ACTA·GRAPHICA

ISSN: 0353-4707
e-ISSN: 1848-3828

ZNANSTVENI ČASOPIS ZA TISKARSTVO I GRAFIČKE KOMUNIKACIJE
JOURNAL FOR PRINTING SCIENCE AND GRAPHIC COMMUNICATIONS

SCIENTIFIC PAPER

Evaluation of Scumming Printing Defect by Using Computer Vision-based Bit Plane Slicing Method

Jayeeta Saha, Shilpi Naskar, Arpitam Chatterjee, Kanai Chandra Paul

Department of Printing Engineering, Jadavpur University, Saltlake Campus, Kolkata-700106, India

Correspondence: jayeetas.printing.rs@jadavpuruniversity.in

ABSTRACT:

In offset lithography scumming print defect occurs when non-image areas of the image carrier accepts ink and transfer it to the blanket. The quality of printed image is compromised with the presence of scumming pixels. Visual impact in terms of readability affected due to this unwanted print defect. Moreover, archival process through scanning and OCR can be interfered by the presence of scumming. Various press conditions are responsible for this phenomenon. Generally, this kind of printing problem is detected manually and preventive measure is taken according to the factor responsible for the occurrence of this defect. Traditionally, detection of this defect relies heavily on manual inspection, which is prone to inconsistency and human error. In response to this challenge, the present study proposes an automated, low-cost computer vision-based approach for detecting scumming defects using bit-plane slicing and discrete cosine transform (DCT). In this present study a computer vision-based approach has been proposed to evaluate this kind of print defects and its percentage of occurrence over the existing detection procedure. As the density difference of scum pixels and print pixels are not that much differentiable, it becomes a challenging task to identify the scum pixels from print pixels in the present computer vision-based study. Intensity of pixels can also be varied in different kind of images which makes the segregation of scum pixels even more difficult. An automated algorithm selects the bit plane containing the most scum-relevant information, followed by DCT filtering and two-stage segmentation to isolate print and scum pixels. To segment the scum pixels from the print pixels, an adaptive thresholding method with a proposed range of intensity values have been applied to high key, low key and mid key images. The proposed method demonstrated high agreement with manual visual assessment, achieving consistent results across high-key, mid-key, and low-key images. Quantitative scumming percentages are calculated, offering a reproducible metric for defect severity. The result shows the efficiency of the proposed algorithm as compared to present subjective method.

KEYWORDS:

Scumming Print Defect, Bit-Plane Slicing, DCT, Image Processing, Computer Vision

1. INTRODUCTION

Scumming print defect mostly arises when non image regions of a printing plate become ink receptive. In offset lithography printing, image to be printed is at first transferred to a rubber blanket and the blanket then transfers the image to the printed surface from image carrier [1]. In Figure 1. a sample image with its printed area and non-image area is shown. The non-image area in good copies is supposed to be blank but here ink transfers into this area and as a result scumming print defect occurs. Scumming can happen on any color unit due to various press conditions during the production. One of them is, improper ink water balance. Generally, the ink has lower viscosity at 30-40oC operating temperature of a web press [1]. If the temperature increases, due to mechanical shear, viscosity of ink decreased and it may then transfer to the non-image areas [1]. Moreover, dampening levels can cause over emulsification of the ink which is also responsible for scumming print defect [2]. Also, if the dampening rollers are not cleaned properly and if it contains dried ink, as a result plate wear can happen which may cause the scumming. Therefore, to avoid this kind of defect, dampening rollers must be cleaned and replaced. Conductivity of the fountain solution must be checked to avoid scumming. Density of halftone areas increased in scumming which effects in tonal rendition [1] and loss of halftones shadow detail. Poor plate development, incorrect plate exposure and insufficient acid in ink and fountain solution are also responsible for scumming [1]. If ink is too soft and not tacky enough it may transfer over the top of the fount film in the non-image portion. The only remedy is to increase the stiffness of ink film by adding gum or water-resistant varnish. Desensitizing film on the non-image areas of the plate can be destroyed if ink vehicle contains too much fatty acid which results scumming print defect [1]. The pH value of the fountain solution can also be responsible for the occurrence of scumming in non-image area. If the pH is high (pH>7) in the fountain solution it may cause emulsification between ink and water which results as scumming [1] [3]. Moreover, scumming can be prevented if the surface tension of the fount solution is reduced to the similar value to the surface tension of the ink, that is approximately 37 X 10-3 Nm-1 [1]. This kind of printing defect cannot be accepted in any way and should be removed to avoid any interference in archival process through scanning and OCR. In printing industry, it is most important to identify and optimize this kind of print defects for the sake of better quality of production. Generally, in high speed offset presses for newspaper and commercial printing, the detection of scumming printing defect is done manually during press run and for that reason the results can vary man to man's ability of identification of scumming print defect as well as quantification of scumming print defect. In recent years, several automated solutions have been integrated into industrial printing workflows to monitor and detect printing defects in real-time for industrial packaging, label and premium commercial printing. Inline inspection systems such as Heidelberg's "Zero Defect Production" utilize high-speed cameras and AI algorithms for defect detection and process correction during production [4]. Similarly, proofing systems like Gradient's Pixel Proof offer high-resolution image comparison and quality control in packaging workflows [5]. However, these systems are often proprietary, expensive, and tailored for packaging and other premium commercial prints not for general commercial, high speed newspaper printing presses. In contrast, this present study introduces a cost-effective, flexible computer vision-based approach for scumming defect detection, which can be utilized for both high speed production diagnostics and archival validation.

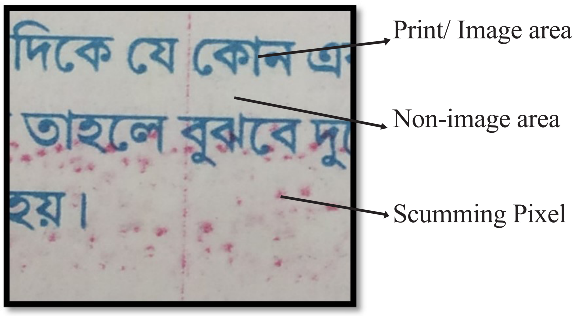

Figure 1 *Sample image showing image, non-image area and scumming pixel*

In the domain of detection of printing defect includes defects detection by using local high order correlation method by [6], chromatism and SSIM (Structural Similarity Measurement) algorithm used to detect print defect on metal container by [7] , pharmaceutical tablet package print defect was identified [8], PCB print defect identification [9], Self-Organizing Map (SOM) and Independent Component Analysis (ICA) were implemented to separate the showthrough print defect from a scanned document [10]. Print scum identification using DCT based computer vision method [11], Extraction of lungs region using bit plane slicing technique [12], A simplified homocentric square filter is implemented based on computer vision to detect the defect in printing plate [13], Breast cancer image classification based on CNN and bit plane slicing [14].

2. PRESENTED METHOD

The proposed method of scumming print defect identification has several steps and is tested with 300 printed sample collected from different offset printing presses encompassing diverse tonal ranges and defect severities. MATLAB programming software is used for image processing and for other computer vision-based application. Here the method with illustration of each step is described below and in Figure 2 each and every step is depicted as flow chart.

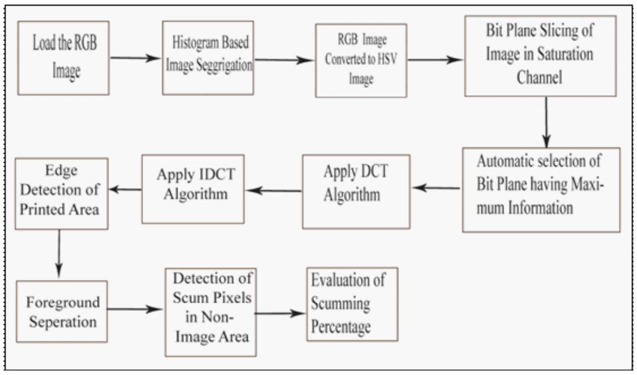
Load the RGB Image
Histogram Based Image Seggrigation
RGB Image Converted to HSV Image
Bit Plane Slicing of Image in Saturation Channel
Automatic selection of Bit Plane having Maxi-mum Information
Apply DCT Algorithm
Apply IDCT Algorithm
Edge Detection of Printed Area
Foreground Seperation
Detection of Scum Pixels in Non-Image Area
Evaluation of Scumming Percentage

Figure 2 *Flow Chart of Presented method of Scumming Printing Defect Detection*

2.1 Image Segregation

Digital copy of all samples collected from different offset printing presses are captured by using an imaging device. Mobile camera with 48-megapixel resolutions in D65 illumination, using a diffused LED panel, used for the digitalization of all samples. The imaging setup involved a fixed camera-to-sample distance of 26.5 cm to ensure consistent resolution and focus across all samples The camera setup for this experiment is shown in Figure 3. Then the samples are divided into three groups which are high-key image, mid-key image and low-key image. Here sample 1, sample 2 and sample 3 is considered as high-key, mid key and low-key image shown Figure 4, Figure 5 and Figure 6 respectively. This separation of image is done based on the histogram of images. After the segregation the RGB image is converted to HSV color space through Matlab programming Software. Unlike RGB, HSV is channel independent color space, for which the result does not vary from device to device. Moreover, colors are interpreted more perceptually in HSV rather than RGB and for performing image processing algorithms like image segmentation, image smoothening, sharpening, saturation and value channels information are more helpful [15].

In the HSV model, hue is represented in a circular coordinate system and can be expressed as an angle. The hue value is typically normalized on a scale from 0 to 255. Generally, hue indicates purity of a color like pure red or blue whereas saturation defines the intensity or vividness of a color in relation to its brightness and it also ranges from 0 to 255. The value component represents the color sensation that can be measured by the gray level of image [15]. When pixel value is at its max or min intensity level, hue and saturation do not make any difference. So, for further operations as the saturation channel is more informative than hue channel and value channel, the whole work have been proceeded with saturation channel.

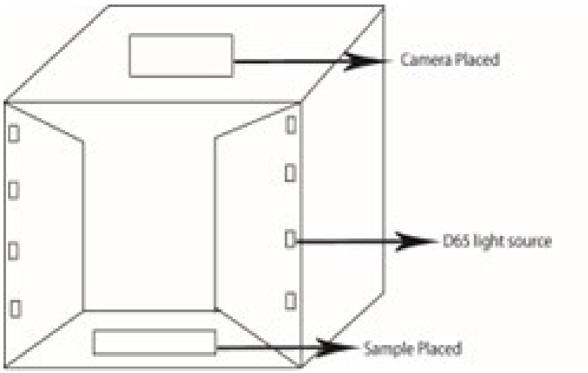

Figure 3 *Schematic diagram of the Camera set-up*

2.2 Automated Bit Plane Selection

After selecting saturation channel, it has been sliced into bit planes using bit-plane slicing technique. This step is necessary to collect more information or features bit by bit. For an 8-bit image, it is sliced from zero-bit plane to seven-bit planes. Here the zero-bit plane holds the information about least significant bit whereas the seven-bit plane carries the information of most significant bit. As in scumming print defect the scum pixels can be in foreground as well as in background also. So, it is necessary to segment the image into bit planes to extract the raw information of each plane. Generally, it can be assumed that the lower bit plane consists of foreground pixels mostly whereas the upper bit plane is expected with background pixel. Now an automated approach has been done to select the bit plane which carries the most information of scumming pixels than other bit planes. This automated bit plane selection approach has been done based on the axis's properties of image histogram. Where y axis of the bit plane contains most of the higher intensity pixels, that plane is chosen for further operations. Figure 4. shows the different bit-planes of a sample image. For sample 1(high-key) the 6-bit plane contains most of the scum pixel whereas for sample 2(mid-key) and sample 3(low-key) 7- bit plane has the most scumming pixels. Therefore, for detection of scum pixels the right bit plane is necessary to choose and further operations are performed into that bit plane only.

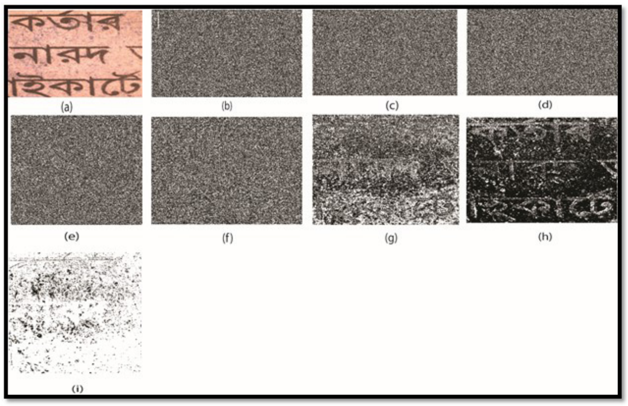

Figure 4 *The eight binary images after applying bit- plane slicing technique (a) Sample Image; (b) bit plane zero image; (c) bit plane one image; (d) bit plane two image; (e) bit plane three image; (f) bit plane four image; (g) bit plane five image; (h) bit plane six image; (i) bit plane seven image*

After the bit plane selection discrete cosine transform (DCT) algorithm [16], [17] has been applied to discard the unwanted frequency retaining the required ones. Typically, the low-frequency components located near the origin exhibit similar density characteristics and are preserved as part of the image or text content. Frequency component reduction is needed to identify the pixels of text or image discarding unwanted pixels. After discrete cosine transform inverse DCT is applied to get the reconstructed image. The resulted image contains only the print pixel which is text or image discarding other scumming pixels. 2D-DCT can be expressed as [11], [15]

$$\mathrm{F(u,v)}=\left(\frac{2}{N}\right)^{\frac{1}{2}}\left(\frac{2}{M}\right)^{\frac{1}{2}}\sum_{i=0}^{N\text{-}1}\sum_{j=0}^{M\text{-}1}\mathrm{a(i)a(j)}\cos\left[\frac{(2i+1)u\mathrm{p}}{2N}\right]\cos\left[\frac{(2j+1)v\mathrm{p}}{2M}\right]f(\mathrm{i,j}) \quad 1)$$

where,
The input image is N by M.
f (i, j) = intensity of the pixel in row i and column j.
F(u, v) = DCT coefficient of row and column of the DCT matrix.

2.4 Image Segmentation

After discrete cosine transform the contour of print pixel is identified by applying Prewitt operator. Here two steps of segmentation have been done. First to segment the print pixel from non-image area and second is to segment the scum pixels from the non-image area. In the first segmentation an adaptive thresholding algorithm is applied to segment the image into foreground and background. It is expected that the foreground image contains the print pixel and background image contains the scum pixels. For the second segmentation according to the type of images (low-key, high-key and mid-key image) a range of intensity value is proposed for different images to identify the scum pixel. To finalize the range of intensities for different type of images several samples of same kind are tested. For high-key images the intensity value of the pixels within the range of 0.65-0.85 is identified as foreground pixel or print pixel. And the pixels having intensity values less than 0.65 is identified as background or scum pixels. Whereas for mid-key and low-key images this range is 0.45-0.75. And below 0.45 pixels are identified background or scum pixels. These ranges of intensity values are considered when the second level of segmentation has been done to segregate the scum pixels from non-image area.

2.5 Percentage of Scumming

After the identification of scum pixels, the percentage of scumming is calculated. First the total no of pixel of the original image is calculated. Then from the scum pixel detected image the total no of scum pixels is calculated. The percentage of scum pixel is calculated against total no of print pixel.

3. RESULTS AND DISCUSSION

The proposed method is tested with 300 samples collected from different offset printing presses and the digital form of these samples are taken by a mobile camera only. The pictorial results for the three samples including high-key, mid-key and low-key image are shown in Figure 5, Figure 6 and Figure 7 respectively. Figure 8 and Figure 9 are the pictorial results of comparatively good copies excluding scumming pixels.

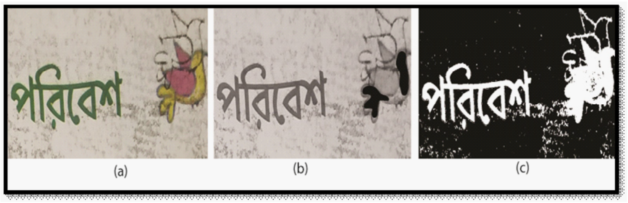

Figure 5 *(a) Original Image Sample-1 (High-Key) (b) Foreground Image (c) Detected Scum Pixel in background*

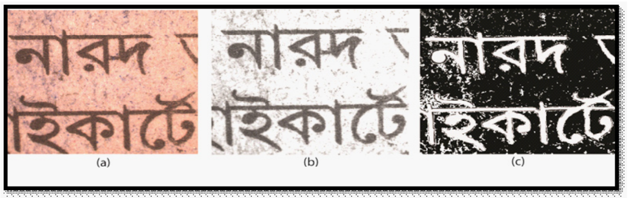

Figure 6 *(a) Original Image Sample-2 (Mid-Key) (b) Foreground Image (c) Detected Scum Pixel in background*

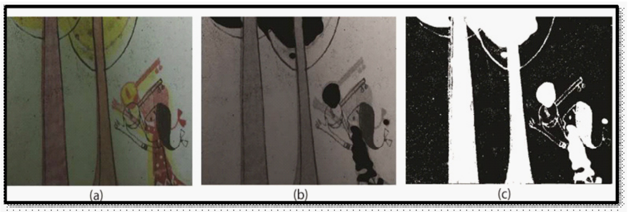

Figure 7 *(a) Original Image Sample-3 (Low-Key) (b) Foreground Image (c) Detected Scum Pixel in background*

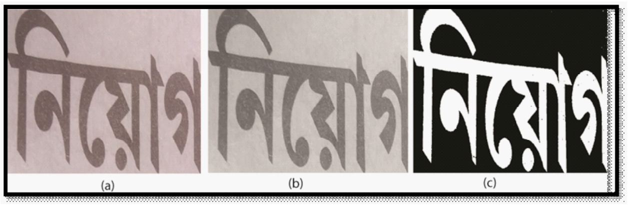

Figure 8 *(a) Original Image Sample-4 (Low-Key) (b) Foreground Image (c) Detected Scum Pixel in background*

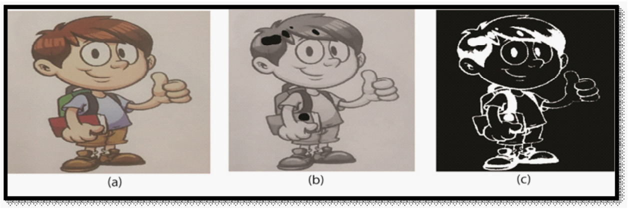

Figure 9 *(a) Original Image Sample-5 (Low-Key) (b) Foreground Image (c) Detected Scum Pixel in background*

In Figure 5a original high-key image of a sample is shown. From the bit plane information, it is identified that among all the seven-bit planes, for this high-key sample image, bit plane-6 contains most of the scum pixels. Therefore, further operations are done into this bit plane only for this sample image to identify the scum pixels retaining the print pixels. After the first segmentation, the foreground image with image area is segmented that is shown in Figure 5b. Now for the second segmentation in non-image area a range of intensity values are considered. For high-key image this intensity range is 0.65-0.85. After the second level of segmentation the scum pixels in white in non-image area is identified. The contour of the printed text and images with scum pixels in background is shown in Figure 5c. Percentage of scumming pixel occurrence is calculated and shown in table 1. Here for high-key image in Figure 4 the scum percentage is 16.

In Figure 6a another sample of mid-key image is shown. This sample mainly contains text-based image. For this sample bit plane-7 contains the most scumming pixels. Therefore, further operations are performed into that bit plane only. Figure 6b shows the printed text area segmented as foreground of the original image. Figure 5c shows the scumming pixels in white in background. For this mid-key sample, the intensity range is kept 0.45-0.75 to perform the second level of segmentation. The scum percentage of this sample is 84.

In Figure 7a the original image of low-key sample is shown. After performing bit plane slicing technique, it is noticed that the bit plane -7 contains the scumming pixels mostly. Figure 7b shows the printed image area in foreground. Like mid key image here also same the intensity range for second level segmentation is considered. Figure 7c shows the scum pixels after segmentation. The percentage of scum pixels measured of this sample is 32.

In Figure 8a and Figure 9a the original images have comparatively less scum and accepted as good printed copy. The percentage of scum present in these samples is 0.04 and 0.05 respectively which proves the effectiveness of the proposed algorithm. Here in Figure 8c and 9c there is no scum pixels identified, only contour of printed area is shown.

White pixels (e.g., in Figure 6c) indicate the presence of scum pixels in the non-image area. In some cases, residual dark pixels within the text (e.g., in Figure. 8c) are preserved as part of the printed area and may visually resemble other defects such as mottling, but they are not identified as scum by this algorithm. The human eye perceives visual noise or defects based on contrast, distribution, and location of artifacts, not just quantity. A few well-placed scum spots near text may look more severe than a larger number scattered in the periphery. For example, In Figure 5 (high-key), scum pixels appear more distinct due to the bright background. In Figure 7 (low-key), the dark background may mask or blend some scum pixels to the human eye, even though their pixel count is higher. Moreover, in Figure7 the intensity of scum pixels is very low to distinguish but it contains a larger number of small scum pixels across a broader area, while Figure 5 have fewer but more concentrated scum pixels. Therefore, the visual appearance of scum pixels in Figure5 is more but actually it is less than Figure7. This proposed algorithm detects pixel-wise occurrences of scum pixels without weighing shape or distribution or concentration, which generally affects visual judgment.

Figure

Table 1 *Scumming Percentage of Test Images*

Test Image	Scumming Percentage (%)
Sample 1(Figure 5)	16
Sample 2(Figure 6)	84
Sample 3(Figure 7)	32
Sample 4(Figure 8)	0.04
Sample 5(Figure 9)	0.05

4. CONCLUSION

In the present study an automatic bit plane selection approach has been proposed to identify the scum pixels from printed pixels. The presented result shows that the method can effectively identify and segregate the scum pixels in the sample test images. Moreover, percentage of scumming occurrence is calculated in this study. Artificial intelligence application to identify this kind of print defect can be the future scope of this study. However, this automated bit plane selection approach made the detection procedure more moderate. Pictorial representation of the results supports this fact also. Moreover, it becomes a popular research direction to identify this kind of print defect with the help of image processing algorithm. The application of image processing technique makes the detection procedure less manual, less time consuming and cost effective as well. A key direction for future research involves implementing a mobile based approach of detection and removal of scum pixels while reconstruction of image is taking place for any kind of archival process of printed document.

LITERATURE

[1] Leach R, Pierce R. The Printing Ink Manual. Springer Science & Business Media; 2007.

[2] Barnard M. The Print and Production Manual. 11th ed.

[3] Eldered. Chemistry for the Graphic Arts. GATF Press; 2007.

[4]https://www.polygrafia-fotografia.sk/zero-defect-production-new-inline-inspection-system-heidelberg/, 27 apríla, 2017 [Accessed: 23 June 2025].

[5] https://www.gradient.de/en/products/proof-solutions/pixel-proof-workflow[Accessed:23 June 2025].

[6] Tu Y, Chen Q, Liao L, Deng W. Research on print quality assessment and identification: Evaluation of print edge roughness. 2009 International Conference on Computational Intelligence and Software Engineering; 2009. doi:10.1109/cise.2009.5364481.

[7] Zhou M, Wang G, Wang J, Hui C, Yang W. Defect detection of printing images on cans based on SSIM and chromatism. 2017 3rd IEEE International Conference on Computer and Communications (ICCC); 2017. doi:10.1109/compcomm.2017.8322912.

[8] Karthik D, KV, Ar A. Printing defect identification in pharmaceutical blisters using image processing. Asian Journal of Pharmaceutical and Clinical Research. 2018;11(3):210. doi:10.22159/ajpcr.2018.v11i3.23407.

[9] Indera Putera SH, Ibrahim Z. Printed circuit board defect detection using mathematical morphology and MATLAB image processing tools. 2010 2nd International Conference on Education Technology and Computer; 2010. doi:10.1109/icetc.2010.5530052.

[10] Zhang X, Lu J, Yahagi T. Blind separation methods for image show-through problem. 2007 IEEE International Symposium on IT in Medicine and Education; 2007. doi:10.1109/ITAB.2007.4407395.

[11] Saha J, Naskar S, Chatterjee A, Paul KC. Print scum identification using DCT-based computer vision method. 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN); 2018. doi:10.1109/icrcicn.2018.8718734.

[12] Kiran T, Kwon GR. An advanced segmentation using bit-plane slicing technique in extraction of lungs region. 2011 Second Asian Himalayas International Conference on Internet (AH-ICI); 2011. doi:10.1109/ahici.2011.6113949.

[13] Zhang WJ, Xu F, Li XY, Xiao H, Peng SH, Nam HD. Automatic printing plate defect detection based on a simplified homocentric square filter. 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI); 2016. doi:10.1109/cisp-bmei.2016.7852732.

[14] Chen G, Chen Y, Yuan Z, Lu X, Zhu X, Li W. Breast cancer image classification based on CNN and bit-plane slicing. 2019 International Conference on Medical Imaging Physics and Engineering (ICMIPE); 2019. doi:10.1109/icmipe47306.2019.9098216.

[15] Gonzalez RC, Woods RE. Digital Image Processing. New York, NY: Pearson; 2018.

[16] Abu NA, Ernawan F. A novel psychovisual threshold on large DCT for image compression. The Scientific World Journal. 2015;2015:1–11. doi:10.1155/2015/821497.

[17] Pang CY, Zhou RG, Hu BQ, Hu W, El-Rafei A. Signal and image compression using quantum discrete cosine transform. Information Sciences. 2019;473:121–141. doi:10.1016/j.ins.2018.08.067.