Inspection with machine vision techniques

Liu, Zheng; Ukida, Hiroyuki

Publisher's version / Version de l'éditeur:

Visual Testing (VT), 2010

INSPECTION WITH MACHINE VISION TECHNIQUES

Zheng Liu and Hiroyuki Ukida

Machine vision is the application of computer vision to industry and manufacturing. It is a specialization within system engineering, which encompasses computer science, optics, mechanical engineering and industrial automation. According to Batchelor and Whelan[1], machine vision is the "use of devices for optical, non-contact sensing to automatically receive and interpret an image of a real scene in order to obtain information and/or control machines or processes."

An experienced inspector usually performs the non-destructive visual inspection. However, if the task is tedious and difficult, the inspector may tire prematurely, thus reducing the quality of the inspection. Repetitive and dangerous inspection demands machine vision so precise information can be extracted and interpreted consistently. With technological advances in computers, cameras, illumination and communication, a wide range of possible applications for the use of machine vision in nondestructive inspection is foreseen.

System Architecture of a Machine Vision System

The basic architecture of a PC-based machine vision system is given in Figure 1. The main components include a light source, camera(s), optics, the frame grabber and computer. However, machine vision does not necessarily mean a computer is used. Specialized image processing hardware is capable of achieving a higher processing speed and can replace the computer [1]. The modern approaches may use a camera with the capability to interface directly with a PC, a system designed on an image processing board or a vision engine that plugs into a PC [2].

A smart camera is a self-contained and stand-alone unit with communication interfaces. A typical smart camera may consist of an image sensor, image digitization circuit, memory, a digital signal processor, communication interface, I/O ports and a built-in illumination device. An embedded vision computer, which is a stand-alone box with frame storage and intelligence, is between the PC-based vision system and a smart camera [2]. It varies from smart cameras in that the camera is tethered to the unit rather than self-contained. Different system configurations have their own advantages for different applications. A PC-based machine vision system has the greatest flexibility and capability of handling a wider range of applications.

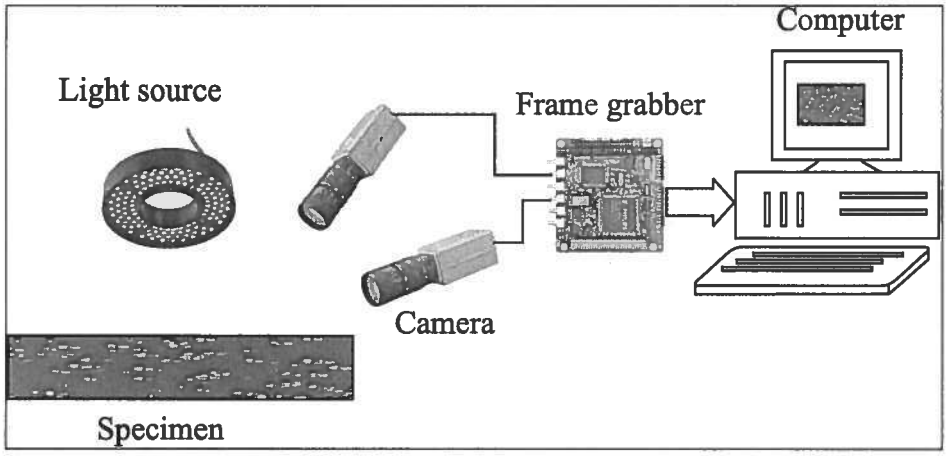


Figure 1. Typical architecture of a machine vision system

Optics and Lighting

Optics

A basic element for a machine vision system, the optics creates an image where object points and image points correspond. The optics also contribute to the object enhancement [3]. However, the optics may introduce distortions and aberrations. The optics can include lenses, mirrors, beam splitters, prisms, polarizers, color filters and gratings. The optics in a machine vision system [3]:

- produce a two-dimensional image of the scene at the sensor;
- eliminate some of the undesired information from the scene image with various filters; and
- transfer or modify the light before it arrives at the scene.

As illustrated in Figure 2, there are four basic parameters for specifying the optics of a vision system: the field of view (FOV), resolution, working distance and the depth of view [4]. Field of view is the angular extent of the observable scene. The resolution of a system is the minimum distinguishable feature size of the object under inspection. The depth of field of a lens is the ability to maintain a desired resolution as the object is positioned closer to or further from the lens.

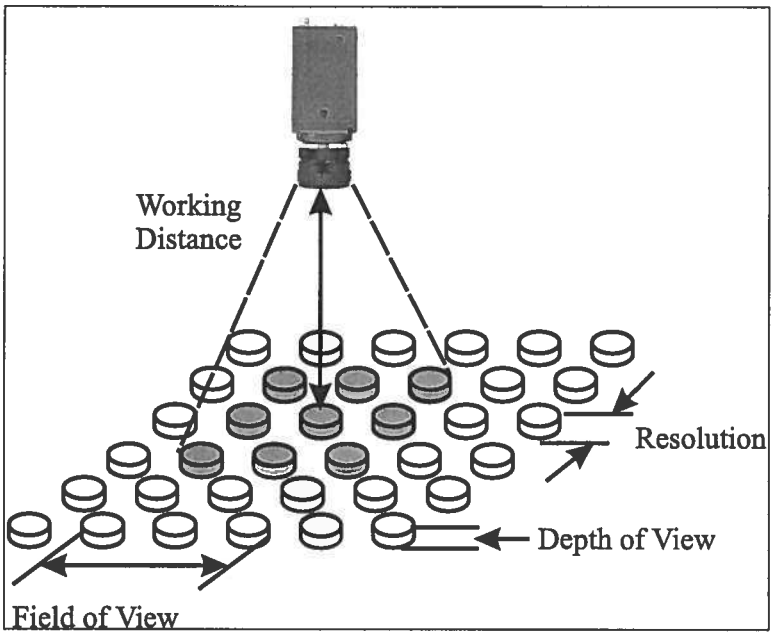


Figure 2. Four basic parameters for optics

The second element of a machine vision system is lighting. Application-specific lighting or illumination can yield a consistent appearance, which is essential to the success of the system. The lighting should maximize the contrast of the feature of specific interest while minimizing all other features.

Illumination

Illumination includes the following techniques:

- front lighting
- backlighting
- coaxial
- structured illumination
- strobed illumination
- polarization.

As illustrated in Fig. 3(a), a bright field mode for front lighting is defined as any light source that is in the line of sight of the camera upon direct reflection from the source. A matte surface will appear darker than specular surfaces since less light returns to the camera due to the scattering of the matte surface. In contrast, the sharp reflection returns more light. A dark field is defined as any light source that is outside the line of sight of the camera upon direct reflection. When using a dark field, light scattering from a matte surface will reach the camera creating a bright region. Similarly, a bright field for backlighting is defined as any light source that is in the line of sight of the camera upon direct transmission from the source, while a dark field is outside of the line of sight of the camera upon direct transmission as shown in Fig. 3(b).

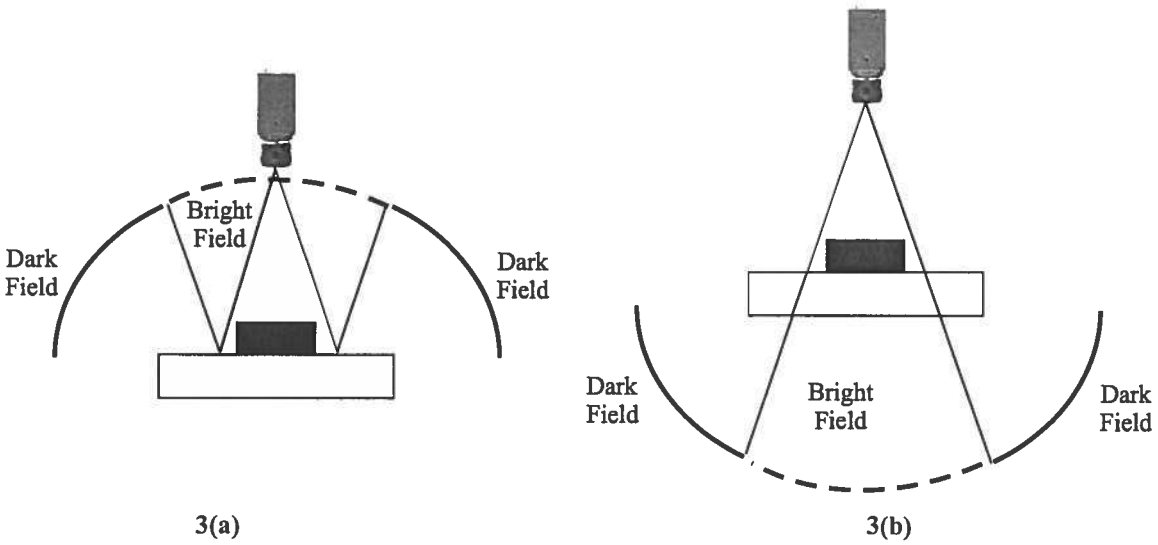


Figure 3. Bright and dark field modes for front light 3(a) and back light 3(b)

Front lighting is the most convenient illumination configuration for machine vision systems. A front lighting setup may be a single point source, a combination of lighting or a setup that encompasses the entire dome. Fig. 4 shows directional lighting, diffuse dome lighting and oblique lighting. Light from one direction can create high contrast transitions to highlight an object's features. With a diffuse dome light, the light may scatter in many different directions. This makes it useful for lighting curved and specular surfaces. Oblique lighting is obtained by restricting the light to lower incoming angles. With this technique, flat specular surfaces reflect light away from the camera while small raised or depressed areas reflect light back.

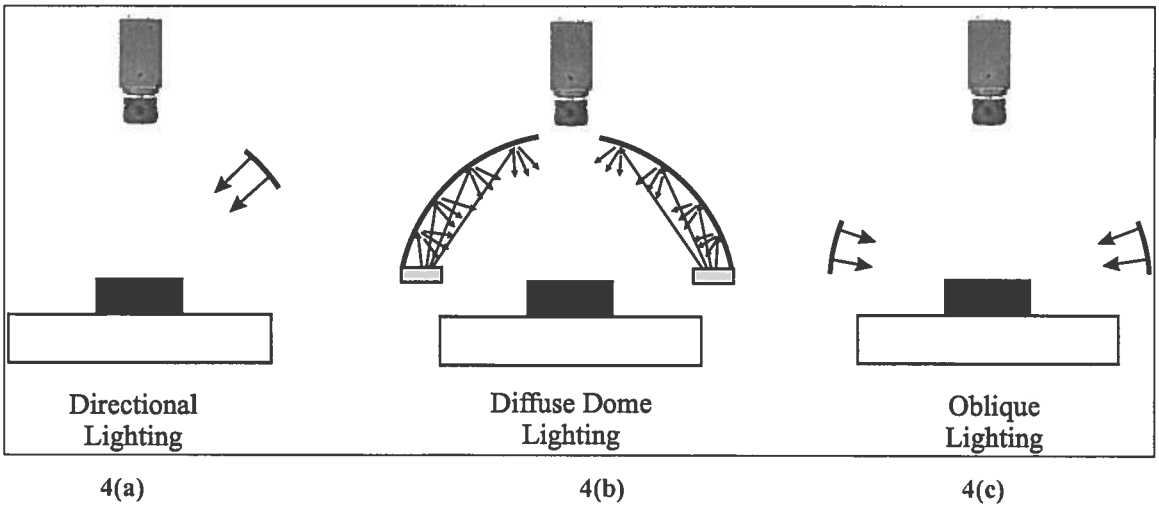


Figure 4. Some configurations of front lighting -- 4(a) directional lighting; 4(b) diffuse dome lighting; 4(c) oblique lighting

In back lighting in Figure 5(a), the light source and camera are placed on opposite sides of the object under inspection. This creates dark silhouettes against a bright background. A beam splitter is used to create a coaxial or on-axis illumination as shown in Figure 5(b). On-axis or coaxial illumination is particularly effective for enhancing differentially angled, textured or topographic features on a relatively flat object.

Structured illumination is the projection of a light pattern at a known angle onto an object so the dimensional information can be acquired [5]. The light pattern can be a plane, grid or other complex shape. The intersection of the pattern and an object can be translated into geometric information. Thus, the shape of an object can be determined.

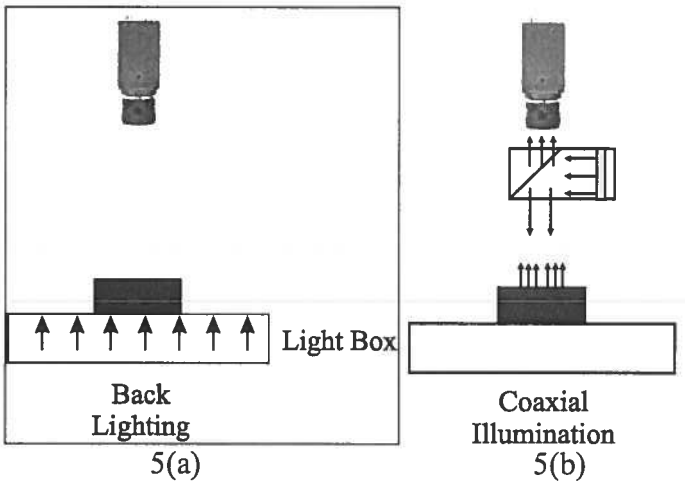


Figure 5. (a) back lighting and (b) coaxial illumination

Strobed illumination is also known as strobe light, in which a flash of light is used to illuminate the test object momentarily [6]. Alarm systems, theatrical lighting and high-visibility running lights are common applications of strobe lighting.

Unpolarized light is an electromagnetic wave vibrating in various directions; polarization limits such vibration to a single plane. The methods include transmission, reflection, refraction and scattering. Polarization techniques can improve the optical front end of a machine vision system.

Light shapes

The shape of the light source includes point, line and area [7]. A point source can be approximated by an extremely small sphere and has two types of models: a nearby point source and a point source at infinity [7]. The line source has the geometry of a line. A single fluorescent light bulb is an example. An area source is an area that radiates light.

Area sources are often modeled as surface patches and the emitted radiance is independent of position and direction.

Lighting arrangement

Six lighting sources are now commonly used in machine vision [8]:

- fluorescent
- quartz halogen
- LED (light emitting diode)
- metal halide (mercury)
- xenon
- sodium.

Table 1 gives a brief description of each type of lighting source.

Table 1. Brief description of lighting sources

Lighting	Description
Fluorescent	Fluorescent lighting uses electricity to excite mercury vapor to produce shortwave ultraviolet light, which causes a phosphor to fluoresce producing visible light.
Quartz halogen	A quartz halogen lamp is an incandescent light bulb where the envelope is made of quartz and the filament is surrounded by halogen gas.
LED	LED is a semiconductor diode that emits light when an electrical current is applied.
Metal halide	Metal halide lamps produce light by passing an electric arc through a high-pressure mixture of argon, mercury and a variety of metal halides.
Xenon	Xenon is used in flash and arc lamps. A xenon arc lamp uses ionized xenon gas to produce a bright white light while a xenon flash lamp is an electric glow discharge lamp that produces extremely intense, incoherent, full-spectrum white light for very short durations.
Sodium	A sodium vapor lamp is a gas discharge lamp using sodium in an excited state to produce light. There are two types available: low-pressure and high-pressure lamps.

The choice of a light source depends on the requirements for the brightness and spectrum content. Each technique has its advantages and disadvantages. An inspection must find the suitable lighting solution for the application.

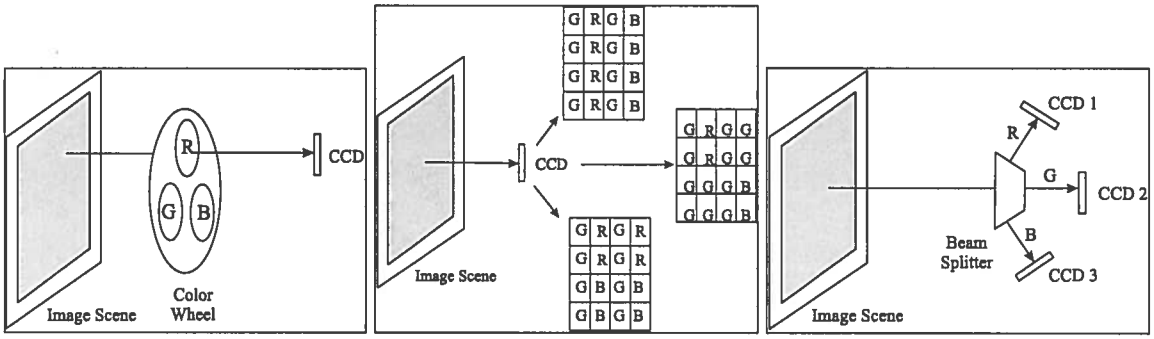
Cameras

A camera is used to capture images of still or moving objects. Traditional cameras capture light onto photographic film or photographic plate. Nowadays, digital cameras use the solid imaging device, for example, a CCD (charge coupled device) or a CMOS (complementary metal oxide semiconductor), to capture images, which can be stored in a computer's memory for later use.

Charge coupled device

A CCD is an electronic detector consisting of many square photosensitive pixels [9]. This is known as the capacitor array (photoactive region) or transmission region. An image is projected on the capacitor array through a lens. Each capacitor accumulates an electric charge proportional to the light intensity at that location. Using a control circuit, the charge can be transferred to a charge amplifier, which converts the charge into a voltage reading.

Silicon-based CCDs are monochrome. Three techniques are commonly used to extract color information for a given scene as shown in Fig. 6 [10]. The color sequence is switched in optical filters with the desired RGB characteristics. A color image can also be created using integral color filter arrays (CFA) over the CCD [10]. A CFA registers the intensity of a single color at each pixel [11]. By interpolation with the color intensities of adjacent pixels, the intensity of a color at each pixel can be estimated.



6(a) 6(b) 6(c)

Figure 6. Three techniques to extract color information – 6(a) color sequential capture; 6(b) integral color filter array; 6(c) three-chip color capture

Three-CCD cameras have three separate CCDs and provide higher image quality compared to one-CCD cameras. In three-CCD cameras, each CCD takes a separate measurement of red, green and blue light for each pixel. Light is split by a trichroic prism assembly, and a corresponding color filter is placed in each of the three imaging planes. A color image is obtained by synchronizing the outputs of the three CCDs.

Complementary metal oxide semiconductors

In a CMOS camera, each pixel has its own charge-to-voltage conversion. Most functions, such as amplifiers, noise correction, and digitization circuits, are integrated into a chip, which can output digital bits [12]. Although less flexible, such a design makes CMOS cameras more reliable.

Change injection devices

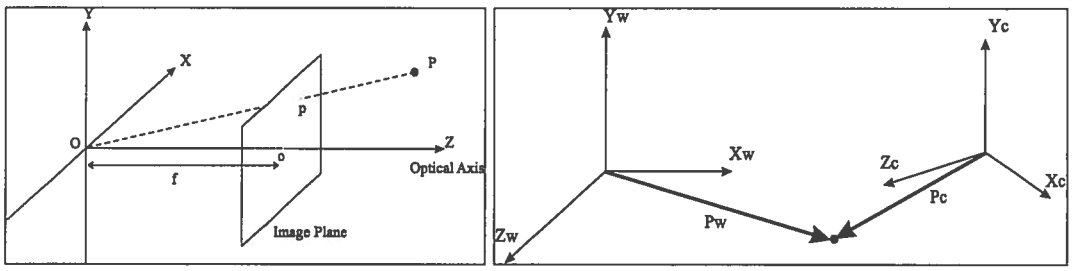
Charge injection device (CID) cameras have a similar detection mechanism as CCDs. The difference lies in their readout system. The collected charge in a CID camera does not transfer from site to site in the CID array. Instead, a displacement current, which is proportional to the stored charge, is read when charge packets are shifted between capacitors within individually selected pixels [13]. The displacement current is then amplified, converted to a voltage and read as a digitized signal. To clear the array for a new frame integration, the electrodes in each pixel are momentarily switched to ground releasing.

Camera calibration

The most common camera model is the pinhole model shown in Fig. 7 [14]. It consists of an image plane and center of projection O. The focal length is f, which represents the distance between O and the image center o. OZ is the optical axis. The three-dimensional reference frame XYZ is called the camera frame. The purpose of camera calibration is to estimate the intrinsic and extrinsic parameters of the camera model.

- *Intrinsic parameters* link the pixel coordinates of an image point with the corresponding coordinates in the camera reference frame.
- *Extrinsic parameters* define the location and orientation of the camera reference frame with respect to a known world reference frame.

In detail, the intrinsic parameters include the focal length f, the transformation between camera frame coordinates and pixel coordinates, and the geometric distortion introduced by the optics. The extrinsic parameters include a translation vector T and a rotation matrix R, which relate the coordinates of a point P in the world (P_w) and in camera frame (P_c) with equation $P_c = R(P_w - T)$ [14]. Numerous approaches are available to achieve this [15] [16].



7(a) 7(b)

Figure 7. 7(a) pinhole camera model; 7(b) transformation between camera frame coordinates and pixel coordinates

Camera Interface

The camera interface hardware connects different image sensors and provides a standard output. A frame grabber can capture individual frames from an analog or digital video stream and store and/or compress these frames in real time. Analog frame grabbers accept and process analog video signals while digital ones deal with digital video streams. The physical interface standards include Camera Link™, USB (universal serial bus), GigE Vision™ and IEEE-1394 [17]. Table 2 gives the basic information about these standards.

Table 2. Summary of camera interfaces

Interface Standard	Description
Camera Link™	A serial communication protocol that extends the base technology of Channel Link®, for vision application [18].
USB 2.0	A serial bus standard to interface with a computer [19]. It is designed for simplicity and low cost.
GigE Vision™	Based on the gigabit ethernet standard with fast data transfer rates, it allows the use of low-cost standard cables over very long lengths [20].
IEEE 1394	A serial bus interface standard for high-speed communication and isochronous (real-time) data transfer [21] used for high-performance and time-sensitive applications.

The distinguishing difference between these interfaces is speed [17]. Other differences include cable length, whether power is supplied over the cable and the level of software support. Table 3 shows the major differences between these interfaces [22].

Table 3. Comparison of different camera interfaces

Category	Camera Link™	USB	GigE Vision™	IEEE 1394
Topology	Master-slave	Master-slave (on-the-go)	Networked, peer-to-peer	Peer-to-peer
Maximum bit rate	2380 Mbps	480 Mbps	1000 Mbps	400~800 Mbps
Isochronous mode	Yes	Yes	No	Yes
Maximum sustained bit rate	2380 Mbps	432 Mbps	930 Mbps	320~640 Mbps
Cable distance (Copper)	10 m	5 m	25 m	4.5~100 m
Bus power	None	Up to 0.5 A	None	Up to 1.5 A

Algorithms and Software Implementation

Image Processing

Convolution operator

In the image processing, the convolution operator is the most useful and basic operation. Figure 8 shows an example of this operation.

Let image $I(x,y)$ be an image to process. To apply the convolution operator, a small image $w(i,j)$, which includes weights multiplied by each pixel in $I(x,y)$, is prepared. $w(i,j)$ is also called the operator, mask or kernel. When the size of $w(i,j)$ is $M \times N$, the convolution operator in a coordinate (x,y) is described as:

$$I'(x,y) = \sum_{i=-M/2}^{M/2} \sum_{j=-N/2}^{N/2} w(i+M/2, j+N/2) \cdot I(x+j, y+j)$$

In this calculation, $w(i,j)$ and $I(x,y)$ overlap with the center of $w(i,j)$ and the coordinate (x,y) located at the same position. Corresponding pixels are multiplied and summed. This calculation is the discrete version of the orthogonal transformation. Hence, the convolution operator is related to the Fourier transformation in the frequency domain. The following sections describe image processing using the convolution operators.

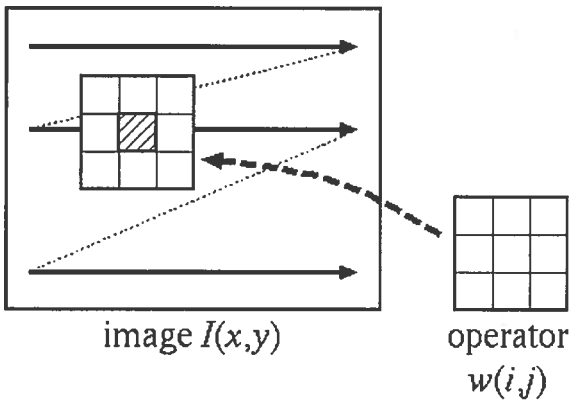


Figure 8. Convolution operator

Edge detection

The edge in the image is defined as the part in which the pixel value (intensity or color) changes drastically. Therefore, parts of the edge in the image are estimated by calculating the differentiation of pixel values. However, image data cannot be expressed by mathematical functions, so the difference is usually used instead of the differentiation. Moreover, the convolution operator is used to calculate the difference. Figure 9 shows a simple operator of difference. Fig. 9(a) is the difference along the x-axis. The image data are spread on a two-dimensional plane; hence, the operator along the y-axis (Fig. 9(b)) is also used.

0	0	0
0	1	-1
0	0	0

(a) X direction

0	0	0
0	1	0
0	-1	0

(b) Y direction

Figure 9. Differential operator

Here, $I_x(x,y)$ and $I_y(x,y)$ are the results of the difference (differentiation) along the x-axis and y-axis respectively. $\sqrt{I_x^2(x,y)+I_y^2(x,y)}$ is called the magnitude of the gradient, and $\tan^{-1}\left(I_y(x,y)/I_x(x,y)\right)$ is the direction of edge. A simpler version of the magnitude of the gradient, that is, $\left|I_x(x,y)\right|+\left|I_y(x,y)\right|$, is also used.

Moreover, there are other operators to calculate differences, such as the Roberts, Prewitt and Sobel operators illustrated in Figure 10. The Roberts operator can detect edges in the direction of a slant. The edges detected by the Prewitt and Sobel operators tend to be thick.

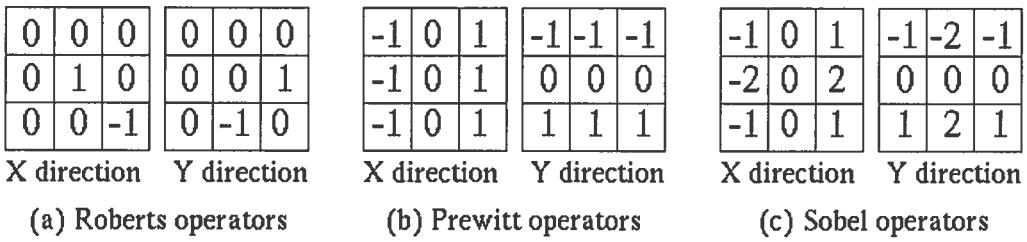


Figure 10. Operators for edge detection

Edge parts can be extracted by using the first and second order differentiations. A typical operator is the Laplacian. Three commonly used small kernels are given in Figure 11. This operator expresses the magnitude of the edge, combining x and y directions. According to definitions, there are some variations. The position of edge is the zero-crossing point, because large gradient values are found around the edge points.

The edge parts correspond to the high frequency parts of the image intensity. Therefore, the edge extraction process is the high pass filtering in the frequency domain. (This is proved mathematically.)

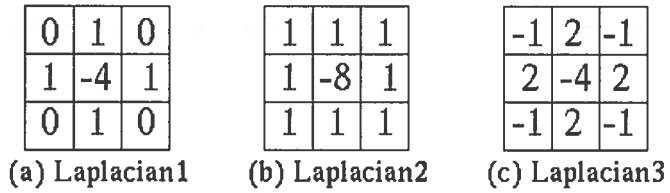


Figure 11. Laplacian operators

Noise reduction

In image data, various noises are included in pixels whose positions are not fixed (randomly). The intensities of these noises are different from surrounding pixels, so the noise is conspicuous. The *salt-and-pepper noise*, for example, is caused by the flicker of the illumination and the variation in the performance of imaging sensor elements.

Smoothing reduces such noises by obtaining the local averaged intensity value. Such a calculation can be performed by the convolution operation. Figure 12 shows a 3×3 smoothing operator. By applying the smoothing method, the image is blurred. The smoothing operation is a kind of low-pass filtering in the frequency domain.

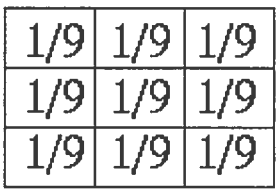


Figure 12. Smoothing operator

A *median filter effectively removes noise.* In this method, the pixels in a local region are sorted in terms of intensity value, and the median is picked as the new intensity for pixels located at (x,y) ((x,y) is the center position) (Figure 13). If the noise is included in the local region, it will be arranged in the first or the last after sorting. Unlike the convolution operator, the median filter is a non-linear operation. The smoothing method blurs the edges of an image, but the median filter does not.

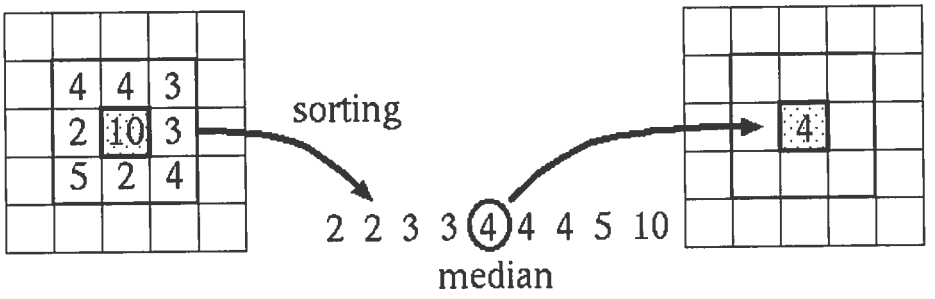


Figure 13. Process of median filtering

Image Segmentation

Binarization

Binarization is the identification of pixels in the image, which can be either the object or the background. From the assumption that the intensities of the pixels in the object are brighter (or darker) than the background, the binarization can be expressed as follows.

- For an object with a higher intensity than the background:
  $I(x,y) \geq t \rightarrow B(x,y) = 1$
  $I(x,y) < t \rightarrow B(x,y) = 0$

- For an object that is darker than the background:
  $I(x,y) \leq t \rightarrow B(x,y) = 1$
  $I(x,y) > t \rightarrow B(x,y) = 0$

Where:

$I(x,y)$ is the original monochrome image; and
$B(x,y)$ is the binary image.

$B(x,y) = 1$ denotes the object and 0 is the background. The boundary value t, which divides the object and the background, is called the "threshold".

The importance of the binarization is how to estimate the threshold value. Two popular methods for estimating the threshold--P-tile and the discriminant analysis method--are described below.

P-tile method

The P-tile method can estimate the threshold when the number of pixels (rate of areas) of the object and the background of the image are known. When this method is applied to a monochrome image, the histogram (the frequency distribution) of the intensity in the image can be obtained. The area of the object occupies P percent in the histogram from the bright (dark) side. Therefore, the threshold is decided from the histogram (Figure 14).

In the P-tile method, the rate of area P should be known to estimate the threshold. For example, in a document image there are only letters and the rate of black pixels (letters) is considered about 5 percent in the whole image, but if figures, tables and photographs are included in the image, it is difficult to define the rate P.

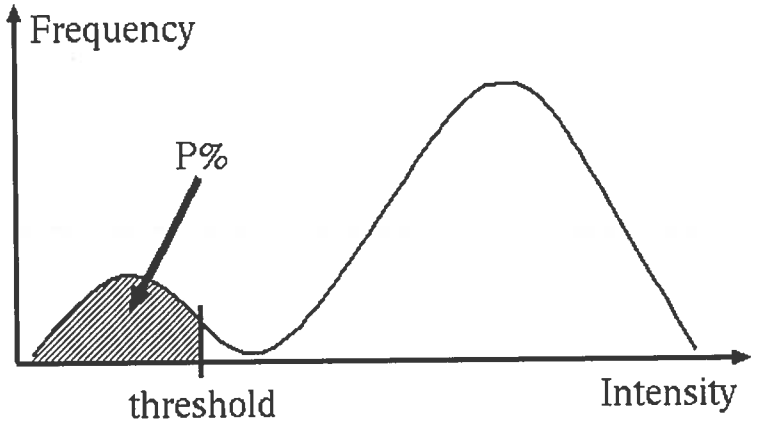


Figure 14. P-tile method

Discriminant analysis method

The discriminant analysis method can estimate a threshold from only image data. This method also uses the histogram of pixel intensity.

With this method, the histogram is divided into two classes: dark and bright pixels. Using the between-class variance (σ_{bc}^2) and the within-class variance (σ_{wc}^2), the separation metrics ($\sigma_{bc}^2/\sigma_{wc}^2$) are defined. The threshold is determined when the separation metric reaches its maximum value.

Assuming that the histogram is divided into two classes by a threshold t (Figure 15), let ω_1 be the number of pixels, m_1 be the average and σ_1^2 be the variance in the dark pixel class. In the bright pixel class, ω_2, m_2 and σ_2^2 are also defined similarly while m and σ^2 show the average and variance in the whole image.

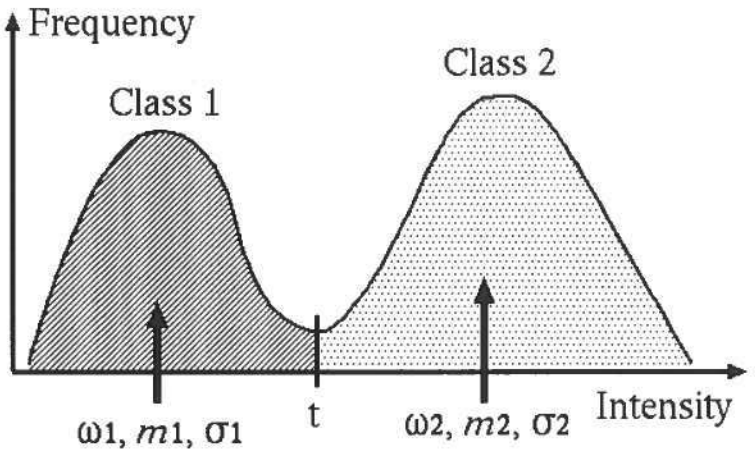


Figure 15. Discriminant analysis method

The within-class variance (σ_{wc}^2) is expressed as:

$$\sigma_{wc}^2 = \frac{\omega_1\sigma_1^2 + \omega_2\sigma_2^2}{\omega_1 + \omega_2} \tag{1}$$

The between class variance (σ_{bc}^2) is:

$$\sigma_{bc}^2 = \frac{\omega_1(m_1 - m)^2 + \omega_2(m_2 - m)^2}{\omega_1 + \omega_2} = \frac{2\omega_1\omega_2(m_1 - m_2)^2}{(\omega_1 + \omega_2)^2} \tag{2}$$

On the other hand, there is a following relation between σ^2, σ_{wc}^2 and σ_{bc}^2:

$$\sigma^2 = \sigma_{wc}^2 + \sigma_{bc}^2 \tag{3}$$

From the above equations, the separation metrics is formulated as:

$$\frac{\sigma_{bc}^2}{\sigma_{wc}^2} = \frac{\sigma_{bc}^2}{\sigma^2 - \sigma_{bc}^2} \tag{4}$$

Since σ^2 is fixed, when σ_{bc}^2 reaches its maximum value, the separation metric is maximized. Therefore, in the discriminant analysis method, σ_{bc}^2 is calculated by changing the value of t and searching for the adequate threshold in case of maximum σ_{bc}^2.

By using the discriminant analysis method, the threshold is uniquely estimated for any monochrome images. Although only the case of binarization (two classes, black and white) is demonstrated, this method can also be applied to estimate multiple thresholds.

Mathematical Morphology

A logical process applied to a binary image, *the mathematical morphology* can perform noise reduction, lack restoration and so on. Some practical methods of this process are described below.

Dilation

The *dilation* expands the object region in a binary image X using the structuring element Y. The *structuring element* is a small binary image with various shapes. A pixel in the structuring element is defined as an origin of the element.

The operation of dilation is defined as:

$$X \oplus Y = \{z \mid z = x + y \text{ for } x \in X, y \in Y\} \tag{5}$$

Note that this formulation is only valid when $\forall x \in X$ and the origin of the structuring element y_{org} should be located in X. Examples of the dilation are shown in Figure 16. When the structuring elements are different, the results are of a different shape.

Erosion

Erosion is the opposite operation to dilation. It shrinks the binary image according to the structuring element. The operation of erosion is defined as:

$$X \otimes Y = \{z \mid z + y \in X \text{ for } \forall y \in Y\} \tag{6}$$

Note that this formulation is only valid when z and y_{org} are located at the same position. Examples of erosion are shown in Figure 17. As in the case of dilation, when the structuring elements are different, the resulted shapes are different.

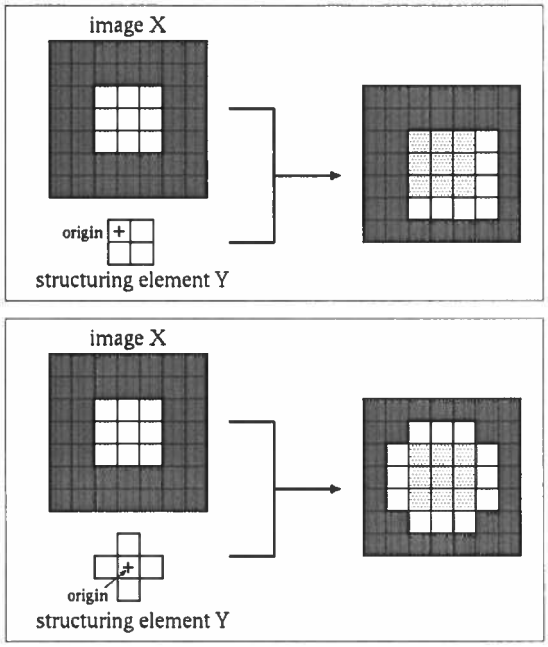


Figure 16. Examples of dilation

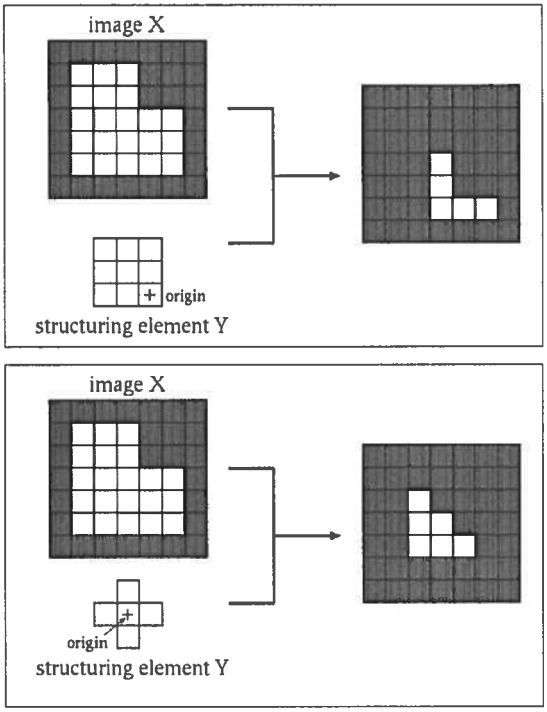


Figure 17. Examples of erosion

Opening and closing

The operations that combine dilation and erosion are called the opening and closing operations. They are defined as follows:

- Opening: $X \circ Y = (X \otimes Y) \oplus Y$
- Closing: $X \bullet Y = (X \oplus Y) \otimes Y$

The opening operation eliminates the isolated noise and small regions, while the closing operation fills the holes.

When the dilation is applied iteratively, the object region becomes large. When the erosion is iterated, the object region becomes small. The opening and closing operations can conserve the size of main parts of the image, even if the iterative operations are applied.

Geometric Transformation

The geometric transformation (for example, the object size is changed, the slant corrected and so on) is a major technique in image processing. Many geometric transformations can be expressed as the affine transformation using a 3×3 matrix. Let (x,y) be the coordinate before transformation and (x',y') be the transformed coordinates. The affine transformation is formulated as:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{7}$$

The following equations describe the geometric transformations (see Figure 18).

- **scaling**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & 0 \\ 0 & \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{8}$$

  α, β: scaling rates for x and y directions.

- **rotation**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{9}$$

θ: angle from x-axis around the origin.

- **translation**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{10}$$

t_x, t_y: amounts of translation

- **skewing**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & p & 0 \\ q & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{11}$$

p,q: slants from y or x-axis

- **reflection**

x-axis:
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{12}$$

y-axis:
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{13}$$

$y=x$:
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{14}$$

By multiplying several matrices, a composite transformation can be obtained. Since the transformation is a multiplication of multiple matrices, the change of the sequence of these matrices will give a different (transformation) result.

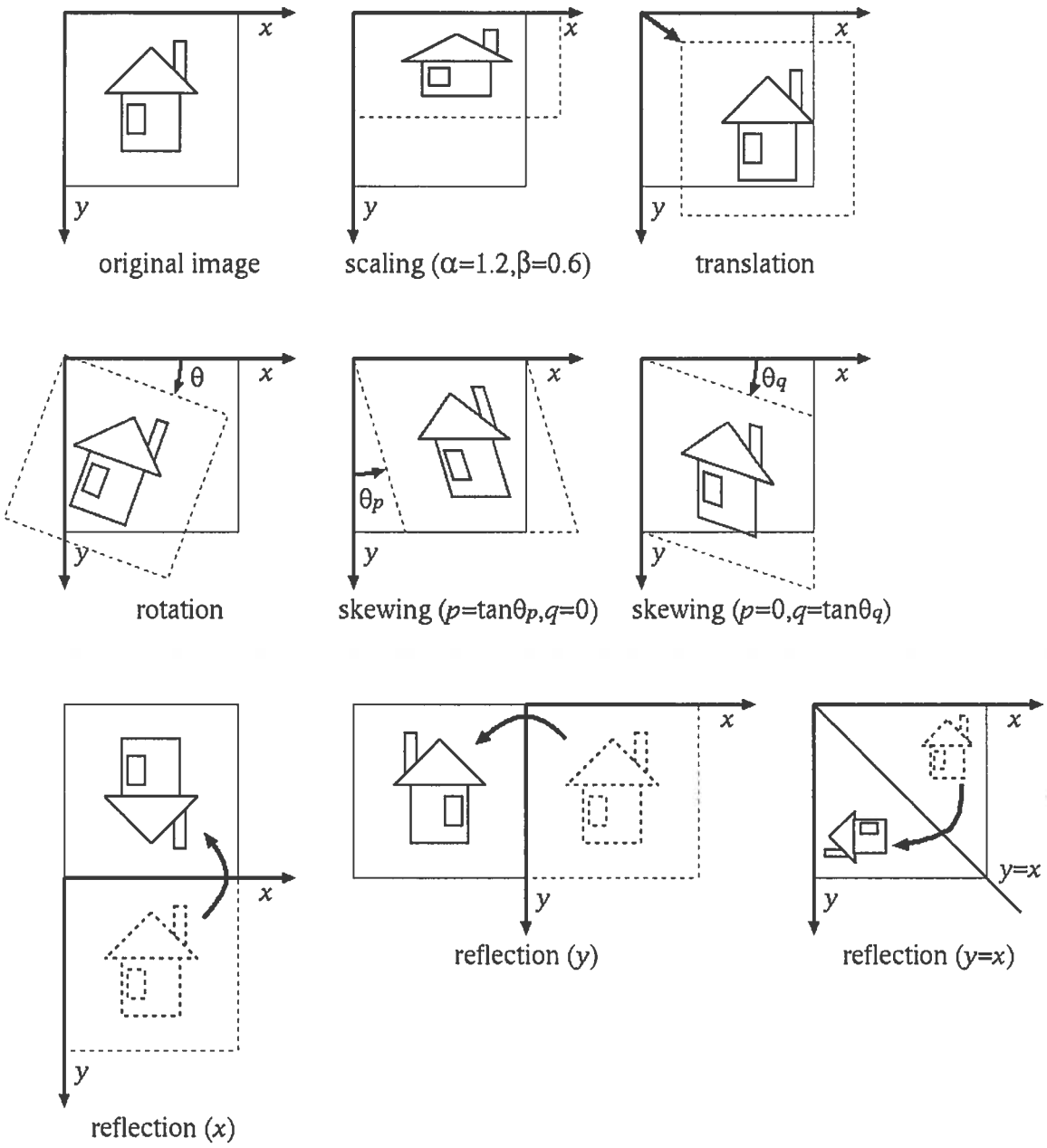


Figure 18. Geometric transformations

Pattern Recognition

Template matching

The *template matching method* is usually used to search for an object in an image. A small image of an object called a *template* is prepared first. The next step is to match the template with the whole image. In this process, the similarity is estimated at each location

of the image. As a result, the object in the template is considered to be located at the position where the similarity reaches the maximum value for the whole image (Figure 19).

Let the size of template image $T(x,y)$ be $t_w \times t_h$. To calculate the similarity at (x,y) in the input image $I(x,y)$, the following equations are applied:

- SAD (Sum of Absolute Difference)

$$R_{SAD}(x,y) = \sum_{i=0}^{t_w-1} \sum_{j=0}^{t_h-1} \left| I(x+i,y+j) - T(i,j) \right| \tag{15}$$

- SSD (Sum of Squared Difference)

$$R_{SSD}(x,y) = \sum_{i=0}^{t_w-1} \sum_{j=0}^{t_h-1} \left(I(x+i,y+j) - T(i,j) \right)^2 \tag{16}$$

- NCC (Normalized Cross-Correlation)

$$R_{NCC}(x,y) = \frac{\sum_{i=0}^{t_w-1} \sum_{j=0}^{t_h-1} I(x+i,y+j) \cdot T(i,j)}{\sqrt{\sum_{i=0}^{t_w-1} \sum_{j=0}^{t_h-1} I(x+i,y+j)^2} \cdot \sqrt{\sum_{i=0}^{t_w-1} \sum_{j=0}^{t_h-1} T(i,j)^2}} \tag{17}$$

In the above equations, the smaller the value of SAD and SSD, the higher the similarity. For NCC, a higher value indicates a higher similarity. The similarity estimation is carried out over the whole input image, and the position at which the highest similarity is obtained is the location of the object.

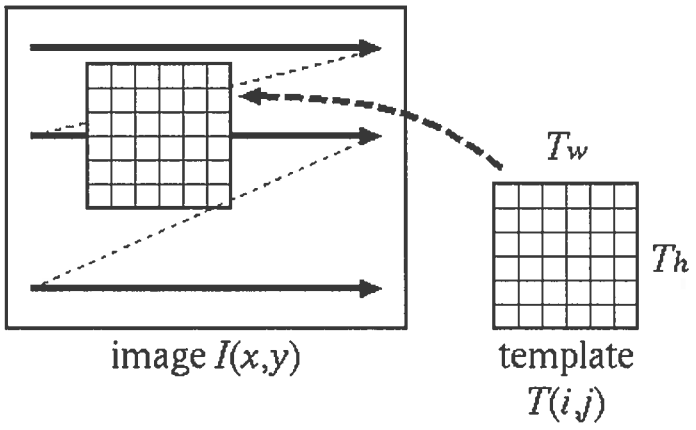


Figure 19. Template matching method

The computational costs of SAD and SSD are small and can be estimated rapidly. However, if the brightness (gain) of the input image is different from that of the template image (in other words, the input and template images are of different illumination conditions), the similarity will be low. Therefore, an accurate match will not be achieved. The NCC estimates the correlation between the input and template images, which is less likely to be affected by illumination changes, but comes with a higher computational cost.

So far as the template matching method is concerned, the size and pose of the object in the template image and those of the corresponding patterns in the input image need to be the same. If not, the similarity becomes low and an accurate match will not be obtained. In this case, it is necessary to apply the geometric transformations, described in the previous section to the template image. However, to do this requires estimating the transformation parameters, so it is not efficient. When the size and pose of the object in the template image vary from those in the input image, it is still possible to do the matching with color information or other high dimensional features.

Hough transform

The template matching method can be applied to any pattern of an object. If the object can be expressed by the mathematical models (line, circle, etc.), it can be searched more efficiently in the input image with a Hough transform.

- ***a-b* Hough transform**

The line detection with Hough transform is first described. A straight line in *x-y* space is modeled as following using parameters *a* and *b*:

$$y = ax + b \tag{18}$$

This formulation can be transformed as:

$$b = (-x)a + y \tag{19}$$

This formulation shows another straight line in a-b space. Here, the *a-b* space is called the *parameter space.*

As shown in Fig. 20 (left), when a line crosses two points, (x_1, y_1) and (x_2, y_2) in *x-y* space, parameters $\hat{a}$ and $\hat{b}$ of this line can be expressed as:

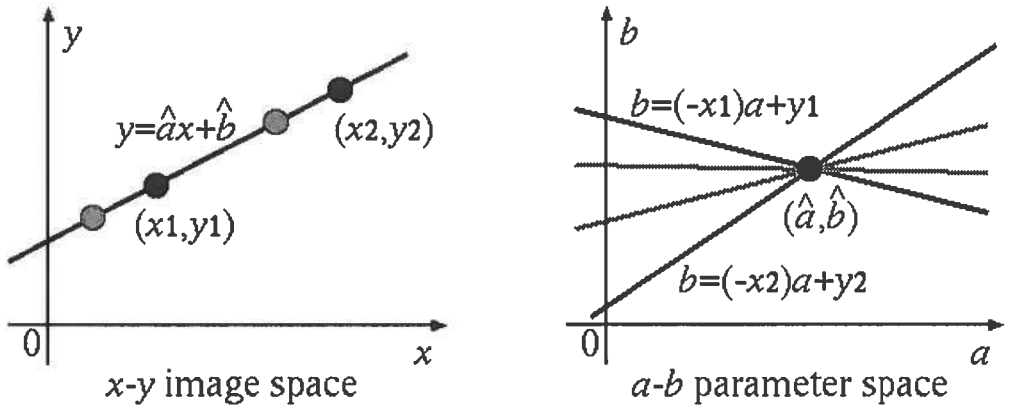
$$(\hat{a}, \hat{b}) = \left(\frac{y_2 - y_1}{x_2 - x_1}, \frac{x_2 y_1 - y_2 x_1}{x_2 - x_1} \right) \tag{20}$$

On the other hand, points, (x_1, y_1) and (x_2, y_2) in *x-y* space, correspond to the following lines in the parameter space:

$$b = (-x_1)a + y_1 \tag{21}$$

$$b = (-x_2)a + y_2 \tag{22}$$

And, the cross point of these lines is equal to $(\hat{a}, \hat{b})$ (Fig. 20(b)).



20(a) 20(b)

Figure 20. a-b Hough transform

Therefore, points on a straight line in an image (*x-y* space) correspond to lines that cross a point in the parameter space. By estimating the crossing point of straight lines in the parameter space, we can obtain parameters (*a* and *b*) of a straight line in an image (*x-y* space).

However, there are many cross points of the straight lines in the parameter space, so it is difficult to estimate adequate points mathematically. To overcome this problem, the cross points are obtained by a voting process in Hough transform. In this method, the parameter space is divided into small rectangular regions along each axis. Each small rectangular region is called a "cell". The cell works as a counter. If a line crosses a cell, the counter increases by one. Finally, the coordinates of the cell, which have maximum value of voting, are the estimated parameters of the line in the image. The voting process in the parameter space can be considered as a parameter estimation process for a line, which crosses (or is supported by) many points.

The least square method can also be used to estimate line parameters. Basically, this method estimates only one line. Hough transform can estimate multiple lines simultaneously by picking up cells whose voting number is beyond a threshold.

- **ρ-θ Hough transform**

Generally, the range of line parameters is from $-\infty$ to $+\infty$, which is the same as the range of the parameter space. The range of the image (in *x-y* space) is limited but the range of parameter *a* (slant of line) is from $-\infty$ to $+\infty$. It is difficult to prepare cells in such a range for computation. Therefore, the model of a straight line can be rewritten as:

$$\rho = x\cos\theta + y\sin\theta \tag{23}$$

where ρ is a signed distance from the origin to the line and θ is an angle representing the altitude of the line. When the range of the image (*x-y* space) is limited to

$0 \le x \le w, 0 \le y \le h$, the ranges of ρ and θ are $-\sqrt{w^2+h^2} \le \rho \le \sqrt{w^2+h^2}$ and $0 \le \theta \le \pi$ respectively.

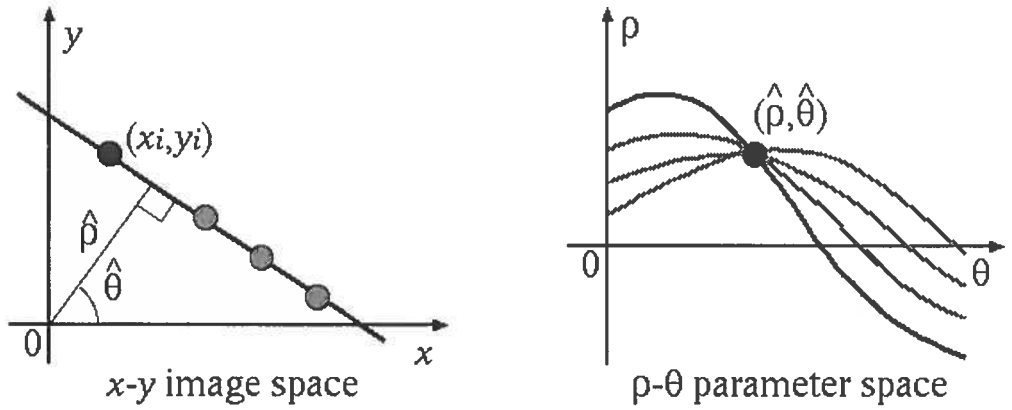
For a coordinate (x_1, y_1) on the image (x-y space), equation (23) becomes:

$$\rho = A\sin(\theta + \alpha) \qquad (24)$$
$$(A = \sqrt{x_1^2 + y_1^2}, \alpha = \tan^{-1}(x_1 / y_1))$$

It means that a point in x-y space corresponds to a sine wave in ρ-θ parameter space. By using the voting process along to this trajectory, the cross points can be estimated (Fig. 21).

The ρ-θ Hough transform has a limited parameter space, but the computational cost is still high, because of the calculation of sine waves. To avoid such a problem, another method is proposed. The γ-ω Hough transform [26] uses piece-wise line segments in the parameter space to perform the voting process rapidly. Moreover, because the coordinates of the image plane are discrete values, by considering the cell re-division in the parameter space, line parameters can be estimated more accurately [27].

If the parameter space is expanded to three dimensions, circles can be detected in an image. Moreover, if high dimensional parameter space is considered, it is possible to detect various patterns. However, the higher the dimension of parameter space, the more the computational costs. In this case, huge numbers of cells are needed, and it is difficult to perform any pattern detection. To overcome this problem, a generalized Hough transform, which detects the position and pose of a pattern by the voting process, is an option.



21(a) 21(b)

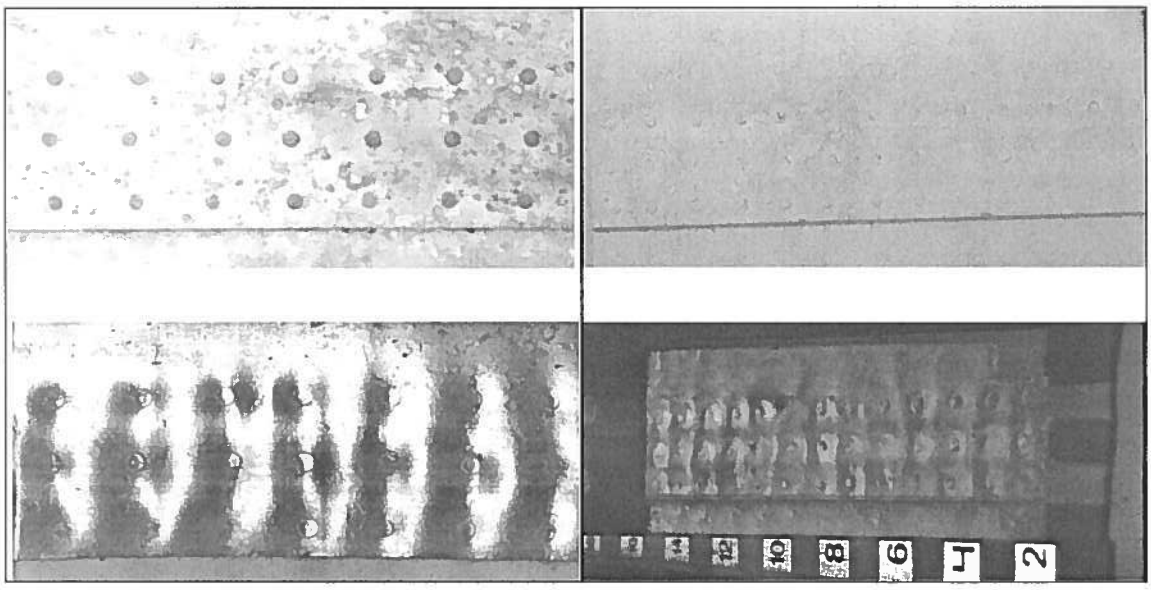
Figure 21. ρ-θ Hough transform

Machine Vision for Nondestructive Inspection

Nondestructive inspection (NDI) is an important applications of machine vision, which captures and characterizes defects of the object under inspection, that is, the inspection of surface quality and structural quality [28]. The use of light beyond the visible spectrum, such as infrared (IR), ultraviolet (UV) and X-ray for nondestructive inspection is described in other chapters of this handbook; this chapter focuses on the use of visible light in a machine vision system.

Three types of results may be obtained from a machine vision system. The first type is an enhanced image, in which the discontinuities are highlighted or intuitively presented so the inspector can easily make a subjective assessment. One example is the edge of light (EOL) technique, which uses the edge of light to highlight the surface slope or deformation [29].

Fig. 22 shows the result of EOL inspection of an aircraft lap joint. The top picture in 22(a) is the lap joint and the bottom one is the corresponding EOL scan, where bright and dark regions present the surface deformation. Such deformation implies the potential hidden corrosion between the two layers. A similar technique known as D-Sight has also been applied to the same application [30]. In 22(b), the top image is the specimen while the bottom image is the D-Sight image. These two techniques implement enhanced visual inspection through the design of a special machine vision system.



22(a) **22(b)**

Figure 22. Aircraft lap joints inspection with edge of light and D-Sight techniques – 22(a) top: aircraft lap joint; bottom: edge of light inspection result; 22(b) top: aircraft lap joint; bottom: D-Sight inspection result

Image processing techniques can also achieve an enhanced image to facilitate the inspection. In [31], a three-dimensional stereoscopic visual system was built to inspect aircraft skin. Algorithms to enhance monoscopic and stereoscopic images were developed. A high frequency emphasis algorithm consists of two steps as illustrated in Figure 23. The live image passed through a low-pass and high-pass filter respectively. Then, a fraction of low-frequency content was added back to the high-pass filtered image. This algorithm emphasized the high frequency features while attenuating the background low-pass information. Therefore, the potential surface flaws or cracks were highlighted.

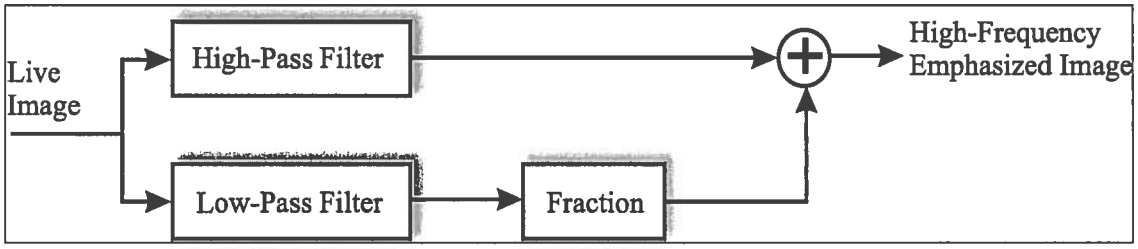


Figure 23. High frequency emphasis algorithm [26]

For the stereoscopic image enhancement, the high-frequency emphasis algorithm was applied to the left and right images of the stereoscopic image. An augmented stereoscopic high-frequency emphasis algorithm was implemented in three steps.

- Apply the high-frequency emphasis algorithm to the left and right images.
- Find the threshold of the filtered images dynamically to identify the features of interest.
- Overlay the identified features on the original left and right images with the desired depth offset.

The processed images were displayed stereoscopically on the display screen. The inspector or operator can use active eyeware to view these images.

The second type result is binary, that is crack or non-crack. This is useful for the inspection of a specific part, where a binary decision of reject or accept may follow. As described in Gunatilake et al. [32], the crack detection algorithm shown in Figure 24 was developed to identify the surface cracks on aircraft skin. Cracks frequently happen near rivets; therefore, the first step is to detect rivets by detecting the circular arcs in the image. Once the edge maps of the rivets are detected, the region of interest (ROI) can be determined with the centroid of the rivet.

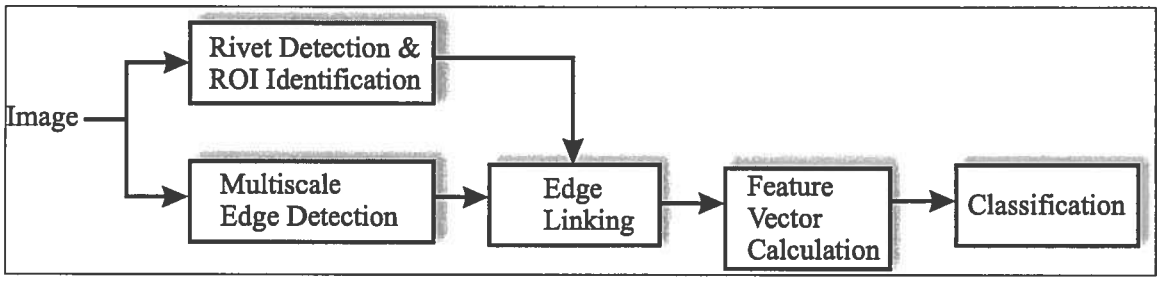


Figure 24. Surface crack detection algorithm

Once the ROI is identified, the multiscale edge detection is applied to the ROI to generate a list of edges at different scales. This method will help discriminate cracks from non-cracks according to the size of a typical crack in comparison to other objects, such as scratches and repair plates appearing on the surface. A coarse-to-fine edge linking process traced an edge from the coarse resolution (high scale) to a fine resolution (low scale). The propagation depth of all edges presented at scale one was found. Here, the propagation depth means the number of scales in which the edge appears. A feature vector for each edge in scale was generated so the edges of cracks could be discriminated from those of non-cracks. The feature vector includes [32]:

- average wavelet magnitude of active pixels, which belong to the edges;
- propagation depth number;
- average wavelet magnitudes of any linked edges in scale two and scale four;
- spins of sum (W_x) and sum (W_y), where W_x, W_y are the wavelet coefficients in the x and y direction of an active pixel at scale one; and
- the number of active pixels.

A neural network (NN) as shown in Figure 25 was trained to classify the inputs, that is, feature vectors of edges in the ROI, into cracks and non-cracks. The feature vectors used for the training may represent the cracks that need immediate repair. In this case, the classification result, that is, a crack, indicates a further investigation of the corresponding ROI for repairing. An accuracy rate of 71.5 percent and a false alarm rate 27 percent for the NN-based classification were reported.

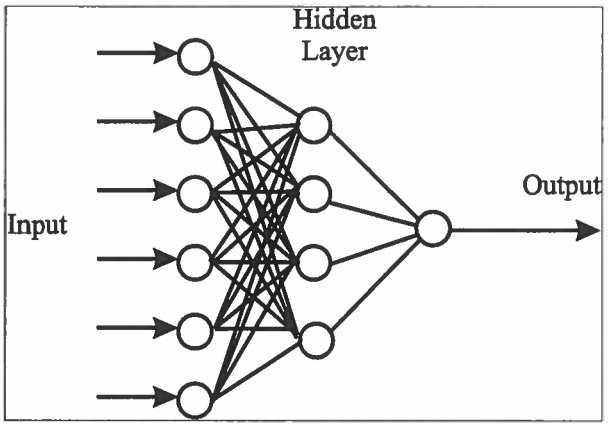
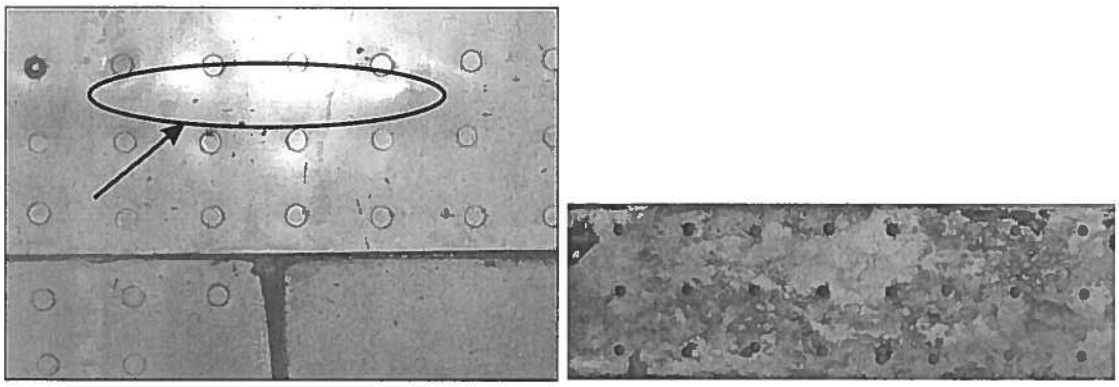


Figure 25. Neural networks used for crack classification



26(a) **26(b)**

Figure 26. Pillowing deformation on aircraft lap joints (a) and faying (hidden) surface (b)

The third type is more informative, which allows quantitative information about the discontinuity to be derived. In aircraft inspection, corrosion detection is crucial to the risk assessment of structural integrity. One type of corrosion occurs on the interior, hidden surface of aircraft lap joints if sealant and corrosion protection systems break down. Corroded product is of much higher volume than the original material and this will cause an expansion of the skin between rivets. This phenomenon is known as "pillowing". An example is shown in Fig. 26 where 26(a) shows pillowing on a lap joint while 26(b) shows the corroded area on the faying surface. Another type of corrosion happens to the surface, which can be detected by its suggestive texture captured by a machine vision system. A procedure for surface corrosion detection was present in Gunatilake et al. [32]. The image was first decomposed into sub-images with a discrete wavelet transform.

Figure 27 shows a three-level decomposition, which consists of ten sub-images. Let $w_j(k,l)$ be the wavelet coefficient at (k,l) in the sub-image W_j. The original image was divided into non-overlapping blocks of size 8×8 pixels. For each block $B(i)$, a ten-

dimensional feature vector was created. The element is $E_j(i)$ $(j=1,\cdots,10)$, which is the corresponding energy function in sub-images and can be expressed as:

$$E_j(i)=\sum_{(k,l)\in B(i)} w_j(k,l)^2$$

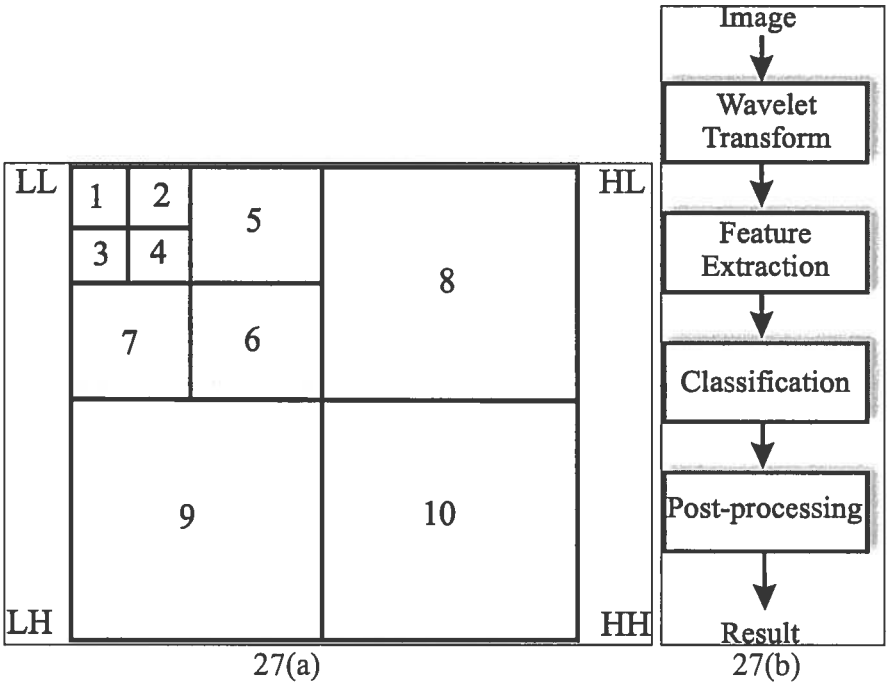


Figure 27. Wavelet decomposition of an image (a) and the procedure for classification (b)

Then, a nearest neighbourhood classifier was trained to classify the original image into corrosion and corrosion-free regions. A detection rate of 95 percent of the test set was reported [32]. Once the original image is classified, post-processing can be carried out to calculate the corrosion area. Therefore, the information about the size of the corroded area is available.

A more general procedure is shown in Figure 28. The image is first pre-processed for enhancement and noise removal so the features of targeted objects (discontinuities) are highlighted. The discrimination of different objects will be achieved in a feature space. Extraction of image features can be done in the spatial domain and frequency domain; numerous approaches can be used. Some of these techniques have been described in previous sections of this chapter. Sometimes, post-processing is needed to further characterize the classified results as described in the above example. The measurement results can also be compared with calibrated samples for quantitative analysis. Such comparison can also be done in the feature space.

References

[1] B. G. Batchelor, and P. F. Whelan, "Intelligent Vision Systems for Industry," 2002.

[2] N. Zuech. "Smart Cameras vs. PC-Based Machine Vision Systems," 2008; http://www.machinevisiononline.org/public/articles/archivedetails.cfm?id=1328, retrieved August 2009.

[3] N. Zuech, "Optics in Machine Vision Applications," 2005.

[4] G. Fales, "Ten Lens Specifications You Must Know for Machine-Vision Optics," *Test & Measurement World*, no. 11, 2003.

[5] StockerYale. "What is Structured Light?," http://www.stockeryale.com/i/lasers/structured_light.htm, retrieved August 2009.

[6] ASNT, "Part 4: Machine Vision Technology," *Nondestructive Testing Handbook*, M. W. Allgaier and S. Ness, eds., pp. 93-107, Columbus, Ohio: American Society of Nondestructive Testing, 1993.

[7] D. A. Forsyth, and J. Ponce, *Computer Vision: A Modern Approach*: Pearson Education, 2004.

[8] National_Instruments. "Tutorial: A Practical Guide to Machine Vision Lighting," 2008; http://zone.ni.com/devzone/cda/tut/p/id/6901, retrieved August 2009.

[9] O. Hainaut. "Basic Image Processing," 2006; http://www.sc.eso.org/~ohainaut/ccd/, retrieved August 2009.

[10] KODAK, "Kodak Image Sensors," 2008, pp. 1-24.

[11] C. Peterson, "How It Works: The Charged-Coupled Device, or CCD," *Journal of Young Investigators,* vol. 3, no. 1, 2001.

[12] D. Litwiller, "CCD vs. CMOS: Facts and Fiction," *Photonics Spectra*, pp. 1-4, January, 2001.

[13] "Charge Injection Device Research at RIT," 2009; http://www.cis.rit.edu/research/CID/, retrieved August 2009.

[14] E. Trucco, and A. Verri, *Introductory Techniques for 3-D Computer Vision*: Prentice Hall, 1998.

[15] J.-Y. Bouguet. "A Few Links Related to Camera Calibration," http://www.vision.caltech.edu/bouguetj/calib_doc/htmls/links.html, retrieved August 2009.

[16] J. Wang, F. Shi, J. Zhang *et al.*, "A New Calibration Model of Camera Lens Distortion," *Pattern Recognition* vol. 4, pp. 607-615, 2008, 2008.

[17] A. Wilson, "Camera Connections," *Vision Systems Design*, 2008.

[18] AIA, "Camera Link: Specifications of the Camera Link Interface Standard for Digitial Cameras and Frame Grabbers," 2004].

[19] "Universal Serial Bus," 2008; http://www.usb.org/home, retrieved August 2009.

[20] AIA. "GigE Vision Standard," http://www.machinevisiononline.org/public/articles/index.cfm?cat=167 retrieved August 2009.

[21] "1394 Technology," http://www.1394ta.org/Technology/. retrieved August 2009.

[22] J. Sgro, "USB Advantages Offset Other Interfaces," *Vision System Design*, 2008.

[23] R.Jain, R.Kasturi, and B.G.Schunck, *Machine Vision*: McGRAW-HILL, 1995.

[24] R.C,Gonzalez and R.E.Woods, *Digital Image Processing*: Prentice Hall, 2002.

[25] N.Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Trans. Sys., Man, and Cybernetics*, SMC-9, No.1, pp.62-66, 1979.

[26] T.Wada, T.Fujii, and T.Matsuyama, "$\gamma - \omega$ Hough Transform - Linearizing Voting Curves in an Unbiased $\rho - \theta$ Parameter Space-," *IEICE D-II*, Vol.J75-D-II, No.1, pp.21-30, 1992 (in Japanese).

[27] T.Wada, M.Seki, and T.Matsuyama, "High Precision $\gamma - \omega$ Hough Transform Algorithm to Detect Arbitrary Digital Lines," *IEICE D-II*, Vol.J77-D-II, No.3, pp.529-530, 1994 (in Japanese).

[28] E. N. Malamas, E. G. M. Petrakis, M. Zervakis *et al.*, "A Survey on Industrial Vision Systems, Applications and Tools," *Image and Vision Computing,* vol. 21, pp. 171-188, 2003.

[29] J. P. Komorowski, and D. S. Forsyth, "The Role of Enhanced Visual Inspection in the New Strategy for Corrosion Management," *Aircraft Engineering and Aerospace Technology,* vol. 72, no. 1, pp. 5-13, 2000.

[30] J. P. Komorowski, N. C. Bellinger, R. W. Gould *et al.*, "Quantification of Corrosion in Aircraft Structures with Double Pass Petroreflection," *Canadian Aeronautics and Space Journal,* vol. 42, no. 2, pp. 76-82, 1996.

[31] M. Siegel, and P. Gunatilake, "Remote Inspection Technologies for Aircraft Skin Inspection," in IEEE Workshop on Emergent Technologies and Virtual Systems for Instrumentation and Measurement, Niagara Falls, Ontario, Canada, 1997, pp. 1-10.

[32] P. Gunatilake, M. W. Siegel, A. G. Jordan *et al.*, "Image Understanding Algorithms for Remote Visual Inspection of Aircraft Surface," in SPIE Conference on Machine Vision Applications in Industrial Inspection, 1997, pp. 2-13.