Wavelet and Eigen-Space Feature Extraction for Classification of Metallography Images

Pavel Praks\textsuperscript{a,1}, Marcin Grzegorzek\textsuperscript{b}, Rudolf Moravec\textsuperscript{c}, Ladislav Válek\textsuperscript{d}, and Ebroul Izquierdo\textsuperscript{b}

\textsuperscript{a} Dept. of Applied Mathematics, Technical University of Ostrava, Czech Republic
\textsuperscript{b} Multimedia & Vision Research Group, Queen Mary University of London, UK
\textsuperscript{c} Research and Development, Mittal Steel Ostrava plc, Ostrava, Czech Republic
\textsuperscript{d} QT - Production Technology, Mittal Steel Ostrava plc, Ostrava, Czech Republic

Abstract. In this contribution a comparison of two approaches for classification of metallography images from the steel plant of Mittal Steel Ostrava plc (Ostrava, Czech Republic) is presented. The aim of the classification is to monitor the process quality in the steel plant. The first classifier represents images by feature vectors extracted using the wavelet transformation, while the feature computation in the second approach is based on the eigen-space analysis. Experiments made for real metallography data indicate feasibility of both methods for automatic image classification in hard industry environment.

Keywords. Measurement, hard industry, human factors, content-based image retrieval, wavelet transformation, statistical classification, numerical linear algebra, partial symmetric eigenproblem, iterative solvers.

1. Introduction

Any meaningful human activity requires perception. Under perception realization, evaluation, and interpretation of sensory impressions is understood. It allows the human to acquire knowledge about the environment, to react to it, and finally to influence it. There is no reason in principle why perception could not be simulated by some other matter, or instance, a digital computer [6]. The aim of the simulation is not the exact modeling of the human brain activities, but the obtainment of similar perception results. Research activities concerned with the mathematical and technical aspects of perception are the field of pattern recognition. One of the most important perceptual abilities is vision. The processing of visual impressions is the task of image analysis. The main problem of image analysis is the recognition, evaluation, and interpretation of known patterns or objects in images.
In this paper, the problem of automatic pattern classification in real metallography images from the steel plant of Mittal Steel Ostrava plc (Ostrava, Czech Republic) is addressed. The objective is to monitor the process quality in the steel plant. For this reason two different image classification algorithms are used and compared in this contribution. The first one computes feature vectors with the wavelet transformation, while in the second one the eigen-space analysis is applied.

The paper is structured as follows. Section 2 describes the theoretical background of the wavelet-based image classifier. In Section 3, intelligent image retrieval using the partial eigen-problem is presented. Experimental comparison of these two approaches for image classification follows in Section 4, while Section 5 closes the paper with some conclusions.

2. Statistical Wavelet-Based Classification

In this section, a statistical wavelet-based approach for image classification is presented. Section 2.1 describes the training of statistical models for different image concepts. These models are then used for image classification, which is presented in Section 2.2.

2.1. Training of Statistical Concept Models

Before images can be classified in the recognition phase (Section 2.2), statistical models $\mathcal{M}_\kappa$ for all image concepts $\Omega_\kappa$ considered in a particular classification task are learned in the training phase. The concept modeling starts with the collection of training data. In this work real metallography images from a cooking plant are used for this reason. Subsequently, the original training images are converted and resized into gray level images of size $2^n \times 2^n$ ($n \in \mathbb{N}$) pixels. In all these preprocessed training images 2D local feature vectors $c_{\kappa,m}$ are extracted using the wavelet transformation [4]. Training images are divided into neighborhoods of size $2^{[3]} \times 2^{[3]}$ (in Figure 1, $4 \times 4$ pixels). These neighborhoods are treated as 2D discrete signals $b_0$ and decomposed to low-pass and high-pass coefficients. The resulting coefficients $b_\tilde{s}$, $d_{0,\tilde{s}}$, $d_{1,\tilde{s}}$, and $d_{2,\tilde{s}}$ are then used for feature vector computation.
\[ c_{\kappa,m}(x_m) = \begin{pmatrix} \ln(2^2|b_2|) \\ \ln[2^2(|d_0| + |d_1| + |d_2|)] \end{pmatrix}. \]  

The feature vectors \( c_{\kappa,m} \) are modeled by normal density functions \( p_{\kappa,m} = p(c_{\kappa,m} | \mu_{\kappa,m}, \sigma_{\kappa,m}) \).

Due to the large number of training images for each concept \( \Omega_\kappa \), it is possible to estimate the mean value vector \( \mu_{\kappa,m} \) and the standard deviation vector \( \sigma_{\kappa,m} \) for all image locations \( x_m \), i.e., all feature vectors \( c_{\kappa,m} \). Finally, statistical models \( M_\kappa \) for all image concepts \( \Omega_\kappa \) are created and ready for use in the classification phase (Section 2.2).

### 2.2. Image Classification

Once the concept modeling (Section 2.1) is finished, the system is able to classify images taken from a real world environment. First, a test image \( f \) is taken, preprocessed, and local feature vectors \( c_m \) are computed in it in the same way as in the training phase (Section 2.1). Second, the classification algorithm based on the Maximum Likelihood (ML) Estimation is started.

The task of the image classification algorithm is to find the concept \( \Omega_\kappa \) (or just its index \( \hat{\kappa} \)) of the test image \( f \). In order to do so, the density values for all concepts \( \Omega_\kappa \) have to be compared to each other. Assuming that the feature vectors \( c_m \) are statistically independent on each other, the density value for the given test image \( f \) and concept \( \Omega_\kappa \) is computed with

\[ p_\kappa = \prod_{m=1}^{M} p(c_m | \mu_{\kappa,m}, \sigma_{\kappa,m}) \]  

where \( M \) is the number of all feature vectors in the image \( f \). All data required for computation of the density value \( p_\kappa \) with (3) is stored in the statistical concept model \( M_\kappa \). These density values are then maximized with the Maximum Likelihood (ML) Estimation [10]

\[ \hat{\kappa} = \arg\max_\kappa p_\kappa. \]

Having the index \( \hat{\kappa} \) of the resulting concept the classification problem for the image \( f \) is solved.

### 3. Latent Semantic Indexing

In this section, we present the intelligent image retrieval using the partial eigen-problem. The numerical linear algebra is used as a basis for the information retrieval in the retrieval strategy called Latent Semantic Indexing, see for instance [1], [2]. LSI can be viewed as a variant of a vector space model, where the database is represented by the document matrix, and a user’s query of the database is represented by a vector. LSI also contains a low-rank approximation of the original document matrix via the Singular Value Decomposition (SVD) or the other numerical methods. The SVD is used as an automatic tool.
for identification and removing redundant information and noise from data. The next step of LSI involves the computation of the similarity coefficients between the filtered user’s query and filtered document matrix. The well-known cosine similarity can be used for a similarity modeling. Recently, the methods of numerical linear algebra are also successfully used for the face recognition and reconstruction [5], image retrieval [8,7], as a tool for information extraction from internet data [9] and for iris recognition problem [7].

The "classical" LSI application in information retrieval algorithm has the following basic steps:

i) The Singular Value Decomposition of the term matrix using numerical linear algebra. SVD is used to identify and remove redundant noise information from data.

ii) The computation of the similarity coefficients between the transformed vectors of data and thus reveal some hidden (latent) structures of data.

Numerical experiments pointed out that some kind of dimension reduction, which is applied to the original data, brings to the information retrieval following two main advantages: (i) automatic noise filtering and (ii) natural clustering of data with "similar" semantic.

3.1. Image Coding

In our approach [8,7], a raster image is coded as a sequence of pixels. Then the coded image can be understood as a vector of an \( m \)-dimensional space, where \( m \) denotes the number of pixels (attributes). Let the symbol \( A \) denote a \( m \times n \) term-document matrix related to \( m \) keywords (pixels) in \( n \) documents (images). Let us remind that the \((i,j)\)-element of the term-document matrix \( A \) represents the colour of \( i\)-th position in the \( j\)-th image document.

3.2. Implementation Details of Latent Semantic Indexing

In this section we will describe the possible software implementation of the Latent Semantic Indexing method.

Let the symbol \( A \) denotes the \( m \times n \) document matrix related to \( m \) pixels in \( n \) images. The aim of SVD is to compute decomposition

\[
A = USV^T,
\]

where \( S \in \mathbb{R}^{m \times n} \) is a diagonal matrix with nonnegative diagonal elements called the singular values, \( U \in \mathbb{R}^{m \times m} \) and \( V \in \mathbb{R}^{n \times n} \) are orthogonal matrices\(^1\). The columns of matrices \( U \) and \( V \) are called the left singular vectors and the right singular vectors respectively. The decomposition can be computed so that the singular values are sorted in decreasing order.

The full SVD decomposition (5) is memory and time consuming operation, especially for large problems. Although the document matrix \( A \) is often sparse, the matrices \( U \) and \( V \) have a dense structure. Due these facts, only a few \( k \)-largest singular values of \( A \) and the corresponding left and right singular vectors are computed and stored in memory. The number of singular values and vectors which are computed and kept in memory

\(^1\) A matrix \( Q \in \mathbb{R}^{n \times n} \) is said to be orthogonal if the condition \( Q^{-1} = Q^T \) holds.
can be chosen experimentally as a compromise between the speed/precision ratio of the LSI procedure. We implemented and tested LSI procedure in the Matlab system by Mathworks. Following [2] the Latent Semantic Indexing procedure can be written in Matlab by the following way.

**Procedure Original LSI [Latent Semantic Indexing]**

```matlab
function sim = lsi(A,q,k)
    % Input:
    % A ... the m x n matrix
    % q ... the query vector
    % k ... Compute k largest singular values and vectors; k ≤ n
    % Output:
    % sim ... the vector of similarity coefficients

    [m,n] = size(A);

    1. **Compute the co-ordinates of all images in the k-dimensional space by the partial SVD of a document matrix A.**
       
       \[ [U,S,V] = svds(A,k); \]
       
       % Compute the k largest singular values of A; The rows of V contain the co-ordinates of images.

    2. **Compute the co-ordinate of a query vector q**
       
       \[ qc = q' * U * pinv(S); \]
       
       % The vector qc includes the co-ordinate of the query vector q; The matrix \( \text{pinv}(S) \) contains reciprocals of non-negative singular values (an pseudoinverse); The symbol ‘ denotes the transpose superscript.

    3. **Compute the similarity coefficients between the co-ordinates of the query vector and images.**
       
       \[ \text{sim}(i) = (qc*V(i,:))' / (\text{norm}(qc) * \text{norm}(V(i,:))); \]
       
       % Compute the similarity coefficient for i-th image; V(i,:) denotes the i-th row of V.

    The procedure \( \text{lsi} \) returns to a user the vector of similarity coefficients \( \text{sim} \). The i-th element of the vector \( \text{sim} \) contains a value which indicate a "measure" of a semantic similarity between the i-th document and the query document. The increasing value of the similarity coefficient indicates the increasing semantic similarity.

3.3. Partial Eigen-problem

The image retrieval process can be powered very effectively when the time consuming Singular Value Decomposition of LSI is replaced by the partial symmetric eigenproblem, which can be solved by using fast iterative solvers [7].

Let us assume the following relationship between the singular value decomposition of the matrix A and the symmetric eigenproblem of the symmetric square matrices \( A^T A \):
\[ A = U S V^T \]  

(6)

\[ A^T = (U S V^T)^T = V S^T U^T \]  

(7)

\[ A^T A = V S^T (U^T U) S V^T = V S^T S V^T \]  

(8)

Moreover, let us assume the SVD decomposition (5) again. Because of the fact that the matrix \( V \) is orthogonal, the following matrix identity holds:

\[ AV = US. \]  

(9)

Finally, we can express the matrix \( U \) in the following way:

\[ AV S^+ \approx U \]  

(10)

Here the symbol \( S^+ \) denotes the Moore-Penrose pseudoinverse (pinv). Let us accent that the diagonal matrix \( S \) contains only non-negative singular values for real cases; The singular values less than \( \text{tol} \approx 0 \) are cut off by the Matlab \( \text{eigs}(A'*A, k) \) command.

There is no exact routine for the selection of the optimal number of computed singular values and vectors [3]. For this reason, the number of singular values and associated singular vectors used for the partial symmetric eigenproblem was estimated experimentally, but it seems that \( k < 10 \) is suitable for real image databases. For example, we choose \( k = 8 \) for the large-scale NIST TRECVID 2006 data [11].

In contrast to the SVD approach, the size of the partial symmetric eigenproblem (the size of \( A^T A \) matrix) does not depend on the number of pixels (keywords) at all. Since the number of computed singular values \( k \ll n \) for real problems and \( k \) is small, the image retrieval using the partial symmetric eigenproblem is more efficient [7] than the "classical" SVD approach [2].
4. Experiments and Results

4.1. Experimental Data

We experimented with real metallography images taken from the steel plant of Mittal Steel Ostrava plc, Ostrava, Czech Republic. In fact, we deal with sample images of con-
Figure 5. An example of results of the wavelet (left) and partial eigen-problem based image retrieval (right). The query image is situated in left up corner and it is related to the query SDK53M27. All retrieved images (except image no. 2) are well-classified by the wavelet method. All retrieved images (except image no. 4) are well-classified by the partial eigen-problem method.

continuously cast steel from billet device for continuous steel casting. This device produces billets of 180 mm square, 160 and 210 mm round. The closer parameters are stated in the Table 1. Steel samples from the cast billets are taken away for the device for continuous steel casting. These are crosscuts of the cast billets. These samples are conveyed into metallography lab where they are mechanically adjusted. In order to stress a sample macrostructure, crosscut etching is done. Consequently, photographs of these etched crosscuts are being taken.

<table>
<thead>
<tr>
<th>Commissioned on:</th>
<th>7 December 1993</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>billet, radial, two-point alignment</td>
</tr>
<tr>
<td>Heat volume:</td>
<td>205 tons</td>
</tr>
<tr>
<td>Casting method:</td>
<td>closed, through submerged nozzles and stoppers</td>
</tr>
<tr>
<td>Casting arc radius:</td>
<td>10.5 ; 21m (two-point alignment)</td>
</tr>
<tr>
<td>Cooling of semi-product:</td>
<td>water (single component)</td>
</tr>
<tr>
<td>Cutting of semi:</td>
<td>torch cutting</td>
</tr>
<tr>
<td>Slab marking:</td>
<td>punching, 10-character code</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Table 1.</strong> Basic chosen parameters of the device for continuous casting No. 1.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Evaluation time of one image</th>
<th>0.36 secs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local feature vectors from neighborhoods</td>
<td>8 × 8 pixels</td>
</tr>
<tr>
<td>Type of wavelet transformation</td>
<td>8 T AB Johnshon Wavelet</td>
</tr>
</tbody>
</table>

Table 2. Properties of the Statistical Wavelet-Based Classification method.
### Properties of the document matrix \(A\)

<table>
<thead>
<tr>
<th>Properties</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of keywords</td>
<td>(458 \times 480 = 219,840)</td>
</tr>
<tr>
<td>Number of documents</td>
<td>40</td>
</tr>
<tr>
<td>Size in memory</td>
<td>67.089 MB</td>
</tr>
</tbody>
</table>

### The SVD-Free LSI processing parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dim. of the original space</td>
<td>40</td>
</tr>
<tr>
<td>Dim. of the reduced space ((k))</td>
<td>6</td>
</tr>
<tr>
<td>Time for (A^T A) operation</td>
<td>0.64 secs.</td>
</tr>
<tr>
<td>Results of the eigensolver</td>
<td>0.219 secs.</td>
</tr>
<tr>
<td>The total time</td>
<td>0.859 secs.</td>
</tr>
</tbody>
</table>

Table 3. Image retrieval using the partial eigen-problem method. Properties of the document matrix (up) and LSI processing parameters (down).

Photographs from the verification of electromagnetic steel mixing in the crystallizer have been used for verifying of the described methods. The total number of images in the image database was 83. The number of images in the training set was 20.

### 4.2. Experimental Results

The results of image retrieval experiments are presented in Fig. 2 - Fig. 5. The resulted images are presented by decreasing order of similarity. The query image is situated in left upper corner.

The similarity of the query image and the retrieved image is written in parentheses. In order to achieve well arranged results, only 9 most significant images are presented.

The presented shape of the crosscuts does not respond to a reality completely (they were slightly deformed at the photograph evaluation). It can be stated that these are the first results for billets 180mm square and 210mm round. The evaluated subject was the whole crosscut of billet samples.

### 4.3. Conclusions for Experiments

Our results indicate that the both methods can automatically recognize the shape and the type of images found in our image database. The behaviour of both methods is close to the classification of a human expert. Moreover, the results of Table 2 and Table 3 indicate a possibility of real-time analysis.

The first results point out that the discussed methods can be also used for the evaluation of crosscut macro structure of billet samples. In order to achieve more precise evaluation results, individual areas of a sample crosscut images should be deeply analyzed in the future work. This deeper image analyze is also important for searching metallurgical relations in images, which are hidden in the image database.

### 5. Conclusion

In this paper, a comparison of two approaches for automatic pattern classification in images taken from a real world environment has been presented. The experimental data in the form of metallography images has been provided by the Mittal Steel Ostrava plc (Ostrava, Czech Republic). The objective of this research activity is to monitor the quality.
process in the steel plant. The first classifier used for this reason (Section 2) represents image patterns by feature vectors extracted with the wavelet transformation, while the second one (Section 3) is based on the eigen-space analysis. Classification results for experiments presented in Section 4 prove a very high performance of both approaches in a real world environment.

In the future, the statistical wavelet-based approach (Section 2) will be combined with the eigen-based analysis (Section 3). One can imagine that a fusion of these two methods will bring a significant improvement in terms of classification rates.

Acknowledgments

The research has been also partially supported by a program "Information Society" of the Academy of Sciences of the Czech Republic, project No. 1ET401940412. The work leading to this contribution has been partially supported by the European Commission under contract FP6-027026-K-SPACE.

References