Detection of Scarfing Faults on the Edges of Slabs

Sang-Gyu RYU,[1] Doo-chul CHOI,[1] Yong-Ju JEON,[1] Sang Jun LEE,[1] Jong Pil YUN[2] and Sang Woo KIM[1,3]*

1) Department of Electrical Engineering, POSTECH, Pohang, South Korea.
2) System Research Group, Engineering Research Center, POSCO, Pohang, South Korea.
3) Department of Creative IT Excellence Engineering, POSTECH, Pohang, South Korea.

(*Received on July 23, 2013; accepted on September 26, 2013*)

At present, quality control is becoming a major issue in steel production. Thus we developed an algorithm that uses machine vision to detect scarfing faults on slabs, which impairs the steel quality of subsequent products such as steel plates. Scarfing faults typically occur in three locations: the top, middle, and edge of the slab. Our proposed algorithm is focused on detecting scarfing faults on the edge of slab, which is tiny and sometimes indistinct. A machine vision system with a line scan camera was designed, which facilitates the detection of brightness differences and texture differences between well-scarfed and poorly-scarfed slab surface. Scarfing faults are tiny on the edges, so we propose a new segmentation method that takes advantage of capabilities of the line scan camera. A segmented image is filtered using Gabor filters, which were designed to focus on the boundary with scarfing faults to identify specific regions with defect, referred to as defect candidates. Each defect candidate is classified using a Support Vector Machine (SVM) classifier based on its extracted features. Our proposed algorithm was effective according to the experimental trials using 2 061 frame images acquired from real samples, where the true detection rate was 97.26% and the false detection rate was 1.66%. Our proposed system and algorithm based on machine vision technology facilitates scarfing faults detection, which can be detected before rolling process, resulting in improved steel quality.

KEY WORDS: quality control; machine vision; surface inspection; defect detection; line scan camera.

1. Introduction

Quality control is a major consideration during manufacturing, particularly in steel production, where marketplace is supersaturated and the quality is highly important if producers are to dominate markets. Further, the maturation of the global economy is improving, the standard of living and the production of consumer assets, such as automobiles and housing, is expanding, which demands a higher grade of steel.[1]

Quality control in steel production belongs to two categories: quality control on the production line and between each production line process. Quality control on the production includes testing correct composition during steel making, roller control, phase control, and the efficient plating. Quality control between each production line process includes inspections to detect internal or surface defects in steel products and eliminating such defects after each production line process is complete. Continuous production lines are widespread in steel factories, so the quality control processes conducted after each production line task include inter-production line inspections to eliminate defects on slabs, which are the semi-finished products of casting, before they move onto the rolling line.

There are many inspection systems, but machine vision-based surface defect detection has been applied widely in steel production[2] because this approach is advantageous in terms of its repeatability, consistency, and costs. A number of machine vision-based methods have been proposed to improve steel quality and the efficiency of manufacturing. The applications of this approach in steel production include the detection of various defects such as scratches, roller marks, hot roller marks, pinch roller marks, seam cracks, edge cracks, corner cracks, pin-holes, and blow-holes.

In their early applications to steel production, defect detection systems were focused on finished products such as Bar-in-coil (BIC) products and steel sheets. In BIC, cracks, stain lines, and spot defects are detected using designed edge preserving filter.[3] Holes, scratches, coil breaks and rusts are also detected in steel sheet using Hough transform and gray value analysis.[4] This approach improved the steel quality to some degree. However, quality of semi-finished products is known to affect the finished product, so studies have been conducted to improve the quality of semi-finished products, such as slabs[1,5] and billets.[6,7]

The algorithms to detect scarfing faults on slabs have also been proposed. For example, Choi[8] proposed detection algorithm using sine and cosine Gabor filter and Ryu[9] used multi-angle Gabor filter to detect scarfing faults in the middle of slab. However, these algorithms are unable to detect scarfing faults located in the edge regions, because of their small size and occurring location. Therefore, a new algorithm is required to detect scarfing faults in edge regions. In this present study, we developed an algorithm to detect

* Corresponding author: E-mail: swkim@postech.edu

DOI: http://dx.doi.org/10.2355/isijinternational.54.112

scarfing faults on the edges of slabs. System configuration and the details of our scarfing faults detection approach are described in the following section.

2. System and Target Defect

Slabs are semi-finished steel products, which are scarfed to remove small surface defects (oxide scales or other defects) that would impair the finished steel product quality such as Hot Rolled Coil (HRC), steel sheets, strips and other flat-rolled steel products. In some cases, slabs are not scarfed properly because of minor problems or control faults with the scarfing device: this phenomenon is known as faulty scarfing; resultant defects are called "Scarfing Faults". Unremoved surface defects due to faulty scarfing can degrade the quality of steel products[10] and damage the rollers, which then leads to continuous roller-mark defects or other defects. Therefore, scarfing faults need to be detected and corrected before the rolling process by partially grinding the scarfing faults. In this section, we describe the system configuration and provide type definitions of scarfing faults.

2.1. System Configuration for Scarfing Faults Detection

The scarfing faults detection system is configured as shown in **Fig. 1**. A slab is the semi-finished product before the rolling process, which measures 740–2 200 mm in width and 10 m in length. The cameras are aligned in the width direction of the slab and line-scan camera is used, so the image of long slab can be acquired efficiently. Line scan cameras produce no smearing so they have less motion blurring effects than area cameras. These advantages explain why many inspection systems use line scan cameras[11,12] and our proposed system exploits these advantages fully. For simplicity, a center camera is used as a single camera but multiple cameras can be used, depending on the actual slab width.

The continuous image arrays acquired by the line scan camera are saved to a server every 1 000 lines and the resolution used in the width direction is 4 096, so each image has a resolution of 4 096 × 1 000 pixel with an 8-bit gray level per image. Each pixel can represent 0.25 × 0.25 mm, and the length of the smallest scarfing fault can be detected in 40 to 120 pixels per image, the actual size of which is 1–3 cm. Cameras are installed perpendicular to surface of slab and lighting is installed diagonally at a 45° angle relative to the surface of the slab so there are differences in the light reflection from any uneven surfaces caused by faulty scarfing. Using the system described above, gray value differences and texture differences can be observed when scarfing faults are present (**Fig. 2**).

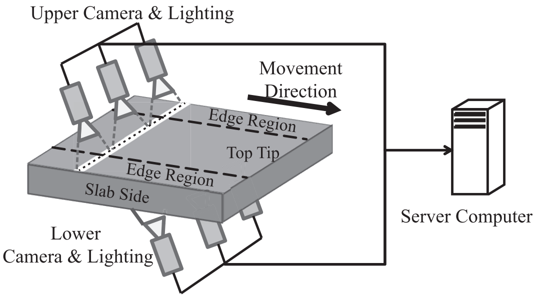

Fig. 1. System configuration for scarfing faults detection.

In the acquired image, the slab appears to be upside down when the line scan camera is used, so the start region of the scarfing process is present at the top of the image. In the following section, we describe the algorithm based on the image view-point, where the scarfing start point appears at the top.

2.2. Type Definition of Scarfing Faults

Scarfing fault can occur anywhere on a slab and it has different characteristics depending on its location, so it is not effective to examine the details of entire slab using a single algorithm. We defined scarfing faults into three types, depending on its location and applied different detection algorithm to each type. Each type of scarfing fault is shown in Fig. 2.

Type 1 Scarfing Fault (T1SF) occurs at the start of the scarfing process, so it is located at the top of the slab. Before starting the scarfing process, the scarfing device heats the top tip of the slab so it can scarf softly. However, minor problems with the heating unit may cause the top tip of the slab is melted partially and it leads to scarfing faults. If the problems with the the heating unit continue during the scarfing process, scarfing faults may appear in vertical direction (Fig. 2(a)). Before starting scarfing, the top of the slab is heated to a sufficient level so the top tip of the slab has a dented shape and it appears as a thick bright line in an image. Choi[8] examined discontinuities in the thick bright line at the top of slab images using a sine and cosine Gabor filter in the horizontal direction first. Then, he detected gray value differences on the boundary line, which started from the bright line, using a vertical directional Gabor filter. T1SFs are large in size, so they can be detected readily using this method.

Type 2 scarfing fault (T2SF) arises when the scarfing process is conducted in the middle of slabs. In scarfing process (Fig. 2(b)), partial trouble in scarfing device arises and scarfing fault occurs as it is spreading toward edges, resulting the boundary line appears to diagonal direction. T2SF is similar to T1SF, but there are two main characteristic differences: the boundary appears to be diagonal and it lacks a

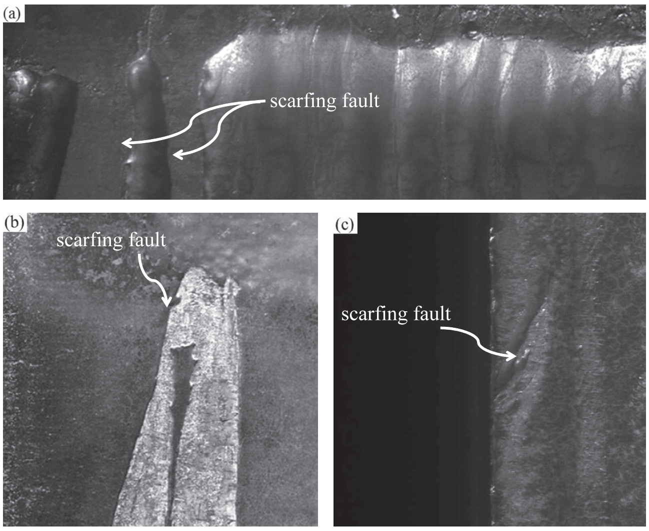

Fig. 2. Type definition of scarfing faults depending on the location: (a) T1SF in the scarfing start region; (b) T2SF in the middle of a slab; (c) T3SF in an edge region.

bright line at the top of the scarfing fault in the image. Therefore, T2SFs are detected using a diagonally directed Gabor filter based on the size and orientation after filtering by binarization of the Gabor filtered image.[9]

Type 3 scarfing fault (T3SF), which is the target defect of our proposed algorithm, occurs in the edge regions of slabs. Compared with the other scarfing faults, T3SFs are relatively small but they occurs more frequently (Fig. 2(c)). If the edge of a slab is not melted uniformly, scarfed splinters may remain or parts may be torn away so T3SF is appeared. The slab has a rounded shape so the edge regions of slab are darker than the center because of differences in the reflectance during illumination.[13] Thus, differences in the gray level values on the boundaries may not be apparent with T3SF. Therefore, it is necessary to examine the boundary line, as well as the texture difference along the boundary line, to detect T3SF. In the next section, we provide a step by step description of the T3SF detection algorithms.

3. Defect Detection Algorithm

In general, T3SF, the target defect, has two characteristics: a sharp change in the gray level value and texture differences on region of scarfing faults. Sharp changes in the gray level values are observed on the boundary of scarfing faults, and texture differences are observed between a well-scarfed and poorly-scarfed region. To avoid confusion, the gray value changes at the boundary of scarfing faults are referred as the "edge line" (**Fig. 3**) and the region that is inspected to detect T3SF is referred as the "edge region" (Fig. 1) in the following sections. In a single slab, T3SF can occur in four region: the left edge region of upper surface, the right edge region of upper surface, the left edge region of lower surface, and the left edge region of lower surface. However each edge region is similar if the left and right is reversed, so the proposed algorithm is described only based on the edge region of the left upper surface in the following sections.

First, the original image is segmented to separate the slab region from the background region. Next, the segmented image is filtered using Gabor filters which are designed to detect edge lines caused by faulty scarfing. The filtered image is binarized to find defect candidates and the features are extracted from the defect candidates. Finally, T3SFs are detected using pre-trained support vector machine (SVM) classifier with the extracted features (**Fig. 4**). Details of the defect detection algorithm are provided in the following subsections.

3.1. Slab Segmentation

The raw images acquired by the camera located on edge of slab needed to be segmented to separate the slab region from the background region (**Fig. 5**(a)). Methods to segment the slab image examine the vertical projection profiles of slab images to detect the boundary regions because the slab image is large, so a simple and rapid segmentation algorithm is used. After finding the boundary between slab and background, the slab image is segmented using a specific margin to prevent mis-segmentation, which results the failure of detection.

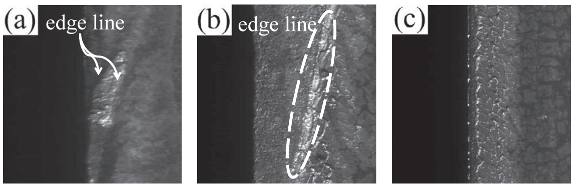

Fig. 3. Various T3SF and similar pseudo-defect: (a) Small size T3SF; (b) Messy texture T3SF; (c) Pseudo-defect.

The conventional segmentation methods use a simple sum of the vertical gray values as the projection profile (1). In some slab images, the black region contains a guide structures, which are used for aligning the slab; and may show the side of the slab; appear as peaks on the vertical projection profile (Fig. 5(b)). These peaks are not helpful to segment the slab accurately. To suppress unwanted peaks, gray values lower than a certain limit are ignored.[13] To prevent mis-segmentation, the images are also segmented with a large margins. However, when detecting T3SFs, segmentation with a large margin sometimes segment edges roughly, so removes the T3SFs. Furthermore, the method by Choi[13] is difficult to find an appropriate constant to limit the lower gray value.

We propose a segmentation method that uses vertical standard deviation as the projection profile (2). The proposed profile is simple to calculate, but it can reduce unwanted peaks on the projection profile (Fig. 5(b)). The peaks caused by the slab side are reduced in the proposed profile, and the slope of the profile is very sharp at the boundary between the slab and background regions. Using the proposed profile, the image can be segmented more accurately than by conventional segmentation, so T3SFs can be preserved. To remove noise, the vertical standard deviation profile is used after median filtering.

$$P_\mu(x) = \frac{1}{H}\sum_{i=1}^{H} I(x,i), \quad \text{........................} \; (1)$$

$$P_\sigma(x) = \sqrt{\frac{1}{H}\sum_{i=1}^{H}[I(x,i) - P_\mu(x)]^2}\,. \quad \text{...............} \; (2)$$

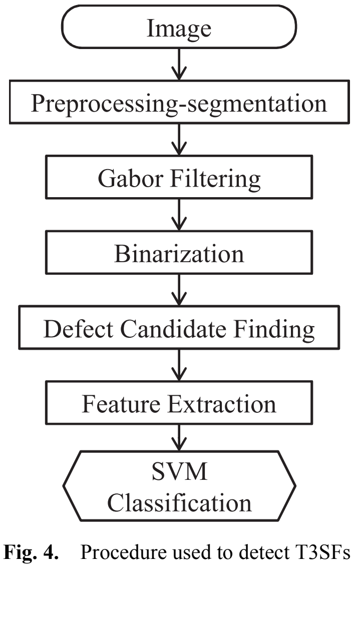

Fig. 4. Procedure used to detect T3SFs.

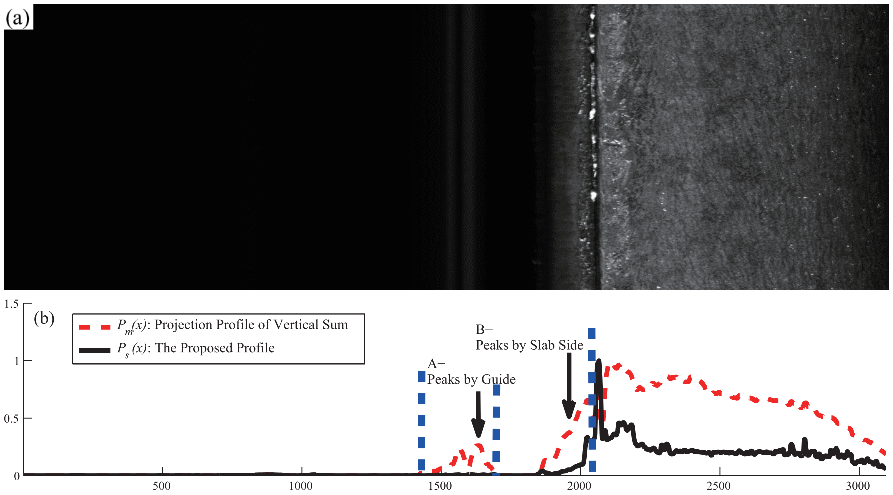

Fig. 5. Slab image and its profiles: (a) Raw slab image acquired from camera of edge region; (b) Vertical profiles of (a); unwanted peaks (A, B) are suppressed with the proposed profile. (Online version in color.)

3.2. Filter Design and Filtering

Gabor filter is used as a joint spatial, spatial-frequency representation with specific frequency and orientation characteristics.[14] The general form of Gabor function (3) can be split into a real part (4) and an imaginary part (5).

$$f(x,y) = exp\left(-\frac{x'^2+\gamma y'^2}{2\sigma}\right) exp\left(i(2\pi\frac{x'}{\lambda}+\psi)\right), \quad \text{....} (3)$$

$$g_{re}(x,y) = exp\left(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right) cos\left(2\pi\frac{x'}{\lambda}+\psi\right), \quad \text{....} (4)$$

$$g_{im}(x,y) = exp\left(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right) sin\left(2\pi\frac{x'}{\lambda}+\psi\right), \quad \text{....} (5)$$

where $x' = xcos\theta + ysin\theta \quad y' = -xsin\theta + ycos\theta$.

The real part of the Gabor function is a proven blob detector, while the imaginary part of the Gabor function is a proven edge detector.[15] Gabor filter is designed by selecting filter size and appropriate parameters: the orientation of the Gabor filter θ, wavelength of the sinusoidal factor λ, the gaussian envelope σ, phase offset ψ, and the spatial aspect ratio γ between x and y axis.

In early applications of Gabor filter, it was used for texture classification and texture segmentation by exploiting its ability to represent textures well.[16] In steel defect detection, Gabor filter is designed focusing on the characteristics of defect because surface of steel is inhomogeneous. Because Gabor filter can represent characteristics of defects well, it is widely used and proven to be effective.[3,7–9,13,17,18]

The filter design is also focused on the characteristics of T3SF, so the imaginary part of Gabor function is used for edge line detection, where ψ is set to 0 and γ is set as larger than 1 to consider the elongated shape of edge line. Because the orientation of edge line is various, we used four Gabor filters with different angle θ from 20° to 80°. With designed filters, the segmented image $S(x, y)$ is filtered and output image $R_i(x, y)$ is obtained as:

$$\begin{aligned} R_i(x,y) &= g_i(x,y) * S(x,y) \\ &= \sum_{m=1}^{M-1}\sum_{n=1}^{N-1} g_i(m,n)S(x-m, y-n), \end{aligned} \quad \text{..........} (6)$$

where $g_i(x, y)$ are imaginary part of the Gabor filters, with i represents angle of the Gabor filter and * denotes the convolution operation.

3.3. Defect Candidate Selection

After filtering the segmented images with each of the filter, the filtered images are binarized to find possible regions of T3SF, which are called as defect candidates. In binarization of slab images, classical methods such as Otsu's algorithm result in poor robustness,[19] because of irregular reflection of illumination from the nonuniform surface of the slab. Instead, to consider global textures, a method that selects threshold values adaptively is applied in the proposed algorithm. Global texture is considered by calculating the mean and standard deviation of the filtered image as follows:

$$T = mean[R_i(x,y)] + k \cdot std[R_i(x,y)], \quad \text{...........} (7)$$

where k is a constant determined empirically and *std* denotes standard deviation.

The value of the designed Gabor filter changes from positive to negative from left to right because the sine Gabor filter is used. Therefore, the edge line that is formed as the inverse distribution of the Gabor filter, which is a dark-to-bright transition, appears to be a high negative response on the filtered image. To consider both high positive and high negative response in filtered images, the energy of the Gabor filtered image $E_i(x, y)$ is used in binarization. Energy image can be calculated by simple squaring of the filtered image:

$$E_i(x,y) = R_i(x,y)^2 . \quad \text{..........................} (8)$$

In addition, to preserve the shape of the edge line and to remove unwanted irregular reflections, a double-threshold method is applied. First, the energy image is binarized with a low threshold value T_{low}. Then blobs, which are sets of adjacent pixels, are defined in binarized image $B_{Li}(x, y)$. If all pixels in each blob are lower than the high threshold T_{high}, that the blob is eliminated and the remaining blobs are the result of double-threshold. Consequently, the shape of edge line is preserved by T_{low} and unwanted irregular reflection responses are removed by T_{high}. T_{low} and T_{high} are deter-

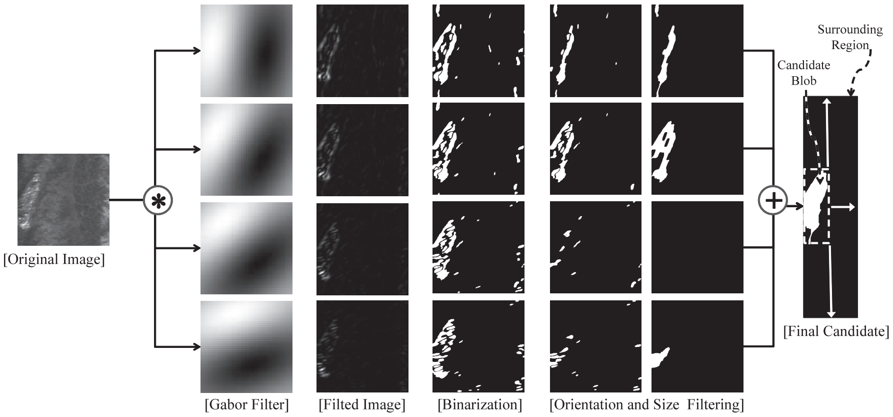

Fig. 6. Procedure used to find defect candidates.

mined by setting low and high k values in (7).

$$B_{Li}(x,y)=\begin{cases}1 & , \; if \;\; E_i(x,y)>T_{low} \\ 0 & , \; otherwise\end{cases}, \text{ .... } (9)$$

where $B_{Li}(x, y)$ is binarized image by low threshold value.

After binarization of each filtered images using the double-threshold method, small and unnecessary blobs are eliminated by using orientation and size filtering. The orientation and size filtered images are merged to form one image (**Fig. 6**) by simple OR operation. Finally, defect candidates are selected from the remaining blobs by considering their size[20] again on the merged image. By eliminating unnecessary defect candidates, computational cost can be reduced in the next feature extraction process.[9] In the following subsection, feature extraction from defect candidates is explained.

3.4. Feature Extraction from Defect Candidate

Conventional detection methods[8,9] for T1SF and T2SF are focused only on shape features when finding the edge lines, because these faults are relatively large so finding their edges are easy. By contrast, T3SF is relatively small, and its boundary line is unclear in some cases (Section 2.2). Therefore, both shape and texture features must be used to distinguish T3SF from pseudo-defects. The overall number of features is 29: eight shape features, 12 texture features, and nine proposed Histogram of Oriented Gradients (HOG) ratio. Each feature will be described in the following subsections.

3.4.1. Shape Features

By analyzing the shape features, a defect candidate that is blob from an edge line due to T3SF can be distinguished from a pseudo-defect caused by random surface texture or scales. The shape features comprise the simple properties of blob such as the width, height and area. Blobs from edge lines are narrow with a diagonal direction; these characteristics are considered by measuring the extent ratio (10) and the orientation of the blob. Orientation, minor axis length L_{minor}, and major axis length L_{major} can be obtained by drawing an ellipse that encloses the blobs, and measuring the ellipse's orientation and axis lengths (**Fig. 7**). In addition, characteristics of pseudo-defects are considered by measuring solidity (11) because some pesudo-defects have nonuniform shape and low solidity.[21]

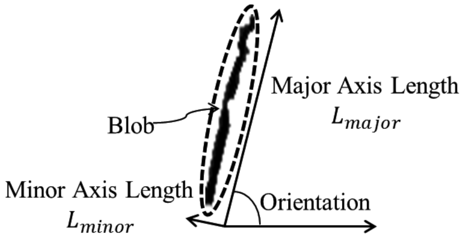

Fig. 7. Orientation, minor and major axis lengths.

$$\text{Extent ratio} = \frac{L_{minor}}{L_{major}}. \text{ .... } (10)$$

$$\text{Solidity} = \frac{Area}{Convex\ Area}. \text{ ....} (11)$$

3.4.2. Texture Features

To consider the texture differences, the candidate blob and its surrounding region (Fig. 6) are compared using 12 texture features; surrounding region is defined by setting minimum rectangular window containing candidate blob and extending it with some margin. In texture analysis, original segmented image $S(x, y)$ is used. The first six texture features are calculated based on properties of the candidates, *i.e.*, the mean μ, standard deviation σ, skewness *skew* (12), smoothness S (13), uniformity U (14) and entropy e (15). Features extracted from the defect candidate blob are absolute differenced by the features extracted from its surrounding region; the resulting values are the first 6 texture features. Because textures on scarfing faults are nonuniform and have high randomness compared to the surrounding region, these feature values are high if defect candidate is a real defect.

$$skew = \sum_{i=0}^{L-1}(x_i-\mu)^3\, p(x_i), \text{ .... } (12)$$

$$S = 1-\frac{1}{1+\sigma}, \text{ .... } (13)$$

$$U = \sum_{i=0}^{L-1} p^2(x_i), \text{ .... } (14)$$

$$e = -\sum_{i=0}^{L-1} p(x_i)\log_2 p(x_i), \text{ .... } (15)$$

where L is number of pixels, $p(x)$ is histogram of x_i normalized by its length and x_i are pixel values of $S(x, y)$ belonging to each region.

Fisher (16), Mahalanobis-Singh (17), and Unser (18) criteria[22] are also calculated to produce a better representation of the difference between two textures. Total six features are extracted with the criteria: three features from the original image and three from magnitude of gradient image. Magnitude of gradient image $M(x, y)$ is obtained as $M(x, y) = \sqrt{g_v(x,y)^2 + g_h(x,y)^2}$, where $g_v(x, y)$ is convolution result of original image with vertical gradient filter and $g_h(x, y)$ is convolution result of original image with horizontal gradient filter. We used $[-1\ 0\ 1]^{\mathrm{T}}$ as $g_v(x, y)$ and $[-1\ 0\ 1]$ as $g_h(x, y)$.

$$J_F = \frac{(\mu_D - \mu_S)^2}{\sigma_D^2 + \sigma_S^2}, \quad \text{......................... (16)}$$

$$J_{MS} = \frac{\mu_S}{\mu_D}, \quad \text{............................... (17)}$$

$$J_U = \frac{(\mu_D - \mu_S)^2}{\mu_D \mu_S}, \quad \text{......................... (18)}$$

where the subscript D represents "Defect candidate blob" and S represents "Surrounding region".

3.4.3. HOG Ratio

The HOG feature was developed to represent local directional trends[23] and is calculated in a sub-image, which is called cell. HOG divides orientation channels into a certain number and accumulates magnitude of gradient of pixels in a cell into a divided histogram channel of corresponding orientation. Orientation $O(x, y)$ is calculated using (19). In the accumulated values of histogram channels $H(n)$, one orientation corresponding to the maximum of the histogram channel values becomes the descriptor of the cell. In pedestrian detection,[23] one image is divided into multiple cells, and pedestrians are detected by HOG descriptors of each cell.

$$O(x,y) = \arctan\left(\frac{g_v(x,y)}{g_h(x,y)}\right). \quad \text{................. (19)}$$

The proposed HOG ratio is identical to the original HOG in using a gradient filter and calculating a gradient image. With a larger gradient filter than the original filter, which is [–101] or $[-101]^T$ gradient filter, directional tendency diminishes as the size of the gradient filter increases. The proposed method of accumulating the histogram channels is similar to the original HOG feature and uses a nine-channel histogram because this number of histogram channel delivered the most effective descriptors.[23] The unique part of the proposed method is two: 1) the proposed HOG ratio regards pixels of each candidate blob and surrounding region as different cells, so each defect candidate image has two cells, 2) it uses all values of the compiled histogram channel, not the one descriptor. To consider the difference of directional tendency between candidate blob and surrounding region, the values of histogram channels from a candidate blob are divided by the values of histogram channel of surrounding region, so nine HOG ratio are calculated. With the proposed HOG ratio, 50°–90° values of histogram channel are high in images containing T3SF, but flat or have a low center value in pseudo-defects (**Fig. 8**).

3.5. Defect Candidates Classification

In the final defect candidates classification, a SVM is used to distinguish defects and pseudo-defects. SVM[24] is based on structural risk minimization; it provides better generalization abilities compared to classical learning approaches, which follow empirical risk minimization (ERM).[25] Therefore, SVM performance in generalization is superior to conventional learning method,[26] and is also effective in steel image classification.[27]

We used SVM with the Radial Basis Function (RBF) kernel. The RBF kernel maps samples into a higher dimensional space, and therefore successfully classifies when the relation between class labels and attributes is nonlinear. The RBF kernel also contains fewer parameters than the polynomial kernels and has fewer numerical difficulties.[28] SVM is trained using 29 features extracted from defect candidates. The objective function J and RBF kernel $K(x_i, y_i)$ are

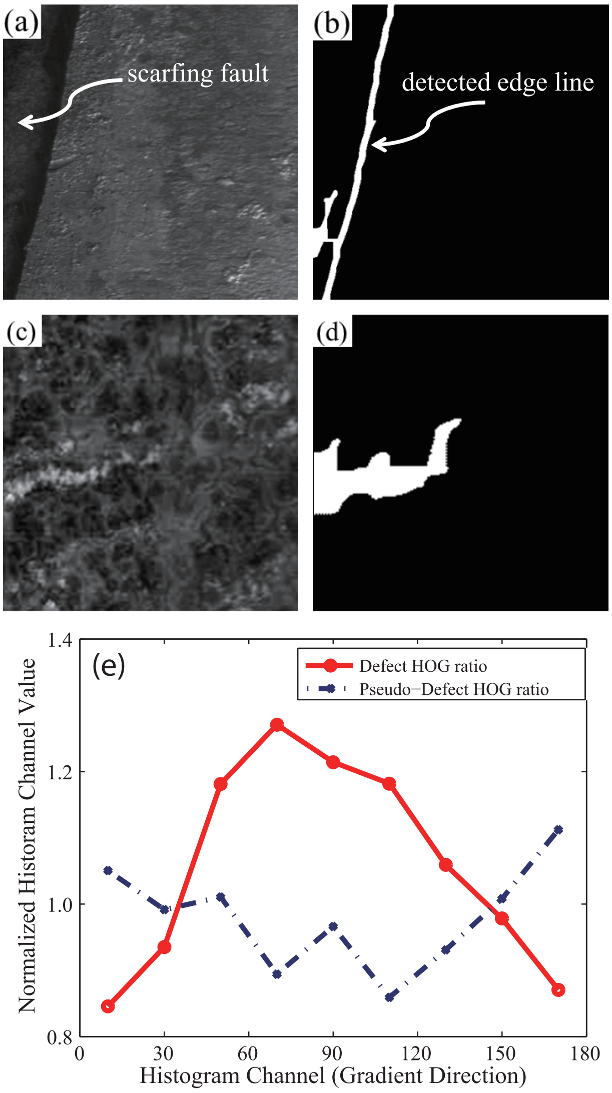

Fig. 8. Defect candidate images and proposed HOG ratio of each defect candidate: (a) Defect image; (b) Binarization result of (a); (c) Pseudo-defect image; (d) Binarization result of (c); (e) HOG ratio of defect and pseudo-defect. (Online version in color.)

Table 1. T3SF detection rates using the proposed methods on 73 images containing defect and 1 988 normal images.

	Accuracy		
Detection	Training	Test	Total
True	98.04% (50/51)	95.45% (21/22)	97.26% (71/73)
False	1.80% (25/1 391)	1.26% (8/597)	1.66% (33/1 988)

described in (20), (21):

$$J = \frac{1}{2}w^T w + C\sum_{i=1}^{N}\xi_i, \quad \text{......................} (20)$$

with constraint $y_i(w\cdot\phi(x_i)-b)\le 1-\xi_i$, $\xi \ge 0$, and $i = 1, \ldots, N$.

$$\begin{aligned} K(x_i,x_j) &= exp(-\gamma \text{P}x_i - x_j\text{P}^2) \\ &= \phi(x_i)\phi(x_j), for\, \gamma = \frac{1}{2\sigma^2}. \end{aligned} \quad \text{..........} (21)$$

4. Experimental Result

The validity of our proposed algorithm was tested with 2 061 slab images acquired from a real production line: 73 images containing T3SF and 1 988 normal images. The slab images were acquired using line scan cameras located on the edge region of slab and have 4 096 pixel × 1 000 pixel resolution. The original slab images were segmented using the proposed methods, and possible defect candidates were selected (Fig. 6).

Multiple T3SF and pseudo-defects can occur in a image, so the proposed algorithm found multiple defect candidate in some images. Consequently, the proposed algorithm found 96 defect candidates in the 73 images containing T3SF and 2 761 defect candidates in the 1 988 normal image. By classifying defect candidates with SVM, defect detection accuracy was calculated per image (**Table 1**).

The accuracy in experimental result is selected in balanced accuracy between true and false detection rate by iterating with different C values, σ values of the RBF kernel, and training sets: not partial to high true detection rate or low false detection rate. By iterating with different parameters of SVM, we can select a suitable SVM classifier for the production line. For high true detection rate, 97.89% true detection rate and 1.94% false detection rate would be appropriate. For low false-detection rate, 0.12% false detection rate, and 92.63% of defect accuracy would be the best classifier.

5. Conclusion

In this study, we developed an algorithm for detecting scarfing faults in edge regions, which we defined as T3SF. T3SFs are usually tiny, so we proposed an accurate slab image segmentation method. After segmenting the slab image accurately, defect candidates are selected by filtering them using a specially designed Gabor filter, followed by binarization. In defect candidate selection, many pseudo-defects are selected as defect candidates (2 761 defect candidates in the 1 988 normal image), so this step seems inefficient. However, the most important thing is that tiny and indistinctive T3SFs should not be missed in defect candidate selection, so we selected defect candidates liberally. Consequently, defect candidates included many pseudo-defects, which are ramdom textures.

Further, It is difficult to find the edge line of T3SFs, so our proposed algorithm exploited the shape and texture information. Using the extracted shape, texture features, and the proposed HOG ratio, scarfing faults are detected by a SVM classifier in the final stage. Our experiments using the proposed algorithm with real slab images demonstrated that our method was effective. Our proposed scarfing fault detection algorithm allows scarfing faults to be removed before the rolling process, which will improve the quality of finished products such as steel plates or steel coil.

Acknowledgements

This research was supported by the MSIP(Ministry of Science, ICT and Future Planning), Korea, under the "IT Consilience Creative Program" (NIPA-2013-H0203-13-1001) supervised by the NIPA(National IT Industry Promotion Agency).

REFERENCES

1) H. Kagechika: *ISIJ Int.*, **47** (2007), 773.
2) P. P. Sahoo, A. Kumar, J. Halder and M. Raj: *ISIJ Int.*, **49** (2009), 521.
3) J. P. Yun, S. H. Choi, B. Seo and S. W. Kim: *Opt. Eng.*, **47** (2008), 077204.
4) M. Sharifzadeg, S. Alirezaee, R. Amirfattahi and S. Sadri: 12th IEEE Int. Multitopic Conf., IEEE, New York, (2008), 125.
5) L. M. Zhao, Q. Quyang, D. F. Chen and L. Y. Wen: *Ironmaking and Steelmaking.*, **38** (2011), 464.
6) M. J. Cho, E. B. Park and S. W. Kim: *ISIJ Int.*, **50** (2010), 1180.
7) J. P. Yun, S. H. Choi, J. W. Kim and S. W. Kim: *NDT&E Int.*, **47** (2009), 389.
8) D. Choi, Y. J. Jeon and S. W. Kim: *Adv. Mater. Res.*, **462** (2012), 185.
9) S. G. Ryu, Y. J. Jeon, D. Choi, J. P. Yun and S. W. Kim: ICROS 28th Annual Conf., ICROS, Buchun, (2013), 278.
10) Y. Kondo, H. Tanei, N. Suzuki, K. Ushioda and M. Maeda: *ISIJ Int.*, **51** (2011), 1696.
11) K. L. Mak and P. Peng: *Robotics and Comput. -Integ. Manuf.*, **4** (2008), 359.
12) J. L. Mou, Y. M. Sun, Y. C. Chen, C. W. Liao and Y. S. Tarng: *ISIJ Int.*, **47** (2007), 1280.
13) D. Choi, Y. J. Jeon, J. P. Yun and S. W. K: *Appl. Opt.*, **50** (2011), 5122.
14) M. Porat and Y. Y. Zeevi: *IEEE Trans. on Pattern Anal. and Mach. Intel.*, **10** (1988), 452.
15) D. Casasent and J. S. Smokelin: *Opt. Eng.*, **33** (1994), 2255.
16) P. Kruizinga, N. Petkov and S. E. Grigorescu: *IEEE Trans. on Image Process.*, **11** (2002), 1160.
17) G. Kang and H. Liu: Proc. of 4th Int. Conf. on Mach. Learn. and Cybern., IEEE, NewYork, (2005), 5034.
18) D. M. Tsai and S. K. Wu: *The Int. J. of Adv. Manuf. Tech.*, **16** (2000), 474.
19) Q. Chen, Q. Sun, P. A. Heng and D. Xia: *Pattern Recogn.*, **41** (2008), 1254.
20) R. C. Gonzalez and R. E. Wood: Digital Image Processing, Vol. 3, Prentice Hall, New Jersey, (2002), 668.
21) A. Bosson, G. C. Cawley, Y. Chan and R. Harvey: Int. Conf. on Image and Video Retrieval, Springer, New York, (2002), 50.
22) T. Randen and J. H. Husoy: *IEEE Trans. on Image Process.*, **8** (1999), 571.
23) N. Dalal and B. Triggs: Conf. on CVPR, IEEE, New York, (2005), 886.
24) C. Cortes and V. N. Vapnik: *Mach. Learn.*, **20** (1995), 283.
25) B. Gutschoven and P. Verlinde: Proc. of the 3rd Int. Conf. on Inform. Fusion, IEEE, New York, (2000), THB3/3.
26) H. Byun and S. Lee: *Pattern Recog. with SVM*, **2388** (2002), 213.
27) H. Jia, Y. L. Murphey, J. Shi and T. S. Chang: Proc. of 4th Int. Conf. on Pattern Recog., IEEE, New York, (2004), 239.
28) C. -W. Hsu, C. C. Chang and C. J. Lin: A Practical Guide to Support Vector Classification, National Taiwan University, Taipei, (2003), 1.