APPLICATION OF ADVANCED IMAGE PROCESSING TECHNIQUES TO AUTOMATIC KIKUCHI LINES DETECTION

Rafał Frączek, Tomasz Zieliński

Instrumentation & Measurement Department, AGH University of Science and Technology
Al. Mickiewicza 30, 30-059 Cracow, Poland
tel/fax: +(4812) 6338565, email: tzielin@uci.agh.edu.pl
www.agh.edu.pl

ABSTRACT

Automated crystal orientation measurement (ACOM) in the scanning electron microscope (SEM) is a standard technique of texture analysis (pattern recognition) that is used in materials science. The measurement is carried out by interpreting backscatter Kikuchi patterns, in particular by the extraction of the position of so-called Kikuchi bands, i.e. pairs of parallel lines. Their detection strongly depends on appropriate processing of a source image, which usually is highly corrupted by noise and has uneven background illumination. Such advanced processing is addressed in this paper. It exploits wavelet transform based de-noising as well as curve modification and curvelet transform based contrast enhancement methods. Additionally, directional, ridge detection type 2D filters are used for searching lines missing to pairs.

1. INTRODUCTION

Possibility of reproducible manufacturing of crystal materials with desired physical properties is one of the most important problems in material engineering. The crystal material properties are mainly determined by size and spatial orientation of crystal domains that in turn can be deduced (calculated) from the position of so-called Kikuchi bands, i.e. pairs of parallel lines, existing in electron microscopy images (see figure 1). When an electron beam impinges on a crystalline solid, it is spread out as a wide shower consisting of scattered electrons. After leaving the crystal, a Kikuchi diffraction pattern (a set of pairs of parallel lines) is produced on a phosphor screen. Its geometry is unique for the crystal structure and hence its extraction is essential for the evaluation of the crystal material properties. Kikuchi bands usually form characteristic broad stripes of intensity in the image and their edges (Kikuchi lines) are often fragmented and diffused. Moreover, the microscopy images are usually corrupted by noise. Therefore some of the Kikuchi lines are hardly visible and difficult to detect, and in such situation image processing plays a very important role. Its goal is noise suppression and lines efficient enhancement.

The goal of this paper is to propose significant improvements of the Kikuchi lines detection algorithm developed by authors and described in [1]. Two efficient image pre-processing steps have been added to it, namely wavelet transform-based image de-noising and curvelet transform-based contrast enhancement. Additionally, the algorithm has been equipped with special procedure for searching lines missing to pairs that exploits directional, ridge detection type filters.

2. KIKUCHI LINES DETECTION ALGORITHM

The discussed method is an extension to the one presented in [1] and can be summarized as follows:

1) **Correction of unequal background illumination** by means of subtraction of local average values.
2) **Image de-noising** by shrinkage of its wavelet coefficients (section 3 and 4).
3) **Contrast enhancement** using curve modification method and curvelet transform (section 3 and 5).
4) **Directional filtering**. The pre-processed image is filtered twice with a mask oriented at each of four orientations: vertical, horizontal and at the angles of 45 and 135 degrees. As a result four filtered images are obtained and they are processed separately later.
5) **Segmentation (binarization)**. At this step each filtered image is segmented into background and objects (lines). A global thresholding method is applied.
6) **Binary image enhancement**. A sequence of morphological operations is used to make binary images more suitable for further processing.
3. NOISE ESTIMATION

After correction of unequal background illumination, realized by subtraction of local average values, estimation of noise standard deviation is carried out. For this purpose it is necessary to extract image regions with noise only. First, the global mean value is subtracted from the image. Next, the image is partitioned into square blocks of 4×4 pixels that are transformed separately by 2D DFT. Two situations are possible: an individual block includes a Kikuchi line(s) or does not. In order to distinguish between these two cases a special parameter \( v \) is calculated:

\[
v = F(0,0)^2 + F(0,1)^2 + F(1,0)^2 + F(1,1)^2
\]

where \( F(i, j) \) denotes a magnitude of DFT coefficient. If \( v \) is higher than empirically found threshold then it is assumed that relevant block includes a Kikuchi line. The threshold value is found on the basis of the number of incorrectly classified blocks. From figure 2 one can conclude that the optimal threshold value is equal to 2.9.

Finally, the entire image is divided into two subsets: the first one consists of blocks that include only noise and the second one is made up of blocks containing lines. At this point, the first subset is used to estimate noise standard deviation \( \sigma \) which is needed in further steps.

4. IMAGE DE-NOISING

Since images from electron microscope are highly corrupted by noise, it is very difficult to detect in them hardly visible lines. The presence of strong noise makes wrong decision and false “detection” of a line (that does not exist) very probable.

For that reason several de-noising methods were carefully tested. The following methodology was applied. First, an artificial, noise-free, reference image containing different types of lines was created. Generated lines had different width, length and contrast in respect to the background. Then, the reference image was embedded in noise extracted from original microscope images (ten different 8×8 square blocks containing only noise were added to the reference in random order). Finally, the generated noisy image was processed by several de-noising methods, and obtained results were compared to the reference image using the standard peak signal to noise ratio (PSNR).

The following de-noising methods have been applied and tested: pixel averaging within 3×3 window, median filtering with 3×3 window, adaptive Wiener filtering, linear and non-linear complex diffusion method [2], automatic noise removal using fourth order partial differential equations (PDE) [3], standard wavelet de-noising using details’ coefficients thresholding [4], wavelet de-noising using pyramidal directional filter bank decomposition (curvelet-like transform) [5] and de-noising using neighbouring wavelet coefficients (\textit{NeighShrink} wavelet de-noising method) [6]. Comparison of the observed efficiency of different approaches is presented in table 1.

In the standard and \textit{NeighShrink} wavelet de-noising methods, application of the following wavelet types was verified: Haar, Beylkin, Coiflet, Daubechies, Symmlet, Vaidyanathan.

\[\text{Tab. 1. De-noising efficiency of different methods (PSNR}_{\text{out}}=21\text{dB).}\]

<table>
<thead>
<tr>
<th>Method</th>
<th>PSNR(_{\text{out}}) [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fourth order PDE [3]</td>
<td>24.81</td>
</tr>
<tr>
<td>Linear complex diffusion [2]</td>
<td>25.23</td>
</tr>
<tr>
<td>Pixel averaging</td>
<td>25.63</td>
</tr>
<tr>
<td>Median filtering</td>
<td>25.78</td>
</tr>
<tr>
<td>Curvelet-like de-noising [5]</td>
<td>28.03</td>
</tr>
<tr>
<td>Adaptive Wiener filtering</td>
<td>28.60</td>
</tr>
<tr>
<td>Standard wavelet de-noising (Haar) [4]</td>
<td>29.39</td>
</tr>
<tr>
<td>\textit{NeighShrink} wavelet de-noising (Haar) [6]</td>
<td>30.41</td>
</tr>
</tbody>
</table>

\[\text{Tab. 2. Efficiency of wavelet transform-based de-noising methods.}\]

<table>
<thead>
<tr>
<th>Wavelet type</th>
<th>PSNR(_{\text{out}}) [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Haar</td>
<td>29.39</td>
</tr>
<tr>
<td>Beylkin</td>
<td>27.87</td>
</tr>
<tr>
<td>Coiflet</td>
<td>28.48</td>
</tr>
<tr>
<td>Daubechies</td>
<td>28.72</td>
</tr>
<tr>
<td>Symmlet</td>
<td>28.63</td>
</tr>
<tr>
<td>Vaidyanathan</td>
<td>27.74</td>
</tr>
<tr>
<td>Battle</td>
<td>28.32</td>
</tr>
</tbody>
</table>
nathan and Battle. The observed maximum PSNR for each wavelet type is shown in table 2. Since the NeighShrink algorithm with Haar wavelet outperformed other methods and offered the highest PSNR, it was added to the old version of the Kikuchi line detection algorithm [1] and tested with it in the further part of the paper.

5. CONTRAST ENHANCEMENT

Some Kikuchi lines are characterized by very low contrast between the line and surrounding background that should be enhanced.

5.1. Contrast enhancement by curve modification

Image curve modification is known to be a very simple and fast technique for modification of the pixel distribution in any image. In order to avoid the amplification of remaining noise only pixels lying in 4x4 blocks already classified as “lines” are mapped to their new values according to the function:

\[ y = 128 + 127 \cdot \frac{\text{atan}(m \cdot (x - 128))}{\text{atan}(128)} \]  

(2)

which is depicted in figure 3. The parameter \( m \) controls the function slope. If \( m \) is too high then both, possible artefacts after de-nosing step and remaining noise, can be significantly amplified. If parameter \( m \) is too low then contrast increase is not sufficient. Thus, a trade-off is necessary and \( m = 0.07 \) has been chosen experimentally. The values of the transfer function are computed and rounded in advance and stored in an array of integers. Next, for the given input pixel value, the transformed output pixel value is directly obtained from this array.

As it can be observed from the outline of the proposed transfer function presented in figure 3, the applied curve modification yields the contrast increase especially for pixels values lying near 128 where pixels belonging to hardly visible lines are the most frequently concentrated.

\[ f(x) = \begin{cases} 
1, & \text{if } x < c \\
\frac{x-c}{m} \cdot \frac{2c-x}{c}, & \text{if } c \leq x < 2c \\
\left(\frac{m}{x}\right)^p, & \text{if } 2c \leq x < m \\
\left(\frac{m}{x}\right)^s, & \text{if } x \geq m 
\end{cases} \]  

(3)

and 3) calculating the inverse transform. In (3) the \( p \) parameter determines the degree of non-linearity whereas \( s \) controls dynamic range compression. Nonzero value of \( s \) leads to enhancement of weak lines but the same time strong lines can be soften when this value is too high. In turn, the \( c \) parameter is a normalization factor. Its value is related to the noise standard deviation \( \sigma \). If \( c \) is larger than 3\( \sigma \) then the noise will not be amplified. Finally, the \( m \) parameter determines the value under which curvelet transform coefficients are amplified. The value of \( m \) is derived from the maximum curvelet coefficient \( M \) of the relative band (\( m < kM \), with \( k < 1 \)). Exemplary outline of the \( f(x) \) function is presented in figure 4 for \( p = 0.5 \), \( s = 0.6 \), \( c = 3 \), \( m = 30 \), (the dashed line is drawn only as reference). All parameters’ values have been chosen experimentally.

![Fig. 3. The outline of the applied transfer function (2)](image)

![Fig. 4. The exemplary outline of the function \( f(x) \) (3).](image)
6. TUNED DIRECTIONAL FILTERING

The directional 2D filter is designed to meet the optimal conditions for a ridge (line) detection and fulfills the following criteria [8].

- **Signal-to-Noise Ratio (SNR).** The response of a filter \( h(x, y) \) to a particular signal \( f(x, y) \) (e.g., an idealised edge or ridge) centered at the origin is equal:

\[
M = \int_{\mathbb{R}^2} f(x, y)h(-x,-y)dxdy
\]  

(4)

\( M \) is given by the height of the response at its maximum. It is desired to have a high value of \( M \).

- **Localization.** The estimated feature position corresponds to the maximum of the response in the direction orthogonal to the feature boundary. The localization term which is a measure of the width of the maximum (peak) is given by:

\[
L = -\int_{\mathbb{R}^2} f(x, y)h_{yy}(-x,-y)dxdy
\]  

(5)

It is desired to maximize the \( L \) term.

- **Elimination of false oscillations.** The response should be relatively free from oscillations orthogonal to the feature boundary. This can be achieved by penalizing the term:

\[
R = \int_{\mathbb{R}^2} h_{yy}(x,y)^2dxdy + \int_{\mathbb{R}^2} h_{xx}(x,y)^2dxdy
\]  

(6)

The individual terms are combined to obtain a single criterion:

\[
C = M \cdot L - \mu \cdot R
\]  

(7)

The filter that maximizes this criterion is our optimal filter. The parameter \( \mu \) controls the smoothness of the filter and in consequence the level of oscillations along the ridge. The higher value results in a filter having less tendency to false maxima. The criterion (7) can be transformed to a matrix form and the optimal solution is found analytically by using Lagrange’s multiplier method [8]. We have used 2D filter with \( M = 2 \) and \( \mu = 0 \) because this detector is better than classical detector in terms of SNR and localization. The higher order detector would be more oscillatory and it could give more false detections. In our Kikuchi lines detection method the image is filtered by the directional filter tuned to a detected unpaired line and again processed with the Hough transform.

7. RESULTS

The whole algorithm has been implemented in Matlab and tested with and without the image processing techniques presented in the paper. The tests were carried out on 30 images having 312 Kikuchi bands (pairs of parallel lines), obtained from the Institute of Metallurgy and Materials Science of the Polish Academy of Sciences, Cracow, Poland. The final detection results are presented in table 3. The standard algorithm, taken also into account in the table, denotes the program that is routinely used by material science professionals at IMMS-PAS. This method exploits the material science professionals at IMMS-PAS. This method exploits the

a typical way and does not utilize advanced: 1) image pre-processing methods, 2) results verification and 3) searching for missing lines to pairs. Numbers of “true” and “false” lines, presented in the table, were evaluated by means of manual verification of automatic decisions (each “detected” line pair was manually verified whether it corresponds to a real Kikuchi pair on the image or not).

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Number of “true” line pairs</th>
<th>Number of “false” line pairs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard – used at IMMS-PAS</td>
<td>224 (71.8%)</td>
<td>36</td>
</tr>
<tr>
<td>Old – described in [1]</td>
<td>261 (86.2%)</td>
<td>2</td>
</tr>
<tr>
<td>[1] + contrast enhancement</td>
<td>282 (90.4%)</td>
<td>4</td>
</tr>
<tr>
<td>[1] + de-noising + contrast</td>
<td>282 (90.4%)</td>
<td>2</td>
</tr>
<tr>
<td>[1] + de-noising + contrast + pairs</td>
<td>287 (91.9%)</td>
<td>2</td>
</tr>
</tbody>
</table>

The obtained results confirm the fact that proper choice of image pre-processing methods has a big impact on effective Kikuchi bands detection. The most successful was addition of the carefully selected contrast enhancement techniques resulting in significant increase of detected “true” line pairs. In contrary, it turned out that additional sophisticated image de-noising causes only small reduction of the number of detected “false” line pairs (what is also very important in practice). Finally, 5 additional line pairs were detected due to extra application of tuned directional 2D filters.

REFERENCES


ACKNOWLEDGMENTS

Authors would like to thank very much doctor Adam Morawiec from the Institute of Metallurgy and Materials Science PAN in Cracow, Poland for making the microscopy available to them.