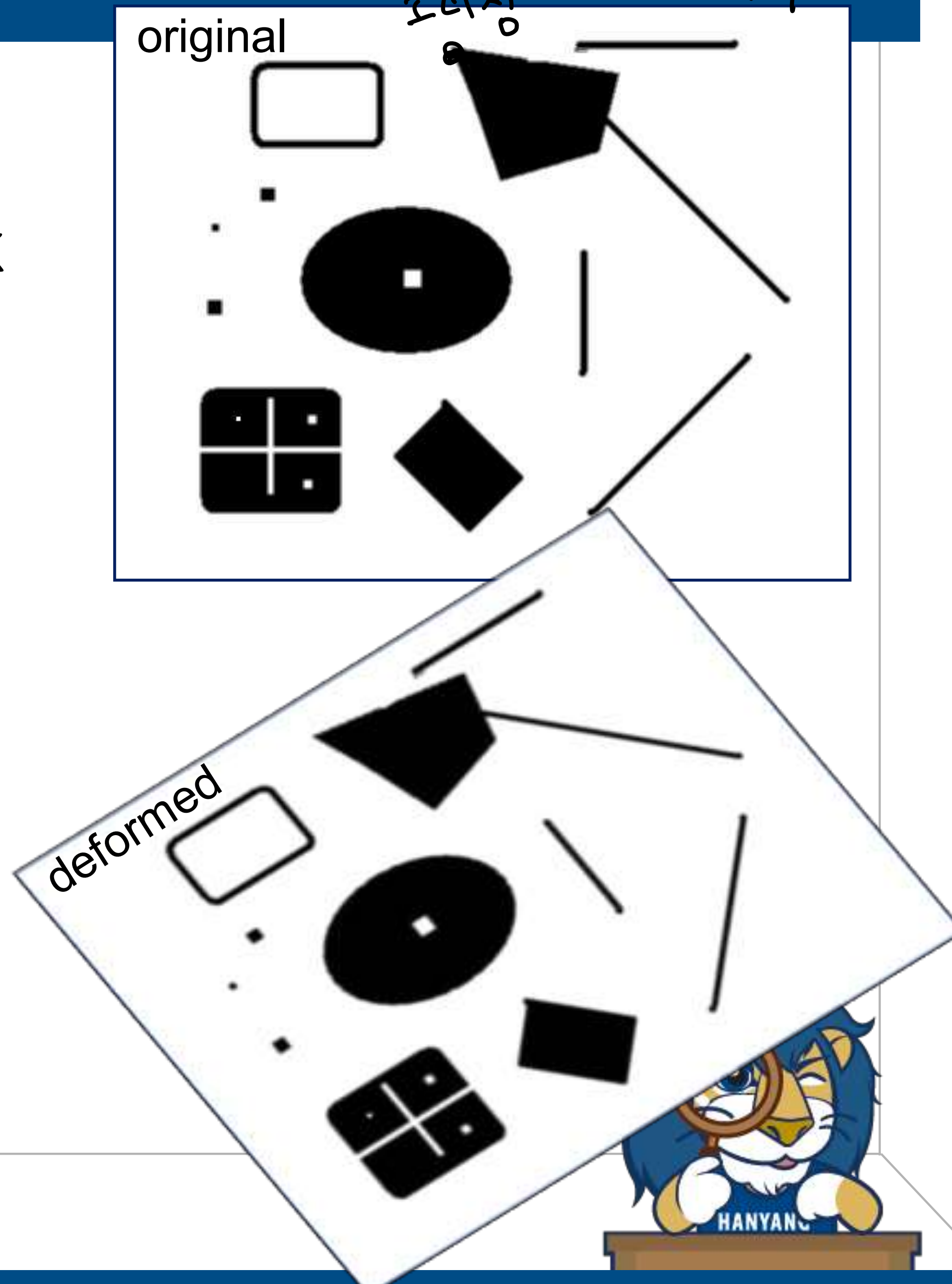
- Almost ***everywhere*** in computer vision

Choosing Interest Points

- Which points would you choose?
  - Suppose you have to click on some point, go away and come back after I deform the image, and click on the same points again.

변형시키다

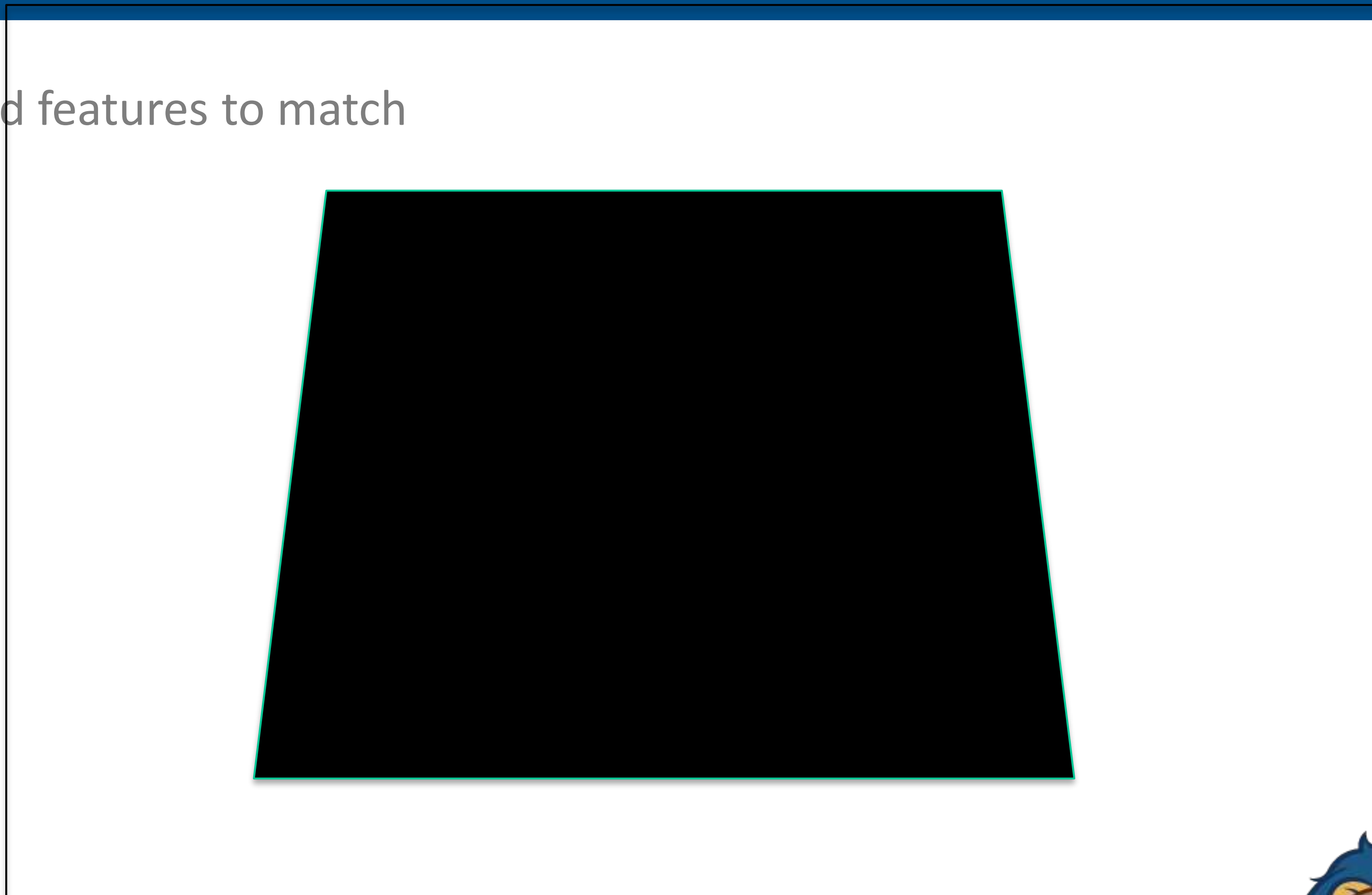
코너점 - 중요한 피쳐



corner detector

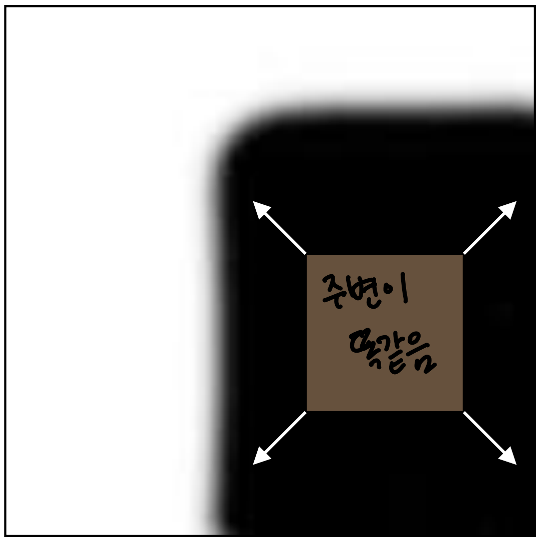
Intuition

- Good features to match

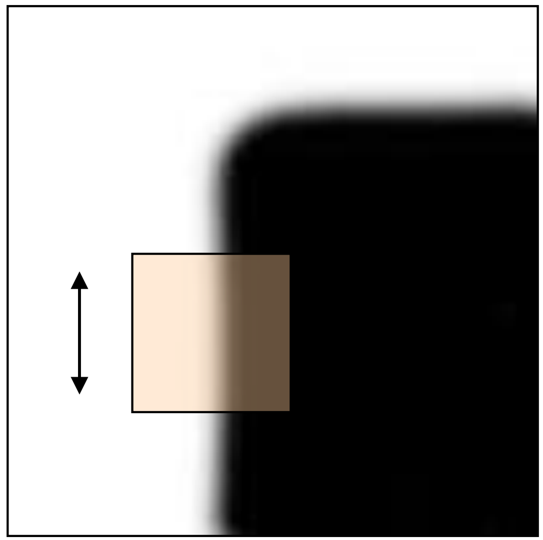


Corners

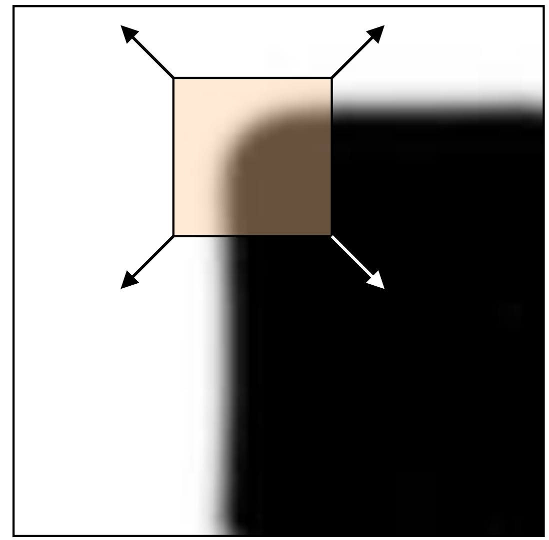
- How does the window change when you shift it?



"flat" region:
no change in
all directions



"edge":
no change along
the edge direction



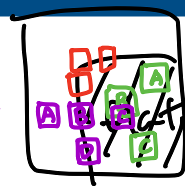
"corner":
significant
change in all
directions

Credit: A. Efros

Corners

- Why Corners?

A ≠ B ≠ C
B = D
→ edge 특징

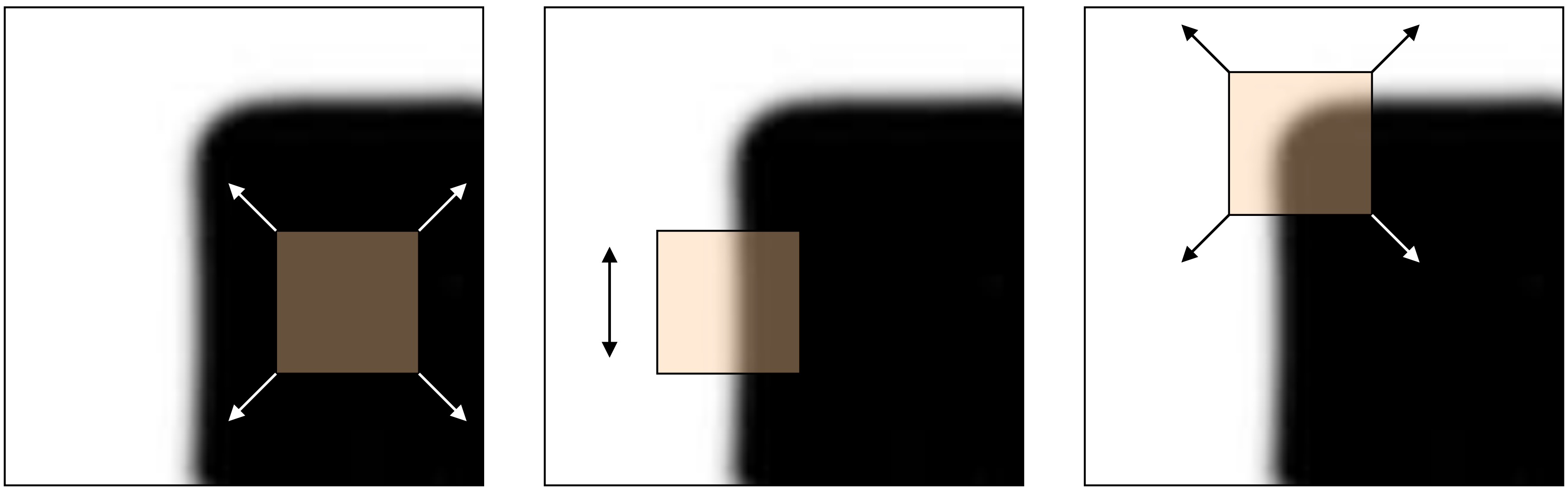


A ≠ B ≠ C
→ corner는 어떤 점을 잡더라도 다 다름

A,B,C patch same
→ flat region 특징

이걸 바탕으로
corner detection

  - Shifting the window in any direction causes a big change



"flat" region:
no change in
all directions

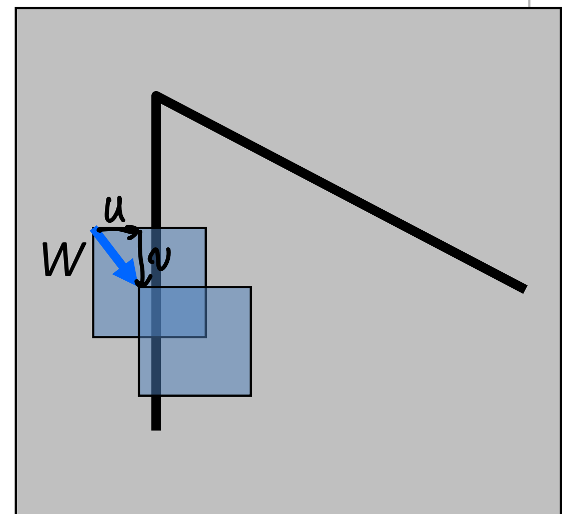
"edge":
no change along
the edge direction

"corner":
significant
change in all
directions

Credit: A. Efros

Harris corner detection: the math

- Consider shifting the window W by (u,v)
  - how do the pixel values in W change?
  - compare each pixel before and after by summing up the squared differences (SSD)
  - this defines an SSD "error" E(u,v):



두 패치의 difference 정의

$$E(u,v) = \sum_{(x,y)\in W} \left(I(x+u,y+v) - I(x,y)\right)^2$$

Error

밝기값의 차이

Harris corner detection: the math

- ***Small motion assumption*** (small u, v)

이미지에서 특정위치가
코너인지 아닌지 검사

Taylor Series expansion of *I*:

$$I(x+u, y+v) = I(x,y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \text{higher order terms}$$

If the motion (u,v) is small, then first order approximation is good enough;

linear, 2차형태로 근사 가능

$$\begin{aligned} I(x+u, y+v) &\approx I(x,y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v \\ &\approx I(x,y) + [I_x \ I_y] \begin{bmatrix} u \\ v \end{bmatrix} \end{aligned}$$

근사 위치가 좋음

x+u가 굉장히 가까우면 근사가 잘됨

$$\text{shorthand: } I_x = \frac{\partial I}{\partial x}$$

Harris corner detection: the math

근사가 유효할 때 : $x+u \approx u$: u가 굉장히 작을 때

- Under small motion assumption

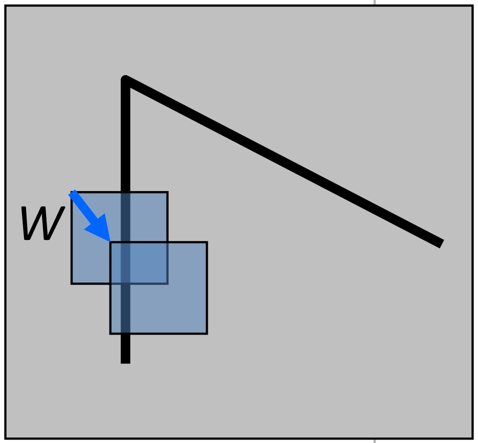
$$E(u,v) = \sum_{(x,y)\in W} \left(I(x+u, y+v) - I(x,y)\right)^2$$

$$\approx \sum_{(x,y)\in W} \left(I(x,y) + I_x(x,y)u + I_y(x,y)v - I(x,y)\right)^2$$

$$\approx \sum_{(x,y)\in W} \left(I_x(x,y)u + I_y(x,y)v\right)^2$$

Harris corner detection: the math

- *E(u,v)* is locally approximated as a *quadratic form*



$$E(u,v) \approx \sum_{(x,y)\in W} (I_x(x,y)u + I_y(x,y)v)^2$$

$$\approx \sum_{(x,y)\in W} (I_x^2 u^2 + 2 I_x I_y uv + I_y^2 v^2)$$

$$\approx Au^2 + 2Buv + Cv^2$$

$$A = \sum_{(x,y)\in W} I_x^2 \qquad B = \sum_{(x,y)\in W} I_x I_y \qquad C = \sum_{(x,y)\in W} I_y^2$$

Harris corner detection: the math

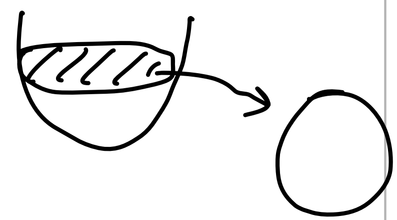
u, v가 엄청 작을때 유효함

가까운 패치들에서 가능

밝기값 1:1 비교

α β

했어야 했는데 → β 패치의 밝기 정보 알 필요 없음



$$E(u,v) \approx Au^2 + 2Buv + Cv^2$$

$$\approx \begin{bmatrix} u & v \end{bmatrix} \underbrace{\begin{bmatrix} A & B \\ B & C \end{bmatrix}}_{H} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$A = \sum_{(x,y)\in W} I_x^2$$

x축방향의 밝기값²

인접한 두 픽셀의 밝기값 차이²

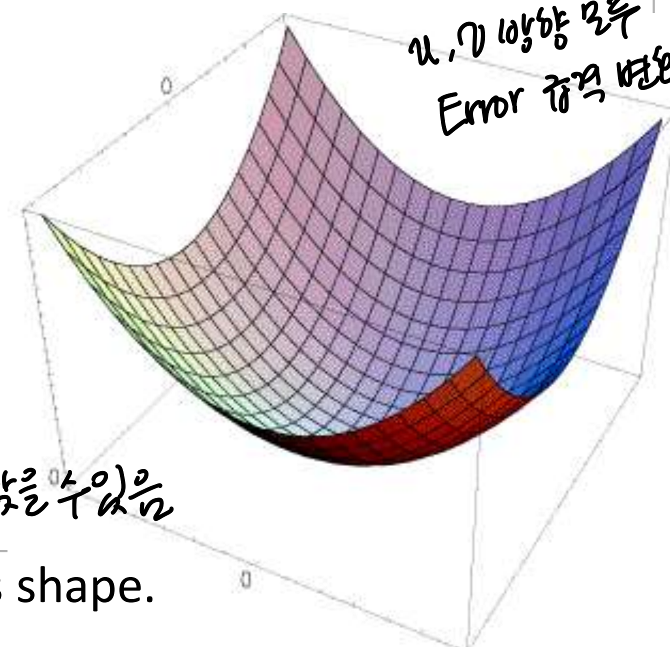
$$B = \sum_{(x,y)\in W} I_x I_y$$

$$C = \sum_{(x,y)\in W} I_y^2$$

H ↓ decompose

(두개의 eigen vector
〃 value) 찾을 수 있음

corner 점.
u, v 방향 모두
Error 급격 변화



Let's try to understand its shape.

Harris corner detection: the math

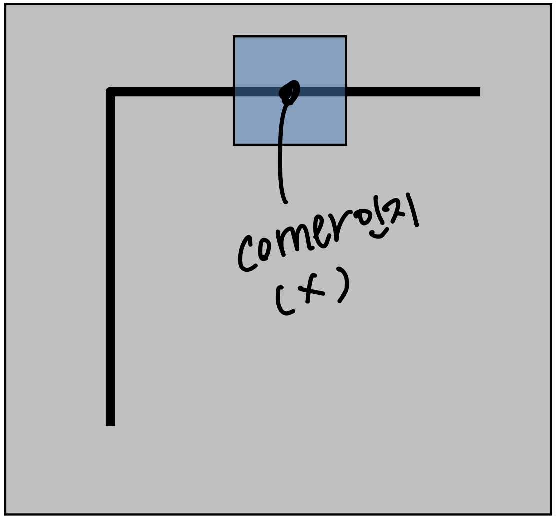
- At horizontal edges

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} \underbrace{\begin{bmatrix} A & B \\ B & C \end{bmatrix}}_{H} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$A = \sum_{(x,y)\in W} I_x^2$$

$$B = \sum_{(x,y)\in W} I_x I_y$$

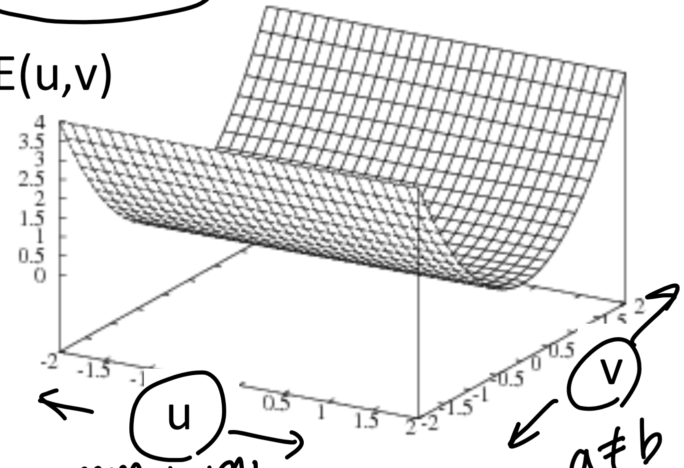
$$C = \sum_{(x,y)\in W} I_y^2$$



Horizontal edge: $I_x = 0$

$$H = \begin{bmatrix} 0 & 0 \\ 0 & C \end{bmatrix}$$

E(u,v)



error
E

patch

Harris corner detection: the math

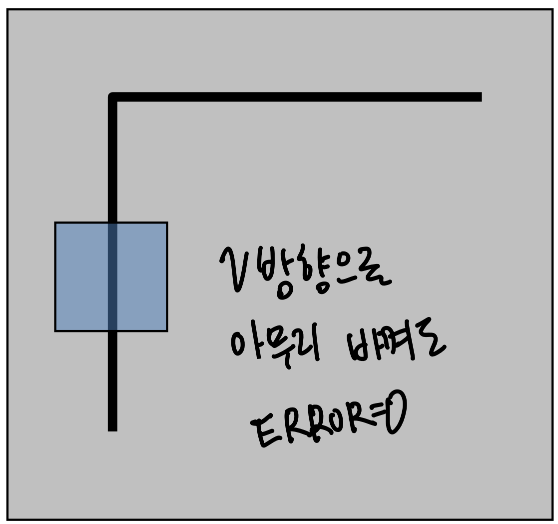
- At vertical edges

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} \underbrace{\begin{bmatrix} A & B \\ B & C \end{bmatrix}}_{H} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$A = \sum_{(x,y)\in W} I_x^2$$

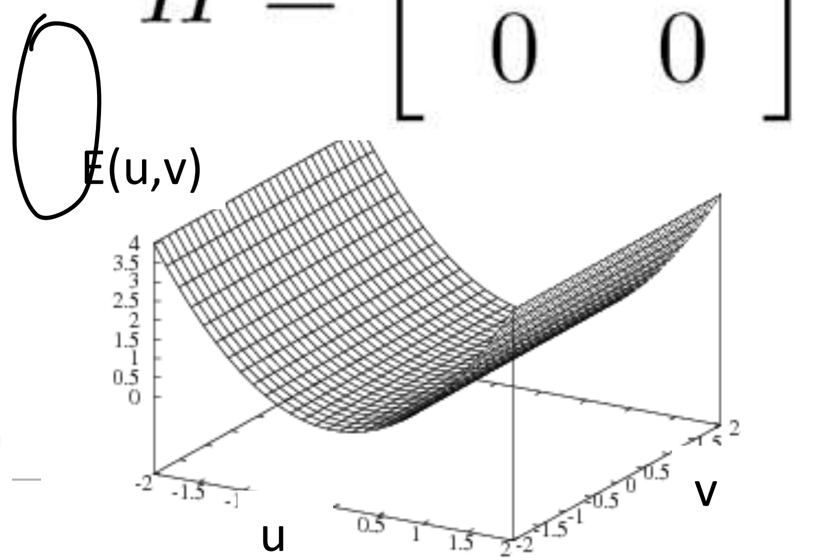
$$B = \sum_{(x,y)\in W} I_x I_y$$

$$C = \sum_{(x,y)\in W} I_y^2$$

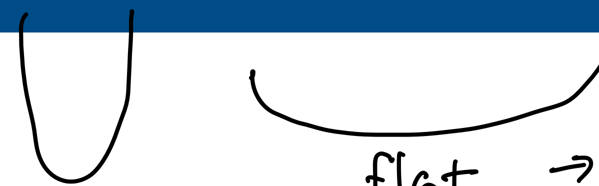


Vertical edge: $I_y = 0$

$$H = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}$$



Harris corner detection



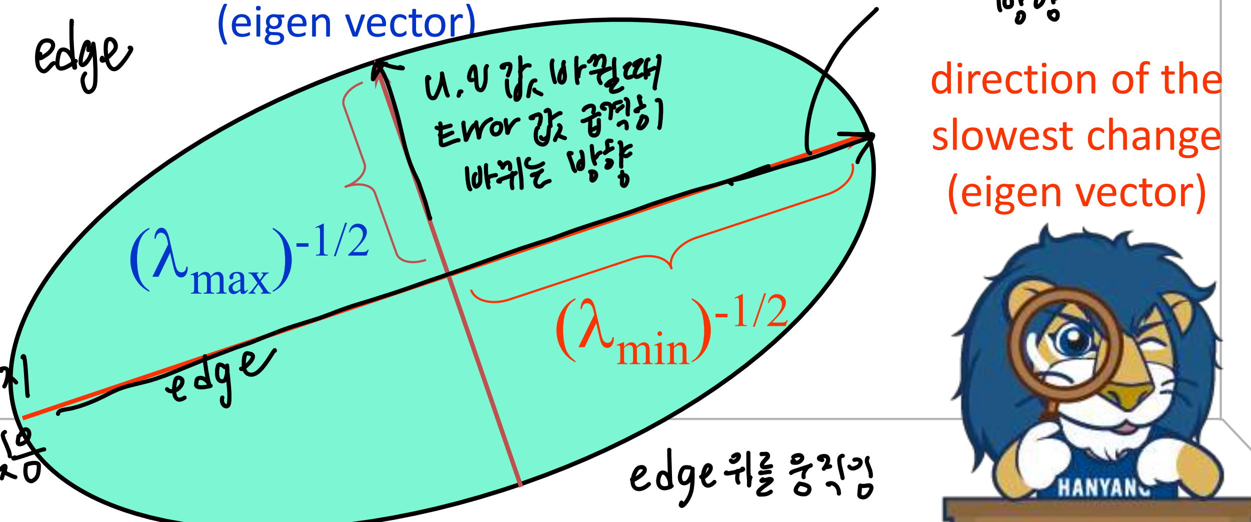
flat → big

- General case
  - Shape of H tells us something about the *distribution of gradients* around a pixel
  - We can visualize *H* as an ellipse where the axis lengths determined by the *eigenvalues* of *H,* and orientations determined by the *eigenvectors* of *H*

λ_{max}, λ_{min} : eigenvalues of *H*

Ellipse equation:

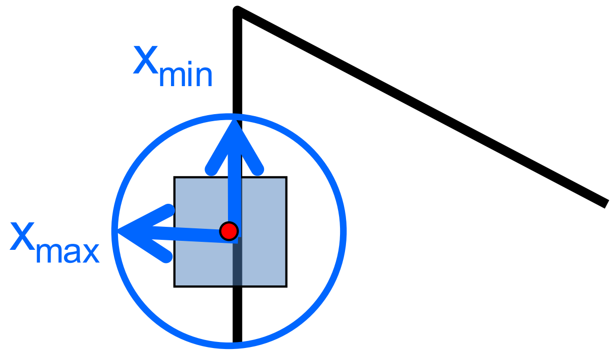
$$[u \;\; v] \; H \begin{bmatrix} u \\ v \end{bmatrix} = \text{const}$$



$H \xrightarrow[\text{decompose}]{}$ eigenvalue / vector → 코너인지 알수있음

Harris corner detection

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} \underbrace{\begin{bmatrix} A & B \\ B & C \end{bmatrix}}_{\substack{H \\ 2\times 2}} \begin{bmatrix} u \\ v \end{bmatrix}$$



$$Hx_{\max} = \lambda_{\max} x_{\max}$$

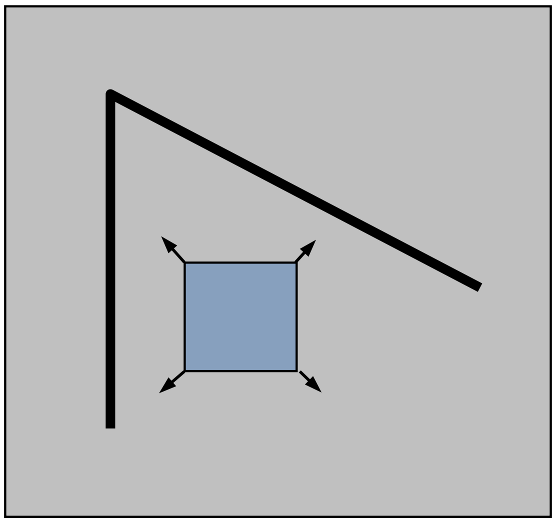
$$Hx_{\min} = \lambda_{\min} x_{\min}$$

- Eigenvalues and eigenvectors of H
  - Define shift directions with the smallest and largest change in error
  - x_{max} = direction of largest change in *E*
  - λ_{max} = amount of change in direction x_{max}
  - x_{min} = direction of smallest change in *E*
  - λ_{min} = amount of change in direction x_{min}

eigen vector x:
Error가 엄청 빨리/천천히 바뀌는 방향
eigen value λ:
타원의 장축/단축 길이 결정

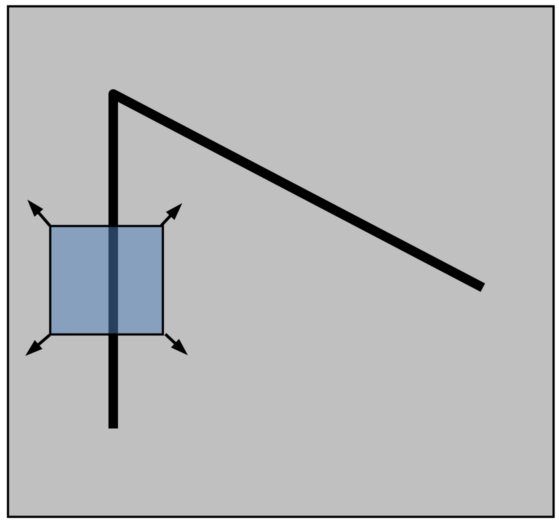
Harris corner detection

Small λ_{max} & small λ_{min}



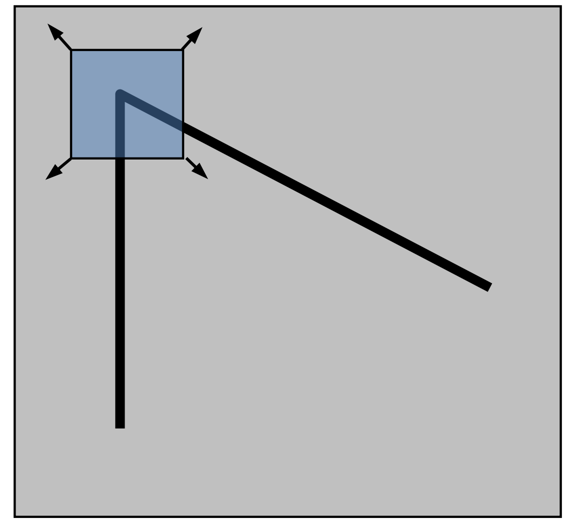
"flat" region:
no change in all directions

Large λ_{max} & small λ_{min}



"edge":
no change along the edge direction

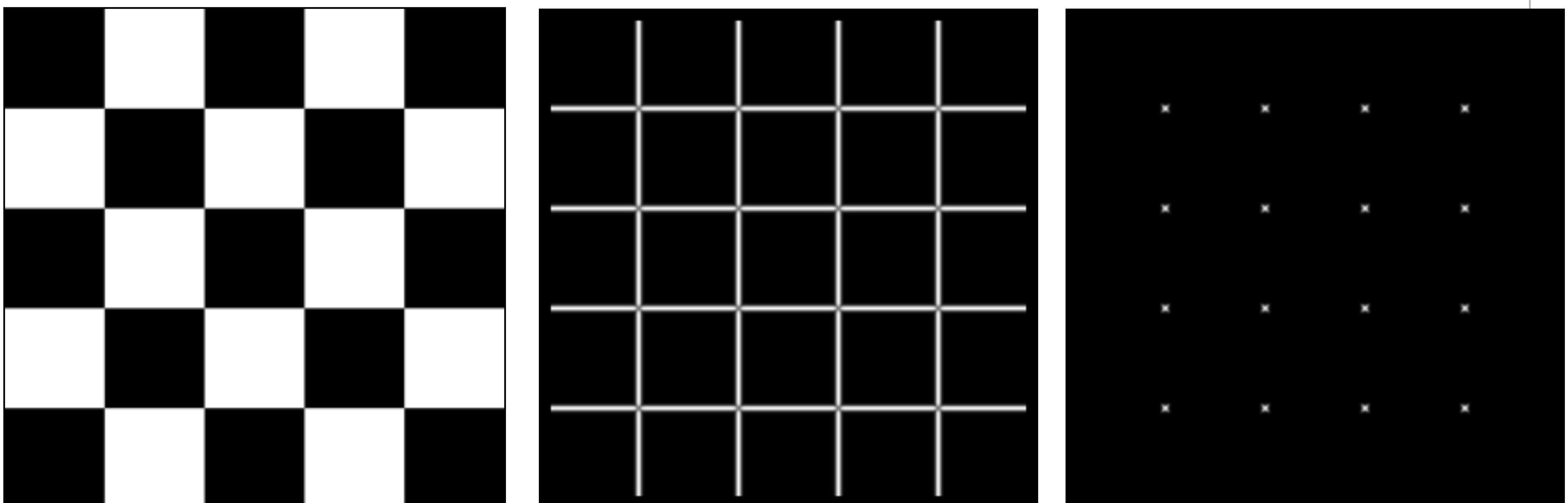
Large λ_{max} & large λ_{min}



"corner":
significant change in all directions

Harris corner detection

- Real Example
  - At corner points, both λ_{max} and λ_{min} are large



I $\lambda_{\max}$ $\lambda_{\min}$

eigenvalue sorting

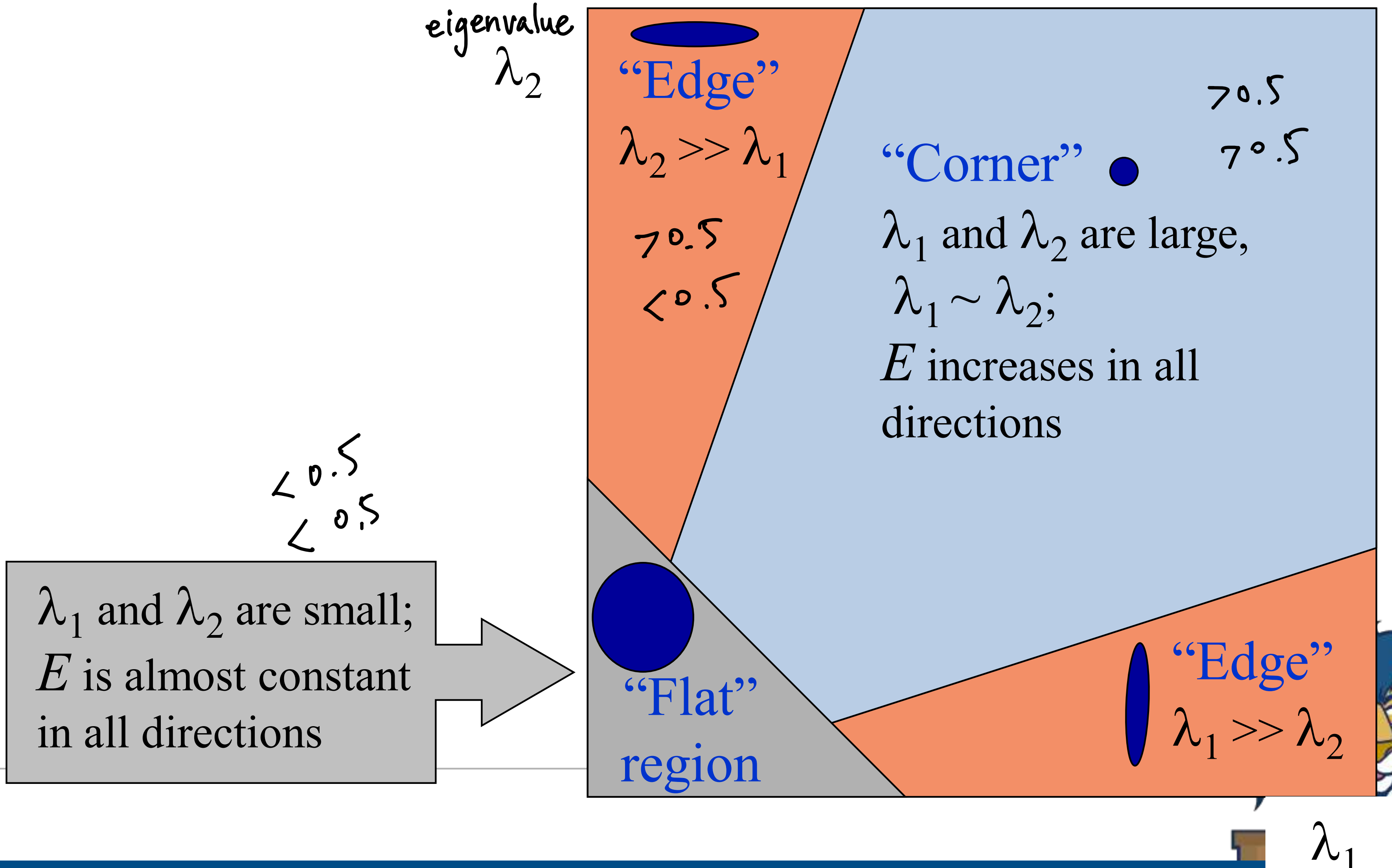
→ edge, corder 부분에서 값이 큼

corner에서만 살아남음

⇒ λmin 값이 크면 코너

Harris corner detection

- Two eigen values and corresponding region in the given image



Harris corner detection

- Harris operator (metric)
  - Called the “Harris Corner Detector” or “Harris Operator”
    - Shi-Tomasi corner detector: $\lambda_{\min}$
    - Harris and Stephens: $\det(H) - \alpha \cdot trace(H)$
    - Harmonic mean: $\det(H) / trace(H)$
  - Among lots metrics, this is one of the most popular
    - Harris response

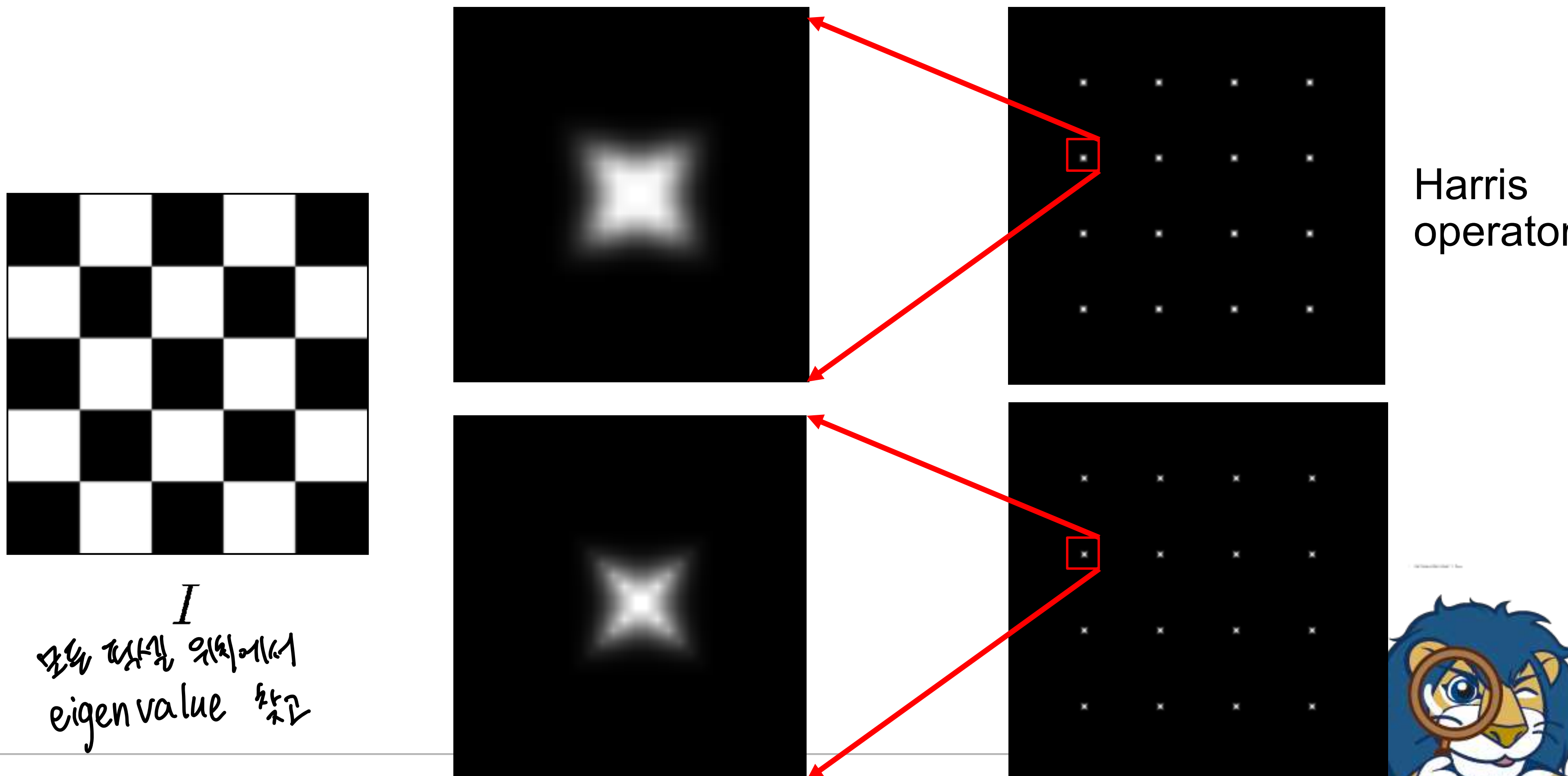
$$f = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2} = \frac{determinant(H)}{trace(H)}$$

(λ가 둘다 크면 값이 큼)

Harris corner detection

- Harris operator value $f \approx \lambda_{\min}$

코너 검출에 사용　f가 특정 수 이상이면 코너

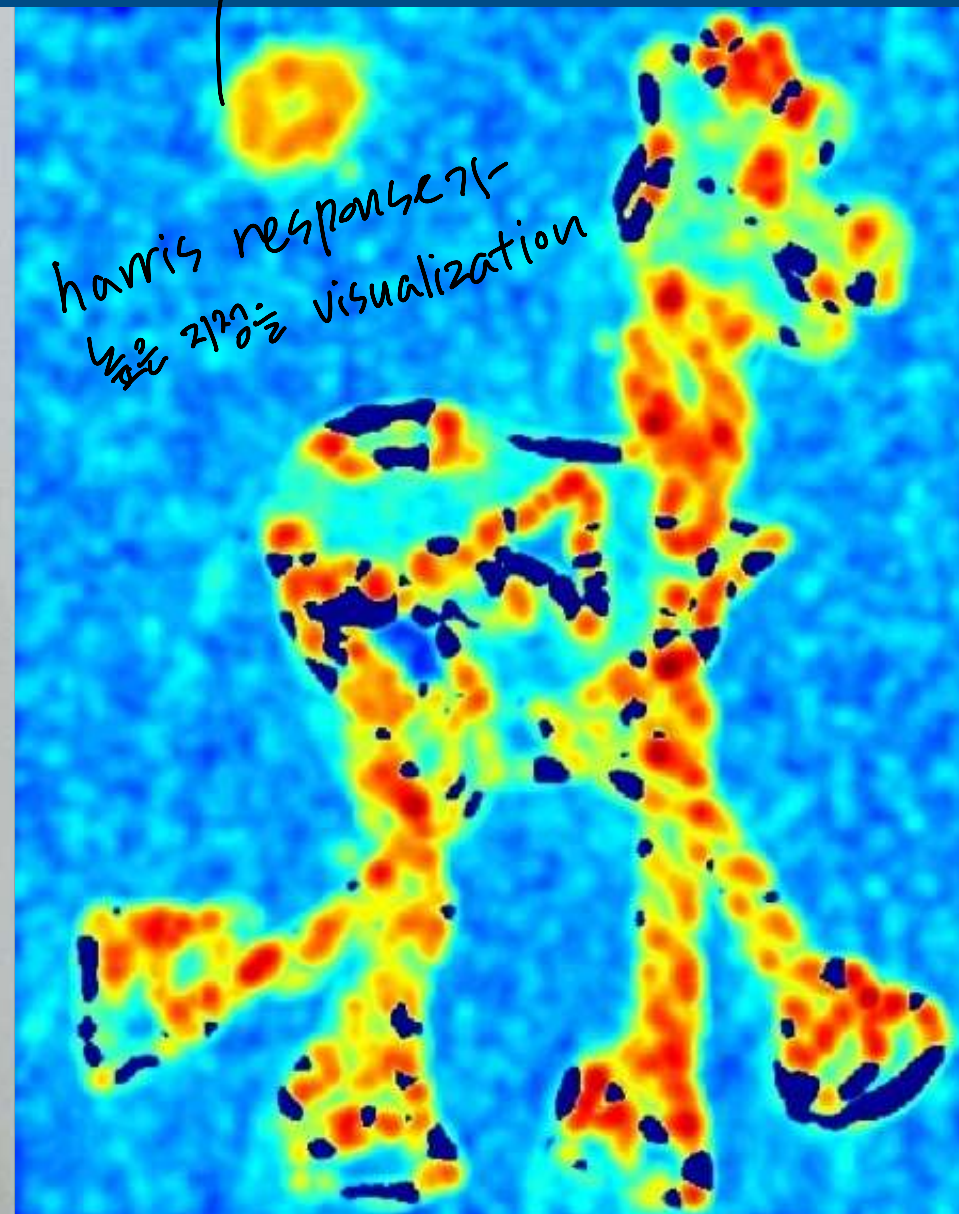


모든 픽셀 위치에서 eigenvalue 찾고

Harris detector example

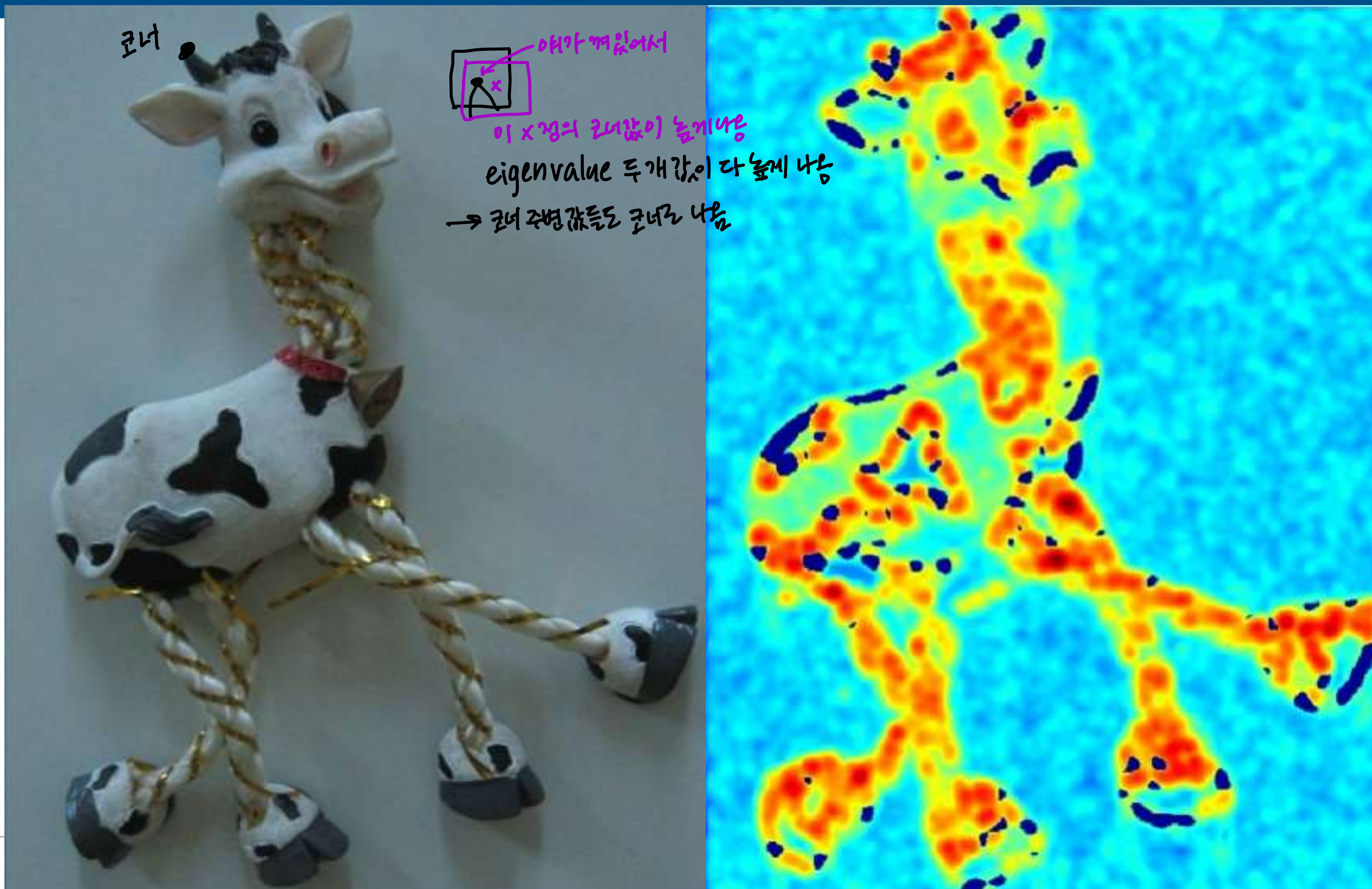
Harris detector example

빨간 부분이 cornerness가 높은 부분



harris response가
높은 지점을 visualization

Harris detector example



코너

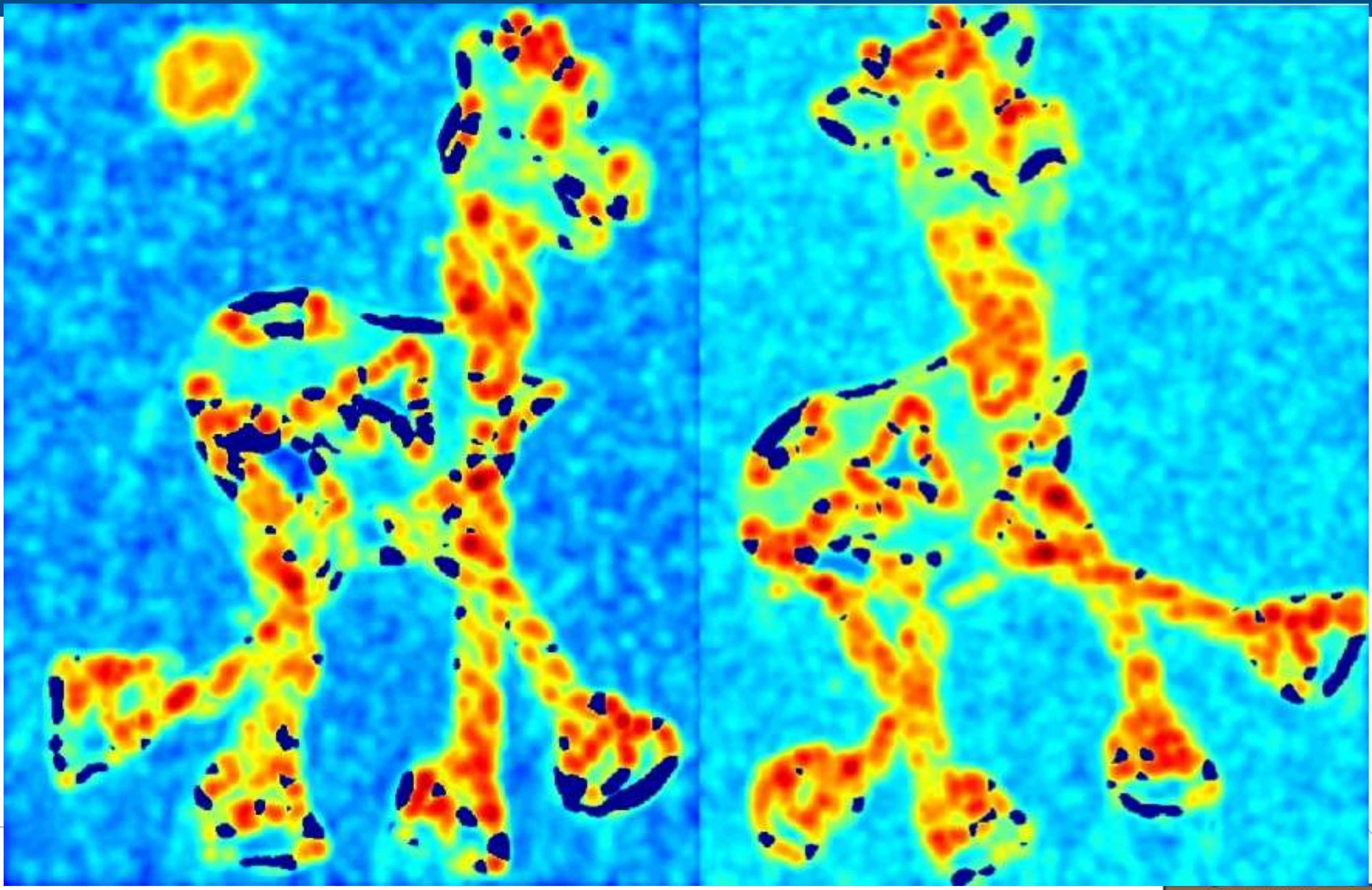
얘가 껴있어서

이 x점의 코너값이 높게나옴

eigenvalue 두개값이 다 높게 나옴

→ 코너 주변값들도 코너로 나옴

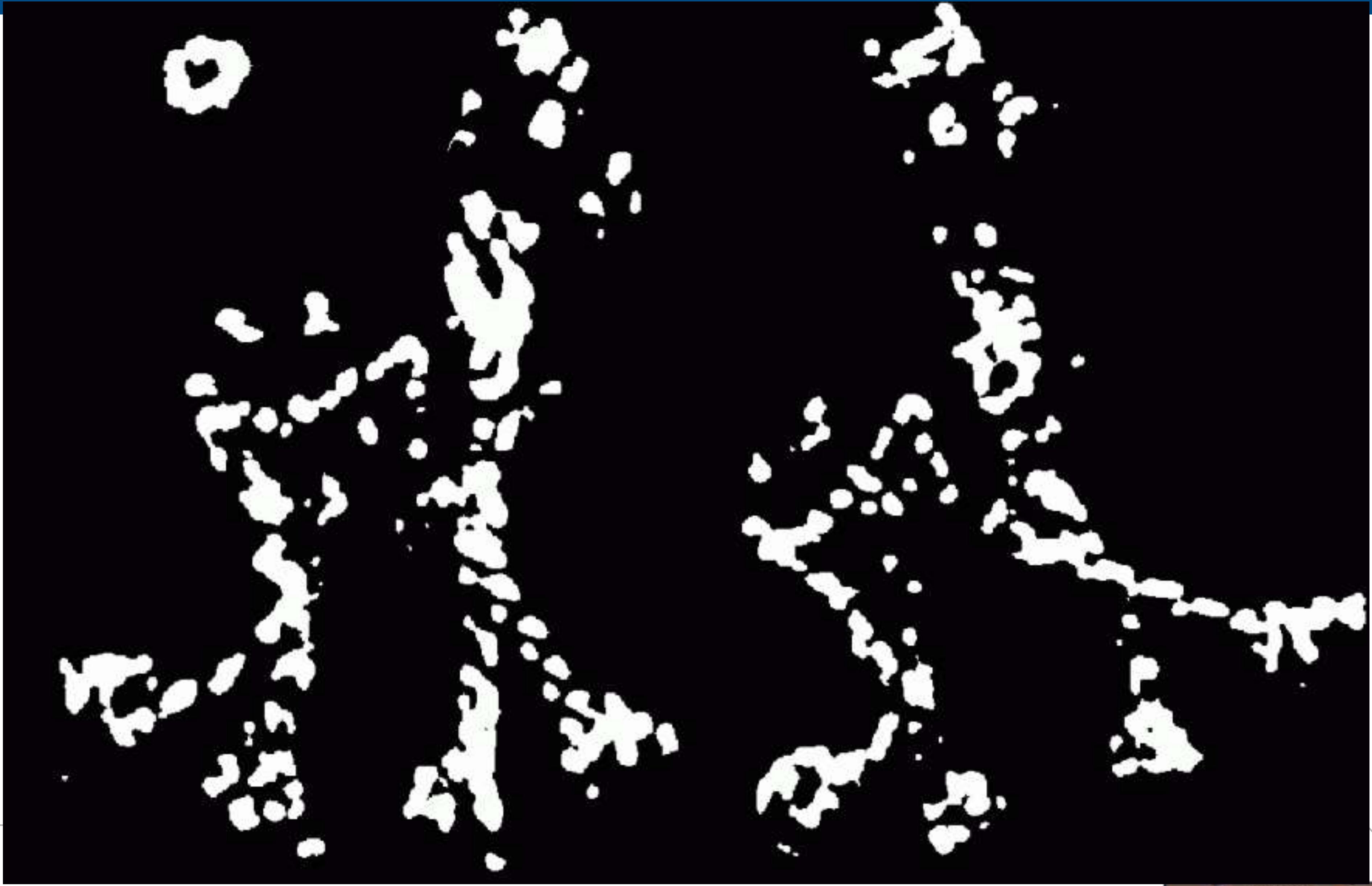
Harris detector example



Harris detector example

특정 threshold로 harris response...

corner 검출 넓은 영역으로 cornerness가 높게 나옴



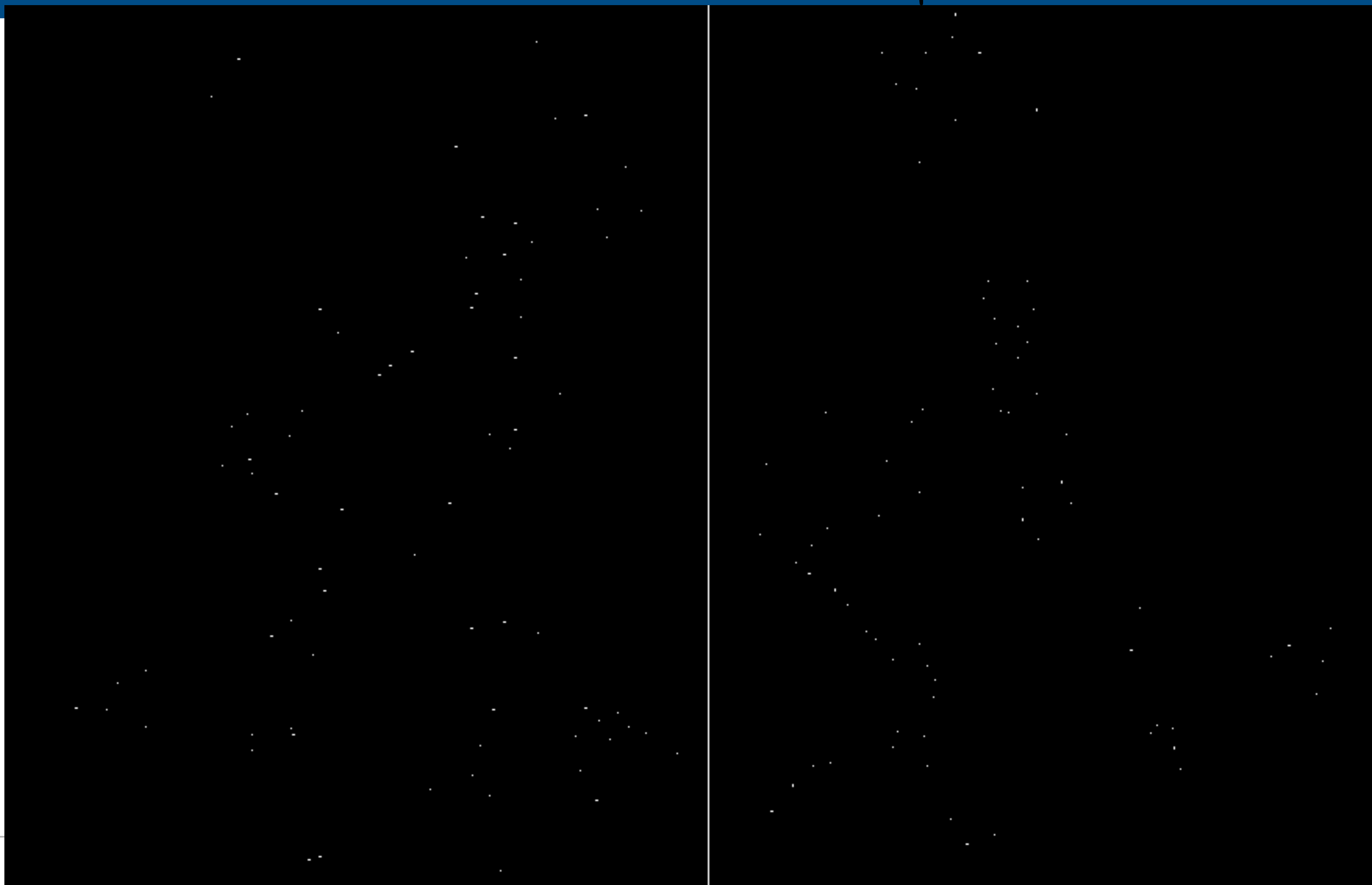
Local maxima of f

NMS

Non Maximul Surpression

젤 코너값이 높은 것 빼고 다 버림

harrison response 값이 큰 애들만 남김



How to improve Harris detector?

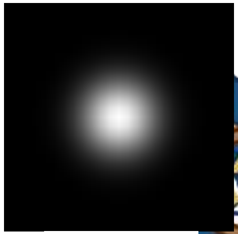
skip

- Weighting the derivatives
  - In practice, using a simple and equally weighted window *W* doesn't work too well

$$H = \sum_{(x,y)\in W} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

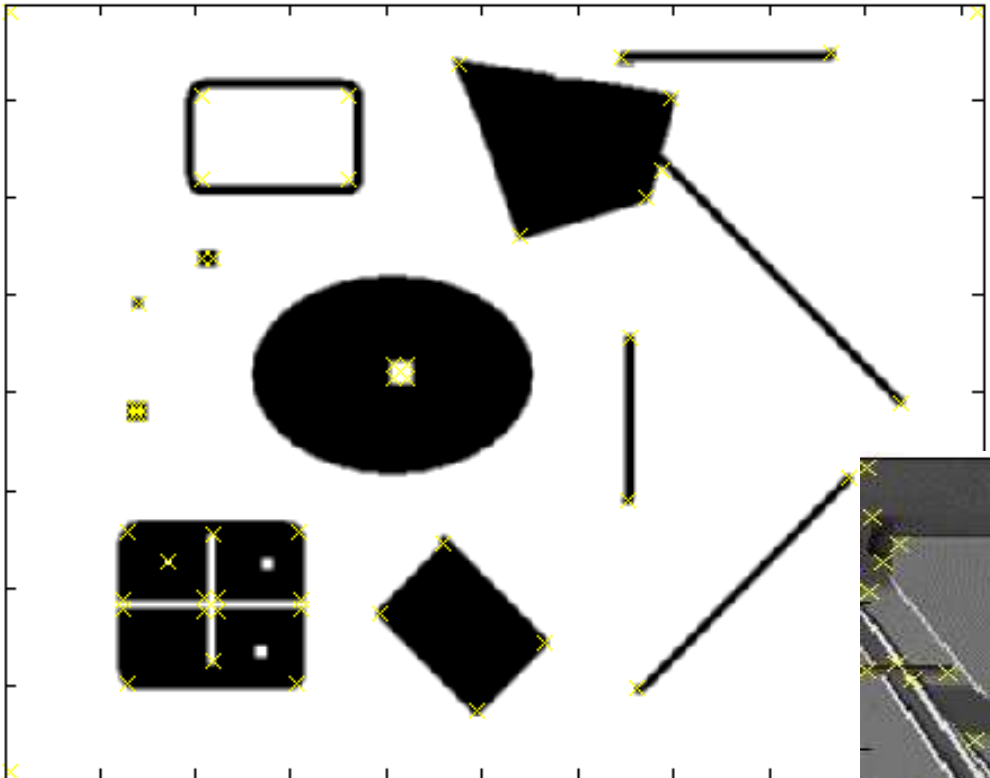
  - Instead, we'll *weight* each derivative value based on its distance from the center pixel

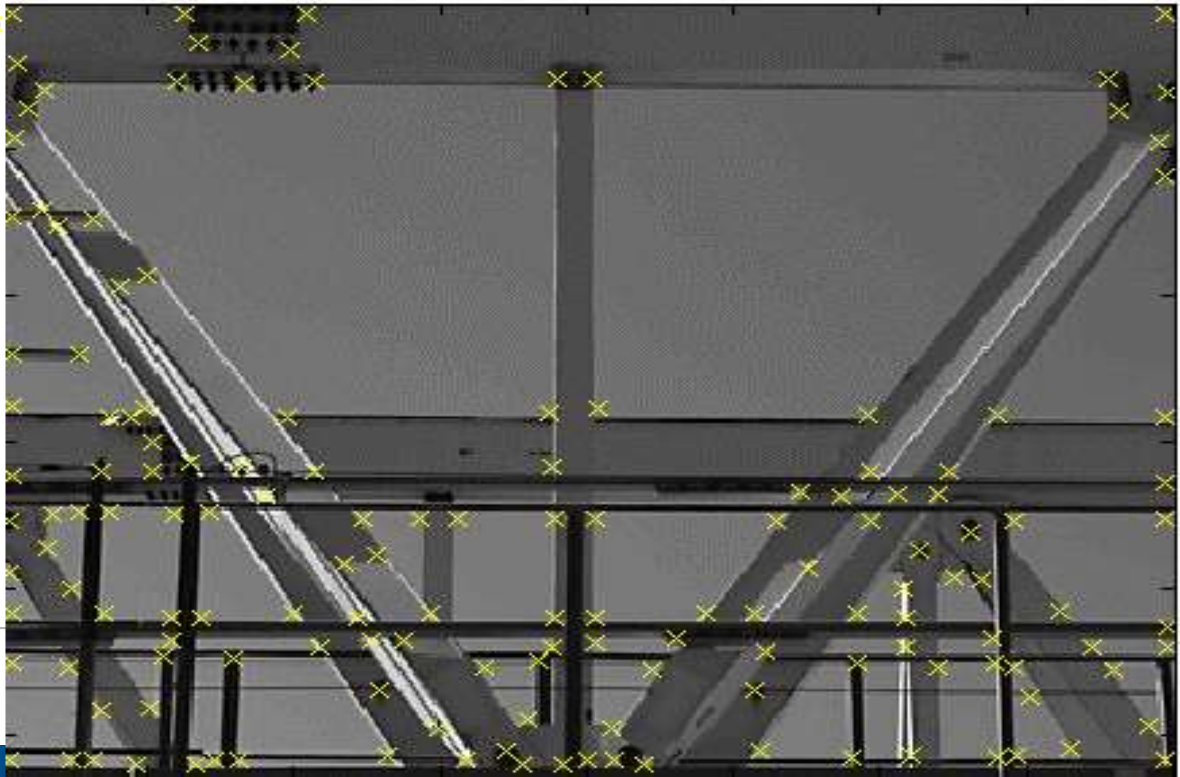
$$H = \sum_{(x,y)\in W} w_{x,y} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$



$w_{x,y}$

Harris detector: results



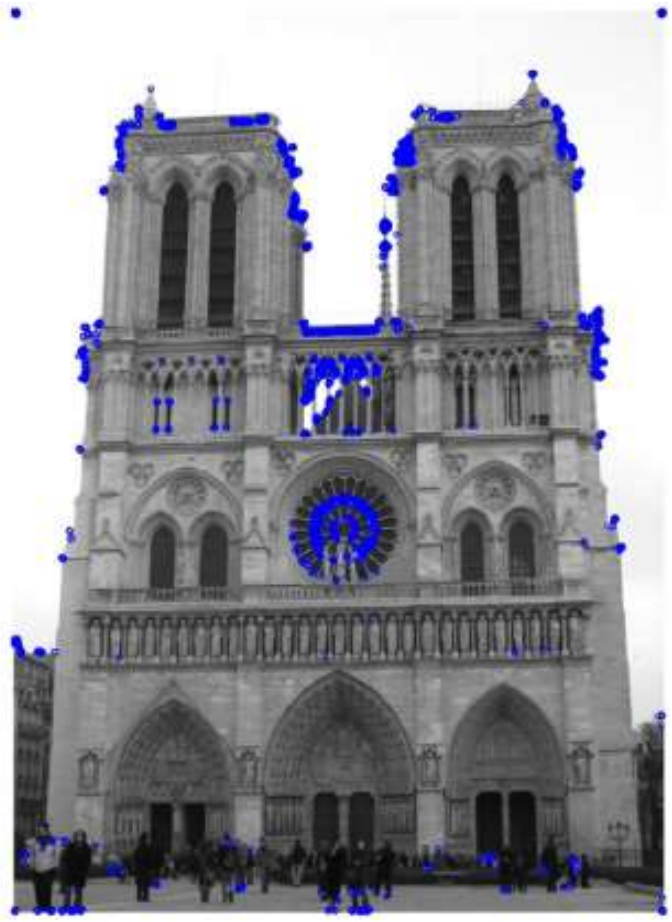


Effect: A very precise corner detector.

Harris detector: results

Adaptive Non-maximal Suppression (CVPR'05)

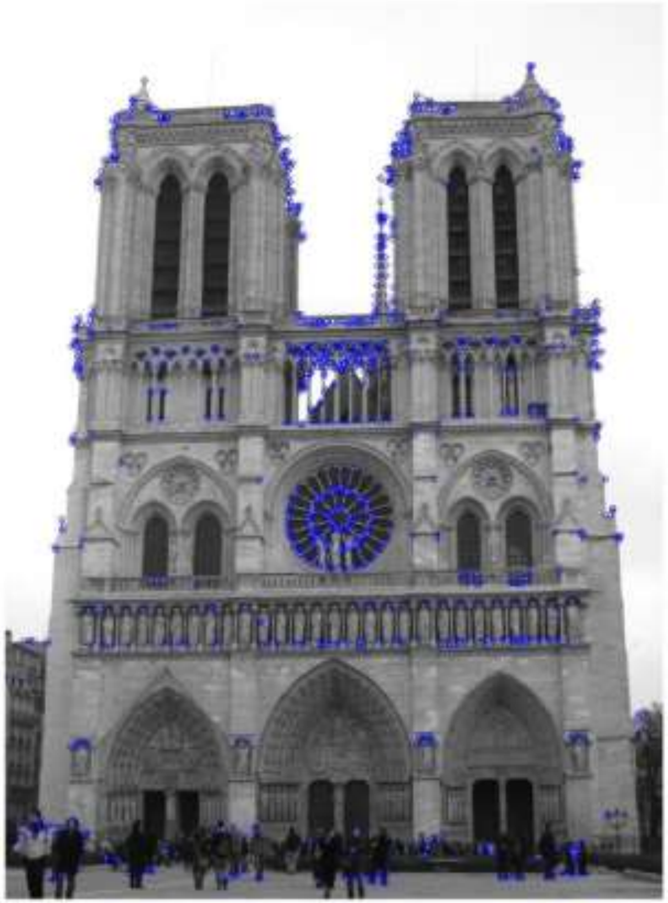
- To be a feature, it must be
  - A local maxima
  - Response (f) value is significantly greater than neighbors



Harris corner detection result

Adaptive Non-maximal Suppression (CVPR'05)

- Local Non-Maximum suppression
  - Slide a 3x3 window through the image and suppressing all the key points except for the key point with maximum confidence



Local Non-Maximum suppression

Adaptive Non-maximal Suppression (CVPR'05)

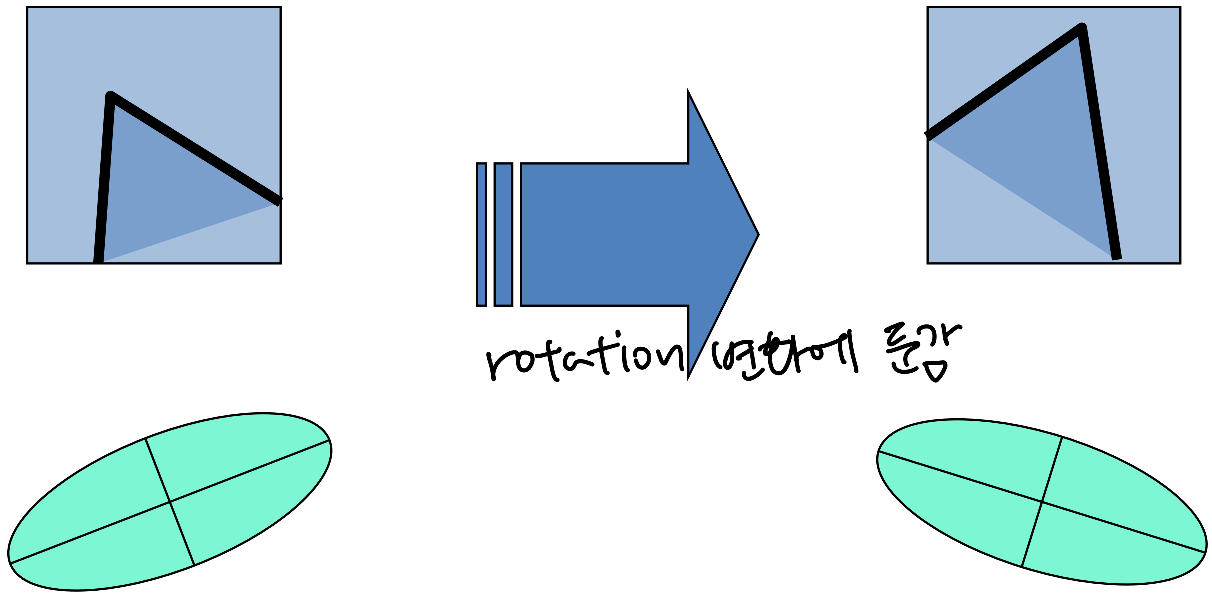
- Adaptive Non-Maximal Suppression
  - Apply Local non-maximum suppression
  - Keep track of the minimum distance to a larger magnitude interest point per point
  - Sort the list of interest points by descending radius and take the top N

Adaptive Non-Maximal Suppression

Harris Detector: Invariance Properties

- Translation and rotation invariance

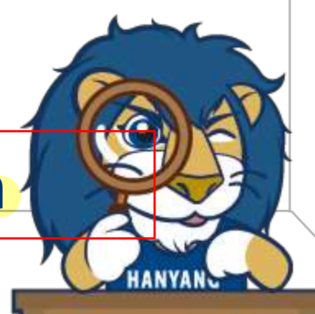
Harris Corner Detector는 rotation, translation 에 관계없이 코너 검출 잘됨



rotation 변화에 둔감

Ellipse rotates but its shape (i.e. eigenvalues) remains the same

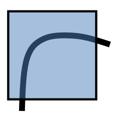
Corner response is invariant to image rotation



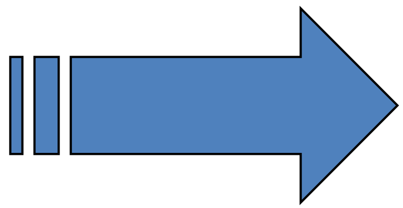
Harris Detector: Invariance Properties

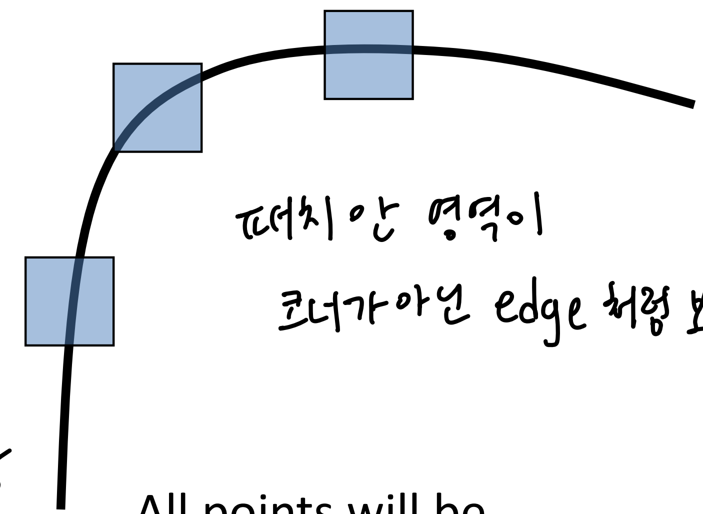
- Scaling

① corner 또는 ② 패치 크기 바꿈



Corner





패치 안 영역이

코너가 아닌 edge 처럼 보임

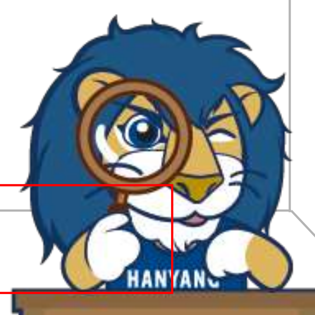
코너 커지면 패치사이즈도 커져야 함

All points will be classified as edges

$H = \begin{bmatrix} A & B \\ B & C \end{bmatrix}$ 계산

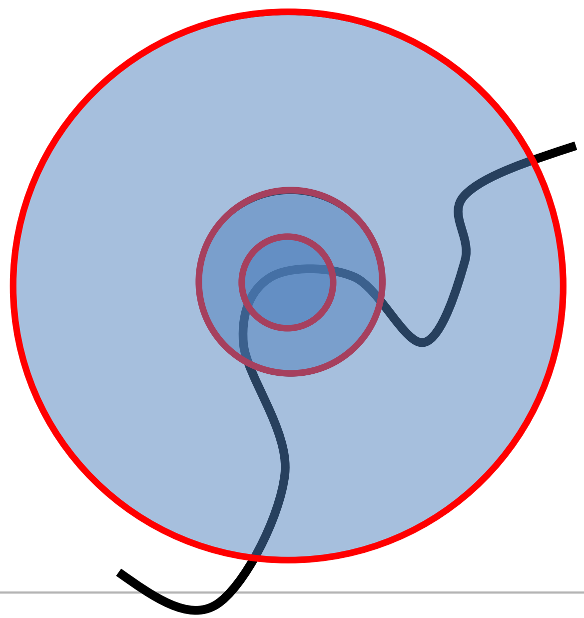
scale 변화에 민감

***Not invariant* to scaling**



Harris Detector: scale invariance

- Suppose you're looking for corners
  - Key idea: find scale that gives local maximum of f in both position and scale
  - One definition of f: the Harris operator



큰 코너는 큰 패치로 찾아야 함

Harris Detector: scale invariance

- Real example

original size

확대됨

$$f(I_{i_1 \ldots i_m}(x, \sigma)) = f(I_{i_1 \ldots i_m}(x', \sigma'))$$

Harris Detector: scale invariance

- Automatic scale detection: compare cornerness

$f(I_{i_1 \ldots i_m}(x,\sigma))$

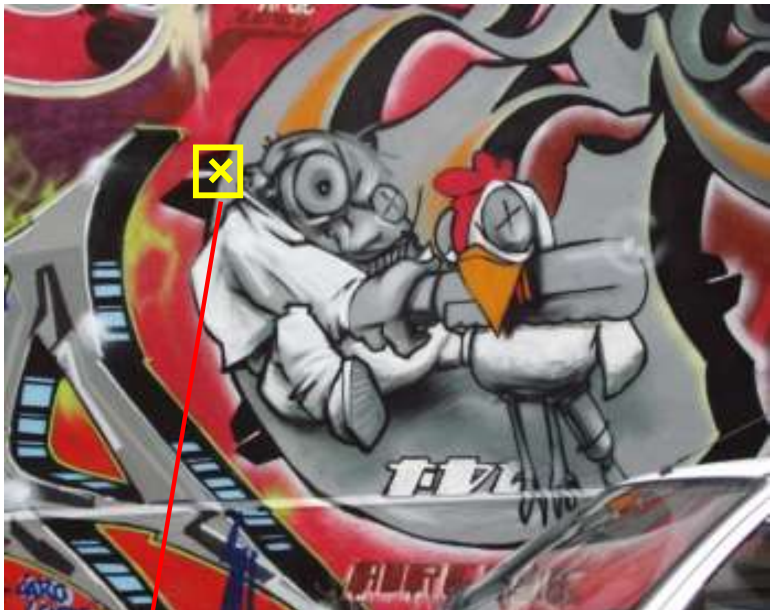
동일한 패치 사이즈를 사용해서 harrisc response를 보면

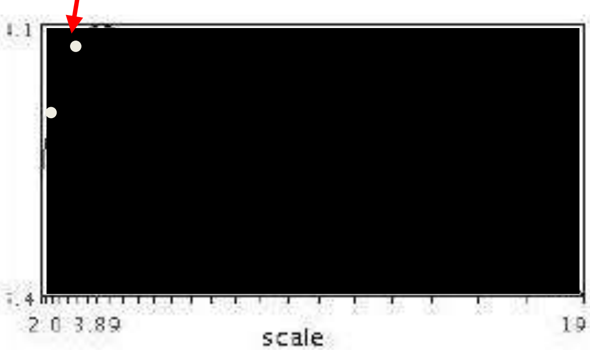
$f(I_{i_1 \ldots i_m}(x',\sigma))$

K. Grauman, B. Leibe

Harris Detector: scale invariance

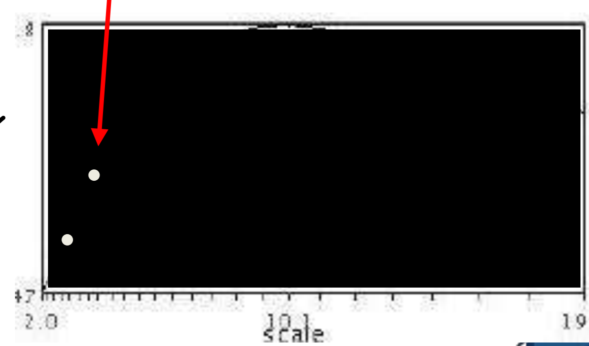
- Automatic scale detection: compare cornerness





$f(I_{i_1 \ldots i_m}(x,\sigma))$

큰 패치를 쓰면
harris response
더 좋아짐

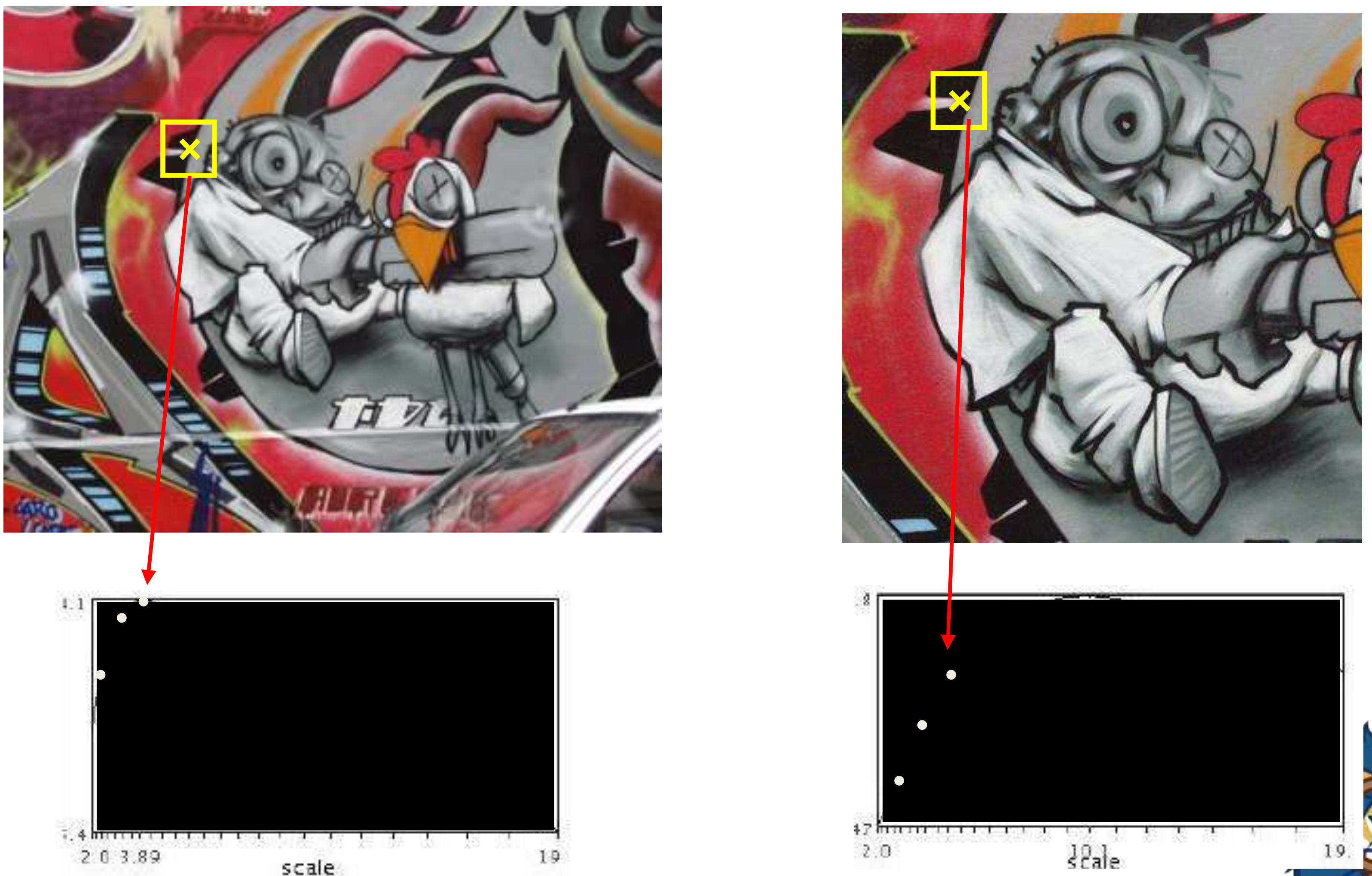


$f(I_{i_1 \ldots i_m}(x',\sigma))$

K. Grauman, B. Leibe

Harris Detector: scale invariance

- Automatic scale detection: compare cornerness



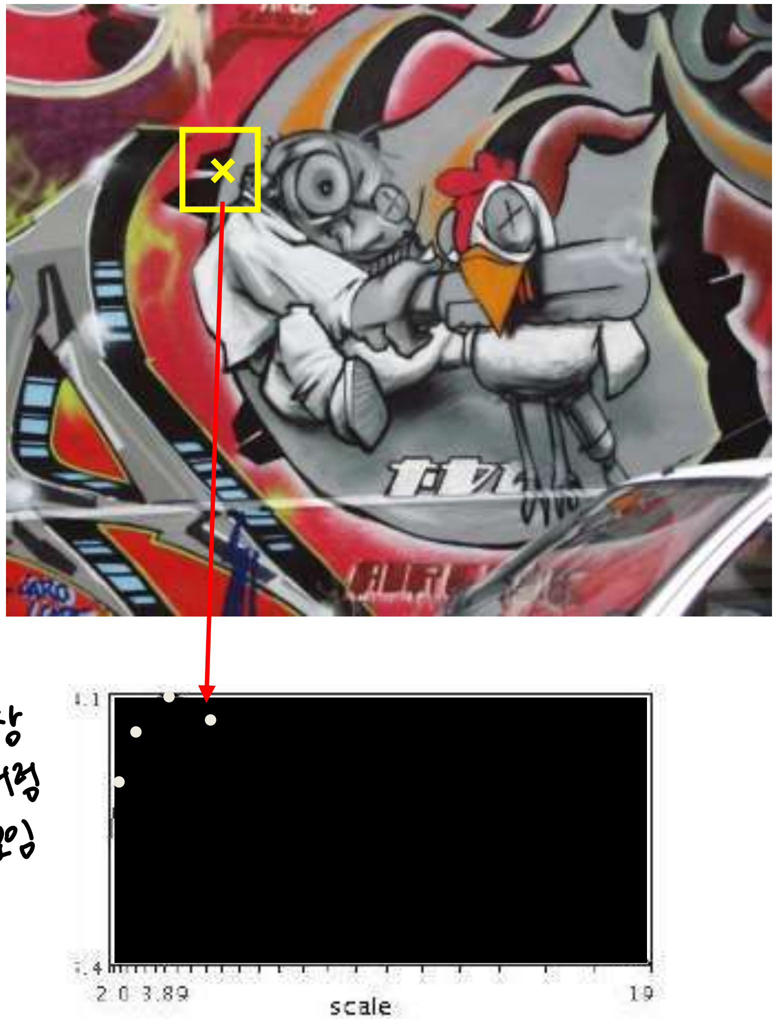
$f(I_{i_1 \ldots i_m}(x,\sigma))$ $f(I_{i_1 \ldots i_m}(x',\sigma))$

K. Grauman, B. Leibe

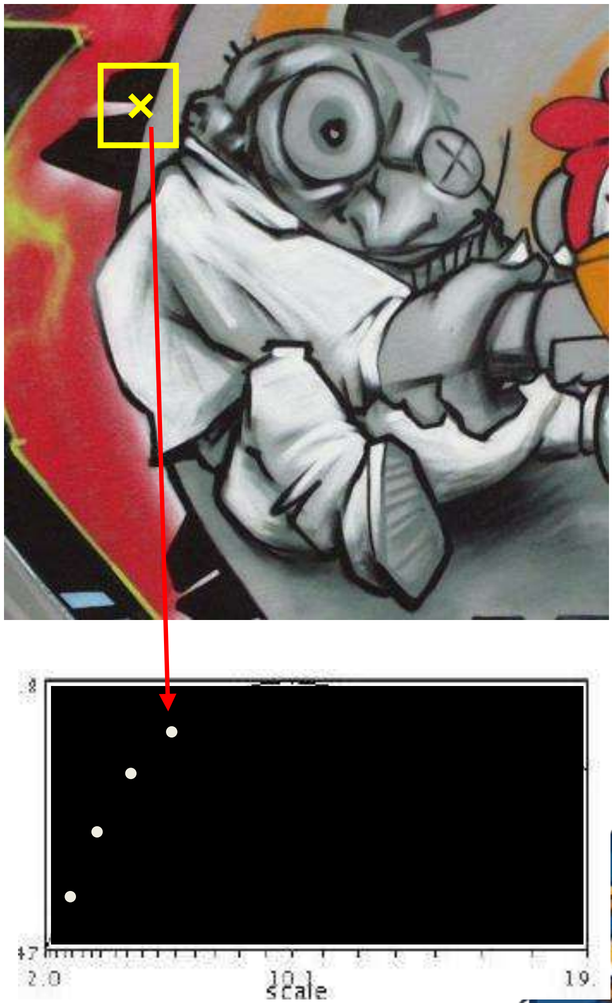
Harris Detector: scale invariance

- Automatic scale detection: compare cornerness

더이상
코너처럼
안보임



$f(I_{i_1 \dots i_m}(x, \sigma))$

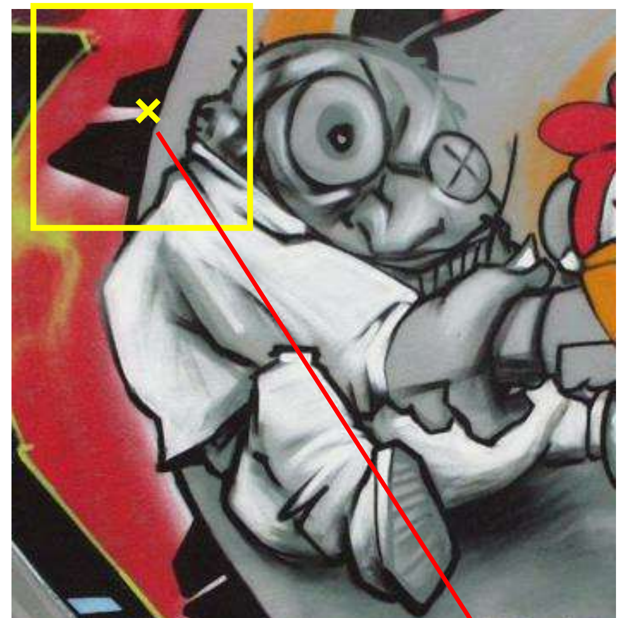


$f(I_{i_1 \dots i_m}(x', \sigma))$

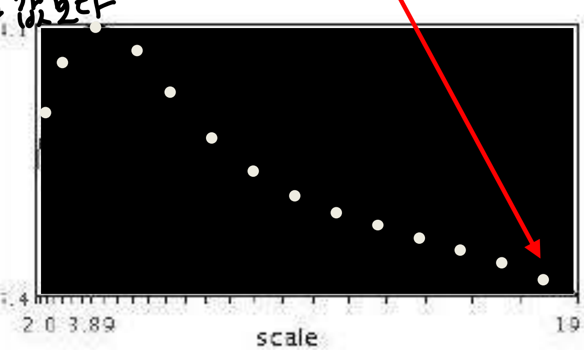
K. Grauman, B. Leibe

Harris Detector: scale invariance

- Automatic scale detection: compare cornerness

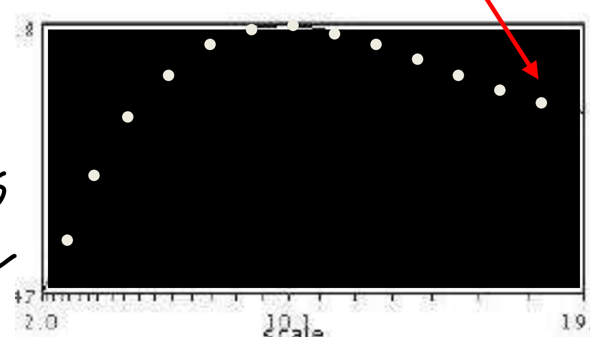


가장 높은 값을 기준으로
내가 정한 threshold 값보다
크면 코너라고 검출



$f(I_{i_1 \ldots i_m}(x, \sigma))$

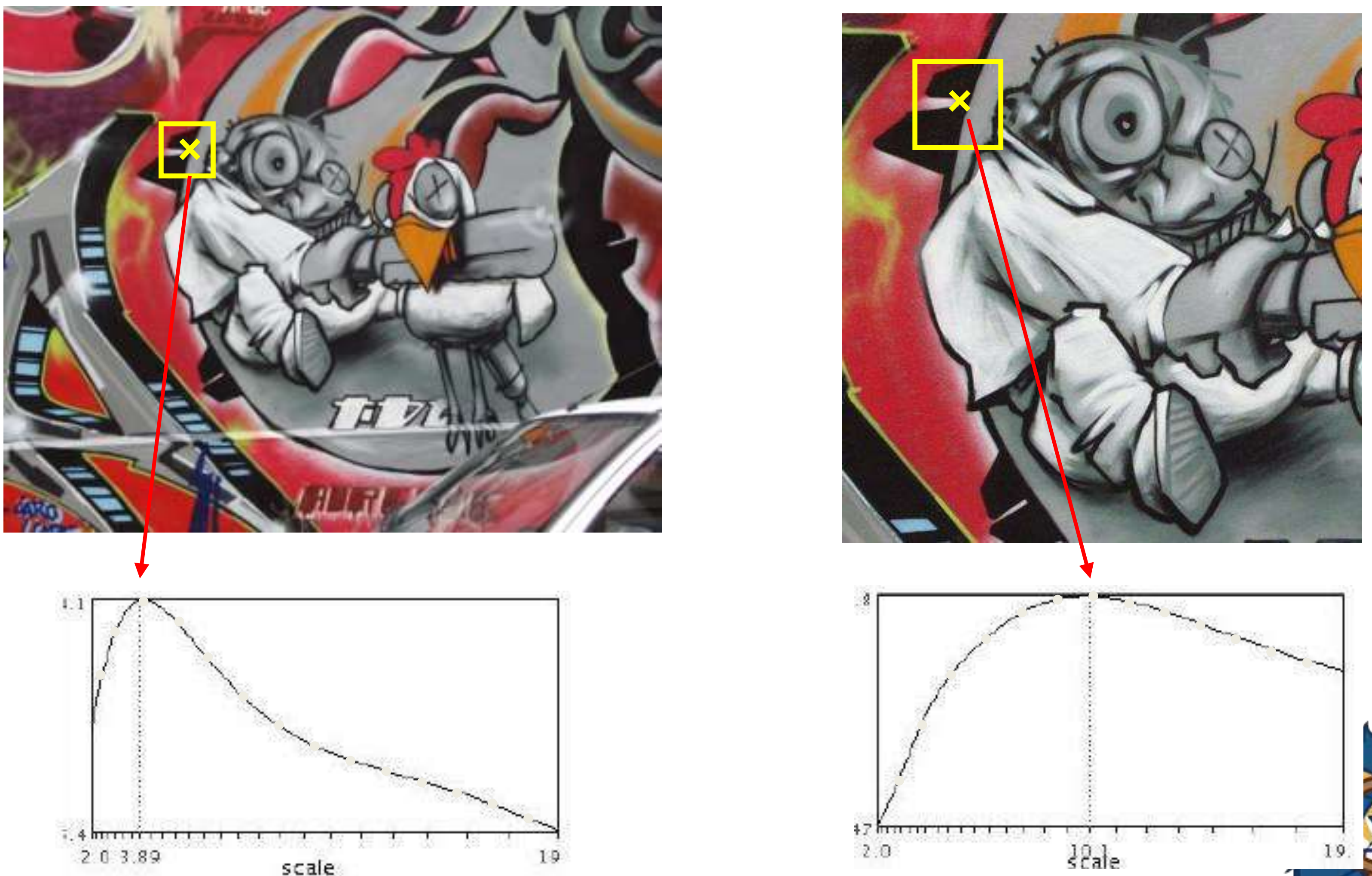
Cornorness가
가장 좋게 나오는
패치 사이즈가 있음
큰코너는 큰패치로
인식해야 함



$f(I_{i_1 \ldots i_m}(x', \sigma))$

Harris Detector: scale invariance

- Automatic scale detection: compare cornerness



$f(I_{i_1 \ldots i_m}(x, \sigma))$

$f(I_{i_1 \ldots i_m}(x', \sigma'))$

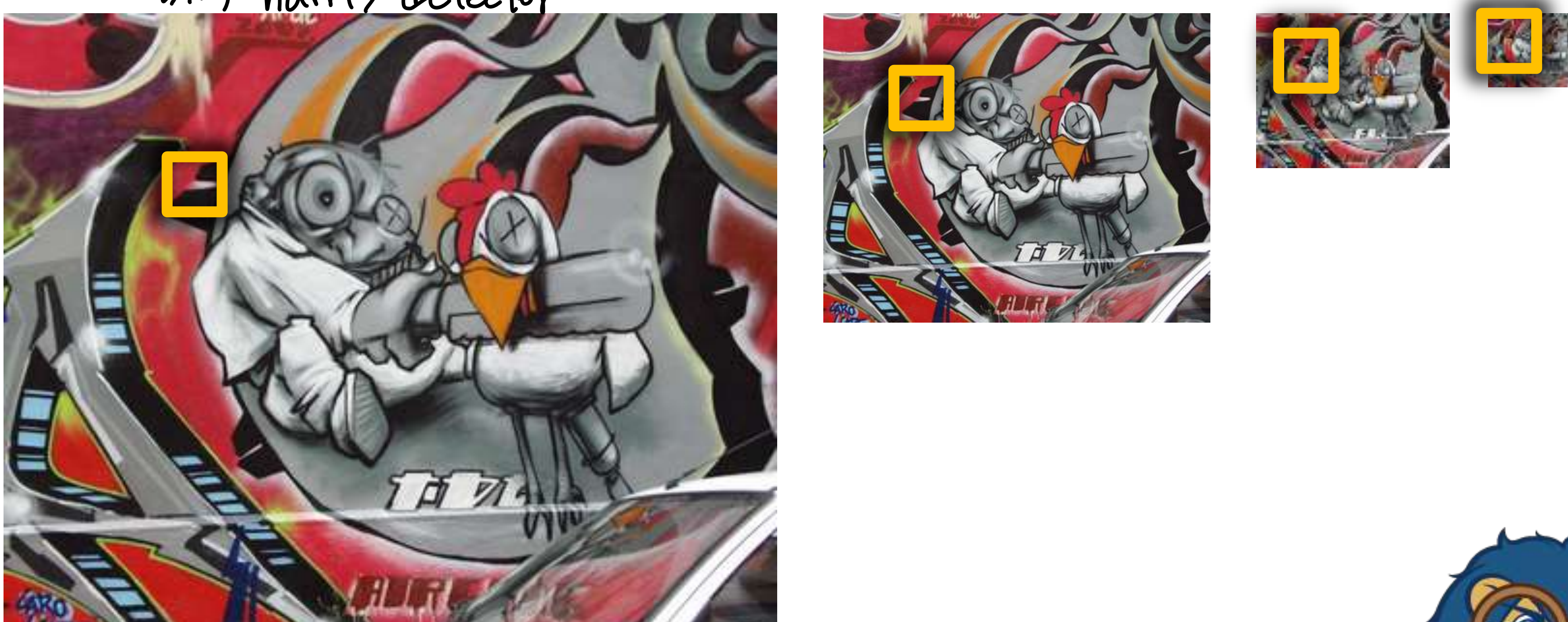
Harris Detector: scale invariance

- Practical implementation

그림 사이즈를 줄임 → 계산 측면 이점

  – Instead of computing *f* with larger windows, implement using a fixed window size with an image pyramid

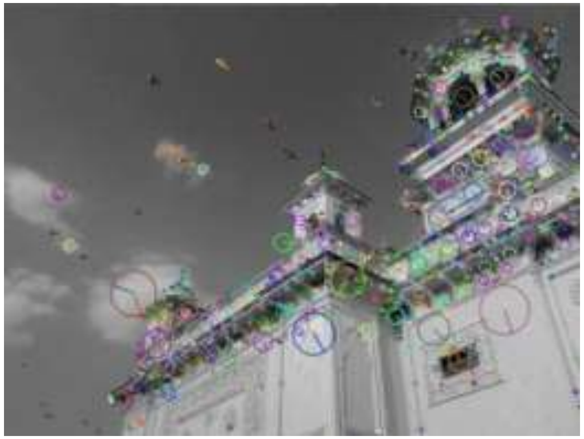
5x5 harris detector



Original image size

Other Detectors

- Scale-invariant feature transform (SIFT)
  - Lowe. ICCV'99
  - Find local maxima in position-scale space of Difference-of-Gaussian



SIFT result

- Maximally Stable Extremal Regions (MSER)
  - Matas et al. BMCV'02
  - Based on Watershed segmentation algorithm
  - Select regions that stay stable over a large parameter range

MSER result

Summary

- Why interest (feature) points?
  - Used in many applications for matching
- Harris corner detector
  - Traditional approach
  - Give motivation in many follow-up studies
- Invariant properties of Harris corner feature
  - Rotation invariant
  - Partially invariant to affine changes of brightness
  - Originally not invariant to scaling
    - But can be overcome by comparing cornerness in different scale

Matlab

- Download (licensed ver.)
  - https://iic.hanyang.ac.kr/office365

- Quick guide (official)
  - FYI
  - https://kr.mathworks.com/videos/introduction-to-matlab-81592.html

- HW1 (09/20~09/25)
  - Check out our course website
  - Linear image filtering

Thank you!