

GLOSSARY OF COMPUTER VISION TERMS

ROBERT M. HARALICK and LINDA G. SHAPIRO

University of Washington, Dept. of Electrical Engineering, Seattle, WA 98195, U.S.A.

(Received 20 September 1989; in revised form 2 March 1990; received for publication 23 March 1990)

1. THE IMAGE

1. An *image* is a spatial representation of an object, a two-dimensional or three-dimensional scene, or another image. It can be real or virtual as in optics. In computer vision, "image" usually means recorded image such as a video image, digital image, or picture. It may be abstractly thought of as a continuous function I of two variables defined on some bounded and usually rectangular region of a plane. The value of the image located at spatial coordinates (r, c) is denoted by $I(r, c)$. For optic or photographic sensors, $I(r, c)$ is typically proportional to the radiant energy received in the electromagnetic band to which the sensor or detector is sensitive in a small area around (r, c) . For range finder sensors, $I(r, c)$ is a function of the line of sight distance from (r, c) to an object in the three-dimensional world. For a tactile sensor, $I(r, c)$ is proportional to the amount that the surface at and around (r, c) deforms the sensor. When the image is a map, $I(r, c)$ is an index or symbol associated with some category such as a color, a thematic land use category, a soil type, or a rock type. A recorded image may be in photographic format, video signal format, or digital format.

2. A *video image* is an image in electronic signal format capable of being displayed on a cathode ray tube screen or monitor. The video signal can be generated from devices like a CCD camera, a vidicon, a flying spot scanner, a tactile sensor, a range sensor, or a frame buffer driving a digital to analog converter. Video images have two common formats. In the frame format, the video signal itself is a sequence of signals, the i th signal representing the i th line of the image. The i th signal is separated from the $(i + 1)$ st signal by a horizontal sync or pulse. Each video frame is separated from the next video frame by a vertical sync pulse. In the interlaced format, the video signal is divided into two fields. The first field contains all the odd numbered lines and the second field contains all the even numbered lines. As in the frame format, the i th line of the field is its i th signal, and it is separated from next line of the field by a horizontal sync pulse. Successive fields are separated by vertical sync pulses.

3. The *gray level*, *gray shade*, *gray tone*, *gray tone intensity*, *image intensity*, *image density*, *brightness*, or *image value* is a number or value assigned to a

position on an image. For optic or photographic sensors, the image intensity at (r, c) is proportional to the integrated output, reflectance, or transmittance of a small area, usually called a resolution cell or pixel, centered on the position (r, c) . Its value can be related to transmittance, reflectance, a coordinate of the tristimulus, ICI, YIQ, or RGB color coordinate system, brightness, radiance, luminance, density, voltage, or current.

4. *Resolution* is a generic term which describes how well a system, process, component, material, or image can reproduce an isolated object consisting of separate closely spaced objects or lines. The *limiting resolution*, *resolution limit* or *spatial resolution* is described in terms of the smallest dimension of the target or object that can just be discriminated or observed. Resolution may be a function of object contrast and spatial position as well as element shape (single point, number of points in a cluster, continuum, or line etc.).

5. A *resolution cell* is the smallest most elementary areal constituent having an associated image intensity in a digital image. A resolution cell is referenced by its spatial coordinates which are the center coordinates of its area. The resolution cell or spatial formations of resolution cell constitute the basic unit for low level processing of digital image data. Resolution cells usually have areas which are square, rectangular, or hexagonal.

6. *Acutance* is a measure of the sharpness of edges in a photograph or image. It is defined for any edge by the average squared rate of change of the image intensity across the edge divided by the total image intensity difference from one side of the edge to the other side of the edge.

7. The *contrast* of an object against its background can be measured by: (1) its *contrast ratio*, which is the ratio between the higher of object transmittance or background transmittance to the lower of object transmittance or background transmittance; (2) its *contrast difference*, which is the difference between the higher density of object or background to the lower density of object or background; (3) its *contrast modulation*, which is the difference between the darker of object or background image intensity and the lighter of the two divided by the sum of object image intensity and background image intensity.

8. A *pixel*, *picture element*, or *pel* is a pair whose

first member is a resolution cell or (row, column) spatial position and whose second member is the image intensity value or vector of image values associated with the spatial position.

9. A *voxel*, short for volume element, is an ordered pair whose first component is a (row, column, slice) location of a volume rectangular parallelepiped and whose second component is the vector of properties in the rectangular parallelepiped volume.

10. An *edgel*, short for edge element, is a triplet whose first component is the (row, column) location of a pixel, whose second component is the position and orientation of an edge running through the pixel, and whose third component is the strength of the edge.

11. *Raster scan order* refers to the sequence of pixel locations obtained by scanning the spatial domain of an image in a left to right scan of each image row with the rows taken in a top to bottom ordering. Frame format video images are images scanned in raster scan order.

12. A *range image* is an image in which each pixel value is a function of the distance between the pixel and the object surface patch imaged on the pixel. Depending on the sensor and preprocessing used to create the range image, the distance can be the distance between the image plane and the ranged surface patch, the line of sight distance between the pixel and its corresponding ranged surface patch or some function of these distances and the pixel's position.

13. A *digital image*, *digitized image*, or *digital picture function* is an image in digital format and is obtained by partitioning the area of the image into a finite two-dimensional array of small uniformly shaped mutually exclusive regions called resolution cells and assigning a representative image value to each such spatial region. A digital image may be abstractly thought of as a function whose domain is the finite two-dimensional set of resolution cells and whose range is the set of possible image intensities.

14. *Rangel* is the range data element produced by a range sensor. It is a pair whose first member is a row column spatial position and whose second member is the range value or a vector whose first component is the range value and whose second component is the image intensity value.

15. A *depth map* or *range map* is a digital range image in which the range value in each pixel's position is the distance between the image plane and the ranged surface patch corresponding to the pixel.

16. An *orientation map* is a digital image in which each pixel contains the 3D orientation vector of the normal to the 3D surface patch corresponding to the pixel position.

17. A *multi-image set* or *multi-band image* is a set of registered images each related to the same subject but taken at different times, from different positions, with different lighting, with different sensors, at different electromagnetic frequencies, with different

polarizations, or from different sections of the subject. Although there is a high degree of information redundancy between images in a multi-image set, each image usually has some information not available in any one of or combination of the other images in the set. If the multi-image set has N images, then each resolution cell is associated with an N -tuple of image values.

18. A *multi-spectral image* is a multi-band image in which each band is an image taken at the same time, but sensitive in a different part of the electromagnetic spectrum.

19. A *time varying image*, *multi-temporal image*, *dynamic imagery*, or *image time sequence* is a multi-image set in which each successive image in the set is taken of the same scene at a successive time. Between successive snapshots, the objects in the scene may move or change and the sensor may move.

20. A *binary image* is an image in which each pixel takes either the value zero or the value one.

21. A *gray scale image* or a *gray level image* is an image in which each pixel has a value in a range larger than just 0 or 1. Gray scale images typically have values in the range 0 to 63, 0 to 255, or 0 to 1023 corresponding to 6 bit, 8 bit, or 10 bit digitizations.

22. A *symbolic image* is an image in which the value of each pixel is an index or symbol.

23. A *histogram* or *image histogram* is a function h defined on the set of image intensity values to the non-negative integers. The value $h(k)$ is given by the number of pixels in the image having image intensity k . For images having a large gray tone range, the image will often be quantized before being histogrammed or will be quantized on the fly during the histogramming process.

2. PHOTOMETRY AND ILLUMINATION

24. *Luminous flux* is radiant power evaluated according to its capacity to produce visual sensation. *Luminous intensity* in a given direction is measured in terms of luminous flux per steradian. The unit of luminous intensity is the *candela*. The luminance of a black body radiator at the temperature of solidification of platinum is 60 candelas per square centimeter. The unit of luminous flux is the *lumen*. The luminous flux emitted by a uniform point light source of luminous intensity of one candela in one steradian solid angle is one *lumen*.

25. The *illumination* at a point on a surface is the luminous flux incident on an infinitesimal element of the surface centred at the given point divided by area of the surface element. The unit of illumination is the *lux* or meter-candle. The lux is equal to one lumen per square meter. Another unit of illumination is the *foot candle* and it is equal to one lumen per square foot. The illumination at a point on a surface due to a point source of light is proportional to the luminous intensity of the source in the direction of the surface point and to the cosine of the

angle between this direction and the surface normal direction. It is inversely proportional to the square of the distance between the surface point and the source.

26. The *illuminance* in a given direction at a surface point is the luminous intensity in that direction of an infinitesimal surface element containing the given point divided by the area of the orthogonal projection of the element on a plane perpendicular to the given direction.

27. The *radiance* of an object is a measure of the power per unit foreshortened surface area per unit solid angle radiated or reflected by the object about a specified direction. Radiance can be a function of the viewing angle and the spectral wavelength and bandwidth.

28. The *radiant intensity* of a point object is a measure of the radiant power per steradian radiated or reflected by the object. Radiant intensity can be a function of the viewing angle and the spectral wavelength and bandwidth.

29. *Irradiance* is the power per unit area of radiant energy incident on a surface.

30. The *reflectance*, the *reflection coefficient*, or the *bidirectional reflectance distribution function* of a surface is the ratio of the radiant power per unit area reflected by the surface to the radiant power per unit area incident on the surface. The reflectance can be a function of the incident angle of the radiance, the viewing angle of the sensor, and the spectral wavelength and bandwidth.

31. A *reflectance image* or *reflectance map* is a digital image in which the value in each pixel's position is proportional to the reflectance of the surface patch imaged at the pixel's position for a given illumination and viewing direction.

32. A *Lambertian surface* is a uniformly diffusing surface. It appears as a matt surface and it has a reflectance function which is a constant. The reflectance function of a Lambertian surface does not depend on the viewing angle and, therefore, a planar surface having a Lambertian reflectance appears equally bright from all viewing angles. For a Lambertian surface, the luminous intensity per unit area in a given direction varies as the cosine of the angle between the direction and the surface normal direction.

33. *Backlighting* refers to an illumination arrangement in which the light source is on the opposite side of the object from the camera. Backlighting tends to produce images which are black and white silhouettes.

34. *Frontlighting* refers to an illumination arrangement in which the light source is on the same side of the object as the camera.

35. *Ambient light* refers to the light which is present in the environment around a machine vision system and which is generated from sources outside of the system. From the point of view of the machine vision system, ambient light is unplanned light which

might adversely affect the image processing. Care is usually taken to minimize its effect.

36. The *r, g, b chromaticity* coordinates of a multi-spectral pixel, where red brightness is R , green brightness is G and blue brightness is B , is given by

$$r = \frac{R}{R + G + B}$$

$$g = \frac{G}{R + G + B}$$

$$b = \frac{B}{R + G + B}$$

37. The *H, S hue saturation* coordinates of a multi-spectral pixel whose chromaticity coordinates are r , g , b is given by

$$H = \begin{cases} \theta & \text{if } b < g \\ 360 - \theta & \text{if } b > g \end{cases}$$

where

$$\theta = \arccos \frac{2r - g - b}{\sqrt{6[(r - \frac{1}{3})^2 + (g - \frac{1}{3})^2 + (b - \frac{1}{3})^2]^{1/2}}}$$

$$S = 1 - 3 \min\{r, g, b\}.$$

38. The *YIQ coordinate* used in NTSC color TV transmissions is related to the RGB coordinates by the following linear transformation:

$$\begin{pmatrix} Y \\ I \\ Q \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.144 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}.$$

3. PHOTOGRAMMETRY

39. *Analytic photogrammetry* refers to the analytic mathematical techniques which permit the inference to geometric relations between points and lines in the two-dimensional perspective projection image space and the three-dimensional object space.

40. *Digital photogrammetry* refers to the computer processing of perspective projection digital images with analytic photogrammetry techniques as well as other computer techniques for the automatic interpretation of scene or image content.

41. *Relative orientation* in analytic photogrammetry is the relative position and orientation of one common reference frame with respect to another. When two common reference frames are in known relative orientation, the rays emanating from the same object point located on each camera's image will intersect exactly at a point in 3D space.

42. The *exterior orientation* or *outer orientation* refers to the position and orientation of a camera reference frame with respect to a world reference frame.

43. *Absolute orientation* in analytic photogrammetry is the rotation and translation transformation(s) by which one or more camera reference

frame(s) can be made to correspond to a world reference frame.

44. The *optic axis* or *principal axis* is the straight line which passes through the centers of curvatures of the lens surfaces.

45. The *principal point* is that point on the image which is the intersection of the image plane with the optic axis.

46. The *center of perspectivity* of a perspective projection is the common point where all rays meet.

47. The *principal distance* is the distance between the center of perspectivity and the image projection plane. For a fixed image, it is sometimes called the *camera constant* or *Gaussian focal length*.

48. The *inner orientation* or *internal orientation* is given by the triple (u_0, v_0, f) whose (u_0, v_0) is the position of the principal point in the measurement image plane coordinate system and f is the principal distance. Inner orientation may also include the values of the free parameters which describe the lens distortion.

49. A *perspective projection* is defined in terms of a projection plane and a center of perspectivity. Imaging sensors whose ideal model is the pinhole camera generate perspective projection images. There are three commonly used frameworks for defining the relationship between a 3D point and its 2D perspective projection. All take the projection plane to be perpendicular to the z -axis. In the first framework, the center of perspectivity is taken to be the origin and the projection plane is a distance f from the origin on the positive z -axis. In this case the projection (P_x, P_y) of the point (x, y, z) is given by

$$P_x = \frac{fx}{z}, P_y = \frac{fy}{z}.$$

In the second framework, the center of perspectivity is taken to be $(0, 0, -f)$ and the projection plane passes through the origin. In this case the projection (P_x, P_y) of the point (x, y, z) , $z > 0$, is given by

$$P_x = \frac{fx}{f+z}, P_y = \frac{fy}{f+z}.$$

In the third framework, the center of perspectivity is taken to be $(0, 0, f)$ and the projection plane passes through the origin. In this case the projection (P_x, P_y) of the point (x, y, z) , $z < 0$, is given by

$$P_x = \frac{fx}{f-z}, P_y = \frac{fy}{f-z}.$$

50. A *parallel* or *orthographic projection* onto a plane perpendicular to the z -axis of a point (x, y, z) produces the projected (P_x, P_y) defined by

$$P_x = sx, P_y = sy$$

where s is the scale factor of the projection.

51. The *parallax* is the observed positional difference of a projected 3D point on a pair of 2D

perspective images. The difference in position is caused by a shift in the position of the perspective centers and optical axis orientation. That portion of the parallax in the direction of the x -axis is called the *x-parallax*. That portion of the parallax in the direction of the y -axis is called the *y-parallax*. For a pair of stereo images where the line joining the centers of perspectivity is parallel to the x -axis, the parallax will be entirely in the direction of the x -axis when the two image planes are in the same orientation. Should one image plane be tilted with respect to the other, the y -parallax will not be zero.

52. A *vanishing point* is the point in the 2D perspective projection image plane where a system of 3D parallel lines converge. The vanishing points of all systems of 3D parallel lines parallel to a given plane will lie along a corresponding line in the 2D perspective projection image plane called the *vanishing line* for the given plane.

4. IMAGE OPERATORS

53. An *image operator*, *image transform*, or *image transform operator* is a function which takes an image for its input and produces an image for its output. The domain of a transform operator is often called the spatial or space domain. The range of the transform operator is often called the transform domain. Some image transform operators have spatial and transform domains of entirely different geometry or character; the image in the spatial domain may appear entirely different from and have a different interpretation from the image in the transform domain. Specific examples of these kinds of image transforms include Fourier, Sine, Cosine, Slant, Haar, Hadamard, Mellin, Karhunen-Loeve, and Hough transforms. Image operators which have spatial and transform domains of similar geometry or character include point operators, neighborhood operators, and spatial filters.

5. POINT OPERATORS

54. A *point operator* is an image operator in which the output image value at each pixel position depends only on the input image value at the corresponding pixel position.

55. *Thresholding* is an image point operation which produces a binary image from a gray scale image. A binary one is produced on the output image whenever a pixel value on the input image is above a specified minimum threshold level. A binary zero is produced otherwise. Alternatively thresholding can produce a binary one on the output image whenever a pixel value on the input image is below a specified maximum threshold level. A binary zero is produced otherwise.

56. *Level slicing* or *density slicing* is a point operation which employs two thresholds and produces a binary image. A binary one is produced on the output

image wherever a pixel value on the input image lies between the specified minimum and maximum threshold levels. A binary zero is produced otherwise.

57. *Multi-level thresholding* is a point operator employing two or more thresholds. Pixel values which are in the interval between two successive threshold values are assigned an index associated with the interval.

58. *Contrast stretching* refers to any monotonically increasing point operator whose effect is to increase or enhance the visibility of an image's detail.

59. *Quantizing* is a monotonically increasing point operator by which each image intensity value in a digital image is assigned a new value from a given finite set of quantized values. The quantized image has fewer distinct gray levels but may make better use of the dynamic range. Thus quantizing often enhances the image's appearance. There are four often used methods of quantizing: equal interval quantizing, equal probability quantizing, minimum variance quantizing, and histogram hyperbolization. In each method, the range of image values from the maximum to the minimum value is divided into contiguous intervals and each image value is assigned either the mean value of the quantized class to which it belongs or the index of the quantized class to which it belongs.

(a) 60. In *equal interval quantizing* or *linear quantizing*, the range of image values from maximum value to minimum value is divided into contiguous intervals each of equal length, and each image value is assigned to the quantized class which corresponds to the interval within which it lies;

(b) 61. In *equal probability quantizing*, the range of image values is divided into contiguous intervals such that after the image values are assigned to their quantized class there is an equal frequency of occurrence for each quantized value in the quantized digital image; equal probability quantizing is sometimes referred to as or *histogram equalization*;

(c) 62. In *minimum variance quantizing*, the range of image values is divided into contiguous intervals such that the weighted sum of the variances of the quantized intervals is minimized. The weights are usually chosen to be quantized class probabilities which are computed as the proportional areas on the image which have values in the quantizing intervals.

(d) 63. In *histogram hyperbolization quantizing*, the range of image values is divided into contiguous intervals and each image value is assigned to the mean of its quantized class. The division is done in such a way that the quantized image has a uniform perceived brightness. Histogram hyperbolization takes into account the nonlinearity of the human eye brain combination.

64. *Masking* is a point operator applied to a two-band image. One image band is a binary image B

and is called the mask image band. The second image band I is called the image to be masked. Masking produces a resulting image J whose pixels take the value zero wherever the mask image has value zero and whose pixels take the value of the image I wherever the mask image has value one. That is,

$$J(r, c) = \begin{cases} 0 & \text{if } B(r, c) = 0 \\ I(r, c) & \text{if } B(r, c) = 1. \end{cases}$$

65. *Change detection* is the process by which two registered images may be compared, pixel by pixel, and a binary one value given to the output pixel whenever corresponding pixels on the input images have significantly different enough gray levels. Corresponding pixels on the input images which do not have significantly different enough gray levels generate a binary zero value on the output image. A change detection operator is a point operator.

6. SPATIAL OPERATORS

66. *Two-dimensional signal processing* refers to that area of image processing in which the one-dimensional signal processing techniques of noise filtering, restoration, data compression, and detection have been generalized to two dimensions and thereby made applicable to image data.

67. A *neighborhood operator* is an image operator in which the output image value at each pixel position depends only on the input image values in a neighborhood containing or surrounding the corresponding input pixel position.

68. A *spatial filter* is an image operator in which the spatial and transform domain have similar geometries and in which the image output value at each pixel depends on more than one pixel value in the input image. Usually, but not always, the image output value has its highest dependence on the image input values in some neighborhood centered in the corresponding pixel in the input image.

69. A *linear spatial filter* is a spatial filter for which the image intensity at coordinates (r, c) in the output image is some weighted average or linear combination of the image intensities located in a particular spatial pattern around coordinates (r, c) of the input image. A linear spatial filter is often used to change the spatial frequency characteristics of the image. For example, a linear spatial filter which emphasizes high spatial frequencies will tend to sharpen the edges in an image. A linear spatial filter which emphasizes low spatial frequencies will tend to blur the image and reduce salt and pepper noise. When the purpose of the filter is to enhance neighborhoods having certain shapes, the operation is sometimes called mask matching.

70. The *kernel* of a linear spatial filter is a function defined on the domain of the spatial pattern of the filter and whose value at each pixel of the domain is the weight or coefficient of the linear combination which defines the spatial linear filter.

71. A *box filter* is a linear spatial smoothing filter in which each pixel in the filtered image is the equally weighted average of the pixels in a rectangular window centered at its spatial position in the input image.

72. The *Gaussian filter* is a linear spatial smoothing filter whose kernel is given by the two-dimensional Gaussian

$$k(r, c) = \frac{1}{2\pi} e^{-\frac{1}{2}\left(\frac{r^2}{\sigma_r^2} + \frac{c^2}{\sigma_c^2}\right)}.$$

Filtering an image with a Gaussian filter will smooth the image.

73. *Convoluting* an image I with a kernel k having support or domain K produces a convolved image, denoted by $I * k$, which is defined by

$$(I * k)(r, c) = \sum_{(i,j) \in K} I(r-i, c-j)k(i, j).$$

Convolution is a linear operator.

74. A two-dimensional filtering operation is called *separable* if the convolution can be decomposed into two successive one-dimensional convolutions, one convolution operating on the image row by row and the second convolution operating on the image column by column.

75. *Correlating* an image I with a kernel k having support or domain K produces a correlated image J defined by

$$J(r, c) = \sum_{(i,j) \in K} I(r+i, c+j)k(i, j).$$

Correlation is a linear operator.

76. The *discrete Fourier transform* \hat{I} of a digital image I represents the image in terms of a linear combination of complex exponentials. The Fourier transform \hat{I} is defined by

$$\hat{I}(w_r, w_c) = \frac{1}{RC} \sum_{r=0}^{R-1} \sum_{c=0}^{C-1} I(r, c) e^{-2j\pi\left(\frac{rw_r}{R} + \frac{cw_c}{C}\right)}.$$

$\hat{I}(w_r, w_c)$ is the coefficient of the complex exponential $e^{2j\pi\left(\frac{rw_r}{R} + \frac{cw_c}{C}\right)}$ in the linear combination representing I as can immediately be seen from the corresponding relation

$$I(r, c) = \sum_{w_r=0}^R \sum_{w_c=0}^C \hat{I}(w_r, w_c) e^{2j\pi\left(\frac{rw_r}{R} + \frac{cw_c}{C}\right)}$$

which is called the *inverse discrete Fourier transform*. The variables w_r and w_c have the interpretation of being row and column *spatial frequencies*.

77. A *high pass filter* is a linear spatial filter which attenuates the low spatial frequencies of an image and accentuates the high spatial frequencies of an image. It is typically used to enhance small details, edges, and lines.

78. A *low pass filter* is a linear spatial filter which attenuates the high spatial frequencies of an image and accentuates the low spatial frequencies of an

image. It is typically used to suppress small undesired details, eliminate noise, enhance coarse image features, or smooth the image.

79. A *band pass filter* is a linear spatial filter which attenuates those spatial frequencies outside the band and accentuates those spatial frequencies within the band. It is typically used to enhance details of the image whose spatial size characteristics are related to the spatial frequencies within the band.

80. A *pyramid* or *image pyramid* is a sequence of copies of an image in which both sample density and resolution are decreased in regular steps. The bottom level of the pyramid is the original image. Each successive level is obtained from the previous level by a filtering operator followed by a sampling operator.

In the *Gaussian image pyramid*, the resolution is decreased by successive convolutions of the image at the previous level of the pyramid with a Gaussian-like kernel. After the low pass Gaussian convolutions, the sample density is typically decreased by sampling every other pixel.

In the *morphological image pyramid*, each successive level is obtained by an opening and closing operation on the previous level followed by sampling.

In the *Laplacian image pyramid*, each successive layer is obtained by taking the Laplacian of the corresponding level on the Gaussian pyramid. The Laplacian convolution kernel here is typically defined as the kernel obtained by taking the Laplacian of a Gaussian having an appropriately chosen value for its standard deviation.

81. An *edge operator* or *step edge operator* is a neighborhood operation which determines the extent to which each pixel's neighborhood can be partitioned by a simple arc passing through the pixel where pixels in the neighborhood on one side of the arc have one predominant value and pixels in the neighborhood on the other side of the arc have a different predominant value. Some edge operators can also produce a direction which is the predominant tangent direction of the arc as it passes through the pixel.

There are four classes of edge operators: *gradient operators*, *Laplacian operators*, *zero-crossing operators*, and *morphologic edge operators*. The gradient operators compute some quantity related to the magnitude of the slope of the underlying image gray tone intensity surface of which the observed image pixel values are noisy discretized sample. The Laplacian operators compute some quantity related to the Laplacian of the underlying image gray tone intensity surface. The zero-crossing operators determine whether or not the digital Laplacian or the estimated second direction derivative has a zero-crossing within the pixel. The morphologic edge operators compute a quantity related to the residues of an erosion and/or dilation operation.

82. An *edge image* is an image in which each pixel is labelled as "edge" or "non-edge". In addition to

this basic labeling, pixels in an edge image may carry additional information such as edge direction, edge contrast, or edge strength.

83. *Edge linking* refers to the process by which neighboring edge labeled pixels can be aggregated to constitute a chain or sequence of edge pixels.

84. *Boundary detection* or *boundary delineation* refers to any process which determines a chain of pixels separating one image region from a neighboring image region.

85. An *occluding edge* is an image edge which arises from a range or depth discontinuity. This typically happens where one object surface projects to a pixel on one side of the edge and another object surface which is some distance behind the first object surface projects to a pixel on the other side of the edge. Step edges in depth maps are always occluding edges.

86. A pair of straight edges are said to be *anti-parallel* if there are no edges between them and the edges have opposite contrast.

87. *Homomorphic filtering* is a filtering process in which the filter is applied to the logarithm of the image and the output image is obtained by exponentiating the filtered logarithm image.

88. A *median filter* is a non-linear neighborhood image smoothing spatial filter in which the value of an output pixel is the median value of all the input pixels in the supporting neighborhood of the filter about the given pixel's position. Median filters are used to smooth and remove noise from images.

89. *Image smoothing* refers to any spatial filtering producing an output image which spatially simplifies and approximates the input image. Image smoothing suppresses small image details and enhances large or coarse image structures.

90. A *scale space image* is an image in which each pixel's value is a function indicating for each standard deviation σ the value at the pixel's position of the convolution of the image with a Gaussian kernel having standard deviation σ .

91. *Scale space structure* refers to that analysis of a scale space image in which each pixel's value is a function specifying for each possible standard deviation σ whether the pixel contains a zero crossing of some combination of a fixed order of spatial partial derivatives evaluated at the pixel's position.

7. MORPHOLOGIC OPERATORS

92. *Mathematical morphology* refers to an area of image processing concerned with the analysis of shape. The basic morphologic operations consist of dilating, eroding, opening, and closing an image with a structuring element.

93. The *structuring element* of a morphologic operator is a function defined on the domain of the spatial pattern of the morphologic operator and whose value at each pixel of the domain is the weight or coefficient employed by the morphologic operator at that pixel

position. The structuring element of a morphologic operator has a role in morphology exactly analogous to the role of the kernel in a convolution operation.

94. *Dilating* an image I by a structuring element s having support or domain S produces a dilated image denoted by $I \oplus s$ which is defined by

$$(I \oplus s)(r, c) = \max_{(i,j) \in S} \{I(r-i, c-j) + s(i, j)\}.$$

Dilating is a commutative, associative, translation invariant, and increasing operation. Dilating is the dual operation to eroding.

95. *Eroding* an image I by a structuring element s having support or domain S produces an eroded image denoted by $I \ominus s$ which is defined by

$$(I \ominus s)(r, c) = \min_{(i,j) \in S} \{I(r+i, c+j) - s(i, j)\}.$$

Eroding is a translation invariant and increasing operation. It is the dual operation to dilating.

96. *Opening* an image I with a structuring element s produces an opened image denoted by $I \circ s$ which is defined by

$$I \circ s = (I \ominus s) \oplus s.$$

Opening is an increasing, anti-extensive, and idempotent operation. It is the dual operation to closing. Opening an image with a disk shaped structuring element smooths the contour, breaks narrow isthmuses, and eliminates islands and capes smaller in size or width than the disk structuring element.

97. *Closing* an image I with a structuring element s produces a closed image denoted by $I \bullet s$ which is defined by

$$I \bullet s = (I \oplus s) \ominus s.$$

Closing is an increasing, extensive, and idempotent operation. It is the dual operation to opening. Closing an image with a disk shaped structuring element smooths the contours, fuses narrow breaks and long thin gulfs, eliminates holes smaller in size than the disk structuring element and fills gaps on the contour.

98. A *thinning operator* is a symbolic image neighborhood operator which deletes, in some symmetric way, all the interior border pixels of a region which do not disconnect the region. Successive applications of a thinning operator reduces a region to a set of arcs which constitute a *skeleton* of the region.

99. A *thickening operator* is a symbolic image neighborhood operator which in some symmetric way aggregates all background pixels near enough to a region into the region.

8. HOUGH TRANSFORM

100. The *discrete Radon transform* $R: Q \rightarrow [0, \infty)$ of a function $I: X \rightarrow [0, \infty)$ relative to a functional form $F: X \times Q \rightarrow [0, \infty)$ is defined by

$$R(q) = \sum_{\{x \in X | F(x, q) = 0\}} I(x).$$

101. The *Hough transform* $H: \mathcal{B} \rightarrow [0, \infty)$ of a

function $I: X \rightarrow [0, \infty)$ relative to a functional form $F: X \times Q \rightarrow [0, \infty)$, where \mathcal{B} is a partition of the parameter space Q , is defined by

$$H(B) = \sum_{\{x \in X \mid \text{for some } q \in B, F(x, q) = 0\}} I(x), \quad B \in \mathcal{B}.$$

When I has the form $I(x) = 1$, for every $x \in X_0$, where $X_0 \subseteq X$, and $I(x) = 0$ elsewhere, the Hough transform of X_0 , relative to a functional form $F: X \times Q \rightarrow [0, \infty)$, has the simple form defined by

$$H(B) = \#\{x \in X_0 \mid \text{for some } q \in B, F(x, q) = 0\}.$$

The Hough transform is a transform which can aid in the detection of image arcs of a given shape or form or 3D object shapes. Each shape or form has some free parameters which when specified precisely define the arc, shape or form. The shape having free parameter q corresponds to the set $\{x \in X \mid F(x, q) = 0\}$. The free parameters constitute the transform domain or the parameter space of the Hough transform. Depending on the information available to the Hough transform, each neighborhood of the image or object surface being transformed will map to a point or a set of points in the Hough parameter space. The Hough transform discretizes the Hough parameter space into bins and counts for each bin how many neighborhoods on the image or object surface has one of its transformed points lie in the volume assigned to the bin.

102. The *Gaussian sphere* refers to a unit sphere and its associated spherical coordinate system. The quantities usually represented on the Gaussian sphere are orientation vectors.

103. *Gradient space* is a two-dimensional space whose axes represent the first order partial derivatives of a surface of the form $z = f(x, y)$. Each point in gradient space corresponds to the orientation of a possible surface normal.

104. The *extended Gaussian image or orientation histogram* of a 3D object is a two-dimensional histogram or Hough transform of the surface normal orientations of the object. It is computed by tessellating the surface of a sphere into cells and assigning to each cell a value which estimates the total area of the object's surface having surface normal orientation which falls within the cell.

9. DIGITAL GEOMETRY

105. The *Euclidean distance* between two points $p = (p_1, \dots, p_N)$ and $q = (q_1, \dots, q_N)$ is defined by

$$d(p, q) = \sqrt{\sum_{n=1}^N (p_n - q_n)^2}.$$

106. The *block or city distance* between two points $p = (p_1, \dots, p_N)$ and $q = (q_1, \dots, q_N)$ is defined by

$$d(p, q) = \sum_{n=1}^N |p_n - q_n|.$$

107. The *square or max distance* between two points $p = (p_1, \dots, p_N)$ and $q = (q_1, \dots, q_N)$ is defined by

$$d(p, q) = \max_{n=1, \dots, N} |p_n - q_n|.$$

108. The *distance transform* of a binary image is an image having in each pixel's position its distance from the nearest binary zero pixel of the input image. Distance can be city block distance, Euclidean distance, or square distances.

109. A *figure* F or a *subimage* F in a continuous or digital image I is any function F whose domain is some subset A of the set of spatial coordinates or resolution cells, whose range is the set G of image intensities and which is defined by $F(r, c) = I(r, c)$ for any (r, c) belonging to A .

110. A *region* R of an image is any subset of resolution cells in the spatial domain of the image.

111. A neighboring pair of pixels are said to be *4-connected* if they share a common side. A neighboring pair of pixels are said to be *8-connected* if they share a common side or a common corner.

112. A region R is *connected* if there is a path between any two resolution cells contained in R . More precisely, R is 4-connected (8-connected) if for each pair of resolution cells (r, c) and (u, v) belonging to R , there exists some sequence $\langle (a_1, b_1), (a_2, b_2), \dots, (a_m, b_m) \rangle$ of resolution cells belonging to R such that $(r, c) = (a_1, b_1)$, $(u, v) = (a_m, b_m)$, and (a_i, b_i) is 4-connected (8-connected) to (a_{i+1}, b_{i+1}) , $i = 1, 2, \dots, m - 1$.

113. A *blob* or *connected component* is a maximal sized connected region.

114. A *digital straight line segment* between resolution cells (r_1, c_1) and (r_2, c_2) is that subset of all pixels such that some part of the line segment joining (r_1, c_1) and (r_2, c_2) has a non-empty intersection with the pixel's area.

115. A region R is *convex* if for every pair of resolution cells in R , R contains the digital straight line segment which joins the pair of resolution cells.

116. A pixel is an *interior border pixel* of a region R if the pixel belongs to R and neighbors a pixel outside of R .

117. A pixel is an *exterior border pixel* of a region R if the pixel does not belong to R and neighbors a pixel belonging to R .

118. A pixel is an *interior pixel* of a region R if every pixel it neighbors belongs to R .

119. A *simple boundary* is an oriented closed curve which does not touch or cross itself. Pixels which are on the inside of a simple boundary constitute a connected region having no holes.

120. The *bounding contour* of a region R consists of the simple boundary which surrounds the pixels of R .

121. A set of pixels H constitutes a *hole* of a region R if H is a maximal connected set of pixels which do not belong to R but which are surrounded by R .

122. The *border* or *boundary* of a connected region R consists of its bounding contour and the (possibly empty) set of simple boundaries each of which surrounds the pixels belonging to some hole of R .

123. *Boundary following* refers to the sequential procedure by which the chain of the boundary pixels of a region can be determined.

124. A *concurve* is a continuous curve, usually representing a blob boundary, consisting of a connected chain of simply described arcs.

125. The *minimum perimeter polygon* of a digital curve C is the polygon of shortest length whose digitization is C . The shape of the minimum perimeter polygon is often similar to the general perceived shape of the digital curve.

126. *Contour tracing* is a searching or traversing process by which the bounding contour of a blob can be indentified.

10. 2D SHAPE DESCRIPTION

127. The *perimeter* of a connected region R is the length of the bounding contour of R .

128. The *area* A of a region R is defined by

$$A = [\#R] \cdot s$$

where s is the scale factor which specifies the area of a pixel.

129. The *centroid* (\bar{r}, \bar{c}) of a region R is the center of mass of the region. It is the mean (row, column) position for all pixels in the region and is given by

$$\bar{r} = \frac{1}{\#R} \sum_{(r,c) \in R} r, \bar{c} = \frac{1}{\#R} \sum_{(r,c) \in R} c.$$

130. The (j, k) th *moment* M_{jk} of a digital shape S is given by

$$M_{jk} = \sum_{(r,c) \in S} r^j c^k.$$

The center of gravity (\bar{r}, \bar{c}) of S can be expressed in terms of the moments of S :

$$\bar{r} = \frac{M_{10}}{M_{00}}, \bar{c} = \frac{M_{01}}{M_{00}}.$$

131. The (j, k) th *central moment* μ_{jk} of a digital shape S is given by

$$\mu_{jk} = \sum_{(r,c) \in S} (r - \bar{r})^j (c - \bar{c})^k.$$

132. (j, k) th *normalized central moment* of S is given by

$$\eta_{jk} = \frac{\mu_{jk}}{\mu_{00}^{\frac{j+k}{2}}}, \text{ where } \gamma = \frac{j+k}{2} + 1.$$

133. *Rotation movement moments* of S are given by

$$\phi(1) = \eta_{20} + \eta_{02}$$

$$\phi(2) = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\phi(3) = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$\phi(4) = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$\phi(5) = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2]$$

$$+ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

$$\phi(6) = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$\phi(7) = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2]$$

$$- (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2].$$

134. The *Euler number* of a region is the number of its connected components minus the number of its holes.

135. The *compactness* of a blob can be measured by the length of its perimeter squared divided by its area or alternatively measured by the standard deviation of the radii from the centroid to the boundary divided by the mean radius. The classical measure of perimeter squared divided by area has the disadvantage that in the digital domain it takes its smallest value not for a digital circle but for a digital octagon or diamond depending on whether 8-connectivity or 4-connectivity is used in calculating the perimeter.

136. The *bounding rectangle* of a region R is a rectangle which circumscribes R . It has its sides aligned with the row and column directions, its leftmost side aligning with the lowest numbered column of R , its rightmost side aligning with the highest numbered column of R , its topmost side aligning with the lowest numbered row of R , and its bottommost side aligning with the highest numbered row of R .

137. An *extremal pixel* of R is a pixel of R having from among all pixels in R either

(a) an extremal row coordinate value r and an extremal column coordinate value taken from among all the column positions c such that $(r, c) \in R$.

(b) an extremal column coordinate value c and an extremal row coordinate value taken from among all the row positions r such that $(r, c) \in R$.

A region may have as many as eight distinct extremal points, each of which must be lying on the bounding rectangle of the region. Extremal pixels can be used to represent the areal extent of a region and to infer the dominant axis length and orientation of the region.

138. The *second moment matrix*

$$\Sigma = \begin{pmatrix} \mu_{rr} & \mu_{rc} \\ \mu_{rc} & \mu_{cc} \end{pmatrix}$$

of a region R is defined by

$$\mu_{rc} = \frac{s_r^2}{A} \sum_{(r,c) \in R} (r - \bar{r})^2$$

$$\mu_{rc} = \frac{s_r s_c}{A} \sum_{(r,c) \in R} (r - \bar{r})(c - \bar{c})$$

$$\mu_{cc} = \frac{s_c^2}{A} \sum_{(r,c) \in R} (c - \bar{c})^2$$

where A is the area of the region, s_r is the row scale factor, and s_c is the column scale factor, \bar{r} is the row centroid, and \bar{c} is the column centroid.

139. The *elongation* or *elongatedness* of a blob or connected region can be measured in a variety of ways. One technique is to use the ratio of the length of the maximum length chord in the blob to the length of its maximum length perpendicular chord. A second technique is to use the square root of the ratio of the largest to smallest eigenvalue of the second central moment matrix of the blob. A third technique is to use the ratio of the largest distance between an opposing pair of extremal points to the distance between that opposing pair of extremal points having next to largest distance.

140. The *symmetric axis* or *medial axis* of a blob is a subset of blob pixels which are the centers of maximal lines, squares or disks which are contained in the blob. Associated with each pixel which is part of a symmetric axis may also be additional information such as the size of the maximal line or square or the radius of the maximal disk of which it is the center.

141. The *connected component* operator has as input a binary image and produces as output an image in which each binary one pixel is given a unique label of the maximally connected component of pixels having a binary one value to which it belongs.

142. In *connected component analysis* or *blob analysis*, the position and shape properties of each connected component are measured. Typical shape properties include area, perimeter, number of holes, bounding rectangle, extremal points, centroid, second moments, and orientation derived from second moments or extremal points. The connected components are then identified or classified by a decision rule on the basis of its measured properties.

143. *Signature analysis* of a binary image analyses the binary image in terms of its projections. Projections can be vertical, horizontal, diagonal, circular, radial, spiral, or general projections. The analysis consists of computing the projections, segmenting each projection, and taking property measurements of each projection segment. Signature analysis may also use the projection segmentation to induce a segmentation of the image.

144. A *binary image projection* is the histogram of the gray scale image produced by masking a projection index image with the given binary image.

Each pixel of the *projection index image* contains a number which is the index of the projection bin to which the pixel belongs. A histogram of the masked projection index image then contains in projection bin i the number of pixels which on the binary image have binary value one and which on the projection index image have index value i .

11. CURVE AND IMAGE DATA STRUCTURES

145. The *Fourier descriptors* of a closed planar curve are the coefficients of the Fourier series of the spatial positions of the curve as a function of arc length. Typically the low frequency coefficients are the ones of greatest interest.

146. *Iterative end point curve fitting* refers to an iterative process of segmenting a curve into a set of piecewise linear segments which approximate the curve. The process begins by constructing a straight line between the end points of the curve. If the furthest distance between the curve and the straight line is less than a specified tolerance then the approximation is considered to be suitable and the curve segment is divided no further. If the furthest distance between the curve and the straight line is greater than a specified tolerance, then the approximation is considered to be not suitable. The curve is then divided into two segments at this furthest distance point and the straight line fitting process independently continues on each segment.

147. The *chain code* representation of a digital arc or blob boundary is a sequence in which each element is a symbol representing the vector joining two neighboring pixels of the digital arc or blob boundary. The most common chain code uses the symbols 0 to 7 to represent the vectors (0, 1), (-1, 1), (-1, 0), (-1, -1), (0, -1), (1, -1), (-1, -1), and (-1, 0) of row column coordinates, which can join two neighboring pixels. More complex chain codes have more symbols and can represent the vector joining two more distant pixels which define the beginning and ending of a digital straight line segment which is part of a digital arc or blob boundary.

148. A *quadtree* is a tree data structure which represents an image. Each node of the quadtree represents a square subset of the image's spatial domain. The root node of the quadtree represents the spatial domain of the entire image. If all the pixels of the spatial domain subset represented by a node have the same value, then the value of the node is the value of the pixels in the subset. Such a node is called a pure node. If the node is a mixed or impure node, then the square represented by the node is partitioned into four quadrants and the node has four children nodes, one child node for each quadrant. If the image being represented is a binary image, then the corresponding quadtree is called a *binary quadtree*. If the image being represented is a gray scale image, then the corresponding quadtree is called a *grey scale quadtree*.

149. A *curve pyramid* consists of a sequence of symbolic images representing curves at multiple resolutions. The main operation in a bottom-up construction consists of locally connecting the short segments of the curve into longer ones. If the segments are described by binary "curve relations" of the labeled sides of a square cell, the concatenation of the short segments can be achieved formally by taking the transitive closure of the curve relations. Overlapping pyramids are necessary if the resulting pyramid is to have the "length reduction property": long curves with many segments survive to high levels, whereas short curves disappear after a few reduction steps.

150. An *octree* is a data in a tree data structure which represents a function defined as a three-dimensional space volume. Each node of the octree represents a cube subset of the volume. The root node of the quadtree represents the entire volume. If all the voxels represented by a node have the same function values, then the value of the node is their function value. Such a node is called a pure node. If the node is a mixed or impure node, then the cube represented by the node is partitioned into eight volume octants and the node has eight children, one child for each octant. If the function is binary, then the corresponding octree is called a *binary octree*. The binary octree is useful for representing three-dimensional volumes. If the function being represented is a non-binary, such as a real or integer valued function, then the corresponding octree is called a *gray scale octree*.

151. *Run length encoding* is a way to compactly represent binary images. There are a variety of run length encoding formats. Each format has a way of representing the starting column position of a maximally long horizontal string of binary one valued pixels as well as the number of pixels in a run. Many vision systems which recognize objects from their binary images use run length encoding to reduce the volume of data to be processed.

152. A *generalized cone* or *generalized cylinder* is a data structure for volumetric representation of a 3D object. The volume is generated by sweeping an arbitrarily shaped cross section along a 3D curve called the *generalized cone axis* or *generalized cylinder spine*. The axis passes through the centroids of the cross sections and is at a fixed angle (usually orthogonal) to them. The cross sectional shape is permitted to have some free parameters such as size or elongation. These values are specified for each axis pint by the *cross section function* or the *sweeping rule*.

153. *Constructive solid geometry* is a mechanism for representing three-dimensional volumes by a constructive process which begins with simple shaped volumes and which are combined and subtracted from each other by the set of operations consisting of union, intersection and set difference.

154. A *boundary surface description* or *boundary*

representation of a three-dimensional object or volume is a representation which contains each of the surface boundaries of the volume. Each surface boundary is represented in terms of simply described pieces, each of which has its own arc boundary. Each arc itself is represented in terms of simply described pieces which begin and terminate at end points or vertices.

155. A *superquadric* is a closed surface spanned by a vector whose x , y , and z components are specified as functions of the angles η and ω via the spherical product of two two-dimensional parametrized curves

$$h = \begin{pmatrix} h_1(\eta) \\ h_2(\eta) \end{pmatrix}$$

and

$$m = \begin{pmatrix} m_1(\omega) \\ m_2(\omega) \end{pmatrix}$$

which come from one of three basic trigonometric forms. The *spherical product* of h with m scaled by the vector $\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}$ is defined by

$$h \otimes m = \begin{pmatrix} a_1 h_1(\eta) m_1(\omega) \\ a_2 h_1(\eta) m_2(\omega) \\ a_3 h_2(\omega) \end{pmatrix}.$$

156. A *strip tree* is a binary tree data structure for hierarchically representing a planar arc segment. The root node of the tree represents the minimal sized rectangle which bounds the arc. Each non-terminal node of the tree splits the arc which its bounding rectangle approximates into two continuous pieces. Its two children nodes then each contain the minimal sized rectangle which bounds the arc segment piece belonging to it. Terminal nodes of the tree have bounding rectangles which are sufficiently close to the arc segment they contain.

157. The *primal sketch* is a data structure for representing gray level intensity changes, their geometrical distribution, and organization in each image neighborhood. Primitives of the primal sketch include zero-crossings, blobs, terminations and discontinuities, edge segments, virtual lines, groups, curvilinear organizations, and boundaries.

158. In *facet model* image processing, the digital image's pixel values are regarded as noisy discretized sampled observations of its underlying and unknown gray tone intensity surface. Any operation to be performed on the image is defined in terms of this underlying gray tone intensity surface. Thus, in order to do any processing, the underlying intensity surface must be estimated. This requires a model which describes what the general form of the surface would be in any image neighborhood if there were no noise. To estimate the surface from the neighborhood

around a pixel then amounts to estimating the free parameters of the general form. Useful image processing operations which then can be performed using the facet model processing approach include gradient edge detection, zero-crossing edge detection, image segmentation, line detection, corner detection, 3D shape estimation from shading, and determination of optic flow.

12. TEXTURE

159. A *discrete tonal feature* on a digital image is a connected set of resolution cells all of which have the same or almost the same image intensity.

160. *Texture* is concerned with the spatial distribution of the image intensities and discrete tonal features. When a small area of the image has little variation of discrete tonal features, the dominant property of that area is gray tone. When a small area has wide variation of discrete tonal features, the dominant property of that area is texture. There are three things crucial in this distinction: (1) the size of the small areas, (2) the relative sizes of the discrete tonal features, and (3) the number of distinguishable discrete tonal features. Texture can be described along dimensions of uniformity, density, coarseness, roughness regularity, intensity and directionality.

161. A *texel*, short for texture element, is a triplet whose first component is the (row, column) location of a small neighborhood, whose second component is the size of the neighborhood, and whose third component is the vector of texture properties.

162. The *gray level dependence matrix* or *gray level co-occurrence matrix* characterizes the micro-texture of an image region by measuring the dependence between pairs of gray levels arising from pixels in a specified spatial relation. For gray level pair (i, j) the gray level dependence matrix P for region R of image I has value $P(i, j)$ where $P(i, j)$ is the number of pairs of pixels in the region having the desired spatial relation where the first pixel has gray level i and the second pixel has gray level j . If S designates the set of all pairs of pixels in the desired spatial relation, then $S \subseteq R \times R$ and

$$P(i, j) = \#\{(r_1, c_1), (r_2, c_2) \in S \mid I(r_1, c_1) = i \text{ and } I(r_2, c_2) = j\}.$$

The gray level dependence matrix P can be normalized. One normalized form which produces a joint probability is given by

$$P_1(i, j) = \frac{P(i, j)}{\#S}.$$

A second normalized form which produces a conditional probability is given by

$$P_2(i, j) = \frac{P(i, j)}{\sum_i P(i, j)}.$$

163. *Statistical texture measures* include the

moments of the gray levels of the given region, typically the variance, the slewness and the kurtosis.

164. The *gray level difference histogram* at a distance d of an image I is the histogram of values

$$\langle |I(r, c) - I(r', c')| : (r - r')^2 + (c - c')^2 = d^2 \rangle$$

165. A *structural texture description* is given by a set of primitives and placement rules which govern the stochastic spatial relation between them.

166. *Fourier related texture descriptions* include the power spectrum and autocorrelation function.

13. SEGMENTATION

167. *Image segmentation* is a process which typically partitions the spatial domain of an image into mutually exclusive subsets, called regions, each one of which is uniform and homogeneous with respect to some property such as tone or texture and whose property value differs in some significant way from the property value of each neighboring region. Regions produced by an image segmentation process using image intensity as a property value produce regions which are called discrete tonal features.

168. *Region growing* refers to a sequential image segmentation process in which pixels are successively added to incomplete regions or initiate new regions when it is not appropriate to make them part of any of the existing incomplete regions. There are three basic kinds of region growing: region tracking, region aggregation, and region merging.

(a) 169. In *region tracking*, the image is scanned in raster scan order. The similarity of each pixel is compared with the regions to which the already processed 4- or 8-connected neighboring pixels belong. If one of these already processed neighboring pixels belongs to a region of sufficient similarity to the current pixel, then the pixel is added to the region. If the regions to which all the already processed pixels belong are dissimilar from the current pixel, then the current pixel initiates a new region.

(b) 170. In *region aggregation*, seed pixels are first found which serve as prototype pixels for the regions in the desired segmentation. Then in a sequential fashion, pixels having neighbors in any incomplete region join themselves into the region if they are similar enough. The aggregation process continues until all pixels are part of some region.

(c) 171. In *region merging*, the neighboring regions of an initial segmentation are successively merged together if they have similar enough properties. After each merging iteration, properties of the new regions are recomputed. The merging iterations continue until the properties of each pair of neighboring regions is sufficiently different from each other.

172. *Contour filling* is the process by which all pixels inside a blob defined by its bounding contour(s) are marked with the same unique label.

14. MATCHING

173. *Template matching* is an operation which can be used to find out how well a template subimage matches a window of a given image. The degree of matching is often determined by translating the template subimage all over the given image and for each position evaluating the cross-correlation or the sum of the squared or absolute image intensity differences of corresponding pixels. Template matching can also be used to best match an observed measurement pattern with a prototype pattern.

174. *Matched filtering* is a template matching operation done by using the magnitude of the cross-correlation function to measure the degree of matching.

175. *Image matching* refers to the process of determining the pixel by pixel, arc by arc, or region by region correspondence between two images taken of the same scene but with different sensors, different lighting, or a different viewing angle. Image matching can be used in the spectral/temporal pattern classification of remote sensing, or in determining corresponding points for stereo, tracking, change analysis, and motion analysis. In one group of approaches, subimages of one image are translated over a second image. For each translation, the differences between appropriately transformed gray tone intensities and/or edges are measured. In this signal level approach, the unit being operated with is the pixel, since the measured difference is between values of pixels in two images.

176. In *symbolic registration* or *symbolic matching*, higher level units are worked with. For example the scene can be segmented and a region matching then performed using segment features such as area, position, perimeter²/area, orientation, length to width ratio, area/area of minimum bounding rectangle, area/area of bounding ellipse, gray tone intensity or color of segment, and number of corresponding neighbors. Because this matching uses a higher level unit than the pixel, it is called symbolic matching or symbolic registration.

177. In *feature point matching*, selected points of each image are first determined, on the basis of the distinctive image values in a neighborhood or on the basis of an intersection between two feature lines. The location of each point can be to subpixel precision. After the locations of distinctive points are determined, a correspondence process associates as many as possible selected points of one image with selected points on the second image. The correspondence is based on similarity of the feature characteristics of the points.

178. A *structural description* is a relational representation of a 2D or 3D entity. It consists of a set of primitives each having its own attribute description and a set of named relations which consist of tuples whose components are primitives which stand in the relation specified by the relation name.

A function $h: A \rightarrow B$ is a *relation homomorphism* from N -ary relation $R \subseteq A^N$ to N -ary relation $S \subseteq B^N$ if

$$R \circ h \subseteq S$$

where $R \circ h = \{(b_1, \dots, b_N) \in B^N \mid \text{for some } (a_1, \dots, a_N) \in R, b_n = h(a_n), n = 1, \dots, N\}$.

179. The relation R is said to *match* relation S if there exists a relation homomorphism satisfying

$$R \circ h = S \quad \text{and} \quad S \circ h^{-1} = R$$

180. *Relational matching* refers to the process by which it is determined whether two relations match or do not match. *Structural matching* is a matching which establishes a correspondence or homomorphism from the primitives of one structural description to the primitives of a second structural description. In the ideal match a tuple of primitives which stand in a given relation in the first description will have its corresponding tuple of primitives stand in the same given relation in the second structural description.

181. The *local feature focus method* is a model based object recognition and location technique in which one feature, referred to as the focus feature, on the image is found and is used along with the object model to predict what other nearby features might be. After locating a set of features a relational matching is performed to infer a consistent correspondence between all the located image features and the object features. Once a consistent correspondence has been found, the object position orientation can be hypothesized. The hypothesized object position and orientation is then verified by template matching.

182. *Relaxation* refers to any computational mechanism which employs a set of locally interacting parallel processes, one associated with each image unit, which in an iterative fashion update each unit's current labeling in order to achieve a globally consistent interpretation of the image data. In *discrete relaxation*, the assessment of each unit's current state consists of that subset of labels not yet ruled out. In *probabilistic relaxation* the assessment of each unit's current state is a probability function associating with each possible label a probability of its being the correct state.

15. LOCALIZATION

183. An *interest operator* is a neighborhood operator which is designed to locate, with high spatial accuracy, pixel positions, or subpixel positions, whose central neighborhoods have distinctive gray tone patterns. Such neighborhoods are typically those whose autocorrelation functions falls off rapidly. Interest operators are usually used to mark pixels on a pair of images taken of the same scene but with some shift of either camera position or

object from one image to another. The marked pixels are then the candidate pixels which are input to an algorithm which establishes correspondences between the selected pixels.

184. An *area of interest operator* is an operator which delineates regions of an image which have potentially interesting patterns. These delineated *regions of interest* are the ones which must be further processed.

185. *Screening* is an operation of selecting photographs or images containing areas of potential interest from those in a set of photographs, some or most of which contain no interesting areas.

186. In *area analysis*, the area of the image containing the objects or entities to be processed is located by some simple algorithm and a more complex processing algorithm is only applied in the located area. This strategy of processing can often increase execution speed. The algorithm locating the area to be processed is called the *focus of attention mechanism*.

16. GENERAL IMAGE PROCESSING

187. *Preprocessing* is an operation applied before pattern identification is performed. Preprocessing produces, for the categories of interest, pattern features which tend to be invariant under changes such as translation, rotation, scale, illumination level, and noise. In essence, preprocessing converts the measurement patterns to a form which allows a simplification in the decision rule. Preprocessing can bring into registration, bring into congruence, remove noise, enhance images, segment target patterns, detect, center, and normalize objects of interest.

188. *Registering* or *registration* is the translation or translation and rotation alignment process by which two images of like geometries and of the same set of objects are positioned coincident with one another so that corresponding points of the imaged scene appear in the same position on the registered images. In this manner, corresponding image values can be made to represent the sensor output for the same object point over the full image frame.

189. *Congruencing* is the geometric warping process by which two images of different geometries, but of the same set of objects, are spatially transformed so that the size, shape, position, and orientation of any object on one image is made to be the same as the size, shape, position, and orientation of that object on the other image.

190. *Image compression* is an operation which preserves all or most of the information in the image and which reduces the amount of memory needed to store an image or the time needed to transmit an image.

191. *Image restoration* is a process by which a degraded image is restored, as clearly or as best as possible, to its ideal condition. Perfect image

restoration is possible only to the extent that the degradation transform is mathematically invertible. Common forms of restoration include *inverse filtering*, *Wiener filtering*, and *constrained least squares filtering*.

192. *Image reconstruction* refers to the process of reconstructing an image from a set of its projections. The projections may be taken along a set of parallel rays, in which case they are called *parallel projections*, or they may be taken along a set of rays emanating from a point, in which case they are called *fan beam projections*. The most commonly employed reconstruction techniques are the *filtered back projection* and the *algebraic reconstruction* techniques. Image reconstruction techniques are important in computerized tomography, nuclear medicine, and ultrasonic imaging.

193. *Image enhancement* is any one of a group of operations which improve the detectability of objects. These operations include, but are not limited to, contrast stretching, edge enhancement, spatial filtering, noise suppression, image smoothing, and image sharpening.

194. *Image processing* or *picture processing* encompasses all the various operations which can be applied to image data. These include, but are not limited to, image compression, image restoration, image enhancement, preprocessing, quantization, spatial filtering, matching, and recognition techniques.

195. *Interaction image processing* is carried out by an operator or analyst at a console with a means of accessing, preprocessing, feature extracting, classifying, identifying and displaying the original imagery or the processed imagery for subjective evaluation and further interactions.

17. VISION

196. *Structured light* refers to a technique of projecting a carefully designed light pattern on a scene and viewing the scene from a different direction. Usually the pattern consists of successive planes of light at different positions and orientations. Those pixels which image a surface patch which is lit by a known light pattern have sufficient information to determine the 3D coordinates of the surface patch since the light pattern is designed so that the line of sight passing through the pixel and the lens will intersect the known light pattern in a unique point. For stereo matching purposes, the structured light pattern may be "unstructured" in the sense of being a texture pattern or consisting of random stipples.

197. *Light striping* refers to a simple form of structured lighting in which the light pattern consists of successive planes of light which are all parallel.

198. *Stereopsis* refers to the capability of determining the depth of a 3D point by observing the point on two perspective projection images taken from different positions.

199. A *stereo image pair* refers to two perspective projection images taken of the same scene from slightly different positions. The common area appearing in both images of the stereo pair is usually 40% to 80% of the total image area.

200. A point p on one image and a point q on a second image are said to form a *corresponding point pair* (p, q) if p and q are each a different sensor projection of the same 3D point. The *visual correspondence* problem consists of matching all pairs of corresponding points from two images of the same scene.

201. *Disparity* or *stereo disparity* refers to the difference in positions of the images of the same 3D point in two perspective projection images taken from different positions.

202. *Stereo matching* refers to the matching process by which corresponding points on a stereo image pair are identified.

203. *Triangulation* refers to the process of determining the (x, y, z) coordinates of a 3D point from the observed position of two perspective projections of the point. The centers of perspectivity and the perspective projection planes are assumed known.

204. The *epipolar axis* of a stereo image pair is the line passing through the center of perspectivities of the image.

205. The two *epipoles* of a stereo image pair consist of one point on each of the perspective projection planes determined as the intersection of the image plane with the epipolar axis. For stereo image pairs having parallel perspective projection image planes, the epipoles are infinitely far to the left and right.

206. An *epipolar ray* is the line segment between an epipole and a point on the perspective projection image plane.

207. An *epipolar line* on one stereo image corresponding to a given point in another stereo image is the perspective projection on the first stereo image of the 3D ray which is the inverse perspective projection of the given point from the other stereo image.

208. An *epipolar plane* relative to a pair of stereo images is any plane determined by an observed 3D point, the position in 3D space of its perspective projection on the left stereo image and the position in 3D space of its perspective projection on the right stereo image. Every epipolar plane contains the epipolar axis and every plane which contains the epipolar axis is an epipolar plane.

209. An *occluding boundary* is a boundary appearing on an image due to a discontinuity in range or depth of an object in the observed scene.

210. For each fixed viewing position and point source of illumination, the *reflectance map* is a function defined on gradient space which specifies a surface's reflectivity. Thus, at each possible orientation of the surface normal (as encoded by the surface's first order partial derivatives) the surface's

reflectance map specifies the reflection coefficient of the surface.

211. *Shape from shading* refers to the capability of determining the 3D shape characteristics of an object from the gray tone shading manifested by the object's surface on a perspective projection or orthographic projection image, that is, from its reflective map.

212. *Local shading analysis* refers to the capability of inferring the shape and predominant tilt of a section of an object's surface by the image intensities in a local neighborhood of a perspective or parallel projection view.

213. *Photometric stereo* refers to the capability of determining surface orientation by means of the shading variations present on two or more images taken of the same scene from the same position and orientation but with the light source in different positions.

214. The *motion field* or *image flow* is an image in which the value of each pixel is the projected translational velocity arising from a surface point of an object in motion relative to the camera. Each projected translational velocity vector is called an *optic flow vector*.

215. An *optic flow* or *optical flow* image is an image in which the value of each pixel is the estimated projected translational velocity arising from a surface point of an object in motion relative to the camera. Because the projected velocity may not be estimable at each pixel, there may be some pixels in an optic flow image having no optic flow information.

216. The *focus of expansion* of a motion field image arising from a moving camera and stationary scene is that point on the image at which the optic flow is zero and such that the optic flows of the neighboring points are directed away from it. In cases of relative motion toward the camera, there will be exactly one focus of expansion point in such a motion field image. In a motion field image arising from an object in relative motion to the camera, the focus of expansion is that point on the image having all the optic flow vectors arising from the moving object directed away from it.

217. The *focus of contraction* of a motion field image arising from a moving camera and stationary scene is that point on the image at which the optic flow is zero and such that the optic flows of the neighboring points are directed toward it. In cases of relative motion away from the camera, there will be exactly one focus of contraction point in such a motion field image. In a motion field image arising from an object in relative motion to the camera, the focus of contraction is that point on the image having all the optic flow vectors arising from the moving object directed toward it.

218. *Structure from motion* refers to the capability of determining a moving object's shape characteristics, and its position and velocity as well, from a sequence of two or more images taken of the

moving object. Equivalently, if the camera is in motion, structure from motion refers to the capability of determining an object's shape characteristics, and its position and the camera's velocity, from a sequence of two or more images taken of the object by the moving camera. One fundamental kind of structure from motion problem is to determine the fixed position of M 3D points from a time sequence of N views and containing the 2d perspective projection of the M 3D points.

219. *Passive navigation* refers to the determination of the motion of a camera from a time varying image sequence.

220. *Dynamic scene analysis* refers to the analysis of time varying imagery. The purpose of the analysis may be to track moving objects, determine the motions of the objects, recognize the objects, determine the spatial positions of the objects at the time each image was obtained, or determine a shape description or characterization of one or more objects.

221. *Shape from texture* refers to the capability of determining the 3D shape characteristics of a homogeneously textured surface from the texture density variations manifested by the surface on a perspective projection or orthographic projection image.

222. *Surface reconstruction* refers to the process by which a 3D surface is analytically described by its boundary representation on the basis of processing a stereo image pair, a range map, or a time varying image sequence of the observed surface.

223. *Shape from contour* or *shape from shape* refers to the capability of inferring the 3D shape of an object from a 2D perspective projection view of a set of regularly marked contours on the object's surface.

224. The $2\frac{1}{2}D$ *sketch* is a multiband image, each pixel providing information about the depth and surface orientation of the surface projected on it as well as providing an indication of the existence of a nearby depth discontinuity and an indication of the existence of a nearby surface discontinuity.

225. The *intrinsic scene characteristics* for each object surface point are: its depth from the image focal plane, its surface orientation, its reflectance, and its incident illumination.

226. An *intrinsic image* is a multiband image in which each pixel contains the predominant intrinsic scene characteristics of the surface patch projecting to its position. Hence each pixel of the intrinsic image specifies depth, surface orientation, reflectance, and incident illumination for the surface patch projecting to its position.

227. *Inverse optics* refers to the capability of inferring the 3D position and/or the surface normal of each point on an object's surface and/or the surface shape from one or more perspective projection images of the object. Included in the techniques of inverse optics are stereo, photometric stereo, shape

from shading, shape from texture, and structure from motion. Inverse optics techniques can be thought of as techniques which invert the perspective projection process and which, therefore, belong to the reconstructionist school of applied physics computer vision.

228. The *blocks world* refers to a world in which all objects have simple surfaces. The most common kind of objects in the blocks world are polyhedral objects.

229. A *view aspect* is a maximal connected region of viewpoint space having the property that when looking at a given object's center from any point of a view aspect, the resulting views are topologically identical.

230. An *aspect graph* is a graph in which the nodes are the view aspects and the arcs connect adjacent view aspects. The views represented by the nodes in the graph are all the stable views, characteristic views and principal views.

231. *3D vision* refers to the capability of a machine vision system to be able to infer some 3D characteristic of an object or object feature such as its position, dimensions, orientation or motion or some 3D characteristic of a point or an object surface such as its 3D position or surface normal orientation.

232. *Automatic visual inspection* or *automatic vision inspection* refers to an inspection process which uses a sensor producing image data and which uses techniques from image processing, pattern recognition, or computer vision to measure and/or interpret the imaged objects in order to determine whether they have been manufactured within permitted tolerances. Automatic visual inspection systems usually integrate the technologies of material handling, illumination, image acquisition and special purpose computer hardware along with the appropriate image analysis algorithms into a system intended to be of practical use in the factory. Benefits from using automated visual inspection can include more accurate, reliable, repeatable, and complete quality assurance at a lower price and a higher speed of inspection than possible by manual labor.

233. A *machine vision system* is a system capable of acquiring one or more images of an object, capable of processing, analysing and measuring various characteristics of the acquired images, and interpreting the results of the measurements in such a way that some useful decision can be made about the object. Functions of machine vision systems include locating, inspecting, gauging, identifying, recognizing, counting, and motion estimating.

234. *Visual fixturing* or *visual pose determination* refers to the capability of inferring the position and orientation of a known object using a suitable object model and one or more cameras, range sensors, or triangulation based vision sensors.

235. *Optical gauging* or *visual gauging* refers to the ability to measure specific positions or dimensions of a manufactured object by using non-contact light

sensitive sensors to compare these measurements to preselected tolerance limits for quality inspection and sorting decisions. Gauging has wide application in manufacturing since it can determine the diameters of holes, openings, or cutouts, the widths of shafts, components, gaps, wires, or rods, and the relative locations of holes, folds, features, components, openings, or breaks.

236. A gauging system can be either a *fixed inspection* or *flexible inspection* system. The fixed inspection system holds the part in a precision test fixture and has one or more sensors to take the required measurements. Flexible inspection utilizes sensors that are moved about the part being inspected, the motion being done along a programmed path trajectory. Flexible inspection gauging is sometimes called *robotic gauging*.

237. A vision procedure is said to be *robust* if small changes in the assumed model on which the procedure or technique was developed produce only small changes in the result. Small fractions of the data which do not fit the assumed model and which in fact are very far from fitting the assumed model, constitute a small change in the assumed model. Data not fitting an assumed model may be due to rounding or quantizing errors, gross errors, or because the model itself is only an idealized approximation to reality.

238. *Computer vision, image understanding, or scene analysis* is that combination of image processing, pattern recognition, and artificial intelligence technologies which focuses on the computer analysis of one or more images, taken with a single/multiband sensor or taken in time sequence. The analysis recognizes, locates the position and orientation, and provides a sufficiently detailed symbolic description or recognition of those imaged objects deemed to be of interest in the three-dimensional environment. The computer vision process often uses geometric modeling and complex knowledge representations in an expectation or model based matching or searching methodology. The searching can include bottom up, top down, blackboard, hierarchical, and heterarchical control strategies.

239. The *bottom up* control strategy is an approach to problem solving that is *data driven*. It employs no object models in its early stages and only uses general knowledge about the world being sensed. In a computer vision system using a bottom up control strategy, the observed image data is interpreted and aggregated. The interpretations and aggregations are then successively manipulated and aggregated until a sufficiently high level description of the scene has been generated.

240. The *top down* control strategy is an approach to problem solving that is *goal-directed* or *expectation directed*. A form of solution is generated or hypothesized. Assuming the hypothesis is true and using the information in the knowledge data base, the inference mechanism then infers, if possible, some

consistent set of values for the unknown variables or parameters. If a consistent set can be inferred, then the problem has been solved. If a consistent set cannot be inferred, then a new form of solution is generated or hypothesized. In a computer vision system using a top down control structure, the number or types of objects being sensed in the image is usually highly constrained and knowledge about the objects, relationships between objects, and object parts are all known. The system hypothesizes that the image shows a particular set of objects, infers values for parameters, and then tests to verify that the hypothesis is consistent with the observed data.

241. A *hierarchical* control strategy is an approach to problem solving in which the given problem is solved by dividing it up into a set of subproblems, each of which encapsulates an important or major aspect of the original problem. Then each subproblem is successively divided into more detailed subproblems. The refinement continues until the most refined subproblems can be solved directly.

242. A *blackboard* control strategy is an approach to problem solving in which the various components of the inference mechanism communicate with one another through a common working data storage area called the blackboard. When the blackboard has sufficient data to permit one component of the inference mechanism to make a deduction, the inference mechanism goes to work and writes its results on the blackboard where it becomes available for the other components of the inference mechanism. In this manner the inferred constraints are successively propagated and the required search is made more limited.

243. *Model based computer vision* is a computer vision process which employs an explicit model of the object to be recognized. Recognition proceeds in a top down manner by matching the object data structure inferred from the observed image to the model data structure.

244. *Knowledge based vision* refers to a computer vision process which has an image processing component, a reasoning or inference component, and a knowledge data base component. The knowledge data base stores information about the environment being imaged. The image processing component extracts primitive point, line, curve, and region information from an observed image. The reasoning or inference component is typically rule based and integrates the information produced by the image processing component with the information in the knowledge data base and reasons about what hypothesis should be next generated, what hypothesis should be next validated, what new information can be inferred from what has already been established, and what new primitives the image processing component should extract next.

18. PATTERN RECOGNITION

245. *Pattern recognition* techniques can be used to

construct decision rules which enable units to be identified on the basis of their measurement patterns. Pattern recognition techniques can also be employed to cluster together units having similar enough measurement patterns. In *statistical pattern recognition*, the measurement patterns have the form of n -tuples or vectors. In *syntactic pattern recognition*, the measurement patterns have the form of sentences from the language of a phrase structure grammar. In *structural pattern recognition*, the measurements do not have the form of an n -tuple or vector. Rather, the unit being measured is encoded in terms of its parts and the relationships as well as properties of the parts.

246. *Pictorial pattern recognition* refers to techniques which treat the image as a pattern and either categorize the image or produce a description of the image.

247. The *unit* is the entity which is observed and whose measured properties constitute the measurement pattern. The simplest and most practical unit to observe and measure in the pattern recognition of image data is often the pixel (the gray tone intensity or the gray tone intensity n -tuple in a particular resolution cell). This is what makes pictorial pattern recognition so difficult, because the objects requiring analysis or identification are not single pixels but are often complex spatial formations of pixels.

248. A *measurement pattern* or *pattern* is the data structure of the measurements resulting from observing a unit.

249. A *measurement n -tuple* or *measurement vector* is the ordered n -tuple of measurements obtained from a unit under observation. Each component of the n -tuple is a measurement of a particular quality, property, feature, or characteristic of the unit. In image pattern recognition, the units are usually picture elements or simple formations of picture elements and the measurement n -tuples are the corresponding gray tone intensities, gray tone intensity n -tuples, or properties of formations of gray tone intensities.

250. The *Cartesian product* of two sets A and B , denoted by $A \times B$, is the set of all ordered pairs where the first component of the pair is some element from the first set and the second component of the pair is some element from the second set. The Cartesian product of N sets can be defined inductively.

251. *Measurement space* is a set large enough to include in it the set of all possible measurement patterns which could be obtained by observing some set of units.

252. The *range set* R_i for the i th sensor, which produces the i th image in the multi-image set, is the set of all measurements which can be produced by the i th sensor. Simply, it is the set of all gray tone intensities which could possibly exist in the i th image. When the units are the pixels, measurement space M is the Cartesian product of the range sets of the sensors: $M = R_1 \times R_2 \times \dots \times R_n$.

253. Each unit is assumed to be of one and only one given type. The set of types is called the set of *pattern classes* or *categories* C , each type being a particular category.

254. A *feature*, or *feature pattern*, or *feature n -tuple*, or *feature vector* or *pattern feature* is a n -tuple or vector whose components are functions of the initial measurement pattern variables or some subset of the initial measurement pattern variables. Feature n -tuples or vectors are designed to contain a high amount of information relative to the discrimination between units of the types of categories in the given category set. Sometimes the features are predetermined and other times they are determined at the time the pattern discrimination problem is being solved. In image pattern recognition, features often contain information relative to gray tone intensity, texture, or region shape.

255. *Feature space* is the set of all possible feature n -tuples.

256. *Feature selection* is the process by which the features to be used in the pattern recognition problem are determined. Sometimes feature selection is called *property selection*.

257. *Feature extraction* is the process by which an initial measurement pattern or some subset of measurement patterns is transformed to a new pattern feature. Sometimes feature extraction is called *property extraction*.

The word *pattern* can be used in three distinct senses:

1. as measurement pattern;
2. as feature pattern; and
3. as the dependency pattern or patterns of relationships among the components of any measurement n -tuple or feature n -tuple derived from units of a particular category and which are unique to those n -tuples, that is, they are dependencies which do not occur in any other category.

258. A *classifier* is a device or process that sorts patterns into categories or classes.

259. The *compactness hypothesis* states that the pattern measurements of a given class are nearer to other pattern measurements in the class than they are to pattern measurements from other classes.

260. The region of space occupied by pattern measurements coming from the same class or category is called a *class region*.

261. Two classes or categories are said to be *separable* if their class regions do not overlap. If for every class region there exists a hyperplane which separates the class region from all other class regions, the classes are said to be *linearly separable*.

262. A *prototype pattern* or *reference pattern* is the observable or characteristic measurement or feature pattern derived from units of a particular category. A category is said to have a prototype pattern only if the characteristic pattern is highly representative of the n -tuples obtained from units of that category.

263. A *data sequence* $S_d = \langle d_1, d_2, \dots, d_n \rangle$ is a

sequence of patterns derived from the measurement patterns or features of some sequence of observed units. d_1 is the pattern associated with first unit; d_2 is the pattern associated with the second unit; and d_j is the pattern associated with the j th unit.

264. A *decision rule* f usually assigns one and only one category to each observed unit on the basis of the sequence of measurement patterns in the data sequence S_d or on the basis of the corresponding sequence of feature patterns.

265. A *simple decision rule* is a decision rule which assigns a category to a unit solely on the basis of the measurements or features associated with the unit. Hence, the units are treated independently and the decision rule f may be thought of as a function which assigns one and only one category to each pattern in measurement space or to each feature in feature space.

266. A *hierarchical decision rule* is a decision rule in a tree form. In binary trees, each non-terminal node of the tree contains a simple decision rule which classifies patterns as belonging to its left child or to its right child. Each terminal node of the tree contains the assigned class or category of the observed unit.

267. A *compound decision rule* is a decision rule which assigns a unit to a category on the basis of some non-trivial subsequence of measurement patterns in the data sequence or in the corresponding sequence of feature patterns.

268. Provision can be made for a decision rule to *reserve judgment* or to *defer assignment* if the pattern is too close to the category boundary in measurement or feature space. With this provision, a deferred assignment is an assignment to the category of "reserved judgment".

269. A *category identification sequence* or *ground truth* $S_c = \langle c_1, c_2, \dots, c_j \rangle$ is a sequence of category identifications obtained from some sequence of observed units. c_1 is the category identification of the first unit; c_2 is the category identification of the second unit; and c_j is the category identification of the j th unit.

270. A *training sequence* is a set of two sequences: (1) the data sequence and (2) a corresponding category identification sequence. A training datum is the pair consisting of a pattern in the data sequence and the corresponding category identification in the category identification sequence. The training sequence is used to estimate the category conditional probability distributions from which the decision rule is constructed or it may be used to estimate the decision rule itself.

271. A *training procedure* is a procedure which uses the training sequence to construct a decision rule. It may operate by passing through the entire training sequence one time and construct the decision rule in a manner which is independent of the order in which the training data occurs in the training sequence. It may operate iteratively in which case it passes through the training sequence many times and

after handling each training datum, it modifies or updates the decision rule. Such iterative training procedures may be affected by the order in which the training data occurs in the training sequence.

272. A *window training procedure* is an iterative training procedure in which each adjustment of the decision rule is made only when the training datum falls within a specified window, a subset of the pattern measurement space. Usually this subset or window contains the decision boundary.

273. An *error correcting training procedure* is an iterative sequential training procedure in which at each iteration the decision rule is adjusted in response to a misclassification of a training datum.

274. A classifier is said to *learn* if its iterative training procedure increases the classification performance accuracy of the classifier after each few iterations.

275. The *conditional probability* of a measurement or feature n -tuple d given category c is usually denoted by $P_c(d)$, or by $P(d|c)$, and is defined as the relative frequency or proportion of times the n -tuple d is derived from a unit whose true category identification is c .

276. A *distribution-free* or *non-parametric decision rule* is one which makes no assumptions about the functional form of the conditional probability distribution of the patterns given the categories.

277. A simple *maximum likelihood decision rule* is one which treats the units independently and assigns a unit u having pattern measurements or features d to that category c whose units are most probable to have given rise to pattern or feature vector d , that is, such that the conditional probability of d given c is highest.

278. A simple *Bayes decision rule* is one which treats the units independently and assigns a unit u having pattern measurements or features d to the category c whose conditional probability, given measurements d , is highest.

279. Let $\langle u_1, u_2, \dots, u_j \rangle$ be a sequence of units with corresponding data sequence $\langle d_1, d_2, \dots, d_j \rangle$ and known category identification sequence $\langle c_1, c_2, \dots, c_j \rangle$. A simple *nearest neighbor decision rule* is one which treats the units independently and assigns a unit u of unknown identification and with pattern measurements or features d to category c_j where d_j is that pattern closest to d by some given metric or distance function.

280. A *discriminant function* $f_i(d)$ is a scalar function, whose domain is usually measurement space and whose range is usually the real numbers. When $f_i(d) \geq f_k(d)$, for $k = 1, 2, \dots, K$, then the decision rule assigns the i th category to the unit giving rise to pattern d .

281. A *linear discriminant function* f is a discriminant function of the form $f(d) = \sum_{j=1}^n a_j \delta_j + a_0$ where $d = (\delta_1, \delta_2, \dots, \delta_n)$ represents the measurement pattern.

282. A *quadratic discriminant function* f is a discriminant function of the form $f(d) = \sum_{i=1}^n \sum_{j=i}^n a_{ij} \delta_i \delta_j + \sum_{j=1}^n a_j \delta_j + a_0$.

283. A *decision boundary* between the i th and k th categories is a subset H of patterns in measurement space M defined by

$$H = \{d \in M | f_i(d) = f_k(d)\},$$

where f_i and f_k are the discriminant functions for the i th and k th categories.

284. A *hyperplane decision boundary* is the special name given to decision boundaries arising from the use of linear discriminant functions.

285. A *linear decision rule* is a simple statistical pattern recognition decision rule which usually treats the units independently and makes the category assignments using linear discriminant functions. The decision boundaries obtained from linear decision rules are hyperplanes.

286. The *pattern discrimination* problem is concerned with how to construct the decision rule which assigns a unit to a particular category on the basis of the measurement pattern(s) in the data sequence or on the basis of the feature pattern(s) in the data sequence.

287. *Pattern identification* is the process in which a decision rule is applied. If $S_u = \langle u_1, u_2, \dots, u_j \rangle$ is the sequence of units to be observed and identified, and if $S_d = \langle d_1, d_2, \dots, d_j \rangle$ is the corresponding data sequence of patterns, then the pattern identification process produces a category identification sequence $S_c = \langle c_1, c_2, \dots, c_j \rangle$ where c_i is the category in C to which the decision rule assigns unit u_i on the basis of the j patterns in S_d . In general, each category in S_c can be assigned by the decision rule as a function of all the patterns in S_d . Sometimes pattern identification is called *pattern classification* or *classification*.

288. A *perceptron* or *neural network* is an interconnected network of nonlinear units or processing elements capable of learning and self organizing. The response of a unit or a processing element is a non-linear monotonic function of a weighted sum of the inputs to the processing elements. The weights, called *synaptic weights* are modified by a learning or reinforcement algorithm. Typical nonlinear processing functions are $\text{sgn}(x)$, $\frac{1}{1 + e^{-x}}$, and $\tanh(x)$.

When each processing element contributes one component to the output response vector and its inputs are selected from the components of the input pattern vector, the perceptron is called a *simple perceptron*. Processing units whose output only indirectly influences the components of the output response vector are called *hidden units*.

289. An *error corrective reinforcement* for a perceptron is a learning or training algorithm in which the change in a synaptic weights is a function of the

degree to which the output of the processing unit is not what it is desired to be. Error corrective reinforcement algorithms are also called *error back propagation* algorithms.

290. A *forward coupled perceptron* is a perceptron in which the processing units are layered. The inputs to the processing units in layer n come from the outputs of processing units in layers prior to layer n . Single layered perceptrons can create linear decision surfaces. Two layered perceptrons can create convex decision regions. Three layered perceptrons can create almost arbitrarily shaped decision regions.

291. A *series coupled perceptron* is a forward coupled perceptron in which the inputs to the processing center in layer n come from the outputs of processing units in layer $n - 1$.

292. A *back coupled perceptron* is a perceptron which is not forward coupled. That is, there is some processor in layer n where output feeds back and is the input to a processor in some layer prior to layer n .

293. A *cluster* is a homogeneous group of units which are very "like" one another. "Likeness" between units is usually determined by the association, similarity, or distance between the measurement patterns associated with the units.

294. A *cluster assignment function* is a function which assigns each observed unit to a cluster on the basis of the measurement pattern(s) in the data sequence or on the basis of their corresponding features. Sometimes the units are treated independently. In this case the cluster assignment function can be considered as a transformation from measurement space to the set of clusters.

295. The *pattern classification* problem is concerned with constructing the cluster assignment function which groups similar units. Pattern classification is synonymous with *numerical taxonomy* or *clustering*.

296. The *cluster identification* process is the process in which the cluster assignment function is applied to the sequence of observed units thereby yielding a cluster identification sequence.

297. A *misidentification*, or *misdetetection*, or *type I error* occurs for category c_i if a unit whose true category identification is c_j is assigned by the decision rule to category c_k , $k \neq i$. A misidentification error is often called an *error of omission*.

298. A *false identification*, or *false alarm*, or *type II error* occurs for category c_i if a unit whose true category identification is c_k , $k \neq i$, is assigned by the decision rule to category c_i . A false identification error is often called an *error of commission*.

299. A *prediction sequence*, or *test sequence*, or a *generalization sequence* is a set of two sequences: (1) data sequence (whose corresponding true category identification sequence may be considered to be unknown to the decision rule) and (2) a corresponding category identification sequence deter-

mined by the decision rule assignment. By comparing the category identification sequence determined by the decision rule assignment with the category identification sequence determined by the ground truth, the misidentification rate and the false identification rate for each category may be estimated.

300. A *confusion matrix* or *contingency table* is an array of probabilities whose rows and columns are both similarly designated by category label and which indicates the probability of correct identification for each category as well as the probability of type I and type II errors. The (*i*th, *k*th) element P_{ik} is the probability that a unit has true category identification c_i and is assigned by the decision rule to category c_k .

301. A unit is said to be *detected* if the decision rule is able to assign it as belonging only to some given subset A of categories from the set C of categories. To detect a unit does not imply that the decision rule is able to identify the unit as specifically belonging to one particular category.

302. A unit is said to be *recognized*, *identified*, *classified*, *categorized*, or *sorted* if the decision rule is able to assign it to some category from the set of given categories. In some applications, there may be a definite distinction between recognize and identify. In these applications, for a unit to be recognized, the decision rule must be able to assign it to a type of category, the type having included within it many subcategories. For a unit to be identified, the decision rule must be able to assign it not only to a type of category but also to a subcategory of the category type. For example, a small area ground patch which may be recognized as containing trees may be specifically identified as containing apple trees.

303. A unit is said to be *located* if specific coordinates can be given for the unit's physical location.

304. *Accuracy* refers to the degree of closeness an estimate has to the true value of what it is estimating.

305. *Precision* refers to the degree of closeness an estimate has to its expected value.

306. The *receiver operating characteristic* or the *receiver operating curve* of a pattern classifier is a function of its misdetection error rate against its false alarm rate.

307. The *leave-K-out* method of evaluating a pattern classifier divides the training set into *L* mutually exclusive subsets each having *K* patterns. The classifier is successively trained using *L* - 1 of the subsets and tested on the *L*th subset. The evaluation is then made on the accumulated performance tests of the experiments where in each experiment *K* patterns were omitted from the training set and then used in the testing set. Performance estimates obtained using the leave-*K*-out method are unbiased. However, for small *K* the estimates will have high variance.

308. The *resubstitution method* of evaluating a pattern classifier uses the same set for training and testing. Performance estimates obtained using the resubstitution method are always biased high.

19. INDEX OF TERMS

Absolute Orientation	43
Accuracy	304
Acutance	6
Algebraic Reconstruction	192
Ambient Light	35
Analytic Photogrammetry	39
Antiparallel	86
Area	128
Area Analysis	186
Area of Interest Operator	184
Aspect Graph	230
Automatic Vision Inspection	232
Automatic Visual Inspection	232
Back Coupled Perceptron	292
Backlighting	33
Band Pass Filter	79
Bayes Decision Rule	278
Bidirectional Reflectance Distribution Function	30
Binary Image	20
Binary Image Projection	144
Binary Octree	150
Binary Quadtree	148
Blackboard	242
Blob	113
Blob Analysis	142
Block	106
Blocks World	228
Border	122
Bottom Up	239
Boundary	122
Boundary Delineation	84
Boundary Detection	84
Boundary Following	123
Boundary Representation	154
Boundary Surface Description	154
Bounding Contour	120
Bounding Rectangle	136
Box Filter	71
Brightness	3
Candela	24
Camera Constant	47
Cartesian Product	250
Categories C	253
Categorized	302
Category Identification Sequence	269
Center of Perspective	46
Central Moment	131
Centroid	129
Chain Code	147
Change Detection	65
Chromaticity	36
City Distance	106
Classification	287
Classified	302
Classifier	258
Class Region	260
Closing	97

Cluster	293	Elongatedness	139
Cluster Assignment Function	294	Elongation	139
Cluster Identification	296	Epipolar Axis	204
Clustering	295	Epipolar Line	207
Compactness	135	Epipolar Plane	208
Compactness Hypothesis	259	Epipolar Ray	206
Compound Decision Rule	267	Epipoles	205
Computer Vision	238	Equal Interval Quantizing	60
Concurve	124	Equal Probability Quantizing	61
Conditional Probability	275	Eroding	95
Confusion Matrix	300	Error Back Propagation	289
Congruencing	189	Error Correcting Training Procedure	273
Connected	112	Error Corrective Reinforcement	289
Connected Component	113, 141	Error of Commission	298
Connected Component Analysis	142	Error of Omission	297
Constrained Least Squares Filtering	191	Euclidean Distance	105
Constructive Solid Geometry	153	Euler Number	134
Contingency Table	300	Expectation Directed	240
Contour Filling	172	Extended Gaussian Image	104
Contour Tracing	126	Exterior Border Pixel	117
Contrast	7	Exterior Orientation	42
Contrast Difference	7	Extremal Pixel	137
Contrast Modulation	7	Facet Model	158
Contrast Ratio	7	False Alarm	298
Contrast Stretching	58	False Identification	298
Convex	115	Fan Beam Projections	192
Convolving	73	Feature	254
Correlating	75	Feature Extraction	257
Corresponding Point Pair	200	Feature N -Tuple	254
Cross Section Function	152	Feature Pattern	254
Curve Pyramid	149	Feature Point Matching	177
Data Driven	239	Feature Selection	256
Data Sequence	263	Feature Space	255
Decision Boundary	283	Feature Vector	254
Decision Rule	264	Figure F	109
Defer Assignment	268	Filtered Back Projection	192
Density Slicing	56	Fixed Inspection	236
Depth Map	15	Flexible Inspection	236
Detected	301	Focus of Attention Mechanism	186
Digital Image	13	Focus of Contraction	217
Digital Photogrammetry	40	Focus of Expansion	216
Digital Picture Function	13	Foot Candle	25
Digital Straight Line Segment	114	Forward Coupled Perceptron	290
Digitized Image	13	Four-Connected	111
Dilating	94	Fourier Descriptors	145
Discrete Fourier Transform	76	Fourier Related Texture Descriptions	166
Discrete Radon Transform	100	Frontlighting	34
Discrete Relaxation	182	Gaussian Filter	72
Discrete Tonal Feature	159	Gaussian Focal Length	47
Discriminant Function	280	Gaussian Image Pyramid	80
Disparity	201	Gaussian Sphere	102
Distance Transform	108	Generalization Sequence	299
Distribution-Free	276	Generalized Cone	152
Dynamic Imagery	19	Generalized Cone Axis	152
Dynamic Scene Analysis	220	Generalized Cylinder	152
Edge Image	82	Generalized Cylinder Spine	152
Edgel	10	Goal-Directed	240
Edge Linking	83	Gradient Operators	81
Edge Operator	81	Gradient Space	103
Eight-Connected	113	Gray Level	3

Gray Level Co-occurrence Matrix	162	Knowledge Based Vision	244
Gray Level Difference Histogram	164	Lambertian Surface	32
Gray Level Dependence Matrix	162	Laplacian Image Pyramid	80
Gray Level Image	21	Laplacian Operators	81
Gray Scale Image	21	Learn	274
Gray Scale Octree	150	Leave-K-Out	307
Gray Scale Quadtree	148	Level Slicing	56
Gray Shade	3	Light Striping	197
Gray Tone	3	Limiting Resolution	4
Gray Tone Intensity	3	Linear Decision Rule	285
Ground Truth	269	Linear Discriminant Function	281
Hidden Units	288	Linear Quantizing	60
Hierarchical	241	Linear Spatial Filter	69
Hierarchical Decision Rule	266	Linearly Separable	261
High Pass Filter	77	Local Feature Focus Method	181
Histogram	23	Local Shading Analysis	212
Histogram Equalization	61	Located	303
Histogram Hyperbolization Quantizing	63	Low Pass Filter	78
Hole	121	Lumen	24
Homomorphic Filtering	87	Luminous Flux	24
Hough Transform	101	Luminous Intensity	24
Hue Saturation	37	Lux	25
Hyperplane Decision Boundary	284	Machine Vision System	233
Identified	302	Masking	64
Illuminance	26	Match	179
Illumination	25	Matched Filtering	174
Image	1	Mathematical Morphology	92
Image Compression	190	Max Distance	107
Image Density	3	Maximum Likelihood Decision Rule	277
Image Enhancement	193	Measurement <i>N</i> -Tuple	249
Image Flow	214	Measurement Pattern	248
Image Histogram	23	Measurement Space	251
Image Intensity	3	Measurement Vector	249
Image Matching	175	Medial Axis	140
Image Operator	53	Median Filter	88
Image Processing	194	Minimum Perimeter Polygon	125
Image Pyramid	80	Minimum Variance Quantizing	62
Image Reconstruction	192	Misdetection	297
Image Restoration	191	Misidentification	297
Image Segmentation	167	Model Based Computer Vision	243
Image Smoothing	89	Moment	130
Image Time Sequence	19	Morphologic Edge Operators	81
Image Transform	53	Morphological Image Pyramid	80
Image Transform Operator	53	Motion Field	214
Image Understanding	238	Multi-Band Image	17
Image Value	3	Multi-Image Set	17
Inner Orientation	48	Multi-Level Thresholding	57
Interaction Image Processing	195	Multi-Spectral Image	18
Interest Operator	183	Multi-Temporal Image	19
Interior Border Pixel	116	Nearest Neighbor Decision Rule	279
Interior Pixel	118	Neighborhood Operator	67
Internal Orientation	48	Neural Network	288
Intrinsic Image	226	Non-Parametric Decision Rule	276
Intrinsic Scene Characteristics	225	Normalized Central Moment	132
Inverse Discrete Fourier Transform	76	Numerical Taxonomy	295
Inverse Filtering	191	Occluding Boundary	209
Inverse Optics	227	Occluding Edge	85
Irradiance	29	Octree	150
Iterative End Point Curve Fitting	146	Opening	96
Kernel	70	Optical Flow	215

Optical Gauging	235	Region Aggregation	170
Optic Axis	44	Region Growing	168
Optic Flow	215	Region Merging	171
Optic Flow Vector	214	Region R	110
Orientation Histogram	104	Regions of Interest	184
Orientation Map	16	Region Tracking	169
Orthographic Projection	50	Registering	188
Outer Orientation	42	Registration	188
Parallax	51	Relational Matching	180
Parallel	50	Relation Homomorphism	178
Parallel Projections	192	Relative Orientation	41
Passive Navigation	219	Relaxation	182
Pattern	248, 257	Reserve Judgment	268
Pattern Classes	253	Resolution	4
Pattern Classification	287, 295	Resolution Cell	5
Pattern Discrimination	286	Resolution Limit	4
Pattern Feature	254	Resubstitution Method	308
Pattern Identification	287	Robotic Gauging	236
Pattern Recognition	245	Robust	237
Pel	8	Rotation Movement Moments	133
Perceptron	288	Run Length Encoding	151
Perimeter	127	Scale Space Image	90
Perspective Projection	49	Scale Space Structure	91
Photometric Stereo	213	Scene Analysis	238
Pictorial Pattern Recognition	246	Screening	185
Picture Element	8	Second Moment Matrix	138
Picture Processing	194	Separable	74, 261
Pixel	8	Series Coupled Perceptron	291
Point Operator	54	Shape From Contour	223
Precision	305	Shape From Shading	211
Prediction Sequence	299	Shape From Shape	223
Preprocessing	187	Shape From Texture	221
Primal Sketch	157	Signature Analysis	143
Principal Axis	44	Simple Boundary	119
Principal Distance	47	Simple Decision Rule	265
Principal Point	45	Simple Perceptron	288
Probabilistic Relaxation	182	Skeleton	98
Projection Index Image	144	Sorted	302
Property Extraction	257	Spatial Filter	68
Property Selection	256	Spatial Frequencies	76
Prototype Pattern	262	Spatial Resolution	4
Pyramid	80	Spherical Product	155
Quadratic Discriminant Function	282	Square	107
Quadtree	148	Statistical Pattern Recognition	245
Quantizing	59	Statistical Texture Measures	163
Radiance	27	Step Edge Operator	81
Radiant Intensity	28	Stereo Disparity	201
Range Image	12	Stereo Image Pair	199
Range Map	15	Stereo Matching	202
Range Set	252	Stereopsis	198
Rangel	14	Strip Tree	156
Raster Scan Order	11	Structural Description	178
Receiver Operating Curve	306	Structural Matching	180
Receiver Operating Characteristic	306	Structural Pattern Recognition	245
Recognized	302	Structural Texture Description	165
Reference Pattern	262	Structured Light	196
Reflectance	30	Structure From Motion	218
Reflectance Image	31	Structuring Element	93
Reflectance Map	31, 210	Subimage F	109
Reflection Coefficient	30	Superquadric	155

Surface Reconstruction	222	Triangulation	203
Sweeping Rule	152	Two and a Half D Sketch	224
Symbolic Image	22	Two-Dimensional Signal Processing	66
Symbolic Registration	176	Type I Error	297
Symbolic Matching	176	Type II Error	298
Symmetric Axis	140	Unit	247
Synaptic Weights	288	Vanishing Point	52
Syntactic Pattern Recognition	245	Video Image	2
Template Matching	173	View Aspect	229
Test Sequence	299	Visual Correspondence	200
Texel	161	Visual Fixturing	234
Texture	160	Visual Gauging	235
Thickening Operator	99	Visual Pose Determination	234
Thinning Operator	98	Voxel	9
Three-Dee Vision	231	Wiener Filtering	191
Thresholding	55	Window Training Procedure	272
Time Varying Image	19	X-parallax	51
Top Down	240	<i>YIQ</i> Coordinate	38
Training Procedure	271	Y-parallax	51
Training Sequence	270	Zero Crossing Operators	81