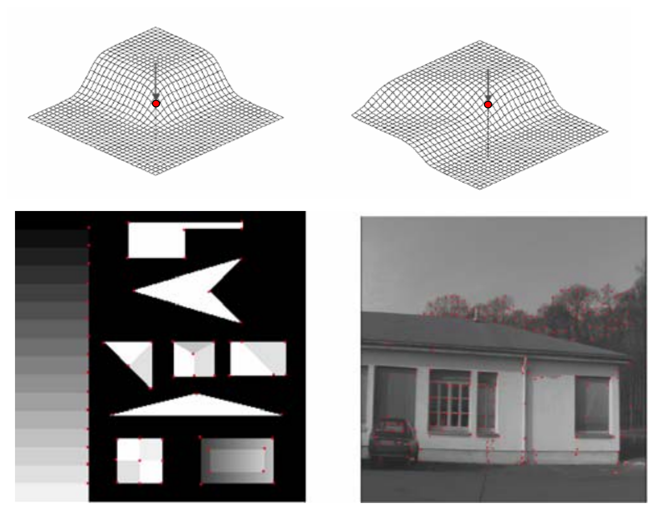
CAP6411
Computer Vision Systems
Lecture 5

Alper Yilmaz
Office: CSB 250
Email: yilmaz@cs.ucf.edu
Web: http://www.cs.ucf.edu/courses/**cap6411/cap6411**/spring2006

Recap
Interest point detection



Recap
Possible Approaches

- Based on brightness of images
  - Usually image derivatives
- Based on boundary extraction
  - First step edge detection
  - Curvature analysis of edges

Recap
Harris Corner Detector

- Corner point can be recognized in a window
- Shifting a window in any direction should give a large change in intensity

$$E(u,v)=\sum_{x,y}\underbrace{w(x,y)}_{\text{window function}}\ [\underbrace{I(x+u,y+v)}_{\text{shifted intensity}}-\underbrace{I(x,y)}_{\text{intensity}}]^2$$

$$E(u,v)=\begin{pmatrix}u & v\end{pmatrix}M\begin{pmatrix}u\\ v\end{pmatrix} \qquad M=\sum_{x,y}w(x,y)\begin{bmatrix}I_xI_x & I_xI_y\\ I_xI_y & I_yI_y\end{bmatrix}$$

$$R=\det M-k\left(traceM\right)^2$$

C.Harris, M.Stephens. “A Combined Corner and Edge Detector”. 1988

SUSAN Detector

- Proposed by Smith and Brady in 1995
- SUSAN stands for Smallest "**U**nivalue **S**egment **A**ssimilating **N**ucleus (USAN)"
- It doesn't use any derivatives
- It is based on the fact that each point within an image is associated with it a local area of comparable brightness

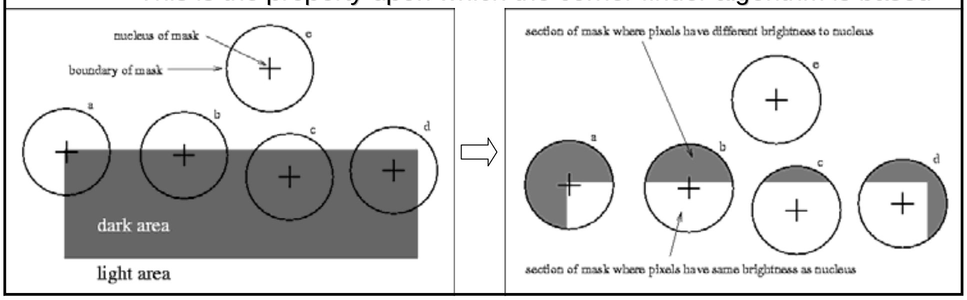
S.M. Smith. SUSAN - *a new approach to low level image processing*. Internal Technical Report TR95SMS1, Defence Research Agency, Chobham Lane, Chertsey, Surrey, UK, 1995.

Principle

- Considered in a circular mask around a pixel
- Comparison of intensity in a neighborhood
  - Area with similar intensities is called **USAN**
- Repeat the procedure for each pixel

The SUSAN Detector

- USAN area varies with respect to **features** of the image
- USAN area
  - is maximum within the rectangular area
  - falls to a **minimum at an edge**
  - smaller value corresponding to a **local minimum at a corner**
- This is the property upon which the corner finder algorithm is based



Algorithm

1. Determine a circular mask
   - Typically 37 pixels around each pixel (nucleus)
2. Calculate the brightness difference between each pixel in the mask with its nucleus

$$c(r, r_0) = \begin{cases} 1 & \text{if } |I(r) - I(r_0)| \le t \\ 0 & \text{otherwise} \end{cases}$$

3. Sum the number of pixels with similar intensity levels to that of the nucleus

$$n(r_0) = \sum_r c(r, r_0)$$

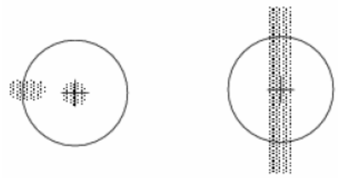
Algorithm

4. Compare *n* with *g*, the *geometric threshold* which is set to half of the maximum value that *n* can be ($n_{max}/2$)
5. At a perfect corner the USAN area will always be less than half the size of the mask area, and will be a local minimum

$$R(r_0) = \begin{cases} g - n(r_0) & \text{if } n(r_0) < g \\ 0 & \text{otherwise} \end{cases}$$

Problems

- Strong edges and noise results in false detection
- (Left figure) The USAN is not continuous. Obviously nucleus is not a corner, even though the function shows it is the local maxima.
- (Right figure) Nucleus lies in a long thin area, which depicts USAN is also very small. However, the value is high, which contradicts the fact that the point in question is not a corner.

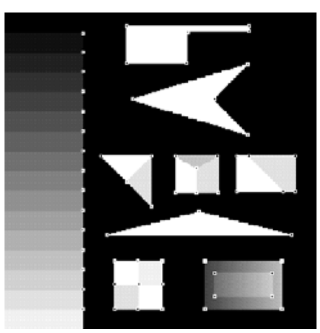


Improving SUSAN Detector

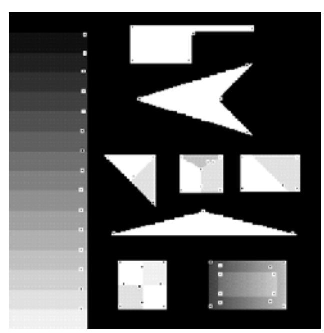
- Two rules
  - Find centroid of USAN area and distance from nucleus. point cannot become a corner if the distance is small.
  - One of the pixels on the line connecting centroid to center of circular region can be a corner

Benefits

- Accuracy, speed and localization



Output of the **SUSAN** corner detector (**t**=10) given the test image. (0.3 sec)



Output of the **Plessey** corner detector (**o** =2.0) given the test image. (3.5 sec)

SIFT - Key Point Extraction

- Stands for scale invariant feature transform
- Patented by university of British Columbia
- Similar to the one used in primate visual system (human, ape, monkey, etc.)
- Transforms image data into scale-invariant coordinates

D. Lowe. *Distinctive image features from scale-invariant key points.*, International Journal of Computer Vision 2004.

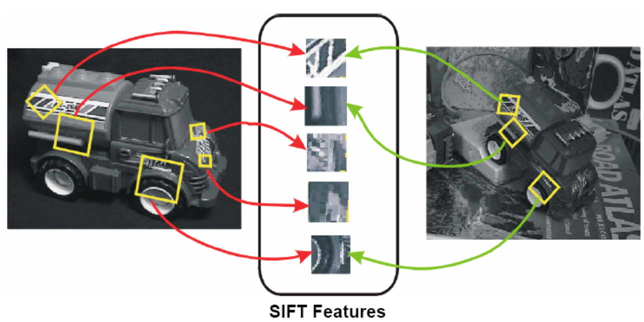
Goal

- Extracting distinctive invariant features
  - Correctly matched against a large database of features from many images
- Invariance to image scale and rotation
- Robustness to
  - Affine distortion,
  - Change in 3D viewpoint,
  - Addition of noise,
  - Change in illumination.

Advantages

- **Locality:** features are local, so robust to occlusion and clutter
- **Distinctiveness:** individual features can be matched to a large database of objects
- **Quantity:** many features can be generated for even small objects
- **Efficiency:** close to real-time performance
- **Extensibility:** can easily be extended to wide range of differing feature types, with each adding robustness

Invariant Local Features

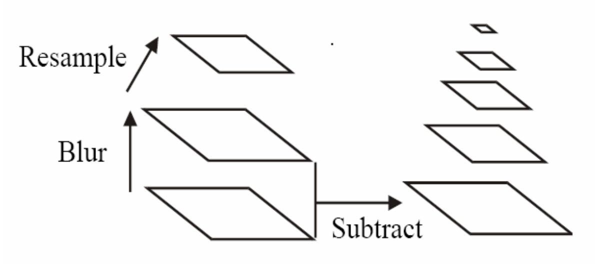


Steps for Extracting Key Points

- Scale space peak selection
  - Potential locations for finding features
- Key point localization
  - Accurately locating the feature key
- Orientation Assignment
  - Assigning orientation to the keys
- Key point descriptor
  - Describing the key as a high dimensional vector

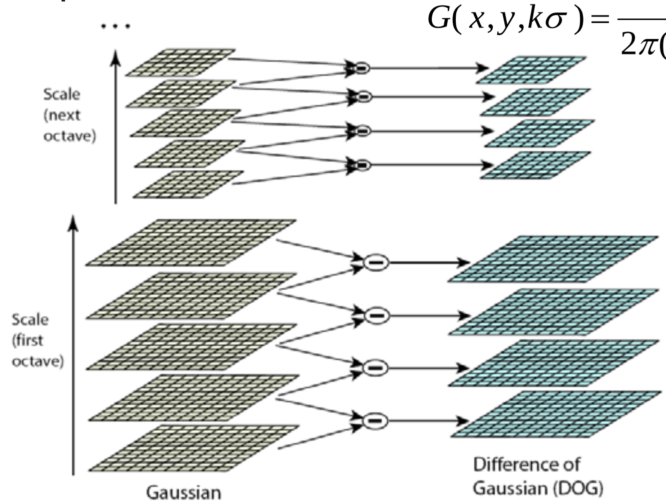
Building a Scale Space

- All scales must be examined to identify scale-invariant features
- An efficient function is to compute the Laplacian Pyramid (Difference of Gaussian) (Burt & Adelson, 1983)



Building a Scale Space

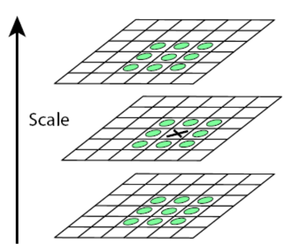
$$G(x,y,k\sigma) = \frac{1}{2\pi(k\sigma)^2} e^{-(x^2+y^2)/2k^2\sigma^2}$$



k is the number of chosen scale in each octave

Scale Space Peak Detection

- Compare a pixel (**X**) with 26 pixels in current and adjacent scales (Green Circles)
- Select if larger/smaller than all 26 pixels
- Large number of extrema, computationally expensive
  - Detect the most stable subset with a coarse sampling of scales



Key Point Localization

- Candidates are chosen from extrema detection

original image

extrema locations

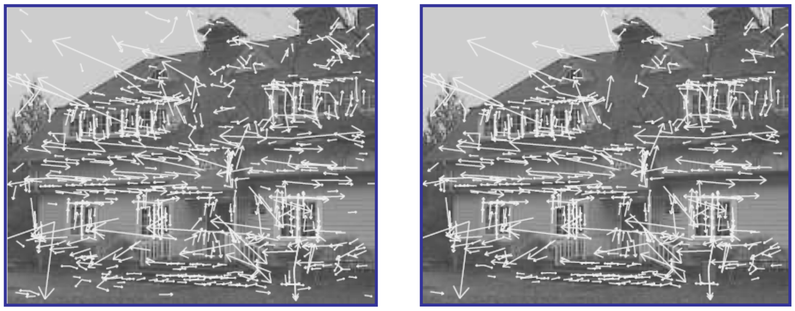
Initial Outlier Rejection

- Low contrast candidates
- Poorly localized candidates along an edge
- Taylor series expansion of DOG, **D**.

$$D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}}\mathbf{x} + \frac{1}{2}\mathbf{x}^T \frac{\partial^2 D}{\partial \mathbf{x}^2}\mathbf{x} \qquad \mathbf{x} = (x, y, \sigma)^T$$

- Minima or maxima is located at $\hat{\mathbf{x}} = -\frac{\partial^2 D}{\partial \mathbf{x}^2}^{-1} \frac{\partial D}{\partial \mathbf{x}}$
- Value of $D(\mathbf{x})$ at minima/maxima must be large, $|D(x)|>th$.

Initial Outlier Rejection



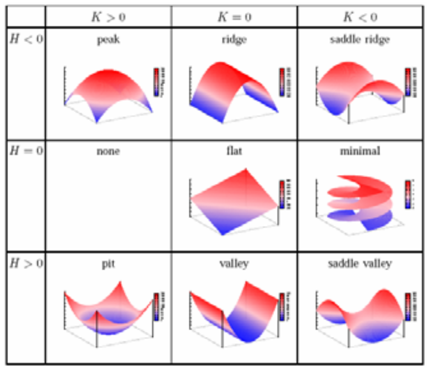
from 832 key points to 729 key points, th=0.03.

Further Outlier Rejection

- DOG has strong response along edge
- Assume DOG as a surface
  - Compute principal curvatures (PC)
  - Along the edge one of the PC is very low

Principal Curvatures

- Notion from differential geometry
- They measure maximum and minimum bending of a surface
- Principal curvatures can be computed from Hessian of the surface



Hessian Matrix

- Jacobian matrix of the derivatives of a function is called Hessian

$$H\left(f(x_1,\ldots,x_n)\right)=\begin{bmatrix} \frac{\partial^2 f}{\partial x_1^{\,2}} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_1^{\,2}} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^{\,2}} \end{bmatrix}$$

Further Outlier Rejection

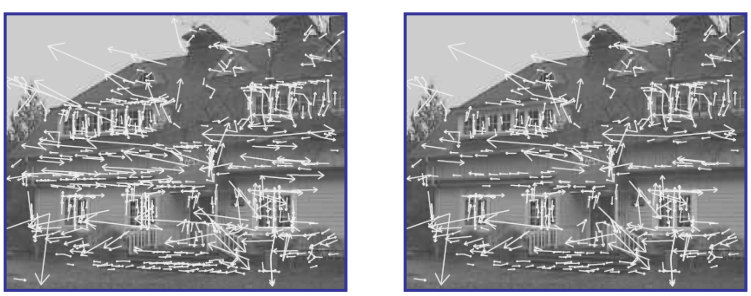
- Analogous to Harris corner detector
- Compute Hessian of D

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

- Find eigenvalues λ_1, λ_2
- Remove outliers by evaluating

$$\frac{\mathrm{Tr}(\mathbf{H})^2}{\mathrm{Det}(\mathbf{H})} < \frac{(r+1)^2}{r}$$

Further Outlier Rejection



from 729 key points to 536 key points.

Orientation Assignment

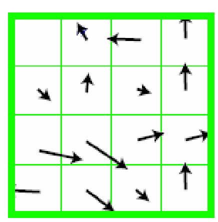
- To achieve rotation invariance
- Computed from Gaussian smoothed image at the scale of key point (x,y)
- Compute central derivatives, gradient magnitude and direction

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

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

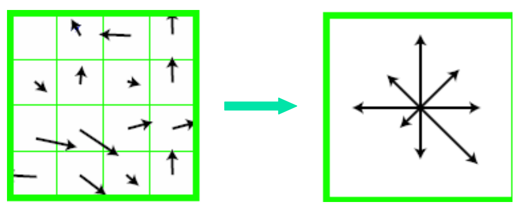
Orientation Assignment

- Create a weighted direction histogram in a neighborhood of a key point (36 bins)
- Weights are
  - Gradient magnitudes
  - Spatial gaussian filter with σ=1.5 x <scale of key point>



Orientation Assignment

- Select the peak as direction of the key point
- Introduce additional key points (same location) at local peaks of the histogram with different directions



Local Image Descriptors at Key Points

- Possible descriptor
  - Store samples in the neighborhood
  - Sensitive to lighting changes, 3D object transformation
- Use of gradient orientation histograms
  - Robust representation

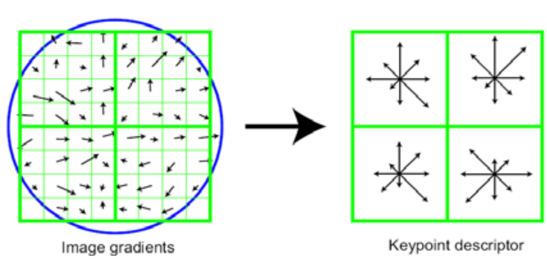
Similarity to IT cortex

- Complex neurons respond to a gradient at a particular orientation.
- Location of the feature can shift over a small receptive field.
- Edelman, Intrator, and Poggio (1997) hypothesized that the function of the cells allow for matching and recognition of 3D objects from a range of view points.
- Experiments show better recognition accuracy for 3D objects rotated in depth by up to 20 degrees

33

Extraction of Local Image Descriptors at Key Points

- Compute relative orientation and magnitude in a 16x16 neighborhood at key point
- Form weighted histogram for 4x4 regions
  - Weight by magnitude and spatial Gaussian
  - Example for 8x8 to 2x2 descriptors



Extraction of Local Image Descriptors at Key Points

- Store numbers in a vector
- Normalize to unit vector (**UN**)
  - Illumination invariance (affine changes)
- For non-linear intensity transforms
  - Bound **UN** items to maximum .2
  - Renormalize to unit vector

Key point matching

- Match the key points against a database of that obtained from training images.
- Find the nearest neighbor i.e. a keypoint with minimum Euclidean distance.
  - Efficient Nearest Neighbor matching
  - Best bin first algorithm: a modification of k-d tree search
    - Looks at ratio of distance between best and 2^{nd} best match

36

1st Programming Project Assignment

- Implementation of SIFT key point detector
- Due date 27 February, 2006
- Deliverables:
  - Source code on a CD,
  - Project report discussing problems encountered, where the algorithm fails,
  - Images of intermediate steps
- Classroom demo is required on an unknown set of images
- Program should display computed features
- A comparison with the author's program (only executable) should show if your algorithm fails
- Look at the IJCV paper by D. Lowe