Review of Computer Vision Techniques

Jason Rebello
16/05/2017

Part I : What is Computer Vision and how is an image formed

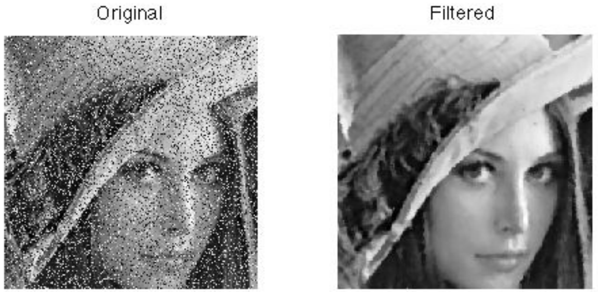
Part || : Feature Detection and Feature Matching

Part ||| : Feature Tracking and Prediction

Part IV : Applications (Bag Of Words)

What is Computer Vision and How is an Image Formed

Image Processing: Image Manipulation (Motion compensation, Filtering)



Noise Removal

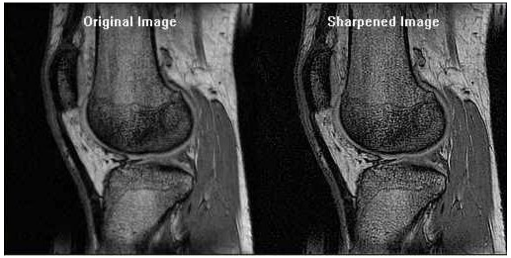


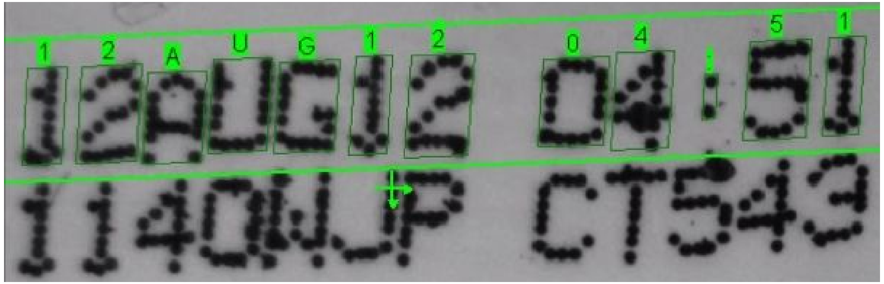
Image Sharpening

Computer Vision: Scene Interpretation

Semantic Segmentation

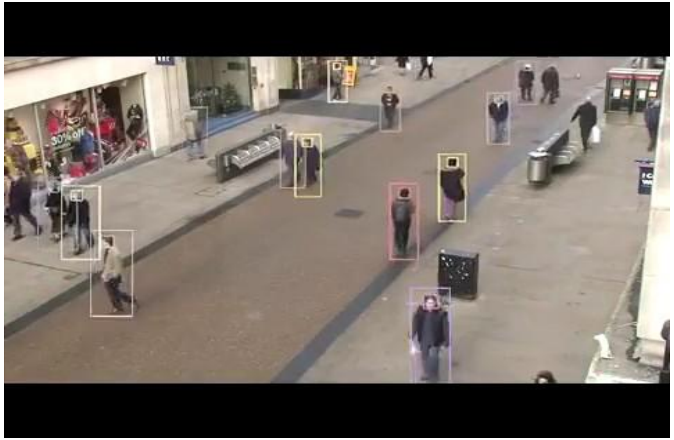
3D Reconstruction

Autonomous driving Source: Tesla

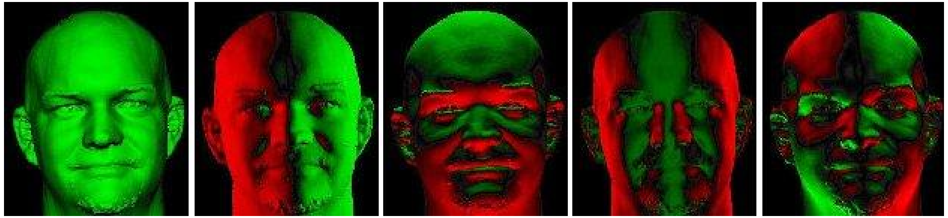


Optical Character Recognition

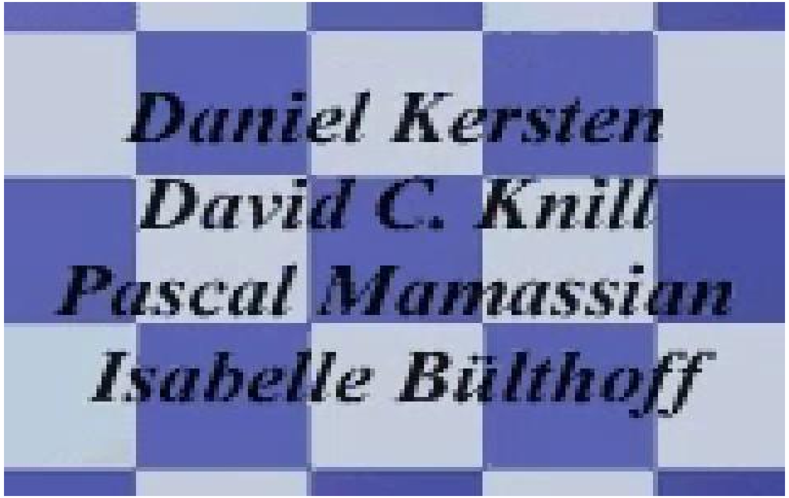
Structure from Motion Source: DrCalleOlsson



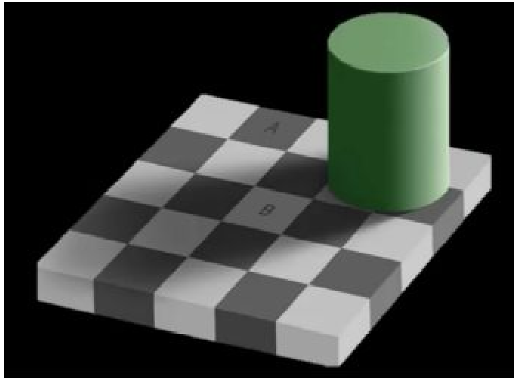
Surveillance Source: Ben Benfold & Ian Reid

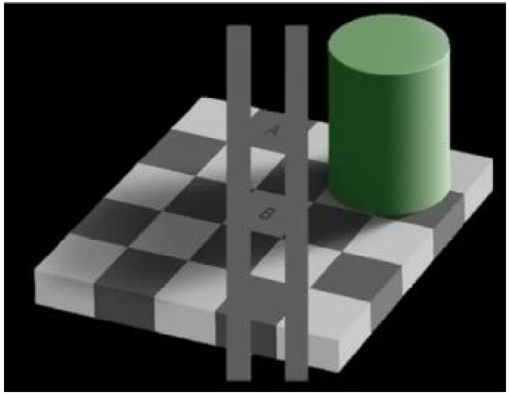


Shadow Effects

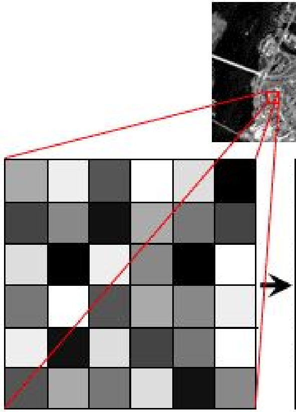


Motion from shadows
Source: Dan Kersten



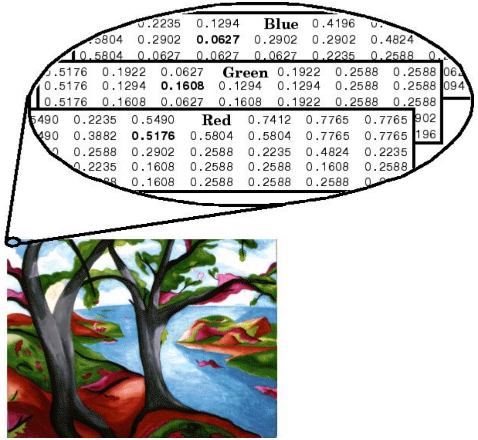


Local vs Global Perception



170	238	85	255	221	0
68	136	17	170	119	68
221	0	238	136	0	255
119	255	85	170	136	238
238	17	221	68	119	255
85	170	119	221	17	136

Gray Scale

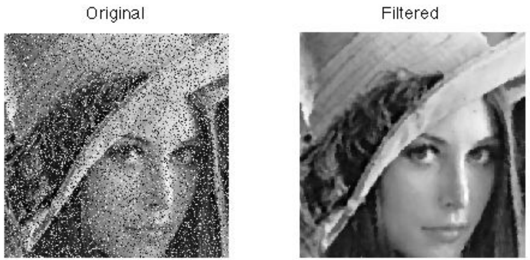


Colour Image

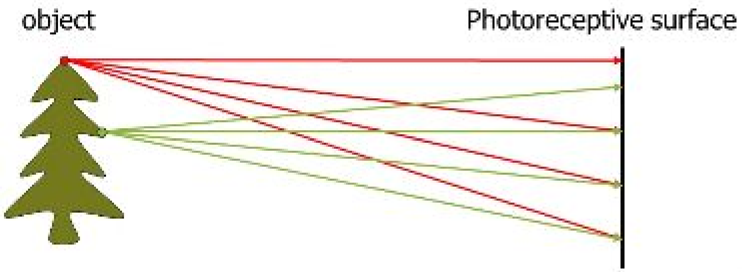
- Images are thought of Functions

- Images:
  Sample 2D space
  Quantize Intensity

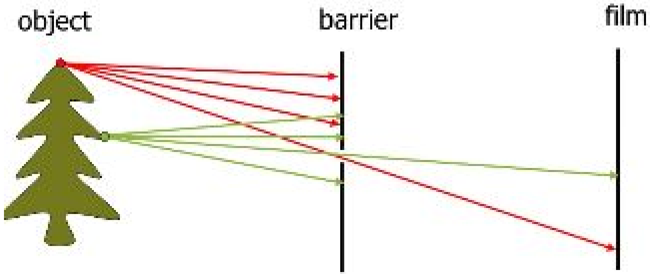
- Noise is another function combined with original function



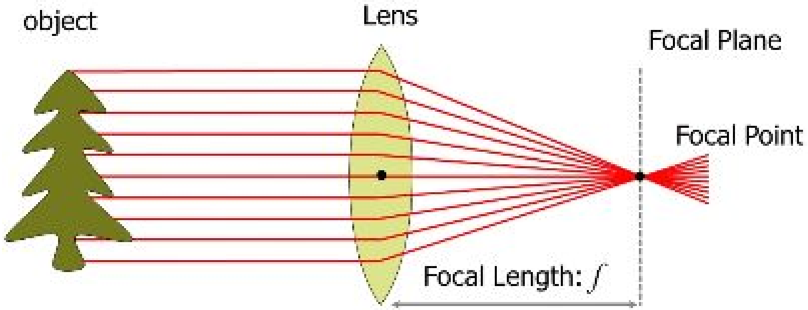
Salt and Pepper Noise



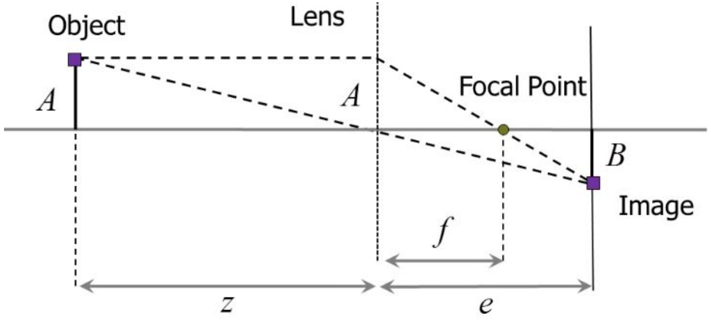
No Aperture



Single Ray: Dim Image, Diffraction



Lens: Multiple Light Rays



Thin Lens Diagram

Thin Lens Equation $$\frac{1}{f} = \frac{1}{z} + \frac{1}{e}$$

Source: Scaramuzza, AMRx

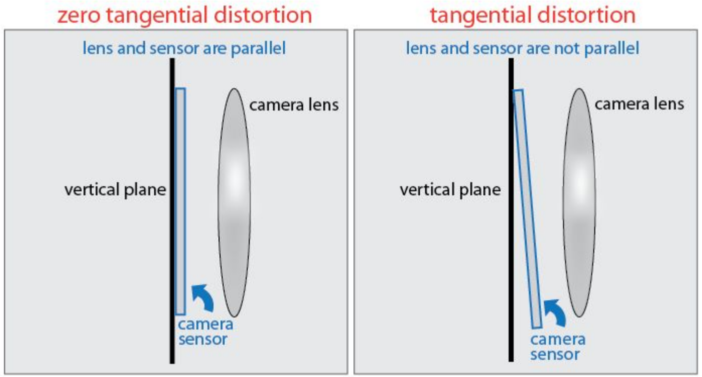
No distortion

Barrel distortion

Pincushion

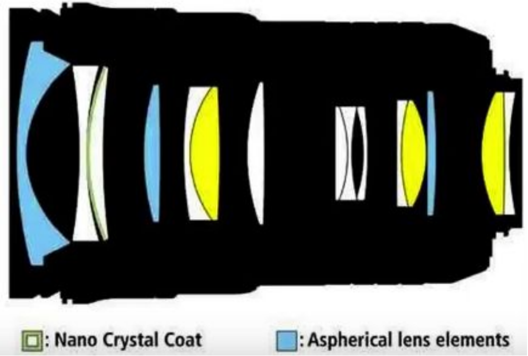
Radial Distortion

Source: Scaramuzza



Tangential Distortion

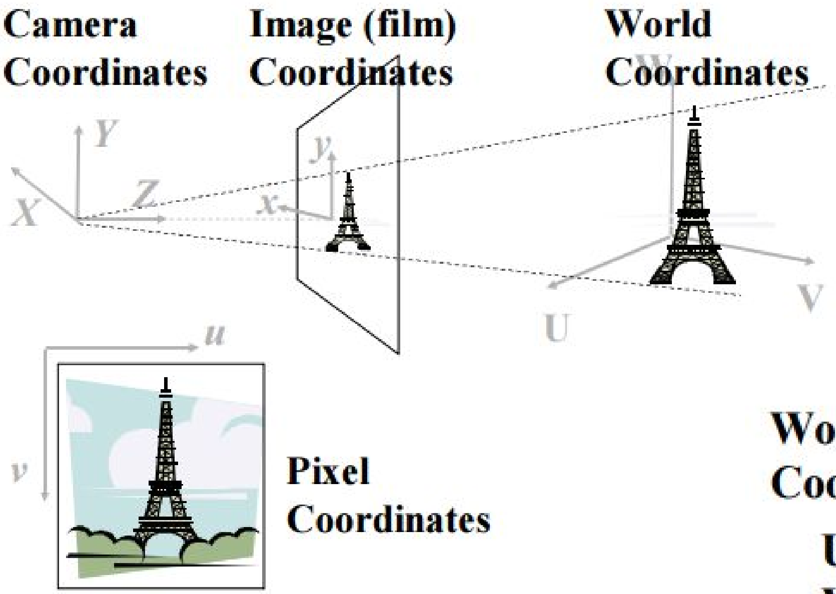
Source: MathWorks



Nikon Lens System

Source: Aaron Bobik

Forward Projection



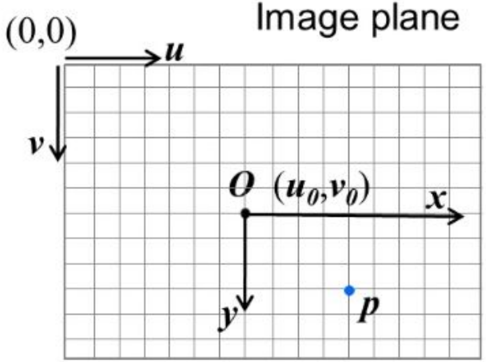
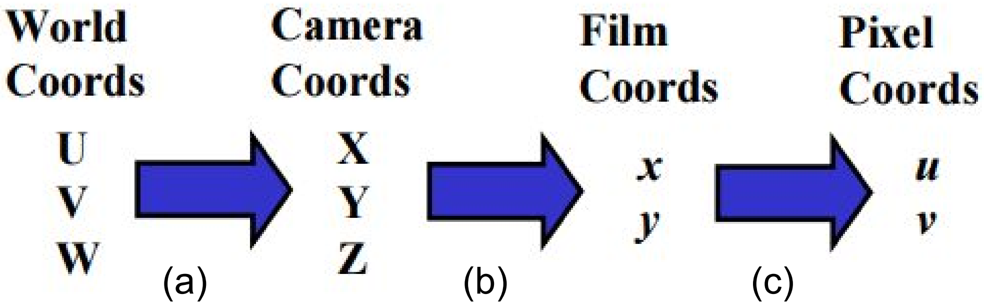


Image to Pixel Coordinates



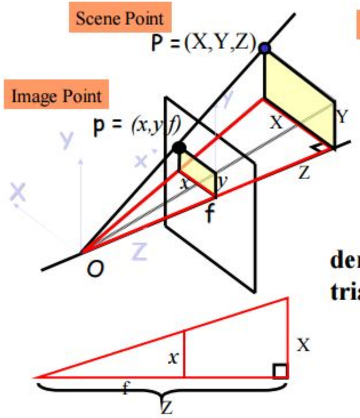
Source: Robert Collins, CSE486

(a) Extrinsic Transformation (Rotation + Translation)

- Transforms points from World to Camera coordinate Frame
- Homogeneous coordinates allow for easy matrix multiplication

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} U \\ V \\ W \\ 1 \end{pmatrix}$$

(b),(c) Perspective Projection



Perspective Projection Eqns

$$x = f\frac{X}{Z}$$

$$y = f\frac{Y}{Z}$$

derived via similar triangles rule

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} f/s_x & 0 & o_x & 0 \\ 0 & f/s_y & o_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$u = \frac{x'}{z'}$$

$$v = \frac{y'}{z'}$$

Camera Coordinates

Pixel Coordinates

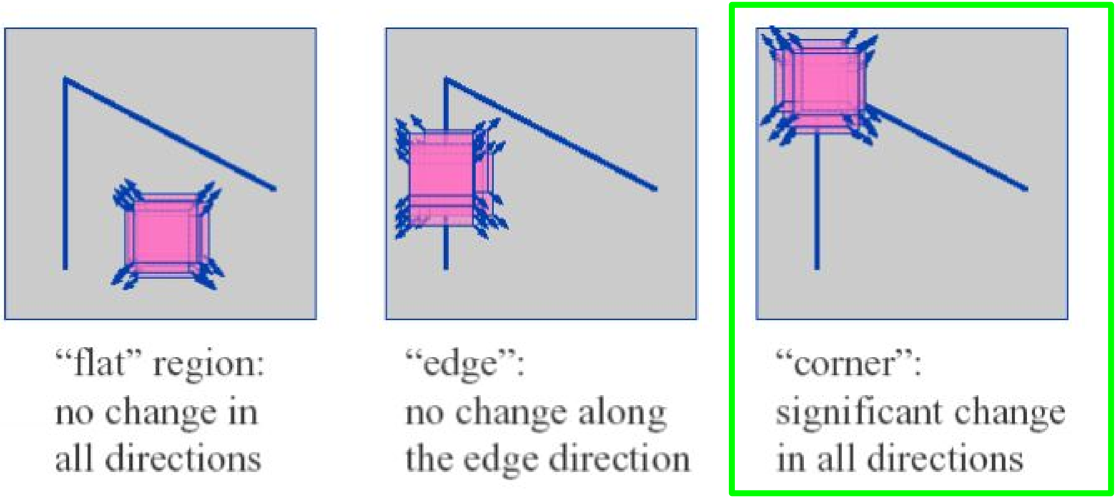
Source: Robert Collins, CSE486

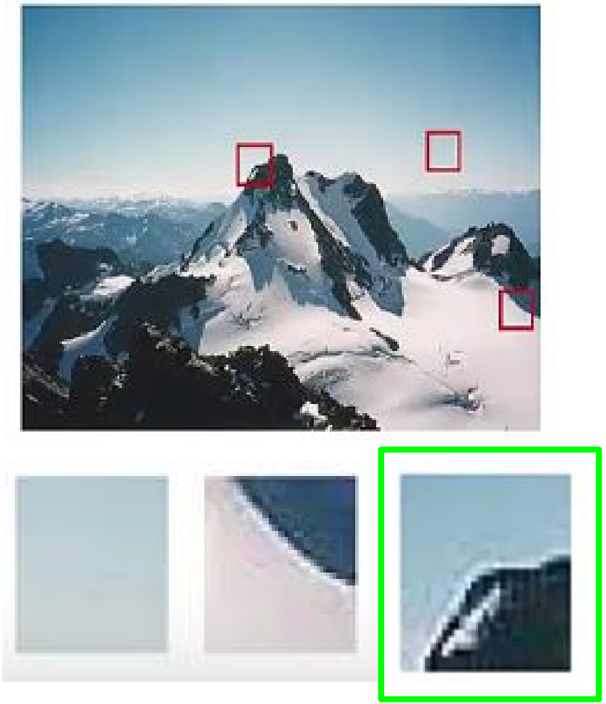

Questions?

Feature Detection and Feature Matching

Need to find reliably detectable and discriminable locations in images

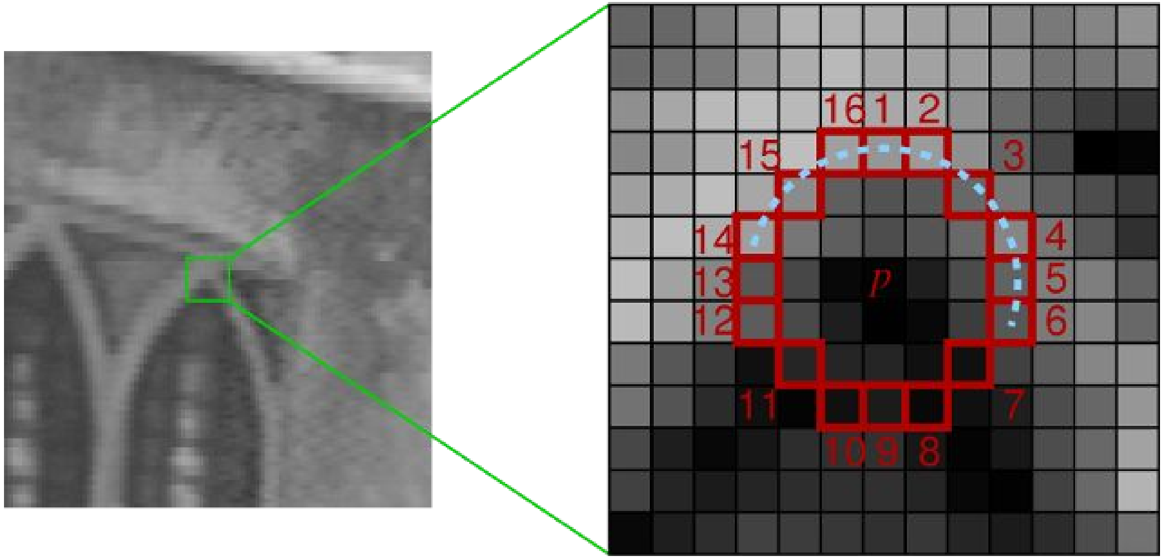
- Repeatability (Across images)
- Precise (Location)
- Saliency (Distinctive description)
- Compactness (Few features)
- Locality (Size of descriptor region)





Source: Mubarak Shah

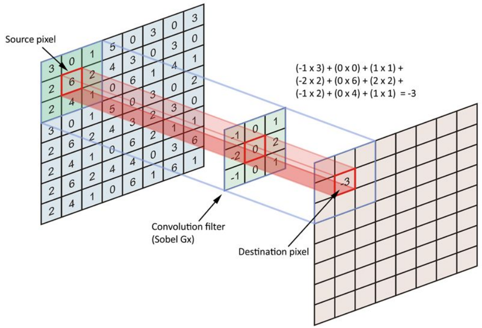
- Let ‘P’ be point of interest
- Select threshold ‘t’
- Consider 16 pixels around Point ‘P’
- ‘P’ is a corner if ‘n’ consecutive pixels with corresponding intensity greater or less than intensity of ‘P’ exist
- Shi-Tomasi, GFTT, BRIEF, SURF, SIFT



Features from Accelerated Segment Test

Source: OpenCV

- Image Convolution using Sobel Operator



- Consider a grayscale image I. Calculate the variation in the gradient by sweeping a window in x and y direction

$$E(u,v) = \sum_{x,y} w(x,y)[I(x+u, y+v) - I(x,y)]^2$$

- Approximation in matrix form

$$E(u,v) \approx [u \;\; v] \left(\sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \right) \begin{bmatrix} u \\ v \end{bmatrix}$$

- Determine Score for each window

$$R = \det(M) - k(\operatorname{trace}(M))^2$$

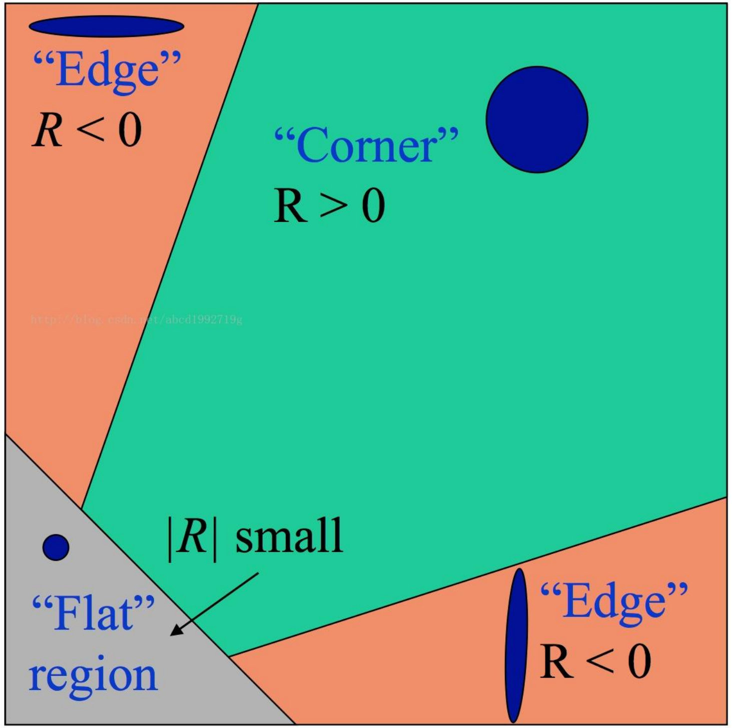
- Score using Eigen values

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

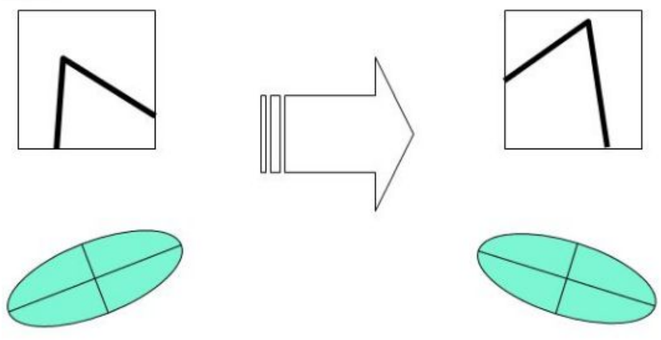
$$\det(M) = \lambda_1 \lambda_2$$
$$\operatorname{trace}(M) = \lambda_1 + \lambda_2$$

$$R = \det(M) - \alpha\, \text{trace}(M)^2 = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

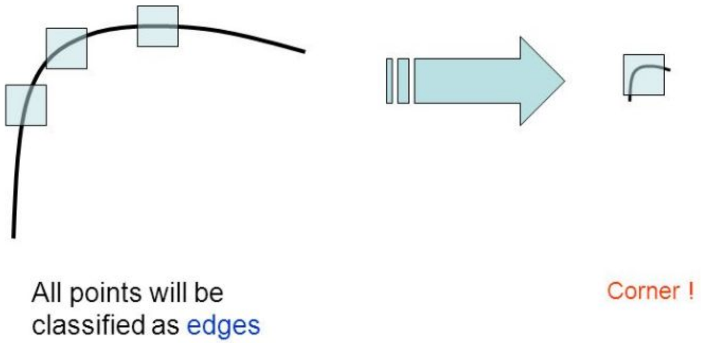
α: constant (0.04 to 0.06)



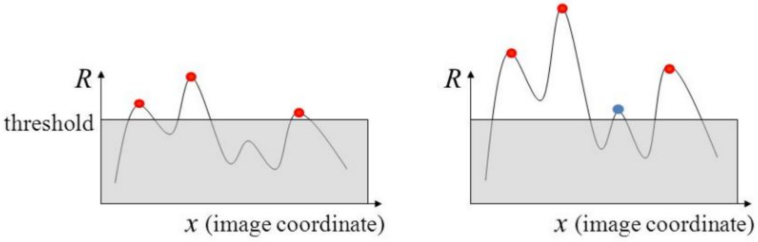
Rotation Invariant



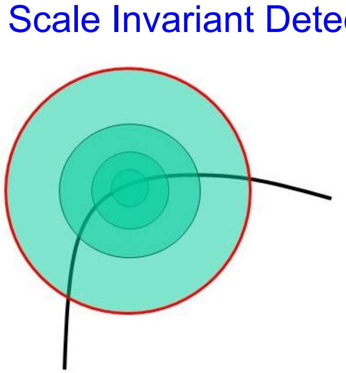
Not Scale Invariant



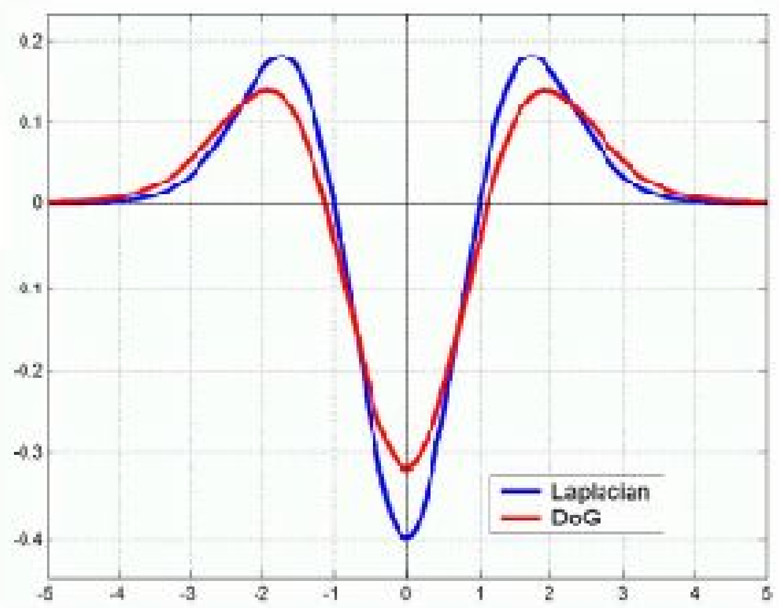
Uniform Intensity Invariance



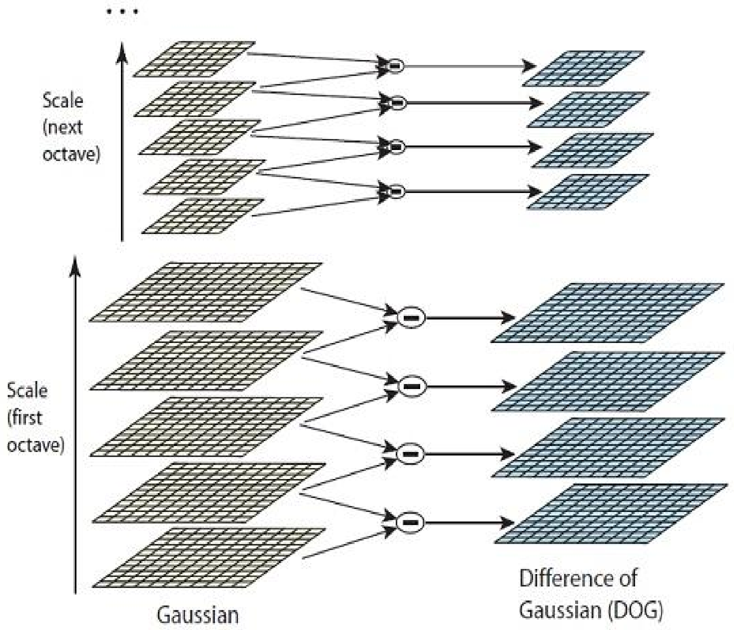
Scale Invariant Detection



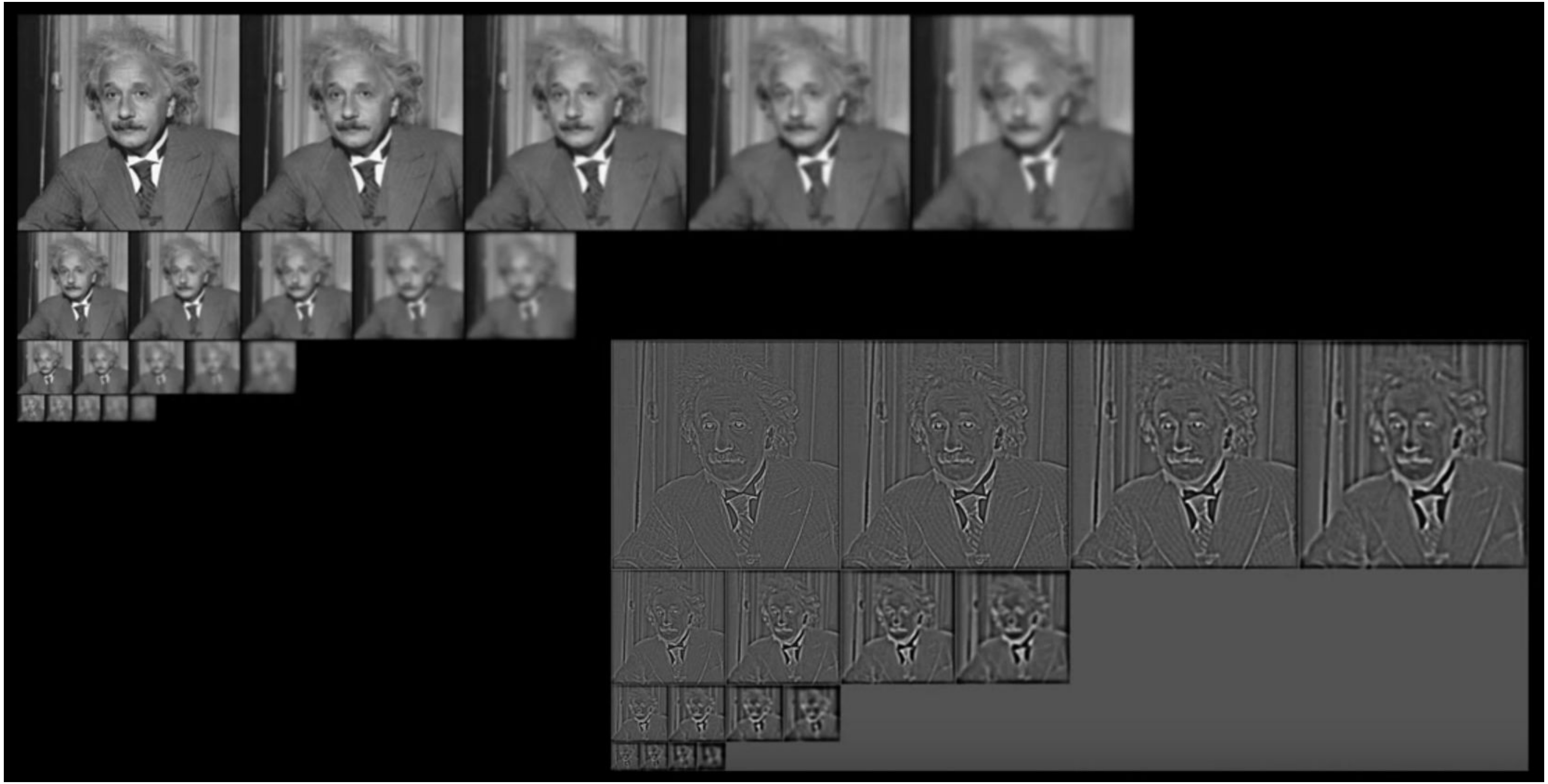
Laplacian of Gaussian vs
Difference of Gaussian



Scale Space Generation

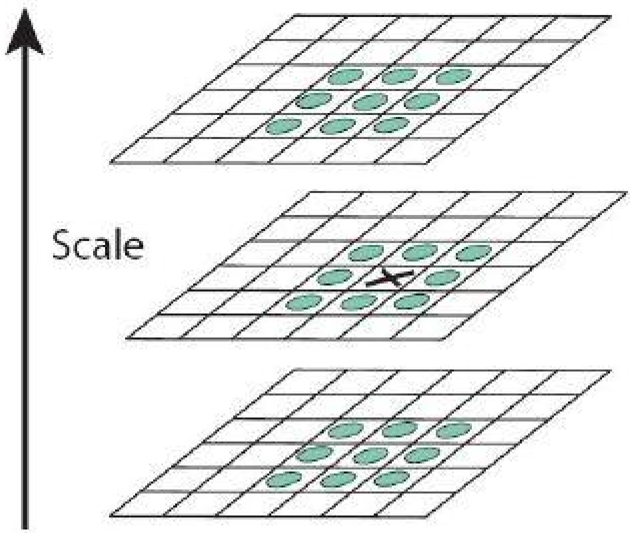


Extrema at different scales

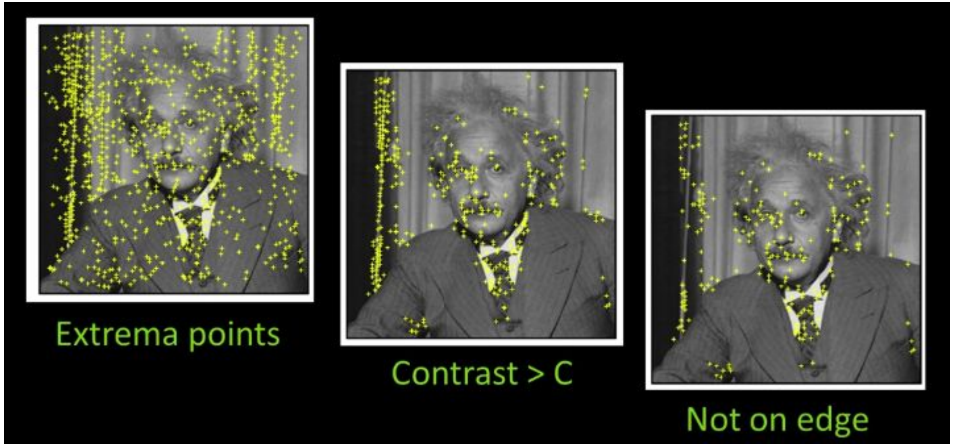


Source: Aaron Bobik

Find Maxima in Scale Space



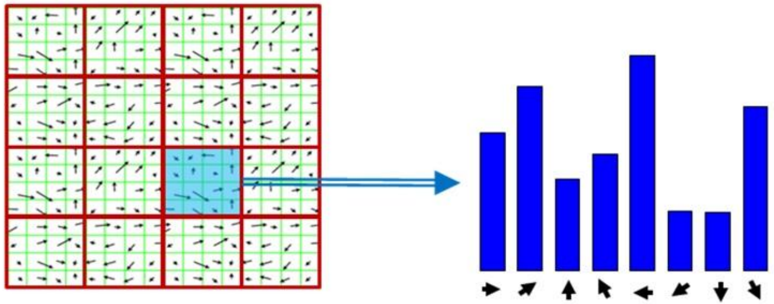
SIFT Feature Points



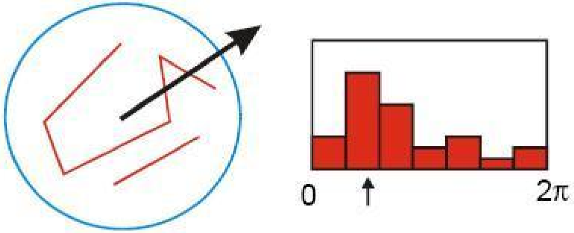
Source: Aaron Bobik

Feature Descriptors | SIFT Descriptor

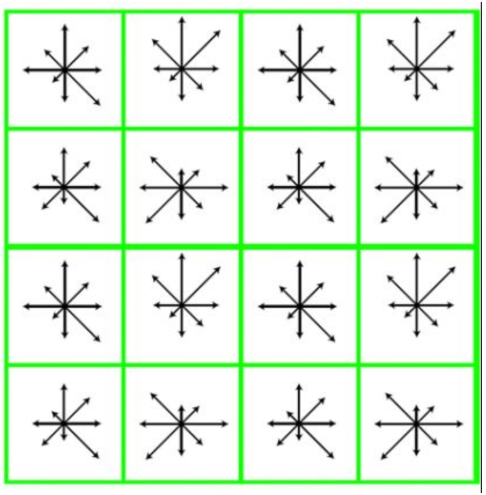
- Find Gradients of image patch
- Get Dominant Orientation using 36 bins and applying gaussian weighting.
- Highest Peak is dominant orientation
- Take 16x16 region around Keypoint and rotate to dominant orientation
- Divide into 16 sub-blocks of 4x4 and create orientation histogram with 8 bins
- Stack histogram to get 128 Dimensional Descriptor



Histogram Orientation

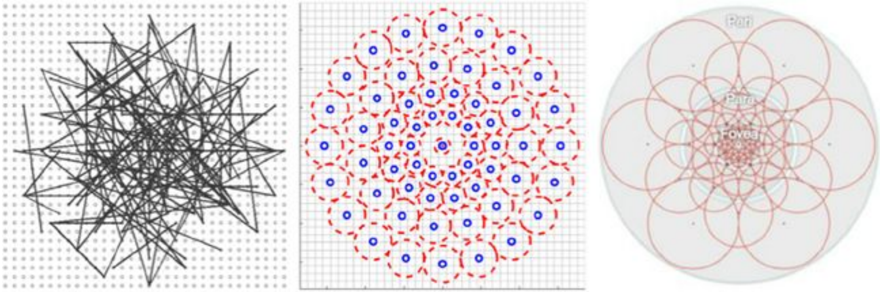


Dominant Orientation



128 Dimensional Descriptor

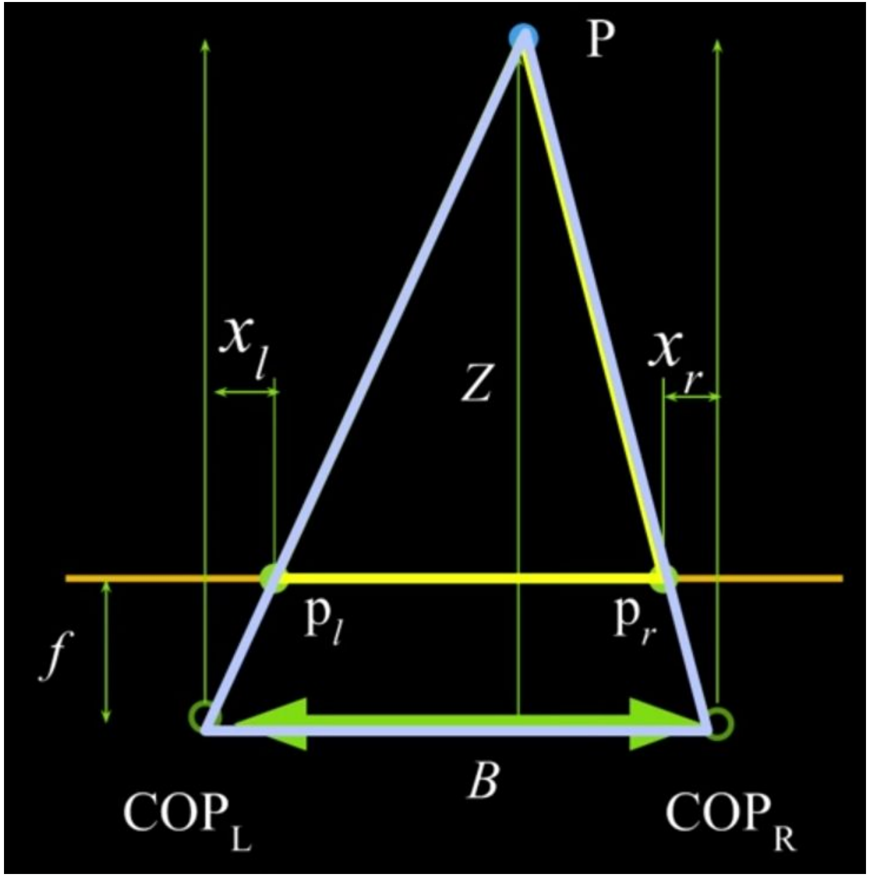
	Sampling pattern	Orientation calculation	Sampling pairs
BRIEF	None.	None.	Random.
ORB	None.	Moments.	Learned pairs.
BRISK	Concentric circles with more points on outer rings.	Comparing gradients of long pairs.	Using only short pairs.
FREAK	Overlapping Concentric circles with more points on inner rings.	Comparing gradients of preselected 45 pairs.	Learned pairs.


(i) (ii) (iii)

(i) BRIEF
(ii) BRISK
(iii) FREAK

Source: Gil

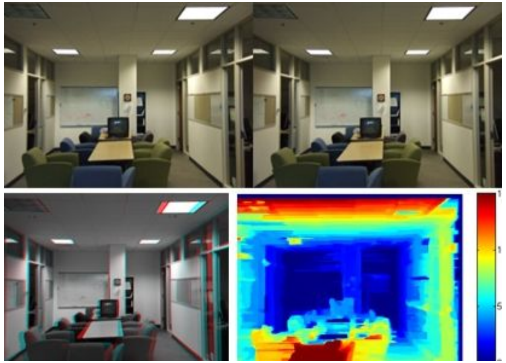
Feature Tracking and Prediction



$$\frac{B - x_l + x_r}{Z - f} = \frac{B}{Z}$$

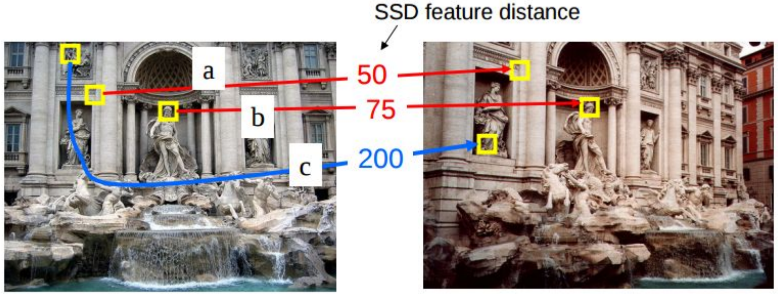
$$Z = f\frac{B}{x_l - x_r}$$

X_l - X_r = disparity



Source: Aaron Bobik

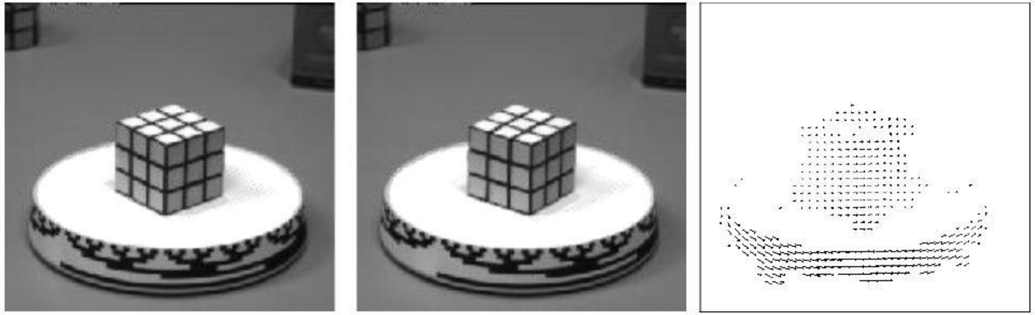
- Sum of Square differences between two descriptors f_1 and f_2
- What value of threshold to use ?



- How to resolve ambiguous matches ? Ratio distance = $SSD(f_1,f_2) / SSD(f_1,f'_2)$
- Ambiguous matches will have ratio close to 1

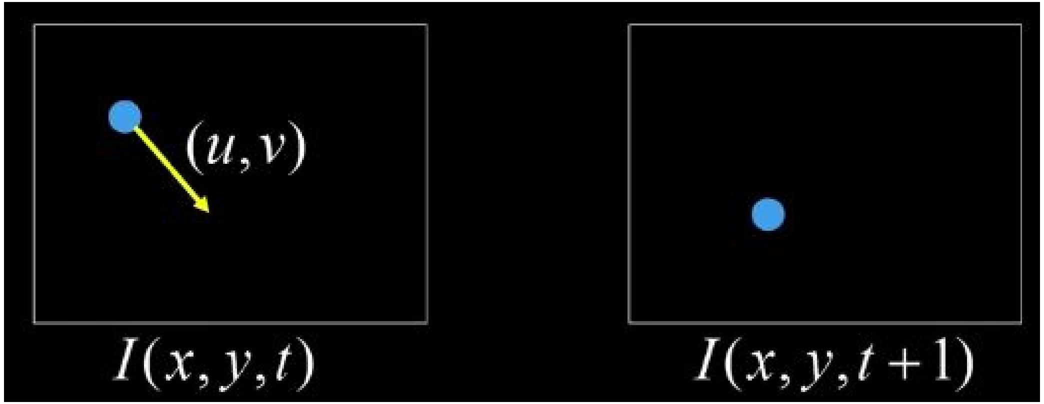
- Determine apparent motion of object in consecutive frames

- Given pixel in I(x,y,t), find nearby pixels of same intensity in I(x,y,t+1)



Brightness Constancy Constraint:

$$0 = I(x+u, y+v, t+1) - I(x, y, t)$$



Small Motion Constraint:

$$I(x+u, y+v) \approx I(x, y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v$$

Source: Aaron Bobik

- I_x I_y are space image derivatives
- I_t is time image derivative
- u v are unknowns

$$I_x u + I_y v + I_t = 0$$

Optical Flow Equation

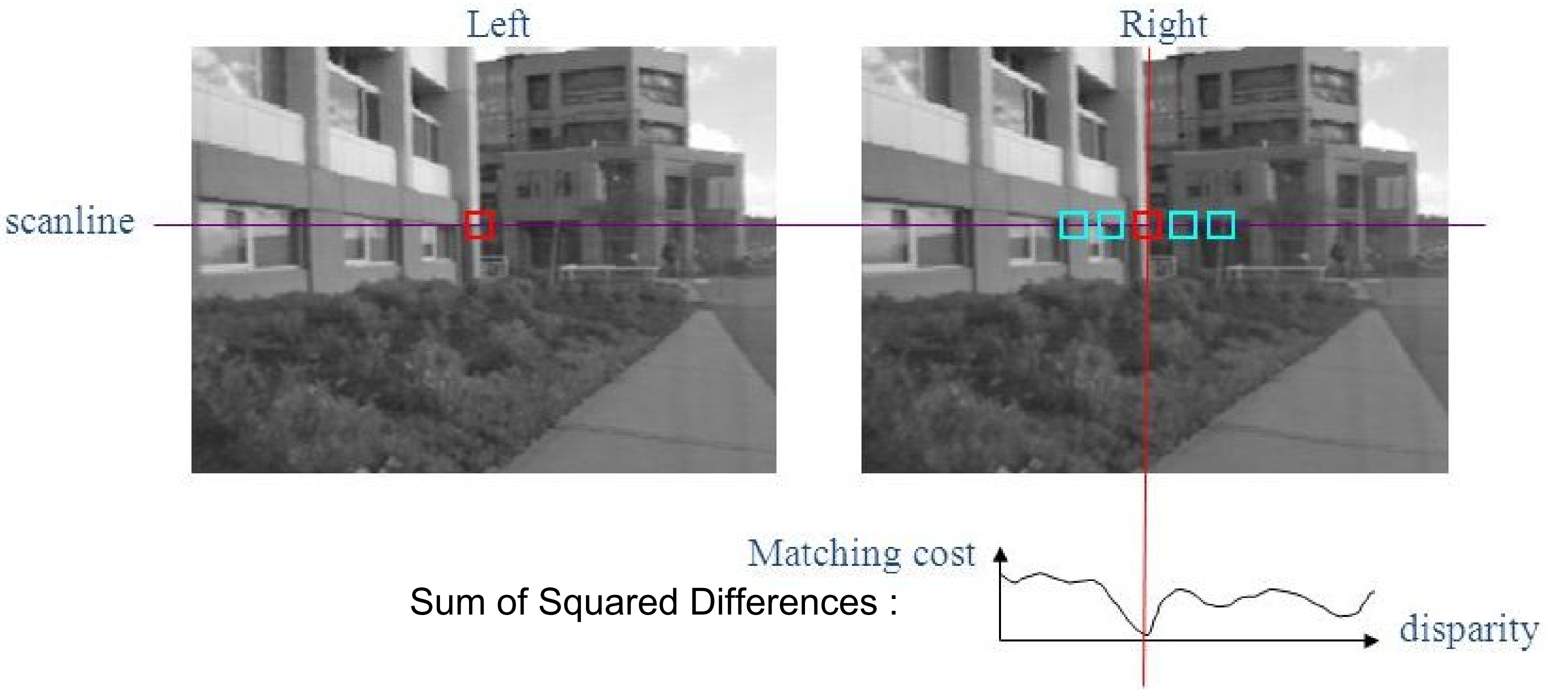
- 5x5 window gives 25 equations per pixel.

$$\underbrace{\begin{bmatrix} I_x(\mathrm{p}_1) & I_y(\mathrm{p}_1) \\ I_x(\mathrm{p}_2) & I_y(\mathrm{p}_2) \\ \vdots & \vdots \\ I_x(\mathrm{p}_{25}) & I_y(\mathrm{p}_{25}) \end{bmatrix}}_{\substack{A \\ 25\times2}} \underbrace{\begin{bmatrix} u \\ v \end{bmatrix}}_{\substack{d \\ 2\times1}} = - \underbrace{\begin{bmatrix} I_t(\mathrm{p}_1) \\ I_t(\mathrm{p}_2) \\ \vdots \\ I_t(\mathrm{p}_{25}) \end{bmatrix}}_{\substack{b \\ 25\times1}}$$

$$\underbrace{\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix}}_{A^T A} \begin{bmatrix} u \\ v \end{bmatrix} = - \underbrace{\begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}}_{A^T b}$$

Solution: $d = (A^T A)^{-1} A^T b$

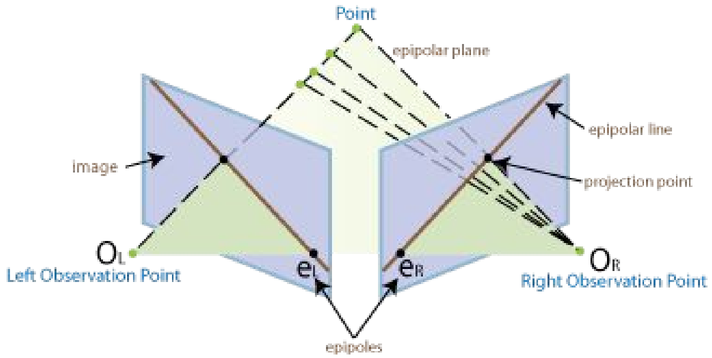
Source: Aaron Bobik



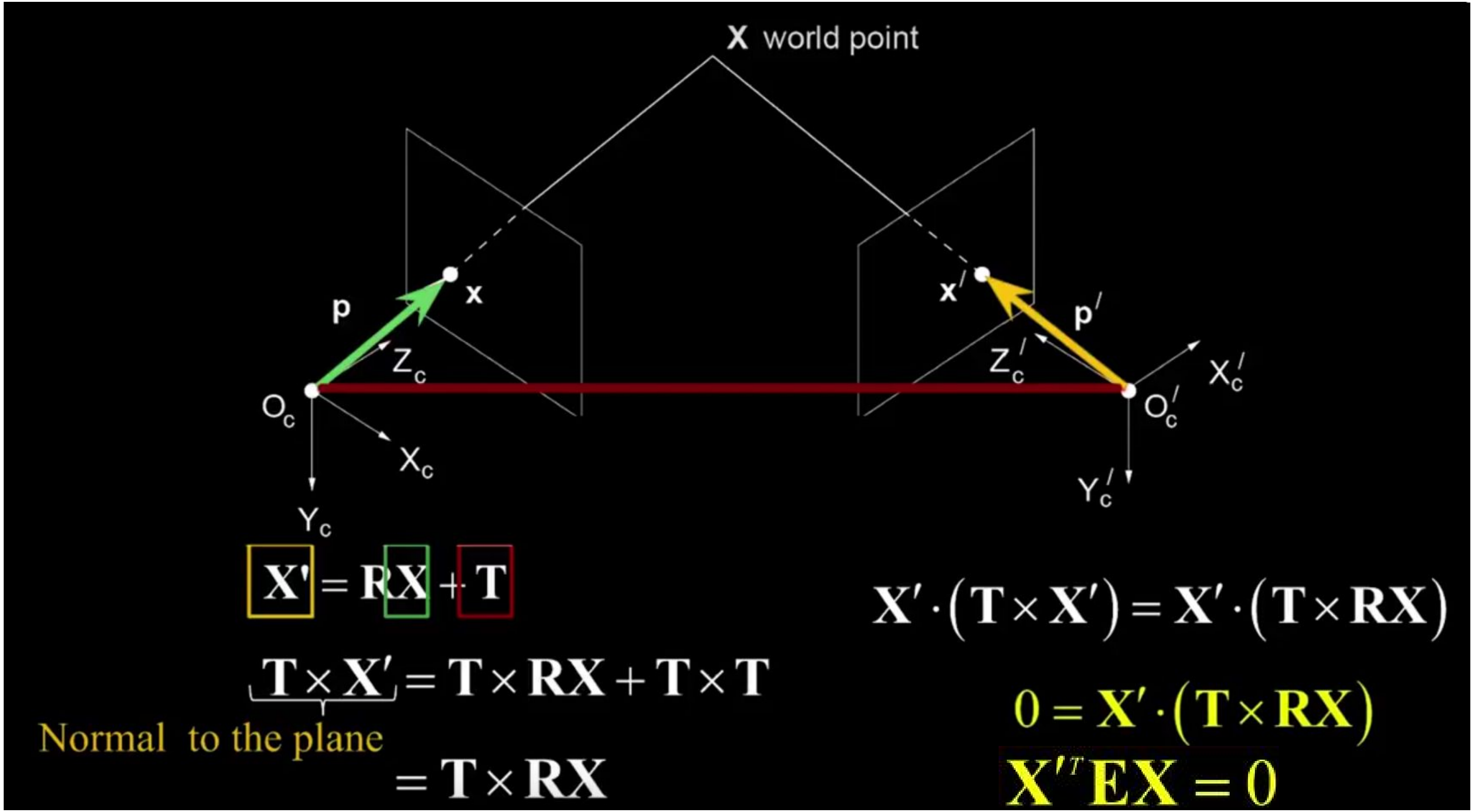
Sum of Squared Differences :

What if the image planes weren't parallel ?

- Lines project to lines in the image
- All epipolar lines intersect at epipole
- One dimensional search for correspondence
- Parallel image planes have epipole at infinity



Epipolar Geometry | Essential Matrix



Source: Aaron Bobik

$$\left(\mathbf{K}_{int,right}^{-1}\mathbf{p}_{im,right}\right)^{\mathrm{T}}\mathbf{E}\left(\mathbf{K}_{int,left}^{-1}\mathbf{p}_{im,left}\right)=0$$

$$\mathbf{p}_{im,right}^{\mathrm{T}}\underbrace{\left(\left(\mathbf{K}_{int,right}^{-1}\right)^{T}\mathbf{E}\mathbf{K}_{int,left}^{-1}\right)}\mathbf{p}_{im,left}=0$$

"Fundamental matrix": $\mathbf{F}$

$$\mathbf{p}_{im,right}^{\mathrm{T}}\mathbf{F}\mathbf{p}_{im,left}=0 \quad or \quad \mathbf{p}^{T}\mathbf{F}\mathbf{p}'=0$$

$\mathbf{l}=\mathbf{F}\mathbf{p}'$ is the epipolar *line* in the *p* image associated with p′

Calculate F from minimum 8 correspondences

$$\begin{bmatrix} u_1'u_1 & u_1'v_1 & u_1' & v_1'u_1 & v_1'v_1 & v_1' & u_1 & v_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_n'u_n & u_n'v_n & u_1' & v_n'u_n & v_n'v_n & v_n' & u_n & v_n & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = 0$$

Source: Aaron Bobik

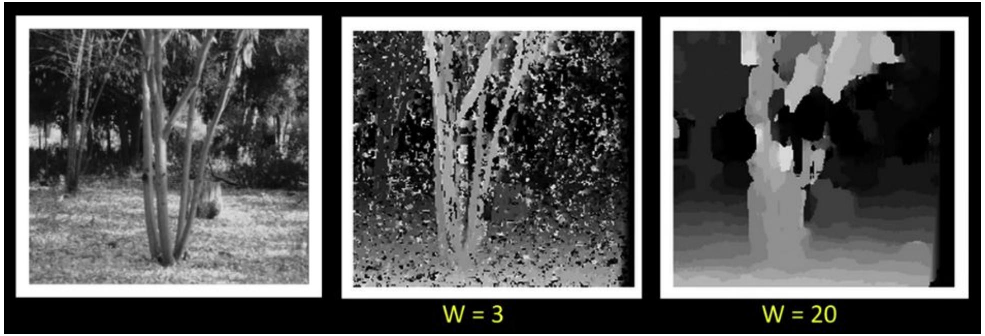
$$\mathbf{F} = UDV^T$$

$$D = \begin{bmatrix} r & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & t \end{bmatrix} \Rightarrow \hat{D} = \begin{bmatrix} r & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

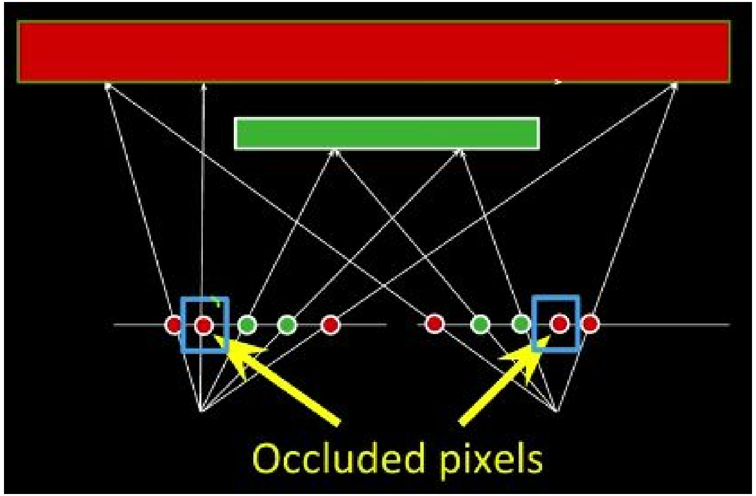
$$\hat{\mathbf{F}} = U\hat{D}V^T$$

Source: Aaron Bobik

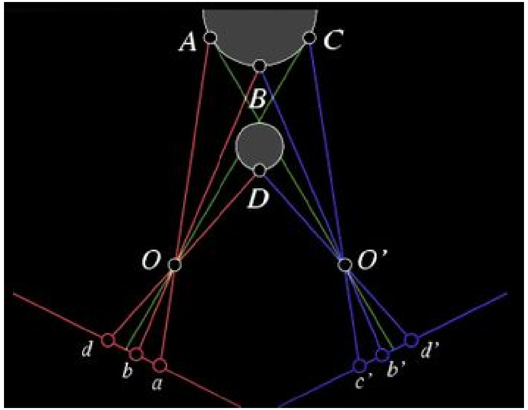
Window Size



Occlusion



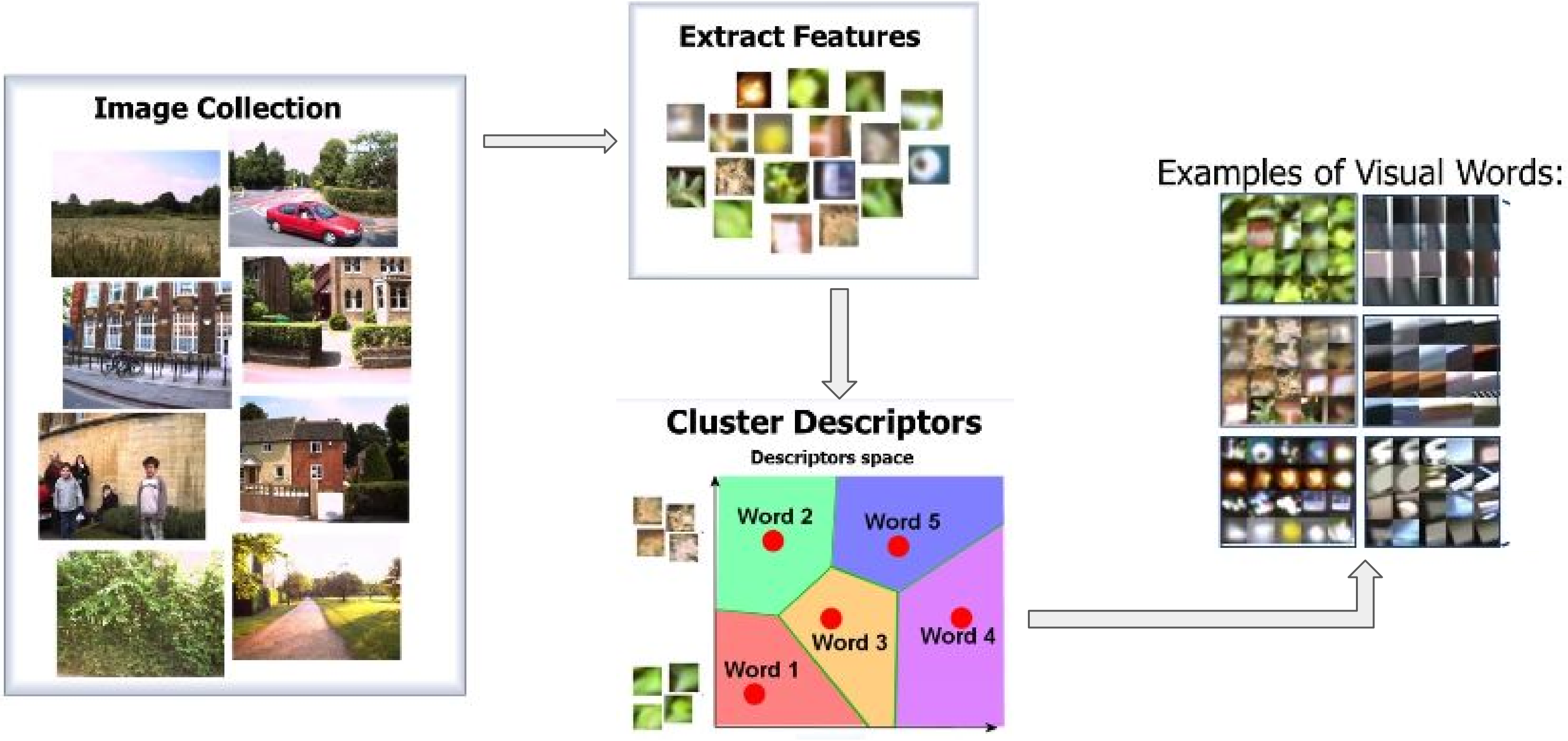
Narrow occluding surface (Ordering problems)



Source: Aaron Bobik

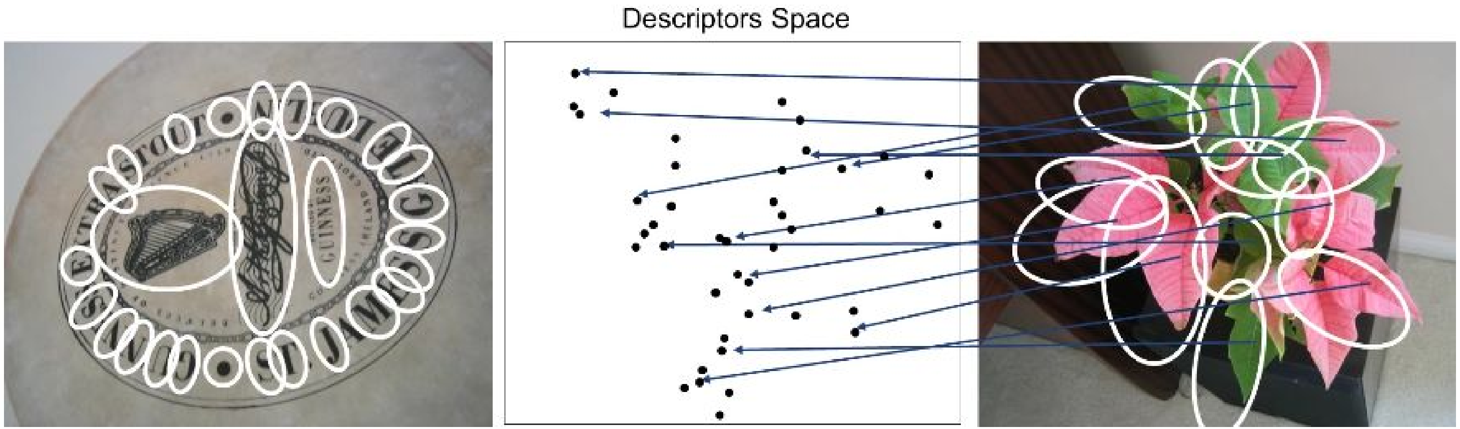

QUESTIONS?

Bag of Words

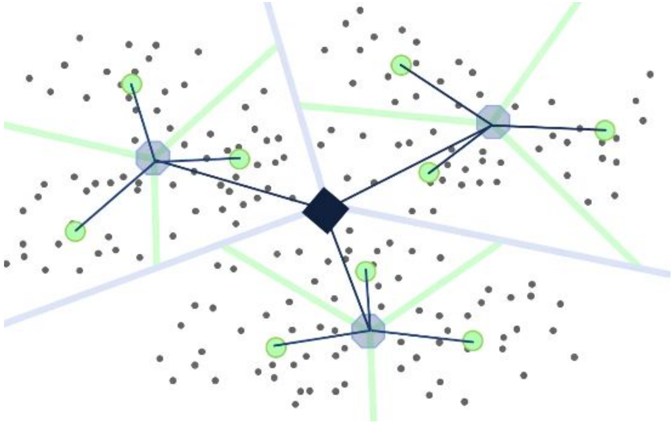


Source: Margarita Chli

Extract Features using SIFT descriptor

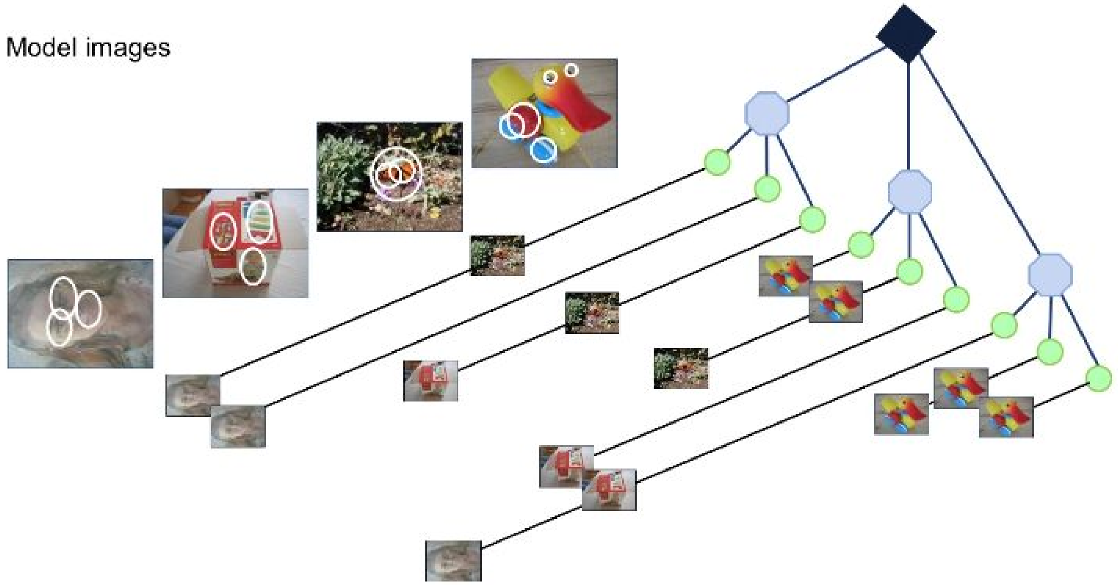


Hierarchical Clustering to get Visual Words



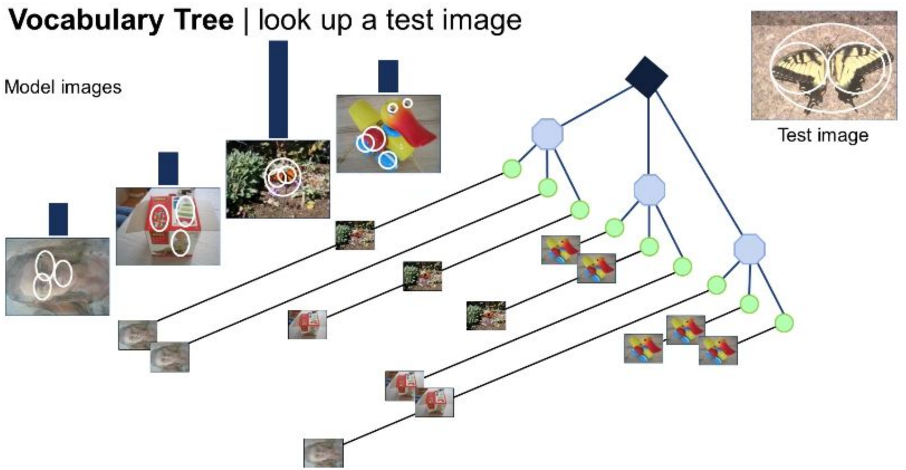
Source: Margarita Chli

- Pass each image through Vocabulary Tree
- Get Visual words based on features of image
- Use inverted file index (Visual word to image mapping) for quick test time



Source: Margarita Chli

- Extract Features from Test Image
- Find corresponding Visual Words
- Determine closest match



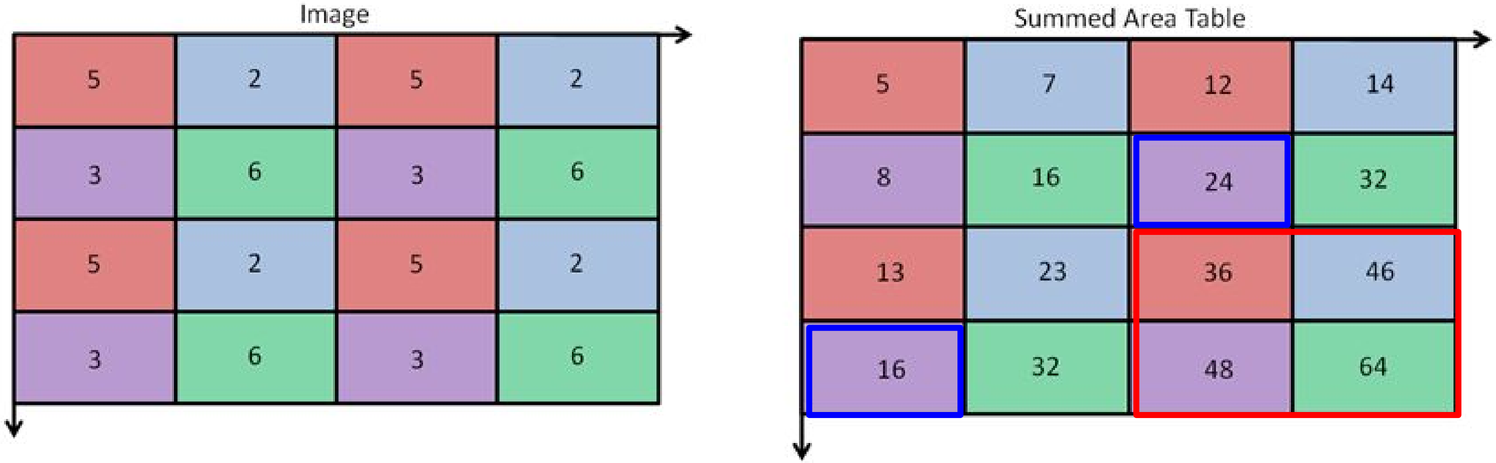
- term frequency: frequency of word w_i in image j: $tf_{ij} = \frac{n_{i,j}}{\sum_k n_{k,j}}$
- inverse document frequency: $idf_i = \log \frac{|D|}{|\{d : w_i \in d\}|}$ ← No. all images (documents) / ← No. all images containing w_i
- tf-idf of word w_i in image j is: $= tf_{ij} \cdot idf_i$

Source: Margarita Chli

Read and Summarize:

1) SURF Detector and Descriptor

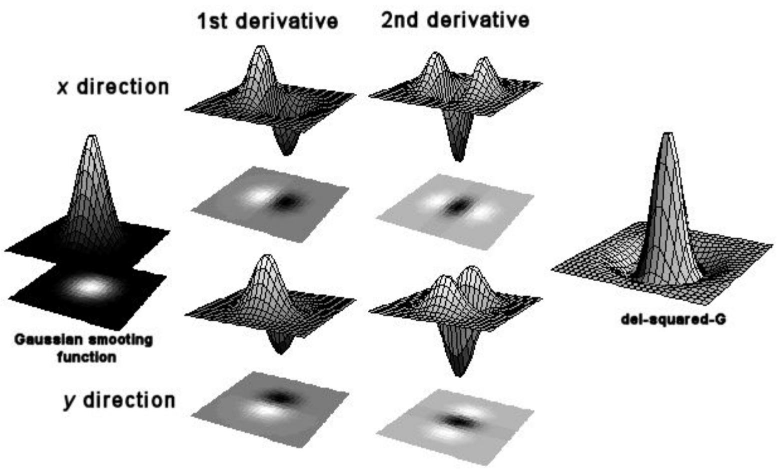
2) ORB Detector and Descriptor



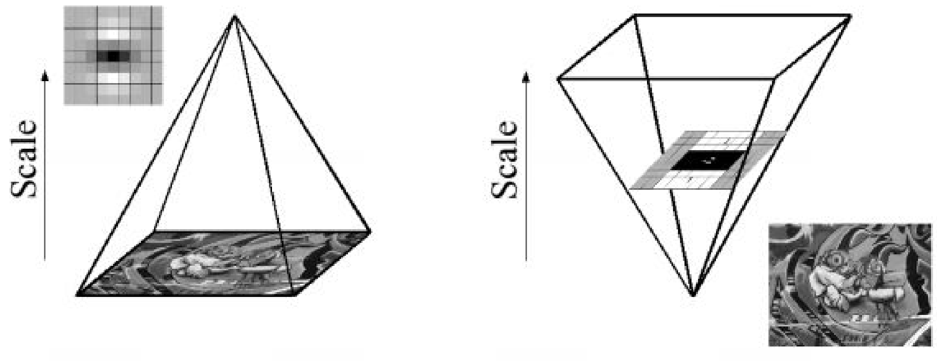
Create: 16 + 12 - 7 + 3 = 24
Create: 16 + 23 - 13 + 6 = 32

Query: 64 + 16 - 32 - 32 = 16
same as 6+5+3+2 = 16

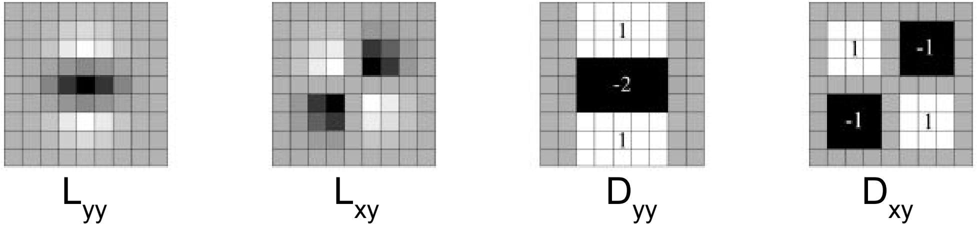
SURF Detector | Speeded Up Robust Features



Gaussian Partial Derivatives



Scale space analysis

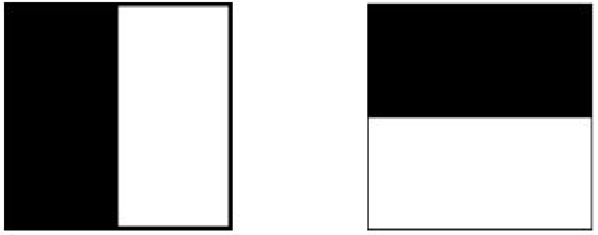


Box Filter Approximations

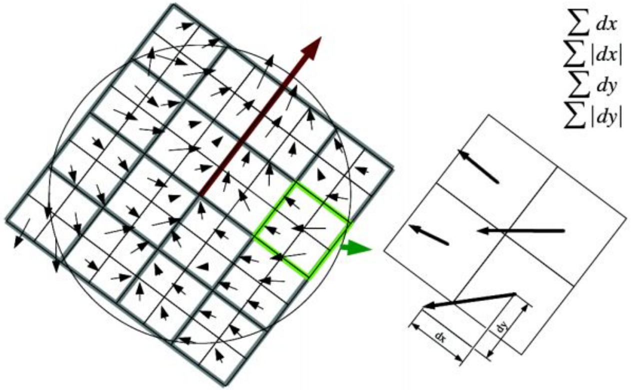
$$\mathcal{H}(\mathbf{x},\,\sigma) = \begin{bmatrix} L_{xx}(\mathbf{x},\,\sigma) & L_{xy}(\mathbf{x},\,\sigma) \\ L_{xy}(\mathbf{x},\,\sigma) & L_{yy}(\mathbf{x},\,\sigma) \end{bmatrix}$$

$$\det(\mathcal{H}_{\text{approx}}) = D_{xx}D_{yy} - (wD_{xy})^2.$$

Hessian Matrix and Determinant

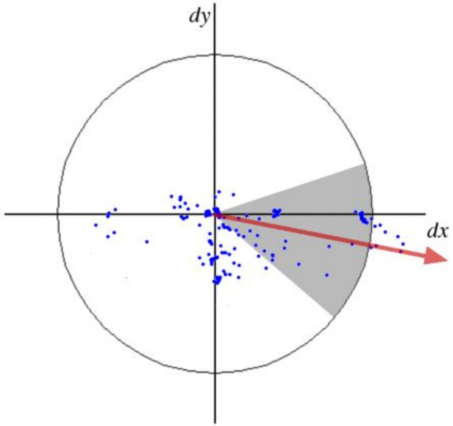


Haar Wavelets

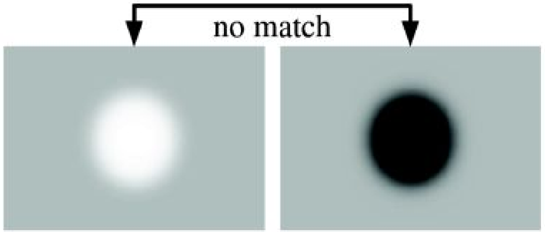


SURF Descriptor

- Each patch is 4 dimensional vector
- 16 Patches give 64 Dimensional Vector
- Sign of Hessian Matrix for black vs white blobs



Dominant Orientation

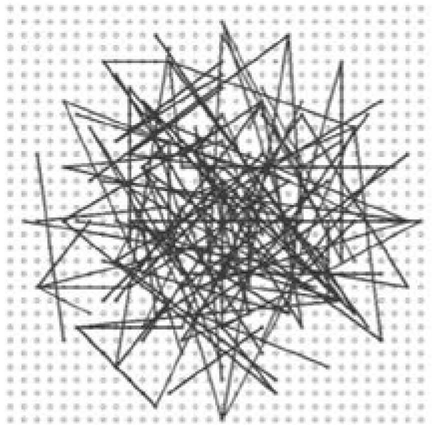


Black vs White blobs

- FAST corner Detector
- Harris Corner Measure
- FAST detected at multiple levels in the Pyramid for Scale Invariance

BRIEF: Binary Robust Independent Elementary Features

- Random Selection of pairs of Intensity Values
- Fixed sampling Pattern of 128, 256 or 512 pairs
- Hamming Distance to compare descriptors (XOR)



ORB Descriptor |

$$m_{pq} = \sum_{x,y} x^p y^q I(x, y)$$

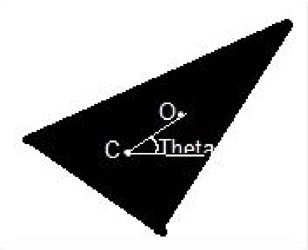
Patch moments

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$$

Center of Mass

$$\theta = \text{atan2}(m_{01}, m_{10})$$

Orientation



Angle Calculation

Learning the paris

1. Run each test against all training patches.
2. Order the tests by their distance from a mean of 0.5, forming the vector T.
3. Greedy search:
   (a) Put the first test into the result vector R and remove it from T.
   (b) Take the next test from T, and compare it against all tests in R. If its absolute correlation is greater than a threshold, discard it; else add it to R.
   (c) Repeat the previous step until there are 256 tests in R. If there are fewer than 256, raise the threshold and try again.

- 300K Keypoints
- 205590 Possible Tests
- 256 dimensional descriptor