Chapitre 1. SIFT Descriptor and Applications

1.1 Introduction
1.2 What is an interest point?
1.3 Detecting blobs
1.4 SIFT Detector
1.5 SIFT Descriptor
1.6 Image Stitching
- Overview
- Image Transformations
- Computing Homography
- Dealing with Outliers: RANSAC
- Warping and Blending Images

Chapitre 1. SIFT Descriptor and Applications

1.1 Introduction

We know how to find edges and corners in images,
And then compute the boundary of objects.

For object recognition with fairly complex appearances, we must be able de match features that are fairly descriptive and unique.

SIFT feature detector is able to response to this query.

Chapitre 1. SIFT Descriptor and Applications

1.1 Introduction

How to recognize the objects of figure 1?
We can use some techniques such as apply threshold and get binary image and compute geometry properties of these objects to recognize them.

Chapitre 1. SIFT Descriptor and Applications

1.1 Introduction

How to find the objects in the left of figure 2 in the right of the same figure?
Solution: We can use template matching.

Problems: When image is rotated and magnified differently.
Solution: We need then different templates of the object under different rotations and scales. In addition, objects are occluded in the query image.

Chapitre 1. SIFT Descriptor and Applications

1.1 Introduction

Other solution:

Extract directly from image some descriptive unique features.

We will use SIFT feature detector (Scale Invariant Feature Transform) proposed by D. Lowe in 2004.

Chapitre 1. SIFT Descriptor and Applications

1.1 Introduction

We study:
- The theory behind this descriptor

- How to use it to solve some vision problems:
  Image alignment, Images stitching, 2D Object recognition.

Chapitre 1. SIFT Descriptor and Applications

1.1 Introduction

-1.2 What is an interest point?
From the state of the art, edges and corners are not interesting for a lot of applications.

-The **blob**, is an interesting point, which has some local appearance within it useful for matching.

Chapitre 1. SIFT Descriptor and Applications

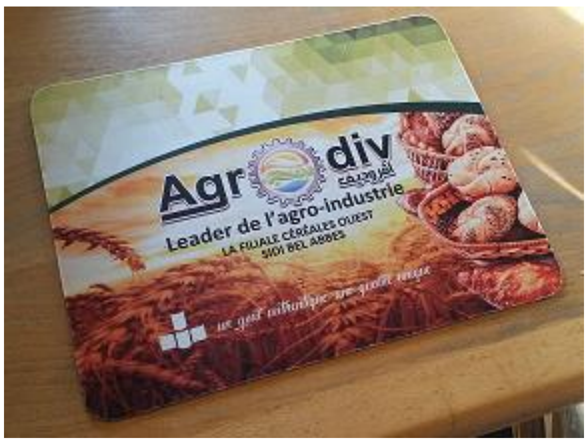
1.1 Introduction

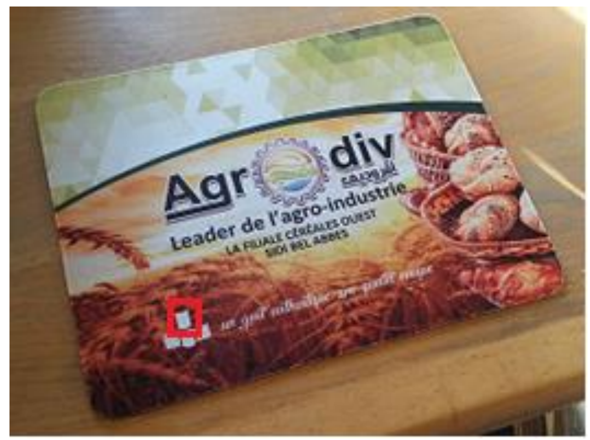
We study:
- What is an interest point?
- Detecting a blob: it is a patch with local appearance
- SIFT Detector (David Low)
- SIFT Descriptor (extract a description or signature)

Chapitre 1. SIFT Descriptor and Applications

1.2 What is an interest point?

- Two images with difference in lighting(distribution), brightness(intensity), orientation, size, etc.



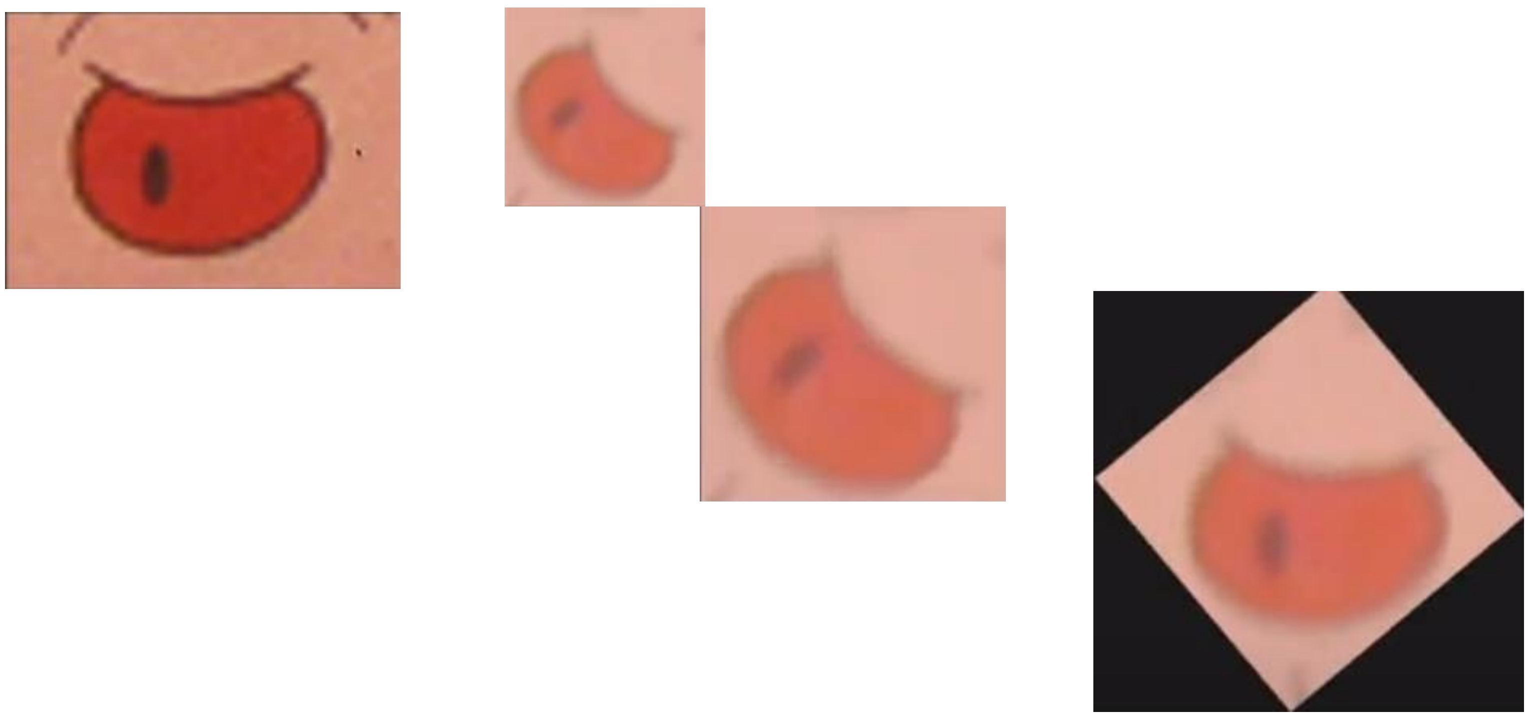


Chapitre 1. SIFT Descriptor and Applications

1.2 What is an interest point?

- Removing sources of variation:

To match the two patches, we want to be able to rescale, rotate (eliminate the differences)



Chapitre 1. SIFT Descriptor and Applications

1.2 What is an interest point?

- Has rich image content (brightness variations, color variation, etc) within the local window.

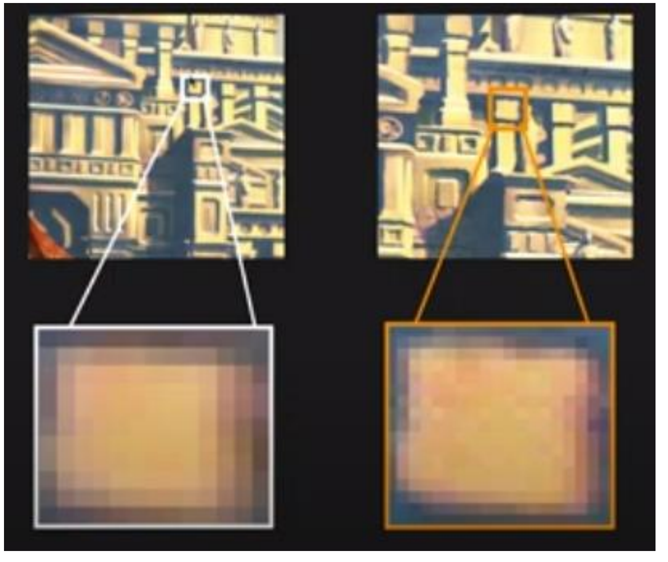
An interesting point is to have rich image content around. The local appearance around this point should be rich enough in terms of brightness variations, color variation, etc., that there is a certain amount of uniqueness to it that can be exploited for matching purposes.

- Has well-defined representation (signature) for matching/comparing with other points.
- Has a well-defined position in the image.
- Should be invariant to image rotation and scaling.
- Should be insensitive to lighting changes.

Chapitre 1. SIFT Descriptor and Applications

1.2 What is an interest point?

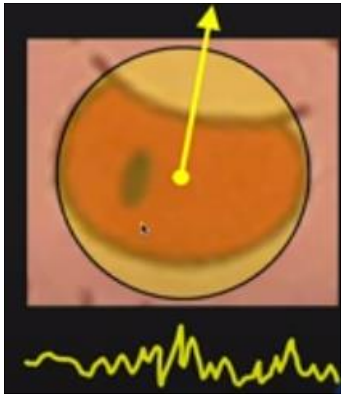
- Boundaries are not descriptive enough
- Blobs are descriptive enough, potentially good interest points.



Chapitre 1. SIFT Descriptor and Applications

1.2 What is an interest point?

For a blob-like feature to be useful, we need to:
- Locate the blob
- Determine its size
- Determine its orientation
- Formulate a description or signature that is independent of size and orientation.



Chapitre 1. SIFT Descriptor and Applications

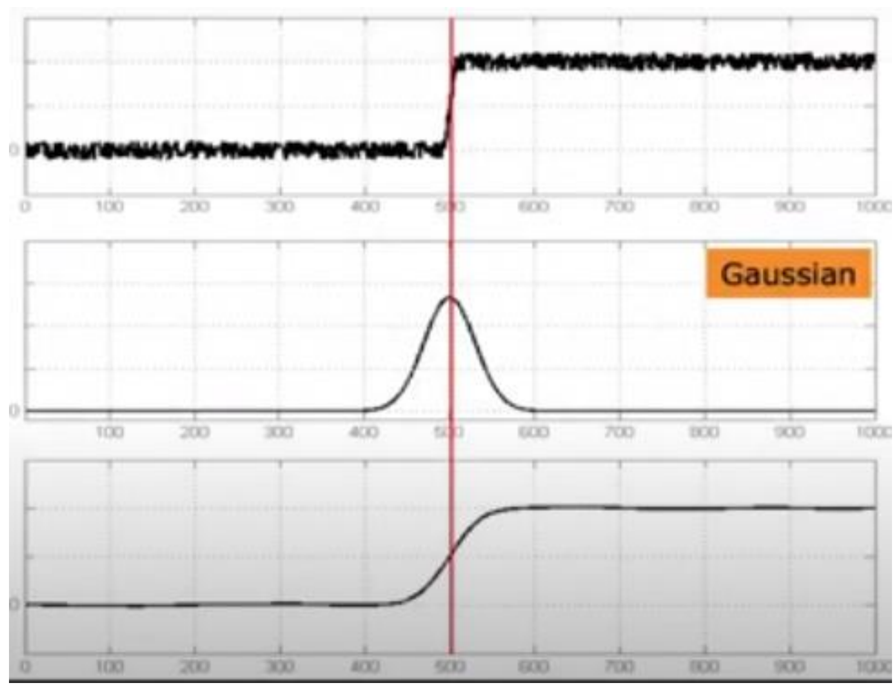
1.3 Detecting blobs

Gaussian filter is used for removing noise by smoothing

f

n_σ

$n_\sigma * f$



Chapitre 1. SIFT Descriptor and Applications

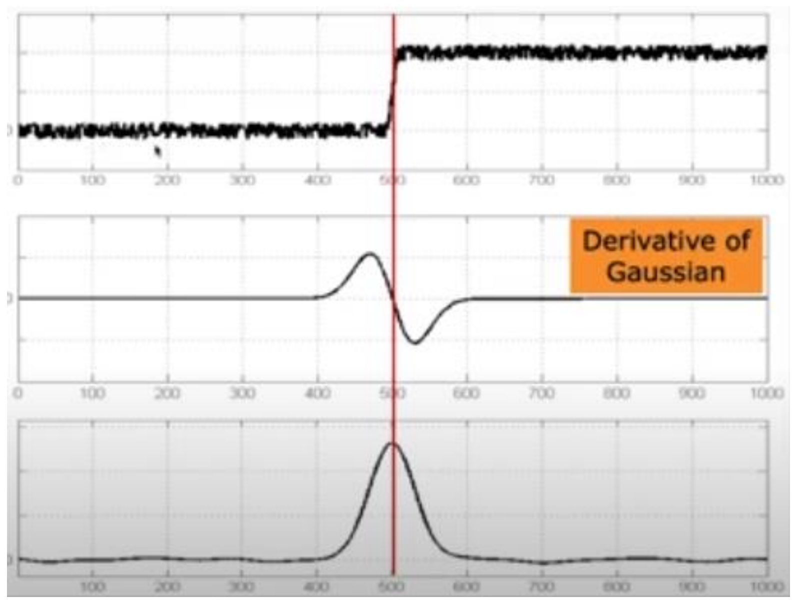
1.3 Detecting blobs

Extremum of Derivative of Gaussian filter denotes an edge.

f

$\nabla(n_\sigma)$

$\nabla(n_\sigma) * f$



Chapitre 1. SIFT Descriptor and Applications

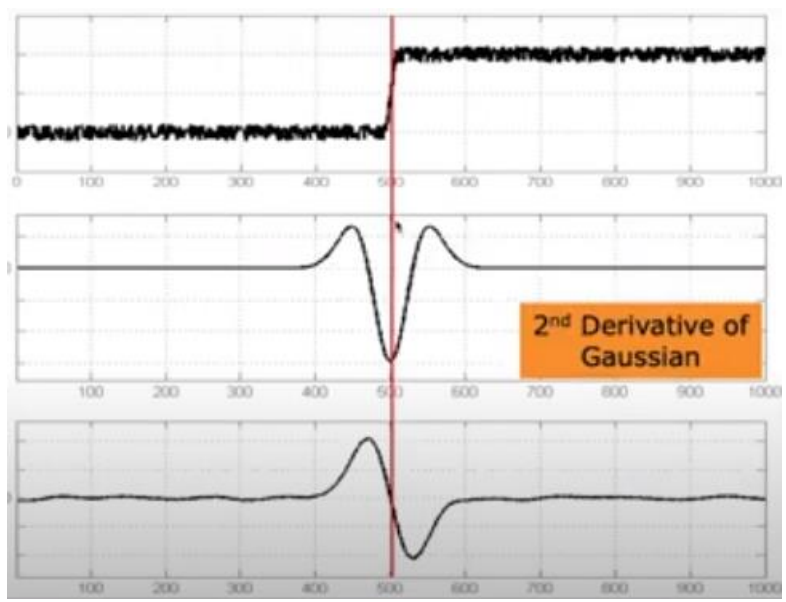
1.3 Detecting blobs

Zero crossing of the second Derivative of Gaussian filter denotes an edge.

f

$\nabla^2(n_\sigma)$

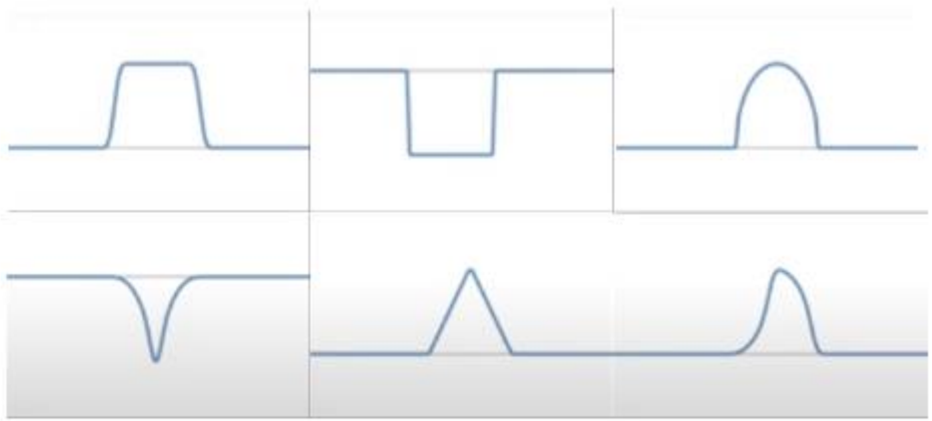
$\nabla^2(n_\sigma) * f$

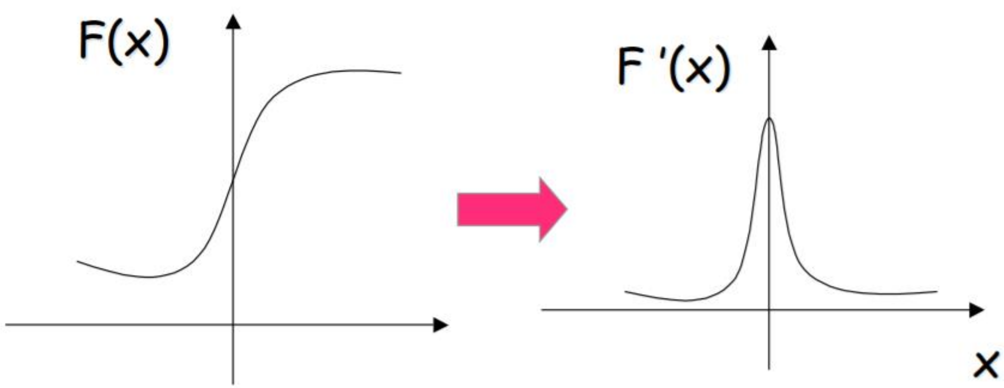


Chapitre 1. SIFT Descriptor and Applications

1.3 Detecting blobs

Example of 1D blob-like structures

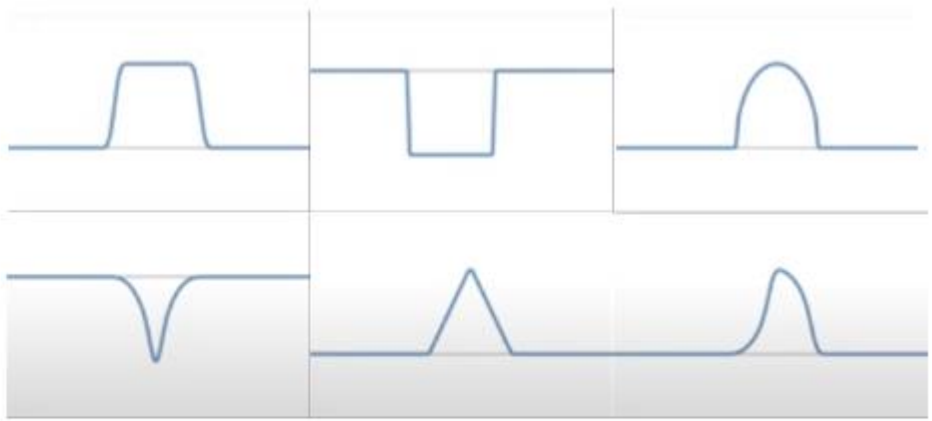


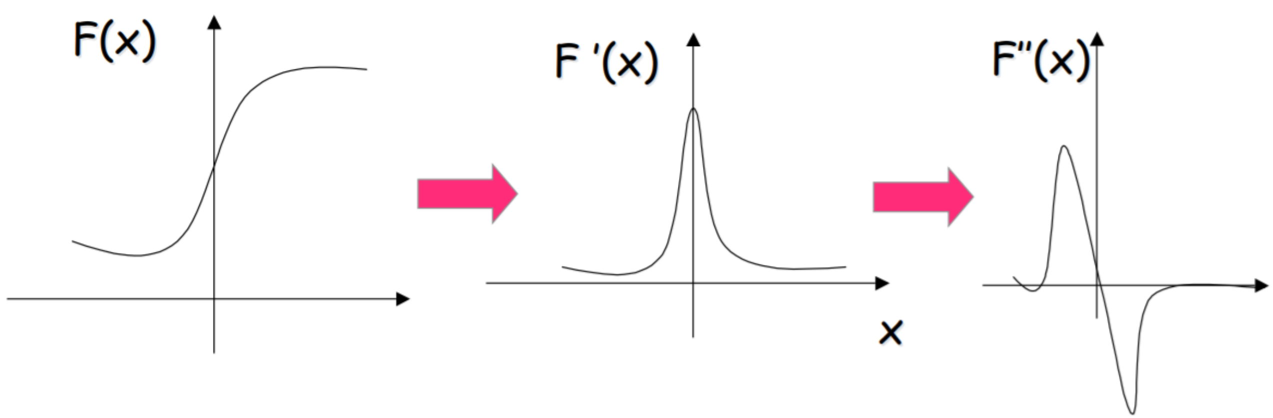


Chapitre 1.   SIFT Descriptor and Applications

1.3 Detecting blobs

Example of 1D blob-like structures





1.3 Detecting blobs

Example of 1D blob-like structures

$$\nabla^2(f(x,y) \otimes G(x,y)) = \nabla^2 G(x,y) \otimes f(x,y)$$

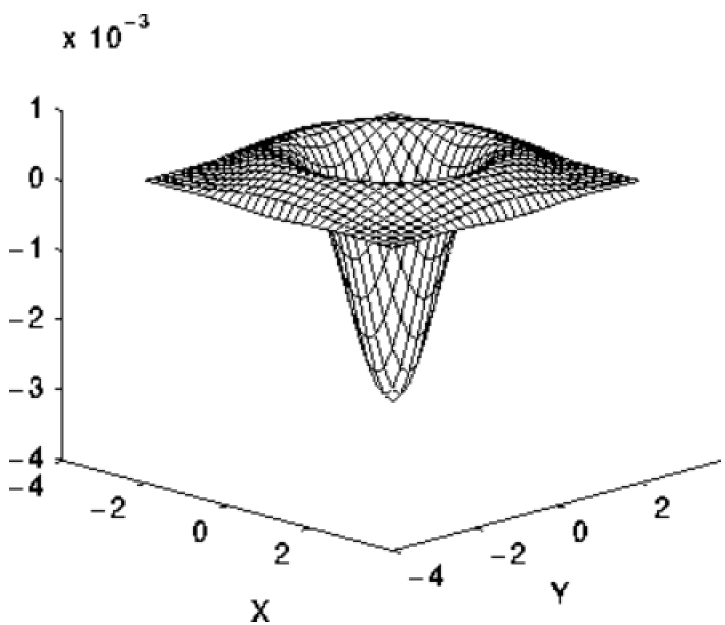
$$G(x,y,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(x^2+y^2)}{2\sigma^2}}$$

Chapitre 1. SIFT Descriptor and Applications

1.3 Detecting blobs

Example of 1D blob-like structures

$$G(x,y,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(x^2+y^2)}{2\sigma^2}} \qquad LoG(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$



0	1	1	2	2	2	1	1	0
1	2	4	5	5	5	4	2	1
1	4	5	3	0	3	5	4	1
2	5	3	-12	-24	-12	3	5	2
2	5	0	-24	-40	-24	0	5	2
2	5	3	-12	-24	-12	3	5	2
1	4	5	3	0	3	5	4	1
1	2	4	5	5	5	4	2	1
0	1	1	2	2	2	1	1	0

for a Gaussian $\sigma = 1.4$

Chapitre 1. SIFT Descriptor and Applications

1.3 Detecting blobs

Example of 1D blob-like structures

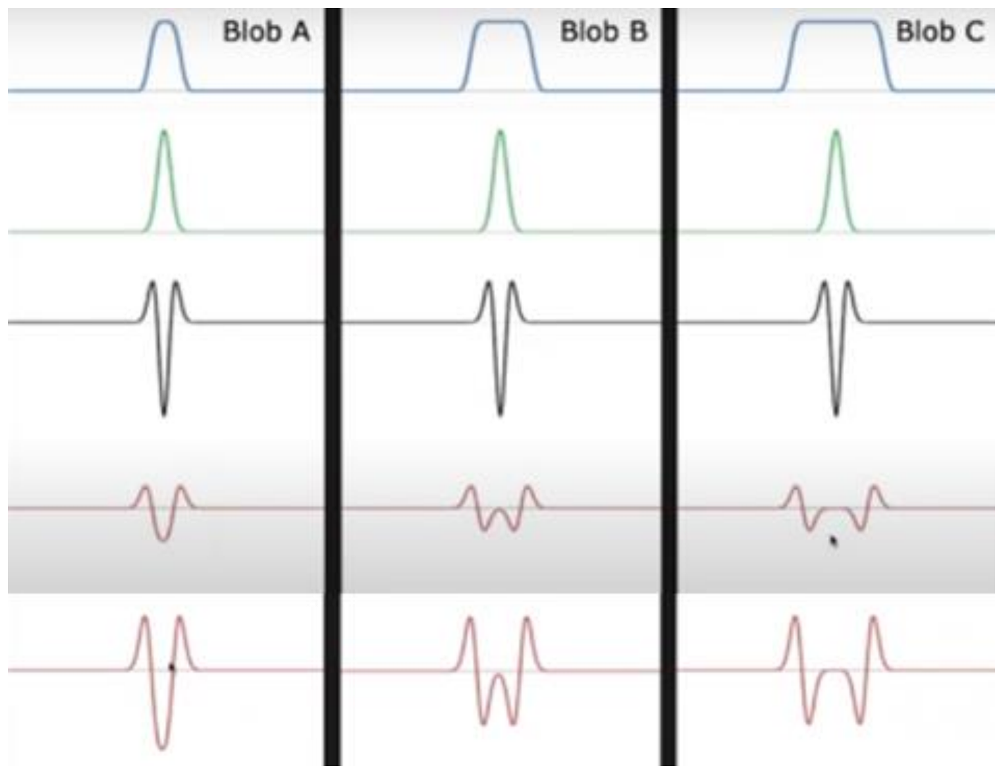
$f(x)$

n_σ

$\frac{\delta^2(n_\sigma)}{\delta x^2}$

$\frac{\delta^2(n_\sigma)}{\delta x^2} * \mathrm{f}$

$\sigma^2 \frac{\delta^2(n_\sigma)}{\delta x^2} * \mathrm{f}$



Chapitre 1. SIFT Descriptor and Applications

1.3 Detecting blobs

If we can find the pic, we have then found the blob. But, in order to find the two other blobs we increase σ.

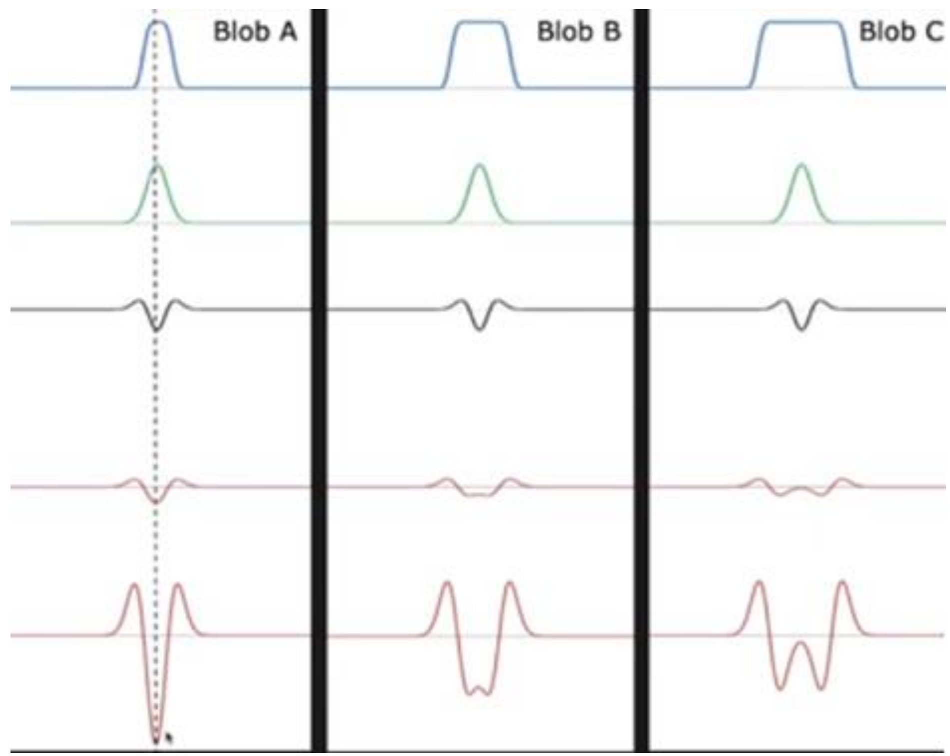
$f(x)$

n_σ

$\frac{\delta^2(n_\sigma)}{\delta x^2}$

$\frac{\delta^2(n_\sigma)}{\delta x^2} * \text{f}$

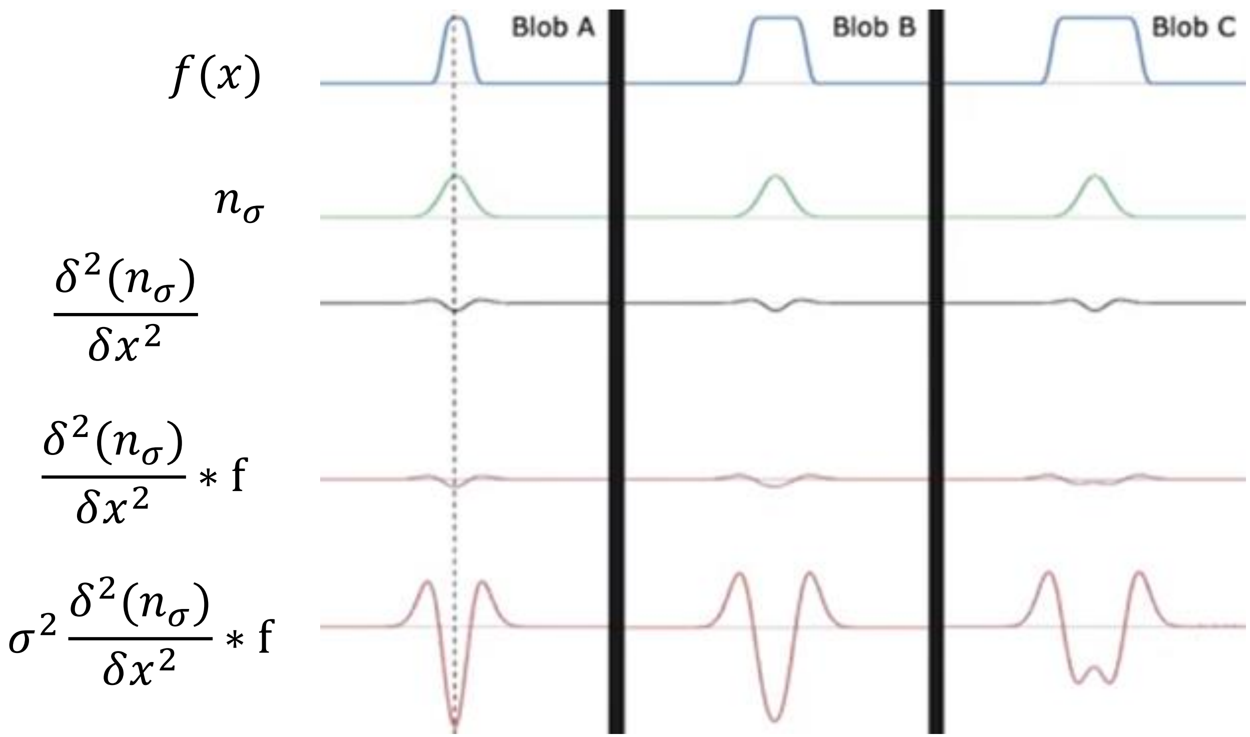
$\sigma^2 \frac{\delta^2(n_\sigma)}{\delta x^2} * \text{f}$



Chapitre 1. SIFT Descriptor and Applications

1.3 Detecting blobs

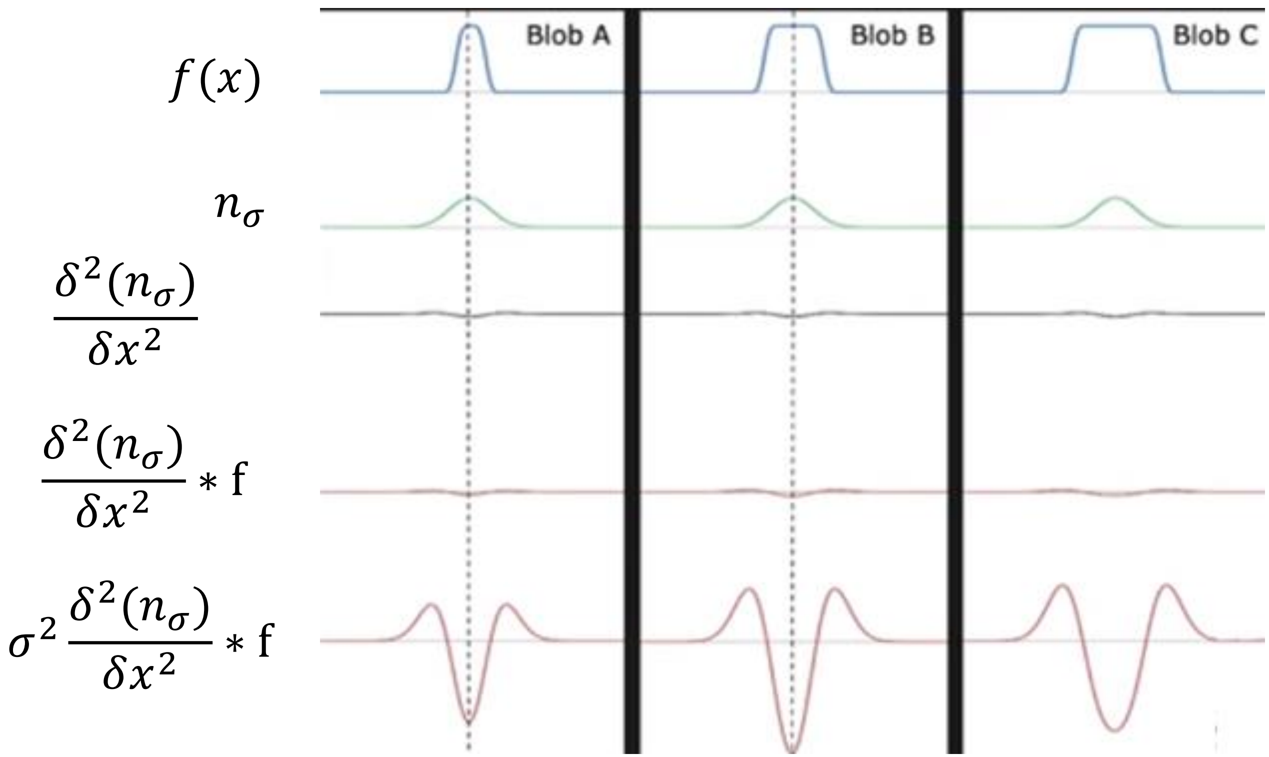
If we can find the pic, we have then found the blob. But, in order to find the two other blobs we continue to increase σ.



Chapitre 1. SIFT Descriptor and Applications

1.3 Detecting blobs

If we can find the pic, we have then found the blob. But, in order to find the two other blobs we continue to increase σ.



1.3 Detecting blobs

If we can find the pic, we have then found the blob. But, in order to find the two other blobs we continue to increase σ.

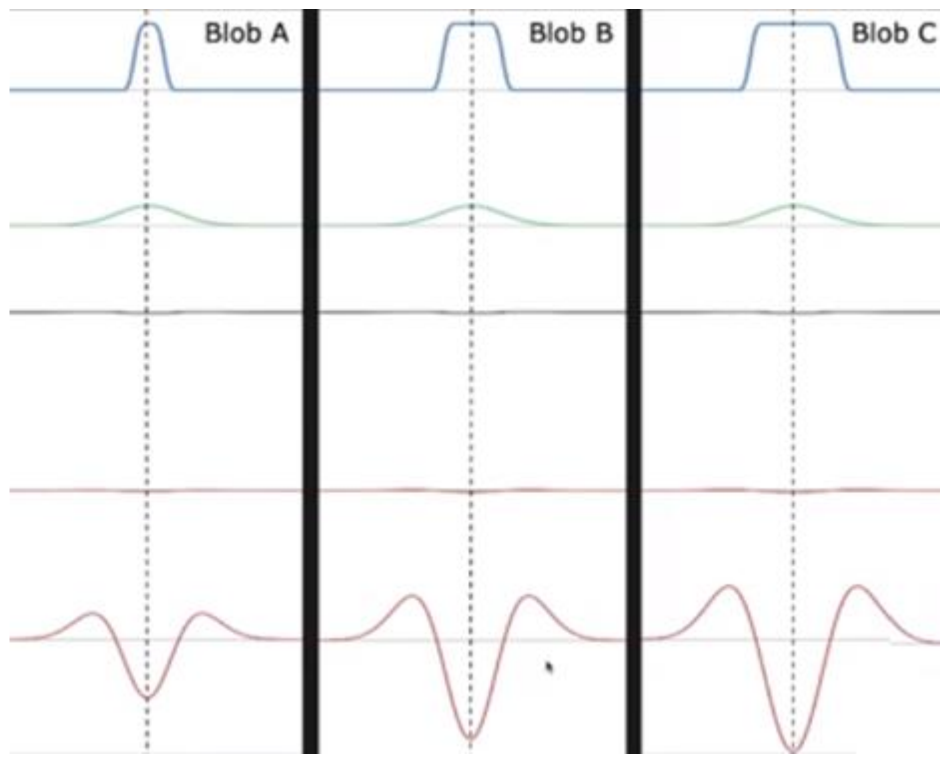
$$f(x)$$

$$n_\sigma$$

$$\frac{\delta^2(n_\sigma)}{\delta x^2}$$

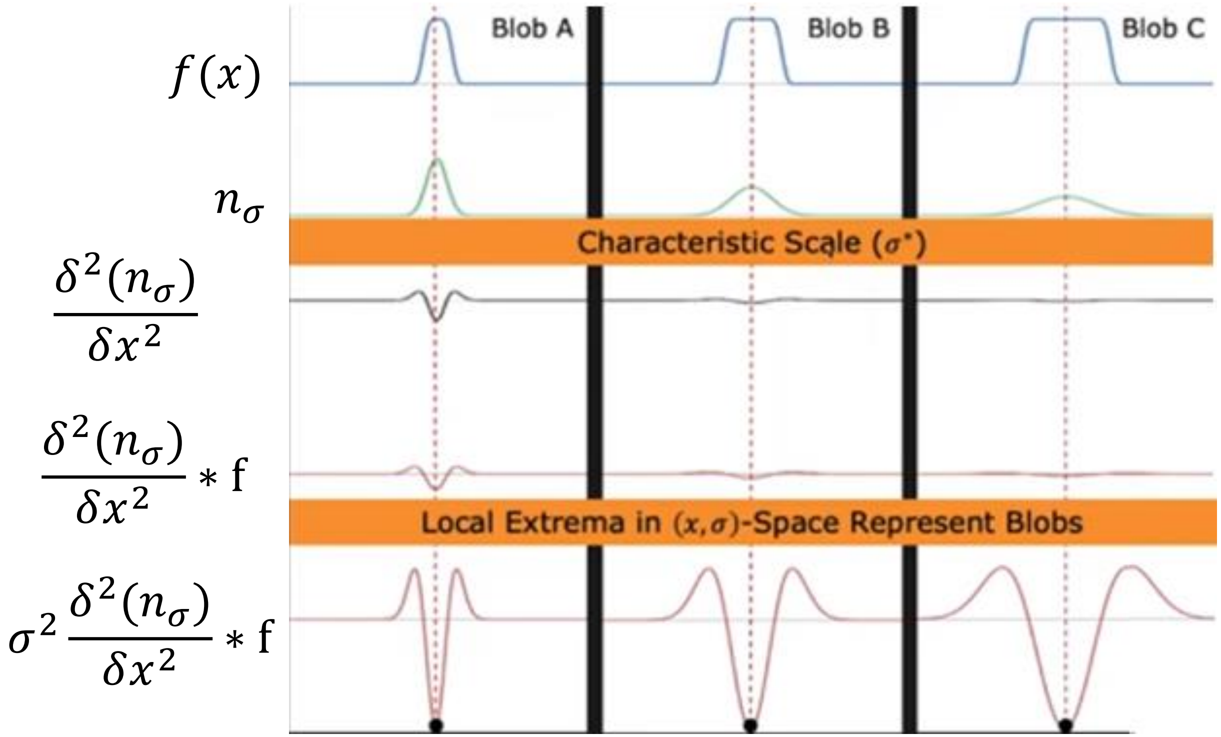
$$\frac{\delta^2(n_\sigma)}{\delta x^2} * \mathrm{f}$$

$$\sigma^2 \frac{\delta^2(n_\sigma)}{\delta x^2} * \mathrm{f}$$



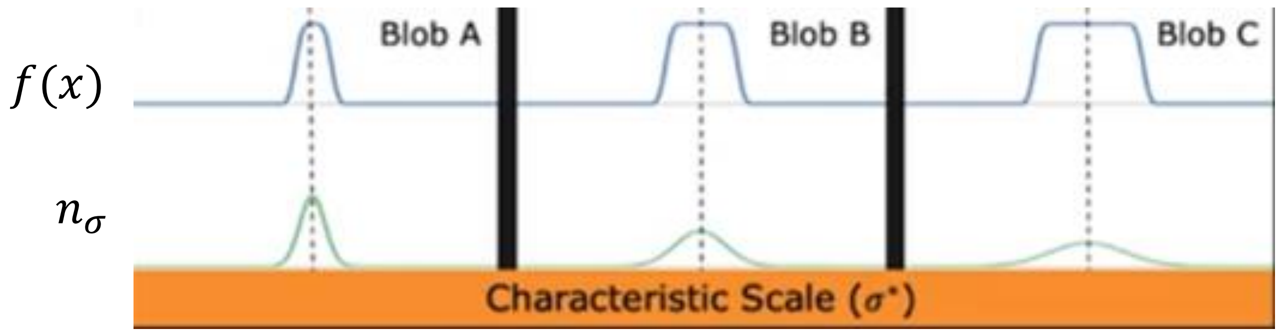
1.3 Detecting blobs

If we can find the pic, we have then found the blob. But, in order to find the two other blobs we continue to increase σ.



1.3 Detecting blobs

If we can find the pic, we have then found the blob. But, in order to find the two other blobs we continue to increase σ.



$\sigma_A^* = \sigma_1$ $\sigma_B^* = 2\sigma_1$ $\sigma_C^* = 3\sigma_1$

Chapitre 1. SIFT Descriptor and Applications

1.3 Detecting blobs

1D Blob Detection Summary

Given 1D signal $f(x)$

Compute $\sigma^2 \frac{\delta^2(n_\sigma)}{\delta x^2} * \text{f}$ at many scales $(\sigma_0, \sigma_1, \sigma_2, \dots, \sigma_k)$

Find: $(x^*, \sigma^*) = \underset{(x,\sigma)}{\arg\max} \left| \sigma^2 \frac{\delta^2(n_\sigma)}{\delta x^2} * \text{f(x)} \right|$

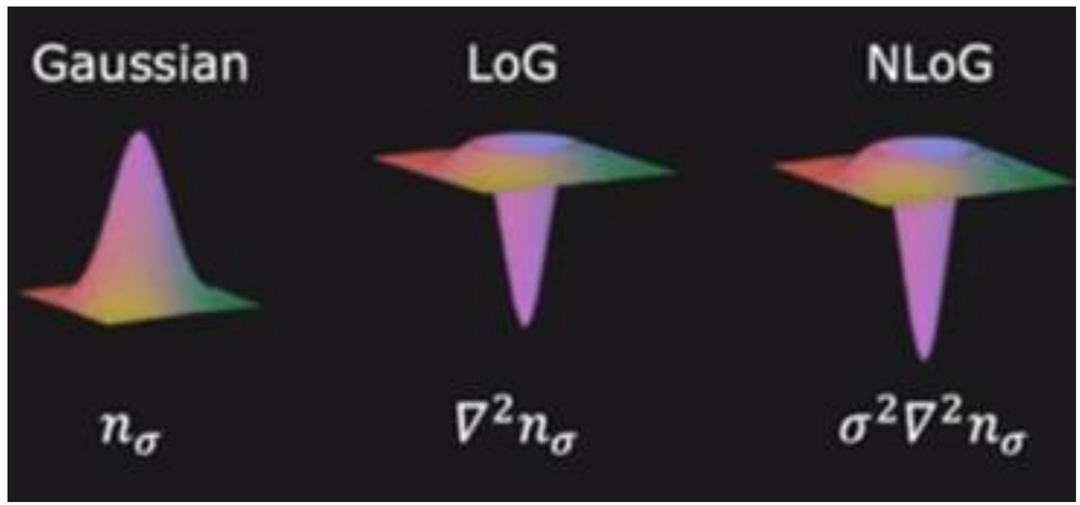
$$\begin{pmatrix} x^*: Blob\ position \\ \sigma^*: Characteristic\ scale(blob\ size) \end{pmatrix}$$

Chapitre 1. SIFT Descriptor and Applications

1.3 Detecting blobs

1D Blob Detection
Normalized Laplacian of Gaussian (NLOG) is used as 2D equivalent for blob detection.

$$\nabla^2 = \frac{\delta^2}{\delta x^2} + \frac{\delta^2}{\delta y^2}$$



Location of blobs given by local extrema after applying Normalized Laplacian of Gaussian at many scales.

Chapitre 1. SIFT Descriptor and Applications

1.3 Detecting blobs

$S(x, y, \sigma_0)$ $S(x, y, \sigma_1)$ $S(x, y, \sigma_2)$

Increasing σ , Higher scale, Low resolution Scale space created by filtering an image with Gaussians of different sigma (σ)

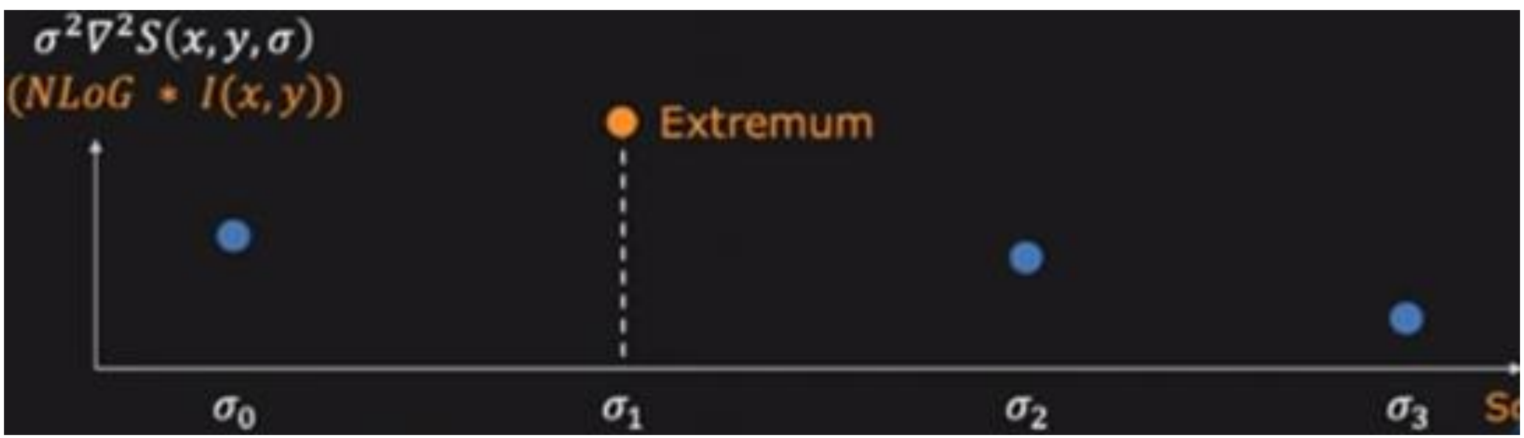
$$S(x, y, \sigma) = n(x, y, \sigma) * f(x, y)$$

$$\sigma^k = \sigma_0 s^k$$

σ_0: Initial scale

S: Constant

1.3 Detecting blobs



Chapitre 1. SIFT Descriptor and Applications

1.3 Detecting blobs

Given an image f(x,y)
Convolve the image using NLoG at many scales σ
Find:
$$(x^*, y^*, \sigma^*) = \underset{(x,y,\sigma)}{\arg\max} |\sigma^2 \nabla^2 n_\sigma * f(x,y)|$$
(x^*, y^*): Position of the blob
σ^*: Size of the blob

Chapitre 1. SIFT Descriptor and Applications

1.4 SIFT Detector

Proposed by David G. Lowe.

Distinctive Image Features from Scale-Invariant Keypoints
International Journal of Computer Vision, 2004
Cited by 65989 papers.

Is widely used in computer vision.

Chapitre 1. SIFT Descriptor and Applications

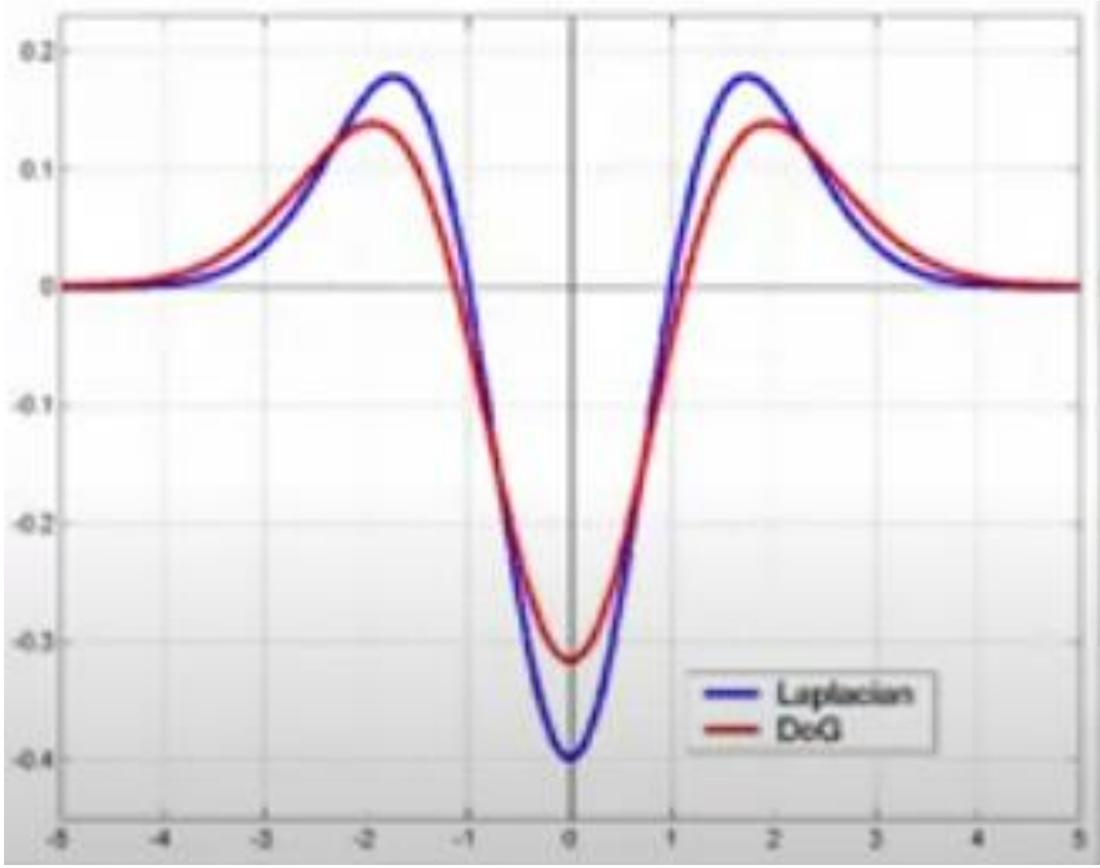
1.4 SIFT Detector

Difference of Gaussian (DOG):

$$DOG = n_{s\sigma} - n_{\sigma} \approx (s-1)\sigma^2 \nabla^2 n_{\sigma}$$

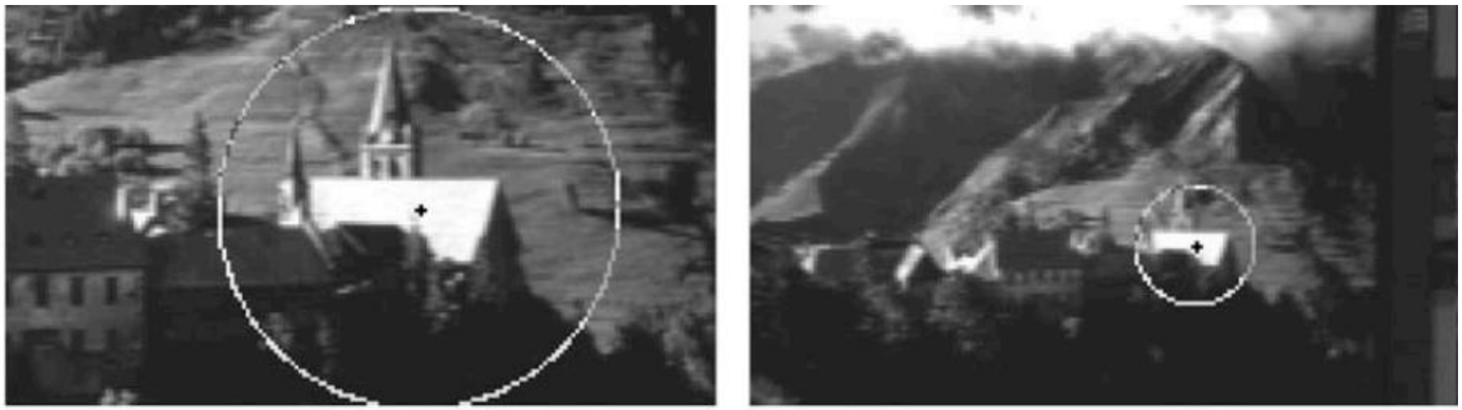
$$DOG \approx (s-1)NLOG$$

NLOG= $\sigma^2 \nabla^2 n_{\sigma}$



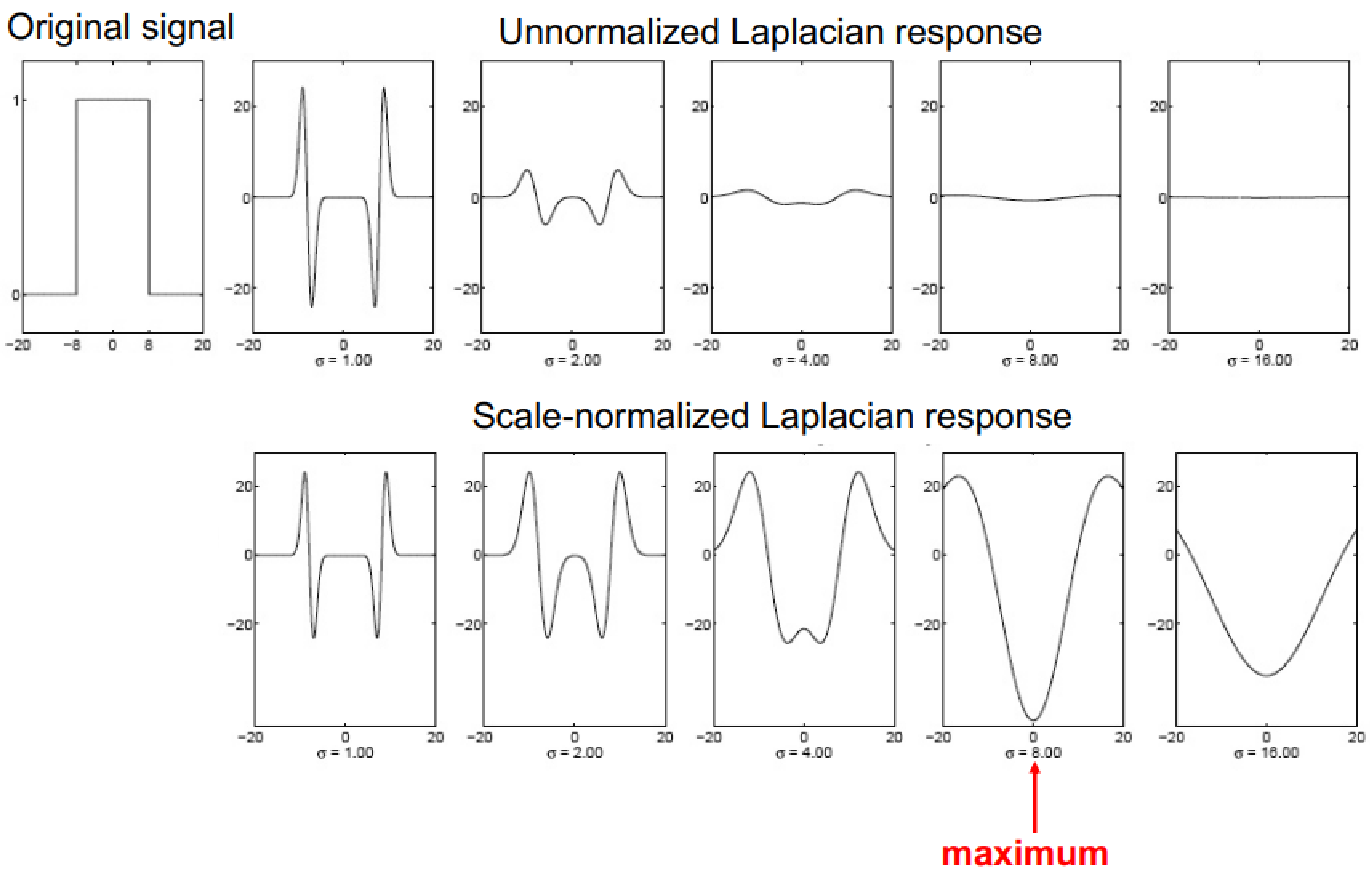
T. Lindeberg, Feature detection with automatic scale selection, IJCV 30(2), pp 77-116, 1998

1.4 SIFT Detector



Chapitre 1. SIFT Descriptor and Applications

1.4 SIFT Detector



Chapitre 1. SIFT Descriptor and Applications

1.4 SIFT Detector

How SIFT is implemented?

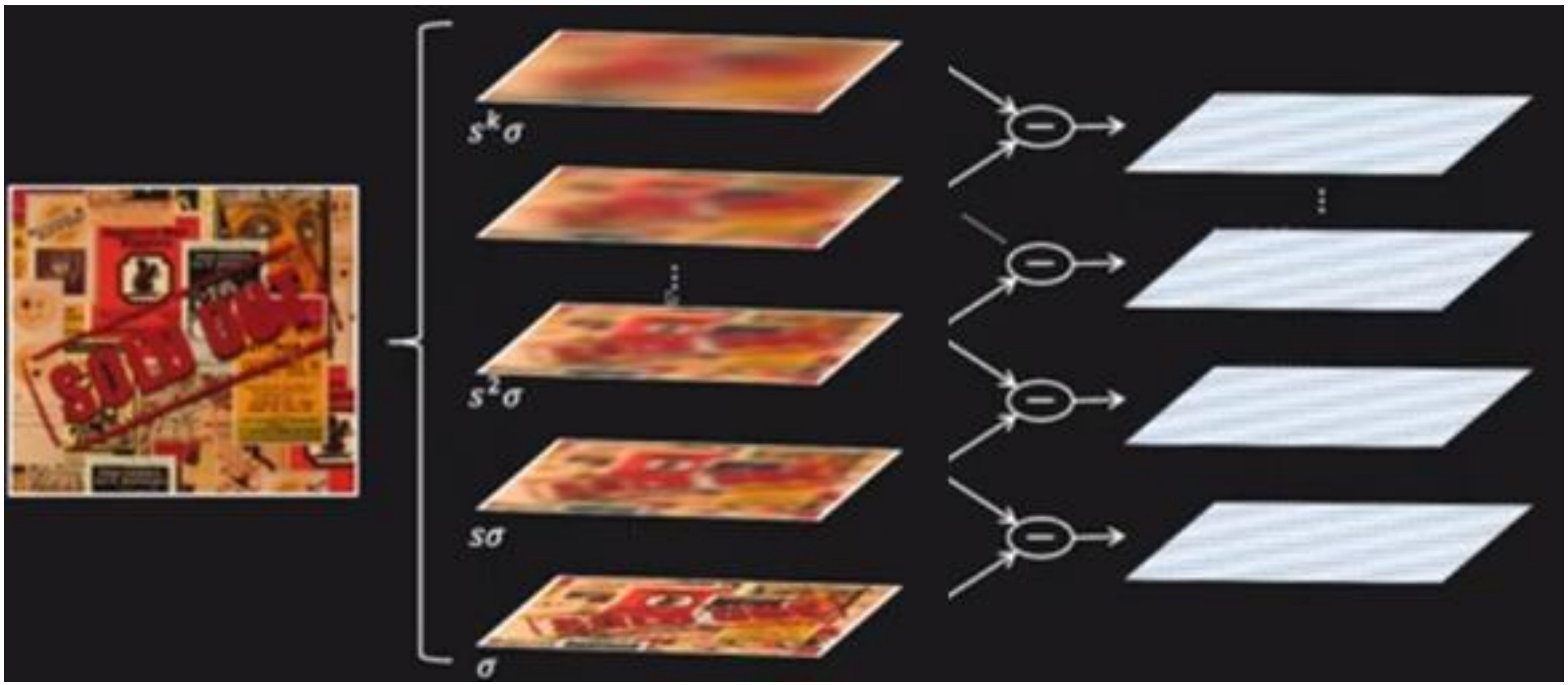


Image I(x,y)

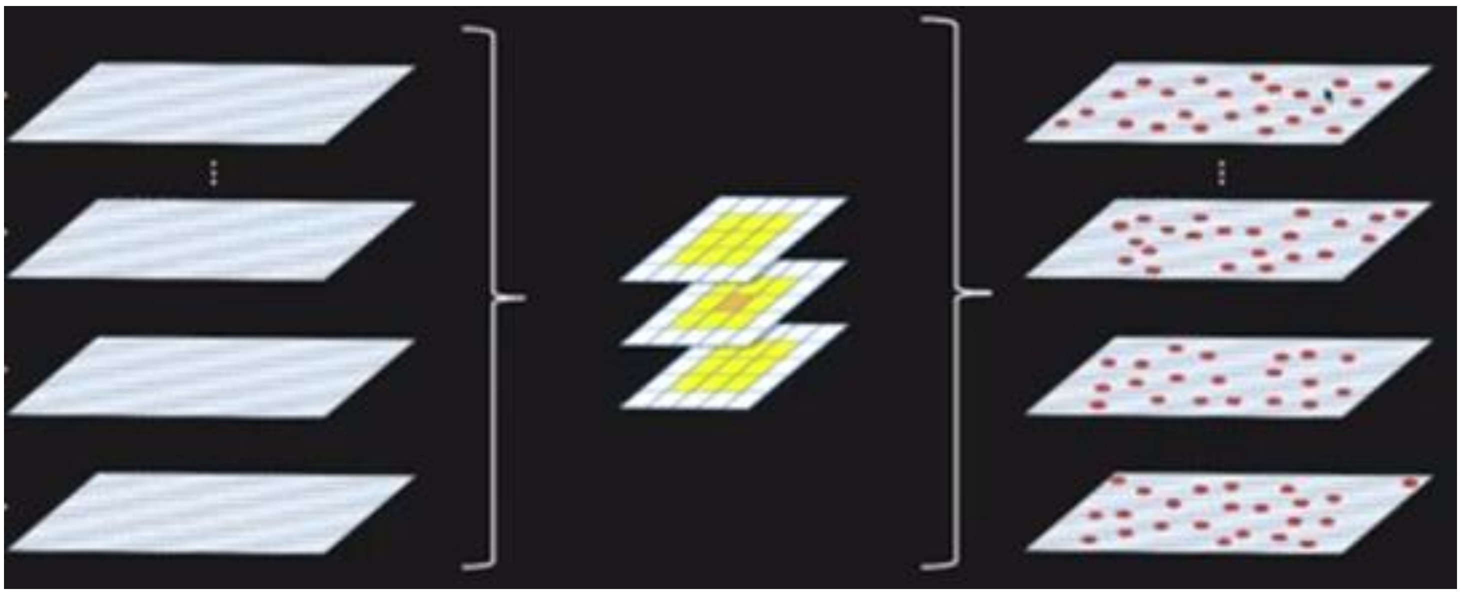
Gaussian Scale-Space S(x,y,σ)

Difference of Gaussians (DoG)= $(s-1)\sigma^2\nabla^2 S(x, y, n_\sigma)$

Chapitre 1. SIFT Descriptor and Applications

1.4 SIFT Detector

How SIFT is implemented?



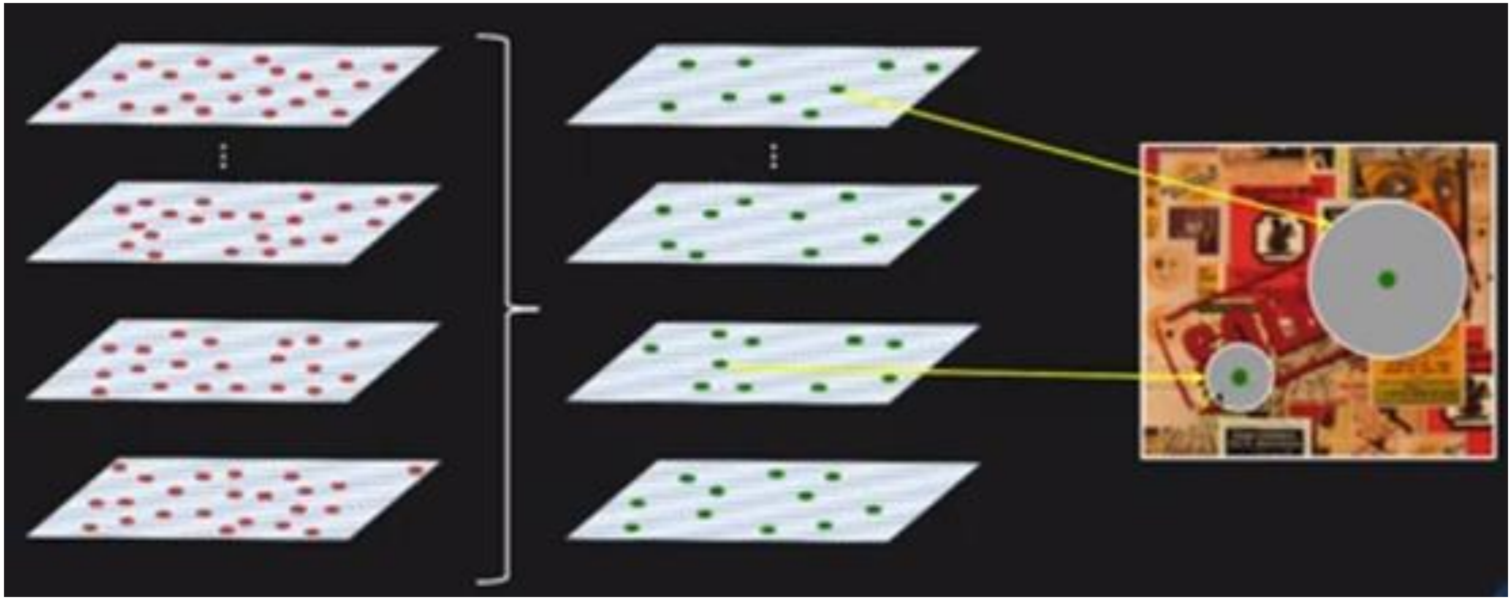
Difference of Gaussians (DoG)

Find Extremum in every 3x3x3 grid

Interest point candidates

1.4 SIFT Detector

How SIFT is implemented?



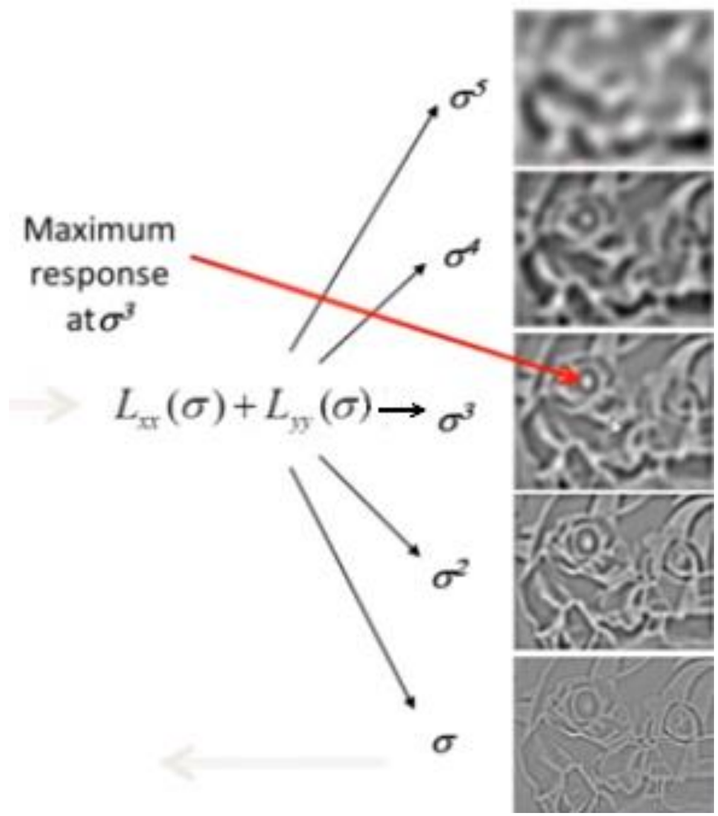
Interest point candidates (includes weak extrema)

SIFT Interest points (after removing weak extrema)

Chapitre 1. SIFT Descriptor and Applications

1.4 SIFT Detector

How SIFT is implemented?

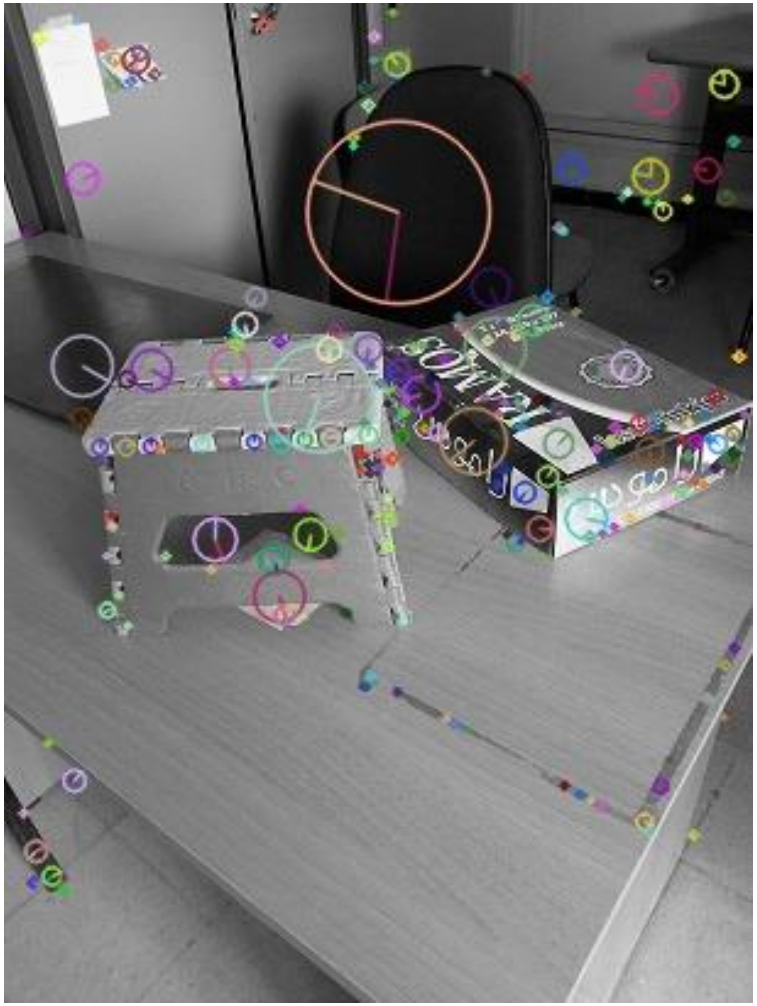


Reference:
https://www.youtube.com/watch?v=U0wqePj4Mx0

Chapitre 1. SIFT Descriptor and Applications

1.4 SIFT Detector

How SIFT is implemented?



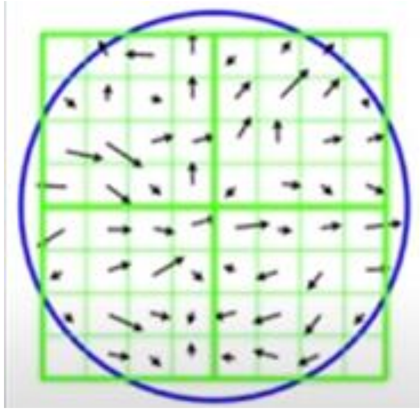
Chapitre 1. SIFT Descriptor and Applications

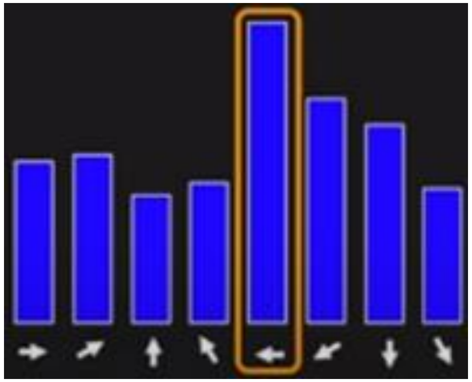
1.4 SIFT Detector

Use the histogram of gradient directions

Compute the image gradient directions θ = $tan^{-1}(\frac{\delta f}{\delta y} / \frac{\delta f}{\delta x})$

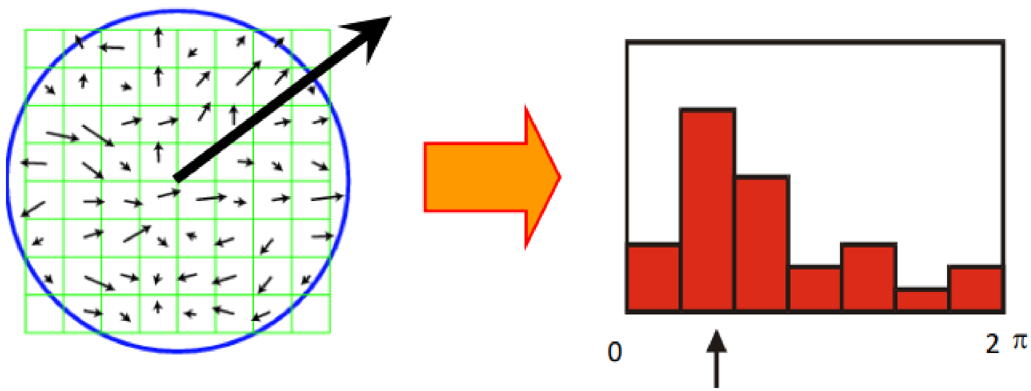
Choose the most prominent gradient direction.





1.4 SIFT Detector

Assign reference orientation at peak of smoothed histogram



The resulting SIFT descriptor is a length 128 vector representing a 4x4 histogram array with 8 orientation bins per histogram.

Chapitre 1. SIFT Descriptor and Applications

1.4 SIFT Detector

SIFT Descriptor

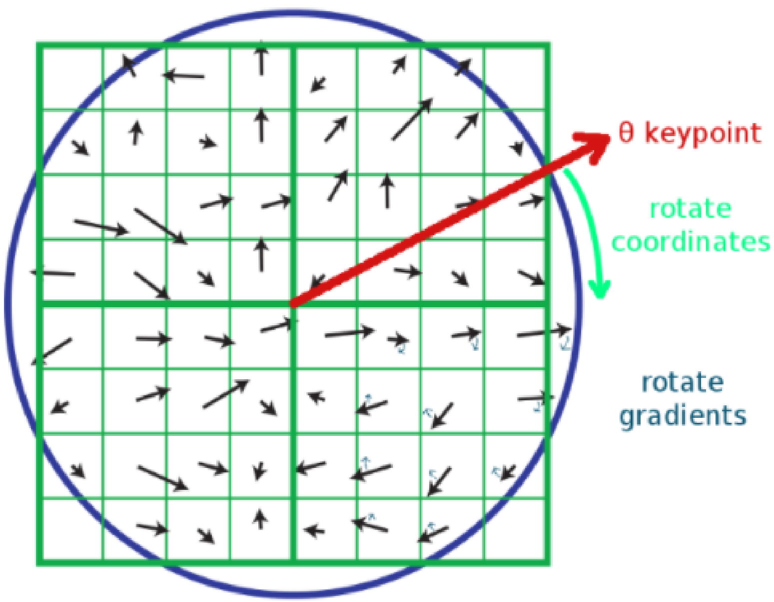
Once the keypoint is found, the next step is to construct a descriptor that contains information of visual characteristics around the keypoint yet is not sensitive to rotation and image illumination.

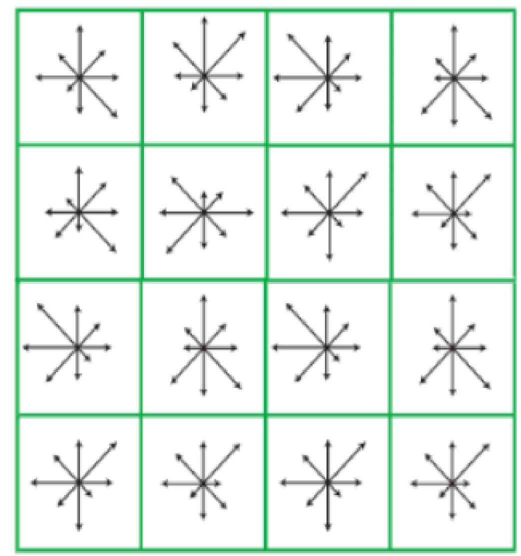
The steps of building the SIFT descriptor are as following:

1. Use the Gaussian blurred image associated with the key point's scale
2. Take image gradients over a 16x16 array
3. Rotate the gradient directions AND locations relative to the keypoint orientation
4. Create an array of orientation histogram.
5. Add the rotated gradients into their local orientation histograms with 8 orientation bins

Chapitre 1. SIFT Descriptor and Applications

1.4 SIFT Detector





Keypoint descriptor

Chapitre 1. SIFT Descriptor and Applications

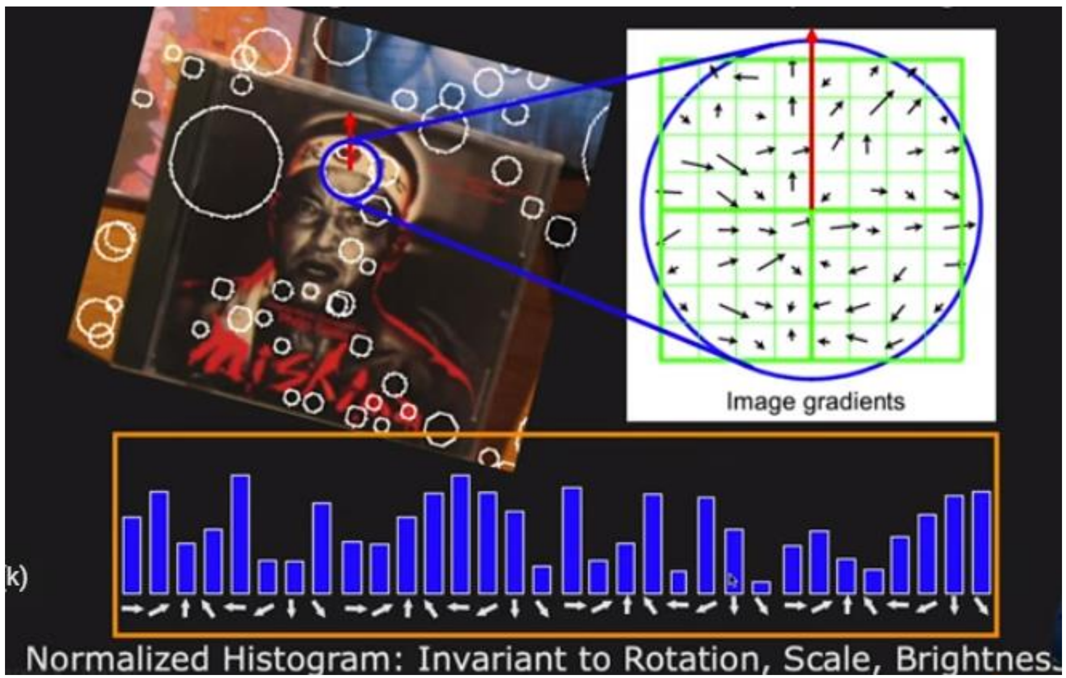
1.4 SIFT Detector

Use the principal orientation to undo rotation

Chapitre 1. SIFT Descriptor and Applications

1.5 SIFT Descriptor

Histogram of gradient directions over spatial directions



Chapitre 1. SIFT Descriptor and Applications

1.5 SIFT Descriptor

Comparing SIFT descriptors

Let $H_1(k), H_2(k)$ be two arrays of data of length N.
L2 Distance:

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

Smaller the distance metric, better the match.
Perfect match when $d(H_1, H_2)$ =0

Chapitre 1. SIFT Descriptor and Applications

1.5 SIFT Descriptor

Comparing SIFT descriptors

Let $H_1(k), H_2(k)$ be two arrays of data of length N.
Normalized Correlation:

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

Where: $\overline{H}_i = \frac{1}{N}\sum_{k=1}^{N} H_i\,(k)$

Larger the distance metric, better the match.
Perfect match when $d(H_1, H_2) = 1$

Chapitre 1. SIFT Descriptor and Applications

1.5 SIFT Descriptor

Comparing SIFT descriptors

Let $H_1(k), H_2(k)$ be two arrays of data of length N.
Intersection:

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

Larger the distance metric, better the match.

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

Overview

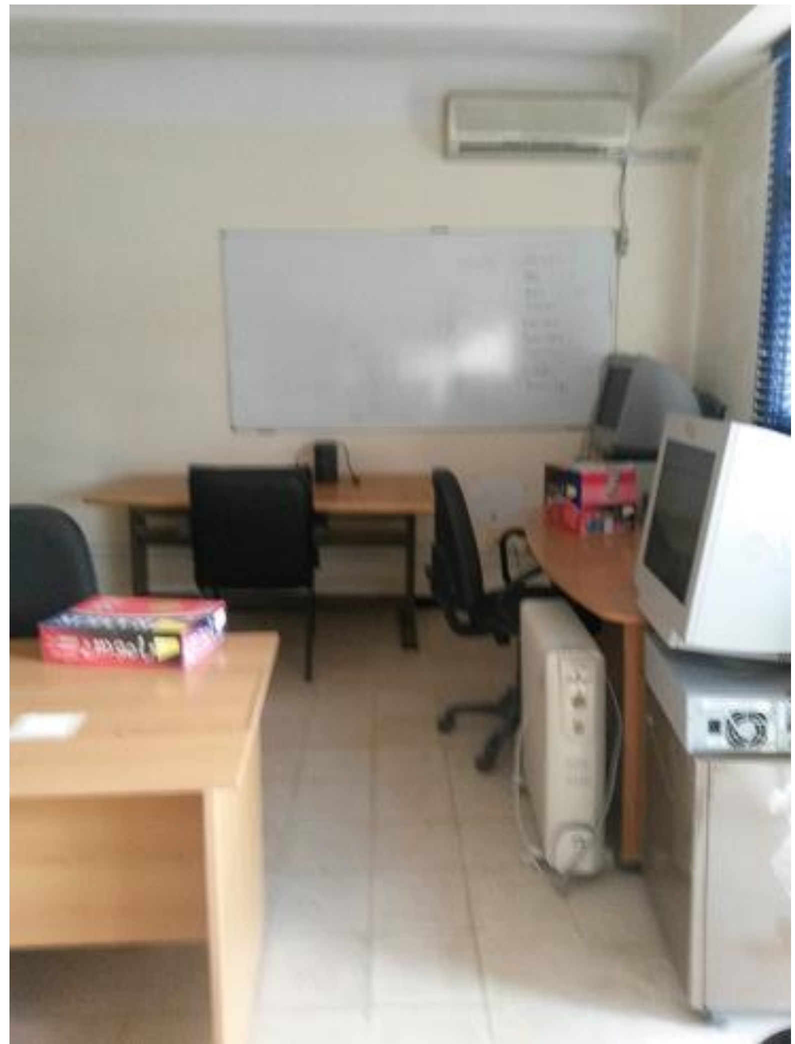
We suppose that we take a set of images of the scene from roughly the same viewpoint, but with rotating the camera

1.6 Image Stitching

Overview

1.6 Image Stitching

Overview



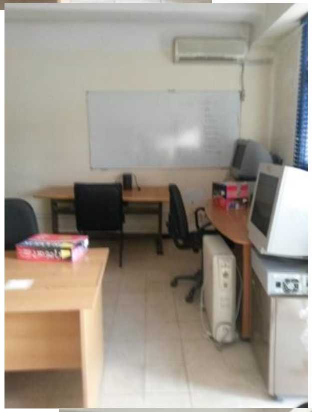
1.6 Image Stitching

Overview

Chapitre 1.   SIFT Descriptor and Applications

1.6 Image Stitching

Overview



We suppose that we take a set of images of the scene from roughly the same viewpoint, but with rotating the camera.
As we are going to make sure that as we rotate the camera, the fields of views of the images overlap.
So we want to take this set of images and automatically create a large image.

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

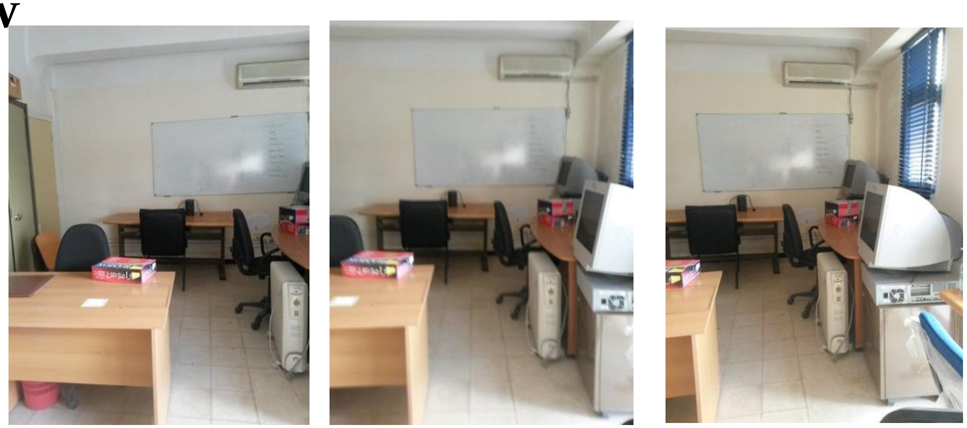
Overview

We want to stitch them together to create a larger image or panorama.

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

Overview

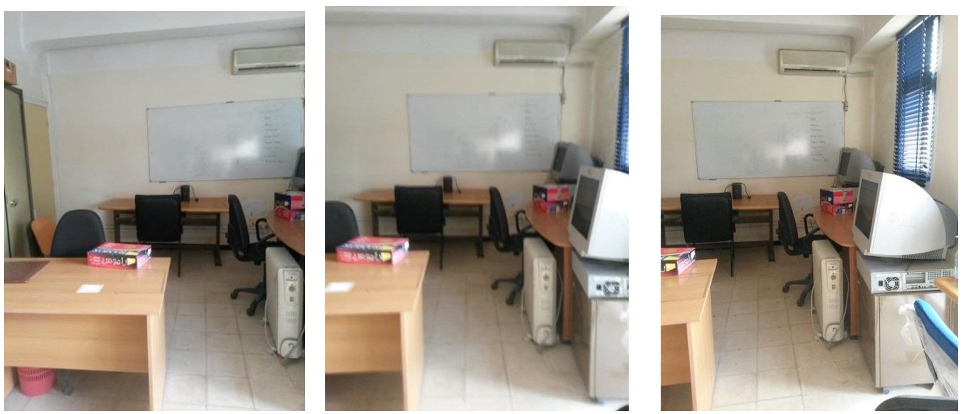


Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

Overview

How would we align these images?
Find corresponding points (using feature detectors like SIFT)
SIFT is perfect for this application.

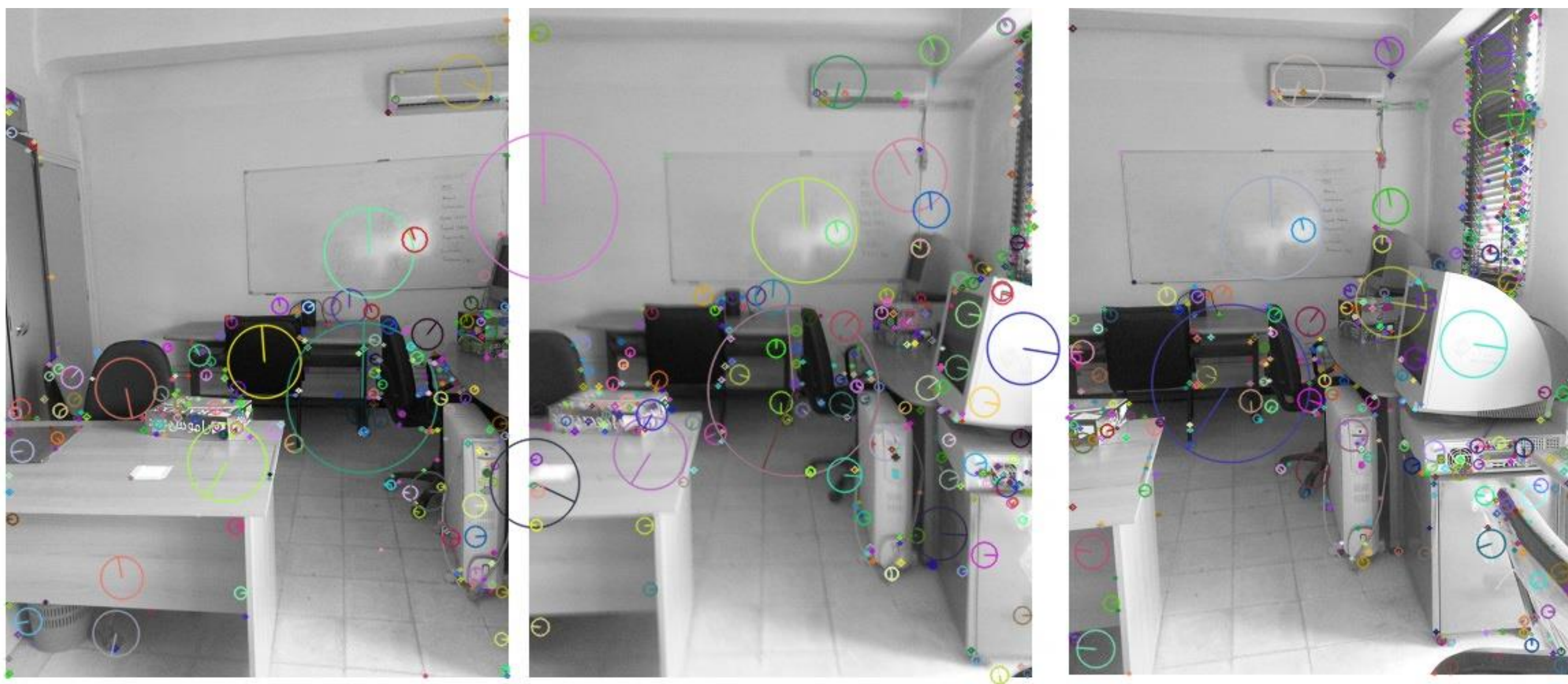


Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

Overview

Find corresponding points (using feature detectors like SIFT)



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

Overview

Find geometric relationship between the images.
What is the transformation that take you from one image to another.
So we can take an image and wrap it to the coordinate frame of the other image.
This transformation is called the Homography.

Image 1 Image 2 Image 3

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

Overview

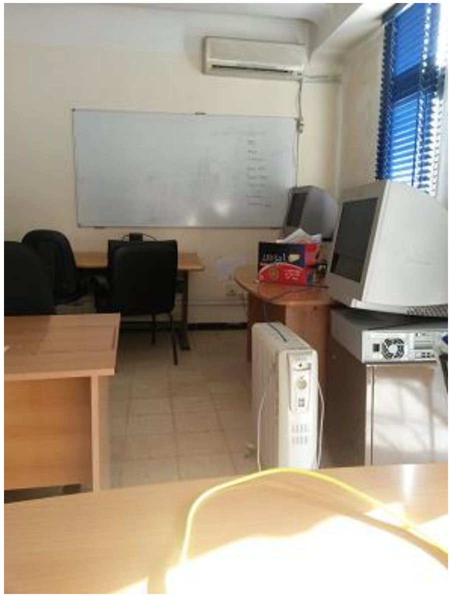
Find geometric relationship between the images.
Once we are able to do that, we can actually wrap the images
to a common coordinates frame and we get a set of a stack of overlapping images
That look like this.

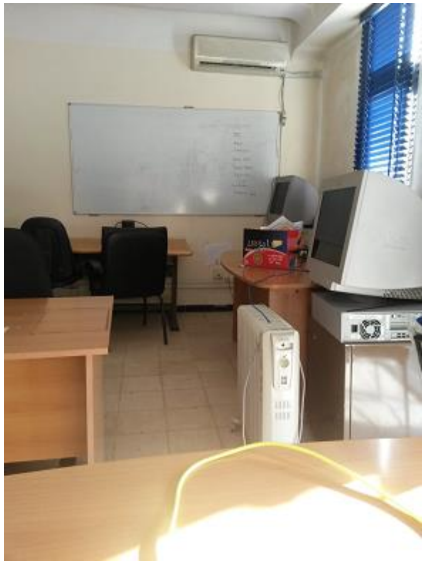
Image 1 Image 2 Image 3

1.6 Image Stitching

Overview

Wrap images so that corresponding points align.
Once we are able to do that, we can actually wrap the images
to a common coordinates frame and we get a set of a stack of overlapping images
That look like this.

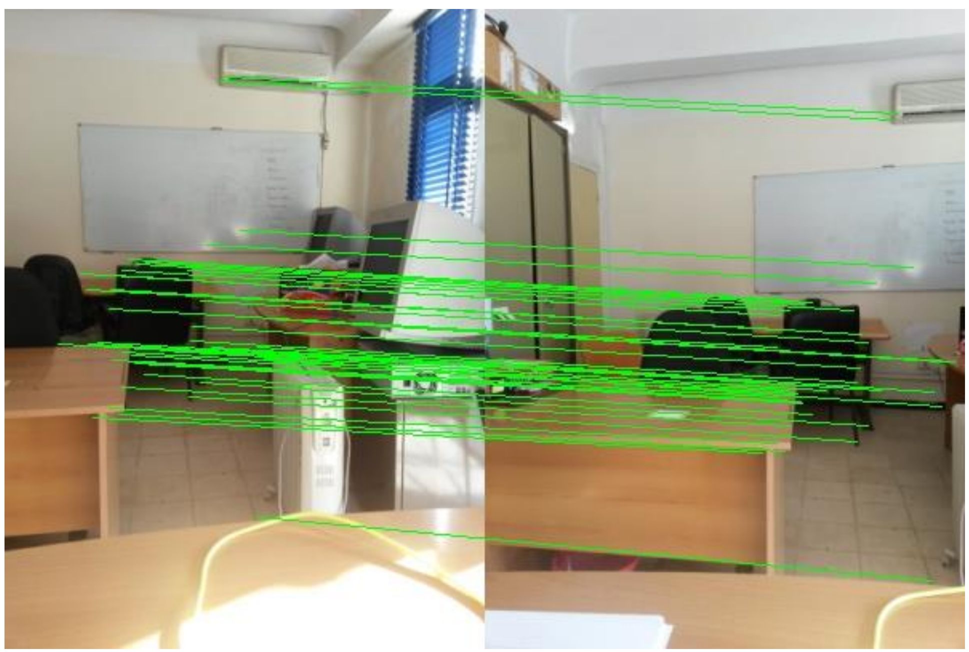




1.6 Image Stitching

Overview

Wrap images so that corresponding points align.
Once we are able to do that, we can actually wrap the images
to a common coordinates frame and we get a set of a stack of overlapping images
That look like this.



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

Overview

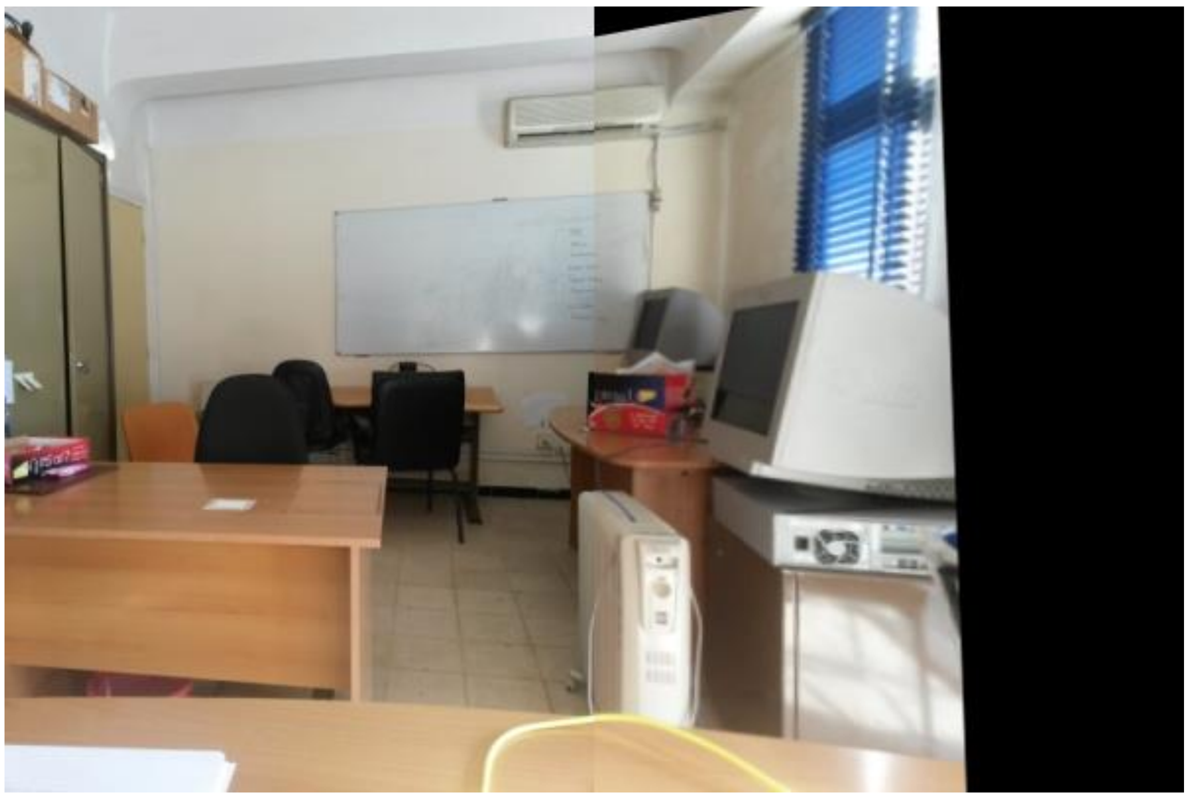
Wrap images so that corresponding points align.
Once we are able to do that, we can actually wrap the images
to a common coordinates frame and we get a set of a stack of overlapping images
That look like this.

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

Overview

Wrap images so that corresponding points align.
Once we are able to do that, we can actually wrap the images
to a common coordinates frame and we get a set of a stack of overlapping images
That look like this.



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

Overview

Blend images to remove hard seams.
One last problem to solve is the problem of seams.
No two images are captured with exactly the same exposure.

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

Overview

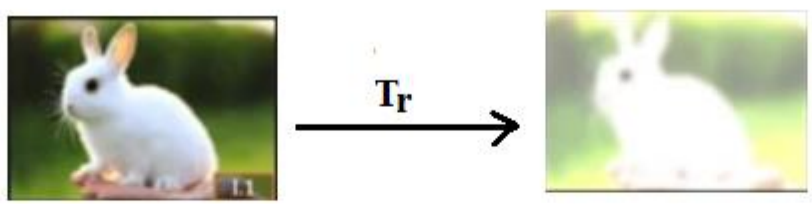
Our aim is to combine multiple photos to create a larger photo.
We will discuss the following topics:
- 2x2 Image Transformations
- 3x3 Image Transformations
- Computing Homography
- Dealing with Outliers: RANSAC
- Warping and Blending images

1.6 Image Stitching

2x2 Image Transformations

Image manipulation

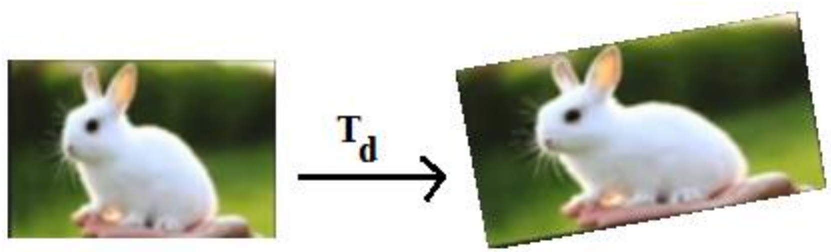
Image Filtering: Change range(Brightness)



$$g(x,y) = T_r(f(x,y)$$

Image Warping: Change domain(Location)

$$g(x,y) = f(T_d(x,y))$$



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

2x2 Image Transformations

Image manipulation

Rotation

$$g(x, y) = f(T(x, y))$$

Projective

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

2x2 Image Transformations

2x2 Linear Transformations

$\xrightarrow{T}$

$p_1(x_1, y_1)$ $p_2(x_2, y_2)$

$$p_2 = Tp_1$$

T can be represented by a Matrix

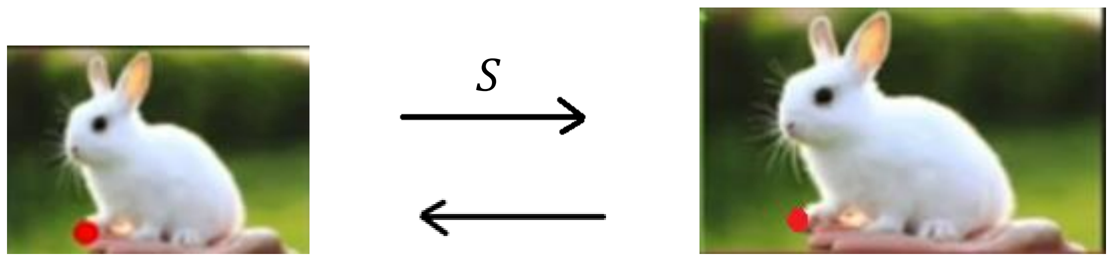
$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = T \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

Chapitre 1.   SIFT Descriptor and Applications

1.6 Image Stitching

2x2 Image Transformations

Scaling (stretching or squishing)



S^{-1}

$p_2(x_2, y_2)$

$p_1(x_1, y_1)$

$$p_2 = Tp_1$$

T can be represented by a Matrix

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = S \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

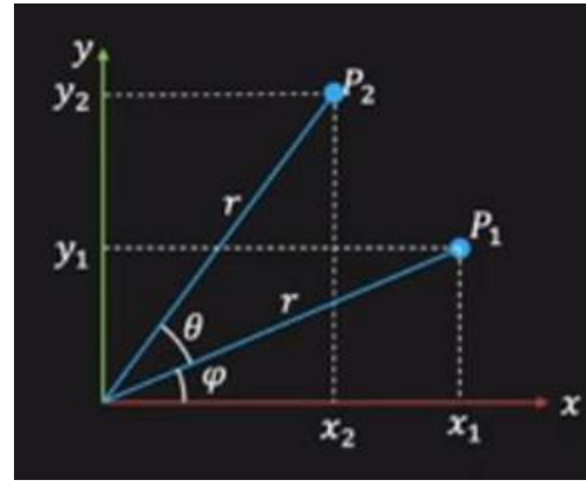
$$S^{-1} = \begin{bmatrix} 1/a & 0 \\ 0 & 1/b \end{bmatrix}$$

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

2x2 Image Transformations

2D Rotation



$x_1 = r\cos(\varphi)\,, y_1 = \mathrm{r}\sin(\varphi)$

$x_2 = r\cos(\varphi+\theta)$

$x_2 = \mathrm{r}\cos(\varphi)\cos(\theta)\text{-}\sin(\varphi)\,\mathrm{r}\sin(\theta)$

$x_2 = x_1\cos\theta\text{-}y_1\sin(\theta)$

$y_2 = \mathrm{r}\sin(\varphi+\theta)$

$y_2 = \mathrm{r}\cos(\varphi)\sin(\theta) + \sin(\varphi)\,\mathrm{r}\cos(\theta)$

$y_2 = x_1\sin\theta + y_1\cos(\theta)$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = R\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}\begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

$$R^{-1} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

1.6 Image Stitching

2x2 Image Transformations

Skew

Horizontal Skew: $x_2 = x_1 + m_x y_1$, $y_2 = y_1$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = S_x \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} 1 & m_x \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

Vertical Skew: $x_2 = x_1$, $y_2 = y_1 + m_y x_1$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = S_y \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ m_y & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

2x2 Image Transformations

Any other transformation

Any transformation of the form: $\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = T \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$

- Origin maps to the origin
- Lines map to lines
- Parallel lines remain parallel
- Closed under composition:

$$p_2 = T_{21} p_1 , \qquad p_3 = T_{32} p_2 , \qquad p_3 = T_{31} p_1$$

$$p_3 = T_{32} T_{21} p_1 \qquad \Rightarrow T_{32} T_{21} p_1 = T_{31} p_1$$

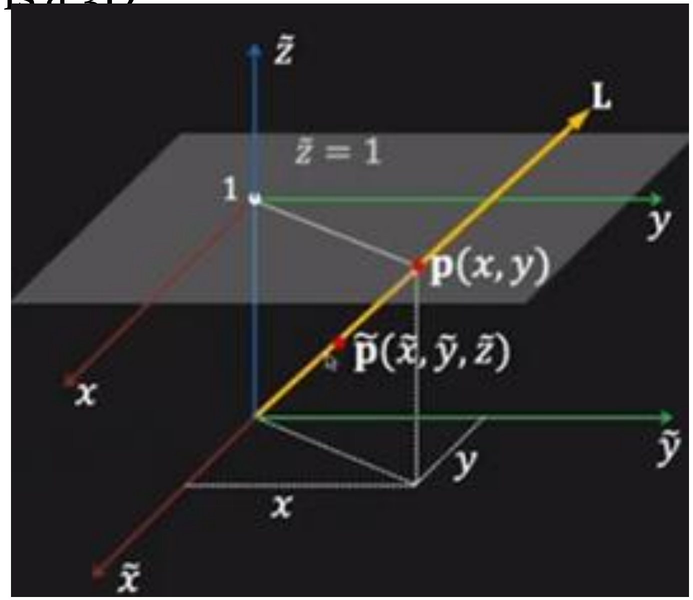
1.6 Image Stitching

3x3 Image Transformations

Homogeneous coordinates

Translation can't expressed by 2x2 matrix.
To express linearity, **Homogeneous coordinates** are widely used in science of engineer.
The homogeneous representation of a 2D point p=(x,y) is a 3D point $\tilde{p} = (\tilde{x}, \tilde{y}, \tilde{z})$.
The third coordinate $\tilde{z}$ is fictitious such that:

$$x = \frac{\tilde{x}}{\tilde{z}} \ , \tilde{y} = \frac{\tilde{y}}{\tilde{z}}$$



Every line on (L), except the origin, represent the homogeneous coordinates of p(x,y)

$$p = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \equiv \begin{bmatrix} \tilde{z}x \\ \tilde{z}y \\ \tilde{z} \end{bmatrix} \equiv \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix} = \tilde{p}$$

1.6 Image Stitching

3x3 Image Transformations

Homogeneous coordinates

Examples:

Scaling:

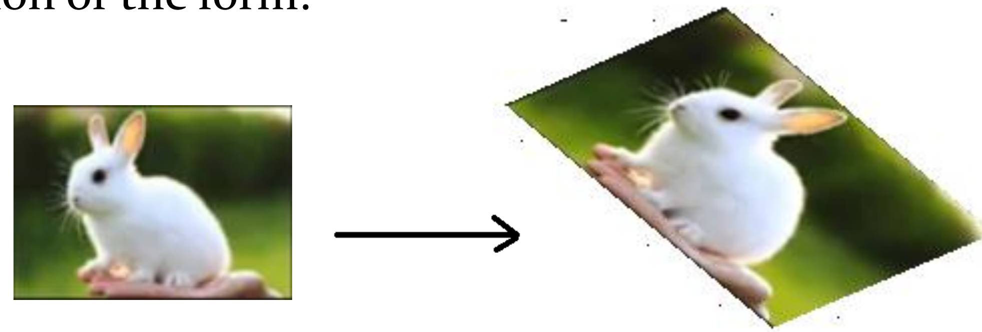
$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix} = \begin{bmatrix} sx & 0 & 0 \\ 0 & sy & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Translation:

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Affine transformation: Any transformation of the form:

$$\begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} \equiv \begin{bmatrix} \tilde{x}_2 \\ \tilde{y}_2 \\ \tilde{z}_2 \end{bmatrix} = \begin{bmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{y}_1 \\ \tilde{z}_1 \end{bmatrix}$$



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

3x3 Image Transformations

Affine transformation: Any transformation of the form:

Any affine transformation :

- Origin does not necessarily maps to the origin
- Lines map to lines
- Parallel lines remain parallel
- Closed under composition:

$$\begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} \equiv \begin{bmatrix} \tilde{x}_2 \\ \tilde{y}_2 \\ \tilde{z}_2 \end{bmatrix} = \begin{bmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{y}_1 \\ \tilde{z}_1 \end{bmatrix}$$

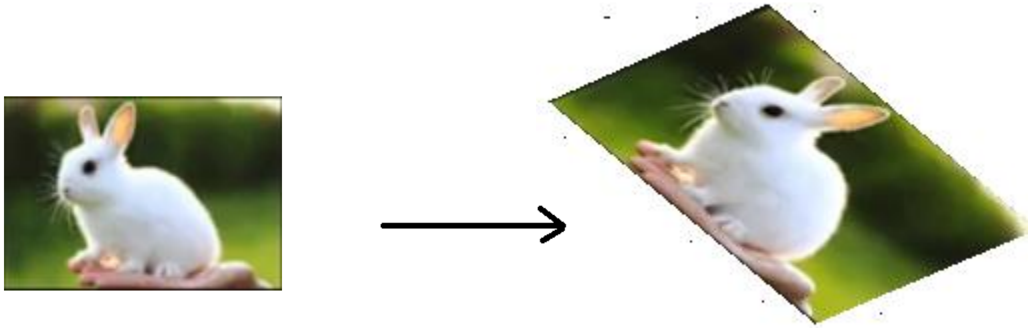
1.6 Image Stitching

3x3 Image Transformations

Projective transformation: Any transformation of the form:

$$\begin{bmatrix} \tilde{x}_2 \\ \tilde{y}_2 \\ \tilde{z}_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{y}_1 \\ \tilde{z}_1 \end{bmatrix}$$

$$\tilde{p}_2 = H\tilde{p}_1$$



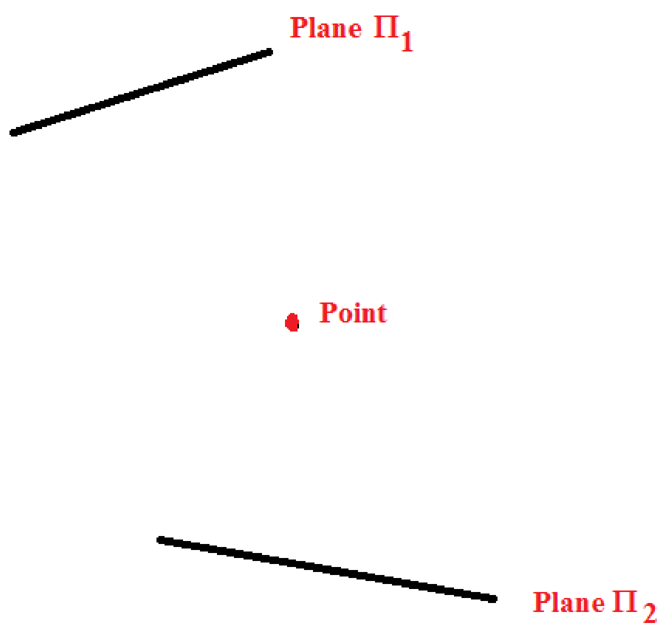
Is called **HOMOGRAPHY**

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

3x3 Image Transformations

Projective transformation: mapping one plane to another through a point:



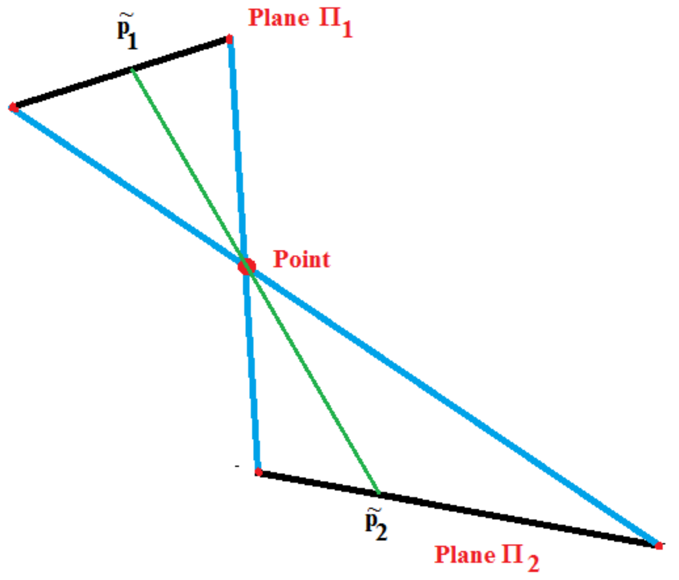
1.6 Image Stitching

3x3 Image Transformations

Projective transformation: mapping one plane to another through a point:

$$\begin{bmatrix} \tilde{x}_2 \\ \tilde{y}_2 \\ \tilde{z}_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{y}_1 \\ \tilde{z}_1 \end{bmatrix}$$

$$\tilde{p}_2 = H\tilde{p}_1$$



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

3x3 Image Transformations

Projective transformation: Homography can only be defined up to scale. Homography is the transformation matrix that take from one plane to another through a point of projection.

$$\begin{bmatrix} \tilde{x}_2 \\ \tilde{y}_2 \\ \tilde{z}_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{y}_1 \\ \tilde{z}_1 \end{bmatrix} \equiv k \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{y}_1 \\ \tilde{z}_1 \end{bmatrix}$$

Suppose that y=Hx. Then by elementary properties of matrix multiplication, (λH)x=λ(Hx)=λy, so
when λ≠0, Hy and (λH)y represent the same point.

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching

3x3 Image Transformations

Projective transformation: Homography can only be defined up to scale. Homography is the transformation matrix that take from one plane to another through a point of projection.

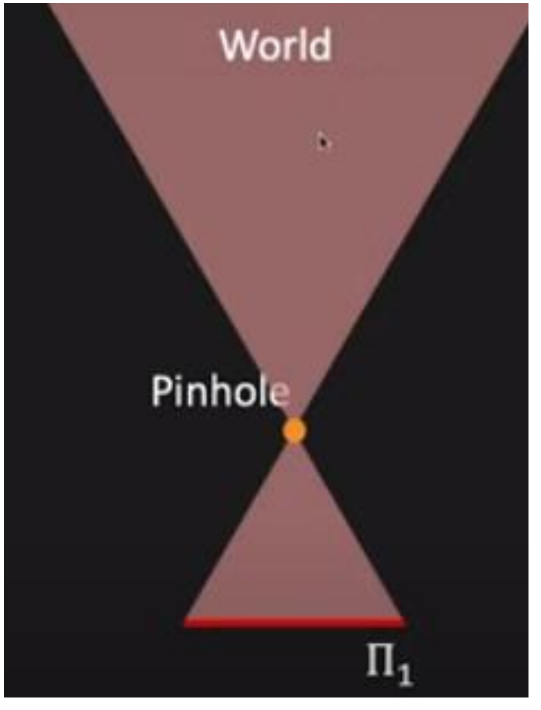
$$\begin{bmatrix} \tilde{x}_2 \\ \tilde{y}_2 \\ \tilde{z}_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{y}_1 \\ \tilde{z}_1 \end{bmatrix} \equiv k \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{y}_1 \\ \tilde{z}_1 \end{bmatrix}$$

If we fix $\sqrt{\sum_{ij} h_{ij}} = 1$, scale such that then 8 free parameters.

- Origin does not necessarily maps to the origin
- Lines map to lines
- Parallel lines does not necessarily remain parallel
- Closed under composition:

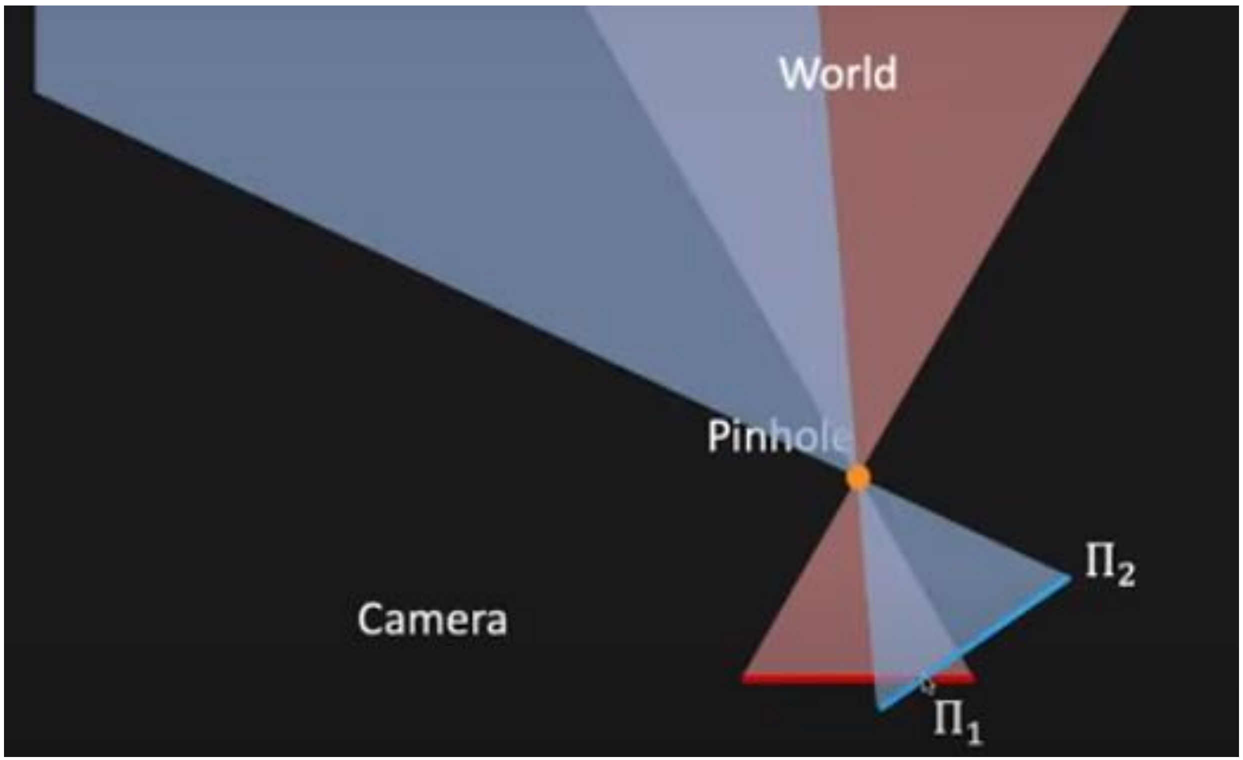
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography



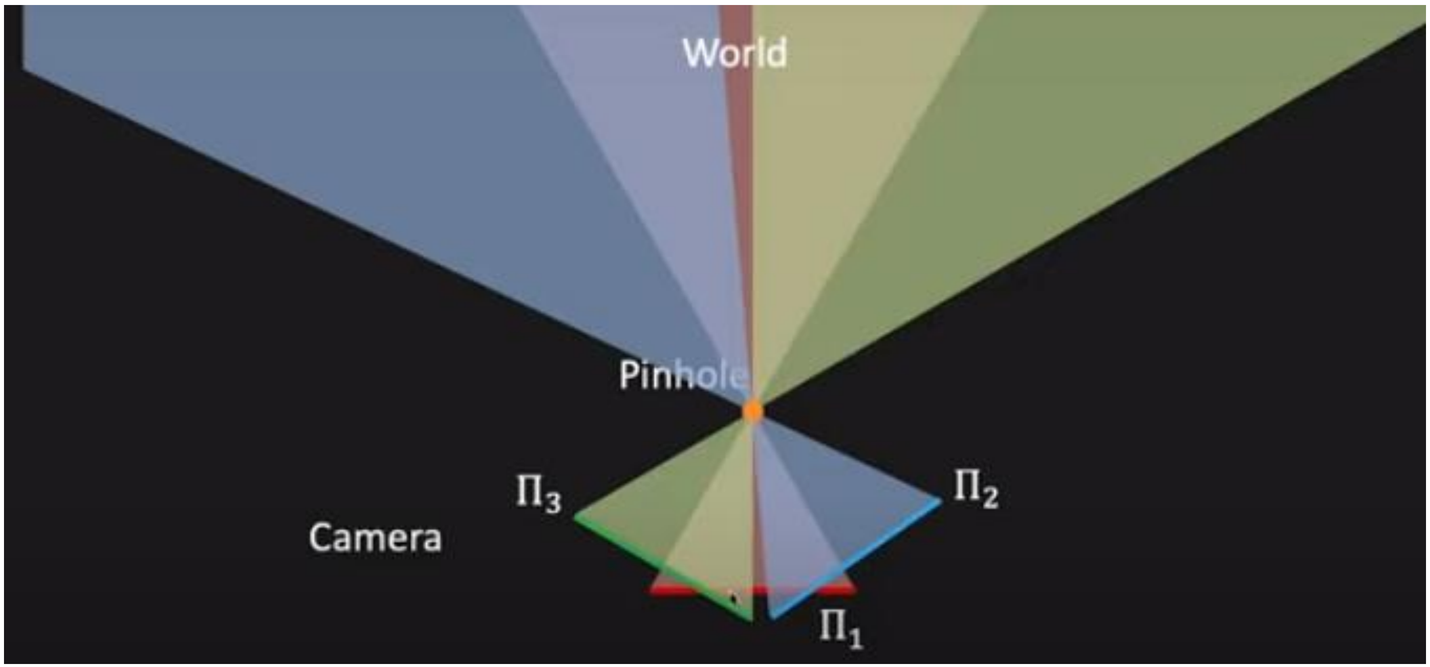
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography



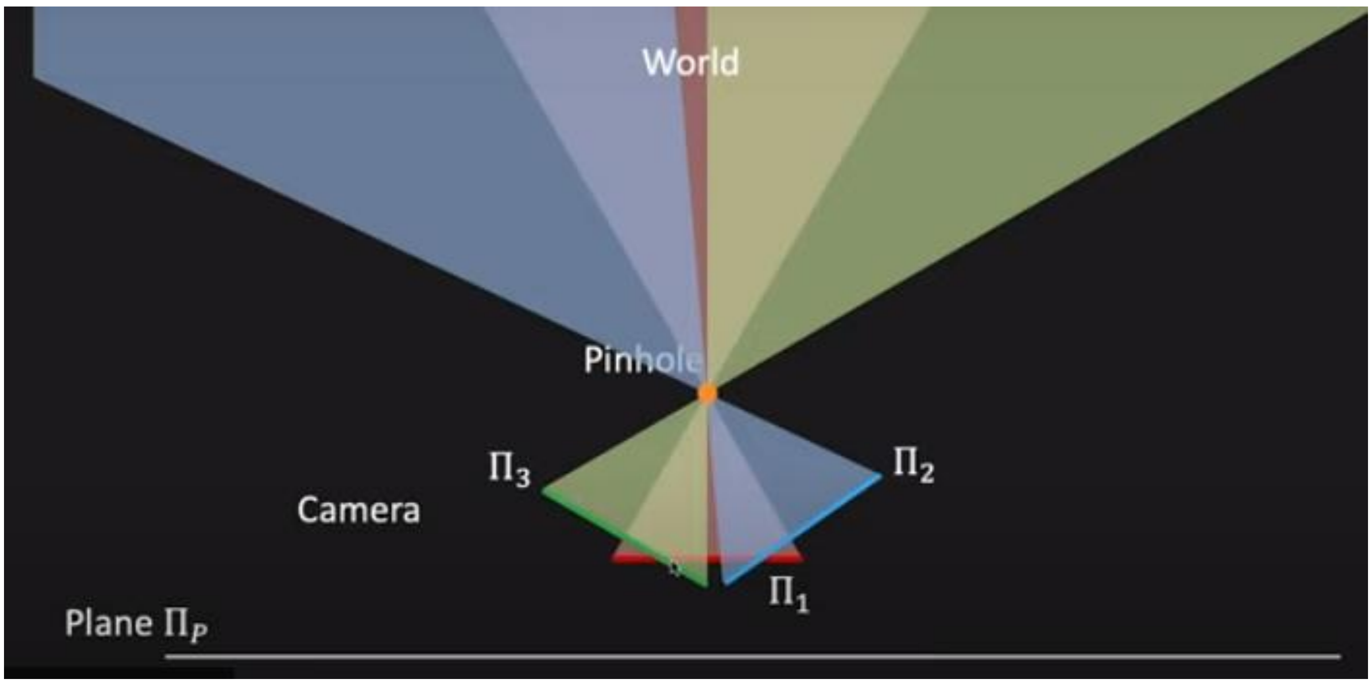
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography



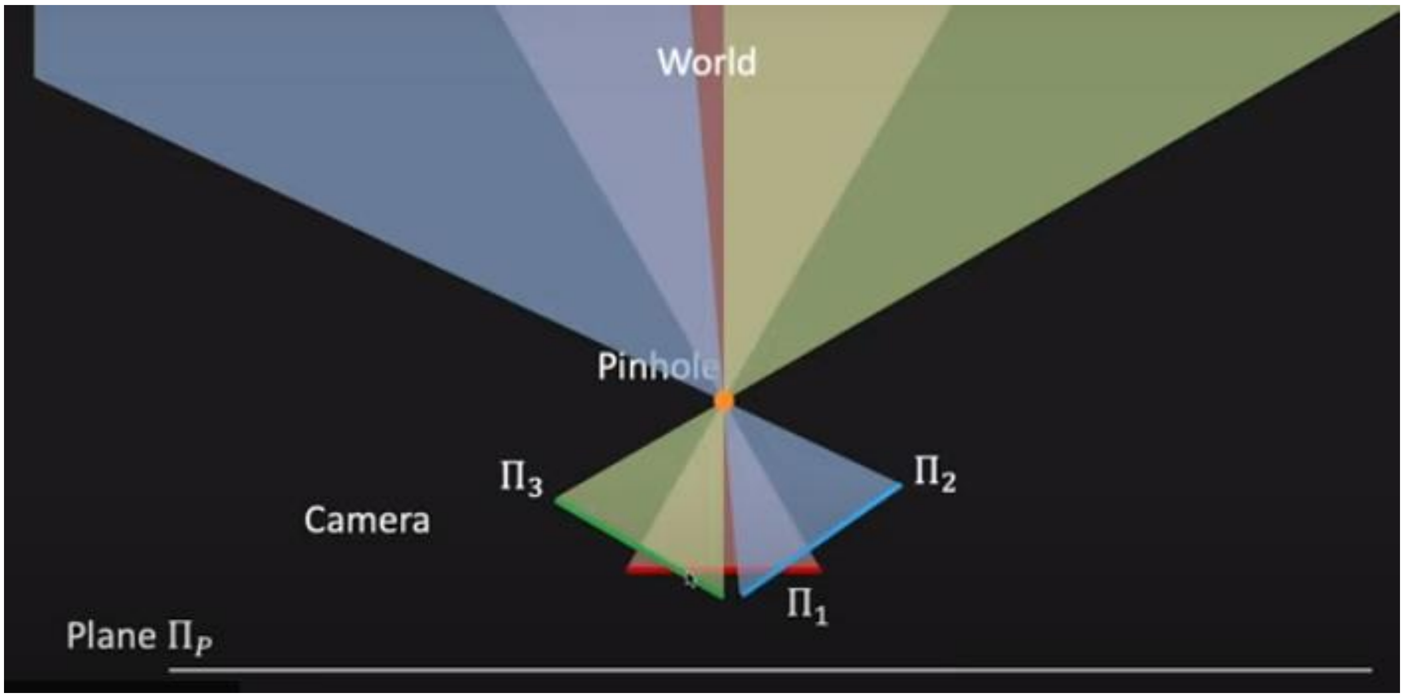
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography



Chapitre 1. SIFT Descriptor and Applications

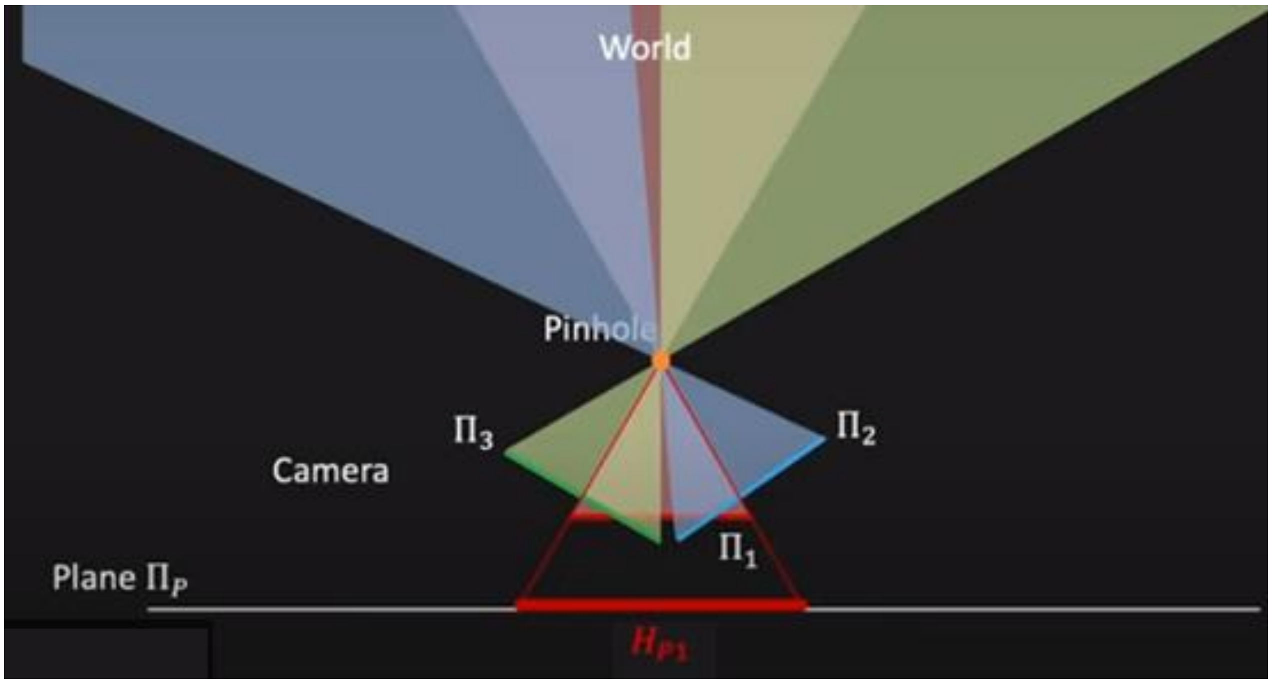
1.6 Image Stitching
Computing Homography



The plane Π_p share with Π_1 the same central projection. We can map Π_1 to Π_p using the Homography H_{1p}

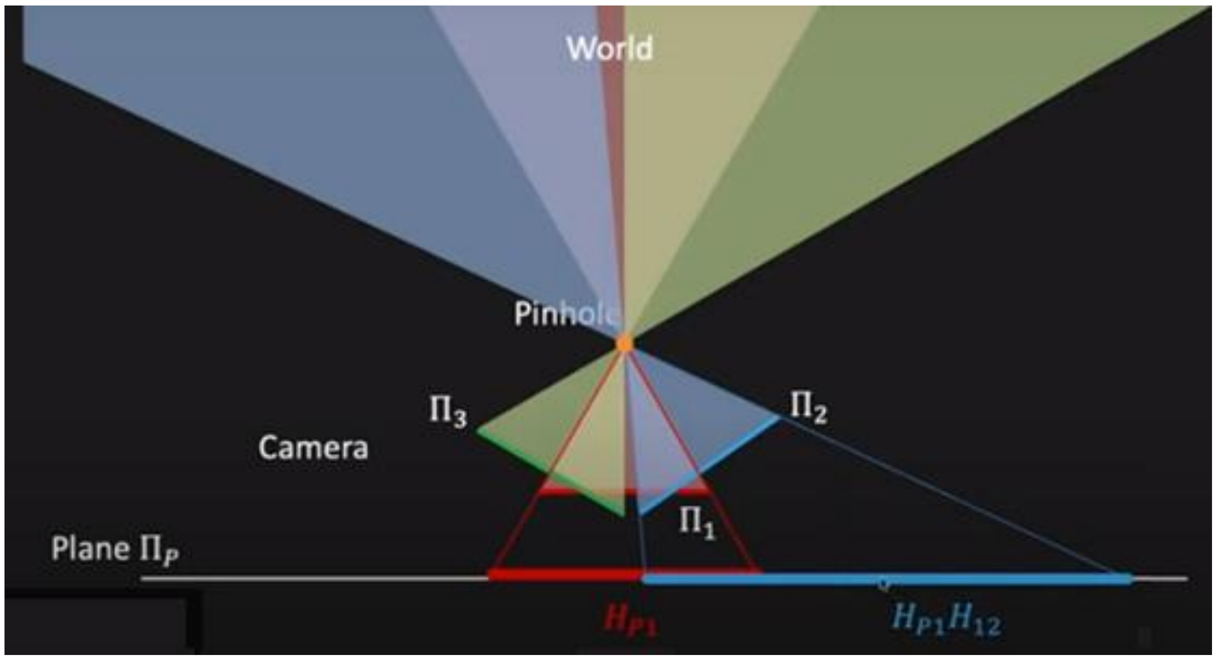
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography



The plane Π_p share with Π_1 the same central projection. We can map Π_1 to Π_p using the Homography H_{1p}

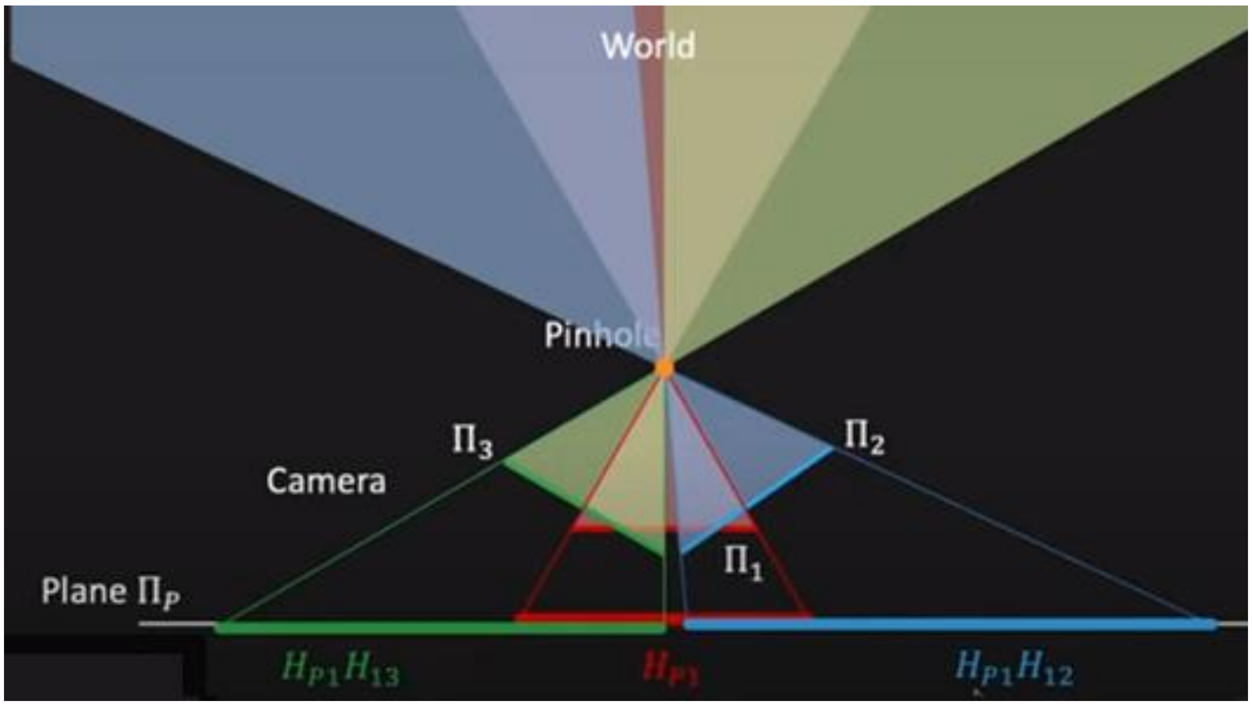
1.6 Image Stitching
Computing Homography



The plane Π_2 is mapped to Π_1 using the Homography H_{12}.

Chapitre 1. SIFT Descriptor and Applications

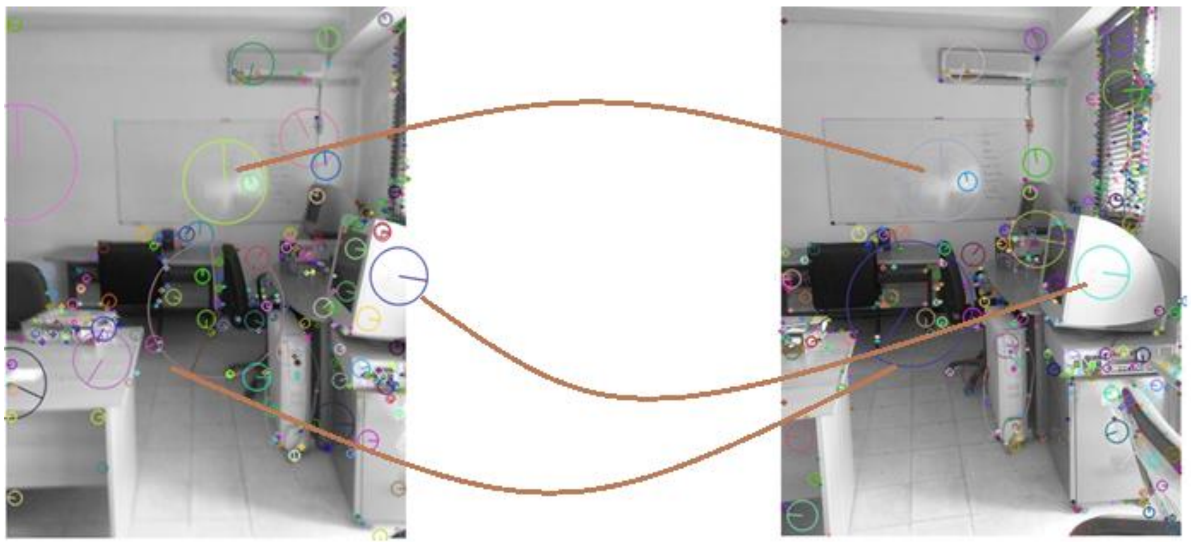
1.6 Image Stitching
Computing Homography



We can map all images to a single plane by simply computing the Homographies between the images. This is useful for stitching panoramas.

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography



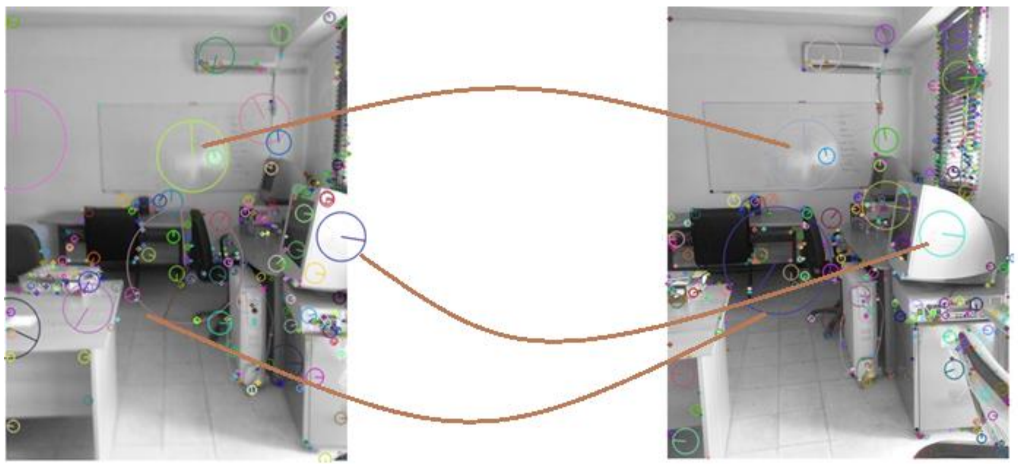
Find the Homography that best agrees with the matches (SIFT descriptors).

Hypothesis:

- the images are acquired from the same view point.
- Or the scene points should lie on a same plane.
- Or the scene is really far away (scene is a plane at infinity).

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography



Source Destination

$$\begin{bmatrix} x_d \\ y_d \\ 1 \end{bmatrix} \equiv \begin{bmatrix} \tilde{x}_d \\ \tilde{y}_d \\ \tilde{z}_d \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix}$$

9 unknowns, 8 degrees of freedom.
Then, at the minimum, 04 pairs of matching points

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography

For a given Source pair i of corresponding points:

$$\begin{bmatrix} x_d \\ y_d \\ 1 \end{bmatrix} \equiv \begin{bmatrix} \tilde{x}_d \\ \tilde{y}_d \\ \tilde{z}_d \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix}$$

$$x_d = \frac{\tilde{x}_d}{\tilde{z}_d} = \frac{h_{11}\ x_s + h_{12}\ y_s + h_{13}}{h_{31}\ x_s + h_{32}\ y_s + h_{33}} = \qquad y_d = \frac{\tilde{y}_d}{\tilde{z}_d} = \frac{h_{21}\ x_s + h_{22}\ y_s + h_{23}}{h_{31}\ x_s + h_{32}\ y_s + h_{33}} =$$

$\Rightarrow$

$$x_d(h_{31}\ x_s + h_{32}\ y_s + h_{33}) = h_{11}\ x_s + h_{12}\ y_s + h_{13}$$

$$y_d(h_{31}\ x_s + h_{32}\ y_s + h_{33}) = h_{21}\ x_s + h_{22}\ y_s + h_{23}$$

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography

For a given Source pair i of corresponding points, we obtain a linear equation:

$$x_d(h_{31}\ x_s + h_{32}\ y_s +\ h_{33}) = h_{11}\ x_s + h_{12}\ y_s +\ h_{13}$$

$$y_d(h_{31}\ x_s + h_{32}\ y_s +\ h_{33}) = h_{21}\ x_s + h_{22}\ y_s +\ h_{23}$$

$$\Rightarrow \begin{bmatrix} x_s & y_s & 1 & 0 & 0 & 0 & -x_d x_s & -x_d\ y_s & -x_d \\ 0 & 0 & 0 & x_s & y_s & 1 & -y_d x_s & -y_d\ y_s & -y_d \\ \multicolumn{9}{c}{\dots.second\ pair\ of\ matched\ points} \\ \multicolumn{9}{c}{----\ -} \\ \multicolumn{9}{c}{Third\ pair\ of\ mathed\ points} \\ \multicolumn{9}{c}{---\ -} \\ \multicolumn{9}{c}{\dots.} \\ \multicolumn{9}{c}{\dots.} \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ . \\ . \\ 0 \end{bmatrix}$$

A: Known

h: Unknown

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography

Solve for h:
$Ah = 0$, such that $\|h\|^2 = 1$

We define least squares problem:
$\min_h \|Ah\|^2$, such that $\|h\|^2 = 1$

We know:
$\|Ah\|^2 = (Ah)^T A\mathrm{h} = h^T A^T Ah$, such that $\|h\|^2 = h^T h = 1$

Solve for h:
$\min_h \|h^T A^T Ah\|^2$, such that $h^T h = 1$.

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Computing Homography

We define the loss function $\mathrm{L}(h, \lambda) = h^T A^T A h - \lambda(h^T h - 1)$

We derive L relatively to h :

$$2A^T A h - 2\lambda h = 0$$

$$A^T A h - \lambda h = 0$$ Eigen Values problem

Solution: We choose the eigen vector h with smalest value of λ of $A^T A$ which minimize the loss function L

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Dealing with Outliers: RANSAC

Problem of Outliers:
We need to robustly compute transformation in presence of wrong matches.

If the number of outliers is < 50%, then **RANSAC** (**RAN**dom **SA**mple **C**onsensus) to the rescue.

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Dealing with Outliers: RANSAC

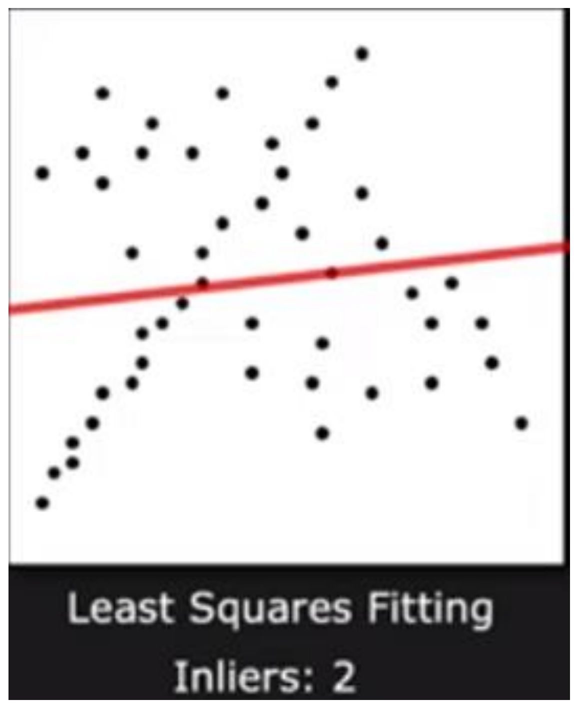
General RANSAC Algorithm

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 1..3 N times
5- Choose a model that has the largest number M of inliers.

For Homography, N=4, ε: acceptable alignment error in pixels.

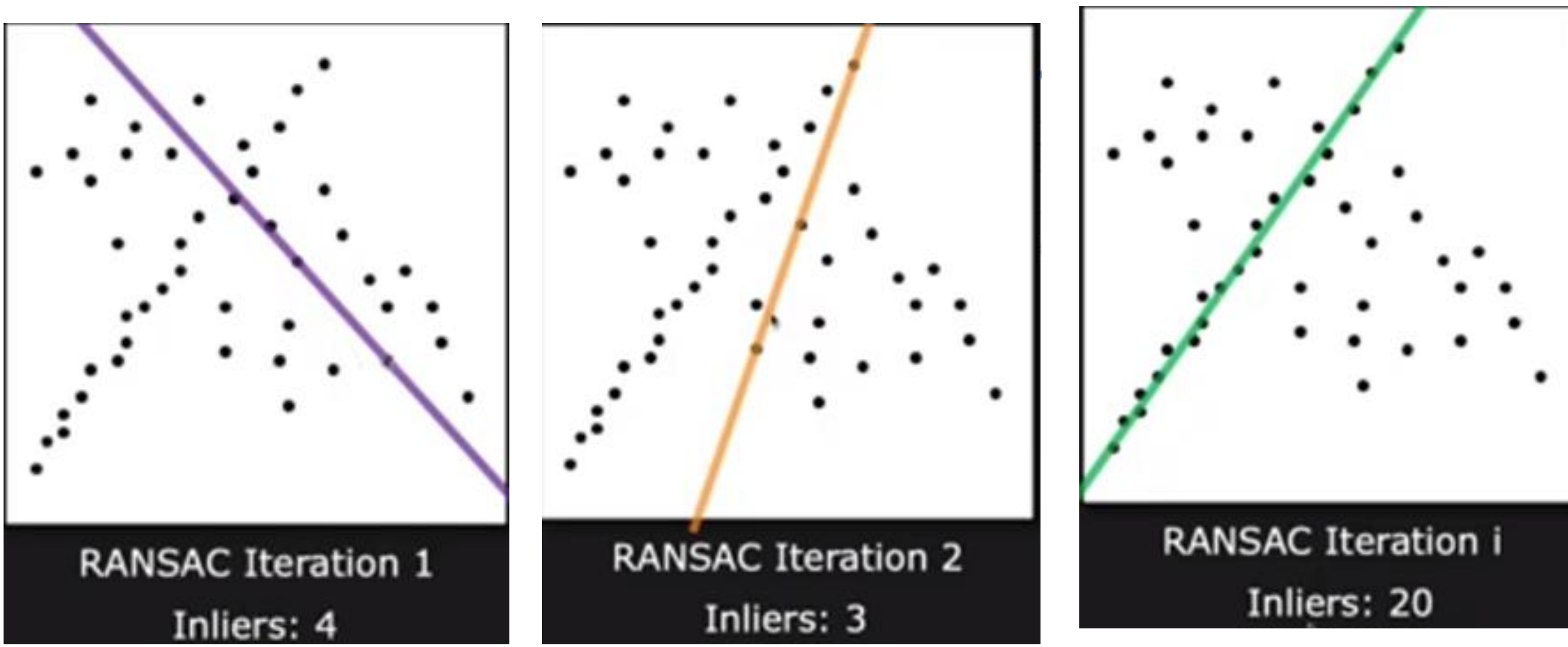
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Dealing with Outliers: RANSAC
Example: Robust line fitting



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Dealing with Outliers: RANSAC
Example: Robust line fitting



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping Images

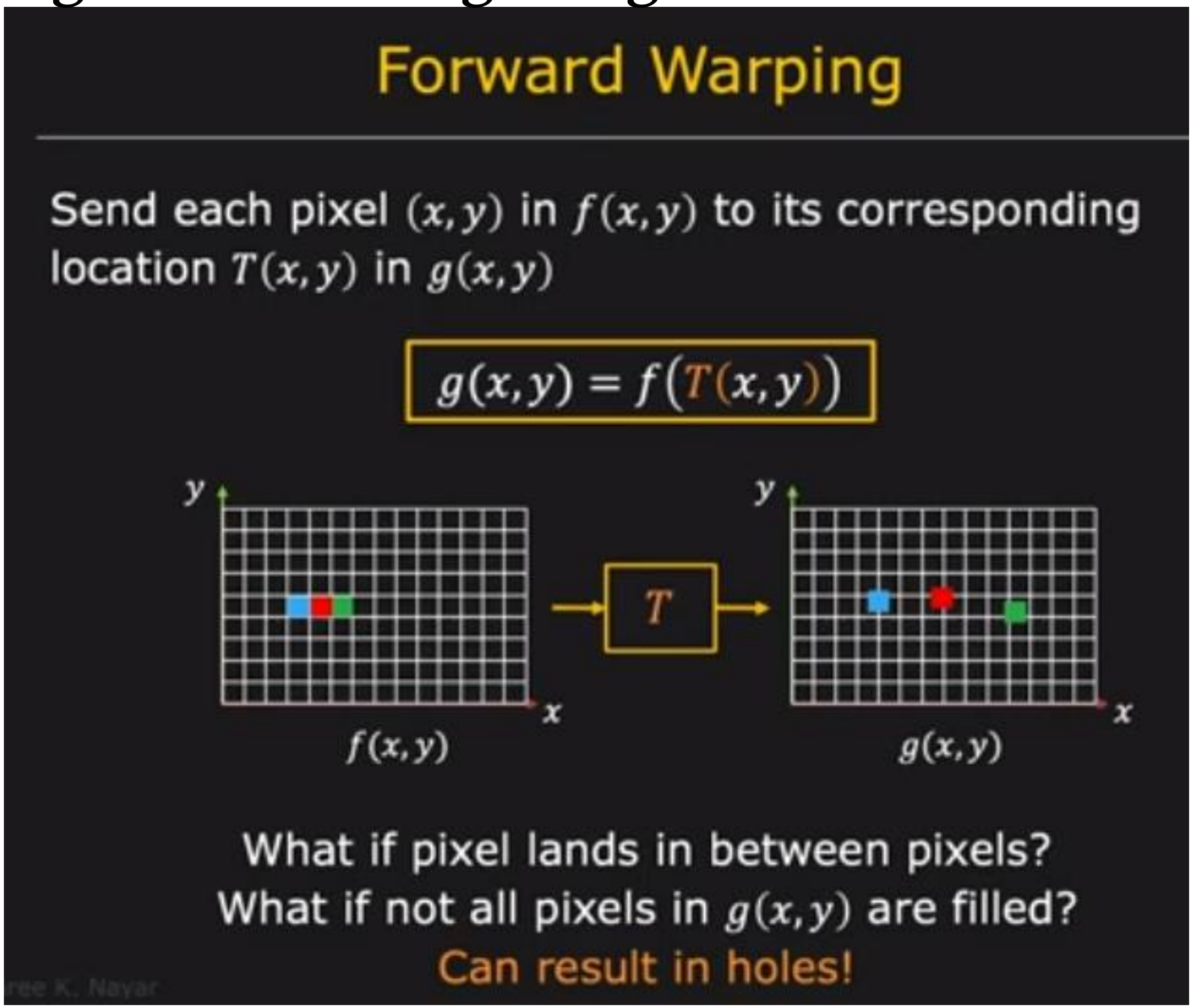
Given a transformation T, and an image f(x,y), we compute the transformed image g(x,y)

f(x,y)

g(x,y)=T(f(x,y))

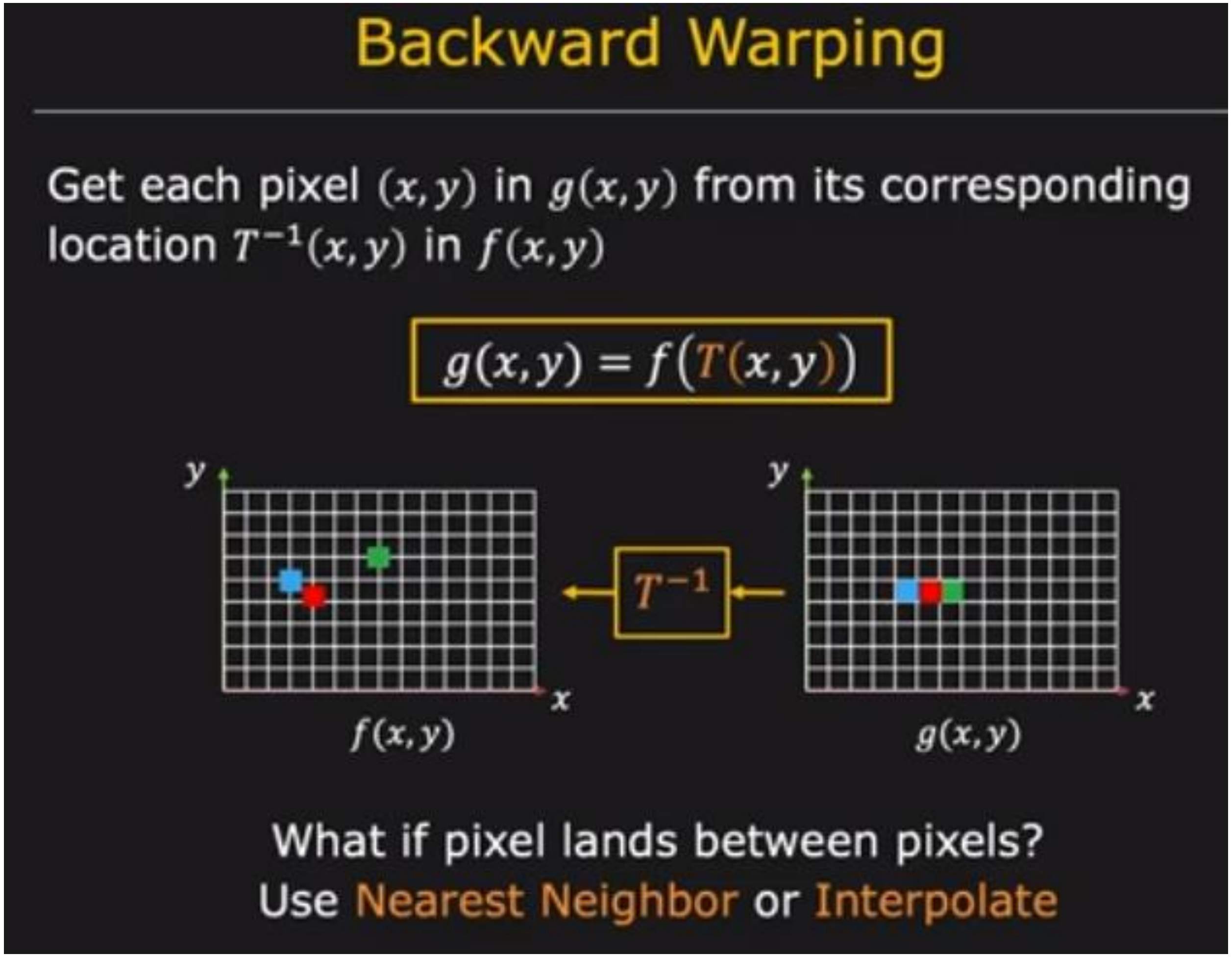
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping and Blending Images



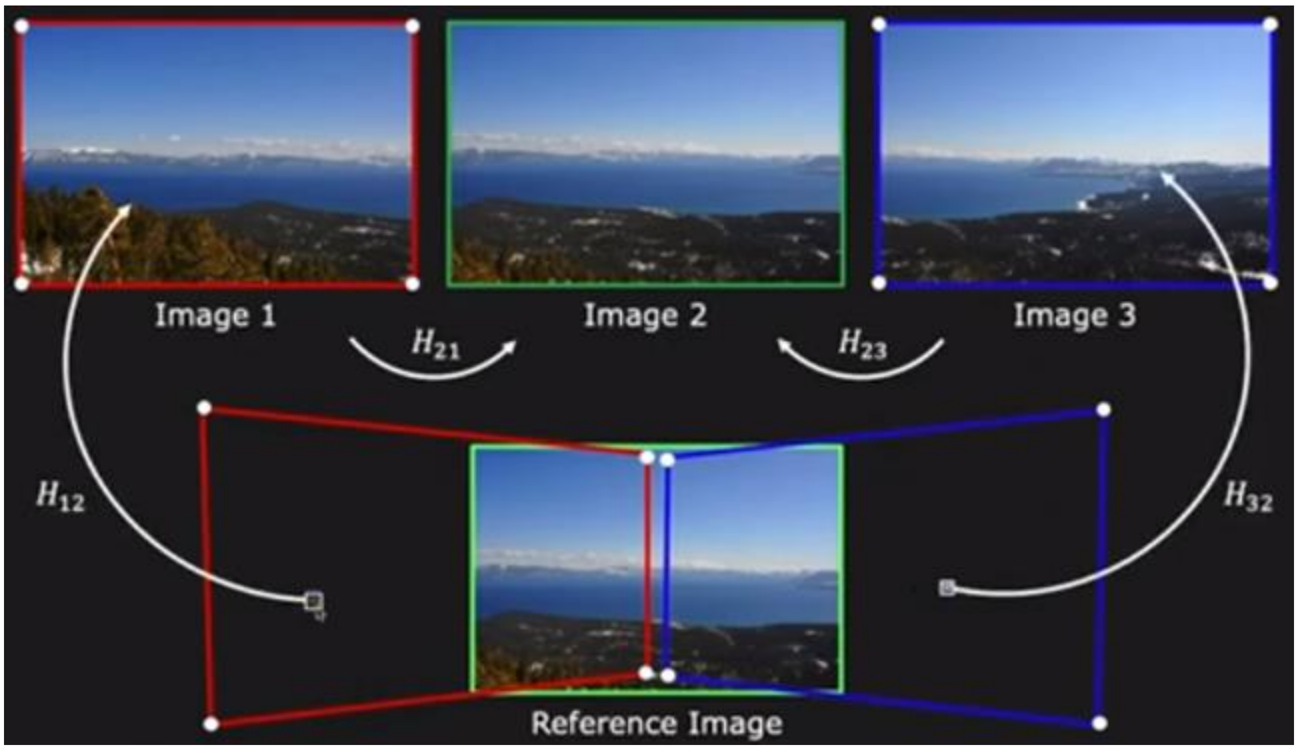
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping and Blending Images



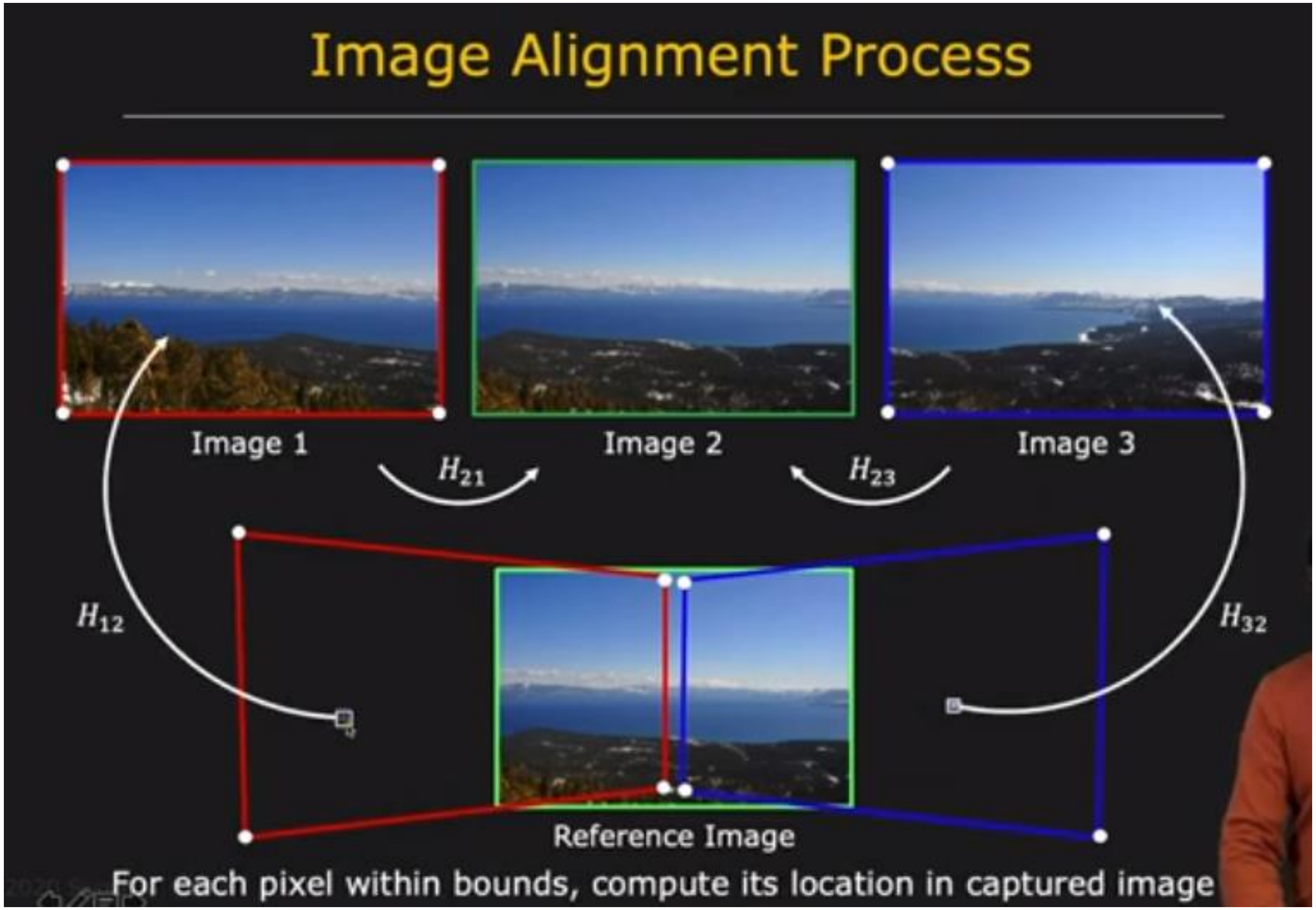
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping and Blending Images



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping and Blending Images



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping and Blending Images

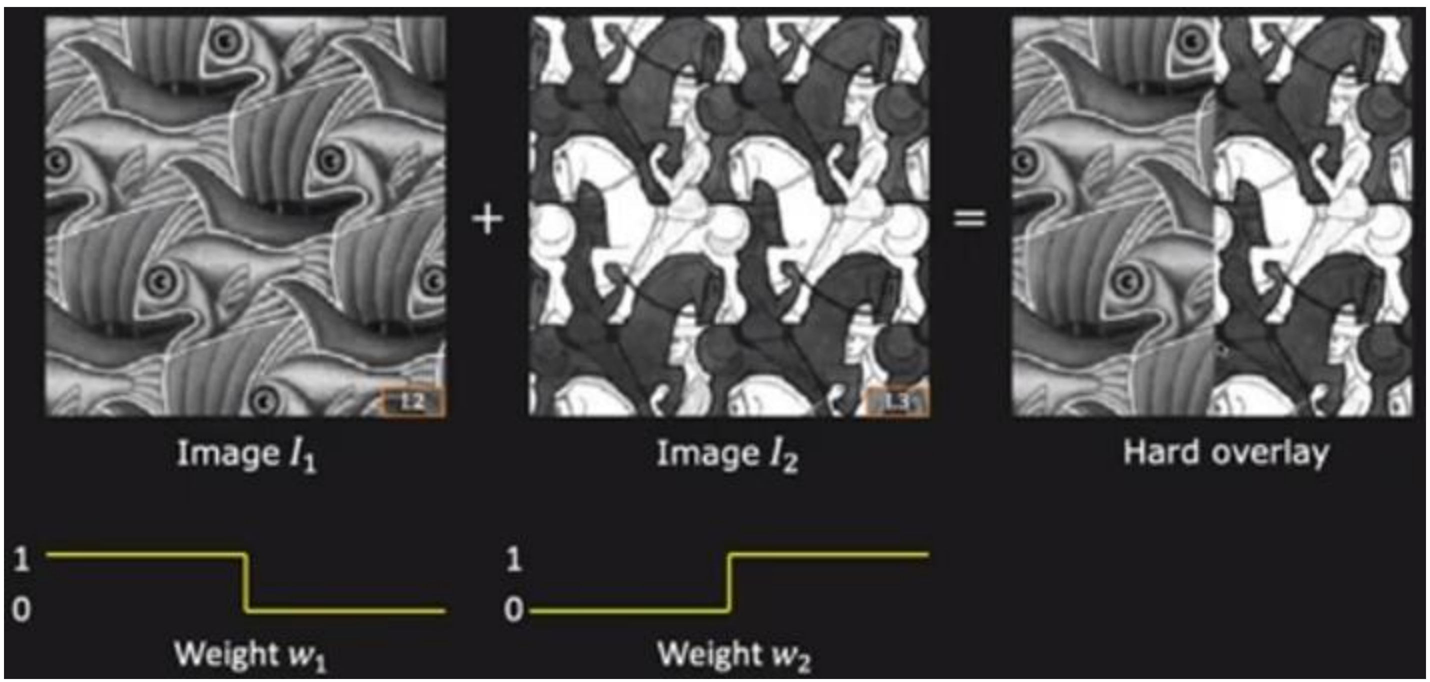
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping and Blending Images

Chapitre 1. SIFT Descriptor and Applications

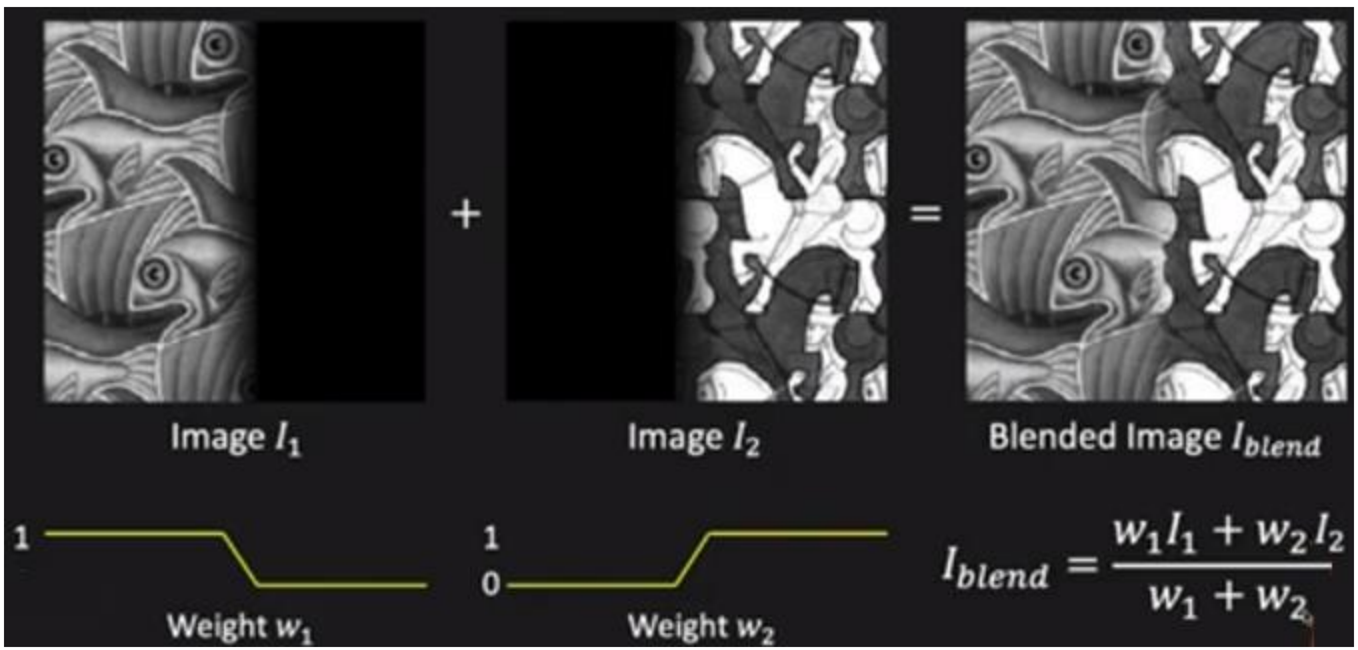
1.6 Image Stitching
Warping and Blending Images

We want to blend images I1 an I2 at the center.



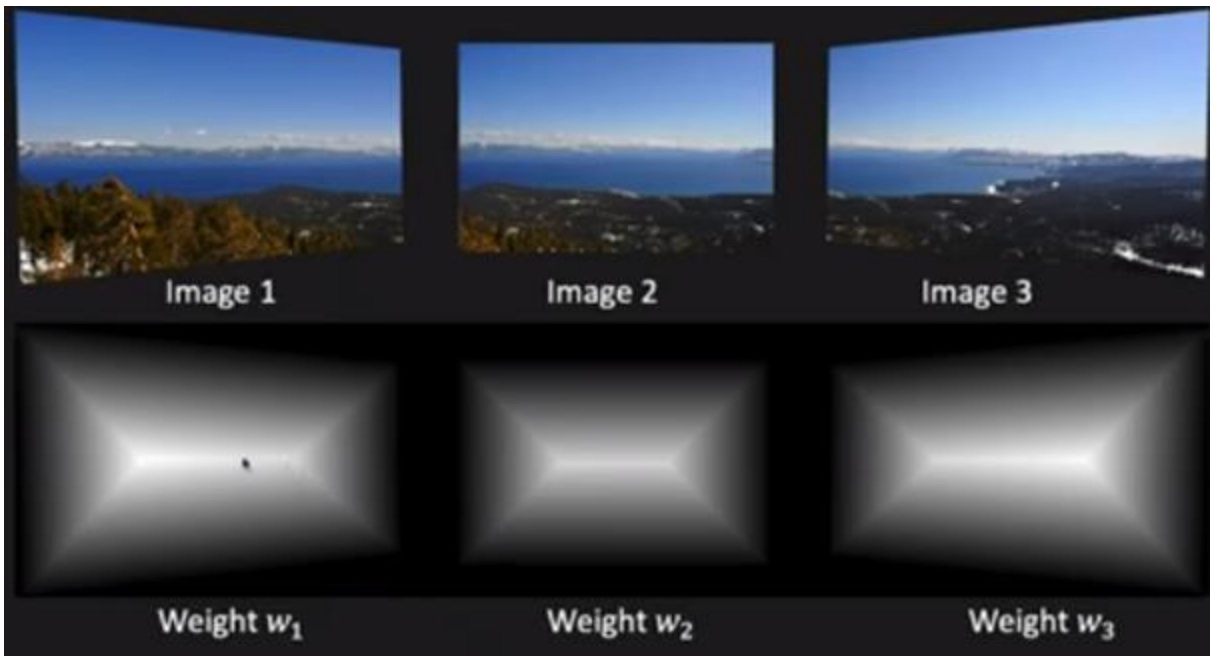
Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping and Blending Images



Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping and Blending Images



Pixels closer to the edges get a lower weight.

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping and Warping and Blending Images

Weighted blending

Chapitre 1. SIFT Descriptor and Applications

1.6 Image Stitching
Warping and Blending Images