Re-HeEd

ΔΗΜΟΚΡΙΤΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΡΑΚΗΣ | DEMOCRITUS UNIVERSITY OF THRACE

Co-funded by the
Erasmus+ Programme
of the European Union

Erasmus +, Reframing Heritage Education in Egypt (ReHeEd)

Theoretical and hands-on Workshop course (WP2.3) & Training (WP3.2) on:
“The implementation of ICT documentation techniques for
Heritage Educational Purposes”

“Digital Photogrammetry Background – Structure From Motion and Multi-view Stereo”

Alexandria
28 May -2 April 2022

Table of Contents.

I. Problem: From images to 3D structures

II. Stages of 3D reconstruction

III. Digital Images

IV. Structure from motion 1. SIFT 2. RANSAC 3. Bundle Adjustment

V. Multi-view Stereo

VI. Surface Reconstruction from Points

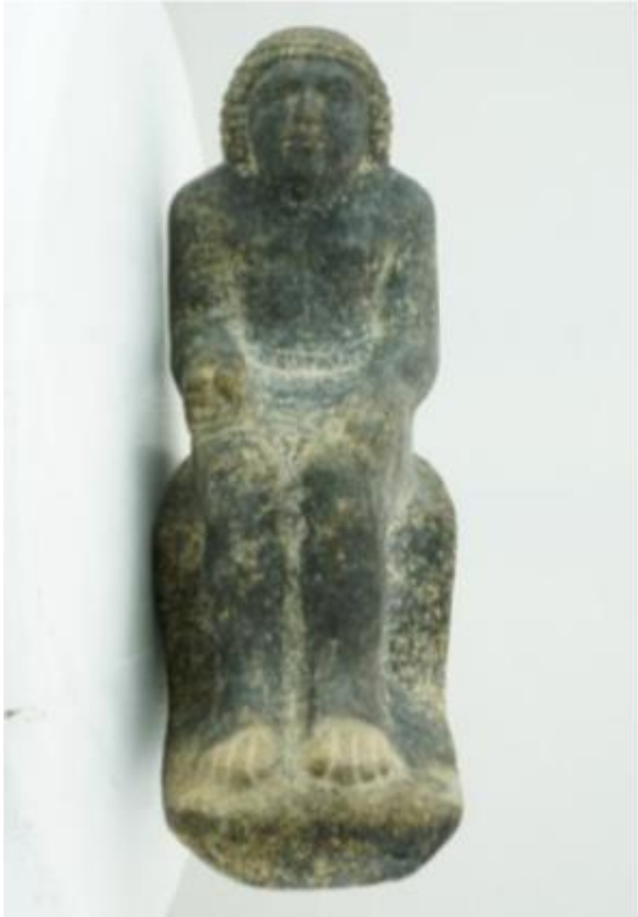
Problem:

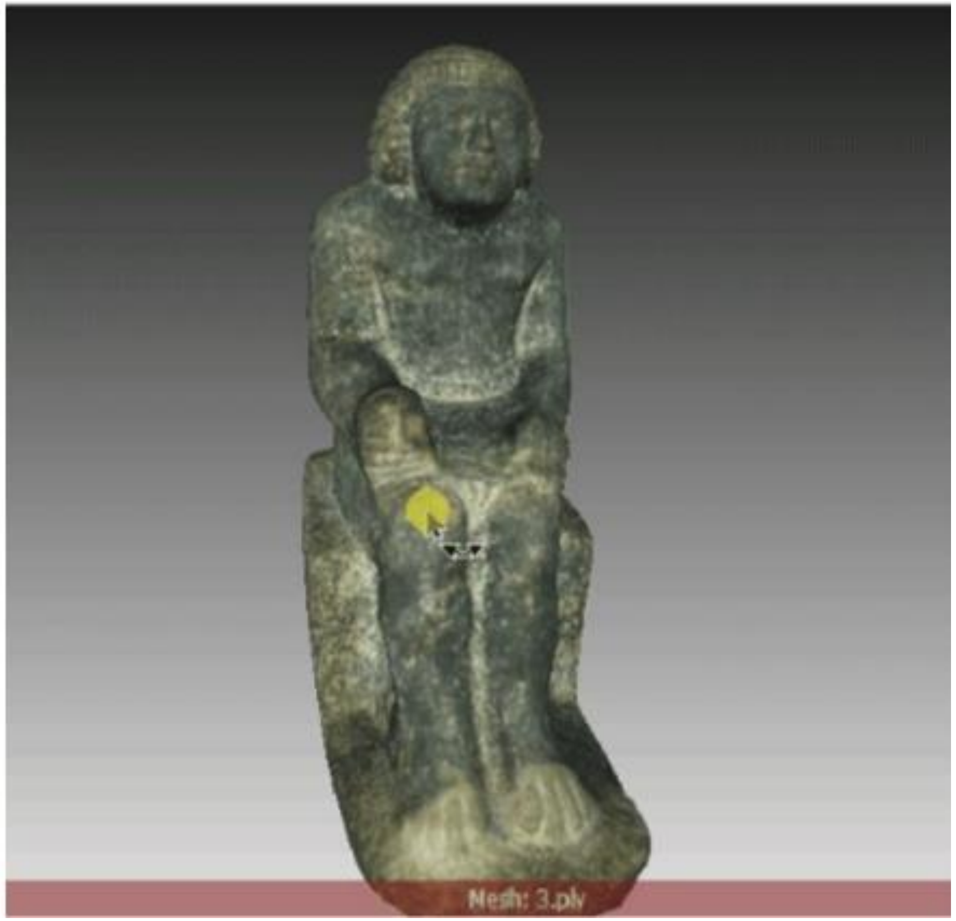
> “Imaging technique where given an optical flow or point correspondences, compute a 3D motion (in terms of translation and rotation) and space (depth)”.
>
> Dr. Mubarak Shah

Goal:

Recovery of 3D (shape) from one or two (2D images).

From Images to 3D structures.





Methods to achieve 3D recovery:

- Stereo
- Motion
- Shading
- Photometric Stereo
- Texture
- Contours
- Silhouettes
- Defocus

Applications of 3D recovery:

- Object Recognition
- Robotics
- Computer Graphics
- Image Retrieval
- Geo-localization
- Archaeology
- Sports

Structure from Motion has been under discussion for the last 20 years…

There is a long list of researchers such as…

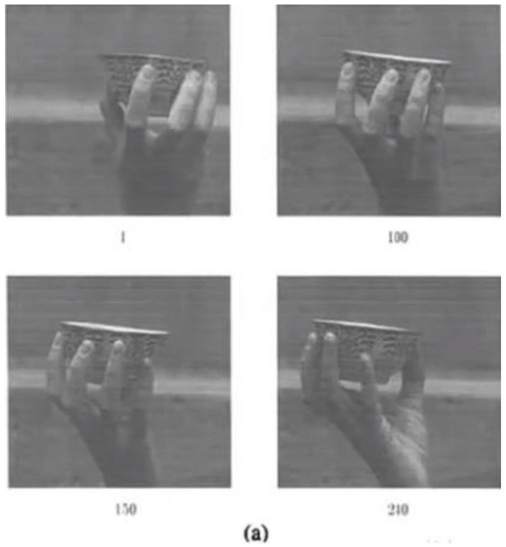
- S. Ullman
- Hanson & Riseman
- Webb & Aggarwal
- T.Huang
- Heeger & Jepson
- Chellappa
- Faugeras
- Zisserman
- Kanade
- Pentland
- Van Gool
- Pollefeys
- Seitz & Szeliski
- Shahsua
- Irani
- Vidal & Yi Ma
- Medioni
- Fleet
- Tian & Shah

Tomasi and Kanade Factorization: Orthographic Projection.

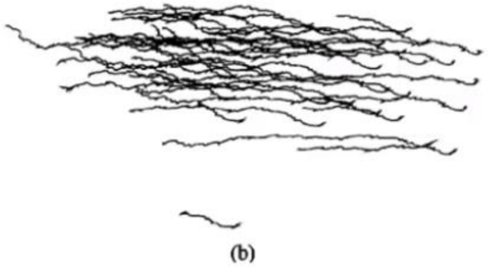
Assumptions:

- The camera model is orthographic.
- The positions of “P” points in “F” frames (F>=3), which are not all coplanar and have been tracked.
- The entire sequence has been acquired before starting (batch mode).
- Camera calibration not needed, if we accept 3D points up to a scale factor.

Input:



a) Images



b) KLT Tracks

Feature Points.

Image points
(This is not optical flow

$$\left\{(u_{fp}, v_{fp}) \mid f = 1, \ldots, F, p = 1, \ldots, P\right\}$$

$$W = \begin{bmatrix} u_{11} \ldots u_{1p} \\ \vdots \\ u_{F1} \ldots u_{FP} \\ v_{11} \ldots v_{1P} \\ \vdots \\ v_{F1} \ldots v_{FP} \end{bmatrix} \qquad W = \begin{bmatrix} U \\ - \\ V \end{bmatrix}$$

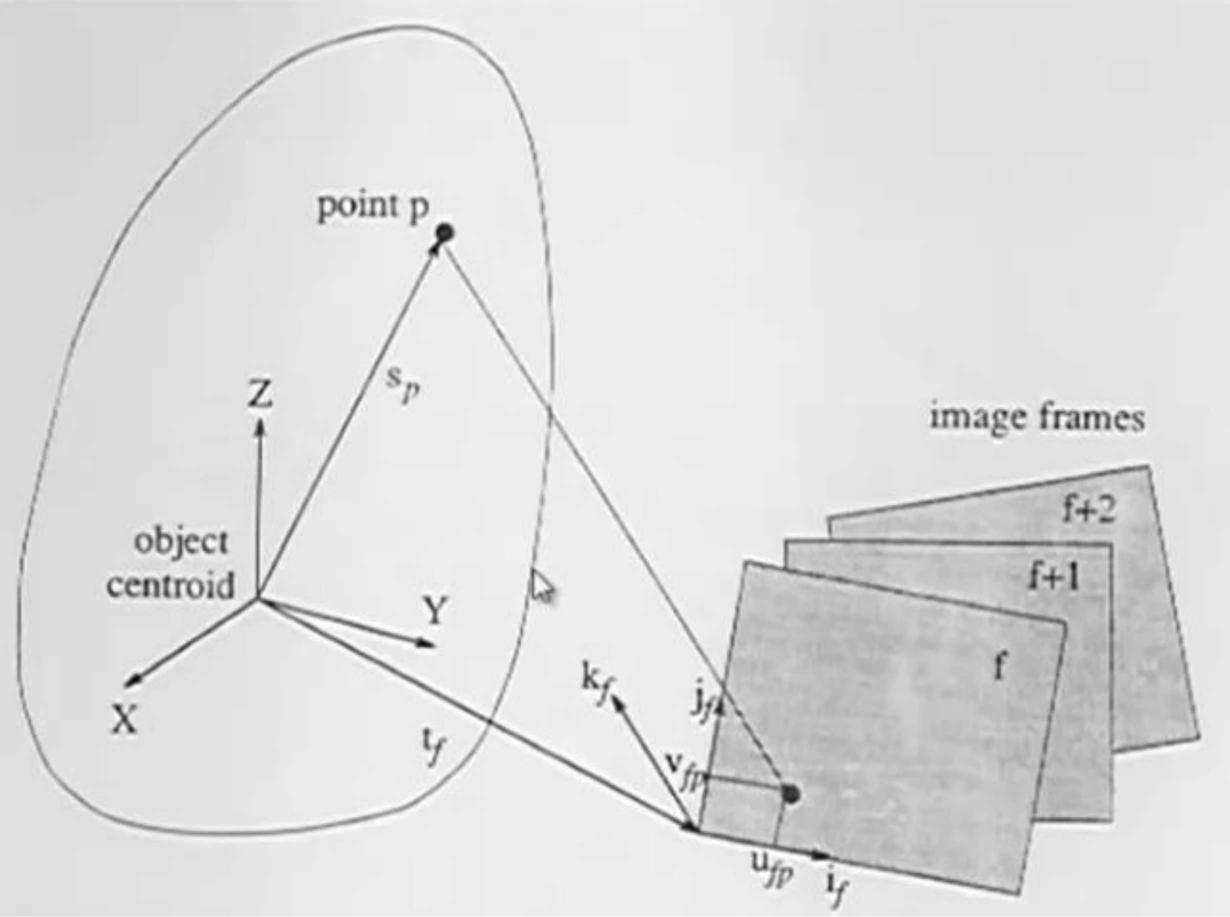
Mean Normalize Feature Points.

$$a_f = \frac{1}{P}\sum_{p=1}^{P} u_p \qquad\qquad b_f = \frac{1}{P}\sum_{p=1}^{P} v_p$$

$$\widetilde{u}_{fP} = u_{fP} - a_{fP} \qquad\qquad (A)$$

$$\widetilde{v}_{fP} = v_{fP} - b_{fP}$$

Orthographic Projection.



$$s_p = (X_p, Y_P, Z_P)$$

3D world point

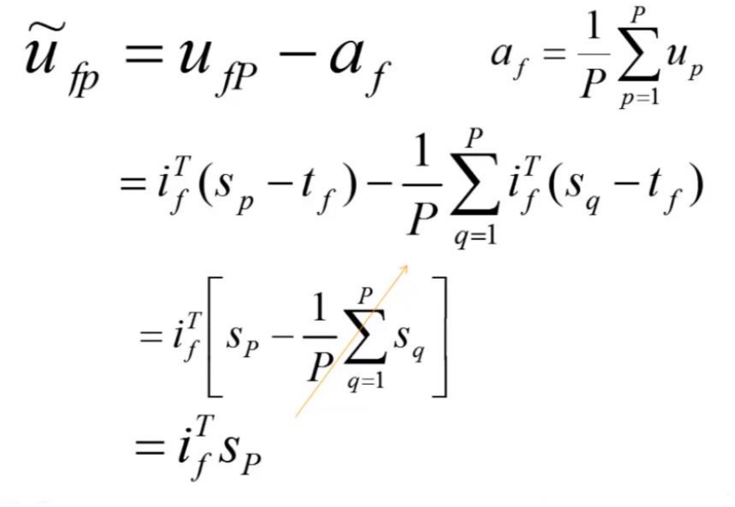
$$u_{fP} = i_f^T (s_P - t_f)$$

$$v_{fP} = j_f^T (s_P - t_f)$$

Orthographic projection

$$k_f = i_f \times j_f$$

i, j, k are unit vectors along *X, Y, Z*



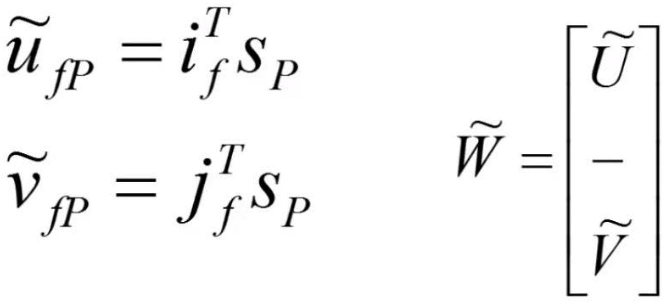
$$\widetilde{u}_{fp} = u_{fP} - a_f \qquad a_f = \frac{1}{P}\sum_{p=1}^{P} u_p$$

$$= i_f^T(s_p - t_f) - \frac{1}{P}\sum_{q=1}^{P} i_f^T(s_q - t_f)$$

$$= i_f^T\left[s_P - \cancel{\frac{1}{P}\sum_{q=1}^{P} s_q}\right]$$

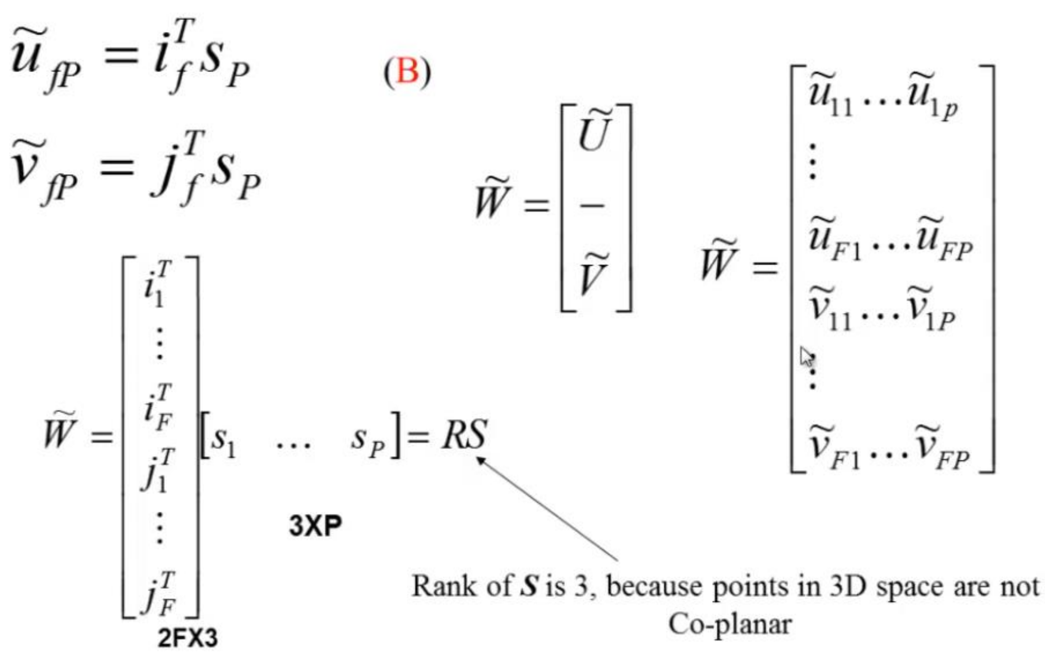
$$= i_f^T s_P$$

If Origin of world is at the centroid of object points, Second term is zero.



$$\widetilde{u}_{fP} = i_f^T s_P \qquad \widetilde{W} = \begin{bmatrix} \widetilde{U} \\ - \\ \widetilde{V} \end{bmatrix}$$

$$\widetilde{v}_{fP} = j_f^T s_P$$



$$\widetilde{u}_{fP} = i_f^T s_P$$

$$\widetilde{v}_{fP} = j_f^T s_P$$

(B)

$$\widetilde{W} = \begin{bmatrix} \widetilde{U} \\ - \\ \widetilde{V} \end{bmatrix} \qquad \widetilde{W} = \begin{bmatrix} \widetilde{u}_{11} \ldots \widetilde{u}_{1p} \\ \vdots \\ \widetilde{u}_{F1} \ldots \widetilde{u}_{FP} \\ \widetilde{v}_{11} \ldots \widetilde{v}_{1P} \\ \vdots \\ \widetilde{v}_{F1} \ldots \widetilde{v}_{FP} \end{bmatrix}$$

$$\widetilde{W} = \underbrace{\begin{bmatrix} i_1^T \\ \vdots \\ i_F^T \\ j_1^T \\ \vdots \\ j_F^T \end{bmatrix}}_{2FX3} \underbrace{\begin{bmatrix} s_1 & \ldots & s_P \end{bmatrix}}_{3XP} = RS$$

Rank of $\boldsymbol{S}$ is 3, because points in 3D space are not Co-planar

Rank Theorem.

Without noise, the registered measurement matrix W, is at most of rank three.

$$\widetilde{W} = \begin{bmatrix} i_1^T \\ \vdots \\ i_F^T \\ j_1^T \\ \vdots \\ j_F^T \end{bmatrix} \begin{bmatrix} s_1 & \cdots & s_P \end{bmatrix} = RS$$

Because W is a product of two matrices,
The maximum rank of S is 3.

Linearly Independence.

A finite subset of *n* vectors, v_1, v_2, …, v_n, from the vector space *V*, is **linearly independent** if and only if there exits a set of *n* scalars, a_1, a_2, …, an, not all zero such that

$$a_1v_1 + a_2v_2 + \ldots a_nv_n = 0$$

Rank of a Matrix.

- The **column rank** of a matrix A is the maximum number of linearly independent column vectors of A.
- The **row rank** of a matrix A is the maximum number of linearly independent row vectors of A.
- The column rank of A is the dimension of the column space of A.
- The row rank of A is the dimension of the row space of A.

How to find translation.

$$\tilde{u}_{fp} = u_{fP} - a_f$$ From (A)

$$u_{fp} = \tilde{u}_{fP} + a_f \qquad \tilde{u}_{fp} = i_f^T s_P$$

From (B)

$$u_{fp} = i_f s_p + a_f \text{ (E)} \qquad u_{fp} = i_f^T (s_p - t_f)$$

Comparing above two eqs

From (C)

$$\boxed{a_f = -t_f i_f^T}$$ (D)

a_f is projection of camera translation along x-axis

$$u_{fp} = i_f s_p + a_f \qquad v_{fp} = j_f s_p + b_f$$

$$\underset{2FXP}{\mathbf{W}} = \underset{2FX3}{\mathbf{R}}\underset{3XP}{\mathbf{S}} + \underset{2FX1}{\mathbf{t}}\underset{1XP}{\mathbf{e}_\mathbf{p}^\mathbf{T}} \qquad a_f = -t_f i_f^T$$

From (D)

$$\mathbf{t} = (a_1, \ldots, a_f, b_1, \ldots, b_f)^T$$

$$\mathbf{e}_\mathbf{p}^\mathbf{T} = (1, \ldots 1)$$

Projected camera translation can be computed:

$$-i_f^T t_f = a_f = \frac{1}{P}\sum_{p=1}^{P} u_p$$

$$-j_f^T t_f = b_f = \frac{1}{P}\sum_{p=1}^{P} v_p$$

Noisy Measurements.

- Without noise, the matrix W must be at most rank 3. When noise corrupts the images, however, W will not be of rank 3. Rank theorem can be extended to the case of noisy measurements.

Singular Valued Decomposition SVD.

$$\underset{\text{2FXP}}{\widetilde{W}} = \underset{\textbf{2FXP}}{O_1}\underset{\textbf{PXP}}{\Sigma}\underset{\textbf{PXP}}{O_2}$$

Theorem: Any m by n matrix A, for which m>=n, can be written as:

$$\underset{\text{mxn}}{A} = \underset{\text{mxn}}{O_1}\underset{\text{nxn}}{\Sigma}\underset{\text{nxn}}{O_2}$$

Σ is diagonal

O_1 , O_2 are orthogonal

$O_1^T O_1 = O_2^T\ O_2 = I$

Steps for 3D Reconstruction:

1. Images to Points: Structure from Motion

2. Points to More Points: Multiple View Stereo

3. Points to Meshes: Model Fitting

4. Meshes to Models: Texture Mapping

SFM Pipeline.

1. Find the 2D Features (Keypoints) →

   Feature Descriptors:
   - SIFT
   - SURF

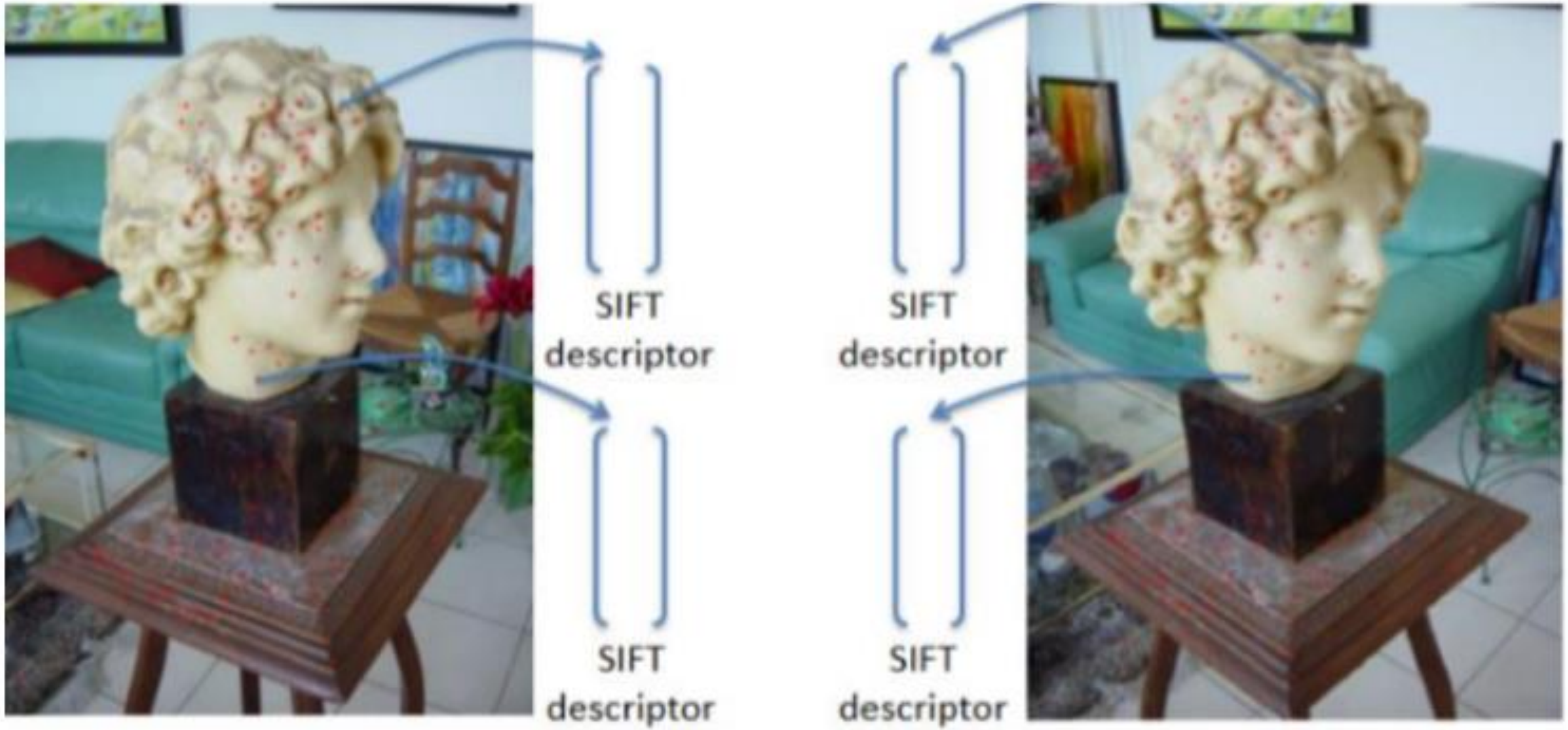
2. Re-match keypoint →

   Feature Matching Algorithms:
   - RANSAC
   - Hough Transform

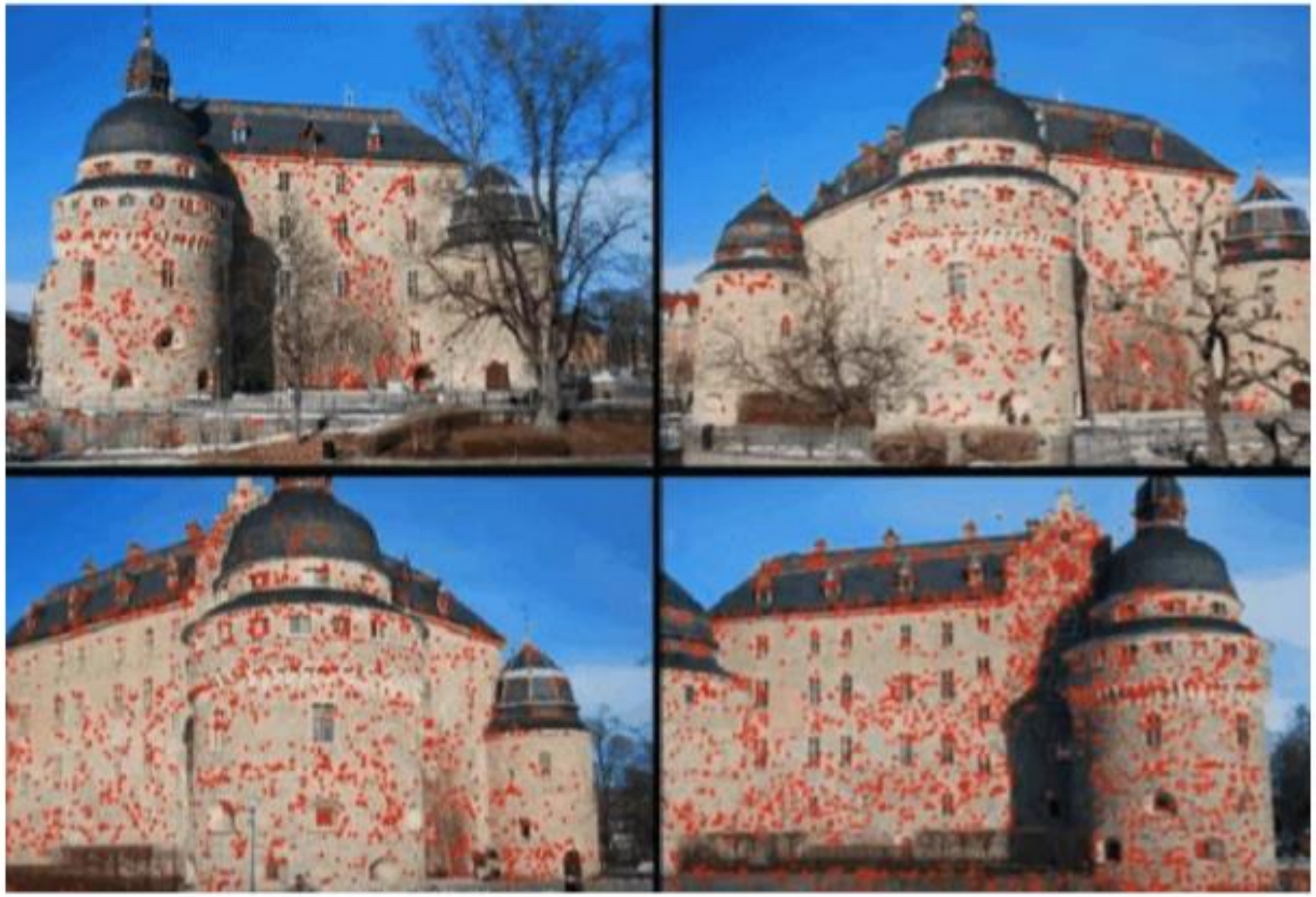
3. Optimize the positions of 3D points →

   - Bundle Adjustment

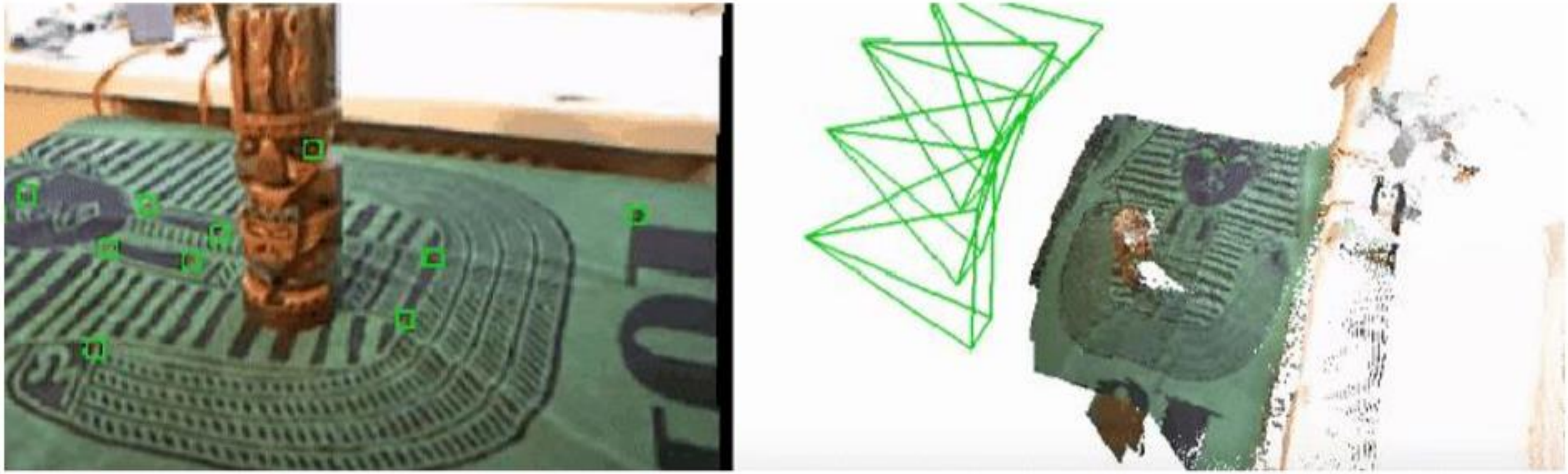
SIFT descriptors.



RANSAC re-matching.



Bundle Adjustment.

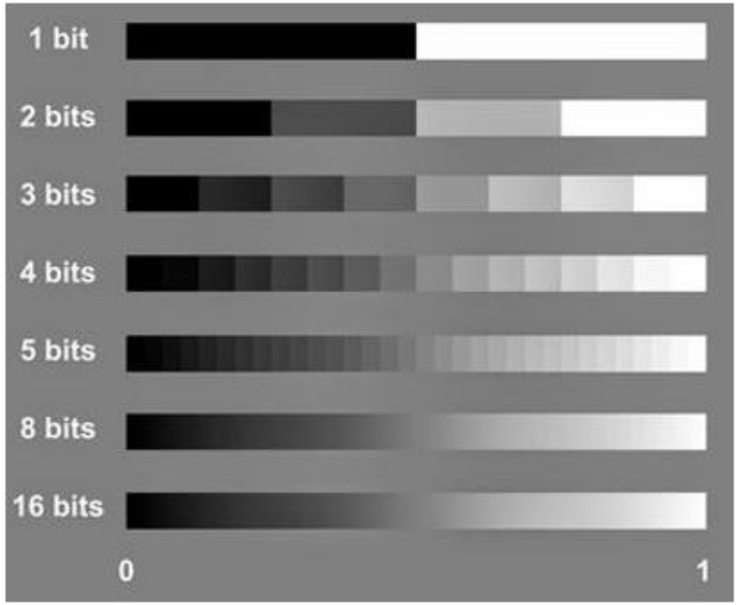


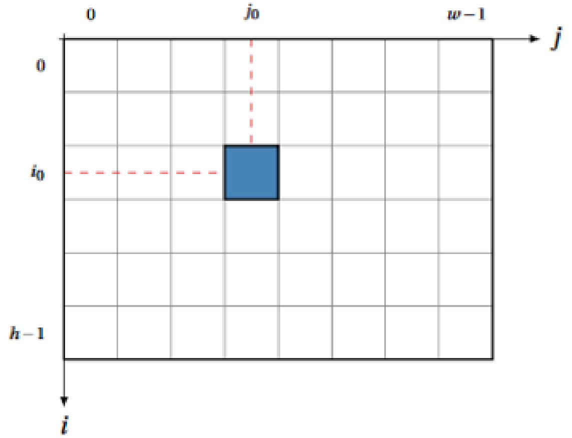
Digital Images.

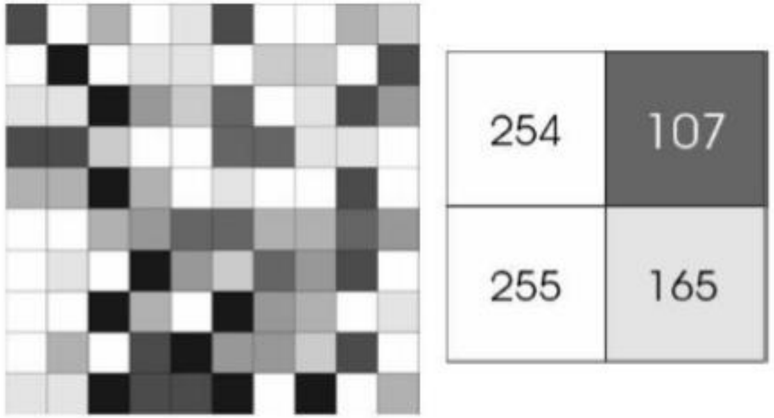
A digital image can be defined as a two-dimensional function, f(x,y), where x and y are spatial coordinates, and the amplitude of f at any pair of coordinates (x, y) is called the intensity of gray level of f of the image at the point.

A digital image can thus be treated as a 2-D array of integers. Let's denote a digital image as f(i, j). The variables take following values:

- i∈[0,h-1], where h is the height of the image
- j∈[0,w-1], where w is the width of the image
- f(i,j) ∈[0,L-1], where L-1=255 for a 8bit image







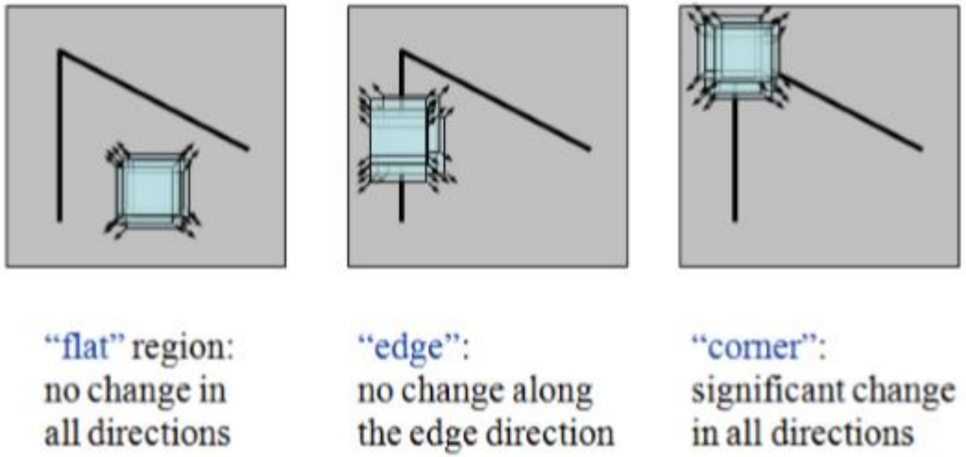
254	107
255	165

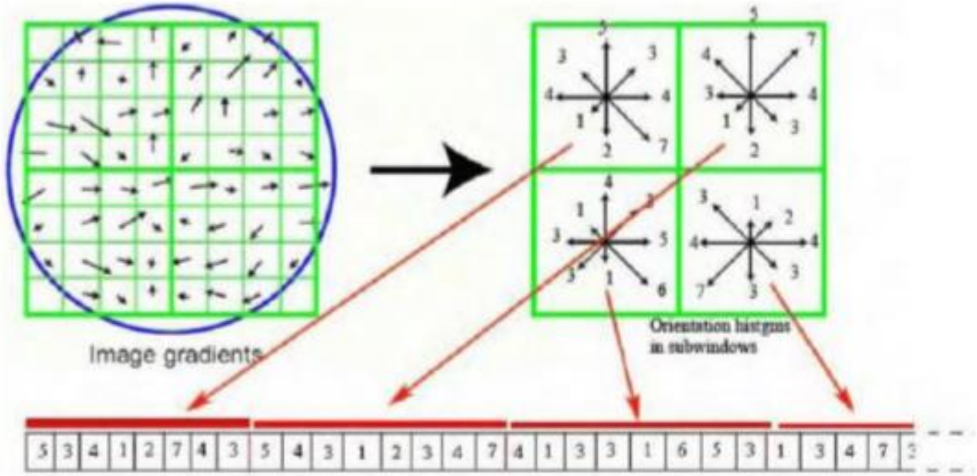
Feature Detectors and Feature Descriptors.

A *feature detector* is an algorithm which takes an image and outputs locations (i.e. pixel coordinates) from the image based on some criterion.

A *feature descriptor* is an algorithm which takes an image and outputs feature vector values, which describes the image patch around an interest point.

Feature descriptors encode interesting information into a series of numbers and act as a sort of numerical **"fingerprint"** that can be used to differentiate one feature from another.





5 3 4 1 2 7 4 3 5 4 3 1 2 3 4 7 4 1 3 3 1 6 5 3 1 3 4 7 3

Point Features (Points of Interest).

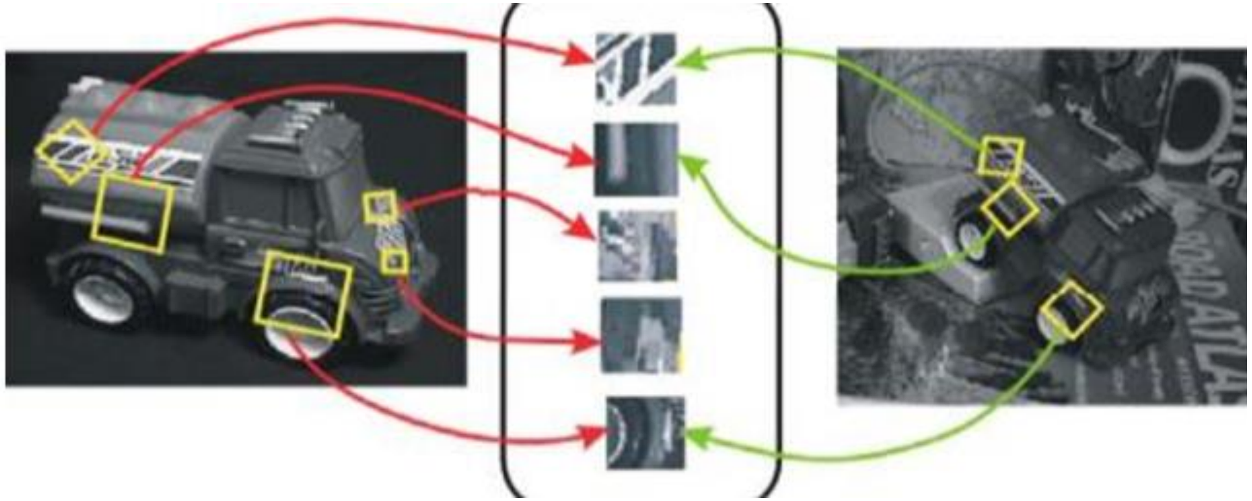
Goal: Detect the same point in each image independently

Challenge: Need repeatability in presence of Scale, Affine distortions and illumination change.

Not all points are good candidates:

-Texture-less regions, edges

Effect of noise on feature extraction

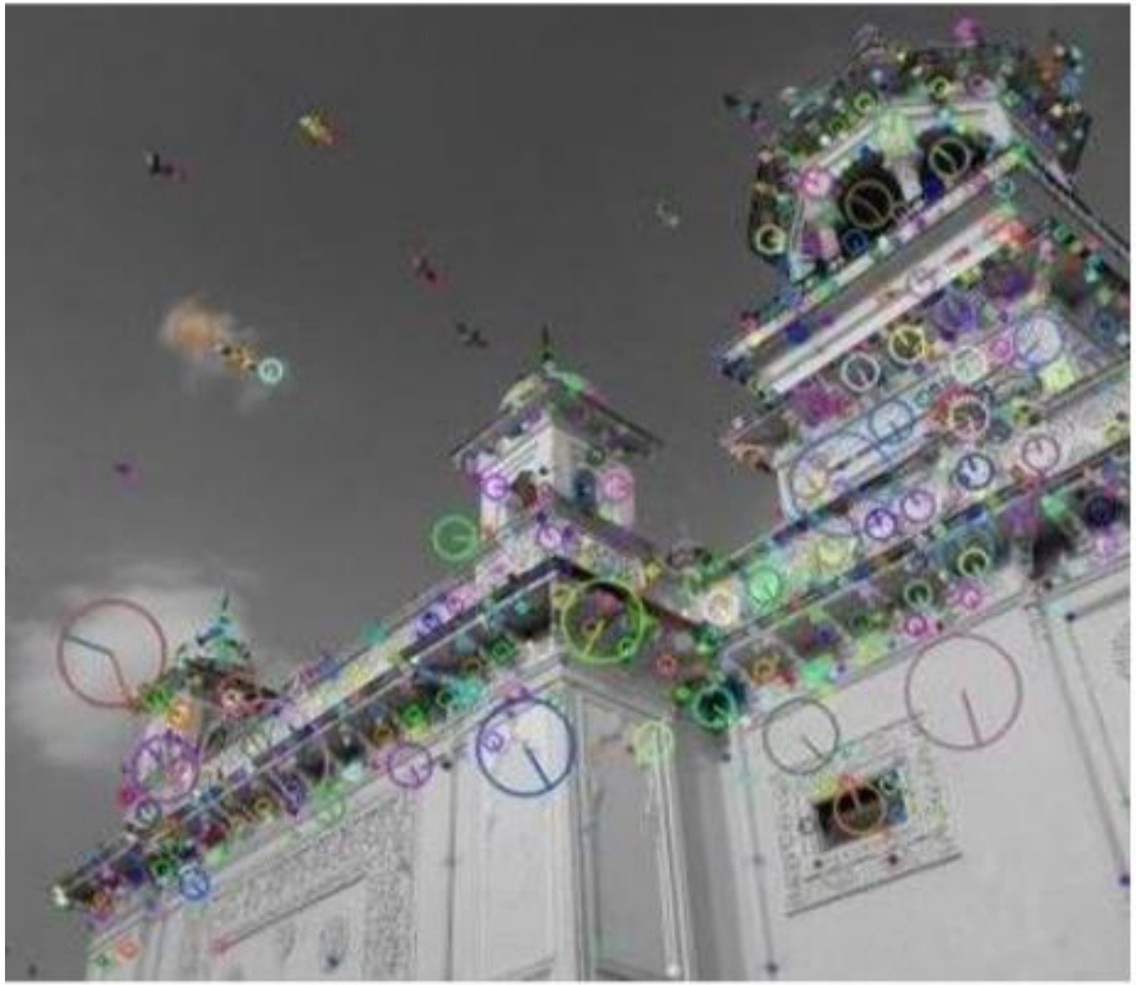


Scale-invariant feature transform (SIFT).

In 2004, D.Lowe, University of British Columbia, came up with a new algorithm, Scale Invariant Feature Transform (SIFT) in his paper, Distinctive Image Features from Scale-Invariant Keypoints, which extract keypoints and compute its descriptors.

For any object in an image, interesting points on the object can be extracted to provide a "feature description" of the object. This description, extracted from a training image, can then be used to identify the object when attempting to locate the object in a test image containing many other objects.

To perform reliable recognition, it is important that the features extracted from the training image be detectable even under changes in image scale, noise and illumination.



Stages of computation SIFT algorithm.

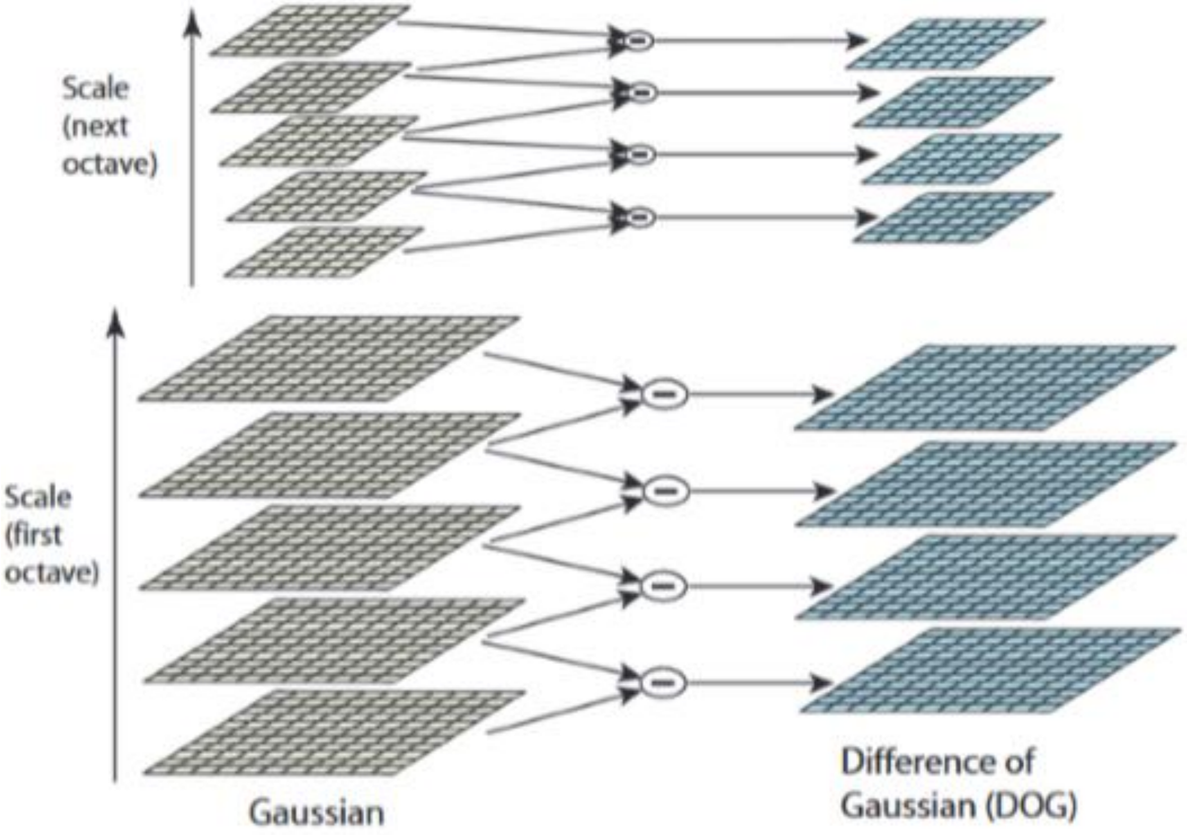
1. **Scale-space extrema detection**: The first stage of computation searches over all scales and image locations. It is implemented efficiently by using a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation.

2. **Keypoint localization:** At each candidate location, a detailed model is fit to determine location and scale. Keypoints are selected based on measures of their stability.

3. **Orientation assignment:** One or more orientations are assigned to each keypoint location based on local image gradient directions. All future operations are performed on image data that has been transformed relative to the assigned orientation, scale, and location for each feature, thereby providing invariance to these transformations.

4. **Keypoint descriptor:** The local image gradients are measured at the selected scale in the region around each keypoint. These are transformed in to a representation that allows for significant levels of local shape distortion and change in illumination.

Construct Scale Space and LoG approximation.



The scale space of an image is defined as a function, $L(x,y,\sigma)$, which is produced from the convolution of a variable-scale Gaussian, $G(x,y,\sigma)$, with an input image, $I\ x,y$.

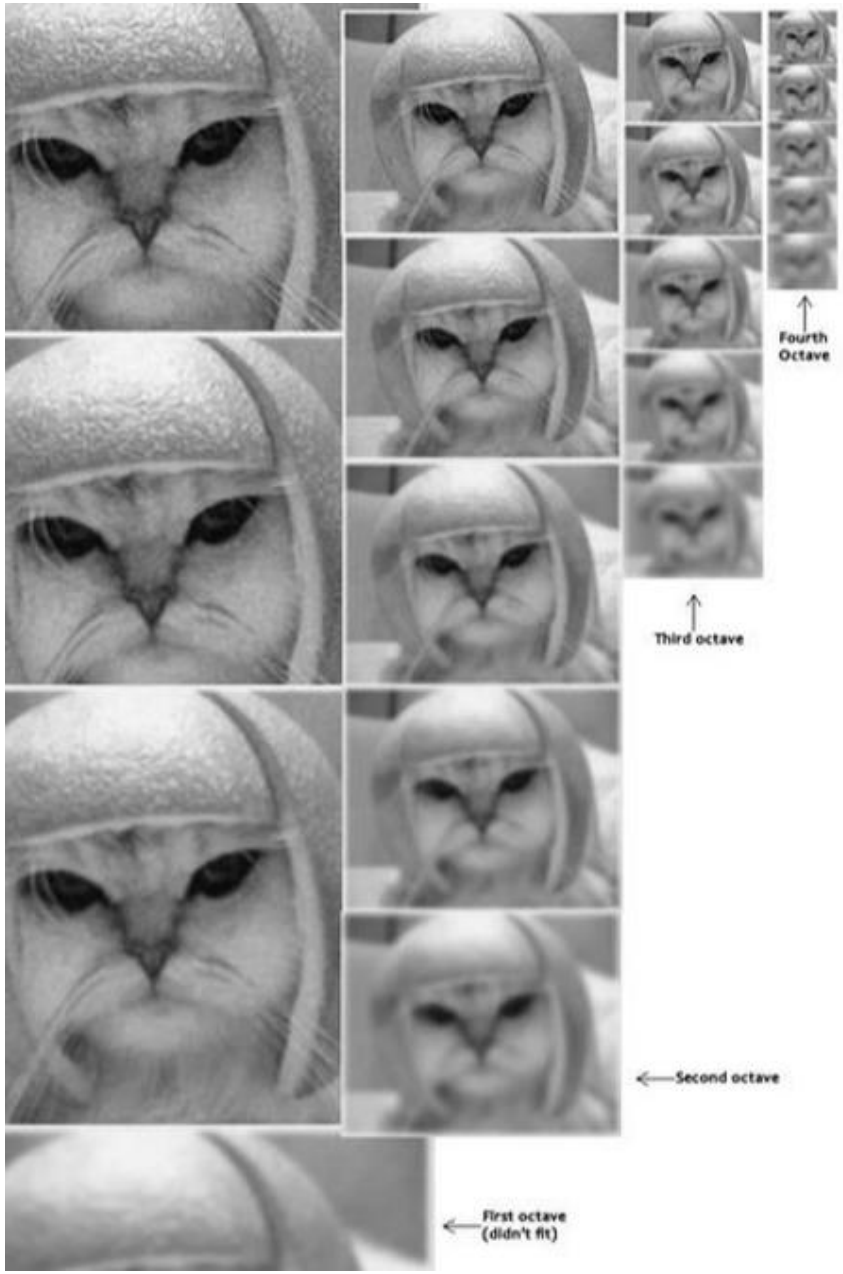
$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y),$$

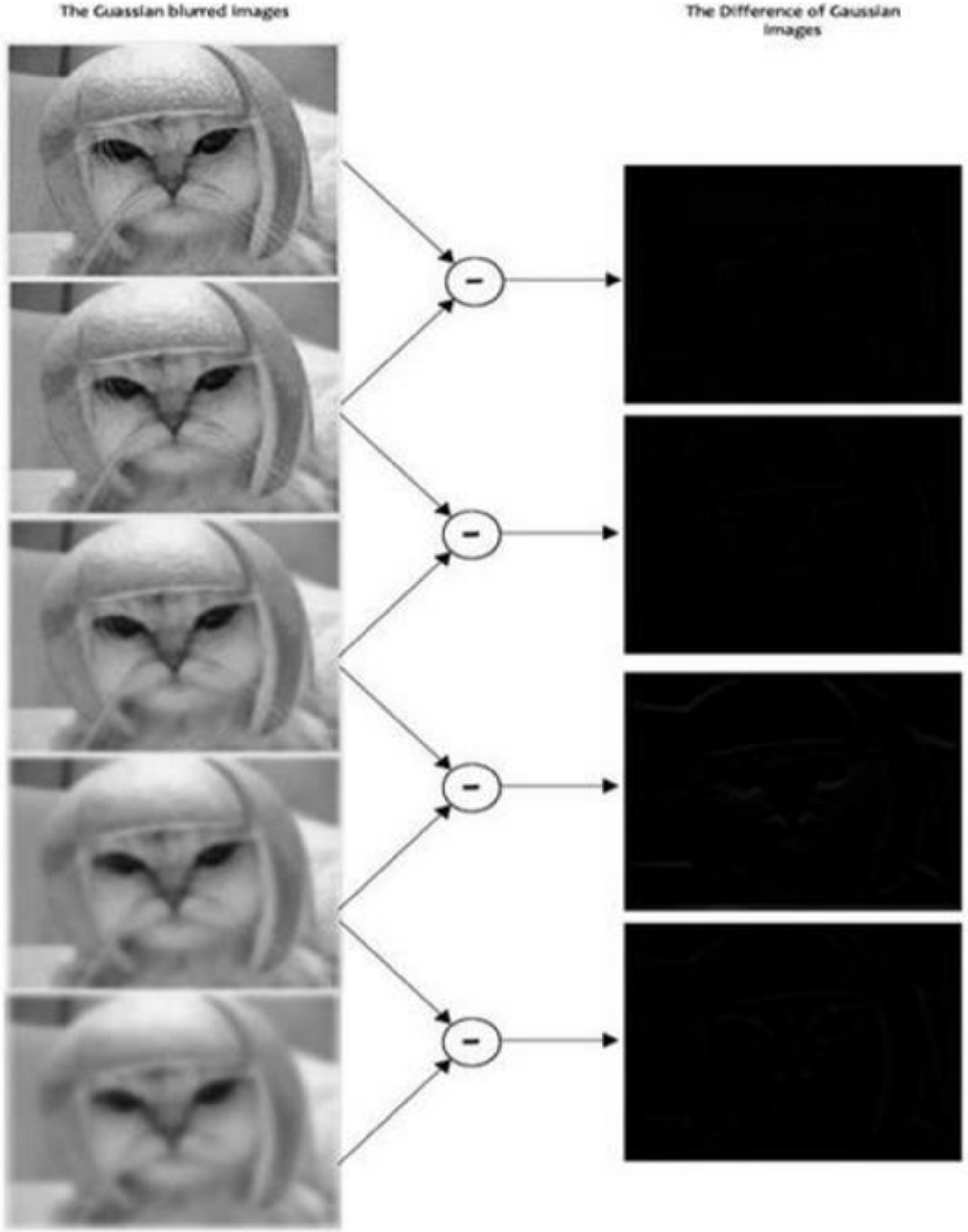
where $*$ is the convolution operation in x and y, and

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

The difference-of-Gaussian function can be computed from the difference of two nearby scales separated by a constant multiplicative factor k:

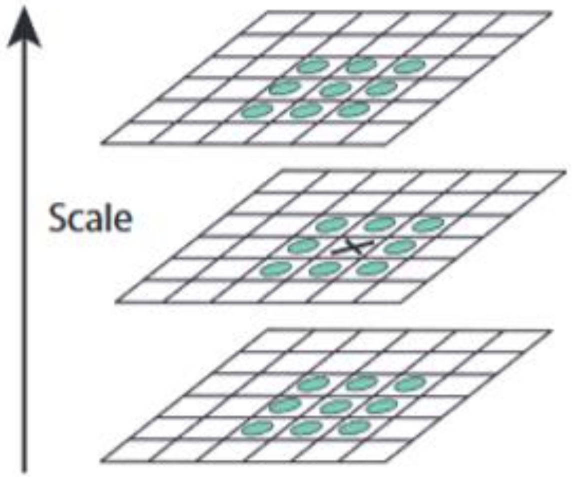
$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y) \\ &= L(x,y,k\sigma) - L(x,y,\sigma). \end{aligned}$$

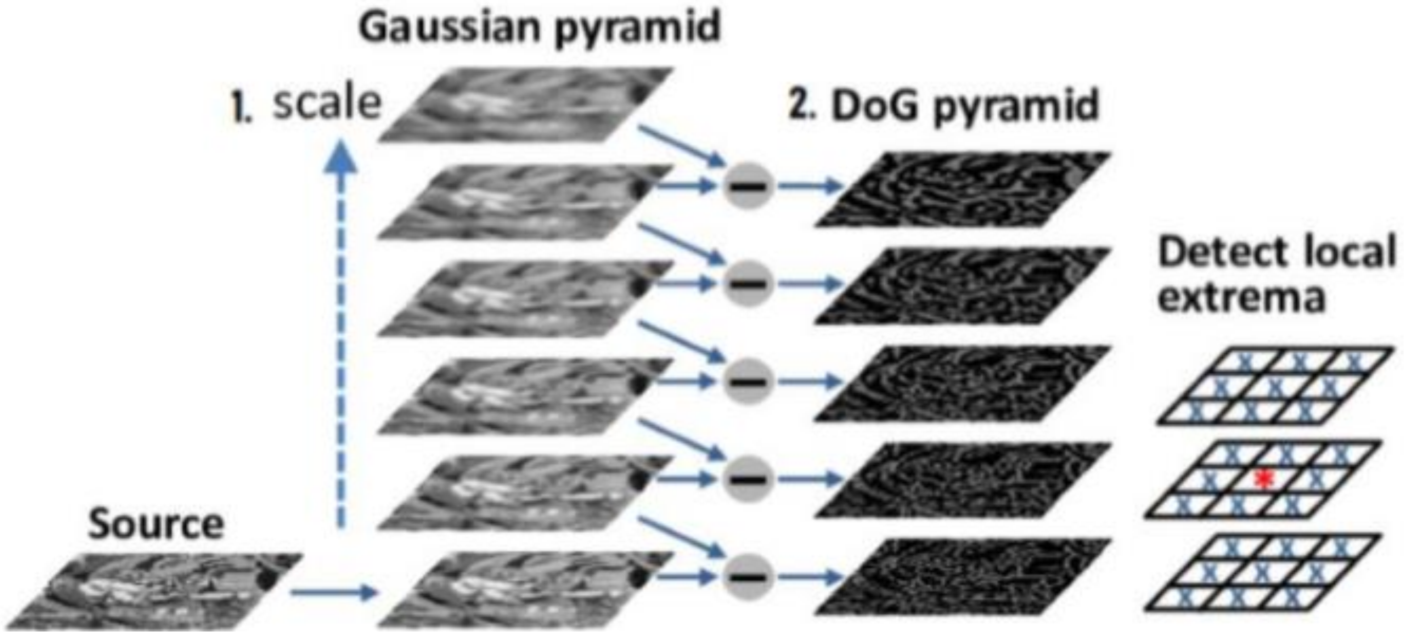

Fourth Octave
Third octave
Second octave
First octave (didn't fit)


The Guassian blurred images
The Difference of Gaussian images
−
−
−
−

Local extrema detection.

In order to detect maxima and minima of the difference-of-Gaussian images, each sample point/pixel (marked with X) is compared to its eight neighbors in the current image and nine neighbors in the scale above and below. Its is selected, only if its larger than all of these neighbors or smaller than all of them.

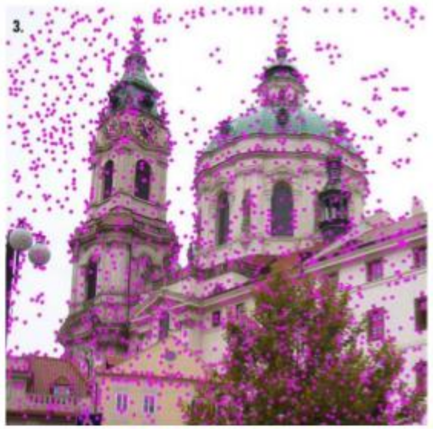


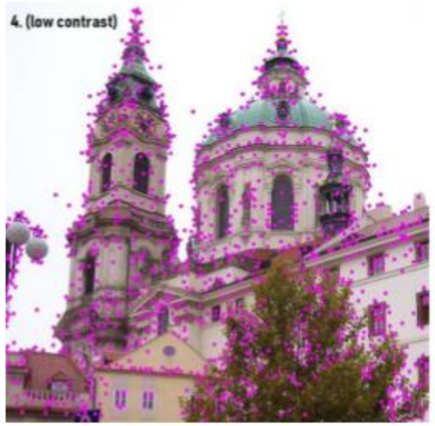


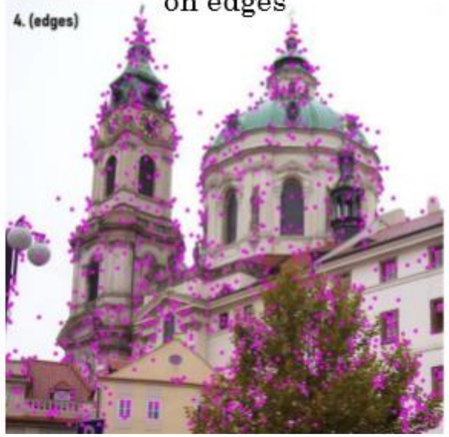
Accurate keypoint localization.

Once a keypoint candidate has been found by comparing a pixel to its neighbors, the next step is to perform a detailed fit to the nearby data for location, scale, and ratio of principal curvatures.

This information allows points to be rejected that have **low contrast** (and are therefore sensitive to noise) or are **poorly localized along an edge**.

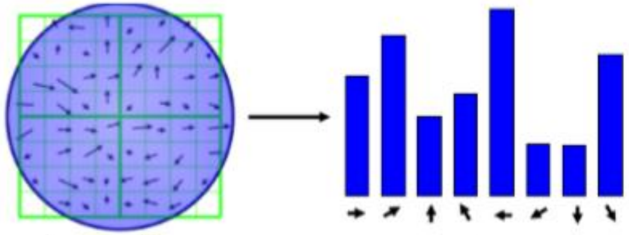

All keypoints

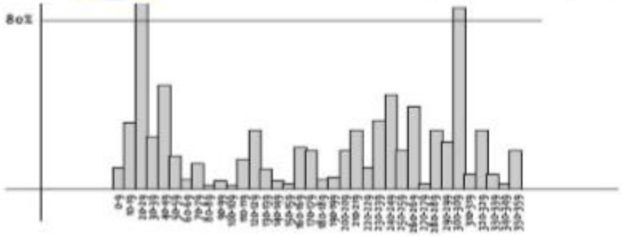

Remove keypoints with low contrast

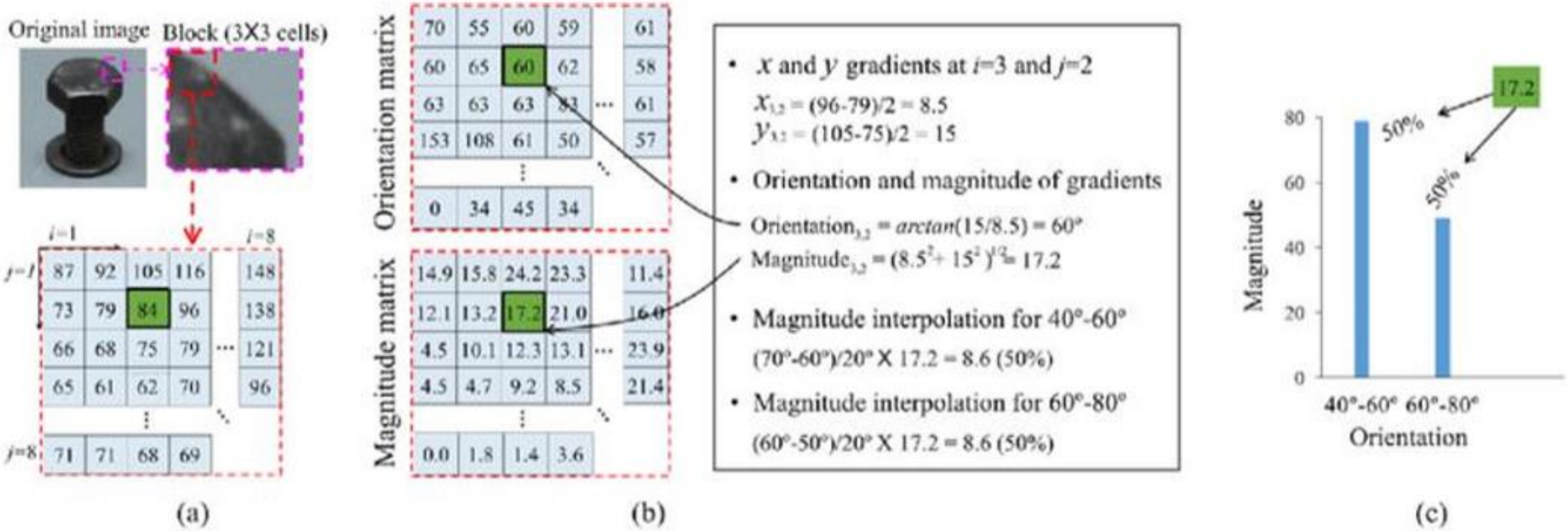

Remove keypoints poorly localized on edges

Orientation Assignment.

Orientation is assigned to each keypoint to achieve invariance to image rotation. A neigbourhood is taken around the keypoint location depending on the scale, and the gradient magnitude and direction is calculated in that region. An orientation histogram with 36 bins covering 360 degrees is created. (It is weighted by gradient magnitude and gaussian-weighted circular window with σ equal to 1.5 times the scale of keypoint. The highest peak in the histogram is taken and any peak above 80% of it is also considered to calculate the orientation. It creates keypoints with same location and scale, but different directions







87	92	105	116		148
73	79	84	96		138
66	68	75	79	⋯	121
65	61	62	70		96
		⋮		⋱	
71	71	68	69		

70	55	60	59		61
60	65	60	62		58
63	63	63	83	⋯	61
153	108	61	50		57
		⋮		⋱	
0	34	45	34		

14.9	15.8	24.2	23.3		11.4
12.1	13.2	17.2	21.0		16.0
4.5	10.1	12.3	13.1	⋯	23.9
4.5	4.7	9.2	8.5		21.4
		⋮		⋱	
0.0	1.8	1.4	3.6		

- x and y gradients at i=3 and j=2

  $x_{3,2}$ = (96-79)/2 = 8.5
  $y_{3,2}$ = (105-75)/2 = 15

- Orientation and magnitude of gradients

  Orientation$_{3,2}$ = *arctan*(15/8.5) = 60°
  Magnitude$_{3,2}$ = $(8.5^2 + 15^2)^{1/2}$ = 17.2

- Magnitude interpolation for 40°-60°

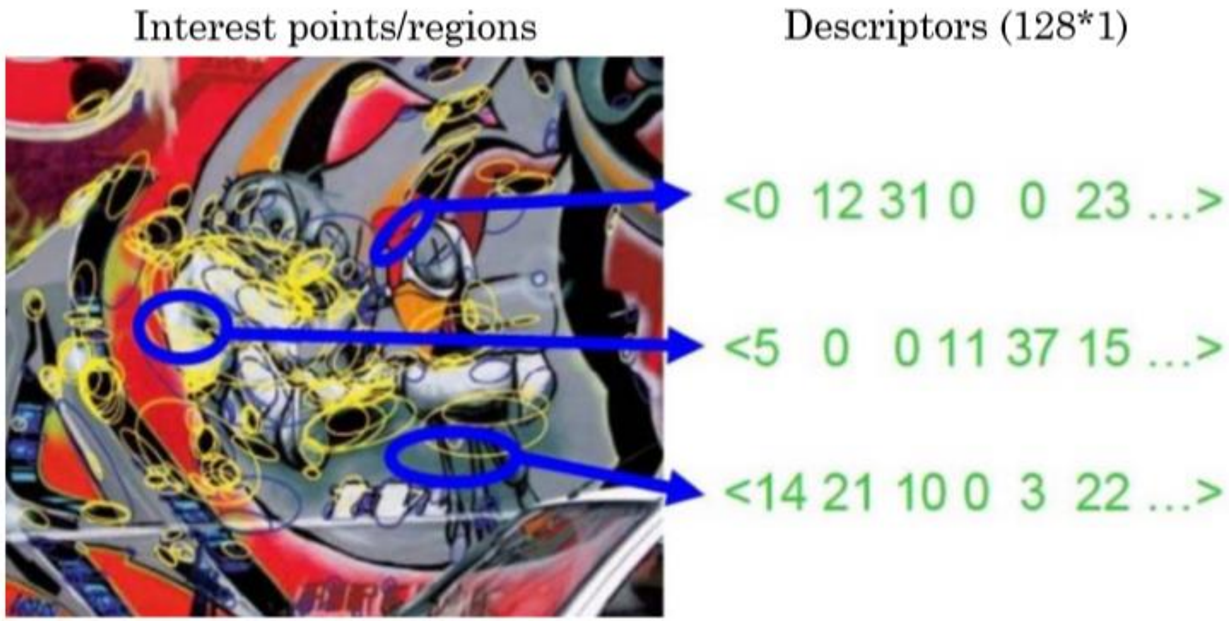
  (70°-60°)/20° X 17.2 = 8.6 (50%)

- Magnitude interpolation for 60°-80°

  (60°-50°)/20° X 17.2 = 8.6 (50%)

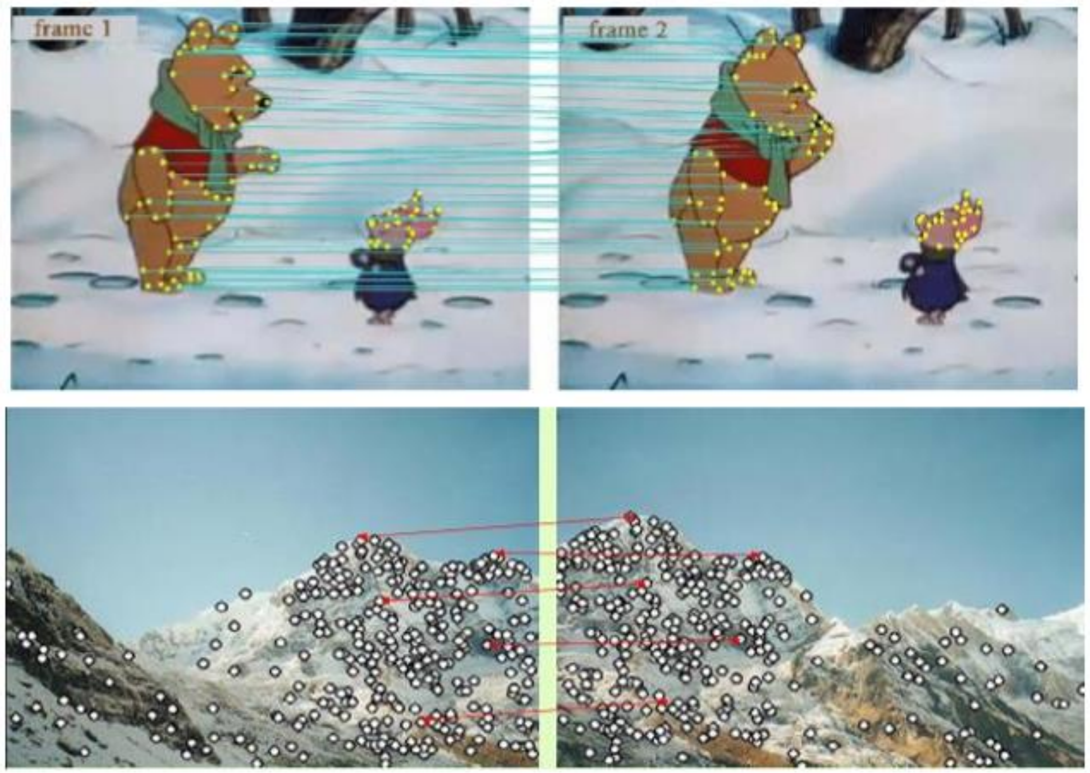
Keypoint Descriptor.

Now keypoint descriptor is created. A 16x16 neighbourhood around the keypoint is taken. It is devided into 16 sub-blocks of 4x4 size. For each subblock, 8 bin orientation histogram is created. So a total of 128 bin values are available. It is represented as a vector to form keypoint descriptor. In addition to this, several measures are taken to achieve robustness against illumination changes, rotation etc.



Keypoint Matching.

The best candidate match for each keypoint is found by identifying its nearest neighbor in the database of keypoints from training images. The nearest neighbor is defined as the keypoint with minimum **Euclidean distance** for the invariant descript or vector.



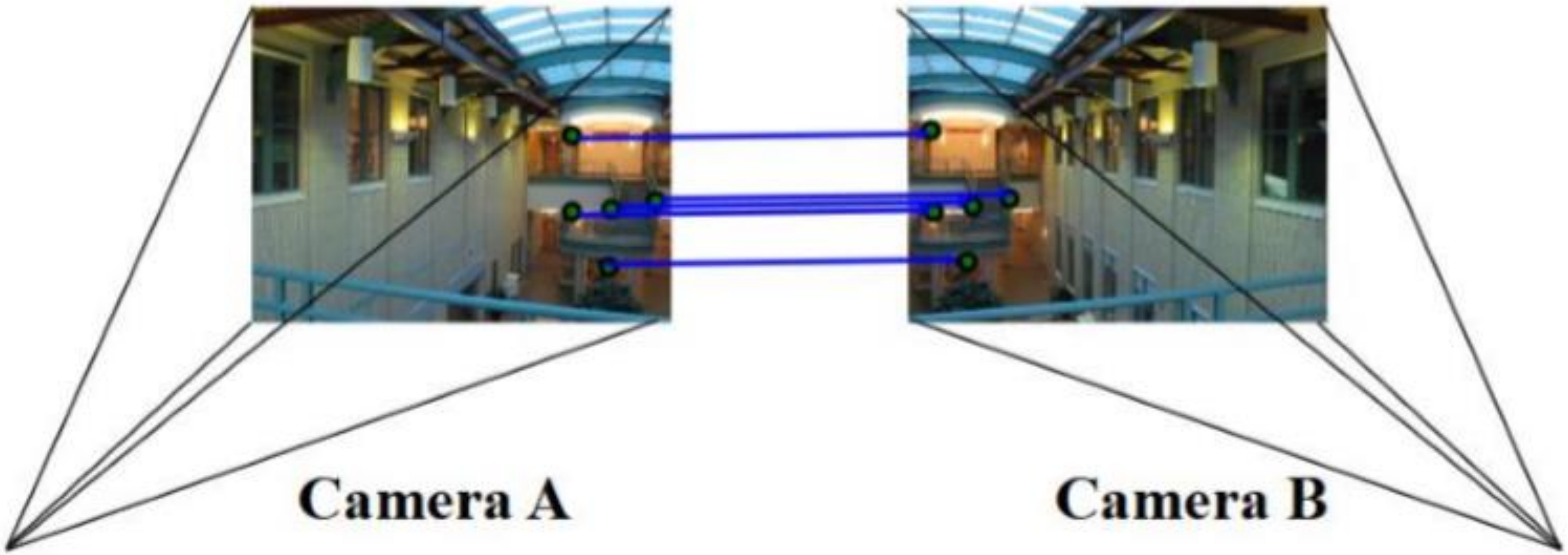
Final SIFT results.

SIFT robustness to:

- affine distortion
- change in 3D viewpoint
- addition of noise
- changes in illumination

RANSAC.

The RANdom SAmple Consensus (RANSAC) algorithm proposed by Fischler and Bolles is a general parameter estimation approach designed to cope with a large proportion of outliers in the input data. It randomly chooses a minimal set of observations and evaluates their likelihood until a good solution is found or a preset number of trials is reached. It has regularly been applied for estimation of model parameters in feature matching, detection and registration.

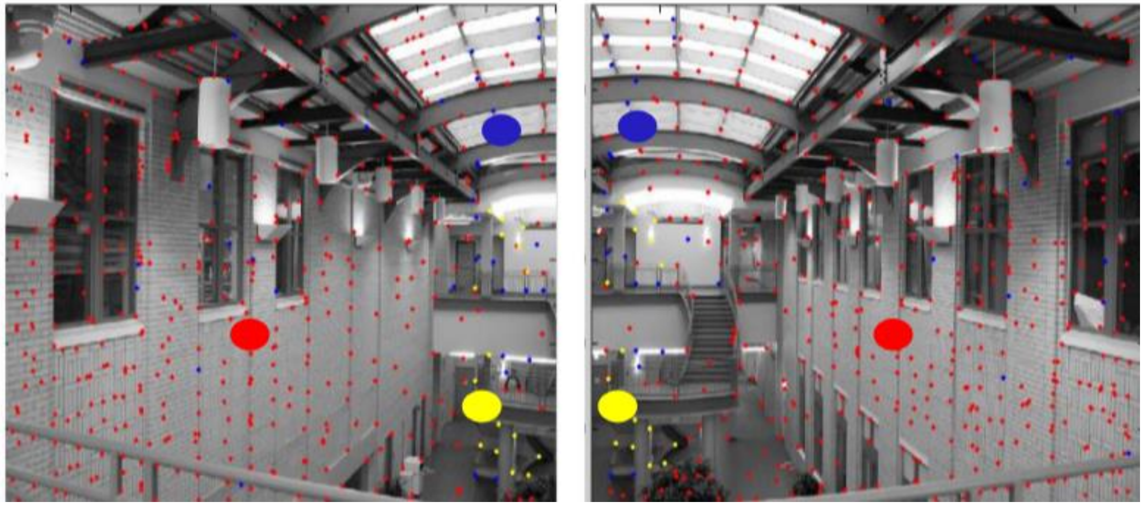


Color coding of points produced by SIFT.

Red points: the points without a "good" match in the other image.In this image, the goodness of the match is decided by looking at the ratio of the distances to the second nearest neighbor and first nearest neighbor. If this ration is high (above some threshold), it is considered a "good" match.

Blue points: these are points with a "good" match in which the match was wrong, meaning it connected two points that did not actually correspond in the world.

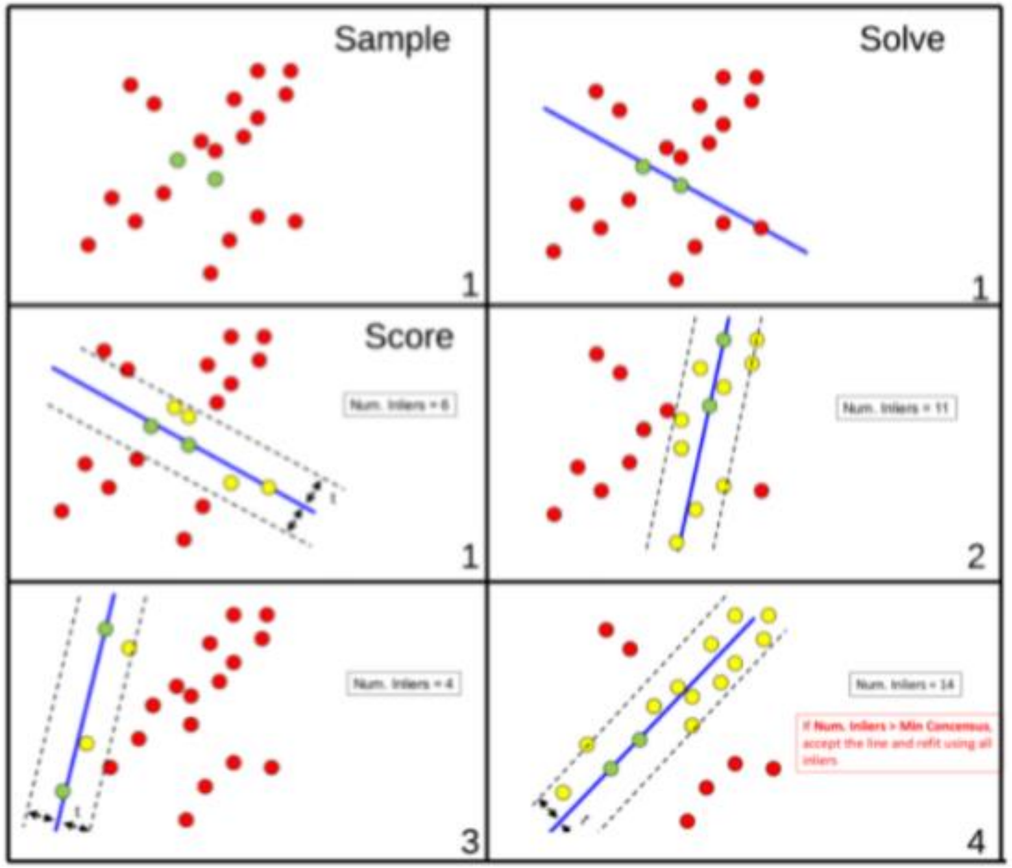
Yellow points: these are correct matches. We need to run RANSAC until it randomly picked 4 yellow points from among the blue and yellow points (the matches estimated to be "good").



RANdom SAmple Consensus (RANSAC).

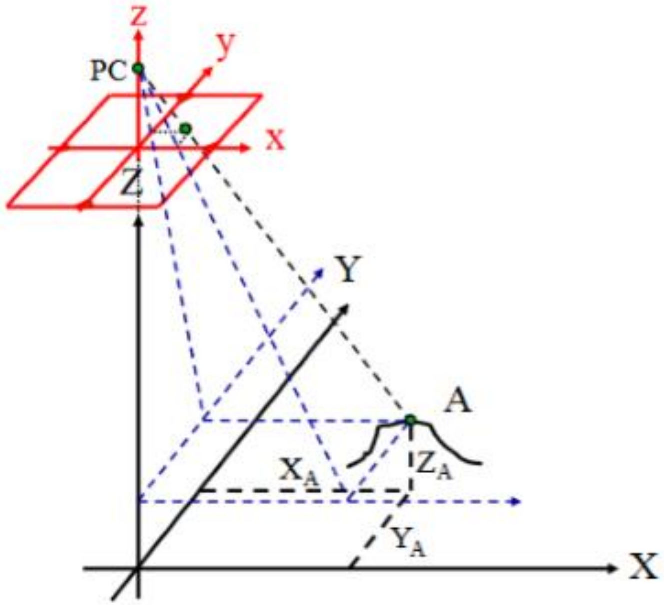
Steps of the Algorithm:

i. Select randomly the minimum number of points required to determine the model parameters.
ii. Solve for the parameters of the model.
iii. Determine how many points from the set of all points fit with a predefined tolerance E.
iv. If the fraction of the number of inliers over the total number points in the set exceeds a predefined threshold τ, reestimate the model parameters using all the identified inliers and terminate.
v. Otherwise, repeat steps 1 through 4 (maximum of N times)



Bundle Adjustment.

Bundle adjustment is a unified triangulation method to simultaneously estimate the internal and external camera parameters and the 3D coordinates of the scene points in a statistically optimal manner. Conceptually, it solves the inverse problem to computer graphics: given the images of an unknown scene the task is to recover the scene structure, i.e., the visible surface together with the parameters describing the cameras used for taking the images.



The task is to automatically estimate:

- Position, Orientation and Focal Length of cameras
- 3D positions of feature points, by minimizing the sum of squares of reprojecting errors.

In other words:

a) We have a set of points in real world defined by their coordinates (X,Y,Z) in some apriori chosen "world coordinate frame".

b) We photograph these points by different cameras, which are characterized by their orientation and translation relative to the world coordinate frame and also by focal length and two radial distortion parameters (so 9 parameters in total).

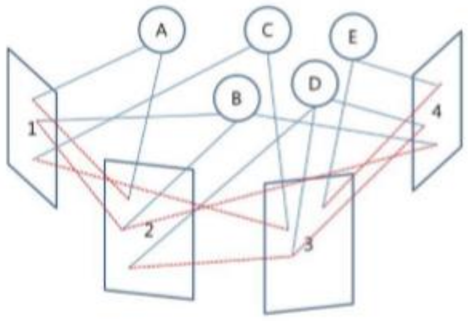
c) Then we precicely measure 2-D coordinates (x,y) of the points projected by the cameras on images.

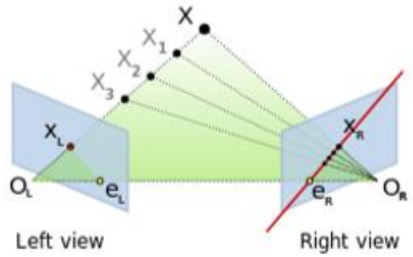
Bundle Adjustment.

Each camera sees several points.

Each point is seen by several cameras.

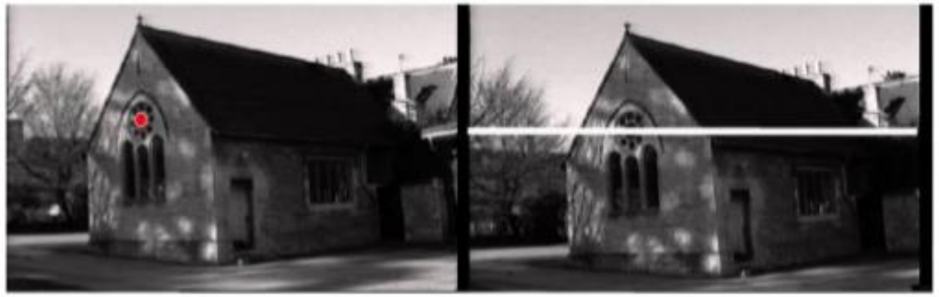
Cameras are independent of each other (given the points), same for the points.





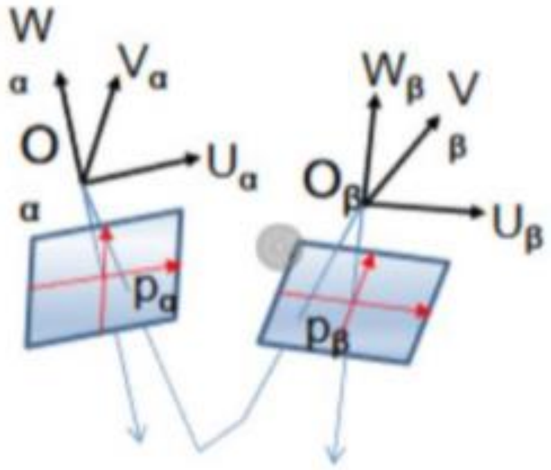
Epipolar geometry .

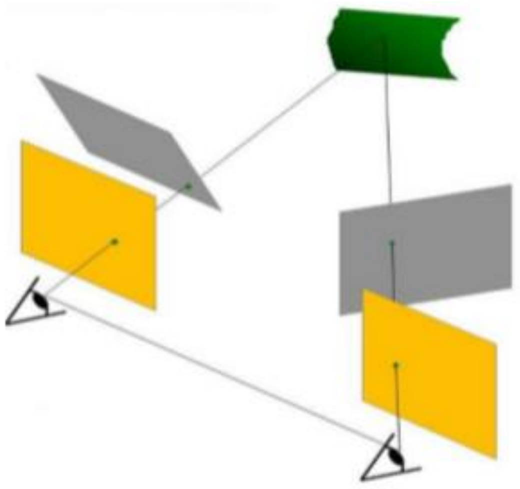
A point in one image “generates” a line in another image (called the epipolar line).

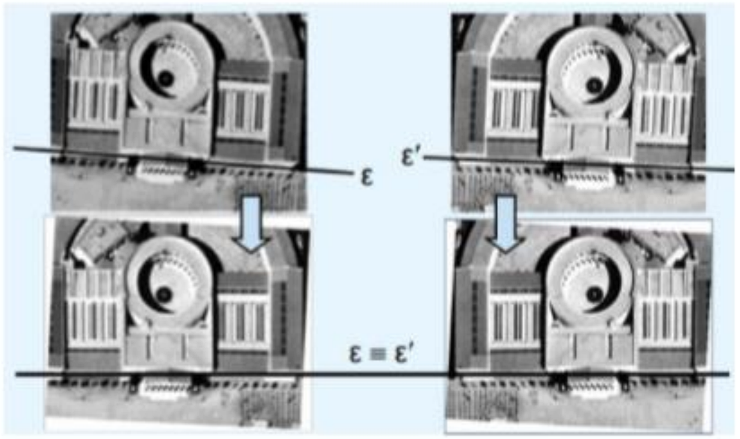


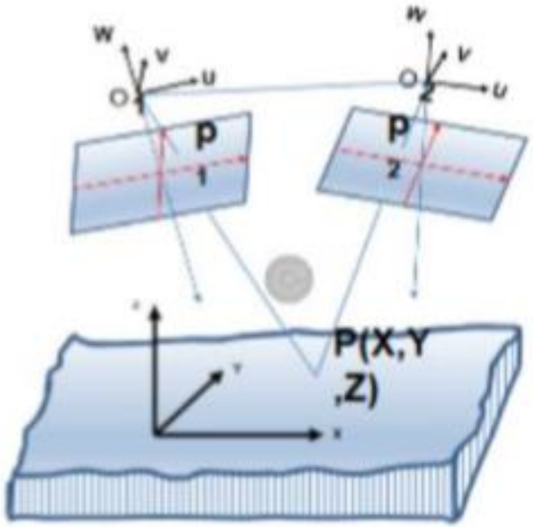
Rectification of images.

In practice, it is convenient if the image scanlines (rows) are the epipolar lines. So bundle adjustment reproject image planes onto a common plane parallel to the baseline. Afterwards pixel motion is orizontal.









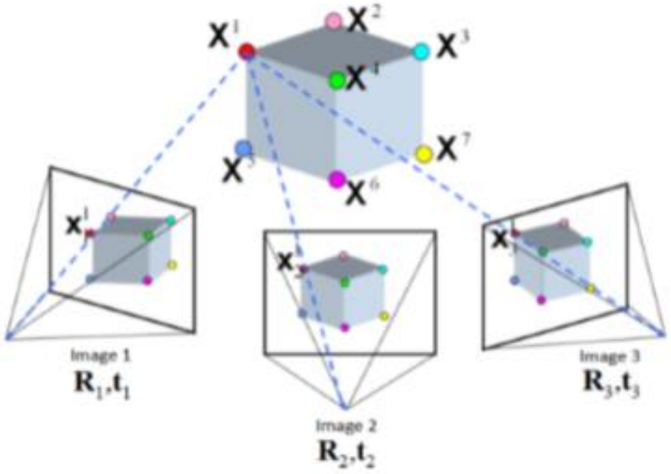
Bundle adjustment review.

Advantages

- Most accurate triangulation technique since we have direct transformation between image and ground coordinates.
- Straight forward to include parameters that compensate for various deviations from the collinearity model.
- Can be used for normal, convergent, aerial, and close range imagery.

Disadvantages

- Model is non linear: approximations as well as partial derivatives are needed.
- Requires computer intensive computations.

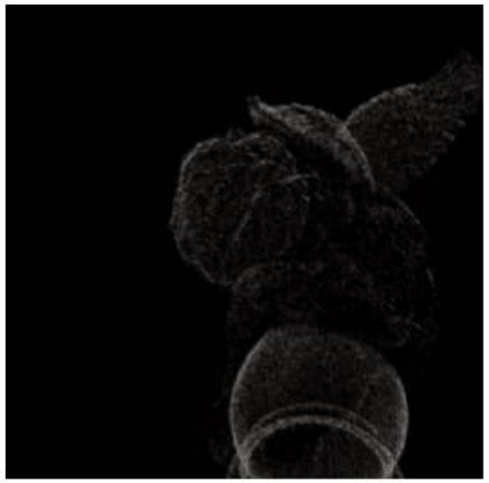


Multi-View Stereo.

MVS refines the mesh produced by SFM technique, to produce a dense reconstruction, given the camera parameters of each image to work.

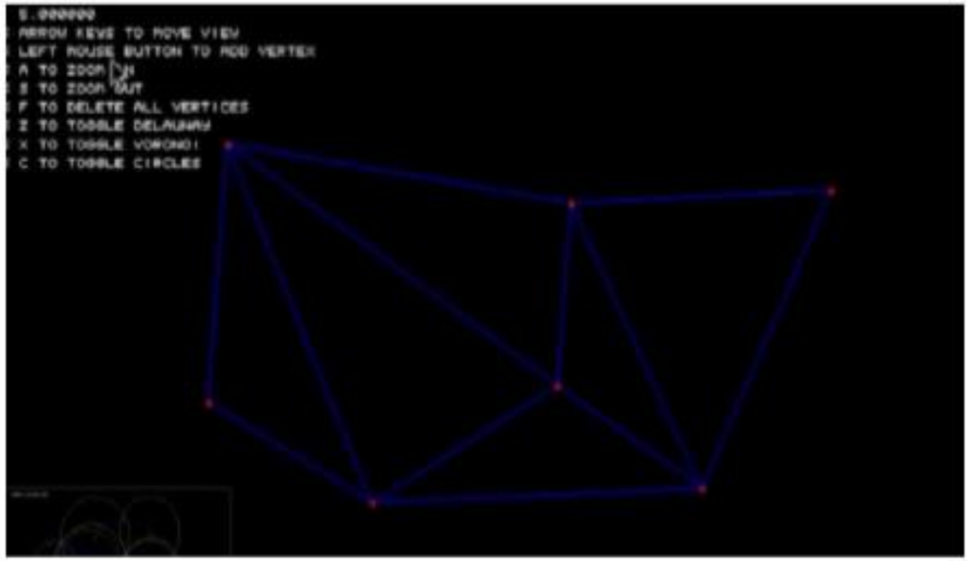
Clustering Views for Multi-view Stereo (CMVS): takes the output of a structure-from-motion (SFM) as an input, then decomposes the input images into smaller clusters.

Patch-based Multi-view Stereo (PMVS): computes 3D vertices which were not correctly detected by descriptors or matched points (creation of dense cloud)



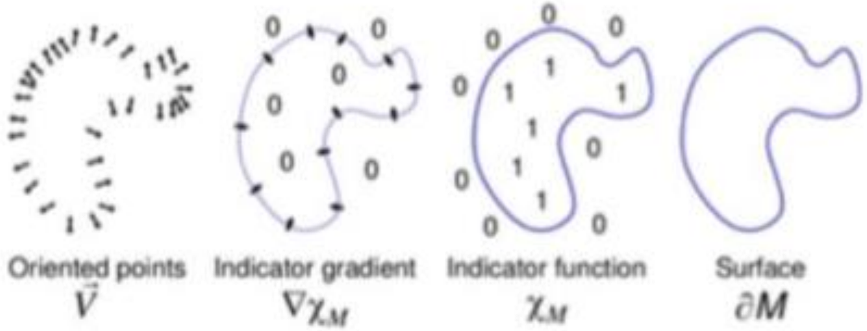
Surface Reconstruction from Points.

Delaunay triangulation: Given set P of discrete points in a plane is a triangulation DT(P) such that no point in P is inside the circumcircle of any triangle in DT(P).



```
5.000000
ARROW KEYS TO MOVE VIEW
LEFT MOUSE BUTTON TO ADD VERTEX
A TO ZOOM IN
S TO ZOOM OUT
F TO DELETE ALL VERTICES
Z TO TOGGLE DELAUNAY
X TO TOGGLE VORONOI
C TO TOGGLE CIRCLES
```

Poisson Surface Reconstruction: Surface reconstructed based on Poisson equation


Oriented points
$\vec{V}$
Indicator gradient
$\nabla \chi_M$
Indicator function
χ_M
Surface
∂M