KENTECH
Korea Institute of Energy Technology

Advanced Computer Vision
Week 07

Oct. 14, 2022
Seokju Lee

Let's Learn SIFT Algorithm

Scale Invariant Feature Transform (SIFT)

and its applications for image alignment and 2D object recognition.

Topics:

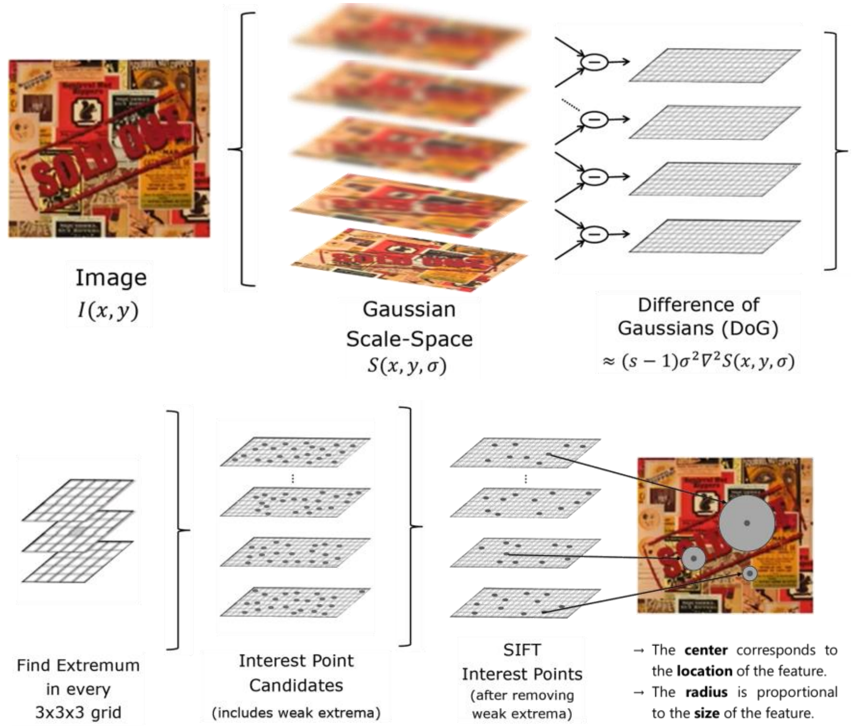
(1) What is an Interest Point?

(2) Detecting Blobs

(3) SIFT Detector

(4) SIFT Descriptor

Detected SIFT features with multiscale



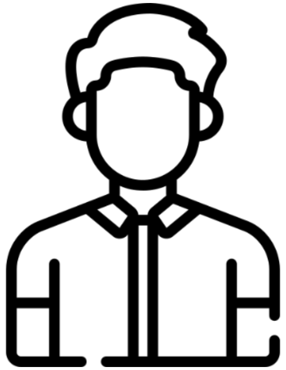
Parts of slides are by Prof. In So Kweon and Prof. Shree Nayar

KENTECH
Korea Institute of Energy Technology

SIFT Descriptor

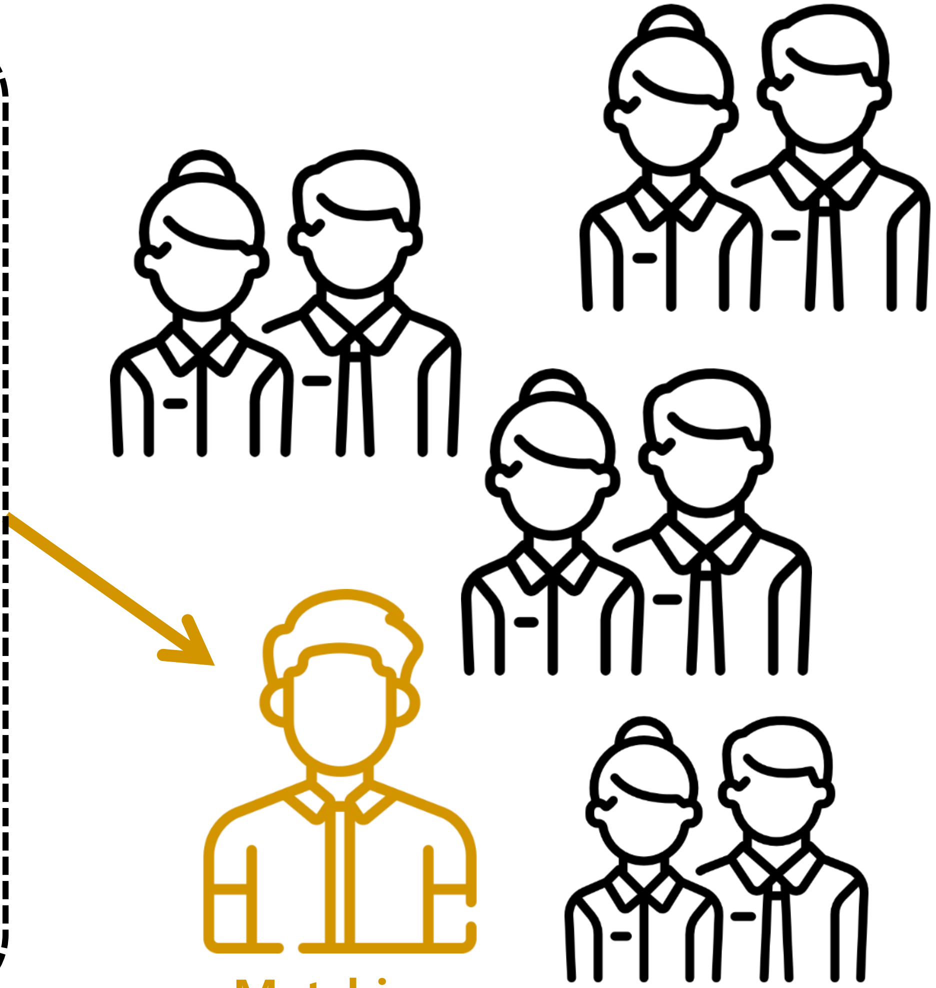
What is Descriptor?

Descriptor = Set of **Distinctive** features and their **identifiers**



Query

Identifiers



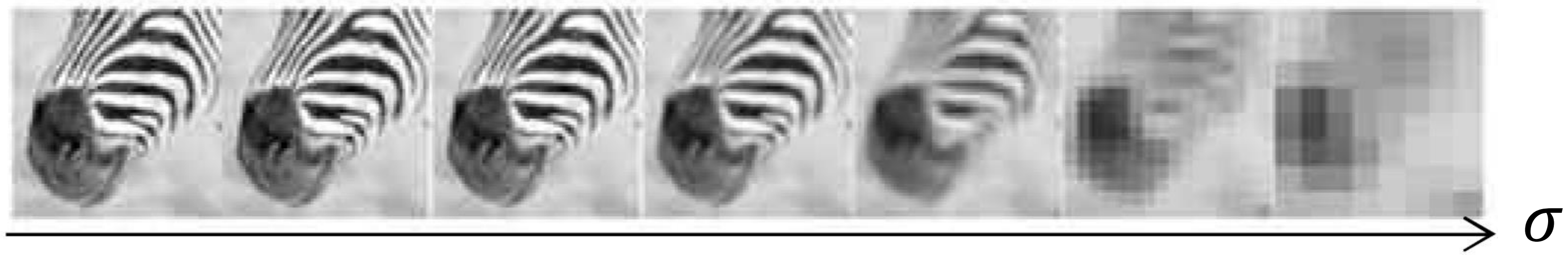
Matching

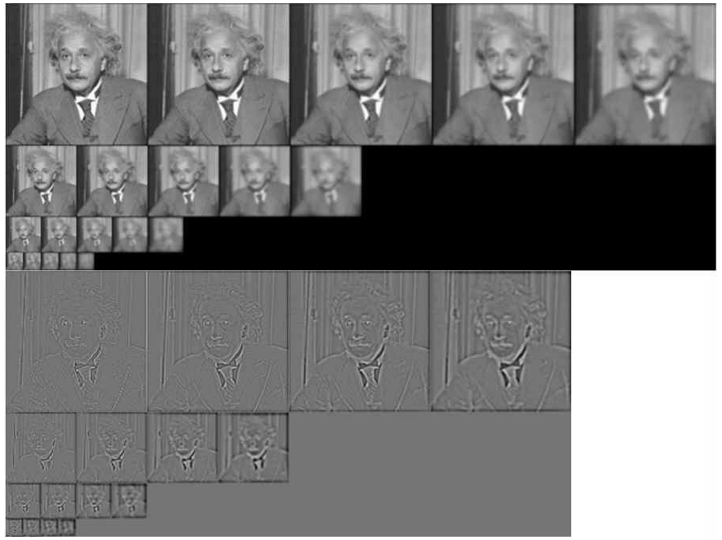
→ Must be **robust** to variances + **generalized/standardized**

SIFT: Scale Invariance

Scale space

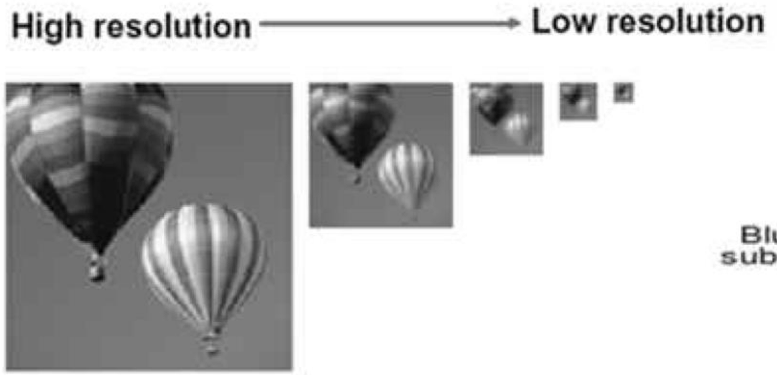
Different scales are necessary for describing different objects in the image, and we may not know the correct scale/size ahead of time (generalization).

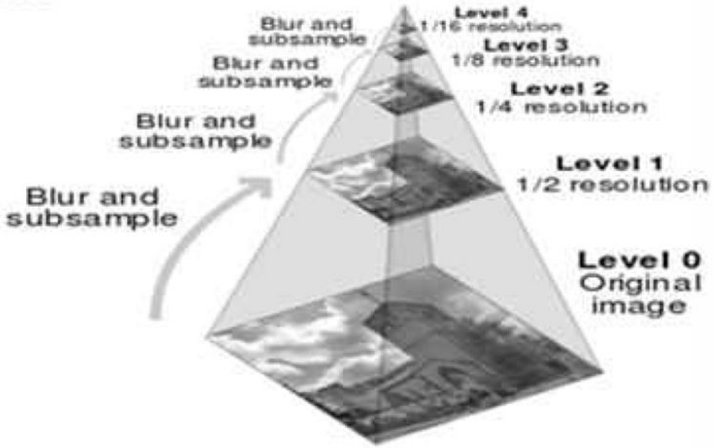




Gaussian Pyramid

Image pyramid technique is used for reducing the memory usage and computational complexity.



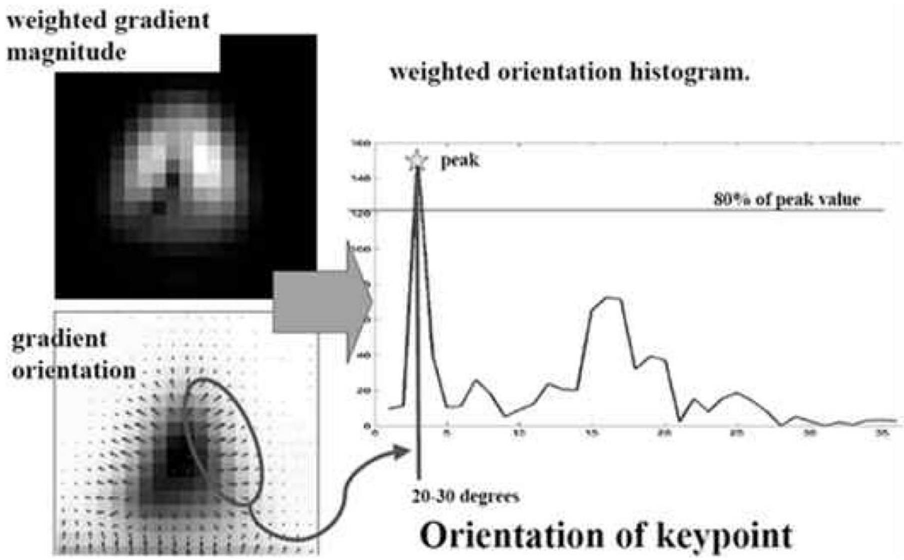


SIFT: Rotation Invariance

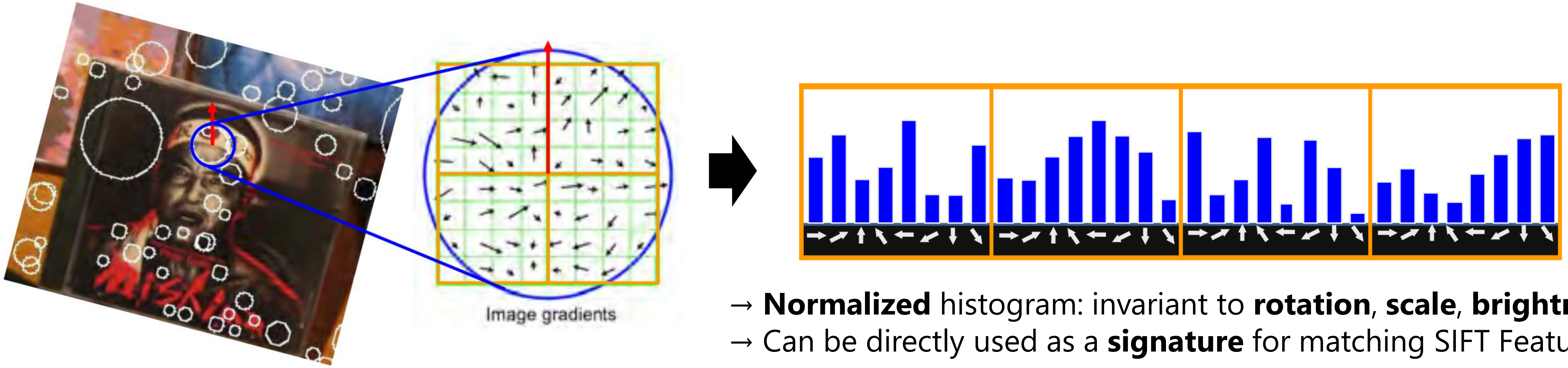
Compute gradient for each blurred image

$$m(x,y)=\sqrt{(L(x+1,y)-L(x-1,y))^2+(L(x,y+1)-L(x,y-1))^2}$$

$$\theta(x,y)=\tan^{-1}((L(x,y+1)-L(x,y-1))/(L(x+1,y)-L(x-1,y)))$$



Histograms of gradient directions over spatial regions



→ **Normalized** histogram: invariant to **rotation**, **scale**, **brightness**
→ Can be directly used as a **signature** for matching SIFT Features!

Comparing SIFT Descriptors

Essentially comparing two arrays of data

Let $H_1(k)$ and $H_2(k)$ be two arrays of data of length N,

L2 Distance :

$$d(H_1, H_2) = \sqrt{\sum_k \left(H_1(k) - H_2(k)\right)^2}$$

Smaller the distance metric, **better** the match.

Perfect match when $d(H_1, H_2) = 0$

Comparing SIFT Descriptors

Essentially comparing two arrays of data

Let $H_1(k)$ and $H_2(k)$ be two arrays of data of length N,

Normalized Correlation :

$$d(H_1, H_2) = \frac{\sum_k [(H_1(k) - \bar{H}_1)(H_2(k) - \bar{H}_2)]}{\sqrt{\sum_k (H_1(k) - \bar{H}_1)^2} \sqrt{\sum_k (H_2(k) - \bar{H}_2)^2}}$$

where: $\bar{H}_i = \frac{1}{N} \sum_{k=1}^{N} H_i(k)$

Larger the distance metric, **better** the match.

Perfect match when $d(H_1, H_2) = 1$

*Correlation?

Source image

Template

How do we **locate** the template in the image?

Minimize :

$$E[i,j] = \sum_{m}\sum_{n}(f[i+m,j+n] - t[m,n])^2$$

$$E[i,j] = \sum_{m}\sum_{n}(f^2[i+m,j+n] + t^2[m,n] - \underline{\boldsymbol{2f[i+m,j+n]t[m,n]}})$$

→ **Maximize**

*Correlation?

Source image

Template

How do we **locate** the template in the image?

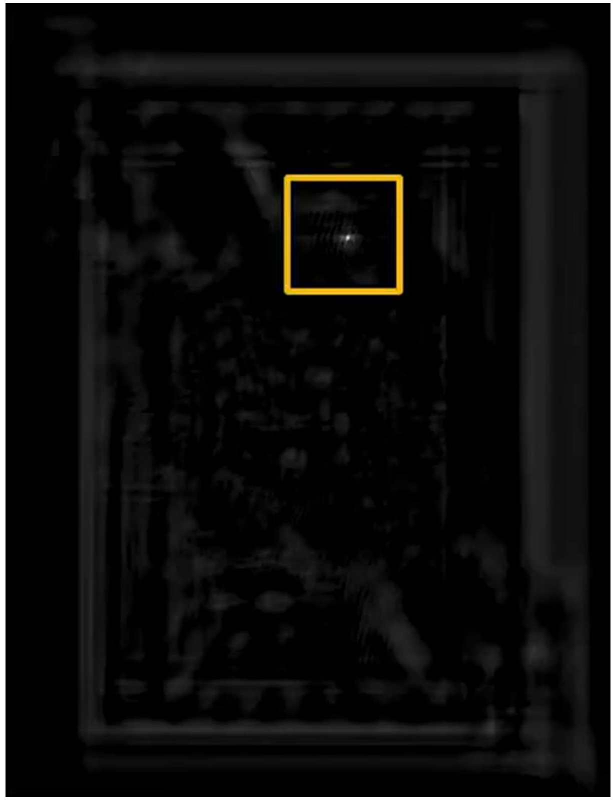
Maximize :

$$R[i,j] = \sum_{m} \sum_{n} t[m,n] f[i+m, j+n] = t \otimes f$$

→ Cross-correlation

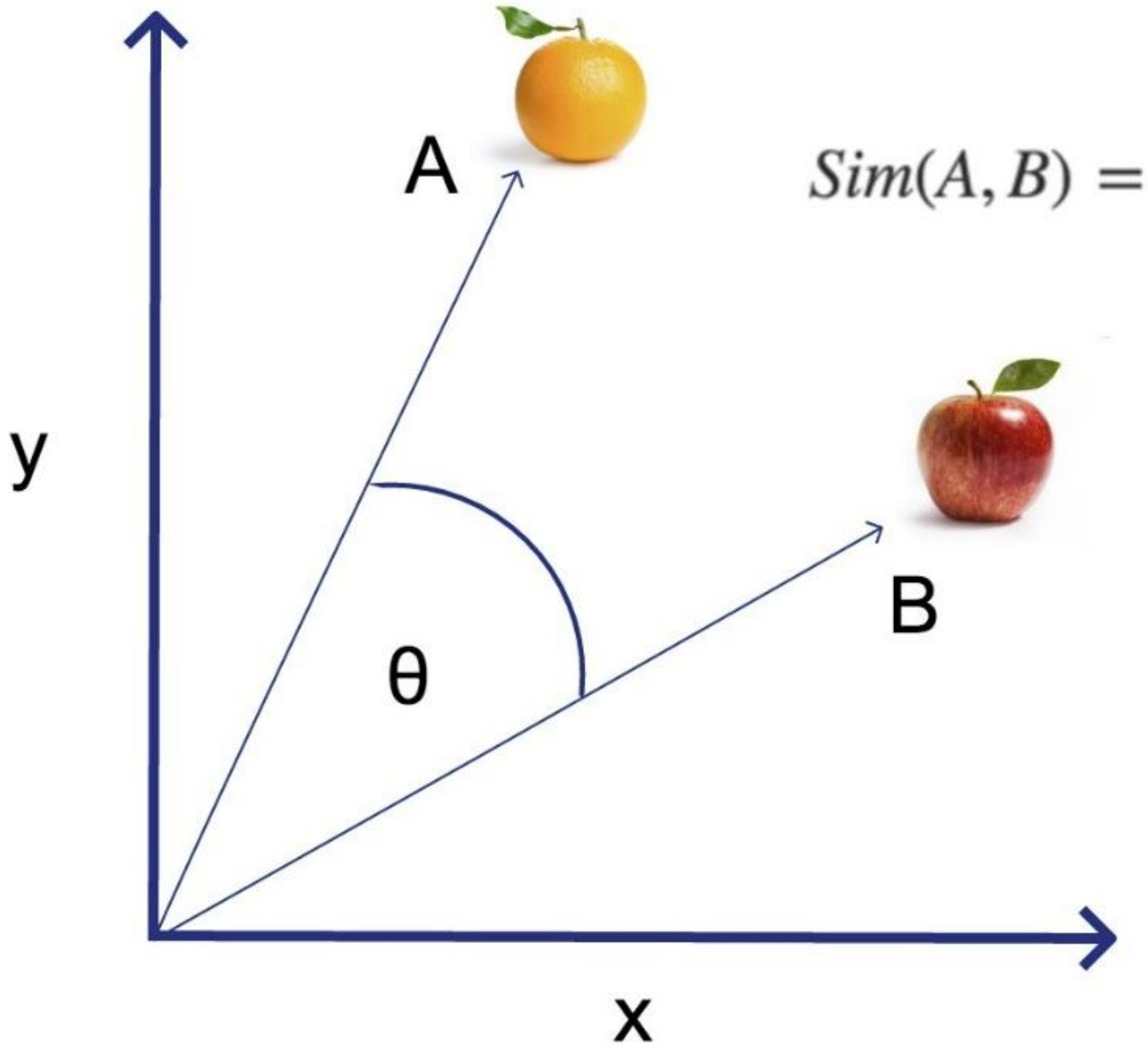
c.f.) convolution

$$\text{cosine similarity} = S_C(A,B) := \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}},$$



→ Result of cross-correlation

*Cosine Similarity



$$Sim(A,B) = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|}$$

Comparing SIFT Descriptors

Essentially comparing two arrays of data

Let $H_1(k)$ and $H_2(k)$ be two arrays of data of length N,

Normalized Correlation :

$$d(H_1, H_2) = \frac{\sum_k [(H_1(k) - \bar{H}_1)(H_2(k) - \bar{H}_2)]}{\sqrt{\sum_k (H_1(k) - \bar{H}_1)^2} \sqrt{\sum_k (H_2(k) - \bar{H}_2)^2}}$$

where: $\bar{H}_i = \frac{1}{N} \sum_{k=1}^{N} H_i(k)$

Larger the distance metric, **better** the match.

Perfect match when $d(H_1, H_2) = 1$

Comparing SIFT Descriptors

Essentially comparing two arrays of data

Let $H_1(k)$ and $H_2(k)$ be two arrays of data of length N,

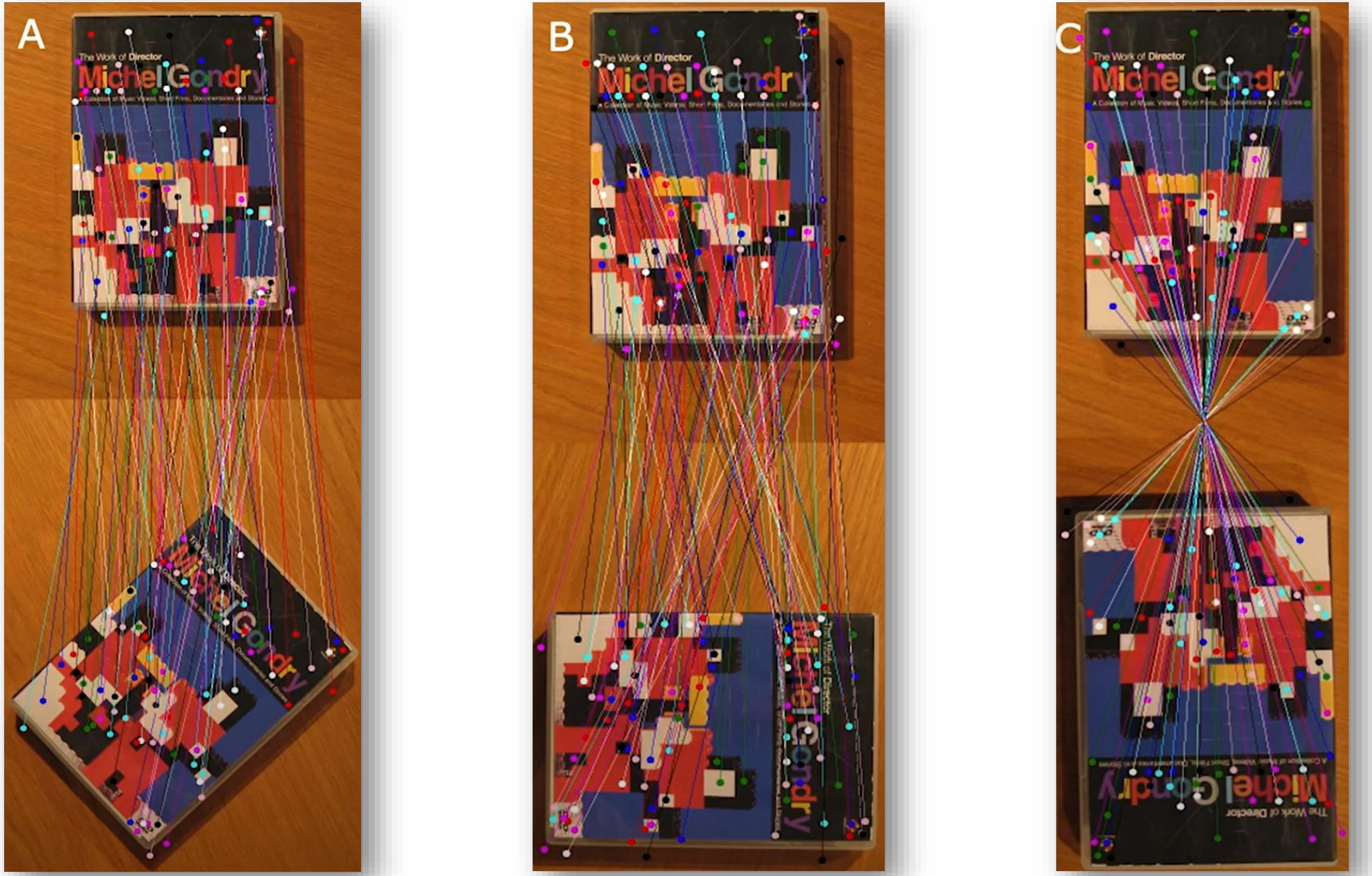
Intersection :

$$d(H_1, H_2) = \sum_k \min\left(H_1(k), H_2(k)\right)$$

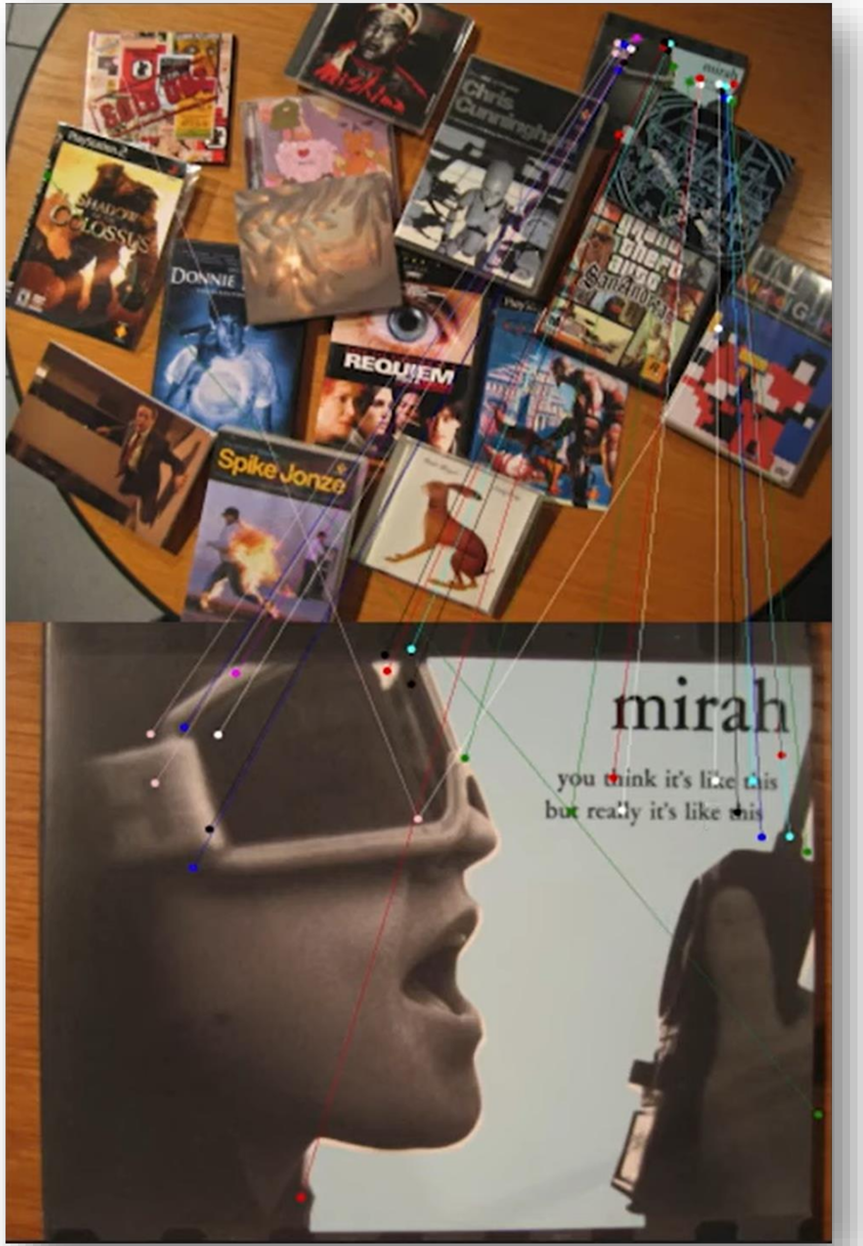
Larger the distance metric, **better** the match.

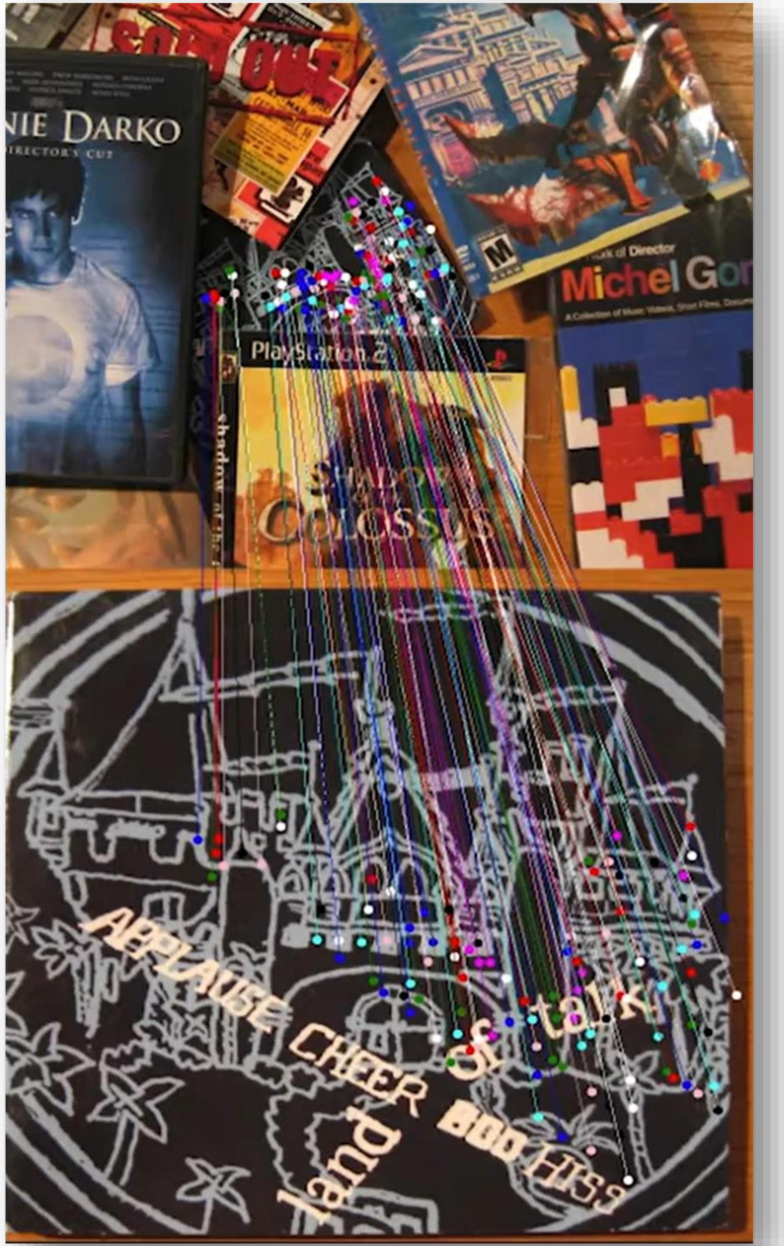
SIFT Results: Scale Invariance

SIFT Results: Rotation Invariance



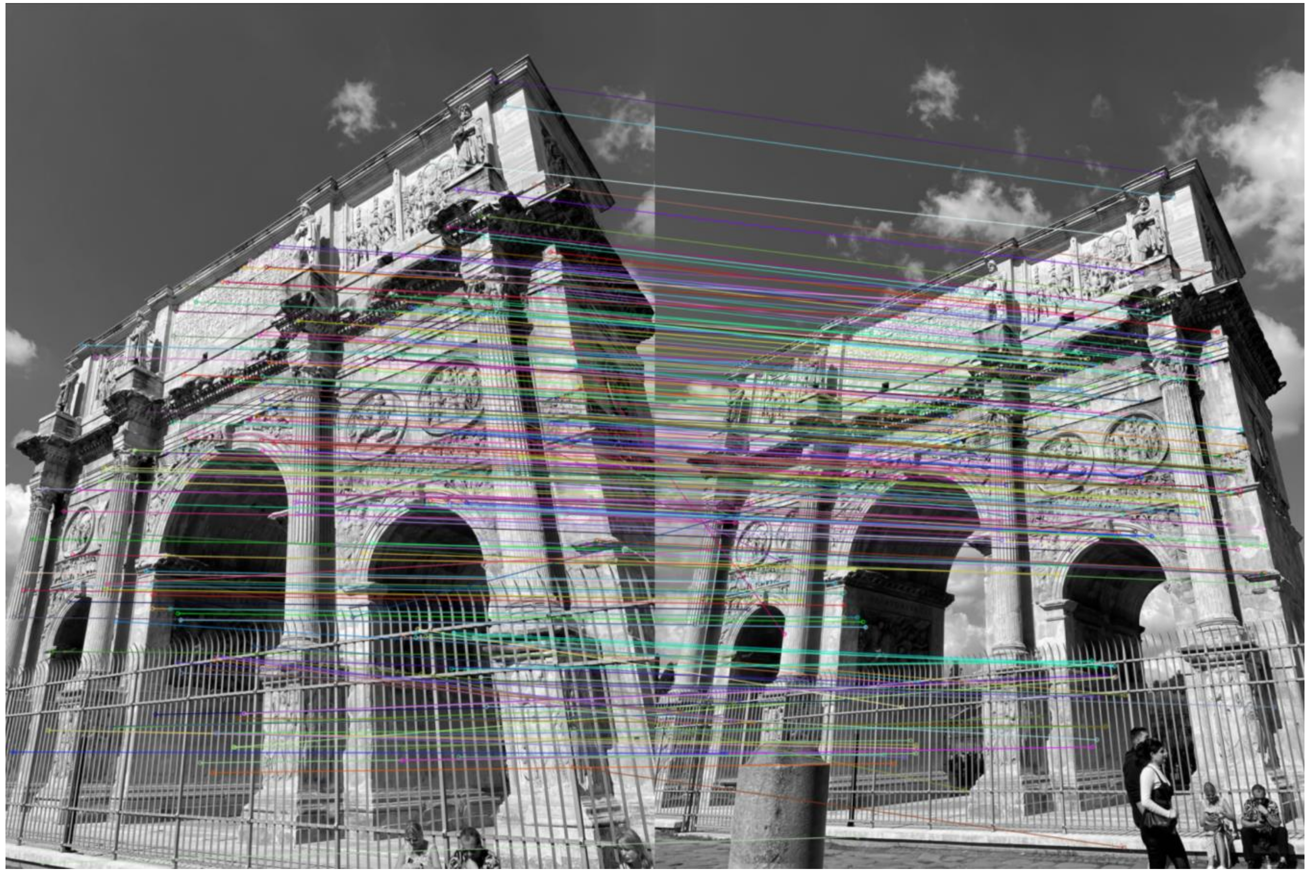
SIFT Results: Robust to Clutter



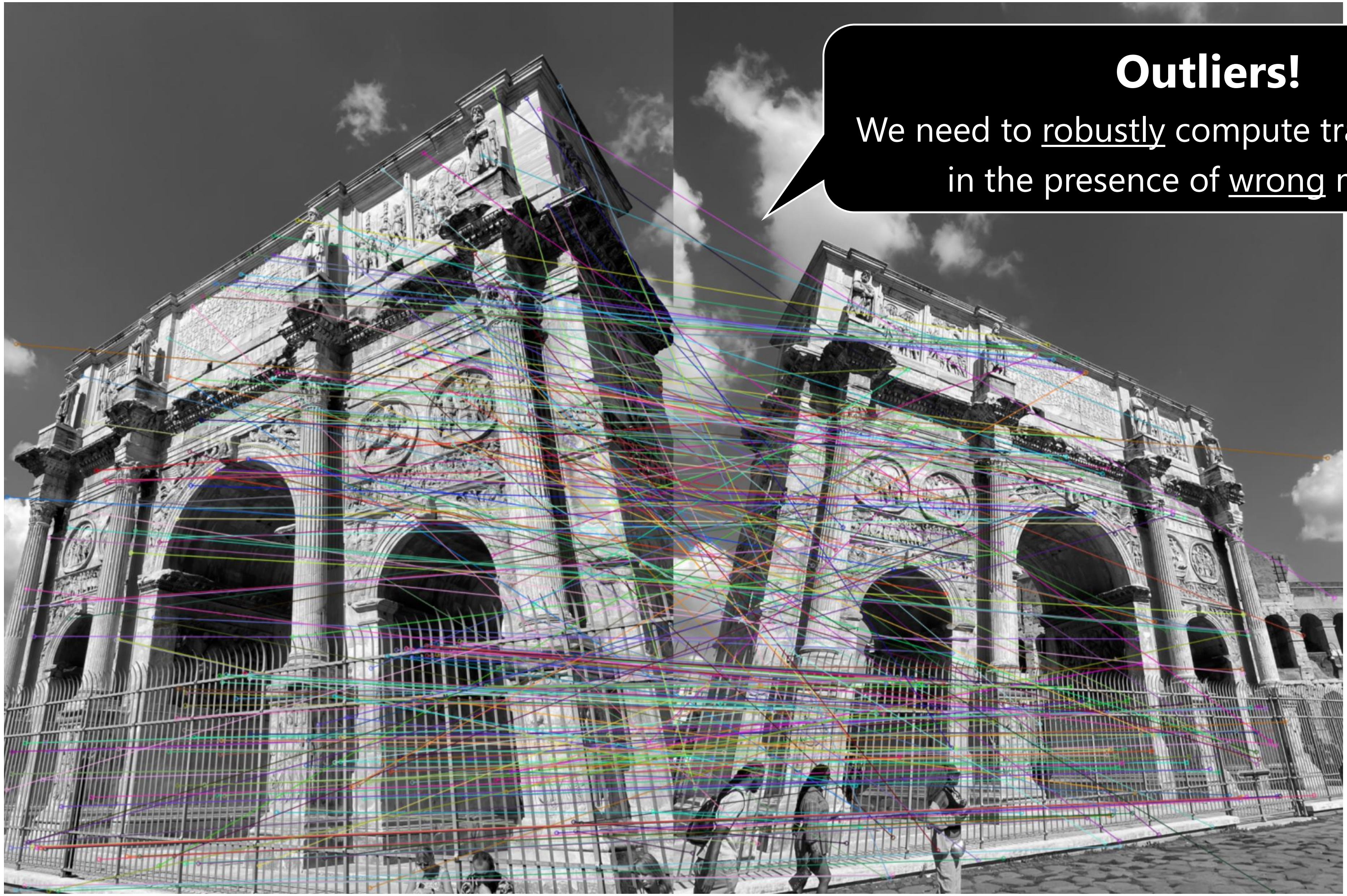


Experiment 1: SIFT Matching with Different Motions and Image Transformations

https://view.kentech.ac.kr/f088fa7f-874e-44bc-bd6d-6084b42dfdf7



How to Handle Outliers?



Outliers!

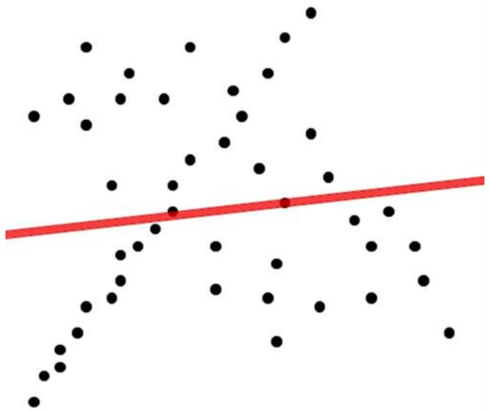
We need to robustly compute transformations in the presence of wrong matches.

RANdom SAmple Consensus

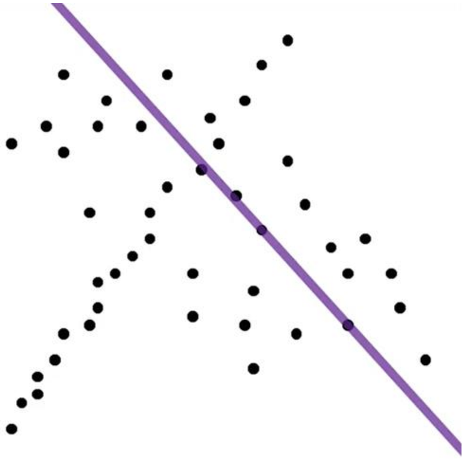
General RANSAC Algorithm :

1. Randomly choose s samples. Typically s is the minimum samples to fit a model.
2. Fit the model to the randomly chosen samples.
3. Count the number M of data points (inliers) that fit the model within a measure of error ε.
4. Repeat Steps 1-3 N times.
5. Choose the model that has the largest number M of inliers.

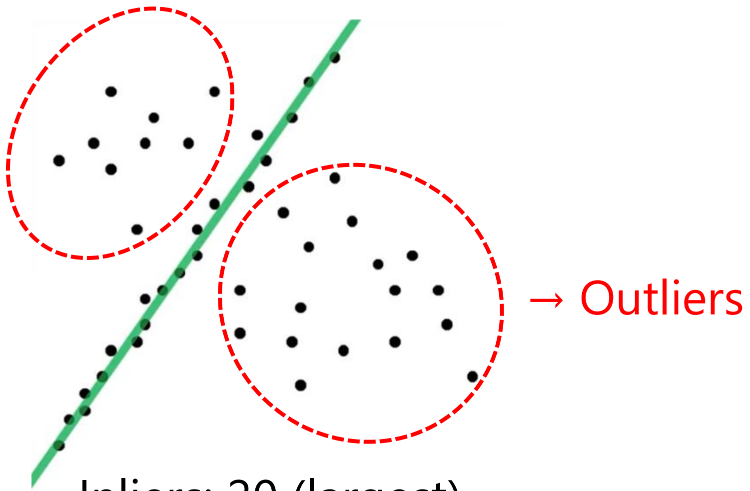
ex) Line fitting :



Inliers: 2



Inliers: 4

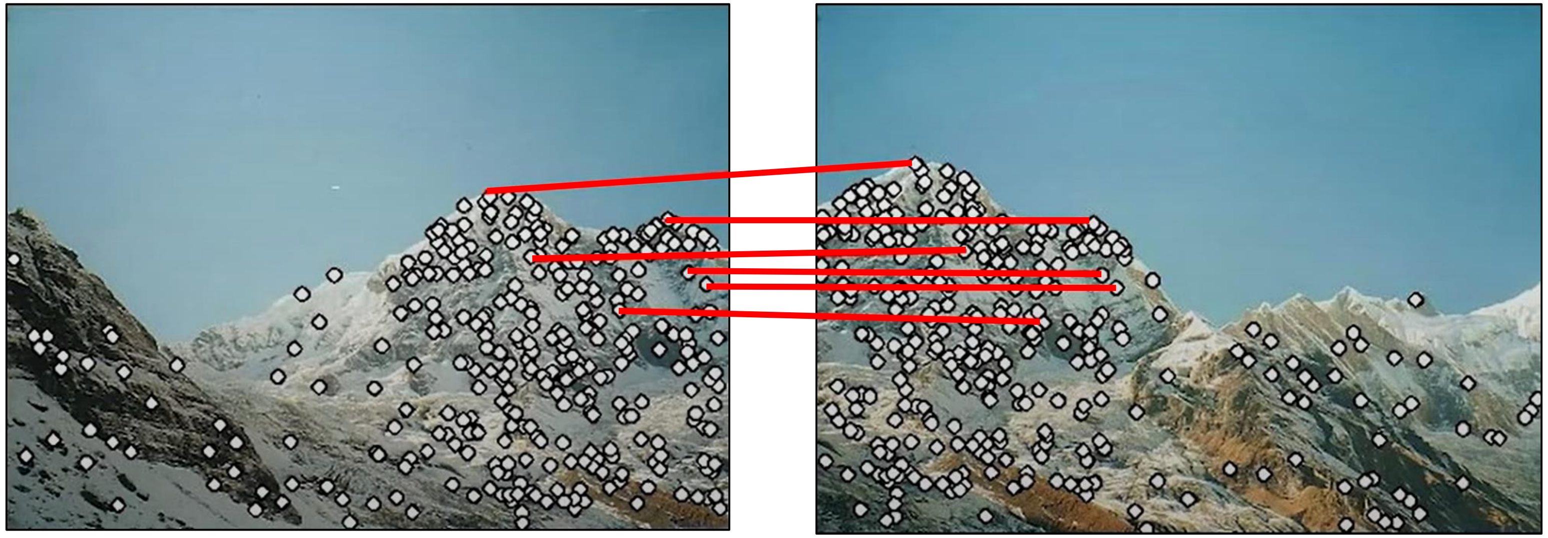


Inliers: 20 (largest)

Experiment 2: Panorama Stitching using SIFT

Image 1

Image 2



Match SIFT interest points.

Experiment 2: Panorama Stitching using SIFT

Warp and combine images to create a larger image.