International Journal of Production Research
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tprs20

Detecting and locating burrs of industrial parts

D.-M. TSAI‡ [a] & W.-J. Lu [a]
[a] Department of Industrial Engineering , Yuan-Ze Institute of Technology , 135 Yuan-Tung Road, Nei-Li, Taiwan, ROC
Published online: 15 Mar 2007.

To cite this article: D.-M. TSAI‡ & W.-J. Lu (1996) Detecting and locating burrs of industrial parts, International Journal of Production Research, 34:11, 3187-3205, DOI: 10.1080/00207549608905084

To link to this article: http://dx.doi.org/10.1080/00207549608905084

INT. J. PROD. RES., 1996, VOL. 34, NO. 11, 3187–3205

Detecting and locating burrs of industrial parts

D.-M. TSAI†‡ and W.-J. LU†

Burrs (excess material) are usually squeezed out around the periphery of the mould, and are irregular and vary considerably between parts. This paper aims at the development of a machine vision system for automatic detection of burrs and peripheral defects of casting parts. This non-contact detection result can be applied to automatic deburring systems and used for automatic inspection of peripheral breakdown.

This research concentrates on flat casting parts comprising piecewise smooth curves on the boundaries. The detection procedure is based on the fact that the irregular burrs show high curvature changes in a small segment of the boundary, whereas a smooth segment without burrs presents a succession of low curvature points. Given a casting part with burrs, we arbitrarily select a few boundary points from the smooth segments and connect these points to form a polygon. The matching process then finds a corresponding polygon from the model part. The transformation parameters of rotational angle and translation between the part in the image and the model part are evaluated from these two equivalent polygons. The proposed matching algorithm is computationally fast and the time complexity is bounded by $O(m)$, where m is the number of points on the digital boundary of the model part.

1. Introduction

Burrs are the excess material in the form of irregular and sharp metal fragments attached to the machined surfaces of a mechanical part. The location and form of burrs are seldom predictable with any real accuracy and differ widely between parts which are otherwise essentially the same. In foundries, burrs are usually squeezed out at the periphery of the mould and are irregular and vary considerably between parts. In this paper we aim at the development of a machine vision procedure for automatic detection of burrs and peripheral breakdown of industrial parts. The detected form and location of burrs can be applied to automatic robot deburring or fettling systems (Kazerooni 1988, Seliger and Hsieh 1991, Kramer and Shim 1990, Luo *et al.* 1983). Also, the non-contact detection results can be used for automatic visual inspection of peripheral defects of parts.

Okawa (1984) has proposed a method of detecting the fins (extruding portions) and notches (intruding portions) of a cast pulley. This algorithm is based on the estimation of the pulley's centre and radius, and is not extensible for detecting the burrs of arbitrary casting parts. Decker (1983) has studied automatic x-ray inspection of shrinkhole flaws inside aluminum casting parts. A flexible matching technique is applied, where the image under test is nonlinearly warped prior to forming the pixel-wise difference with the reference image. This method is computationally expensive and is only appropriate in the case of little noise and small displacement.

Revision received September 1995.

† Department of Industrial Engineering, Yuan-Ze Institute of Technology, 135 Yuan-Tung Road, Nei-Li, Taiwan, ROC.

‡ To whom correspondence should be addressed.

0020–7543/96 $12·00

Detection of burrs is equivalent to the recognition of partially occluded objects since the burrs attached to the periphery can be interpreted as the overlapping portions of the part. In order to bring down the complexity of object matching, the high dimensional parameter space is generally reduced to a lower dimensional space. φ-s curve (Gonzalez and Woods 1992) is a popular 1-D representation of 2-D boundaries, where φ is the angle plotted between a line tangent to the boundary and a reference line as a function of position s along the boundary. Cross-correlation (Perkins 1978), least-squares fit of salient subtemplates (Turney *et al.* 1985) and artificial neural network approaches (Tsang *et al.* 1992) are employed to recognize partially occluded objects in φ-s space. The measurement of tangent angle in a digitized image is sensitive to noise and quantization. Therefore, φ-s curve is not a reliable shape representation for locating the accurate position of an occluded part.

Lin and Chellappa (1987) have developed a method for the classification of 2-D partial shapes represented by the Fourier descriptors (Zahn and Roskies 1972, Persoon and Fu 1977). The matching problem is formulated as one of estimating the Fourier descriptors of the unknown complete shape from the observations derived from an arbitrary rotated and scaled shape with missing segments. Gorman *et al.* (1988) have obtained the local features by splitting a contour into segments which are described by the Fourier descriptors. A dynamic programming formulation is developed for matching the contours of a partial shape and the model shape.

Point pattern matching (Goshtasby and Stockman 1985, Stockman *et al.* 1982, Wong *et al.* 1983) is also used for object recognition. The point correspondence algorithms search in the parameter space and find the transformation parameters that can match the most points in two sets of points, one contains the points extracted from the scene image and the other one contains the points extracted from the model image. These feature points are usually identified by curvature-based corner detection techniques (Rattarangsi and Chin 1992, Bennett and MacDonald 1975, Rosenfeld and Johnston 1973). Point pattern matching is computationally expensive, and may fail for burr detection since many false feature points will be extracted from the burr segments and some actual feature points may be covered by the burrs.

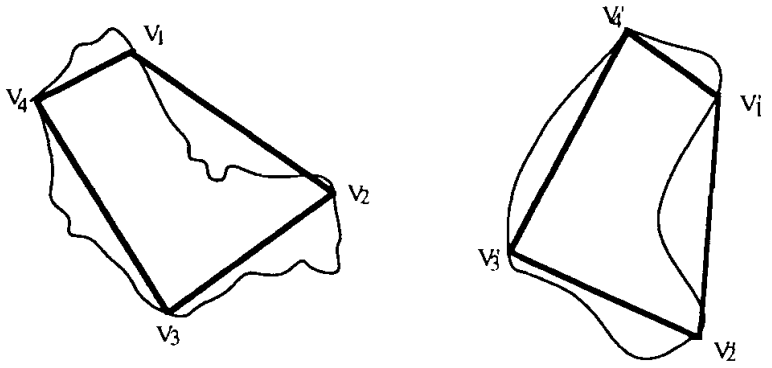
Davies (1989) has used the generalized Hough transform, GHT (Ballard 1981) for object recognition under the effect of occlusions. In the GHT process for the recognition of arbitrary shapes, the model of the shape is built by choosing a reference point r' and a set of points lying on the shape's boundary. The gradient direction at each point p_i' along the boundary of the model object is evaluated. Then it constructs a table, the so-called reference table, which stores the displacement vector $d = r' - p_i'$ and its corresponding gradient direction at point p_i'. In the recognition process, the gradient direction of every boundary point p_j of the scene object is used as a hash-key to find its corresponding displacement vector d in the reference table. The possible reference point of the scene object in the image is then given by $r = p_j + d$. An accumulator array is used to record the number of boundary points that generate the same reference point r. The maximum accumulated count in the array indicates evidence of a given model object present in the scene image. The Hough transform generally requires a tremendous amount of computing power and large storage.

Most of the partial object recognition systems developed so far are computationally expensive since the matching processes have relied on sophisticated optimization techniques such as dynamic programming (Gorman *et al.* 1988), relaxation labelling

(Rutkowski 1982, Davis 1979) or state-space searching (Chaudhury *et al.* 1990). Detection capability of these systems may rely on the appearance of salient features such as interior holes of parts and sharp corners on the boundaries. The estimated location of the object in the image is less accurate since the transformation parameters between the scene object and the model object are evaluated from the extracted features which are sensitive to noise and quantization. These recognition systems currently available are not practical approaches for detecting and locating burrs of parts in the industrial environment where high computational efficiency and high location accuracy are essential.

In this paper, we aim at the detection of burrs (extruding portions) and peripheral defects (intruding portions) of flat casting parts or those that can be represented in two dimensional space. Let model objects refer to the ideal casting part without any burrs or peripheral breakdown, whereas scene objects refer to the parts that may have burrs attached to the peripheries or have peripheral breakdown. The outermost boundaries of model objects under study comprise piecewise arbitrary, smooth segments. The shape and size of the burrs or peripheral defects to be recognized can be generated in arbitrary forms and in random locations, but they must not cover or destroy the entire periphery of the casting part.

The burr detection procedure is based on the fact that the irregular burrs show dramatic fluctuation of curvature changes in a small segment of the boundary, whereas a smooth segment without burrs presents a succession of low curvature points along the boundary. Given a scene object, we arbitrarily select a few boundary points from the smooth segments of the object and connect these points to form a polygon. The matching process then finds a corresponding polygon from the boundary of the model object based on the geometric properties of the polygons and the smooth segments. The translation and rotational angle between the scene object and the model object are evaluated from these two equivalent polygons. Finally, the model object is superimposed onto the image of the scene object using the resulting transformation parameters, and the mismatching portions are identified as the burrs or peripheral defects accordingly. Figure 1 gives a representative example of a scene object with burrs and its corresponding model object. V_1, V_2, V_3 and V_4 shown in Fig. 1(a) are four arbitrary points selected from the smooth segments of the



(a) Scene object with burrs (b) Model object

Figure 1. The scene object and the model object with two equivalent quadrangles shown by bold line segments.

scene object, which are used as four vertices to form a quadrangle. V'_1, V'_2, V'_3 and V'_4 shown in Fig. 1(b) are the four corresponding points on the boundary of the model object, which construct an identical quadrangle as that shown in Fig. 1(a). Identification of vertices V'_i in the model image, each corresponding to a vertex V_i in the scene image, is the main challenge of this matching task. The proposed burr detection algorithm does not rely on the salient features of industrial parts such as interior holes and sharp corners on the boundary, and is invariant to translations and rotations of the parts.

This paper is organized as follows: In § 2, the extraction process for smooth segments of a scene object is discussed. in § 3, formation of a polygon in the scene image and a searching process for finding the corresponding polygon in the model image are presented. The estimation of the transformation parameters between the scene object and the model object followed by a hypothesis testing for the correctness of the estimated transformation is included in this section. In § 4, experimental results are presented. Finally, the conclusion is given in § 5.

2. Extraction of smooth segments

In order to construct a polygon that can be duplicated in both scene image and model image, the vertices of the polygon associated with the scene object must lie on the smooth segments of the boundary. The irregular shape of burrs or peripheral breakdown generally shows a series of zigzag patterns, and this will create a series of high curvature points along the boundary. In contrast, a smooth segment of the casting part will show small directional changes along the boundary, and this will result in a series of low curvature points. The continuous version of curvature is defined as the magnitude of the rate of change of tangent slope with respect to the arc length. Many discrete algorithms (Rattarangsi and Chin 1992, Bennett and MacDonald 1975, Rosenfeld and Johnston 1973) have been developed to measure the curvature of a point lying on a digital curve.

In order to reduce noise and quantization effects on the curvature measurement, we define the curvature at a boundary point p_i as the angle between two fittings lines; one is estimated from a set of neighbouring points preceding p_i and the other one is estimated from a set of neighbouring points succeeding p_i. Let the sequence of n digital points describe a closed boundary P,

$$P = \{p_i = (x_i, y_i), \quad i = 1, 2, \ldots, n\}$$

where p_{i+1} is a neighbour of p_i (modulo n), and (x_i, y_i) are the Cartesian coordinates of p_i.

The curvature at point p_i is defined by

$$\Delta\varphi_i = \begin{cases} \tan^{-1}\left|\dfrac{m_1 - m_2}{1 + m_1 m_2}\right| & \text{if } m_1 m_2 \neq -1 \\ 90^\circ & \text{if } m_1 m_2 = -1 \end{cases}$$

$$0^\circ \leqslant \Delta\varphi_i \leqslant 90^\circ, \quad i = 1, 2, \ldots, n$$

where m_1 = the slope of the best fitting line, in the least-squares sense, given a set of

points $\{p_j, i-k \leqslant j \leqslant i\}$

$$= \left(\sum_{j=i-k}^{i} x_j y_j - k\bar{x}_1 \bar{y}_1 \right) \Big/ \left(\sum_{j=i-k}^{i} x_j^2 - k\bar{x}_1^2 \right) \tag{1}$$

$$\bar{x}_1 = \frac{1}{k} \sum_{j=i-k}^{i} x_j, \qquad \bar{y}_i = \frac{1}{k} \sum_{j=i-k}^{i} y_j$$

m_2 = the slope of the best fitted line given a set of points $\{p_j, i \leqslant j \leqslant i+k\}$, which can be obtained in a similar way as equation (1). k = the region of support that defines the neighbouring points of p_i.

Since the smooth segments of a casting part change the curve directions gradually along the boundary, a point on the smooth segment tends to yield a small intersection angle $\Delta\varphi$. In contrast, a point on the burr segments will result in large intersection angle because the slopes of two fitting lines are noticeably different. Given a model object comprising piecewise smooth linear and curved segments, the experimental results have shown that the curvature is generally less than 20° at any point except the corners on the boundary.

Let $\mu_{\Delta\varphi'}$ and $\sigma_{\Delta\varphi'}$ be the mean and standard deviation of intersection angles $\Delta\varphi_i'$ of the model object for all $\Delta\varphi_i' \leqslant 20°$. The intersection angles larger than 20° are excluded for the computation for $\mu_{\Delta\varphi'}$ and $\sigma_{\Delta\varphi'}$ since a corner point and a point on the burr segment are equivalent in terms of the curvature measurement. We then plot the $\Delta\varphi$-s curve of a scene object on a control chart, where $\Delta\varphi$ is the intersection angle as a function of position s along the boundary of the scene object. Assume the intersection angles $\Delta\varphi'$ of the model object are normally distributed. The control limits of the control chart are given by

$$[\mu_{\Delta\varphi'} - 3\sigma_{\Delta\varphi'}, \mu_{\Delta\varphi'} + 3\sigma_{\Delta\varphi'}]$$

The 3-sigma limits are determined so that an intersection angle $\Delta\varphi_i$ of the scene object has a 99·7% probability of falling in the interval of the control limits if this intersection angle is associated with a point on the smooth segment of the scene object. A point with $\Delta\varphi_i$ falling above the upper limit is interpreted as the one on the boundary of burrs.

Let a run be a succession of $\Delta\varphi_i$ falling below the upper limit $\mu_{\Delta\varphi'} + 3\sigma_{\Delta\varphi'}$ on the control chart. If the run length exceeds some predefined threshold T_r, the associated boundary points of the run are considered as a smooth segment without burrs, and the run is labeled as S_i for a smooth segment i. The length of a run is accumulated until a point falling above the upper limit. Then it is reinitialized to zero for subsequent runs. The region of support k for line fitting and run length threshold T_r are problem-dependent, and can be empirically determined. For determining an appropriate value of T_r, we have adopted a conservative principle that we prefer to misclassify a point on the smooth segment as a class of burrs rather than misclassify a point belonging to the burrs as a class of smooth segments since the vertices of polygons to be selected must be presented in both scene image and model image.

3. The matching algorithm

3.1. *Formation of a polygon in the scene image*

Let $S = \{S_1, S_2, \ldots, S_N\}$ be the sequence of N smooth segments extracted from the boundary of the scene object. We may arbitrarily select M boundary points

from the smooth segments in S and use these M points as the vertices to form an M-polygon. Based on the experimental results, we have found that $M = 3$ (i.e. a triangle) does not guarantee the success of burr detection, and $M = 4$ (i.e., a quadrangle) and $M > 4$ have resulted in similar performance. Since the increment of number of vertices M does not significantly improve the performance but increases the computational load in the matching process, quadrangle (4 vertices) is used to describe the matching algorithm developed in this paper and is suggested for practical implementation. However, the use of quadrangle is not a necessity of the proposed algorithm. The use of vertices larger than four is discussed in a later subsection.

If $N > 4$, i.e., more than four smooth segments are extracted, we select the midpoint of each smooth segment $S_j \in S$ as the vertex, where

$$j = \left[i \cdot \frac{N}{4} + 0{\cdot}5 \right] \quad \text{for } i = 1, 2, 3, 4$$

$[\cdot]$ represents the operation for integer truncation.

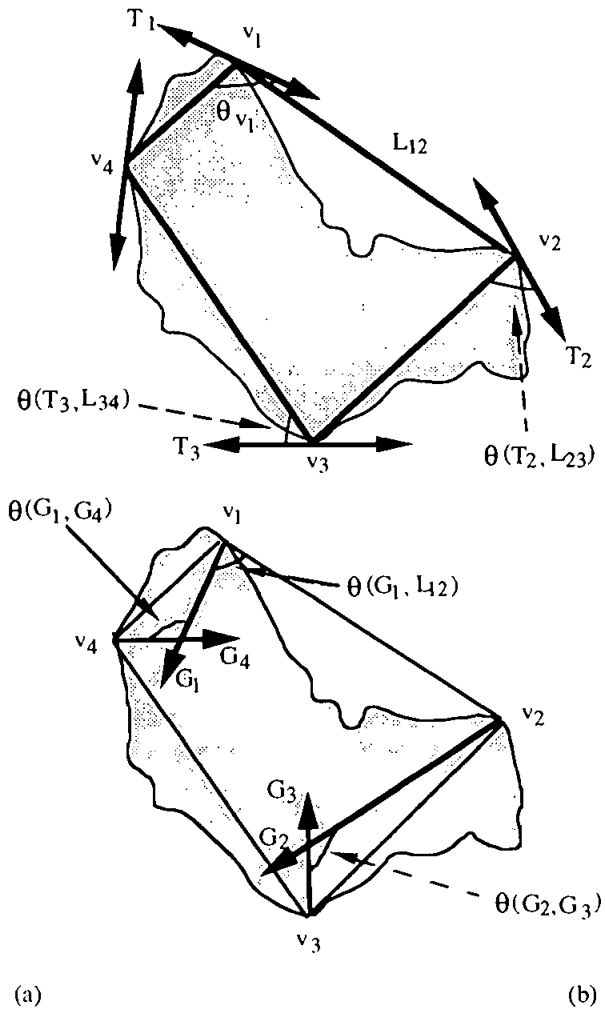


Figure 2. Geometric properties of the polygon and smooth segments.

Since the points toward the ends of each smooth segment may connect to burr segments, the mid-point of a smooth segment ensures the selected vertex has its counterpart showing on the boundary of the model object. S_j is selected so that the vertices will distribute equally in terms of the distances between vertices.

If $N = 3$, i.e., $S = \{S_1, S_2, S_3\}$, two vertices are selected from the points at the 1/3 and 2/3 of the smooth segment with the largest run length, and two more vertices are selected, individually, from the points at the 1/2 of the remaining smooth segments.

If $N = 2$, i.e., $S = \{S_1, S_2\}$, two vertices are selected from the points at the 1/3 and 2/3 of each smooth segment. The formation of a polygon will fail if S_1 and S_2 are straight line segments and parallel to each other.

If $N = 1$, i.e. $S = \{S_1\}$, four vertices are selected from the points at the 1/5, 2/5, 3/5 and 4/5 of the smooth segment. The formation of a polygon will fail if S_1 is a straight line segment.

Let $\{V_1, V_2, V_3, V_4\}$ be the set of four vertices selected from the smooth segments of the scene object. Given a scene object, denote

V_i = the ith vertex of the polygon, $i = 1, 2, 3, 4$
(X_i, Y_i) = the coordinates of V_i
L_{ij} = the line segment connecting vertex V_i and vertex $V_j, i, j = 1, 2, 3, 4, i < j$
m_{ij} = slope of line segment L_{ij}
d_{ij} = length of line segment L_{ij}
θ_{V_i} = interior angle at vertex V_i
= angle between line segments $L_{i-1,i}$ and $L_{i,i+1}$ (modulo 4)
T_i = tangent line to the smooth segment at vertex V_i
m_{T_i} = slope of tangent line T_i
G_i = gradient of the smooth segment at vertex V_i
m_{G_i} = slope of gradient $G_i = -1/m_{T_i}$
$\theta(G_i, G_j)$ = angle between gradients G_i and G_j
$\theta(G_i, L_{ij})$ = angle between gradient G_i and line segment L_{ij}
$\theta(T_i, L_{ij})$ = angle between tangent T_i and line segment L_{ij}

The geometric properties and relations defined above are illustrated in Fig. 2, and are computed as follows:

$$m_{ij} = (Y_j - Y_i)/(X_j - X_i) \tag{2}$$

$$d_{ij} = [(X_j - X_i)^2 + (Y_j - Y_i)^2]^{1/2}, \quad i, j = 1, 2, 3, 4, i < j \tag{3}$$

$$\theta_{V_i} = tan^{-1}\left| \frac{m_{i,i+1} - m_{i+1,i+2}}{1 + m_{i,i+1}m_{i+1,i+2}} \right|, \quad i = 1, 2, 3, 4 \quad \text{(modulo 4)} \tag{4}$$

The tangent line T_i to the smooth segment at vertex V_i is estimated by fitting the neighboring points of V_i in the region of support k to a straight line. Tangent slope m_{T_i} is computed using equation (1).

$$\theta(G_i, G_j) = tan^{-1}\left| \frac{m_{Gi} - m_{Gj}}{1 + m_{Gi}m_{Gj}} \right|, \quad i, j = 1, 2, 3, 4, i < j \tag{5}$$

$$\theta(G_i, L_{ij}) = tan^{-1}\left| \frac{m_{Gi} - m_{ij}}{1 + m_{Gi}m_{ij}} \right|, \quad i, j = 1, 2, 3, 4, i < j \tag{6}$$

θ_{V_i}, $\theta(G_i, G_j)$ and $\theta(G_i, L_{ij})$ are restricted to the first quadrant, i.e.

$$0° \leqslant \theta_{V_i}, \theta(G_i, G_j), \theta(G_i, L_{ij}) \leqslant 90°$$

$\theta(T_i, L_{ij})$ is defined in the full interval [0, 360°], and is computed by the following formula:

Let

$$\begin{aligned}(\Delta X_j, \Delta Y_j) &= (X_j - X_i, Y_j - Y_i) \\ &= \text{the vector from } V_i \text{ to } V_j \\ (1, m_{Ti}) &= \text{the tangent vector at vertex } V_i \\ \phi &= tan^{-1}\left|\frac{\Delta Y_j - m_{Ti}}{\Delta X_j - 1}\right|\end{aligned}$$

Hence,

$$\theta(T_i, L_{ij}) = \begin{cases} \phi, & \text{if } \Delta Y_j - m_{Ti} > 0 \text{ and } \Delta X_j - 1 > 0 \\ \pi - \phi, & \text{if } \Delta Y_j - m_{Ti} < 0 \text{ and } \Delta X_j - 1 > 0 \\ \pi + \phi, & \text{if } \Delta Y_j - m_{Ti} < 0 \text{ and } \Delta X_j - 1 < 0 \\ -\phi, & \text{if } \Delta Y_j - m_{Ti} > 0 \text{ and } \Delta X_j - 1 < 0 \end{cases}$$

3.2. *Searching for the polygon in the model image*

Given the quadrangle with vertices (V_1, V_2, V_3, V_4) in the scene image, the transformation parameters of translation and rotational angle between the scene object and the model object can be determined by detecting four corresponding vertices lying on the boundary of the model object. The quadrangle constructed in the model image must have the same geometrical properties defined in equations (3)–(6). Given a model object, denote

V'_i = the ith vertex of the quadrangle constructed in the model image, $i = 1, 2, 3, 4$. V'_i is assumed to be the counterpart of V_i
(X'_i, Y'_i) = the coordinates of vertex V'_i
L'_{ij} = line segment joining vertex V'_i and vertex V'_j
m'_{ij} = slope of L'_{ij}
d'_{ij} = length of L'_{ij}
θ'_{Vi} = interior angle at vertex V'_i
T'_i = tangent to the boundary at vertex V'_i
m'_{Ti} = slope of T'_i
G'_i = gradient of the boundary curve at vertex V'_i
m'_{Gi} = slope of $G'_i = -1/m'_{Ti}$
$\theta(G'_i, G'_j)$ = angle between G'_i and G'_j
$\theta(G'_i, L'_{ij})$ = angle between G'_i and L'_{ij}

Let $P' = \{p'_i = (x'_i, y'_i), i = 1, 2, \ldots, m\}$ be the set of m boundary points of the model object. Given an arbitrary vertex $V'_1 = p'_h \in P'$, $V'_j, j = 2, 3, 4$, can be estimated by

$$V'_j = \begin{bmatrix} X'_j \\ Y'_j \end{bmatrix} = \begin{bmatrix} X'_1 \\ Y'_1 \end{bmatrix} + \begin{bmatrix} cos\,\alpha & -sin\,\alpha \\ sin\,\alpha & cos\,\alpha \end{bmatrix} \begin{bmatrix} 1 \\ m'_{T1} \end{bmatrix} \frac{d_{1j}}{\sqrt{1 + (m'_{T1})^2}} \tag{8}$$

where $\alpha = \theta(T_1, L_{1j})$
$=$ the angle between tangent T_1 and line segment L_{1j} determined in the scene image.

The last term in the right-hand side of equation (8) computes the vector beginning at V'_1 and ending at V'_j. $V'_j, j = 2, 3, 4$, determined by equation (8) may not lie on the boundary of the model object because of quantization effect. We must adjust V'_j to its nearest boundary point. In order to minimize the computational requirements for the shortest distance from V'_j to the boundary, we have set up a X-Y correspondence table in which the x-coordinates of all boundary points in P' are sorted in non-decreasing order. X'_j determined by equation (8) is used as a pointer to find all corresponding y-coordinates of the boundary points that have their x-coordinates equal to X'_j from the X-Y correspondence table, and the one that has shortest distance to V'_j is selected. Due to the quantization effect, the k neighbouring points (region of support) of the adjusted V'_j are considered as the candidates for locating the true vertex associated with V_j. Note that the X-Y correspondence table is only set up once for the model object, and can be performed off-line.

V'_j determined by equation (8), for $i = 2, 3, 4$ is based on the assumption that V'_1 at a given point on the model boundary is the counterpart of V_1. We need to measure the degree of similarity between the quadrangle with vertices (V_1, V_2, V_3, V_4) and that with vertices (V'_1, V'_2, V'_3, V'_4). Let

$$\Delta d_{ij} = |d_{ij} - d'_{ij}|, \quad i, j = 1, 2, 3, 4, i < j$$

$$\Delta\theta_{Vi} = |\theta_{Vi} - \theta'_{Vi}|, \quad i = 1, 2, 3, 4$$

$$\Delta\theta_{ij} = |\theta(G_i, G_j) - \theta(G'_i, G'_j)| + |\theta(G_i, L_{ij}) - \theta(G'_i, L'_{ij})| + |\theta(G_j, L_{ij}) - \theta(G'_j, L'_{ij})|, \quad i, j = 1, 2, 3, 4, i < j$$

Δd_{ij} and $\Delta\theta_{Vi}$ measure the similarity between two quadrangles in terms of the segment lengths and interior angles, respectively. $\Delta\theta_{ij}$ further measures the geometric relations between the quadrangles and the smooth segments of parts. If (V'_1, V'_2, V'_3, V'_4) are the true counterparts of (V_1, V_2, V_3, V_4), Δd_{ij}, $\Delta\theta_{Vi}$ and $\Delta\theta_{ij}$ will have ideal values of zero. Define a similarity function for a given $V'_1 = p'_h \in P'$,

$$E(h) = E_d + E_\theta + E_G, \quad h = 1, 2, \ldots, m \tag{9}$$

where

$$E_d = \sum_{j=i+1}^{4} \sum_{i=1}^{3} \Delta d_{ij}$$

$$E_\theta = \sum_{i=1}^{4} \Delta\theta_{Vi}$$

$$E_G = \sum_{j=i+1}^{4} \sum_{i=1}^{3} \Delta\theta_{ij}$$

By searching through all model boundary points p'_h in P', the vertices (V'_1, V'_2, V'_3, V'_4) that yield minimal value of $E(h)$ are the best match of (V_1, V_2, V_3, V_4). Note that the measurement of $E(h)$ is invariant to translations and

rotations of objects. The detailed matching process is summarized as follows:

Given the model boundary points $P' = \{p'_h = (x'_h, y'_h), h = 1, 2, \ldots, m\}$,
Let $V'_1 = p'_h$ for $h = 1, 2, \ldots, m$
Compute $V'_j = (X_j, Y_j)$ using equation (8), for $j = 2, 3, 4$
Adjust V'_j to its nearest boundary point, denoted by p'_c, using the X-Y correspondence table.
Let $N_k(p'_c)$ represents the k neighbouring points of p'_c, i.e.,

$$N_k(p'_c) = \{p'_t, c - k \leqslant t \leqslant c + k\} \in P'$$

Let $V'_j = p'_t$ for $t = c - k, c - k + 1, \ldots, c, c + 1, \ldots, c + k$
Compute $e_t = \Delta d_{1j} + \Delta\theta_{1j}$
Let $e^* = \min\{e_t, c - k \leqslant t \leqslant c + k\}$
V'_j is assigned to the boundary point that generates minimal value e^*, provided that $\Delta d_{1j} < \epsilon_d$ and $\Delta\theta_{1j} < \epsilon_\theta$,

where ϵ_d and ϵ_θ are two predefined upper bounds on the amount of error inherent in measuring the lengths (Δd_{1j}) and angles ($\Delta\theta_{1j}$), respectively. Experimental results have shown the length error is less than five pixels and angle error is within 10°. Therefore, $\epsilon_d = 5$ and $\epsilon_\theta = 30$ (a summation of three angle differences) are used in this study. If either $\Delta d_{1j} > \epsilon_d$ or $\Delta\theta_{1j} > \epsilon_\theta$ for all points in $N_k(p'_c)$, it is concluded that V'_1 is a false vertex for V_1. Update V'_1 to the next boundary point in P' and repeat the matching procedure above.

For a given $(V'_1, V'_2, V'_3 V'_4)$, compute the value of the similarity function, $E(h) = E_d + E_\theta + E_G$, defined in equation (9).
Let $E^* = \min\{E(h), h = 1, 2, \ldots, m\}$
The vertices (V'_1, V'_2, V'_3, V'_4) that generate minimal value E^* are identified as the best match of (V_1, V_2, V_3, V_4). This concludes the matching process.

Note that the vertices V'_2, V'_3, and V'_4 are individually measured with respect to V'_1. No accumulated error is inherited for the estimation of $V'_j, j = 2, 3, 4$. Moreover, this makes the proposed matching procedure a parallel algorithm that can be implemented on a parallel processing machine. The computational complexity of this matching procedure is in the order of $O(m)$, where m is the number of boundary points of the model object, and is independent of the size of burrs of the scene object.

3.3. *Estimation of transformation parameters*

We can find a transformation from model points to scene points by considering the mapping of the vertices (V'_1, V'_2, V'_3, V'_4) in the model image to the corresponding vertices (V_1, V_2, V_3, V_4) in the scene image. The mapping consists of a rotation t_θ followed by a translation (t_x, t_y). The rotational angle t_θ is computed from the angle between two line segments, one connects V_i and V_{i+1} in the scene image and the other one connects V'_i and V'_{i+1} in the model image.

$$\text{Let } \Delta X_{i,i+1} = [X_i - X_{(i+1) \bmod 4}] - [X'_i - X'_{(i+1) \bmod 4}], i = 1, 2, 3, 4$$

$$\Delta Y_{i,i+1} = [Y_i - Y_{(i+1) \bmod 4}] - [Y'_i - Y'_{(i+1) \bmod 4}], i = 1, 2, 3, 4$$

$$\beta_{i,i+1} = \tan^{-1} \left| \frac{\Delta Y_{i,i+1}}{\Delta X_{i,i+1}} \right|, i = 1, 2, 3, 4$$

and

$$\beta_{i,i+1} = \begin{cases} \beta_{i,i+1}, & \text{if } \Delta Y_{i,i+1} > 0 \text{ and } \Delta X_{i,i+1} > 0 \\ \pi - \beta_{i,i+1}, & \text{if } \Delta Y_{i,i+1} < 0 \text{ and } \Delta X_{i,i+1} > 0 \\ \pi + \beta_{i,i+1}, & \text{if } \Delta Y_{i,i+1} < 0 \text{ and } \Delta X_{i,i+1} < 0 \\ -\beta_{i,i+1}, & \text{if } \Delta Y_{i,i+1} > 0 \text{ and } \Delta X_{i,i+1} < 0 \end{cases}$$

The rotational angle t_θ is given by the mean of $\beta_{i,i+1}$, i.e.,

$$t_\theta = \frac{1}{4}\sum_{i=1}^{4} \beta_{i,i+1}$$

The translation (t_x, t_y) is determined by the displacement between the centroid of the quadrangle (V_1, V_2, V_3, V_4) and that of the quadrangle (V'_1, V'_2, V'_3, V'_4). Hence,

$$t_x = \frac{1}{4}\left[\sum_{i=1}^{4} X_i - \sum_{i=1}^{4} X'_i\right]$$

$$t_y = \frac{1}{4}\left[\sum_{i=1}^{4} Y_i - \sum_{i=1}^{4} Y'_i\right]$$

By superimposing the model object onto the scene image, the extruding portions of burrs and the intruding portions of peripheral breakdown can be identified. (t_θ, t_x, t_y) give the parameter values that transform points of the model part to the scene image.

In order to verify the correctness of the estimated transformation, all boundary points lying on the smooth segments extracted from the scene object are mapped onto the model image using the transformation values (t_θ, t_x, t_y). Recall that $S = \{S_1, S_2, \ldots, S_N\}$ is the set of smooth segments extracted from the boundary of the scene object. Let $q_i = (x_{q_i}, y_{q_i})$ be a boundary point of the scene object and $q_i \in S$. The mapping of q_i onto the model image is given by

$$\begin{bmatrix} x'_{qi} \\ y'_{qi} \\ 1 \end{bmatrix} = T(t_\theta, t_x, t_y)^{-1} \begin{bmatrix} x_{qi} \\ y_{qi} \\ 1 \end{bmatrix} \forall_{qi} \in S$$

where

$$T(t_\theta, t_x, t_y) = \begin{bmatrix} \cos t_\theta & -\sin t_\theta & t_x \\ \sin t_\theta & \cos t_\theta & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

If (x'_{qi}, y'_{qi}) and its neighbouring points defined within a window of size $2W + 1$ are all background points in the model image, then we conclude that (t_θ, t_x, t_y) are the false transformation parameter values. If this case is encountered, we re-construct a polygon with five vertices (or more) from the set of smooth segments S and repeat the matching procedure. The size of the neighbourhood window $2W + 1$ specifies the tolerable error of the estimated transformation. The hypothesis verification ensures the success of the matching procedure and location accuracy of the scene object.

4. Experimental results

The algorithm discussed previously has been tested on seven artificial objects and two real aluminum casting parts. The algorithm is implemented in C on a PC/486 personal computer. The region of support k for line fitting is 15 (points). The run-length threshold T_r for smooth segment extraction is 30 (points). Quadrangles (four vertices) are used in the construction of a polygon in the scene image and in the process of matching vertices in the model image. Each scene object of study is performed in four different quadrants (orientations), three trials in each quadrant, so that the effect of object rotations can be observed.

All images are taken under natural room lighting without the support of any special light sources. In an industrial setting, we can use the backlighting technique for producing high-contrast images and eliminating the effect of arbitrary lighting of the environment. In backlighting, the light source is directed at the camera and is located behind the part to be recognized. The result of the image seen by the camera is a silhouette of the real part under study and therefore the boundary of the part can be reliably extracted. In this work, the boundary points in an input image are extracted using a simple boundary following technique described in Fairhurst (1988).

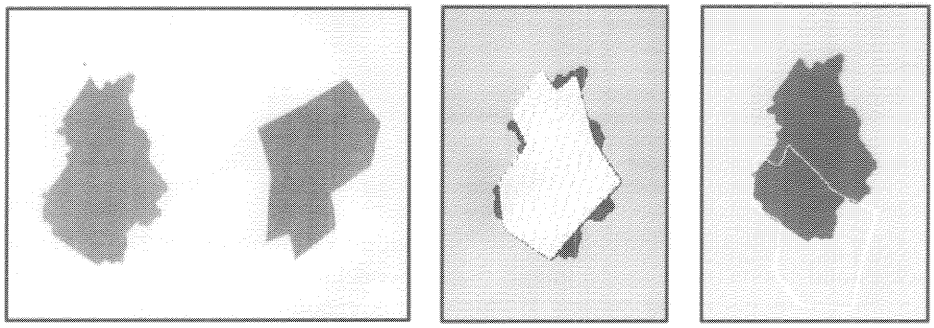
The seven artificial objects are planar, dark parts placed on a white background. The boundary lengths of these seven artificial model objects range from 457 to 807 pixels. The artificial scene objects contain simulated burrs with arbitrary shapes and in random locations. We have used the percentage of mismatching points, denoted by $P\%$, to evaluate the performance of the proposed algorithm, which is defined as

$$P\% = \frac{N_m}{A_m} \cdot 100\%$$

where N_m is the number of model points falling on the background of the scene image after the transformation.

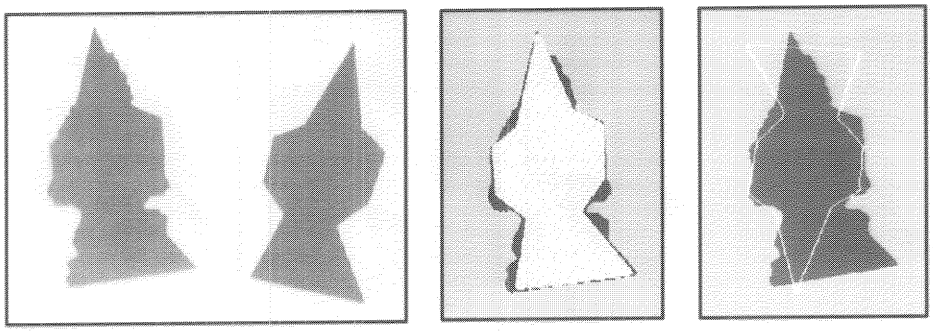
A_m is the area of the model object, and is defined by the total number of model points in the image.

Figures 3a through 9a illustrate the seven artificial objects. The left object in each image is a scene object with burrs, and the right object in the image represents the model object. Table 1 summarizes the percentages of mismatching points, $P\%$, for each of the seven artificial objects in four different quadrants. From Table 1, it can be seen that the average percentages of mismatching points, $P\%$, over the seven artificial



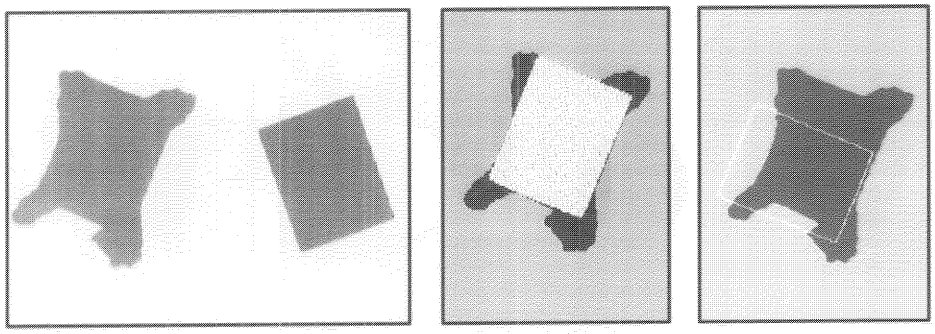
(a) (b) (c)

Figure 3. Test object 1: (a) scene and model images, (b) matching result of the proposed algorithm, (c) matching result of the GHT.



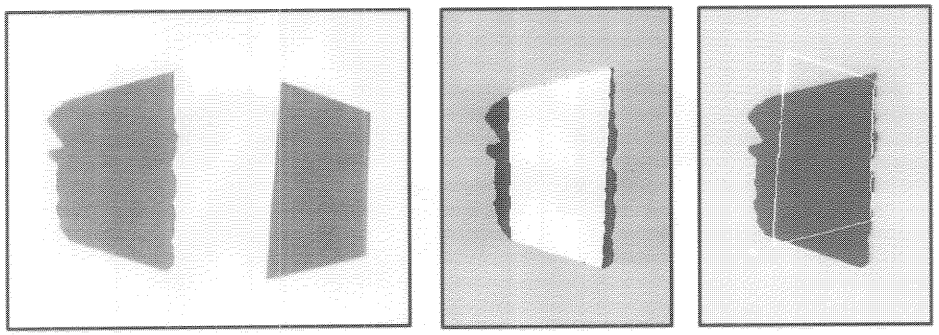
(a) (b) (c)

Figure 4. Test object 2: (a) scene and model images, (b) matching result of the proposed algorithm, (c) matching result of the GHT.



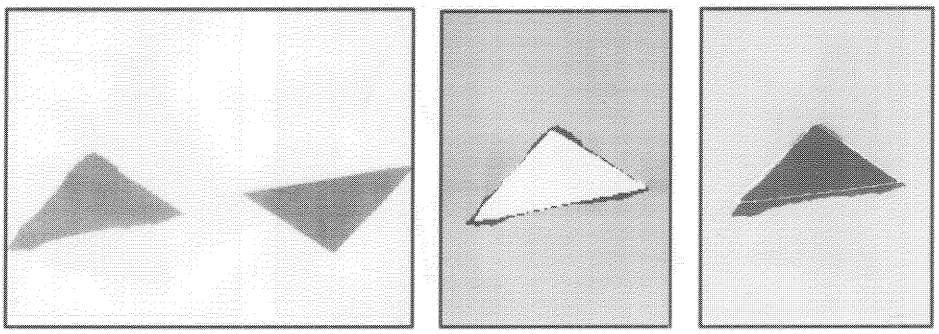
(a) (b) (c)

Figure 5. Test object 3: (a) scene and model images, (b) matching result of the proposed algorithm, (c) matching result of the GHT.



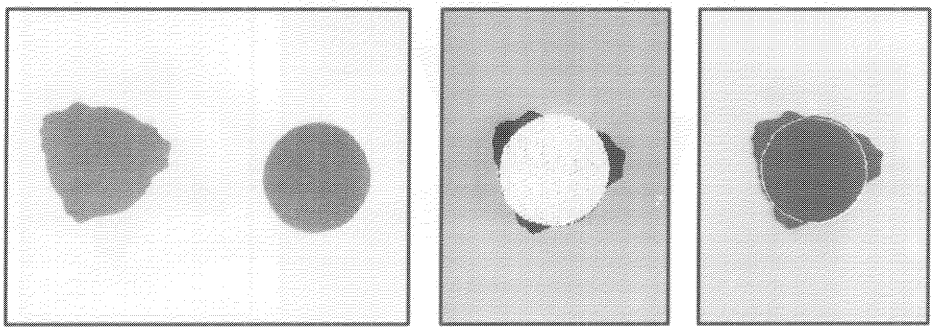
(a) (b) (c)

Figure 6. Test object 4: (a) scene and model images, (b) matching result of the proposed algorithm, (c) matching result of the GHT.



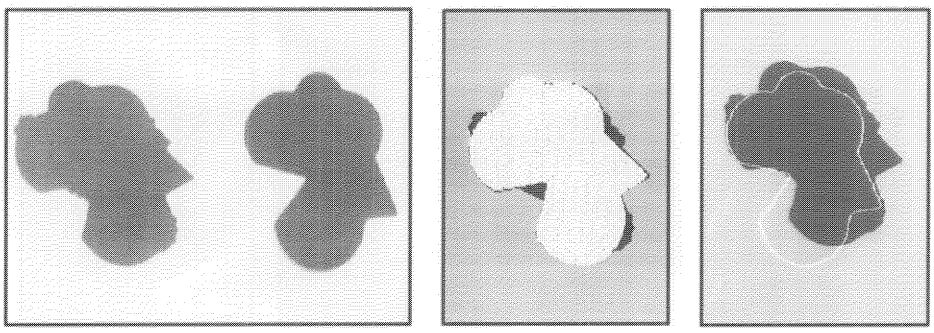
(a) (b) (c)

Figure 7. Test object 5: (a) scene and model images, (b) matching result of the proposed algorithm, (c) matching result of the GHT.



(a) (b) (c)

Figure 8. Test object 6: (a) scene and model images, (b) matching result of the proposed algorithm, (c) matching result of the GHT.

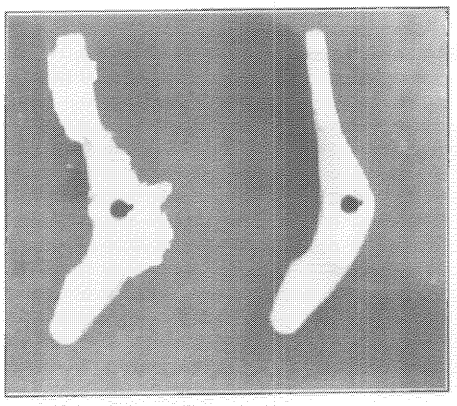


(a) (b) (c)

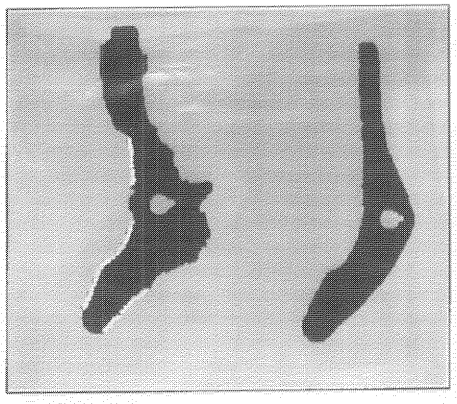
Figure 9. Test object 7: (a) scene and model images, (b) matching result of the proposed algorithm, (c) matching result of the GHT.

Test object	Orientation			
	Quadrant I	Quadrant II	Quadrant III	Quadrant IV
Object 1 (Fig. 3)	1·60	2·09	1·14	1·66
Object 2 (Fig. 4)	1·09	1·08	0·78	0·88
Object 3 (Fig. 5)	0·47	0·49	0·87	0·63
Object 4 (Fig. 6)	0·64	0·88	0·75	0·61
Object 5 (Fig. 7)	1·78	1·71	1·95	1·00
Object 6 (Fig. 8)	1·47	1·11	1·66	1·57
Object 7 (Fig. 9)	1·61	2·03	1·76	1·71
Overall average	1·23	1·34	1·27	1·15

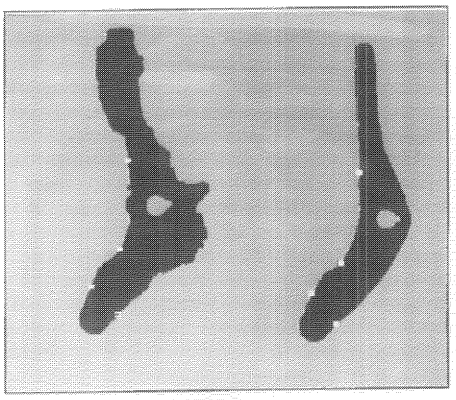
Table 1. Percentages of mismatching points $P\%$ in four quadrants (unit: %).



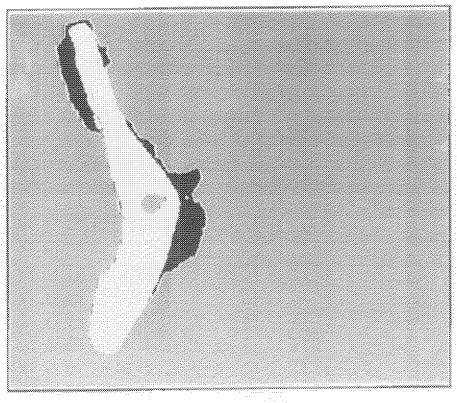
(a) The scene and model images of aluminum casting part A.



(b) Smooth segments extracted from the boundary of the scene object.



(c) The vertices, marked by the bright spots, that form the polygons.

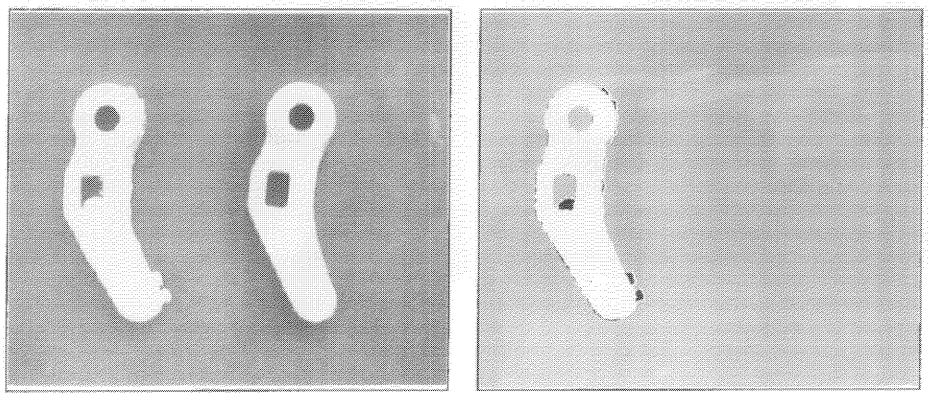


(d) The result of superimposing the model object on the scene image; the black portions shown in the image represent the burrs.

Figure 10. Burr detection for the aluminum casting part A.

objects are 1·23%, 1·34%, 1·27% and 1·15% in quadrants I, II, III, and IV, respectively. These four values are fairly consistent regardless of the rotations of the scene objects. The overall average of $P\%$ is 1·25%. Note that the measurement of $P\%$ includes the effects of quantization and lighting.

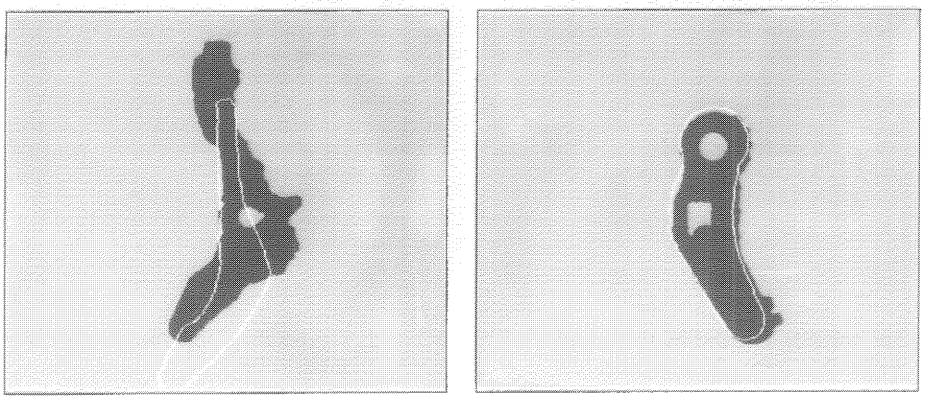
Figure 10a illustrates a real aluminum casting part, denoted by casting A. The scene object with burrs is shown at left and the model object is shown at right in the figure. A large portion of burrs is attached to the periphery of the scene object, as seen in Fig. 10a. The number of points on the digital boundary of model casting A is 1132. The bright lines shown in Fig. 10b are the smooth segments extracted from the boundary of the scene object. Figure 10c illustrates the four vertices (V_1, V_2, V_3, V_4) selected from the smooth segments of the scene object (the left), and the four corresponding vertices (V'_1, V'_2, V'_3, V'_4) on the boundary of the model object (the right). The bright spots shown in Fig. 10c indicate the locations of these vertices. Figure 10d shows the result of superimposing the model object on the scene image. The black portions represent the burrs of the scene object.



(a) The scene and model images of aluminum casting part B.

(b) The result of superimposing the model casting on the scene image using the proposed algorithm

Figure 11. Burr detection for the aluminum casting part B.



(a) (b)

Figure 12. The mapping results of the GHT for (a) the casting part A, and (b) the casting part B.

Figures 11a and 11b show another real aluminum casting part, denoted by casting B. The number of boundary points of model casting B is 982. Both Figures 10d and 11b have demonstrated high accuracy of the estimated transformation parameters. The mean percentages of mismatching points, $P\%$, for casting A (Figure 10a) and casting B (Figure 11a) are 2·64% and 2·54%, respectively. These two values are larger than 1·25% of the seven artificial objects due to the effects of reflection of aluminum surfaces and the thickness (10 mm on average) of the casting parts.

Since the generalized Hough transform (GHT) described in the introductory section is a well-known method for recognizing objects of arbitrary shapes under noisy or occlusion environment, it is also used in the experiments to compare with the proposed algorithm. Figures 3c–9c show the results for the seven artificial objects, and Figures 12a and 12b present the results for the real castings A and B using the generalized Hough transform. In these figures, the outermost boundaries of the model objects are mapped into the scene images. It can be seen from these figures that the generalized Hough transform fails in recognizing most of the nine test objects. It has performed poorly for polygonal objects such as those in Figs 3, 4, 5 and 6. The result is consistent with Grimson and Huttenlocher's finding (Grimson and Huttenlocher 1990) that the GHT can hypothesize many false solutions and its effectiveness is dramatically reduced for objects with moderate level of occlusion. The generalized Hough transform succeeds in recognizing object 5 (Fig. 7), object 6 (Fig. 8) and casting part B (Fig. 12b). However, the estimated poses (orientations and translations) of the GHT are not as accurate as those of the proposed algorithm. Therefore, the proposed algorithm compares very favorably with the generalized Hough transform for detecting burrs of casting parts.

5. Conclusion

In this paper, we have presented a method for detecting irregular burrs and peripheral defects of casting parts in arbitrary positions and orientations. There are two principal components to the method including smooth segment extraction in the scene image and polygon matching in the model image. The smooth segments of the scene object are identified by plotting the curvatures of points along the boundary on a control chart with a 3-sigma upper limit. Four (or more) points can be arbitrarily selected from the smooth segments and used as the vertices to form a polygon in the scene image. In the polygon matching process, the four vertices of the corresponding polygon in the model image are determined by taking a given point on the model boundary as a reference vertex and then computing the remaining vertices with respect to the reference vertex. A similarity function is defined to determine the best match of vertices, which measures the geometric properties between two polygons and their associated relations with the smooth boundaries. The computational complexity of this matching algorithm is $O(m)$, where m is the number of digital boundary points of the model object. Therefore, this algorithm is computationally efficient and is independent of the size of burrs on the scene object. The proposed matching method is a parallel algorithm that can be implemented on a parallel processing machine for real-time applications.

Since the vertices of the polygon constructed in the scene image are selected from the extracted smooth segments, the proposed algorithm for burr detection does not rely on any salient features such as interior holes or sharp corners on the boundary of a casting part. The success of this burr detection system is also independent of the coverage percentage of burrs to the entire boundary of the part if none of the

following are encountered: (1) the periphery of the scene object is completely surrounded by the burrs, (2) only one smooth segment is extracted, and this is a straight line segment, (3) only two smooth segments are extracted, and these two segments are parallel to each other, (4) all extracted smooth segments are concentric arcs. In addition, the proposed algorithm is based on the assumption that burrs show high curvature changes in the boundary. It is not directly applicable to scene objects that have burrs as smooth as the boundaries of the model objects.

References

BALLARD, D. H., 1981, The generalized Hough transform to detect arbitrary shapes. *Pattern Recognition*, **13**, 111–122.

BENNETT, J. R., and MACDONALD, J. S., 1975, On the measurement of curvature in a quantized environment. *IEEE Transactions on Computing*, C-**24**, 803–820.

CHAUDHURY, S., ACHARYYA, A., and SUBRAMANIAN, S., 1990, Recognition of occluded objects with heuristic search. *Pattern Recognition*, **23**, 617–635.

DAVIES, E. R., 1989, Occlusion analysis for object detection using the generalized Hough transform. *Signal Processing*, **16**, 267–277.

DAVIS, L., 1979, Shape matching using relaxation techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-**1**, 60–72.

DECKER, H., 1983, A difference technique for automatic inspection of casting parts. *Pattern Recognition Letters*, **2**, 125–129.

FAIRHURST, M. C., 1988, *Computer Vision for Robotic Systems: Introduction* (Englewood Cliffs, NJ: Prentice-Hall).

GONZALEZ, R. C., and WOODS, R. E., 1992, *Digital Image Processing* (Reading, Mass: Addison-Wesley).

GORMAN, J. W., MITCHELL, O. R., and KUHL, F. P., 1988, Partial shape recognition using dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **10**, 257–266.

GOSHTASBY, A., and STOCKMAN, G. C., 1985, Point pattern matching using convex hull edges. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-**15**, 631–637.

GRIMSON, W. E. L., and HUTTENLOCHER, D. P., 1990, On the sensitivity of the Hough transform for object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**, 255–274.

KAZEROONI, H., 1988, Robotic deburring of two-dimensional parts with unknown geometry. *Journal of Manufacturing Systems*, **7**, 329–338.

KRAMER, B. M., and SHIM, S. S., 1990, Development of a system for robotics deburring. *Robotics and Computer-Integrated Manufacturing*, **7**, 291–295.

LIN, C. C., and CHELLAPPA, R., 1987, Classification of partial 2-D shapes using Fourier descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-**9**, 686–690.

LUO, R.-C., SURESH, S., and GRANDE, D., 1983, Sensors for cleaning casting with robot and plasma-arc. *Proceedings of the 3rd International Conference on Robot Vision and Sensory Controls*, Cambridge, Massachusetts.

OKAWA, Y., 1984, Automatic inspection of surface defects of cast metals. *Computer Vision, Graphics and Image Processing*, **25**, 89–112.

PERKINS, W. A., 1978, A model based vision system for industrial parts. *IEEE Transactions on Computing*, C-**27**, 126–143.

PERSOON, E., and FU, K.-S., 1977, Shape discrimination using Fourier descriptors. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-**7**, 170–179.

RATTARANGSI, A., and CHIN, R. T., 1992, Scale-based detection of planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**, 430–449.

ROSENFELD, A., and JOHNSTON, E., 1973, Angle detection on digital curves. *IEEE Transactions on Computing*, C-**22**, 875–878.

RUTOWSKI, W. S., 1982, Recognition of occluded shapes using relaxation. *Computing in Graphics Image Processing*, **19**, 111–128.

SELIGER, G., and HSIEH, L.-H., 1991, Sensor-aided programming and movement adaptation for robot-guided deburring of castings. *Annals of the CIRP*, **40**, 487–490.

STOCKMAN, G. C., KOPSTEIN, S., and BENETT, S., 1982, Matching images to models for

registration and object detection via clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **4**, 229–241.

TSANG, P. W. M., YUEN, P. C., and LAM, F. K., 1992, Recognition of occluded objects. *Pattern Recognition*, **25**, 1107–1117.

TURNEY, J. L., MUDGE, T. N., and VOLZ, R. A., 1985, Recognizing partially occluded parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-7, 410–421.

WONG, C., SUN, H., YAMADA, S., and ROSENFELD, A., 1983, Some experiments in relaxation image matching using corner features. *Pattern Recognition*, **16**, 167–182.

ZAHN, C. T., and ROSKIES. R. Z., 1972, Fourier descriptors for plane closed curve. *IEEE Transactions on Computing*, C-**21**, 269–281.