

# Fabric defect detection using Fourier Transform and Gabor filters

## Abstract

This paper introduces a new method used to solve the problem of texture defects detection and localization. The analysis focuses mainly on the detection of texture materials, which represents texture with a high degree of homogeneity, focusing on the textile's periodicity properties. Although the field research has followed several directions, the developed application uses frequency domain analysis using Fourier transform and Gabor filters in the texture detection process. The processing system has the ultimate goal of detecting portions of the image in which different textures or non-textures are represented. It is considered to be non-texture, any defect in the material, which alters the periodic physical structure of the texture.

**Keywords:** fabric defect detection, fourier transform, analysis methods, autocorrelation

Volume 3 Issue 4 - 2017

Cristina Modrângă,<sup>1</sup> Raluca Brad,<sup>2</sup> Remus Brad<sup>1</sup>

<sup>1</sup>Department of Computer Science, Lucian Blaga University of Sibiu, Romania

<sup>2</sup>Department of Industrial Equipments, Lucian Blaga University of Sibiu, Romania

**Correspondence:** Remus Brad, Computer Science Department, Lucian Blaga University of Sibiu, 10, Victoriei Blvd., 550024 Sibiu, Romania, Tel 407 2333 3051, Email [remus.brad@ulbsibiu.ro](mailto:remus.brad@ulbsibiu.ro)

**Received:** October 24, 2017 | **Published:** December 12, 2017

## Introduction

Nowadays, fabric defect detection is mainly operated based on human inspection. This method is a subjective one, depending on a large number of factors that can influence the human observer, such as the intensity of the lights, the fatigue or the experience of the human observer.<sup>1</sup> This is why, in order to reduce the inspection process costs and to increase the products quality, this process needs to be done by computer vision.

The state of the art in research studies followed several directions, with some researchers focusing on the exploration of the device of fabric image acquisition, while other researchers have moved on to develop algorithms for fabric model analysis. The methods used can be classified into two large categories: optical analysis methods and image analysis methods.

Image-based methods can be categorized into three categories: frequency-based analysis methods, space-based analysis methods and combined methods.<sup>2</sup> We mainly focus on the frequency-based analysis methods, which can be classified into two groups: methods using Fourier transform and methods using wavelet transforms.

The Fourier transform was originally presented by Imaoka<sup>3</sup> to process the fabric image and estimate the fabric pattern. It has been reported that the overall accuracy of these methods would be about 80%. Wood<sup>4</sup> used in 1990 the Fourier transform and autocorrelation functions to model the spatial periodicity of the fabric pattern. A series of characteristics of the two-dimensional power spectrum and the autocorrelation pattern were extracted and it has been estimated that the measurement accuracy can reach less than two fiber per inch.

In 1996, Xu<sup>5</sup> used Fast Fourier Transform to calculate the power spectrum of the fabric and used the logarithmic operation to compress the spectrum to obtain the grayscale image spectrum. Further, Xu determined the peak points of the power spectrum on the directions of the periodic structure of the texture and separately extracted the frequencies of these periodic structures, both in the horizontal and the vertical direction. Finally, the images can be rebuilt based on

the filtered power spectrum, which only keeps the peak points in the spectrum.

Sari-Saraf and co.<sup>6</sup> used the Fourier transform for detecting defects in the fabric. The method they presented examines and performs a one-dimensional diagram, which is a mathematical technique used to validate image integrity. The one-dimensional diagram is created by integrating the points in every ring of the two-dimensional spectrum in the frequency range. The rings are concentric, with different radius, and are used to monitor the fabric structure at the fiber level. The most important advantage is that their approach it is less sensitive to the background noise.

An approach to the wavelet transforms is presented in the article,<sup>7</sup> using a Gabor filterbank, obtained by varying the parameters of the Gabor filter, such as: orientation, frequency, phase, wavelength. The filter response resulting from the convolution between the input image and the filter values will contain low energy points for the non-defective image portions and high energy points for the defective portions of the image. On the filtered image, a strong binarization operation is applied, resulting in a binary image where the white-colored pixels (intensity 255) are the defect areas in the original image, and the pixels in black (intensity 0) represent the fault-free areas in the image. The algorithm was tested on textile images showing 16 different defects that are present in fabrics produced in textile plants and has an accuracy of 83.5% on these examples. Another approach described in Anil K Jain<sup>8</sup> Hammouda K,<sup>9</sup> where a bank of Gabor filters was used for texture segmentation, in order to separate different patterns of fabric from an image.

## The proposed processing scheme

The Gabor filter bank was generated by tuning the frequency ( $u$ ) and the orientation of the filters ( $\theta$ ). The parameter  $\theta$  can be varied using two methods: with a 30 degree orientation separation angle<sup>10</sup> or with a 45 degree orientation separation angle. The frequency parameter  $u$  takes the values:  $1\sqrt{2}$ ,  $2\sqrt{2}$ ,  $4\sqrt{2}$ , ...,  $(N/4)\sqrt{2}$ , where  $N$  is the width of the image on which the filter is applied. All Gabor filters in the bank are applied to the input image, resulting in a number

of images equal to the number of filters in the bank. A method of extracting properties is applied to these filtered images, such as: using the magnitude received in response to filtering, applying an image smoothing method, using only the actual component from the filter response, using a nonlinear sigmoid function, etc (Figure 1).

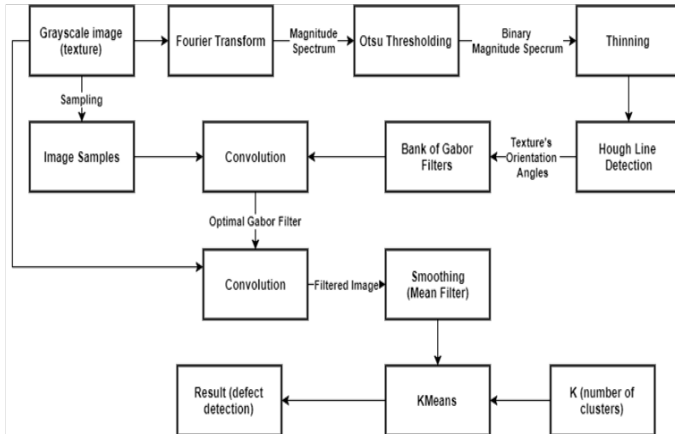


Figure 1 Defect detection processing framework.

### Texture orientation using fourier transform

The Fourier Transform is a mathematical operation that decomposes a signal (any waveform in the real world) into a sum of sinusoidal signals.<sup>11</sup> The Fourier transform decomposes a function or a signal represented in a given representation domain (the time domain, the spatial domain, etc.) into the frequencies it is composed of. In image processing, the input signal is represented in the spatial domain  $(x, y)$ . Let  $N$  be the width and  $M$  be the height of the texture, and  $f(x, y)$  the gray level intensity of the pixel at the position  $(x, y)$ . The Fourier transform of the image is given by equation (1), for frequency variables  $u=0 \dots N-1$  and  $v=0 \dots M-1$ .

The resulting function is a complex function in the frequency domain, which contains the same information as the original function, but in another form of representation, which is easier to analyze in image processing. From the Fourier Transform of the image, we extract the magnitude spectrum, which represents the quantity of each frequency present in the original signal and keeps information about the physical representation of the texture. The magnitude spectrum is calculated based on the equation (2), where  $Fr(u)$  is the real part of the complex Fourier transform result and  $Fi(u)$  is the imaginary part.

$$F(u, v) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x, y) e^{-j2\pi \left( \frac{ux}{N} + \frac{vy}{M} \right)} \quad (1)$$

$$M(u) = |F(u)| = \sqrt{Fr^2(u) + Fi^2(u)} \quad (2)$$

For textured images, the magnitude spectrum represents a small number of components in the frequency domain, and it is periodic in the direction given by the physical structure of the periodicity in the analyzed texture. Forward, we process the peaks from the magnitude spectrum to extract the texture orientation angles.

In order to eliminate the frequencies that have a low rate of apparition and keep only the peaks, we apply an adaptive thresholding operation on the spectrum. We propose Otsu's method to get the optimal threshold value.

### Unsupervised image thresholding

In image processing, Otsu's method is an algorithm used to perform clustering-based image thresholding.<sup>12</sup> The algorithm assumes that the input image contains only two classes of pixels: foreground and background pixels, then calculates the optimal threshold value that best separate the two classes (the intra-class variance is minimal).

Otsu's method finds the threshold value by exhaustively search

for the threshold that minimizes the intra-class variance  $\sigma^2 w(t)$  that represents the weighted sum of the variances of the two classes (equation 3). The weights  $w_0$  and  $w_1$  represents the probabilities of the two classes, separated by the threshold value  $t$  (equation 4 and 5).

$$\sigma_w^2(t) = w_0(t) \sigma_0^2(t) + w_1(t) \sigma_1^2(t) \quad (3)$$

$$w_0(t) = \sum_{i=0}^{t-1} p(i) \quad (4)$$

$$w_1(t) = \sum_{i=t}^N p(i) \quad (5)$$

After the threshold value  $t$  is calculated, we eliminate the frequencies that have a rate of apparition lower than  $t$ , and keep only the peaks from the magnitude spectrum.

The thinning block also operates on the magnitude spectrum, and reduces the neighborhoods of pixels set to value 255 in one pixel. This process is essential to the next block step because a large number of pixels set desired for the visual effect, but they make the line detection process more difficult. Figure 2 shows the amplitude spectrum after the thinning process (left) and before the thinning process (right). From the spectrum, it is noticed that the points removed by the thinning operation do not provide additional information on the texture orientation.

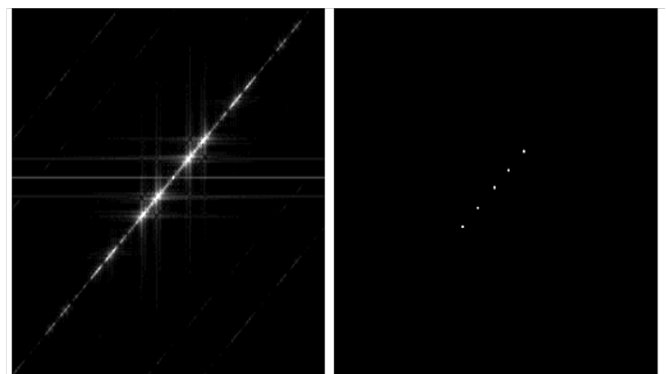


Figure 2 The magnitude spectrum before and after tinning.

### Texture angle estimation by line detection

Line detection in the magnitude spectrum provides information regarding texture orientation. We detect all the lines from the spectrum that pass through the origin point  $(N/2, M/2)$ . Hough line detection detects lines that pass through multiple points in the image, providing information about the line's length and orientations.

Hough transform works with polar coordinates, where the line equation is defined by the equation (6), where  $\rho$  represents the length

of the perpendicular from the origin to the detected line, and  $\theta$  is the angle made by the perpendicular to the Ox axis.

$$\rho = x \cos \theta + y \sin \theta \quad (6)$$

The algorithm operates using a structure called accumulator, which is a two-dimensional array of dimensions equal to the number of possible combinations for the  $\rho$  and  $\theta$  values:  $(-\rho_{\max}, \rho_{\max})$  and  $(-\theta_{\max}, \theta_{\max})$ , initialized to 0.

We iterate the processed magnitude spectrum and for each frequency peak (pixel with the maximum intensity), we increment the position in the accumulator that satisfies the equation (6). After the verification of all the frequency peaks in the magnitude spectrum, the maximum points in the accumulator indicate the presence of some straight lines. High accumulator values indicate that many points in the image have complied with the equation of that line. From the accumulator, we extract the angle indices of a number of maximum points. In this paper, we propose to extract all angle indexes for the values bigger than the maximum value of the accumulator divided by 2. This set of orientation angles will be the input for creating the bank of Gabor filters and represents all the angles for which the texture keeps the periodical structure.

### Generation of optimal oriented gabor filters

In spatial domain, a 2D Gabor filter represents a Gaussian kernel function modulated by a sinusoidal plane wave, and it's one of the most suitable option for texture segmentation and classification and boundary detection.<sup>13</sup> The mathematics of the Gabor filters is presented in the equation (7), where  $x$  and  $y$  represents the pixel coordinates,  $\sigma$  represents the variance of the Gaussian kernel,  $f$  represents the frequency of the texture and  $\theta$  is the orientation parameter.

$$g_e(x, y) = e^{-\frac{1}{2} \frac{x^2 + y^2}{\sigma^2}} \sin(2\pi f(x \cos \theta + y \sin \theta)) \quad (7)$$

A bank of Gabor filters, which represents a set of Gabor filters with different parameters, are applied to a number of samples of the defective texture, and the filter with the best response is taken into consideration and applied to the entire image. The frequency and the phase of the sinusoidal wave, the variance of the Gaussian kernel and the filter orientation represents the filter parameters. We keep the variance constant,  $\sigma = 3.5$ , the optimal value found after testing.

The choice of the filters parameters is one of the most important part of the entire process, because the tuning of the Gabor filters is a NP-hard problem, so we need to reduce the bank of filters in order to choose the filters with the best accuracy. The ideal Gabor filter is the one with the frequency equal to the texture frequency and with the orientation of the texture.

Because the frequency of the texture can't be extracted without any apriori information about the real size of the texture in centimeters, we choose to approximate it based on the image size in pixels. The choice for approximation is presented in<sup>14</sup> where the frequency parameter  $u$  takes values between  $1/\sqrt{2}$ ,  $2/\sqrt{2}$ ,  $4/\sqrt{2}$ , ...,  $(N/4)/\sqrt{2}$ , where  $N$  represents the width of the texture.

For tuning the orientation parameter, we use the orientation angles detected in the previous step, applying the Hough transform on the magnitude spectrum. We generate the bank of filters and search for

the filter with the best response. From the input image, we randomly extract a number of  $N=4$  samples of the size of the filter and apply each filter to the samples extracted. We decided to use  $N$  number of samples and not a single sample because the textures are not perfectly periodic and the filter is not always mapped to a periodic portion of the image, and can also catch areas where only part of the periodical structure is represented from which the image is formed or edge areas.

Thus, with a larger number of samples, the correct filter has more chances to have the sum of the smallest convolution result in unfavorable cases, and the probability of defect decreases. The filter bank is incrementally sorted by the sum of convolution in absolute value for the  $N$  samples, then the first filter in the bench is selected as the correct one and it is further used in processing. The selected filter is applied over the image, representing the response of the image filtered with the optimum Gabor filter.

The response of the filtered image with the optimum Gabor filter undergoes a strong smoothing operation, resulting in a uniform image of different shades of gray depending on the texture in the image or the defective and non-defective areas. For smoothing, a  $25 \times 25$  filter was used, replacing the value of each pixel in the image with the average of its neighbors.

### Defect detection by pixel classification

The K-Means algorithm is an unsupervised learning algorithm that solves the problem of automatically grouping data from a set based on their common features. The basic idea of the algorithm is to divide the data according to its similarity into a number of  $k$  clusters,  $k$  decided in the initialization phase. Each cluster has a class center of randomly selected coordinates. The coordinates of the class center represent a set of different values depending on the data representation in the data set. Generally, for grouping points or pixels in  $k$  distinct classes, the two spatial coordinates  $x$  and  $y$  are used. However, this grouping method is not feasible, so the coordinates of the class center will be the image intensity in grayscale, representing an integer between 0 and 255.

For each pixel in the image, the similarity between the current pixel and all  $k$  class centers is calculated. The pixel will associate with the class center of which it has the strongest similarity. The most commonly used method for calculating similarity is the Euclidean distance between two points, but for this specific problem, we use the difference of class center intensity and the current pixel's intensity. After iterating through the entire image, it is assumed that all the pixels in the image associated with a class center form a cluster. In order for the class center to best characterize associated data (pixels), it must be at the center of the cluster. Thus, for each cluster, the class center moves in the center of the associated points.

Then the entire algorithm is repeated, until each pixel in the image remains associated with the same class center for two iterations in a row. At the end of the algorithm, data in the dataset is grouped into  $k$  clusters and the image is segmented in  $k$  regions. Each of the  $k$  regions can have 0 pixels, all the pixels from the image, or a number between the two limits of pixels associated, depending on the number of different textures or non-textures in the image. For a texture non-defective, only one cluster will contain all the pixels. When the pixels from the image will be associated to two or more clusters, we can conclude that the input image contains more than one texture or non-texture, so we can spot the defective area.

Experimental results


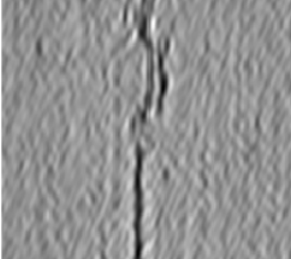
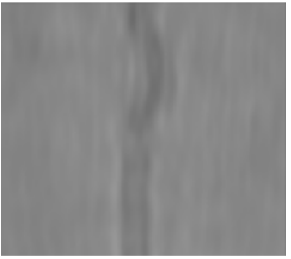


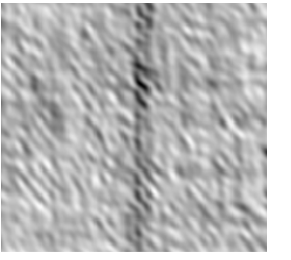



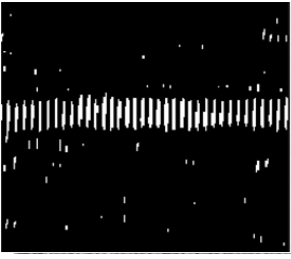

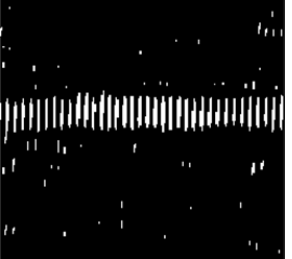
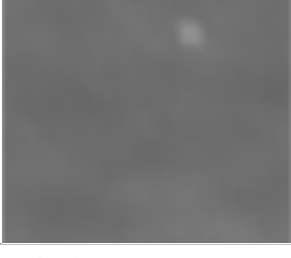
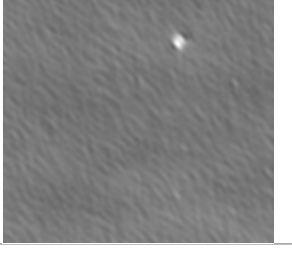
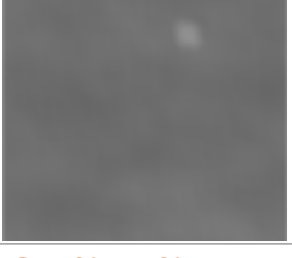
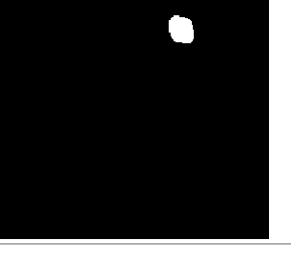
Based on the test results, it was found that the response time is directly proportional to the size of the input image. Small image sizes below 512×512 pixels are preferred. The quality of the response depends on the physical structure of the image and the periodicity of the texture, but also on the size and type of the defect in the image. Images with highly periodic and homogeneous structures are classified with much better accuracy than images where the textures do not follow a perfect pattern of yarn jointing. Therefore, textures with fine stitch patterns are preferred.

The result of the application also depends on the number of class centers of the K-Means algorithm. If the number of classes is less

than the number of textures and non-textures in the input image, the similar areas as intestines in the image will be grouped into the same cluster. Therefore, it is preferred that the number of clusters be greater than or equal to the number of textures and non-textures in the image. On the basis of the experiments, it was determined that a number of class's *k* of a value between 2 and 6 is sufficiently large for most of the input images.

Experimental results are shown in Table 1 below, where textures with various defects have been analyzed. It can be seen that the processing chain quite well separates the defect areas from the faultless areas. Certain areas of the image that are not part of the defect area are classified as defective because some portions deviate from the general structure of the texture.

Table 1 Results on different texture defects

Input Image	Filtered Image with the Selected Gabor Filter	Smoothed Image	Result (Classified Image)
			
			
			
			

Acknowledgements

None.

Conflict of interest

The authors declare that there is no conflict of interests regarding the publication of this paper.



## References

1. Pickett RM. Visual Analysis of Texture in the Detection and Recognition of Objects. In: Lipkin BC, Rosenfeld A, editors. *Picture Processing and Psychopictorics*. Academic Press, New York; 1970. p. 289–308.
2. Zhang J, Xin B, Wu X. A Review of Fabric Identification Based on Image Analysis Technology. *Textiles & Light Industrial Science and Technology*. 2013;2(3):120.
3. Imaoka H, Inui S, Niwaya H, et al. Trial on Automatic Measurement of Fabric Density. *J Society of Fiber Science & Technology*. 1988;44(1):32–39.
4. Wood EJ. Applying Fourier and Associated Transforms to Pattern Characterization in Textiles. *Textile Research J*. 1990;60(4):212–220.
5. Xu BG. Identifying Fabric Structures with Fast Fourier Transform Techniques. *Textile Research J*. 1996;66(8):496–506.
6. Sari-Saraf H, Goddard JS. On-line optical measurement and monitoring of yarn density in woven fabrics. *Automat Opt Inspect Ind, SPIE*. 1996;2899:444–452.
7. Padmavathi S, Prem P, Praveenn D. Locating Fabric Defects Using Gabor Filters. *International J Sci Res Eng & Technol*. 2013;2(8):472–478.
8. Anil K Jain, Farrokhnia F. *Unsupervised Texture Segmentation Using Gabor Filters*. Michigan State University, USA; 1990. p. 14–19.
9. Hammouda K, Ed Jernigan. *Texture Segmentation Using Gabor Filters*. University of Waterloo, Ontario, Canada; 2006. p. 1–8.
10. Clausi D, Ed Jernigan M. Designing Gabor filters for optimal texture separability. *Pattern Recognition*. 2000;33(11):1835–1849.
11. Heideman, Michael T Johnson, Don H Burrus, et al. Gauss and the history of the fast Fourier transform. *Archive for History of Exact Sciences*. 1985;34(3):265–277.
12. Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Trans Sys Man Cyber*. 1979;9(1):62–66.
13. Feng YL, Li RQ. Automatic Measurement of Weave Count with Wavelet Transfer. *J Textile Research*. 2001;22:30–31.
14. Jain A.K, Farrokhnia F. Unsupervised texture segmentation using Gabor filters. *Pattern Recognition*. 1991;24(12):1167–1186.