

1. Computer vision

Developing computational models and algorithms to interpret digital images, understand the visual world we live in.

2. Application examples

Face detection, Human pose estimation, Earth viewers, Photo browsing, Optical character recognition (OCR), Driver assistance systems, Special effects: Shape capture, Special effects: Motion capture, Sports, Vision in space, Robotics, Medical imaging

3. Artistic cues

4. Human visual cues

Stereo parallax, Motion parallax, Shadows, Convergence, Context

We as humans use many different cues to interpret what is in an image.

We use information on what we regard as being a plausible and meaningful interpretation.

This is necessary, because an image is a 2D projection of a complex 3D world.

5. Pinhole camera

Captures pencil of rays – all rays through a single point

Projection rays are straight lines

The point is called center of projection (focal point)

The image is formed on the image plane

Two equivalent projections: positive, negative

6. Perspective projection

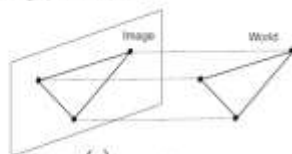
$$x' = f' \cdot \frac{x}{z}$$
$$y' = f' \cdot \frac{y}{z}$$

7. Orthographic projection

Special case of perspective projection

▪ Distance from center of projections to image plane is infinite

▪ Also called "parallel projection"



▪ What's the projection matrix?

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \Rightarrow (x, y)$$

8. Coordinate transformations

Extrinsic camera transformation takes world into camera coordinates. Intrinsic camera transformation describes the image formation process.

Extrinsic: Described as a linear transformation, Using homogeneous coordinates, Combination of rotation and translation

Intrinsic: linear transformation + perspective division

9. Spatial sampling

Images arriving at our CCD or CMOS sensor are spatially discrete with individual pixels. The image sensor performs a "sampling" of this function.

Quantize the values per channel (e.g. 8 bit)

10. Vanishing points

Parallel lines converge at a vanishing point

Each direction in space has its own vanishing point

But parallels also parallel to the image plane remain parallel

All directions in the same plane have vanishing points on the same line

11. Perspective distortions

The exterior columns appear bigger

The distortion is not due to lens flaws

Problem pointed out by Da Vinci

12. Homogeneous coordinates

$$(x, y, z) \rightarrow (f^x \frac{x}{z}, f^y \frac{y}{z}) \quad \text{division by } z \text{ is nonlinear}$$

Trick: add one more coordinate

$$(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (x, y, z) \Rightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Converting from homogeneous coordinates

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w) \quad \begin{bmatrix} x \\ y \\ z \end{bmatrix} \Rightarrow (x/z, y/z)$$

13. Projection matrix


Projection is a matrix multiplication in homogeneous coordinates:

$$\begin{pmatrix} f' & 0 & 0 & 0 \\ 0 & f' & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} f'x \\ f'y \\ z \end{pmatrix} \Rightarrow (f' \frac{x}{z}, f' \frac{y}{z}) \quad \text{divide by the third coordinate}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \text{Camera to pixel coord. trans. matrix (3x3)} \\ \text{3D point (3x1)} \end{bmatrix} = \begin{bmatrix} \text{Perspective projection matrix (3x3)} \\ \text{3D point (3x1)} \end{bmatrix} = \begin{bmatrix} \text{World to camera coord. trans. matrix (3x4)} \\ \text{3D point (3x1)} \end{bmatrix}$$

14. Calibration matrix / camera intrinsic

Principal point offset



principal point: (p_x, p_y)

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \Rightarrow \begin{pmatrix} fX + Zp_x \\ fY + Zp_y \\ Z \end{pmatrix} = \begin{bmatrix} f & 0 & p_x & 0 \\ 0 & f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$K = \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix} \quad \text{calibration matrix} \quad P = K[I|0]$$

Intrinsic parameters

Principal point coordinates

Focal length

Pixel magnification factors

Skew (non-rectangular pixels)

Radial distortion

$$K = \begin{bmatrix} m_x & 0 & p_x \\ 0 & m_y & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & 0 & p_x \\ 0 & \alpha_y & p_y \\ 0 & 0 & 1 \end{bmatrix}$$

15. Extrinsic

Extrinsic parameters

Rotation and translation relative to world coordinate system

16. Linear camera calibration

Given n points with known 3D coordinates X_i and known image projections x_i , estimate the camera parameters

Once we've recovered the values of the camera matrix, we still have to figure out the intrinsic and extrinsic parameters

$$P = [M|m] = [KR \mid KR\tilde{C}]$$

First split the projection matrix into a 3x3 matrix and a 3x1 vector, Next, decompose M into upper triangular part K (calibration) and orthonormal part R (rotation)

Finally, find C as the nullspace of P by means of SVD.

17. Homogeneous least squares

46, 47, 48, 49

18. Estimating projection matrix

42, 43, 44, 45

19. Aperture

Why not make the aperture as small as possible?

Less light gets through and Diffraction effects...

aperture does not affect the visibility, but only the amount of light

20. Thin lens formula

62, 63

21. Depth of field

19

22. Field of view

23

23. Camera artifacts

A digital camera replaces film with a sensor array

Each cell in the array is light-sensitive diode that converts photons to electrons

Charge Coupled Device (CCD)

Complementary metal oxide semiconductor (CMOS)

CCD: transports the charge across the chip and reads it at one corner of the array.

An analog-to-digital converter (ADC) then turns each pixel's value into a digital value by measuring the amount of charge at each photosite and converting that measurement to binary form.

CMOS: uses several transistors at each pixel to amplify and move the charge using more traditional wires.

The CMOS signal is digitized right away, so it needs no separate ADC. Also often faster (video applications)

24. Color cameras

25. Bayer pattern

Color sensing in camera: Color filter array

Estimate missing components from neighboring values (demosaicing)

Bilinear Interpolation

26. Color sensors

The cause of color moiré: Fine black and white detail in image misinterpreted as color information

Color sensing in camera: Prism

Color sensing in camera: Foveon X3

Color sensing in camera: X-Trans

27. Linear filtering

Filtering to Reduce Noise. Intuition: Averaging noise reduces its effect

Assumption: The pixel's neighborhood contains information about its intensity

Average Filter: replaces each pixel with an average of its neighborhood, mask with positive entries that sum to 1

Gaussian Averaging: Weighs nearby pixels more than distant ones

Smoothing kernel proportional to

$$G_{\sigma}(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

28. Convolution kernels

filter (kernel)

convolution is linear - associative and commutative

29. Image smoothing

Smoothing with a Gaussian

We can also sharpen an image by amplifying what smoothing removes

30. Separable filters

The box filter and the Gaussian filter are separable

$$\frac{1}{9} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \frac{1}{3} \cdot \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} * \frac{1}{3} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

31. Boundary handling

Boundary handling strategies: zero, warp, clamp, mirror

32. Median filter

Non-linear filtering

Replace each pixel with the median in a neighborhood around it

33. Morphology

Binary Images

Perform convolution with a “structuring element” s

$$t(f, t) = \begin{cases} 1 & \text{if } f \geq t, \\ 0 & \text{else,} \end{cases}$$

34. Gaussian pyramid

7

35. Aliasing

Cannot shrink an image by taking every second pixel. High frequencies cannot be represented anymore.

If we do that anyway, characteristic errors appear: Spatial frequencies are misinterpreted (aliasing). Typically, small phenomena look bigger; fast phenomena can look slower.

Common examples: Wagon wheels rolling the wrong way in movies. Checkerboards misrepresented in ray tracing. Striped shirts look strange on color television.

36. Edge detection

Correspond to fast changes, where the magnitude of the derivative is large.

Based on 1st derivative: smooth with Gaussian, calculate derivative, finds its local optima

Simplification: 25

Gradient direction is perpendicular to edge. Gradient magnitude measures edge strength

2D Edge Detection: Calculate derivative use the magnitude of the gradient. 37

The scale of the smoothing filter affects derivative estimates, and also the semantics of the edges recovered.

Note: strong edges persist across scales

37. Canny edge detector

Optimal Edge Detection

Assume:

linear filtering, additive i.e.. Gaussian noise

Edge detection should have:

good detection: filter responds to edge, not to noise. good localization: detected edge near true edge.

single response: one per edge

Then: Optimal detector is approximately derivative of Gaussian [Canny 1986]

Detection/localization tradeoff: more smoothing improves detection and hurts localization

38. Non-maximum suppression

Check if pixel is local maximum along gradient direction

choose the largest gradient magnitude along the gradient direction, requires checking interpolated pixels

39. Laplacian

46, 48, 49

40. Laplacian pyramid

52

The Laplacian pyramid provides a simple frequency decomposition into sub bands.

Image sharpening by amplifying high-frequency components

41. Template-based matching

Search every image region (at every scale). Compare each template; chose the best match.

View-Based Methods: Represent objects by their appearance in an ensemble of images, including different poses, illuminations, configurations of shape

Filters are Templates

Filters look like the effects they are intended to find. Filters find effects they look like. We can use filters to match a template against an image. 7

When we filter, we measure the angle (cosine of it, really) between the filter

template and the image patch.

42. SSD

Sum of Squared Differences (SSD)
$$E(I, T) = \sum_{i,j} (I(i, j) - T(i, j))^2$$

Image templates (simplest view-based method)

Storage and computation costs become unreasonable as the number of objects increases. May require very large ensemble of 'training' images.

43. Subspaces

more efficient representations.

Observation: images are not random... especially images of the same object (category) have similar appearance.

Assume images to be represented as points in a high-dimensional space (e.g., one dimension per pixel)

44. Dimensionality reduction

Find a lower dimensional representation that captures the variability in the data. Search using this low dimensional model.

Given that differences are structured, we can use 'basis images' to transform images into other images in the same space.

45. Correlation

17

46. PCA

Image dimension is too large to handle efficiently

Goal: Find the so-called principal directions, and the variance of the data along each principal direction.

28 : 41 60 : 67

Project each training image onto the low-dimensional subspace. Store the vectors of coefficients.

For each image region: Project it onto the low-dimensional subspace. Compare this to each stored coefficient vector (cheap). If the smallest distance is less than some threshold, then it is a mouth.

47. SVD

42 : 48

48. Eigen representations

49. Eigenfaces

First popular use of PCA for object recognition was for the detection and recognition of faces. 49

50. Appearance manifolds

Many objects do not have convex subspaces when one considers different poses and lighting variations.

Limitations of linear representations: PCA will not work!

51. Interest points

General idea of detecting corresponding points:

1. — Find important, i.e. "interesting" points in the images

2. — Find which interest points correspond to one another (later)

Local measure of interest point uniqueness

Compare each pixel before and after by summing up the squared differences (SSD)

11 : 13, 16 : 18

Interest point detection so far:

Based on Harris or Hessian detector

Finds interesting, i.e. discriminative points (Harris detector was the "de-facto" standard for a long time)

Used for recognition, correspondence for stereo, sparse optical flow/motion, etc.

To come: Clarify the goals of interest point detection

Distinctiveness vs. invariance to transformation

Harris & Hessian find distinctive points - but they are not invariant to scale, affine and projective transformations

52. Structure tensor

13, 14

53. Harris points

19 : 26

54. Hessian points

27

55. Invariant features

The Harris interest point detector is rotation invariant, Rotating the image leads to rotation of the gradient

Scale-invariant interest point detection

Matching images of different scales

Automatic scale selection

Scale invariant methods for feature extraction: Harris-Laplace

56. Scale space

Level of details decreases monotonically as the scale of Gaussian smoothing is increased

57. Scale selection

47 : 60

58. Harris-Laplace points

71 : 79

59. Performance evaluation

Repeatability rate: percentage of corresponding points

$$\text{repeatability} = \frac{\# \text{correspondences}}{\# \text{detected}} \cdot 100\% \quad \text{Two points are corresponding if } \frac{|A \cap B|}{|A \cup B|} > T$$

$T=60\%$

60. Local descriptors

2. — Find which interest points correspond to one another (now)

1. Compute local descriptors / features (describing the region around interest point)

2. Compare them (using some distance)

Distinctiveness, invariance, robustness, dimensionality.

Detector finds location, scale and shape of interest regions

Local descriptors are computed for interest regions

Image patches Invariant only when computed on normalized patches Distinctive & easy to implement But high-dimensional and not very robust Match descriptors using: Euclidean distance Cross correlation

Filter bank("jet") Invariant only when computed on normalized patch Responses of Gaussian derivatives
SIFT

Shape context

61. SIFT features (Scale Invariant Feature Transform)

Extraordinarily robust matching technique 40, 41, 42

62. Shape context

Cope with complex geometric transformations

Cannot compare pixel to pixel 43, 44

63. Homography

Perspective image of a plane

Views from same camera center 56, 57, 58 61 : 66

64. Coordinate normalization

65. RANSAC (random sample consensus)

73

66. Panorama stitching

estimate the homography between two overlapping images, transform one into the image plane of the other

Find images related by a homography

Extract relative rotation

Remap to a sphere

Blend colors along seams

67. Stereo

Stereo vision, stereopsis, or short stereo is the perception or measurement of depth from two projections.

Given two views of the same scene, estimate its 3D geometry Densely, not just at interest points...

68. Triangulation

Given projections of a 3D point in two or more images (with known camera matrices), find the coordinates of the point

Intersect two rays originating from the same point in the scene

Requires correspondences (knowledge which pixels are images of “the same point”)

Requires camera pose (to construct the 3D rays)

Geometric midpoint Linear approach 44 non-linear approach: Find X that minimizes the (squared) reprojection error

69. Epipolar geometry

find the relation between two different views of the same scene, recover camera placement, matching and triangulation for multiple views, scene modeling only from images 48:61

Epipole: Image location of the optical center of the other camera.

Epipolar plane: Plane through both camera centers and world point.

Epipolar line: Constrains the location where particular feature from one view can be found in the other.

70. Epipolar constraint

59

71. Essential matrix

60 Essential matrix is singular; has rank 2. The two remaining eigenvalues are equal. 5 degrees of freedom (translation + rotation have 6, but scale is arbitrary)

72. Fundamental matrix

holds for calibrated and uncalibrated cameras

63:68

Seven degrees of freedom, One equation per correspondence, To estimate, need at least 7 pairs of corresponding points, but the solution is non-linear, Linear solution with at least 8 point pairs: the normalized eight-point algorithm

Calibration not required, hence can deal with archival images, photos not taken for reconstruction purposes, varying intrinsics

The cookbook recipe

find interest points in both images

match them without epipolar constraint

compute epipolar geometry

... refine

73. Eight-point algorithm

69:71

74. Binocular stereo

Parallel cameras 62

Special case of epipolar geometry for stereo cameras with a standard binocular

75. Disparity

22:30

76. Rectification

map both image planes to a common plane parallel to the baseline

requires a homography for each image

after the transform, pixel motion is horizontal again

77. Baseline

The baseline is relatively small (compared to the depth of scene points) — “narrow-baseline stereo”

Matching regions are similar in appearance

Most scene points are visible in both views

78. Window-based matching

Choose some disparity range, For all pixels try all disparities and choose the one that maximizes the normalized

correlation or minimizes the SSD.

Challenges and Problems: How do we choose the right window size m ? Mismatches often lead to relatively poor results quality.

The similarity constraint is local, each reference window is matched independently, other points do not influence the result. Need to enforce non-local correspondence constraints, Require spatial regularity (Computer Vision II)

79. Normalized correlation

The normalized correlation computes the cosine of the angle between the patches 40

80. Motion field

Motion field = 2D motion field representing the projection of the 3D motion of points in the scene onto the image plane.

81. Optical flow

2D velocity field describing the apparent motion in the images

Image brightness at time t and location x 58

Estimation of motion parameters in robotics. Reconstruction of the 3-D world from an image sequence (structure-from-motion). Recognition & tracking of moving objects, e.g. human body motion.

82. Image interpolation

Possible interpolation filters: nearest neighbor, bilinear, bicubic (interpolating)

Needed to prevent "jaggies". When iteratively warping, always warp the original image

83. Brightness constancy

Assumption 1:

Image measurements (e.g. brightness) in a small region remain the same although their location may change.

84. OFCE (Optical Flow Constraint Equation)

17:20

85. Aperture problem

Can only measure normal velocity (perpendicular to edge) Barber pole illusion

21:27

86. Lucas-Kanade

solve for the motion 37

LK is a local optical flow method.

87. Image registration

We can use this to register (i.e. align) images: Compute the flow with the entire images. Shift the second image toward the first based on the flow. Iterate until convergence.

88. Image warping

We use image warping to shift the image. Translation rotation aspect affine perspective cylindrical

Add "contribution" to several pixels, normalize later (splatting).

Resample pixel value from interpolated source image

89. Coarse-to-fine estimation

The LK-model only holds for small motions.

Build a Gaussian pyramid, Start with the lowest resolution (motion is small!), Use this motion to pre-warp the next finer scale, Only compute motion increments (small!)

90. View-based recognition

Pixels (or projections onto global basis vectors) are the descriptor. Very limited amount of invariance!

91. Bag-of-Words model

16 Feature representation 18 Dense Sparse

1 - Local Interest Point Detection (done), Finding discriminative points (Harris, Hessian), Scale invariant interest point detection (Harris-Laplace)

2 - Local Descriptors / Features (done)

3 - Bag-of-Words Model (BoW) for Object Categorization

Independent features, Histogram representation, Histogram of features assigned to each cluster

92. Color histograms

19, 20 Recognition Works surprisingly well

Advantages: Invariant to object translations, Invariant to image rotations, Slowly changing for out-of-plane rotations, No perfect segmentation necessary, Histograms change gradually when part of the object is occluded, Possible to recognize deformable objects

Problems: The pixel colors change with the illumination („color constancy problem“), Intensity, Spectral composition, (illumination color), Not all objects can be identified by their color distribution.

93. Histogram distances

Histogram intersection 21 Euclidean distance 22 Chi-square 23

Both intersection and χ^2 give good performance. Intersection is a bit more robust. χ^2 is a bit more discriminative. Euclidean distance is not robust enough.

94. Receptive field histogram

Any local descriptor (e.g. filter, filter combination) can be used to build a histogram. 29

Multidimensional Histograms contains no structural description. Many different objects should result in the same histograms. But Support regions of neighboring descriptors overlap. Neighborhood relations are captured implicitly.

Dx - Dy : Rotation-variant, Mag - Lap : Rotation-invariant

95. Vector quantization

k-means

96. BoW “features”

Feature detection and representation

Detect patches Local interest operator (e.g. Harris-Laplace) or regular grid

Normalize patch

Compute descriptor e.g. SIFT, shape context, etc.

97. Bayesian decision theory

13:20

98. Naive Bayes classifier

21:24

99. Discriminative & generative approaches

Linear discriminant function: 29, 30

100. Neural Networks

101. Back propagation

102. Stochastic gradient descent

103. Convolutional Neural

104. Networks

105. Non-linearity

106. Spatial pooling

107. Network Architectures

108. Convolutional vs. fully

109. connected layers

110. CNN visualization

111. Receptive fields

112. Histogram & Pyramid match kernels

113. CNN training

114. Adversarial examples

115. Role of depth in CNNs

116. Sliding window detector

117. HOG

118. Bootstrapping

119. False positives etc.

120. R-CNN and variants