DOI: 10.1007/s12541-010-0079-z

Morphological Segmentation Under Complex Backgrounds Using Enhanced Gray-Scale Hit-or-Miss Transform

Taek-Guen Joung[1], Hyonam Joo[1,#], Keun-Ho Rew[2] and Yong-Shin Lim[1]

1 Department of Display Engineering, Hoseo University, Asan, Chungnam, South Korea, 336-795
2 Department of Robotic Engineering, Hoseo University, Asan, Chungnam, South Korea, 336-795
Corresponding Author / E-mail: hnjoo@hoseo.edu, TEL: +82-41-540-5969, FAX: +82-41-548-0650

KEYWORDS: Machine Vision, Mark Extraction, IC Package Inspection, Hit-or-miss Transformation, Frame Filter

In the process of packaging for integrated circuits or flexible PCBs, special characters are printed to indicate the status of quality control. During the automated inspection process, these characters are detected and indentified by the inspection equipment. The characters are sometimes difficult to be segmented from the images due to the complex backgrounds, which are complex circuit patterns and other graphic features. In this paper, we develop character segmentation algorithm based on grayscale hit-or-miss morphological transformation. The character's width is assumed to be known. Performance of the proposed algorithm is extensively tested, and the results show successful accuracy.

Manuscript received: December 26, 2008 / Accepted: May 31, 2010

NOMENCLATURE

HMT = Hit-or-miss transform
GHMT = Grayscale hit-or-miss transform
Hset = Hit-set
Mset = Miss-set
Hval = Hit-value
Mval = Miss-value
SNR = Signal-to-Noise Rate
B_1 = Foreground pattern of target pattern
B_2 = Background pattern of the target pattern
Z = Set of all natural numbers
p = Pixel coordinates
w = Width of character
u_i = Pixel coordinates of *Hset*
v_i = Pixel coordinates of *Mset*
k = Sorting order
n = Number of pixels in *Hset*
m = Number of pixels in *Mset*
max_k = Maximum of smallest k pixel values
min_k = Minimum of largest k pixel values
$avgS_k$ = Average of smallest k pixel values
$avgL_k$ = Average of largest k pixel values
S = Strength of character signal
δH = Illumination difference form center to edge

1. Introduction

Algorithms for character separation and segmentation from complex backgrounds have been actively investigated among many image processing researchers.[1-22] In general, the character segmentation algorithms are divided into ones based on statistical method and the others based on structural method using shape and texture modeling, as shown in Fig. 1.

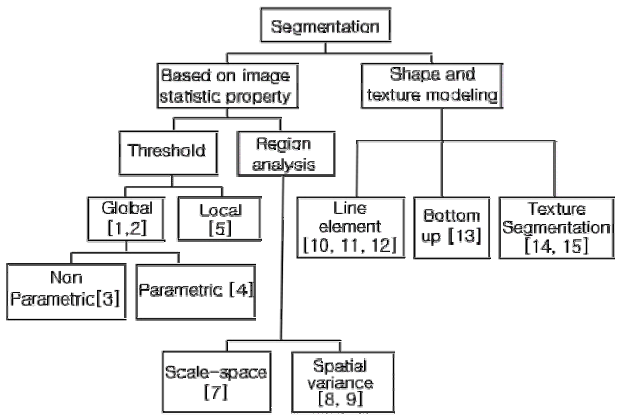


Fig. 1 Hierarchy of segmentation methods

In this paper, statistical character segmentation is researched on IC packages and flexible PCBs as shown in Fig. 2. Character extraction from complex background is a hard problem and much research is devoted on this problem.[17-22] In segmenting characters, the well-known threshold method only performs well under simple background condition.[1-6] In 1996, Khosravi and Schafer[18] introduced the gray-scale hit-or-miss transform (GHMT) and showed its usefulness in template matching. Since then, the GHMT has been used in solving many image processing problems.[23-25] In 2004, Ye, et. al.[19] assumed a known character with and extracted characters by thresholding the result of a carefully-designed morphological filter. In this paper, we propose a character segmentation algorithm based on GHMT under the assumption that the character width is approximately known. A set of new structuring elements is developed to segment characters together with a set of nonlinear functions to compute the morphological operations. With these added features, the proposed algorithm performs well compared with the simple GHMT.

(a) character on IC package (b) character on flexible PCB

Fig. 2 Characters under complex backgrounds

2. Gray-scale Level Character Extraction using Hit-or-miss Transform

2.1 Hit-or-miss transform

The hit-or-miss transform (HMT) is the basic tool for pattern detection from binary image. HMT of image A by the set $B = (B_1, B_2)$ is denoted as $A \otimes B$, and is defined as follows:

$$A \otimes B = (A \ominus B_1) \cap (A^c \ominus B_2) \tag{1}$$

Two structuring elements, B_1, and B_2, are used for the set B. The set B_1 represents the foreground pattern while the set B_2 represents the background pattern of the target pattern. The $A \ominus B$ means A is eroded by B.

By using the theorems of morphological complementation, Eq. (1) can also be expressed as the following.

$$A \otimes B = (A \ominus B_1) - (A \oplus \hat{B}_2) \tag{2}$$

where $\hat{B}_2$ means the reflection of B_2 about the origin.

The HMT is also used widely in finding convex hull, thinning, thickening, skeletons, and pruning, where the detection of a binary pattern is required.

2.2 Proposed algorithm

The gray-scale hit-or-miss transform (GHMT) of a signal f with respect to a structuring element (template) t with support Spt(t) is introduced in[18] and is defined as:

$$\begin{aligned}(f \otimes t)(k) &= (f \ominus t) + (-f \ominus (-t)) \\ &= \min_{n \in Spt(t)} \{f(n+k) - t(n)\} \\ &\quad + \min_{n \in Spt(t)} \{-f(n+k) + t(n)\}\end{aligned} \tag{3}$$

Where the gray-scale morphological erosion is defined as:

$$(f \ominus t)(k) = \min_{n \in Spt(t)} \{f(n+k) - t(n)\} \tag{4}$$

The first gray-scale erosion in Eq. (3) ensures that the template t matches the signal f from above and the second gray-scale erosion ensures that the template t matches the signal f from below. Therefore, the GHMT which is the superposition of the two erosions has a maximum value of zero only at the location of the template.

The GHMT can also be written in the following form:

$$\begin{aligned}(f \otimes t)(k) &= \min_{n \in Spt(t)} \{f(n+k) - t(n)\} \\ &\quad - \max_{n \in Spt(t)} \{f(n+k) - t(n)\}\end{aligned} \tag{5}$$

We modify the Eq. (5) slightly and define the following GHMT. Let Z be the set of all natural numbers, $p = (x, y) \in Z \times Z$ be the pixel coordinates, $f(p)$: $Z \times Z \rightarrow Z$ be the pixel values at location p. In our application, when the width of characters is assumed to be w, the Hit-set (*Hset*) and the Miss-set (*Mset*) can be defined as Fig. 3.

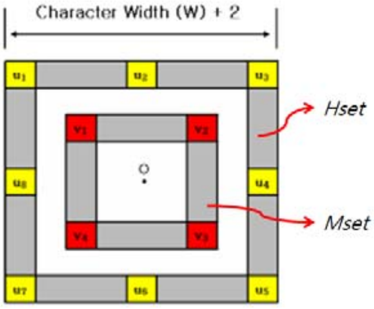


Fig. 3 Illustration of *Hset* and *Mset* used for character segmentation

The *Hset* is the set composed of n pixels on a w×w frame or a ring of radius w/2 centered at origin and is used to match the pixels hitting background.

$$Hset \subseteq \{u_i\},\ i = 1, \ldots\ldots\ldots, n \tag{6}$$

The *Mset* is the set composed of origin, and/or m pixels on a w/2×w/2 frame and is used to match the pixels missing the character background.

$$Mset \subseteq \{O, v_i\},\ i = 1, \ldots\ldots, m$$

The shape of *Hset* and *Mset* can be designed according to the application that GHMT is used. We also define the Hit-value (*Hval*) and Miss-value (*Mval*) at pixel coordinate p as follows which replaces the morphological erosion operations in the original GHMT:

$$\begin{aligned}Hval(p) &= \mathcal{H}_{u \in Hset} \{f(p+u)\} \\ Mval(p) &= \mathcal{M}_{v \in Mset} \{f(p+v)\}\end{aligned} \tag{7}$$

where $\mathcal{H}$ and $\mathcal{M}$ are selected among the followings:

- maximum of smallest k values ($\max_k$)
- minimum of largest k values ($\min_k$)
- average of smallest k values ($avgS_k$)
- average of largest k values ($avgL_k$)

where $k \le n$ and $k \le m$ for *Hval*(*p*) and *Mval*(*p*), respectively. From Eqs. (6) and (7), the modified GHMT proposed in this paper is defined as follows:

$$GHMT(p) = Hval(p) - Mval(p) \tag{8}$$

When the characters are darker than the background, $\{\mathcal{H} = \min_k, \mathcal{M} = \max_k\}$ is adopted to get positive values at the pixels where the characters are located. Fig. 4 shows the flowchart of the proposed algorithm, and the sample results of each step are illustrated in Fig. 5.

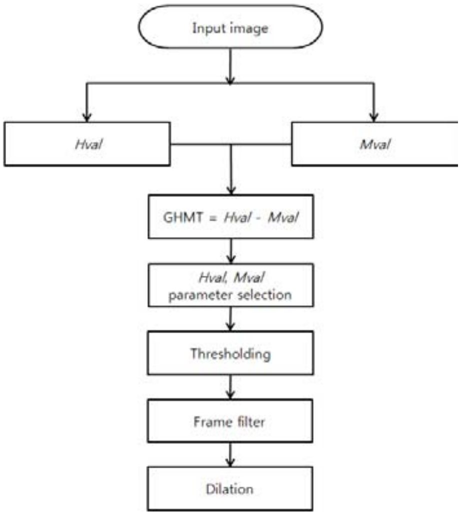
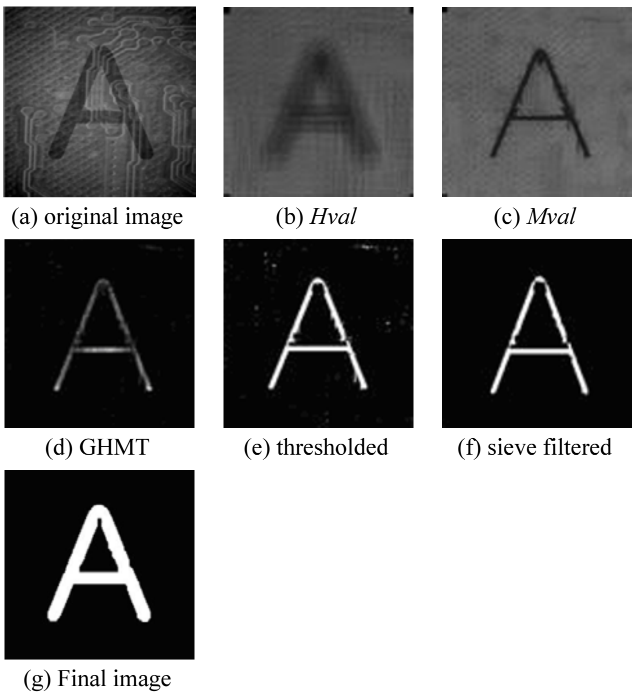


Fig. 4 Flowchart of proposed algorithm



(a) original image (b) *Hval* (c) *Mval*
(d) GHMT (e) thresholded (f) sieve filtered
(g) Final image

Fig. 5 Images at each step of the proposed algorithm

In Fig. 5(a) is the original image and (d) shows the resulting GHMT image. The structuring elements used are: *Hset* with n = 8 and *Mset* with m = 4 with origin O, and the operators are selected as: *Hval* = $avgL_k$ and *Mval* = $\max_k$. As expected the central portions of the relatively dark character pixels get positive values in the GHMT image [5(d)]. Fig. 5(b) and 5(c) show the *Hval* and *Mval* function values, respectively, to help understand the GHMT image.

To recover the whole character pixels [Fig. 5(g)], a morphological dilation with the known character width is performed after threshoding [Fig. 5(e)] the GHMT image with a proper threshold value. Before dilation, to get rid of small noise we use a frame filter, which is a sort of sieve filter and the result is shown in Fig. 5(f). The flowchart of the sieve filter we use is shown in Fig. 6. First, the input image is inverted and then is eroded by a frame-like structuring element. Then the eroded image is subtracted from the input image to remove small noisy pixels. If noisy pixels still remain, the size of the frame-like structuring is reduced or increased and the same process repeats until all the noise is removed, *i.e.*, the image is not changed.

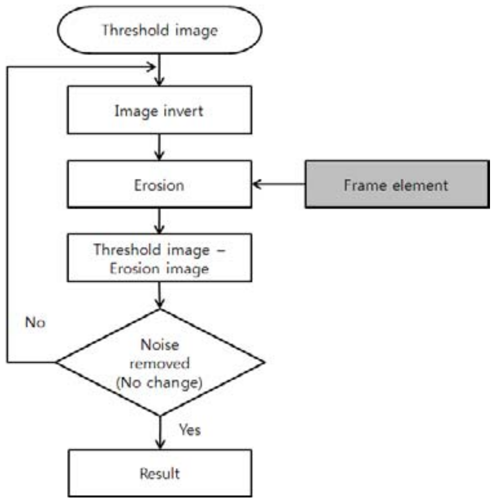


Fig. 6 Flowchart of sieve filter

The GHMT algorithm we propose can be used in variety of image segmentation problems depending on the choice of the nonlinear functions to use for the functions *Hval* and *Mval*. For a problem like the segmentation of relatively dark characters in this paper, *Hval* = $avgL_k$ and *Mval* = $\max_k$ is found to be the best choice. However, readers can easily find a proper choice for the specific problems they encounter. The choice we made is determined by the extensive simulated experiments described in the following section. Table 1 shows the algorithm performance result, in terms of the underkill and the overkill rates, depending on the choice of the nonlinear function combination. We simply choose the one that minimizes both the underkill and the overkill rates (Eq. 9). The algorithm with this function selection performs better than the conventional method,[17-19] which conceptually uses the functions $\min_k$ and $\max_k$ for *Hval* and *Mval*, respectively. The change of function selection and the successful performance enhancement is the key contribution of this paper.

The underkill and overkill rates are defined as follows:

$$\begin{aligned} Underkill\ rate &= \frac{C-D}{C}\times 100(\%) \\ Overkill\ rate &= \frac{E}{T-C}\times 100(\%) \end{aligned} \tag{9}$$

where C is the number of pixels in character pattern, D is the number of detected character pixels, E is the number of the improperly detected pixels, and T is the total number of pixels in image.

Table 1 Function evaluation for *Mval* and *Hval*

Hval : max_k *Mval* : max_k Underkill : 54% Overkill : 26%	*Hval* : max_k *Mval* : $avgS_k$ Underkill : 54% Overkill : 26%	*Hval* : max_k *Mval* : min_k Underkill : 94% Overkill : 1%
Hval : min_k *Mval* : max_k Underkill : 9% Overkill : 20%	*Hval* : min_k *Mval* : $avgS_k$ Underkill : 42% Overkill : 25%	*Hval* : min_k *Mval* : min_k Underkill : 88% Overkill : 2%
Hval : $avgL_k$ *Mval* : max_k Underkill : 2% Overkill : 0.1%	*Hval* : $avgL_k$ *Mval* : min_k Underkill : 67% Overkill : 5%	*Hval* : $avgL_k$ *Mval* : $avgS_k$ Underkill : 12% Overkill : 27%

3. Results and Analysis from Experiments

The performance of the algorithm varies depending on several parameters. The parameters include the shape and type of characters, the complexity of image background, illumination variations, and the signal-to-noise rate (SNR), etc. The set of parameters are varied as follows and the underkill rates are measured to show the performance of the proposed algorithm:

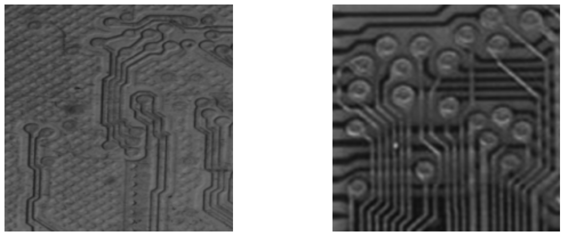
(a) Three characters, ‘A’, ‘C’, and ‘X’, are used in the experiments.
(b) Two realistic complex backgrounds as shown in Fig. 7 are used.
(c) Illumination irregularities such as illumination gradation from left to right in Fig. 8(a), and center to edge in Fig. 8(b) are used.
(d) A Gaussian noise with zero mean and the standard deviation σ is added to the images.

SNR is defined as, $\mathrm{SNR} = 20\log(S/\sigma)$, is varied from 10 to 30. S represents the contrast between the character and the background, and is varied from 10 to 60. The width of character, w, is 23.

We perform 30 experiments for each added Gaussian noise to be statistically correct. The mean values are denoted as symbols and the standard deviations are denoted as the vertical bar to illustrate the underkill rates in each figure.

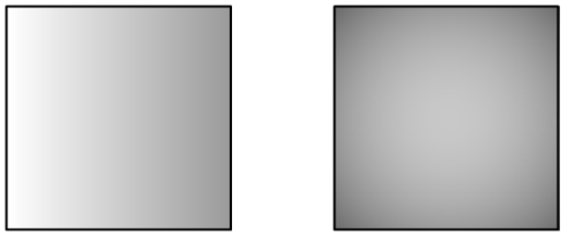
In the first experiment, characters such as ‘A’, ‘C’, and ‘X’ are tested on both IC package and flexible PCB to measure the performance w.r.t. the shape of characters. The parameters used in this experiment are shown as follows and the result is shown in Fig. 9.

① Character shape	‘A’, ‘C’, ‘X’
② Strength of character signal	60
③ Illumination	Uniform
④ Gaussian noise (SNR)	10 ~ 30



(a) IC package (b) flexible PCB

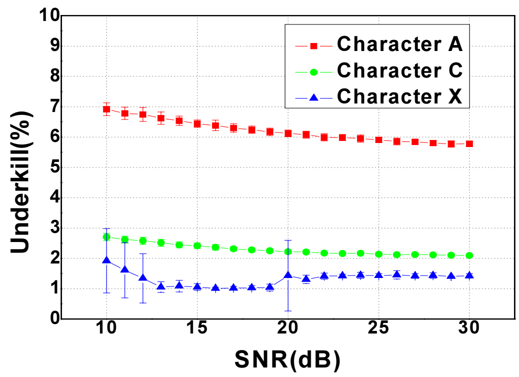
Fig. 7 Realistic background images used in experiments



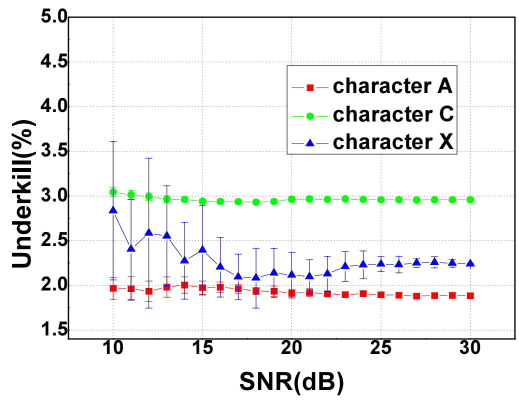
(a) from left to right (b) center to edge

Fig. 8 Image used for illumination irregularity

As shown in Fig. 9, the proposed algorithm shows successful performance. Regardless of the character shape and noise the accuracy scores more than 96% in flexible PCB case and scores 93% in IC package case.



(a) character on IC package



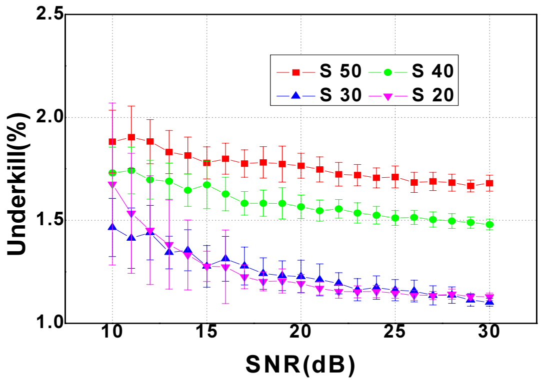
(b) character on flexible PCB

Fig. 9 Underkill rate for each character segmentation

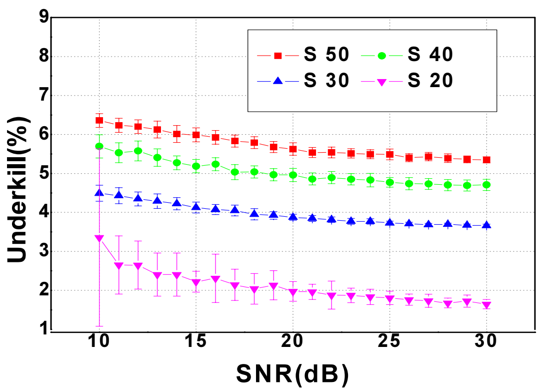
In the second experiments, the strength of character signal S is varied from 20 to 50 to see the performance w.r.t. the contrast between image and background change. The parameters used in these experiments are shown as follows and the result is shown in Fig. 10.

① Character shape	‘A’
② Strength of character signal	20 ~ 50
③ Illumination	Uniform
④ Gaussian noise (SNR)	10 ~ 30

As shown in Fig. 10, more than 98% (IC package) and 93% (flexible PCB) of the character pixels are detected.



(a) character on IC package



(b) character on flexible PCB

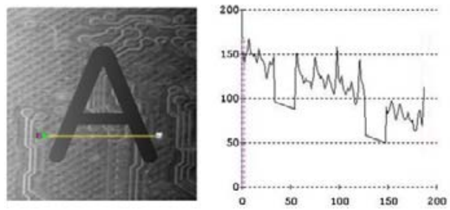
Fig. 10 Underkill rate of character A

In the third experiments, we tested the performance w.r.t. the illumination irregularity with the realistic background. The parameters used in this experiment are shown as follows:

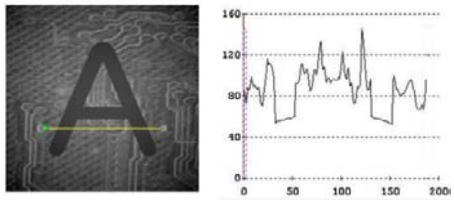
① Character shape	‘A’
② Strength of character signal	60
③ Gaussian noise (SNR)	10 ~ 30
④ Background	IC Package

Sample images used in this experiment are shown in Fig. 11 together with the gray-scale line profile to show the illumination variations. In Fig. 11, the horizontal axis of (a) means the illumination difference from left to right, and the horizontal axis of (b) is the slope of illumination variation from center to edge.

As shown Fig. 12 (a) and (b), on the IC packages, the underkill rates are less than 5.5%. In Fig. 12(b), δH represents the illumination difference from center to edge.

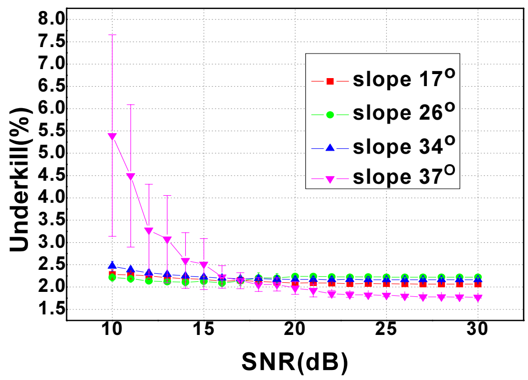


(a) left to right illumination variation

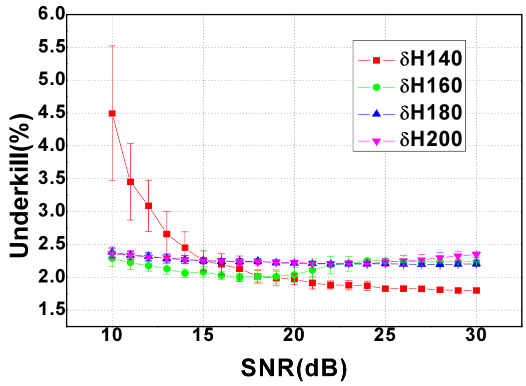


(b) center to edge illumination variation

Fig. 11 Line profiles of character images on IC package



(a) slope irregularity



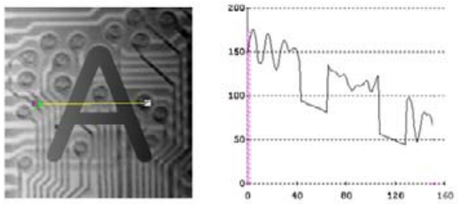
(b) center irregularity

Fig. 12 Character segmentations with illumination irregularities for IC package

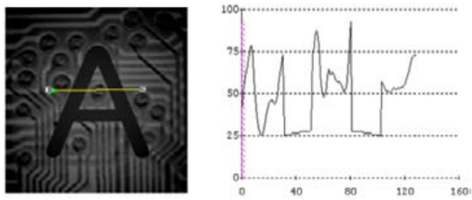
Same experiment for the characters on flexible PCB is performed with the following parameter setting:

① Character shape	‘A’
② Strength of character signal	60
③ Gaussian noise (SNR)	10 ~ 30
④ Background	flexible PCB

Sample images used in this experiment are shown in Fig. 13 together with the gray-scale line profile to show the illumination variations.



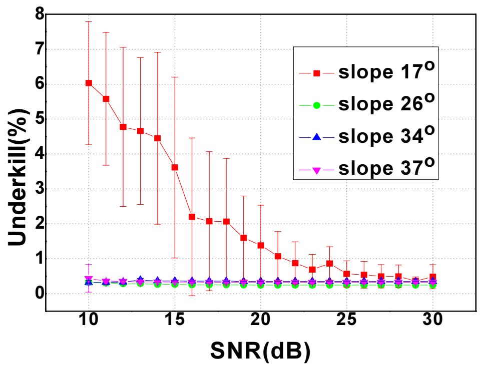
(a) left to right illumination variation



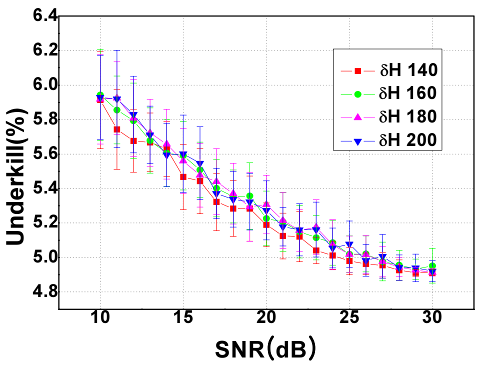
(b) center to edge illumination variation

Fig. 13 Line profiles of character images on flexible PCB

As shown in the Fig. 14(a) and (b), the underkill rates are less than 6.5% when SNR increases from 10dB.



(a) slope irregularity



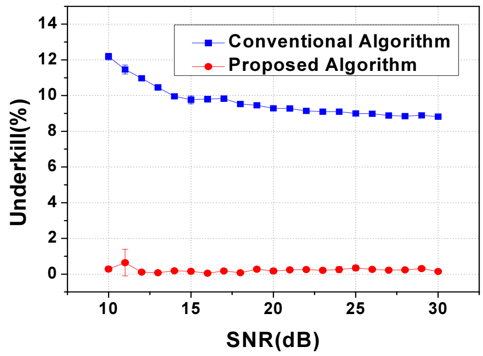
(b) center irregularity

Fig. 14 Character segmentations with illumination irregularities for flexible PCB

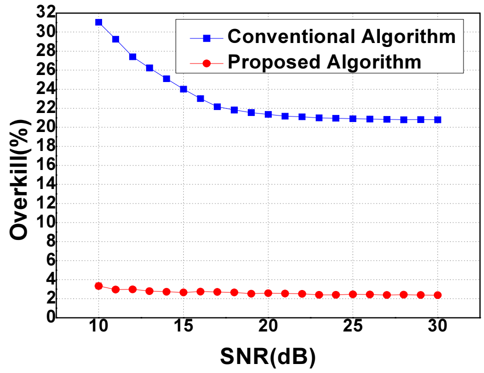
3.1 Comparison with conventional algorithm

We compared the proposed algorithm with the conventional GHMT algorithm[18] to verify the efficiency of the proposed GHMT as Fig. 15.

① Character shape	'A'
② Strength of character signal	60
③ Gaussian noise (SNR)	10 ~ 30
④ Background	IC Package
⑤ Illumination	Uniform



(a) comparison of underkill rate with conventional algorithm



(b) comparison of overkill rate with conventional algorithm

Fig. 15 Comparison with conventional algorithm

As shown in Fig. 15, the proposed GHMT algorithm shows underkill rate lower than 1%, however, the conventional GHMT shows more than 9%. Moreover, when compared in the overkill rate, the proposed GHMT shows 2% or so, and the conventional GHMT shows 21% at the higher SNR.

These performance enhancements are caused from the flexibility of the kernel function selection from min_k, max_k, $avgS_k$ and $avgL_k$, as mentioned before.

4. CONCLUSIONS

A character segmentation algorithm based on GHMT is proposed. A set of new structuring elements used in the transform is developed to segment characters together with a set of nonlinear functions to compute the morphological operations.

Algorithms are proposed for various character extractions along the functions using *Hval* and *Mval.* Simulations are devised with the almost realistic backgrounds, IC package and flexible PCB, and the underkill rates are shown with varying SNR. The characters with significant contrast difference from background are segmented precisely in spite of their shapes. Even from the images that have illumination irregularity, the characters are segmented successfully. However, if the signal difference between the background and character is very small, the segmentation is not effective as expected. From the analysis such as Table 1 and Fig. 15, we conclude that the performance of the proposed GHMT is better than that of the conventional GHMT.

In the future work, automatic function selection algorithm can be developed for character segmentation under complex backgrounds.

ACKNOWLEDGEMENTS

This research was financially supported by the Ministry of Education, Science Technology (MEST) and Korea Institute for Advancement of Technology (KIAT) through the Human Resource Training Project for Regional Innovation.

REFERENCES

1. Glasbey, C. A., "An analysis of histogram-based thresholding algorithms," Graphical Models and Image Processing, Vol. 55, No. 6, pp. 532-537, 1993.
2. O'Gorman, L., "Binarization and multi-thresholding of document images using connectivity," Comp. Vision, Graphics and Image Processing, Vol. 56, No. 6, pp. 494-506, 1994.
3. Otsu, N., "A threshold selection method from gray-level histograms," IEEE Trans. on Systems, Man, and Cybernetics, Vol. 9, No. 1, pp. 62-66, 1979.
4. Kittler, J. and Illingworth, J., "Minimum error thresholding," IEEE Trans. on Pattern Recognition, Vol. 19, No. 1, pp. 41-47, 1986.
5. Sauvola, J., Seppanen, T., Haapakoski, S. and Pietikainen, M., "Adaptive document binarization," Int. Conf. on Document Analysis and Recognition, pp. 147-152, 1997.
6. Trier, O. D. and Jain, A. K., "Goal-directed evaluation of binarization methods," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 17, No. 12, pp. 1191-1201, 1995.
7. Wu, V., Manmatha, R. and Riseman, E. M., "Finding text in images," Proc. ACM Int. Conf. Digital Libraries, pp. 3-12, 1997.
8. Jain, A. K. and Yu, B., "Automatic text localization in images and video frames," IEEE Trans. on Pattern Recognition, Vol. 31, No. 12, pp. 2055- 2076, 1998.
9. Jain, A. K. and Zhong, Y., "Page segmentation using texture analysis," IEEE Trans. on Pattern Recognition, Vol. 29, No. 5, pp. 743-770, 1996.
10. Wahl, F. M., Wong, K. Y. and Casey, R. G., "Block segmentation and text extraction in mixed text/image documents," Computer Graphics and Image Processing, Vol. 20, No. 4, pp. 375-390, 1982.
11. Wong, K. Y., Casey, R. G. and Wahl, F. M., "Document analysis system," IBM Journal of Research and Development, Vol. 26, No. 6, pp. 647-656, 1982.
12. Wang, D. and Srihari, S. N., "Classification of newspaper image blocks using texture analysis," Computer Vision, Graphics and Image Processing, Vol. 47, No. 3, pp. 327-352, 1989.
13. Fletcher, L. A. and Kasturi, R., "A robust algorithm for text string separation from mixed text/graphics images," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 9, No. 1, pp. 149-153, 1987.
14. Jain, A. K. and Bhattacharjee, S., "Address block location on envelopes using gabor filter," IEEE Trans. on Pattern recognition, Vol. 25, No. 12, pp. 1459-1477, 1992.
15. Jain, A. K. and Bhattacharjee, S., "Text segmentation using gabor filters for automatic document processing," Machine Vision and Applications, Vol. 5, No. 3, pp. 169-184, 1992.
16. Etemad, K., Doermann, D. and Chellapa, R., "Multiscale segmentation of unstructured document pages using soft decision integration," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 19, No. 1, pp. 92-96, 1997.
17 Ye, X., Cheriet, M. and Suen, C. Y., "Model-based character extraction from complex backgrounds," Int. Conf. on Document Analysis and Recognition, pp. 511-514, 1999.
18. Khosravi, M. and Schafer, R. W., "Template matching based on a grayscale hit-or-miss transform," IEEE Trans. on Image Processing, Vol. 5, No. 6, pp. 1060-1066, 1996.
19. Ye, Q., Gao, W. and Huang, Q., "Automatic text segmentation from complex background," Int. Conf. on Image Processing, Vol. 5, pp. 2905-2908, 2004.
20. Cavalcanti, G. D. C., Silva, E. F. A. and Zanchettin, C., "A heuristic binarization algorithm for documents with complex background," Int. Conf. on Image Processing, pp. 389-392, 2006.
21. Pan, W. M., Bui, T. D. and Suen, C. Y., "Text segmentation from complex background using sparse representation," Ninth Int. Conf. on Document Analysis and Recognition, pp. 412-416, 2007.
22. Jin, N. and Tang, Y. Y., "Text area localization under complex background using wavelet decomposition," Proc. of Sixth Int. Conf. on Document Analysis and Recog., pp. 1126-1130, 2001.
23. Raducanu, B. and Grana, M., "A grayscale hit-or-miss transform based on level sets," International Conference on Image Processing, Vol. 2, pp. 931-933, 2000.
24. Naegel, B., Passat, N. and Ronse, C., "Grey-level hit-or-miss transforms-part II: Application to angiographic image processing," IEEE Trans. on Pattern Recognition, Vol. 40, No. 2, pp. 648-658, 2007.
25. Kang, D.-J. and Lee, W.-H., "Automatic circle pattern extraction and camera calibration using fast adaptive binarization and plane homography," Int. J. Precis. Eng. Manuf., Vol. 11, No. 1, pp. 13-21, 2010.