Highlights:

- Evaluating statistical, transform, model, and structural-based texture features for DIR
- Comparative analysis of the DIR results obtained from 26 texture features
- Providing a computational time analysis of texture features used for DIR

A Comparative Study of Different Texture Features for Document Image Retrieval

Fahimeh Alaei[1], Alireza Alaei[2], Umapada Pal[3], Michael Blumenstein[4]
[1] School of ICT, Griffith University, Australia
[2]Southern Cross University, Australia
[3]CVPR Unit, Indian Statistical Institute, India
[4]University of Technology Sydney, Australia
fahimeh.alaei@griffithuni.edu.au[1], alireza20alaei@gmail.com[2], umapada@isical.ac.in[3]
michael.blumenstein@uts.edu.au[4]

Abstract: Due to the rapid increase of different digitised documents, there has been significant attention dedicated to document image retrieval over the past two decades. Finding discriminative and effective features is a fundamental task for providing a fast and more accurate retrieval system. Texture features are generally fast to compute and are suitable for large volume data. Thus, in this study, the effectiveness of texture features widely used in the literature of content-based image retrieval is investigated on document images. Twenty-six different texture feature extraction methods from four main categories of texture features, statistical, transform, model, and structural-based approaches, are considered in this research work to compare their performance on the problem of document image retrieval. Three document image datasets, MTDB, ITESOFT, and CLEF_IP with various content and page layouts are used to evaluate the twenty-six texture-based features on document image retrieval systems. The retrieval results are computed in terms of precision, recall and F-score, and a comparative analysis of the results is also provided. Feature dimensions and time complexity of the texture-based feature methods are further compared. Finally, some conclusions are drawn and suggestions are made about future research directions.

Keywords: Statistical texture features, transform-based texture features, model-based features, structural features, document image retrieval.

1. Introduction

A substantial number of documents, magazines, letters, books, etc. are acquired through electronic devices in everyday life. To access the massive and ever-increasing quantity of document images, a primary demand and residual requirement is a fast and effective document image retrieval system. Document image retrieval (DIR) is an automatic approach for finding similar document images from a huge collection of structured and unstructured digitised document images (Marinai et al., 2011). There are many OCR-based document image retrieval approaches in the literature (Gordo et al., 2010). Traditional OCR-based document image retrieval has some deficiencies, such as sensitivity to image resolution, high computational cost, and language dependency (Gordo et al., 2010). Therefore, employing recognition-based approaches cannot provide appropriate results when document images are multilingual, or have low quality and different layouts. To cope with the difficulties of the recognition-based DIR methods, recognition-free DIR methods have been proposed in the literature (Marinai et al., 2011). In the recognition-free DIR methods, the contents of documents are ignored and document images are retrieved based on visual similarity to a query image. Thus, vectorial representations of the document images are considered for retrieval purposes. The features can be extracted from different levels of document images, such as pixel level, connected component level, or word level, to represent document images (F. Alaei et al., 2016a).

As a recognition free-based approach for document image retrieval, layout similarity via the modified X-Y tree was considered for the retrieval process in (Cesarini et al., 2002). Global features

and a layout representation obtained from the corresponding modified X-Y tree of a document page have been considered as features. Similarly, a layout analysis-based system was proposed for document image retrieval in (Pirlo et al., 2014). In this system (Pirlo et al., 2014), the layout of a document image was characterised by a grid-structure and then a Radon transform was used for the layout description. In (Kumar et al., 2014), by using bag-of-visual features, the structural similarity of document images was considered for the classification and retrieval processes. In this method, document images were recursively partitioned in vertical and horizontal directions to model the spatial relationships of the document layout in the feature extraction step. A codebook based on histograms of Speeded-up Robust Features (SURF) extracted from each section of document images was created. Each document image was then encoded using the created codebook. In (Li et al., 2009), local feature sequences have been employed for document image retrieval. For local feature sequence matching, a coarse-to-fine sub-string matching strategy was applied. Another document image retrieval technique in the literature was based on two-dimensional density distributions (Kise et al., 2003). To improve the retrieval results, pseudo-relevance feedback was further employed (Kise et al., 2003). The above-mentioned approaches cannot support the retrieval process or perform well, if documents are unstructured or they are different in size, font, and languages.

Given the spread of digitised documents, it is imperative to consider the methods that can address the difficulties of previous approaches. The effectiveness of texture-based methods has been established in various applications of document image analysis (F. Alaei et al., 2016c; Mehri et al., 2017; Mehri et al., 2015). In (Mehri et al., 2017), texture features have been used for discriminating and segmenting graphical components from textual contents in historical document images. Furthermore, the performance of some texture features, such as local binary pattern (LBP), autocorrelation, Gabor, and wavelet transforms have been investigated for segmentation of historical document images (Mehri et al., 2017). In (K. Chen et al., 2014), a Gabor dominant orientation histogram and LBP, as texture features, have been used for segmenting historical handwritten document images. In (Dey et al., 2016), LBP and spatial sampling have been used for information spotting in historical document images. The usefulness of texture features for document image retrieval has been investigated in (F. Alaei et al., 2016c). Various LBP-based feature extraction methods have been considered to rank the document images based on the visual similarity of training document images to a given query. In (F. Alaei et al., 2016b), fusion has been employed to obtain better document retrieval performance using LBP and wavelet transform texture features, and combining classifier scores,

A considerable amount of document image analysis work using texture features has been published in the literature. To the best of our knowledge, however, no study focusing on a comparison of different texture-based methods for DIR is available in the literature. This research work is to addresses the following questions in relation to the use of different texture-based feature extraction methods for document image retrieval: which category of texture features can provide better performance for document image retrieval?; which texture feature provides the best performance in each category?; will document image retrieval based on texture features be influenced by the resolution of document images?

Investigation and answers to the above-mentioned questions have led to the main contributions of this research work and are summarised below.

- Evaluation of the effectiveness of twenty-six state-of-the-art texture-based feature extraction methods from four different categories applied for document image retrieval.

- Setting up a comparative study of the results obtained from different texture-based features employed for DIR on three document image datasets in terms of the retrieval performance, feature dimensions, and computing time.

The rest of this paper is organised as follows. In Section 2, the overview of the literature of texture features with a focus on DIR are provided and explained. In Section 3, the document image retrieval system is presented. Experimental results and analysis of the experiments are illustrated in Section 4. Finally, conclusions are drawn and future work is presented in Section 5.

2. Overview of the Literature

Texture can be described as local intensity variation in a local region of a digital image from pixel to pixel (Tomita & Tsuji, 2013). Texture features have played a significant role in many fields of research, such as pattern recognition, remote sensing and medical imaging (Nanni et al., 2010; Srinivasan & Shobha, 2008; Van de Wouwer et al., 1999; Wu et al., 2016). Texture-based feature extraction approaches are broadly classified into four main categories: statistical, transform, model, and structural-based approaches.

In the statistical-based approach, the spatial distributions of pixels and relationships between the pixel grey values in an image are computed as local features at the pixel level. Considering the pixel level, statistical-based methods are then categorised into first, second, and higher orders according to the number of pixels considered, to compute the local features (Tomita & Tsuji, 2013). In first-order statistical texture feature extraction, only the properties of each pixel value are estimated. However, in the second-order, and higher-order, the spatial interaction and relationship between two or more pixel values at definite locations are estimated (Srinivasan & Shobha, 2008).

The transform-based approach converts the image into a new form by considering the pixel intensity variations and the spatial frequency properties of the input image (Bharati et al., 2004).

The model-based approaches are able to generate an empirical model of each pixel of an image. The variation of pixel elements of texture produces a set of parameters considered to describe an image model (Bharati et al., 2004). Popular model-based approaches used for texture analysis include Markov random fields, fractal, and autoregressive models (Gonzalez & Woods, 2005; Bharati et al., 2004; Patil et al., 2014).

In the fourth category, the structural-based approach, texture features are characterised by texture primitives or texture elements, such as autocorrelation, edge, and morphological operations. In other words, the texture is defined as the property of texture elements and a placement rule (Tomita & Tsuji, 2013). The structural-based texture features are more useful for synthesis than analysis tasks and they have a limitation in practical uses (Gonzalez & Woods, 2005).

A detailed discussion on different categories of texture-based feature extraction approaches is provided in the subsequent subsections.

2.1. Statistical-based Approaches

In this study, statistical-based feature extraction methods are divided into two groups for better description and comparison. The first group is particularly useful for extracting texture features in a given direction (horizontal, vertical, or diagonal). By considering only a single direction, some information regarding the neighbouring pixels in other directions is lost (Conners & Harlow, 1980; Haralick et al., 1973). Therefore, these methods, such as the grey-level co-occurrence matrix, grey-level run-length method, grey-level difference method, and grey-level texture co-occurrence spectrum, are not rotation invariant. In the second group of the statistical-based approach, the

relationship between a pixel and its neighbour pixels is taken into account. These non-parametric methods summarise the local grey-level information of the image and provide a histogram for the retrieval process. In contrast to the first group, the methods in the second group are rotation invariant. Local binary patterns, median binary pattern, improved local binary pattern, fast-local binary pattern, local ternary pattern, improved local ternary patterns, binary gradient contour, complete local binary pattern, centre-symmetric local binary pattern, and improved centre-symmetric local binary pattern methods are classified as the second group.

2.1.1. Grey-level Co-occurrence Matrix

Grey-level co-occurrence matrix (GLCM) is one of the well-known second-order statistical texture features in the literature. In the GLCM, information regarding the frequency of occurrence of two neighbouring pixels in an image is extracted (Haralick et al., 1973). The GLCM is computed based on two terms: the displacement d of neighbour pixels, and the orientation θ between neighbour pixels. The orientations can be horizontal, vertical, and diagonal. A spatial-dependence probability-distribution matrix, where the number of rows and columns is based on the number of grey-levels in the image, is then generated. From each matrix, up to fourteen textural features can be extracted (Haralick et al., 1973). However, only energy, entropy, contrast, and homogeneity, as four features, are generally extracted from the input image. In the extracted features, the probability of occurrence of a pair of grey-levels (i,j) at the distance d with the direction θ is described by a function $f(i,j|d,\theta)$ in the image. For a statistically reliable estimation of the relative frequency, a sufficiently large number of occurrences for each event is required.

2.1.2. Grey-level Run-Length Method

In the grey-level run-length method (GLRLM), the number of length varieties of grey-level runs is calculated. A set of linearly adjacent pixels, which have the same grey-level value, is called a grey-level run and the number of pixels in a run is the length of the run (Conners & Harlow, 1980). In the GLRLM, $R(\theta) = [\mathrm{r}'(i,j|\theta)]$ for a given image is defined as the number of runs with pixels of grey-level i and run-length j in an angle θ direction. For an image, grey-level run-length matrices $R(\theta)$ can be calculated in four directions where $\theta = 0^{\circ}, 45^{\circ}, 90^{\circ}, 135^{\circ}$. Each matrix can provide seven features: short-run emphasis, long-run emphasis, grey-level non-uniformity, run percentage, run-length non-uniformity, low grey-level run emphasis, high grey-level run emphasis.

2.1.3. Grey-level Difference Method

A certain difference between pairs of grey-level pixels in the image is defined as the grey-level difference method (GLDM) (Conners & Harlow, 1980). By considering the image intensity, a function, $f(x,y)$ is computed for any given displacement $\delta = (\Delta x, \Delta y)$, where Δx and Δy are integer values of displacements in horizontal and vertical directions, respectively. The grey-level difference between the pair of pixels is then calculated by:

$$f_{\delta}(x,y) = |f(x,y) - f(x+\Delta x, y+\Delta y)| \quad (1)$$

In this method, the vector δ can be considered in one of the following forms, $(0,d),(-d,d),(d,0),(-d,-d)$, where d is the inter-sample spacing distance. The probability density functions of four different displacement vectors are produced in order to measure the texture features. The texture features, such as contrast, entropy, mean, angular second moment, and inverse difference moment, are extracted based on the probability density function (Conners & Harlow, 1980; Haralick et al., 1973; Kim & Park, 1999).

2.1.4. Grey-level Texture Co-occurrence Spectrum

The probability of the possible occurrence of grey value pixels while sorting in descending order, are taken into account by the grey-level texture co-occurrence spectrum (GLTCS) (Patel & Stonham, 1992). The grey value pixels of four different orientations (horizontal, vertical, and the two diagonals) are considered to map n-points from the texture image. The group of n-points in the image, which are called an n-tuple, are used to extract the textural information. The nearest neighbours around the centre pixel in the four directions are considered for the n-tuple, which is sorted into a descending order and represents one state of the n-tuple as a rank. The occurrence of ranks in the entire image are recorded to form an $n!$ dimensional rank. The mathematical formulation which is used in this study is as follows:

$$f_{GLTCS}(x) = \sum_{i=1}^{3} \sum_{j=i+1}^{4} i!\, \varepsilon\left[I_{\pi^{-1}(j)} - I_{\pi^{-1}(i)}\right] \quad (2)$$

2.1.5. Local Binary Pattern

The Local binary pattern (LBP) method summarises the local grey-level structure of an image (Ojala, Pietikäinen, & Mäenpää, 2002). In the LBP method, an image is divided into a number of overlapping patches. The features are extracted based on the differences between the centre pixel C and the neighbour pixels N, which are placed in a patch indicated by a circle with a radius R. In different LBP methods, different patch sizes are considered. Patch size is defined based on the values of (N, R), where $\mathrm{N} = \{4, 8, 12, 16, 24\}$ and $\mathrm{R} = \{1, 1, 1.5, 2, 3\}$, correspondingly. In each patch, the centre pixel value of the patch is a threshold for that specific patch and a binary pattern is obtained according to the pixels around the centre pixel. When a limited number of transactions occur in a binary pattern, the binary pattern is called uniform. The number of transactions less than or equal to two as compared to the uniform one, is an important concept in the LBP method. Thus, the central pixel receives the binary valued result as a local image descriptor. The $\mathrm{LBP}_{\mathrm{N,R}}$ produces 2^{N} output values that correspond to the different binary patterns according to the number of neighbours. The LBP operator can be defined as the following:

$$LBP_{N,R} = \sum_{n=0}^{N-1} 2^n \times s(i_n - i_c) \quad (3)$$

$$s(i_n - i_c) = \begin{cases} 1 & if\ (i_n - i_c) \geq 0, \\ 0 & otherwise \end{cases} \quad (4)$$

where i_c and i_n are the grey-level values of the central pixel *c* and its *n* neighbours, respectively. Let S represent a matrix of 3×3 which shows the indexed neighbour pixels. The LBP method has low computational complexity and less sensitivity to changes in illumination, and is also a rotation-invariant method.

$$\mathrm{S} = \begin{bmatrix} i_7 & i_6 & i_5 \\ i_0 & i_c & i_4 \\ i_1 & i_2 & i_3 \end{bmatrix}$$

2.1.6. Median Binary Pattern

The median binary pattern (MBP) is another form of the LBP feature extraction technique (Hafiane, Seetharaman, & Zavidovique, 2007). In the MBP operator, the median value of each patch is considered as a threshold for the patch. The local binary pattern is then obtained by comparing all the pixels in a patch to their median value. The MBP method is invariant to monotonic grey-scale changes, defined as follows:

$$MBP = 2^N \times s(i_c - M) + \sum_{n=0}^{N-1} 2^n \times s(i_n - M) - 1 \quad (5)$$

$$s(i_{c,n} - M) = \begin{cases} 1 & if\ (i_{c,n} - M) \geq 0, \\ 0 & otherwise \end{cases} \quad (6)$$

where M is the median value over a 3×3 neighbourhood. In the MBP method, images are divided into the overlapping patches of 3×3, resulting in $2^9 - 1$ binary patterns being produced. The MBP operator divides the intensity values in each patch extracted from the image into two groups according to the median value. The method then captures the contrast between the two intensity ranges that impact the local structure.

2.1.7. Improved Local Binary Pattern

The improved local binary pattern (ILBP) is a modification of the conventional LBP texture feature extraction method. Since in the LBP method the local structure is missing under certain circumstances, the ILBP method is proposed to address this problem and to reduce the effect of noise. The term 'improved' is used since a weight is given to the central pixel (Jin, Liu, Lu, & Tong, 2004). The average of the intensity values in a patch of size 3×3 is obtained and considered as the threshold value of the patch. The grey-level neighborhood pixel values are then thresholded based on the local average grey-level value. The definition of the ILBP is provided in the following.

$$ILBP = 2^N \times s(i_c - V) + \sum_{n=0}^{N-1} 2^n \times s(i_n - V) - 1 \quad (7)$$

$$s\,(i_{c,n} - V) = \begin{cases} 1 & if\ (i_{c,n} - V) \geq 0, \\ 0 & otherwise \end{cases} \quad (8)$$

where V is the average value of a specific patch. This operator produces $2^9 - 1$ binary patterns.

$$V = \frac{1}{9}(i_c + \sum_{n=0}^{7} i_n) \quad (9)$$

2.1.8. Fast-local Binary Pattern

The fast-local binary pattern (F-LBP) method includes F_1LBP and F_2LBP (F. Alaei et al., 2017). The neighboring pixels in a 3×3 patch size are divided into two categories. The first category contains a central pixel as well as the pixels located in the horizontal and vertical positions in relation to the central pixel. The second category contains a central pixel as well as the pixels located in the diagonal and off-diagonal positions of the central pixel. The diagram of F_1LBP and F_2LBP is illustrated in Figure 1.

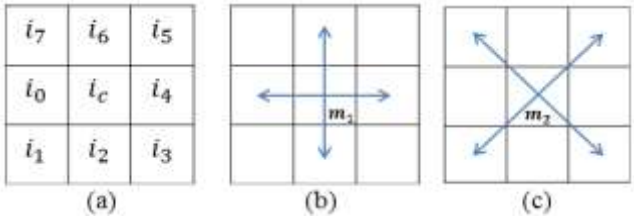


Figure 1. (a) Arrangement of a 3×3 matrix, (b) layout of the F_1LBP, (c) layout of the F_2LBP

For computing the F_1LBP initially, the mean value of the pixels in the first category is obtained using equation (14). The mean value is considered as a threshold in the F-LBP method. Then, a binary pattern is obtained by finding the differences between the threshold value and the pixels located in the horizontal and vertical positions of the centre pixel, using equation (12). Finally, the corresponding decimal number of the binary value is obtained using equation (10). A histogram of 15 bins represents the occurrence frequency of the different binary patterns obtained from the input

image. Consequently, to compute the F_2LBP, the mean value of the pixels in the second category is obtained. To obtain another histogram of 15 bins, similar to the F_1LBP operator, the same procedure is applied using equations (11), (13), and (15). To obtain the F-LBP feature of size 30, two histograms F_1LBP and F_2LBP are concatenated in the final step.

$$F_1LBP = \sum_{n=0}^{n\in even} 2^n \times s(i_n - m_1),\ n < 8 \tag{10}$$

$$F_2LBP = \sum_{n=0}^{n\in odd} 2^n \times s(i_n - m_2),\ n < 8 \tag{11}$$

$$s(i_n - m_1) = \begin{cases} 1 & if\ (i_n - m_1) \geq 0 \\ 0, & otherwise \end{cases} \tag{12}$$

$$s(i_n - m_2) = \begin{cases} 1 & if\ (i_n - m_2) \geq 0 \\ 0, & otherwise \end{cases} \tag{13}$$

$$m_1 = \frac{1}{5}(i_c + \sum_{n=0}^{n\in even<8} i_n) \tag{14}$$

$$m_2 = \frac{1}{5}\left(i_c + \sum_{n=0}^{n\in odd<8} i_n\right) \tag{15}$$

2.1.9. Local Ternary Pattern

The Local ternary pattern (LTP), as an extension of the LBP, is a hybrid between the texture spectrum and local binary patterns (Tan & Triggs, 2010). The LTP method is less noise sensitive compared to the LBP method, although it is less invariant to grey-level transformation. In the LTP method, a user-defined threshold *t* is considered along i_c. Therefore, $(i_c \pm t)$ illustrates the tolerance interval and i_n can take one of the following values based on the following function:

$$s(i_n, i_c, t) = \begin{cases} 1 & i_n \geq i_c + t \\ 0 & |i_n - i_c| < t \\ -1 & i_n \leq i_c - t \end{cases} \tag{16}$$

In the LTP method, uniformity is also an important property. For simplicity of process in the LTP method, ternary patterns are grouped into positive and negative patterns, as shown in Figure 2. Each set of binary patterns, extracted in positive and negative patterns, acts as an individual LBP method. An individual histogram is then computed from each set of binary patterns and the results are finally concatenated to create the feature set.

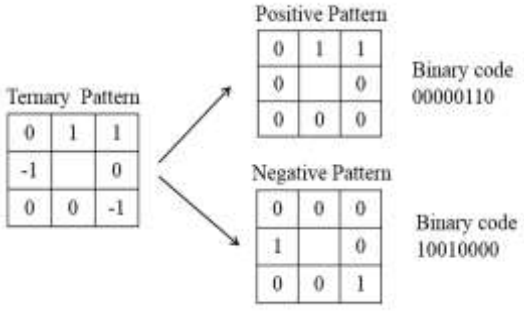


Figure 2. One example of a ternary pattern split into a positive and a negative pattern

2.1.10. Improved Local Ternary Patterns

By combining the LTP and ILBP methods, a new variation of LBP-based feature extraction, called 'improved local ternary patterns' (ILTP), is proposed in (Nanni et al., 2010). In the ILTP method, the neighborhood pixels (i_n) are thresholded by their average grey-scale values. They are

then split into positive and negative patterns similar to the LTP method to construct two sets of binary codes. The rest of the feature extraction process is similar to the LTP method.

2.1.11. Binary Gradient Contours

Intensity invariance plays an important role in the binary gradient contour (BGC) feature extraction method (Fernández et al., 2011). The pair of pixels of closed paths around the central pixel are considered for measuring the gradient in a patch of size 3×3. Three different paths are considered for obtaining the gradient. These paths are called single-loop, double-loop, and triple-loop (Fernández et al., 2011). The pixel order for a single-loop (BGC1) is described as $(i_0, i_7, \dots, i_1, i_0)$. The pixel order in the case of triple-loop (BGC3) is described as $(i_0, i_5, i_2, i_7, i_4, i_1, i_6, i_3, i_0)$. Since, eight components cannot be 0's at the same time, $2^8 - 1$ gradient binary contour features are generated in both of the operators. On the other hand, two closed paths of $(i_1, i_7, i_5, i_3, i_1)$ and $(i_0, i_6, i_4, i_2, i_0)$ are created in double-loop (BGC2) patterns. In each path, $2^4 - 1$ gradient binary contour features are generated, and as a result, the combination of two loops produces $(2^4 - 1)^2$ bins.

2.1.12. Completed Local Binary Pattern

In the completed local binary pattern (CLBP) (Zhenhua et al., 2010) approach, binary patterns generated by the original LBP are transformed to a sign and a magnitude vector. By considering this method, 2^8 features are extracted from an image. The method is denoted by:

$$f_{CLBP} = \sum_{n=0}^{N-1} 2^j \varepsilon\left(\lfloor i_n - i_c \rfloor - \tilde{I}\right) \tag{17}$$

where the average value of the difference between the central grey value and a pixel in the periphery is represented by $\tilde{I}$, and $i_n$and i_c are pixels in a patch and the centre pixel, respectively.

2.1.13. Centre-symmetric Local Binary Pattern

The centre-symmetric local binary pattern (CSLBP) (Heikkilä et al., 2009) is associated with the LBP method, although different structures of neighbourhood pixel comparisons are considered. In the CSLBP method, instead of considering the central pixel as a threshold, the following centre-symmetric couples of pixel values, $(i_0, i_4), (i_1, i_5), (i_2, i_6),$ and $(i_3, i_7),$ are considered to extract features. Hence, only 2^4 local binary patterns are extracted from the CSLBP method. The CSLBP function can be stated in the following equation:

$$f_{CSLBP} = \sum_{n=0}^{3} 2^n \varepsilon\left(i_n - i_{n+4} - \Delta - 1\right) \tag{18}$$

where parameter Δ thresholds the grey-level differences and i_n are pixel intensities in a patch.

2.1.14. Improved Centre-symmetric Local Binary Pattern

Two different varieties of the CSLBP known as improved centre-symmetric local binary pattern are proposed in (Heikkilä et al., 2009; X. Wu & Sun, 2009). In the first improved version (D-LBP), the grey value of the centre pixel is included in each of the four pairs used in the CSLBP. So, the four triplet values that are located in the horizontal, vertical, diagonal, and off-diagonal directions in each patch are considered. The following function represents the mathematical notation for the D-LBP method.

$$f_{D-LBP} = \sum_{n=0}^{3} 2^n s\left(i_n, i_c, i_{n+4}\right) \tag{19}$$

The second improved version of the CSLBP method (ID-LBP) is very similar to the first one, but the average value of the eight neighbour pixels ($\hat{V}$) is computed and used instead of the central pixel value. The function representing the ID-LBP is shown as follows.

$$f_{ID-LBP} = \sum_{n=0}^{3} 2^n s\left(i_n, \hat{V}, i_{n+4}\right) \tag{20}$$

The improved centre-symmetric local binary pattern methods are also encoded based on the following function:

$$s(x_1, x_2, x_3) = \begin{cases} 1, & if\ x_1 \geq x_2\ and x_2 \geq x_3 \\ 1, & if\ x_1 \leq x_2\ and x_2 \leq x_3 \\ 0, & otherwise \end{cases} \tag{21}$$

2.2. Transform-based Approaches

The capabilities of space-frequency decomposition in transform-based texture feature extraction approaches have been considered in many applications of image and signal processing (Mehri et al., 2017). In this section, some frequently used transform-based texture feature extraction approaches are reviewed and considered for document image retrieval.

2.2.1. Wavelet Transform

Wavelet transforms are based on small waves called wavelets. Wavelet transforms are able to represent an image at multiple resolutions based on the desired frequency. The purpose of these multiple resolution images is to consider some of the ignored features, which are not noticeable in different resolutions and provide various impressions of an image in other resolutions. The Haar transform is the simplest known orthonormal wavelet (Haar, 1910). A wavelet is defined by a function $\psi(x) \in L^2(R)$ where $L^2(R)$ satisfies $\int |f(x)|^2\, dx < \infty$, and has limited length and varying frequencies. In order to extract features by the wavelet transform, one of the necessities is to calculate the coefficient distribution over the mother wavelet (Brigham & Morrow, 1967; Mallat, 1989; Vargas et al., 2011). The mother wavelet $\psi(x)$ translated by *u* and scaled by *s* is represented as:

$$\psi_{u,s}(x) = \frac{1}{\sqrt{s}} \psi\left(\frac{x-u}{s}\right) \tag{22}$$

Since the continuous wavelet transform $W_\psi(u, s)$ is not useful in practical scenarios, discrete wavelet transform (DWT) coefficients are considered to extract texture features.

$$W_\psi(u, s) = \int_{-\infty}^{+\infty} f(x)\, \psi_{u,s}(x) dx \tag{23}$$

$$u = (k - l), s = (k - 1) \times 2,\ k = 1, 2 \tag{24}$$

These wavelets are signified by ψ_{lk} and identified by indices l and k. To discretise the wavelet transform, the parameters *s* and *u* are used. To calculate the wavelet coefficients of a discrete signal, all integrals are substituted by sums. The two-dimensional DWT is computed by applying a separate filter bank to the image (Mallat, 1989);

$$L_n(u_i, u_j) = \left[H_x * \left[H_y * L_{n-1}\right]_{\downarrow 2,1}\right]_{\downarrow 1,2} (u_i, u_j) \tag{25}$$

$$D_{n1}(u_i, u_j) = \left[H_x * \left[G_y * L_{n-1}\right]_{\downarrow 2,1}\right]_{\downarrow 1,2} (u_i, u_j) \tag{26}$$

$$D_{n2}(u_i, u_j) = \left[G_x * \left[H_y * L_{n-1}\right]_{\downarrow 2,1}\right]_{\downarrow 1,2} (u_i, u_j) \tag{27}$$

$$D_{n3}(u_i, u_j) = \left[G_x * \left[G_y * L_{n-1}\right]_{\downarrow 2,1}\right]_{\downarrow 1,2}(u_i, u_j) \tag{28}$$

where *H* and *G* are low pass and high pass filters, and '*' symbolises the convolution operator, ↓2,1 (↓1,2) represents subsampling along the rows (columns), and L_n is denoted as the low-resolution image at scale n.

Three detailed coefficient matrices (horizontal, vertical, and diagonal) are generated, in addition to an approximation coefficient matrix that contains a coarse summary of the original image, by employing the equations (25-28). By applying this process iteratively to the approximation images, a hierarchy of these images is generated. The grey-level variation or functional variation intensity of the images in a different direction is measured using three detail images. An important feature of the detail images D_{ni} produced by filtering in a specific direction at scale n is their directional sensitivity. Hence, in order to extract features from document images, these under-sampled images are more relevant for our purposes. The variances of each matrix are extracted column-wise and concatenated to construct a feature vector. Furthermore, energy from each DCM is also computed and added to the feature vector.

2.2.2. Fourier Transform

The Fourier transform is an image-processing tool which represents the frequency domain information of an image (Brigham & Morrow, 1967) and enables an image to be decomposed into its sine and cosine components. Some correspondence exists between the image features of the spatial domain and those on a frequency domain. Furthermore, the images transformed by employing the Fourier transform method include different spectral texture patterns with different densities that can construct discriminative texture features.

The Fourier transform is considered in the polar coordinate system (r, θ), where r is the radius to show the intensity of the image, and θ is the angle to show the direction of the image (Li, Zhang, & Xu, 2002). Let I and I' be the image under a right-angle coordinate system and the image under the polar coordinate system, respectively. The given input image is then transformed from the right-angle coordinate system into the polar coordinate system using the following equations.

$$I'(r, \theta) = I(64 + r\cos\theta, r\sin\theta),\ 0 \le r \le 64,\ 0 \le \theta \le \pi \tag{29}$$

$$R_i = \sum_{\theta=0}^{\pi} \sum_{r=8(i-1)}^{8i} I'(r, \theta),\ i = 1,2,\dots,8 \tag{30}$$

$$\theta_i = \sum_{\theta=0(i-1)}^{i} \sum_{r=0}^{64} I'\left(r, \frac{\theta\pi}{8}\right),\ i = 1,2,\dots,8 \tag{31}$$

To demonstrate the intensity of the image, a sequence of circles with the same centre is created, and the frequency domain image is then divided into small fragments by those circles. Energy in each fragment can be computed using equation (29). Meanwhile, for demonstrating the direction of the image, a sequence of lines that go through the centre of the image is drawn to obtain the frequency domain image from each of those small fragments.

2.2.3. Gabor Wavelet Transform

In the Fourier transform, information regarding time is lost, and it is not easy to say where the exact accordance with the frequency is. However, the Gabor function simultaneously provides a resolution in both the time and frequency domains (Hangarge et al., 2016; Lee, 1996). The Gabor wavelet is an ideal basis to extract local features and it can capture directional energy features from an image (Hangarge et al., 2016). The Gabor wavelet method is efficient in describing the visual

properties of an image and is similar to human visual perception (Hangarge et al., 2016). By characterising the spatial frequency structure in an image, spatial relations with the same time are provided.

By considering the 2-D mother wavelet (equation 22) and Fourier transform, the Gabor wavelet can be defined as:

$$G_{lk}(x,y) = \sum_s \sum_u I(x-s, y-u)\psi^*_{lk}(s,u) \quad (32)$$

where s and u stand for the sizes of the filter mask and ψ^*_{lk} is the complex conjugate of ψ_{lk}. By considering the Gabor wavelet for feature extraction, the transformed coefficients are used for compressed representation or a distance measure. By considering different scales and orientations, the Gabor wavelet can produce a number of filters. Commonly, forty filters are produced using five scales and eight orientations. Then, the mean, energy, and variance are extracted from each filtered image and the features are concatenated to create a Gabor feature vector. The multi-resolution and multi-orientation properties of the Gabor wavelet transform make it an efficient method for feature extraction.

2.2.4. Radon Transform

To calculate the projection of an image using a given set of angles, the Radon transform (RT) method is applied (Beylkin, 1987). The sum of the intensities of the pixels in each direction can provide the projection as a line integral. Directions with more straight lines are defined as texture principal directions, and the variance of the projection is a local maximum in this direction (Jafari-Khouzani & Soltanian-Zadeh, 2005). Applying the Radon transform on an image provides a new image $R(r,\theta)$, defined as follows:

$$R(r,\theta)f[(x,y)] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\,\delta(r - x\cos\theta - y\sin\theta)dxdy \quad (33)$$

where the distance of a line from the origin is described by $r = x\cos\theta + y\sin\theta$, and θ is the angle between the lines. The mean, variance, vertical projection, and horizontal projection are extracted from the Radon transform of each angle $\theta(0{,}180)$, and then concatenated to create a feature vector. This method is robust against illumination changes across the image, as the low frequency components of the variance are removed by extracting the derivation (Jafari-Khouzani & Soltanian-Zadeh, 2005).

2.2.5. Contourlet Transform

The Contourlet transform is a directional multiresolution image representation obtained by extending the wavelet transform, considering multi-scale and directional filter banks (Katsigiannis et al., 2010). As mentioned earlier, by applying the conventional wavelet transform, only horizontal, vertical, and diagonal information can be defined. However, other directional information about the edge remains unidentified. Thus, the Contourlet transform has been proposed as a directional multiresolution image representation that is capable of capturing and representing object boundaries in an image (Do & Vetterli, 2002). To detect the point discontinuities of the image, the Laplacian pyramid is considered, and to link the point discontinuities into linear structures, the directional filter bank is taken into account (Katsigiannis et al., 2010). In each level of the Contourlet filter bank, the Laplacian pyramid generates a down-sampled low-pass and a band-pass version of the image. Then, the directional filter bank is employed on the band-pass image. This process is iterated on the low-pass image to obtain the most relevant texture information (Do & Vetterli, 2002). A set of statistical features, such as mean, energy, contrast, information entropy, and homogeneity is extracted from

each sub-band (Do & Vetterli, 2002; Katsigiannis et al., 2010) to create a feature vector. One of the main weaknesses of the Contourlet transform is that its original image is not localised in the frequency domain (Lu & Do, 2006).

2.2.6. Gist Descriptor

The Gist descriptor was considered for image matching and the retrieval process (Oliva & Torralba, 2001). Summarised gradient information (scales and orientations) of different parts of an image is defined as the Gist of the image. The input image is passed through Gabor filters at 4 scales and 8 orientations. So, 32 feature maps of the same size are generated from the input image. Each one of these feature maps represents one orientation in one scale of the input image. Each map is then divided into a set of subregions of a fixed size grid of 4×4. To represent the feature, average intensity is calculated in each subregion. Finally, 16 averaged values of all 32 feature maps are concatenated, resulting in a Gist feature set of 512 (=16×32) features.

2.3. Model-based Approaches

The aim of model-based texture feature extraction approaches is to interpret an image texture by generating images and stochastic models. The main concern of this type of approach is the computational complexity of estimating the parameters of stochastic models (Materka & Strzelecki, 1998). Moreover, the model-based approaches miss orientation selectivity and are not appropriate for describing local image structures. Three main methods of model-based approaches are discussed in the following subsections.

2.3.1 Markov Random Fields

In texture-based features, neighbouring pixels have a significant impact on certain pixels. Estimating a joint posterior probability is one of the conventional approaches for considering the neighbourhood pixel information (Chellappa & Chatterjee, 1985). Markov random field (MRF) modeling (Blake et al., 2011) formed by the Markov property, is a graphical model of a joint probability distribution. The task of MRF is to detect a suitable intensity distribution for a given image. This suitability can be used as a texture component for texture-based image analysis. The Markov model represents the relationship between a few pairs of pixels that share an edge (Blake et al., 2011; Kato et al., 1999).

A commonly used class of MRF is the class that can be factorised according to the maximal cliques of a graph. The clique potentials can be estimated from a given image or by random variables where their joint distribution is a mixture of distributions. The conditional densities that can describe the MRF model (Chellappa & Chatterjee, 1985; Blake et al., 2011) are of the form:

$$p(y(s)\ all\ y(s+r), r \in N) \tag{34}$$

where N is a symmetric neighbour set, and $y(s)$ is a Gaussian distribution function of the neighbours s in all 8 directions, which is also known as the intensity $y(s)$ at site s. The MRF model includes a set of unknown parameters. The texture $[y(s), s \in \Omega, \Omega = \{s = (i,j): 0 \leq i,j \leq M-1\}]$ is a Gaussian distribution, which assumes to have a zero mean, and follows the difference equation as below:

$$y(s) = \sum_{r \in N_s} \omega_r \big(y(s+r) + y(s-r)\big) + e(s) \tag{35}$$

$$E\big(e(s)\ e(r)\big) = 0,\ (s-r) \in N \tag{36}$$

By considering the Gaussian, $E(e(s) \mid \text{all } y(r),\ r \neq s) = 0$ indicates $p(y(s) \mid \text{all } y(r), r \neq s) = p(y(s) \mid \text{all } y(s+r), r \in N)$. The given equation indicates that $y(.)$ is a Markov model with respect to neighbour set N. The unknown parameter can be estimated by least squares error as below:

$$\omega = [\sum q(s)q^t(s)]^{-1}[\sum q(s)y(s)] \tag{37}$$

$$v = \frac{1}{M^2}\sum[y(s) - \omega^t q(s)]^2 \tag{38}$$

$$q(s) = col.[y(s+r) + y(s-r), r \in N] \tag{39}$$

The iterative conditional estimation (ICM) is considered to maximise the posteriori probability of the label field and realisations of ω. Various textures can be obtained using the labelled/segmented image. In our experiments, the energy of all pixels in the labelled image was computed and considered as a features vector. As in this research work images were resized/normalised into 140×90 based on the smallest sample size of the MTDB dataset, an MRF feature set composed of $12600\ (= 140 \times 90)$ features was extracted and considered for document image retrieval.

2.3.2 Fractal Dimension

Fractal theory is based on geometry and dimension, and fractal dimension is an important characteristic of fractals, as it includes information about their geometric structure. In the field of image analysis, the fractal dimension has been used to estimate the complexity of the texture of images (Costa et al., 2012; Schouten & de Zeeuw, 1999). The fractal dimension measurements are also considered to define the boundary complexity of objects in an image (Costa et al., 2012). Fractal dimension (Fisher, 2012) is calculated based on image self-similarity. The method works similarly to the copy machine metaphor, which is also called an affine transformation. In this method, the image is reconstructed by repeating the transformed image. In other words, a piece of an image can be approximated by other parts of a transformed image. So, the feature can be extracted by considering the image self-similarity. The fractal encoder divides an image into non-overlapping patches of size $\epsilon \times \epsilon$, called a range block, which searches for another block in the image known as a domain block that look like the range block under an affine transformation. The most common fractal dimension is Hausdorff's fractal dimension F_D, which is computed by the following expression:

$$F_D = \lim_{\epsilon \to 0} \frac{\log N(\epsilon)}{\log \epsilon^{-1}} \tag{40}$$

where $N(\epsilon)$ represents dimension counting. It is possible to generate a log $\bar{N}(\epsilon)$ inverse $\log N(\epsilon)$ curve when changing the ϵ value. Finally, a line-fitting method is taken into account to approximate this curve by a straight line. The slope of this line corresponds to the Fractal dimension F_D.

2.3.3. Autoregressive Model

Autoregressive models have been applied for textures analysis, texture classification, and segmentation in the literature (Joshi et al., 2009). In an autoregressive model, the hypothesis is that current time series values might be influenced by past values from the same series. Practically, one or two lagged statements are adequate to estimate the current observation. However, in theory, current time series value may be determined by a large number of past observations. The autoregressive model is related to a linear regression model and can predict future behaviour based on past behaviour. In order to characterise texture, the autoregressive model provides a linear estimation of a grey-level pixel and its grey-level neighbourhood. The autoregressive process can be expressed as:

$$x_t = b + \emptyset_1 x_{t-1} + \emptyset_2 x_{t-2} + \cdots + \emptyset_p x_{t-p} + a_t \tag{41}$$

where $x_{t-1}, x_{t-2}, \dots, x_{t-p}$ stand for the past values, and $\emptyset_1, \emptyset_2, \dots, \emptyset_p$ are lags' coefficients, and a_t represent an uncorrelated process with a mean of zero. The parameter p is the number of periods that affect the time series. The value b is defined by the following equation:

$$b = \left(1 - \sum_{i=1}^{p} \emptyset_i\right)\mu. \tag{42}$$

To estimate the coefficients of an autoregressive process, the covariance method is used. The covariance method minimises the forward prediction error and can find a suitable pth-order autoregressive for an image. As a result, 90 features are extracted from each image for the document image retrieval process.

2.4. Structural-based Approaches

In structural texture feature extraction approaches, the texture of images is characterised by primitives and a hierarchy of spatial arrangements of those primitives (Materka & Strzelecki, 1998). The structural approach mostly gives an emphasis to the shape aspects of the tonal primitives. The structural approaches can also provide a good symbolic description of an image. However, in the structural approaches, attaining a description and representation of primitives is challenging (J. Chen & Jain, 1988). It should also be noted that extracted features using structural approaches are more useful for synthesis than for analysis tasks (Materka & Strzelecki, 1998). Nonetheless, to see the effectiveness of structural approaches for the document image retrieval process, three commonly used methods are reviewed in the following subsections.

2.4.1. Auto-correlation Function

The auto-correlation function (ACF) was considered as a texture-based primitive extraction method for retrieving the scene images (Heilbronner, 1992). The ACF describes the movement of an image with respect to itself in all probable directions where the image is correlating with itself under the specific conditions. In the auto-correlation function, the shape and arrangement of texture-primitives can be established from the locations of peaks. In other words, the ACF is applied to characterise the periodic properties of an image. If x is a stochastic process, then the auto-correlation between times $s$and t can be represented as follows:

$$R_{(t,s)} = \frac{\mathrm{E}(x_t - \mu_t)(x_s - \mu_s)}{\sigma_t \sigma_s} \tag{43}$$

where E is the expected value operator. The significant point of auto-correlation is that the ACF is a directly measurable quantity. In the case of well-defined autocorrelation, the R value must be in the range [−1, 1], where 1 indicates perfect correlation.

In this research study, the auto-correlation function is applied on document images to extract texture-primitives. The generalised Hough transform is then employed to extract texture features from peaks. As a result, a feature vector of 50 dimensions is created for each image to be considered for DIR.

2.4.2. Edge Detection

One of the techniques to extract suitable structural information from an image is edge detection (Dixit & Shirdhonkar, 2018). Edge detection methods are used to intensely reduce the amount of data and keep only the edges of an image that preserve some texture information of the image. In this study, Canny edge detection, as one of the best edge detection methods in the literature (Canny, 1986), is considered for experimentation. Canny edge detection is composed of five steps, as follows:

i. Gaussian filter is applied to smooth the image and to remove noise.

$$g(x, y) = G_\sigma(x, y) \times I(x, y) \tag{44}$$

$$G_\sigma = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \quad (45)$$

ii. The intensity gradients of the image are determined using:

$$X(x,y) = \sqrt{g_x^2(x,y) + g_y^2(x,y)} \quad (46)$$

$$\theta(x,y) = tan^{-1}\left[g_y(x,y)/g_x(x,y)\right] \quad (47)$$

iii. Non-maximum suppression is applied to clear the forged response for detecting edges, and it attempts to find the pixel with the maximum value in an edge.

iv. The double threshold is applied to define the potential edges. The pixels with a high value are most likely to be edges.

v. In the final step, all the other edges that are weak and not connected to strong edges, are destroyed and true edges in the image remain.

After applying edge detection on document images, the Fourier transform method is taken into account to create a feature vector of size 50 features for the document image retrieval process.

2.4.3. Morphological Operation

The morphological operation is a collection of nonlinear processes that can provide some shape and texture information of images. Details smaller than a certain reference shape, called a structuring element, will be removed from the images using a morphological process (Daisy et al., 2012). By considering this method, the value of each pixel in the output image is the result of a comparison between a certain pixel with its neighbourhood pixels in the input image. Morphological operations are usually performed on binary images. Four primary operations of mathematical morphology are: dilation, erosion, opening, and closing. Top-hat filtering and down-hat filtering are two other operations of mathematical morphology.

The dilation of an image I by a structuring element $B(m,n)$ is defined by:

$$(I \oplus B)_{(x,y)} = max\{I(x-m, y-n) + B(m,n)\} \quad (48)$$

The erosion of image I by a structuring element $B(m,n)$ is defined by:

$$(I \ominus B)_{(x,y)} = max\{I(x-m, y-n) - B(m,n)\} \quad (49)$$

The opening of image I by a structuring element is defined by:

$$I \circ B = (I \ominus B) \oplus B \quad (50)$$

The closing of image I by a structuring element is defined by:

$$I \cdot B = (I \oplus B) \ominus B \quad (51)$$

The top-hat filtering of image I is given by:

$$T_t(I) = I - I \circ B \quad (52)$$

The down-hat filtering of image I is given by:

$$T_d(I) = I \cdot B - I \quad (53)$$

As top-hat filtering provides better DIR results, the top-hat filtering method is considered in this research work for experimentation. After applying mathematical morphology for feature extraction, block truncation coding (BTC) is taken into account to calculate feature vectors. The original image is subtracted from the result generated by BTC, and a feature vector is created.

3. Document Image Retrieval Method

The steps involved in general document image retrieval relevant to this study are demonstrated in Figure 3. The proposed method includes two phases, training and testing. Pre-processing, feature extraction and document retrieval based on a similarity measure are the main steps in the proposed document image retrieval method.

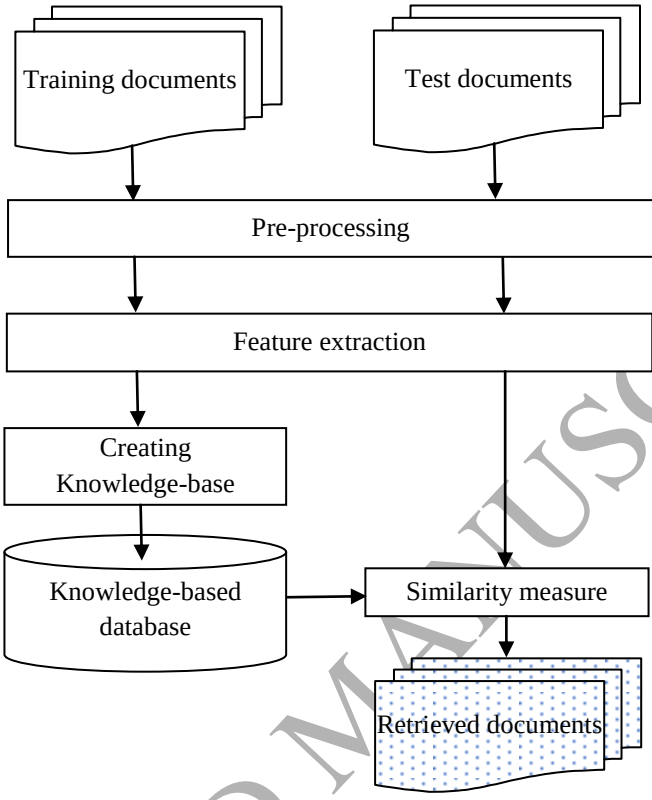


Figure 3. Block diagram of the document image retrieval system

3.1. Pre-processing

Pre-processing, including filtering, skew correction, and normalisation, is a primary step in any DIR system to prepare document images for further processing. Filtering methods are commonly employed on document images to enhance their quality or appearance. Normalisation refers to either changing the image size or normalising the features extracted from the input image (A. Alaei et al., 2016). In this research work, a mean filtering method (Griffin, 2000) is employed on binary document images. By applying a 3×3 mean filtering method 3 times, the binary images are converted to grey scale images. In addition, size normalisation is employed on the document images. Size normalisation is applied on a few texture feature extraction methods to obtain the same image size and consequently, feature vectors of a similar size. Size normalisation is performed only on the MTDB dataset, as in the MTDB dataset, every image has three different versions of it with three resolutions/sizes. Therefore, document images are resized to 140×90 pixels, which is the smallest size of images in the MTDB dataset.

3.2. Feature Extraction

Following pre-processing, in the feature extraction step, a set of features is computed to represent a document image with a feature vector. As mentioned before, in this research work twenty-six feature extraction methods from four categories of statistical, transform, model, and statistical-based

approaches are considered to characterise document images. Feature vectors extracted from document images are further stored in a database, as a knowledge-based model, for retrieval purposes.

3.3. Similarity Measures and Final Retrieval Results

In the literature, nearest neighbor-based methods were commonly used for document image retrieval and good retrieval results have been obtained for documents with complex layouts (K. S. Kumar, Kumar, & Jawahar, 2007). Therefore, in this research work, a nearest neighbor-based method is used to obtain similarities between a query image and the trained document images. To compute these similarities, different similarity measures can be considered. As in most of the cases, the City-block distance method provided better retrieval results compared to the Tanimoto, Euclidean and Cosine distances. The City-block distance (Melter, 1987) is considered in all experiments in this research work. To compute a similarity measure based on the City-block distance measure, the following equation is used:

$$d_C(A,B) = \sum_{i=1}^{L} |A_i - B_i| \tag{54}$$

where A and B are the extracted feature vectors from a query image and a document image in the training set, respectively. The value of L shows the number of features in the feature vector and $d_C(A,B)$ represents the distance between the feature vectors A and B. A_i and B_i are also the i^{th} feature in the feature vectors A and B.

By using the City-block distance, the similarity between a given query and all the trained documents are computed. The Top-n most similar images with the highest similarity to a given query are then listed. To accept or reject the listed Top-n document images in a class, a threshold τ based on the mean (μ) and standard deviation (σ) of distances (distance matrix) obtained from the trained samples is calculated as follows:

$$\tau = \mu + \gamma \times \sigma \tag{55}$$

where γ is a tuning parameter. In our experiments, we considered different small values between 0 and 2 in different methods to get the best retrieval results (F-score) from the training samples. This fixed value of γ was used in the testing phase.

4. Experimental Results and Discussion

4.1. *Document Datasets and Evaluation Metrics*

To validate the performance of different feature extraction methods for document image retrieval, three different document image datasets were used. The first dataset used for experimentation was the CLEF-IP dataset composed of 9 different classes of heterogeneous document images, including abstract drawings, graphs, flowcharts, and gene sequences (Piroi et al., 2011). The flowchart class includes only a few samples, so it was not considered in our experiments. In total, 37,771 document samples were used for the experiments.

The second dataset was ITESOFT, which includes a variety of scanned official documents. The ITESOFT dataset was manually categorised into 12 different classes, and samples were not equally distributed in all the classes. In total, 1,116 document images with a 300DPI resolution were considered for experiments.

The third dataset was the Media Team Document Database (MTDB) from the University of Oulu (Sauvola & Kauniskangas, 1999) that includes three different resolutions with a great diversity of page layouts. Among 19 different types of documents, such as articles, newsletters, advertisements,

and dictionary documents, 8 classes were omitted as those documents related to music, line drawing, and other non-text notation. In total, 1,322 document samples were used for experimentation.

The Precision (P), Recall (R) and F-score, as three commonly used metrics in the image retrieval literature, were considered as evaluation metrics. P is defined as the number of correctly retrieved document images over the number of retrieved documents, while R is defined as the number of correctly retrieved documents over the actual number of document images. The F-score is derived from the precision and recall and is defined as $F = 2 \times (P \times R)/(P + R)$. The F-score is a strict measurement as it reflects the degree of balance between precision and recall instead of only the absolute value of the precision and recall. The document images, which have maximum visual similarity to a given query, were ranked in the first (Top-1), Top-3, Top-5, and Top-10, and accordingly, the F-scores were measured.

The Top-1 value indicates the percentage of correct retrieval results at the first position with respect to a given query. Consequently, the Top-10 value represents the performance of the system when the document image retrieval system ranks the correct document image with respect to a given query in the first 10 places.

In our experiments on the three datasets, a cross-validation technique was used to randomly select 33% of the samples from each class in each round for the training, and the rest of the samples for the testing. The training and testing sets did not have any overlap for the experiments. To evaluate the performance of a feature extraction method incorporated in the DIR system, the experiments were repeated 30 times. The mean P, mean R and mean F-score were computed and reported as final results for each feature extraction method.

It is worth noting that we considered the one-way ANOVA test on F-scores obtained based on each feature extraction method to check the significance of resampling in the final retrieval results . A null hypothesis against the alternative hypothesis was considered. The p-value varied within [0.914, 0.998], [0.823, 0.964] and [0.568, 0.981] for the CLEF_IP dataset, the ITESOFT dataset, and the MTDB dataset, respectively. In the results, the null hypothesis was not rejected as $p > 0.05$ for all three datasets. Thus, resampling in each run did not have much influence on the F-scores.

4.2. *Results and Comparison Analysis*

The results obtained using different texture-based features for DIR on the CLEF_IP, ITESOFT, and MTDB datasets are shown in Tables 1, 2 and 3.

From the retrieval results on the CLEF_IP dataset shown in Table 1, it is clear that in the first group of statistical approaches, the GLTCS provided the highest F-scores of 77.74% and 95.76% in the group at Top-1 and Top-10, respectively. Among the second group of statistical approaches, variations of the BGC method (BGC1, BGC2, and BGC3) showed better retrieval performance. The BGC1 method provided the best retrieval results with F-scores of 79.27% and 96.11% at Top-1 and Top-10, respectively. By employing transform-based approaches for DIR, the Gist descriptor method provided the best F-scores of 80.87% at Top-1 and 96.50% at Top-10. Amongst the model-based approaches, the auto-regressive method provided the best F-score results of 73.75% and 94.54% at Top-1 and Top-10, respectively. In the structural-based approaches, morphology obtained better retrieval results compared to the other methods in the group. From Table 1 it can be concluded that overall, the transform-based texture features provided better document image retrieval performance on the CLEF_IP dataset compared to other categories of texture-based features. It is worth mentioning that the Gist features from the transform-based category achieved the best document

image retrieval results compared to all other texture-based features employed on the CLEF_IP dataset for document image retrieval.

Table 1. Comparison of the results obtained by four texture-based categories applied on the CLEF_IP dataset.

	Metric	**Precision**				**Recall**				**F-score**			
Category	**Method**	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)
Statistical-based approach (1st group)	GLCM	21.75	47.35	61.68	79.00	100	100	100	100	35.73	64.26	76.30	88.27
	GLRLM	33.51	60.87	72.41	84.28	**100**	**100**	**100**	**100**	50.20	75.68	84.00	91.47
	GLDM	37.14	62.36	73.73	85.58	99.97	99.96	99.98	100	54.16	76.80	84.87	92.23
	GLTCS	**65.25**	**82.37**	**87.69**	**92.87**	96.14	97.56	98.02	98.84	**77.74**	**89.32**	**92.57**	**95.76**
Statistical-based approach (2nd group)	LBP	59.85	76.68	82.91	89.37	94.21	96.97	98.21	99.10	73.20	85.64	89.91	93.99
	$LBP_{4,1}$	49.77	70.60	78.60	86.91	97.92	98.99	99.25	99.63	66.00	82.42	87.73	92.84
	$LBP_{8,1}$	52.48	72.98	80.52	88.22	97.65	98.96	99.21	99.66	68.27	84.01	88.90	93.59
	$LBP_{12,1.5}$	62.18	78.11	83.89	90.06	92.08	95.18	96.58	98.55	74.23	85.80	89.79	94.11
	$LBP_{16,2}$	64.01	78.88	84.06	89.83	92.53	94.70	95.79	97.47	75.67	86.07	89.54	93.50
	MBP	63.93	79.55	85.09	91.52	91.23	93.90	95.19	97.10	75.18	86.13	89.86	94.23
	ILBP	67.13	82.22	87.31	92.68	90.38	94.46	96.18	98.28	77.04	87.92	91.53	95.40
	F-LBP	52.78	73.20	80.70	88.34	96.94	98.40	98.79	99.33	68.34	83.95	88.83	93.51
	LTP	56.65	77.31	85.02	92.11	98.90	99.32	99.47	99.71	72.04	86.94	91.68	95.76
	ILTP	58.87	78.58	85.43	91.97	**99.25**	**99.68**	**99.81**	**99.87**	73.91	87.88	92.06	95.76
	BGC1	**69.01**	**84.99**	**89.67**	**94.12**	93.10	95.73	96.85	98.18	79.27	**90.04**	**93.12**	**96.11**
	BGC2	68.39	84.09	88.99	93.90	92.13	95.02	96.32	97.95	78.51	89.22	92.51	95.88
	BGC3	68.82	84.44	89.07	93.58	94.36	96.41	97.27	98.38	**79.59**	90.03	92.99	95.92
	CLBP	67.80	83.67	88.71	93.61	93.85	95.71	96.68	97.86	78.73	89.29	92.52	95.69
	CSLBP	54.58	74.04	81.32	88.77	97.07	98.19	98.53	99.21	69.87	84.42	89.10	93.70
	D-LBP	58.44	77.63	84.03	90.50	97.34	98.63	99.04	99.57	73.04	86.88	90.92	94.82
	ID-LBP	57.12	76.72	83.45	90.02	96.78	98.01	98.50	99.13	71.84	86.07	90.35	94.36
Transform-based approach	Wavelet	60.50	79.38	85.94	91.62	99.97	100	100	100	75.38	88.50	92.44	95.62
	Fourier	66.56	82.74	88.06	93.17	99.89	99.96	100	99.99	79.89	90.54	93.65	96.20
	Gabor	63.25	80.52	86.19	91.82	97.95	98.75	99.08	99.56	76.87	88.71	92.19	95.53
	Radon	44.95	67.96	76.89	86.05	96.33	98.32	98.64	99.25	61.30	80.37	86.42	92.18
	Contourlet	63.25	80.52	86.19	91.82	97.95	98.75	99.08	99.56	76.87	88.71	92.19	95.53
	Gist	**67.89**	**83.71**	**88.56**	**93.24**	**99.98**	**100**	**100**	**100**	**80.87**	**91.13**	**93.93**	**96.50**
Model-based approach	MRF	39.80	55.89	62.84	72.06	82.94	91.72	97.92	98.62	53.79	69.46	76.55	83.27
	Fractal	24.25	35.13	39.96	45.13	**100**	**100**	**100**	**100**	39.03	51.99	57.10	62.19
	Auto-reg.	**58.44**	**76.32**	**82.67**	**89.65**	**99.95**	**99.99**	**99.99**	**100**	**73.75**	**86.56**	**90.51**	**94.54**
Structural-based approach	ACF	**64.95**	**80.76**	**85.97**	**91.55**	**99.91**	**99.98**	**100**	**100**	**78.73**	**89.35**	**92.46**	**95.59**
	Edge det.	27.03	52.51	65.40	79.86	98.57	99.53	99.75	99.89	42.42	68.75	79.00	88.76
	Morphology	49.07	69.56	77.89	87.02	99.91	99.98	99.99	100	65.82	82.04	87.57	93.06

From the retrieval results on the ITESOFT dataset shown in Table 2, it can be noted that in the first group of statistical approaches, the GLTCS feature extraction method, similar to the CLEF_IP dataset, provided better F-scores in the Top-1 to Top-10 with 77.34%, 87.48%, 91.38%, and 94.99%, respectively. In the second group of statistical approaches, better F-scores of 78.26% and 94.89% at Top-1 and Top-10, respectively, were obtained using the $ILBP$ and $LBP_{8,1}$ feature extraction methods. For the transform-based approaches, the Gist operator attained the highest F-scores of 85.35% and 95.53% at Top-1 and Top-10, respectively. For the model-based approaches, the auto-regressive method obtained the highest F-scores at Top-1 to Top-10. The morphology method, amongst the structural-based approaches, provided the best F-scores and precision percentages. From the results shown in Table 2, it can be concluded that the Gist operator provided the best F-scores compared to all the feature extraction methods applied for DIR on the ITESOFT dataset.

Table 2. Comparison of the results obtained by four texture-based categories on the ITESOFT dataset

	Metric	Precision				Recall				F-score			
Category	**Method**	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)
Statistical-based approach (1st group)	GLCM	32.62	53.40	62.15	75.14	**100**	100	100	100	49.19	69.62	76.65	85.80
	GLRLM	35.67	54.73	63.75	75.84	99.93	99.87	99.98	99.99	52.57	70.71	77.86	86.26
	GLDM	54.80	72.53	73.91	83.16	80.00	80.00	99.95	99.99	65.05	76.08	84.98	90.80
	GLTCS	**63.09**	**77.75**	**84.16**	**90.46**	99.89	**100**	**100**	**100**	**77.34**	**87.48**	**91.38**	**94.99**
Statistical-based approach (2nd group)	LBP	56.90	69.71	80.00	88.76	87.33	96.28	99.11	99.75	68.90	80.86	88.53	93.93
	$LBP_{4,1}$	49.16	67.77	74.98	84.49	99.70	99.99	99.98	99.98	65.85	80.79	85.70	91.59
	$LBP_{8,1}$	47.57	69.50	80.04	**90.43**	93.00	99.81	99.80	99.82	62.94	81.14	88.83	**94.89**
	$LBP_{12,1.5}$	54.11	66.86	72.19	76.26	76.86	99.01	99.82	99.92	63.51	79.82	83.79	86.50
	$LBP_{16,2}$	52.71	65.71	73.18	81.69	94.34	98.99	98.85	99.94	67.63	78.99	84.10	89.90
	MBP	62.63	75.72	81.52	87.24	**100**	**100**	**100**	**100**	77.03	86.18	89.82	93.18
	ILBP	**64.29**	**77.85**	82.74	89.08	99.99	99.99	100	100	**78.26**	**87.54**	90.56	94.21
	F-LBP	48.28	68.31	74.82	82.71	99.99	99.99	100	100	65.12	81.17	85.60	90.54
	LTP	49.53	67.08	75.35	85.00	99.98	100	100	100	66.25	80.30	85.94	91.89
	ILTP	46.25	65.30	73.56	81.32	99.97	100	100	100	63.24	79.01	84.77	89.69
	BGC_1	60.08	72.01	**83.85**	82.85	75.20	99.94	100	99.94	66.79	83.70	**91.22**	90.59
	BGC_2	58.22	74.09	76.32	86.43	79.16	98.73	99.08	99.93	67.10	84.65	86.23	92.69
	BGC_3	59.77	76.51	82.35	88.65	82.97	100	100	100	69.49	86.69	90.32	93.98
	CLBP	61.57	70.84	77.84	86.06	64.08	99.98	99.99	99.98	62.80	82.92	87.54	92.50
	CSLBP	49.81	60.58	69.35	86.62	95.95	98.93	100	100	65.58	75.14	81.90	92.83
	D-LBP	55.50	68.48	78.16	87.27	79.86	99.99	99.99	100	65.49	81.29	87.73	93.20
	ID-LBP	55.73	69.18	77.04	85.78	79.99	100	100	100	65.69	81.78	87.03	92.34
Transform-based approach	Wavelet	69.53	80.47	83.75	87.64	91.02	95.52	97.79	98.91	78.84	87.35	90.23	92.94
	Fourier	51.71	57.35	67.73	79.44	50.00	99.99	99.99	99.99	50.84	72.89	80.76	88.54
	Gabor	71.61	81.84	84.59	87.66	92.12	96.14	96.99	99.19	80.58	88.42	90.37	93.07
	Radon	51.76	65.26	71.32	79.43	99.72	99.96	99.91	99.93	68.20	78.96	83.23	88.51
	Contourlet	73.89	83.79	86.98	91.16	98.69	99.34	99.89	99.91	84.51	90.91	92.99	95.38
	Gist	**74.56**	**85.45**	**87.56**	**91.44**	**99.79**	**99.97**	**100**	**100**	**85.35**	**92.14**	**93.37**	**95.53**
Model-based approach	MRF	61.03	73.64	79.94	84.01	**97.96**	98.98	99.29	99.68	75.21	84.45	88.80	91.18
	Fractal	59.19	74.32	79.79	87.46	91.37	95.42	98.24	99.53	71.84	83.56	88.06	93.10
	Auto-reg.	**64.81**	**76.13**	**82.81**	**90.24**	96.53	**99.30**	**99.40**	**100**	**77.55**	**86.19**	**90.35**	**94.87**
Structural-based approach	ACF	32.98	47.59	58.77	71.74	75.00	99.98	99.95	99.98	45.81	64.49	74.02	83.54
	Edge det.	47.81	66.07	75.12	84.31	79.98	**100**	**100**	**100**	59.85	79.57	85.79	91.49
	Morphology	**65.28**	**77.49**	**80.93**	**86.52**	**98.87**	98.94	99.09	99.89	**78.63**	**86.91**	**89.09**	**92.73**

Retrieval results on the MTDB shown in Table 3 reveal that in the first group of statistical approaches, although the GLDM features provided the highest F-scores, the GLTCS feature extraction method, similar to the ITESOFT and CLEF_IP datasets, provided comparable F-scores of 59.42%, 71.77%, 76.08% and 82.02% in the Top-1 to Top-10, respectively. In the second group of statistical approaches, the highest precision and the best F-score of 86.46% at the Top-10 were obtained using the ILBP feature extraction method. For the transform-based approaches, the Gist operator provided the highest F-scores of 74.37% and 89.81% at Top-1 and Top-10, respectively. For the model-based approaches, the auto-regressive method obtained the highest F-scores at Top-1 to Top-10, and the morphological feature extraction method provided the best precision and F-scores amongst the structural-based approaches. From the results shown in Table 3, it is evident that the Gist operator again provided the best F-scores compared to all the feature extraction methods applied for DIR on the MTDB.

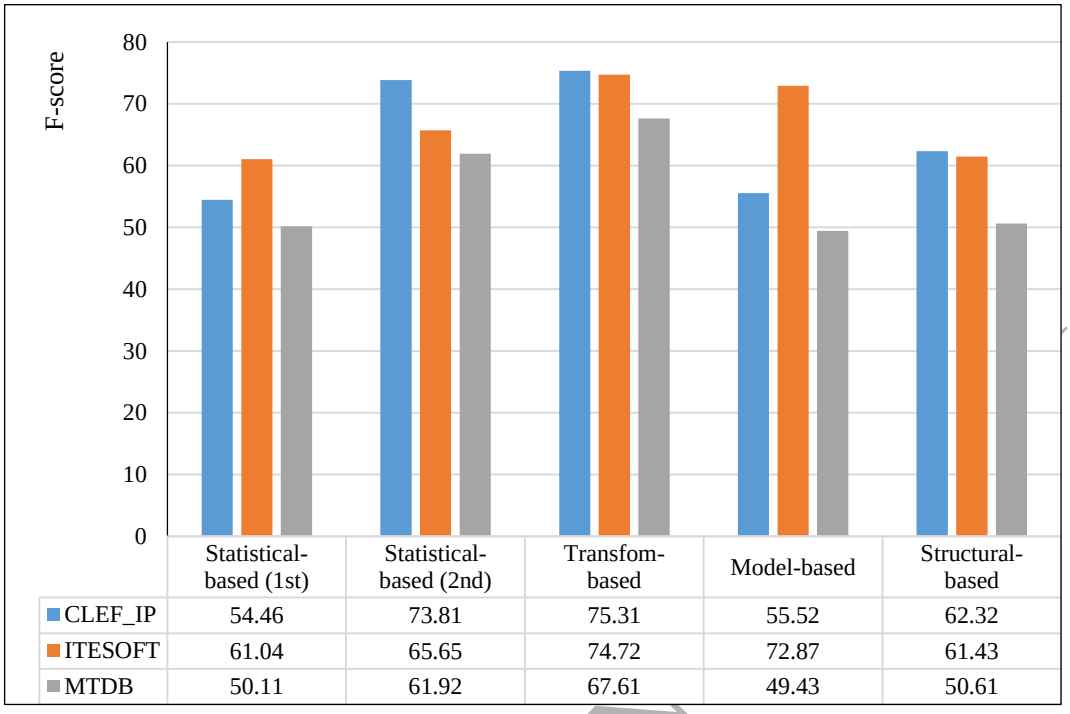
Table 3. Comparison of the results obtained by four texture-based categories on the MTDB.

	Metric	Precision				Recall				F-score			
Category	**Method**	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)
Statistical-based approach (1st group)	GLCM	25.90	42.44	51.17	65.41	99.50	100	100	100	41.10	59.59	67.70	79.09
	GLRLM	24.36	38.80	45.54	58.80	**100**	100	100	100	39.18	55.91	62.58	74.06
	GLDM	**43.83**	**57.00**	**62.38**	**70.44**	98.96	**100**	**100**	**100**	**60.75**	**72.61**	**76.84**	**82.66**
	GLTCS	42.27	55.97	61.40	69.52	99.95	99.97	99.99	100	59.42	71.77	76.08	82.02
Statistical-based approach (2nd group)	LBP	51.63	66.13	69.87	76.00	98.09	99.55	99.96	100	**67.66**	**79.52**	82.25	86.36
	$LBP_{4,1}$	42.30	55.75	60.62	68.56	99.57	99.99	100	100	59.37	71.58	75.48	81.34
	$LBP_{8,1}$	50.33	63.00	68.50	75.17	94.85	98.48	99.67	99.71	65.76	76.84	81.20	85.71
	$LBP_{12,1.5}$	51.84	60.85	65.78	72.03	77.19	97.33	99.21	100	62.02	74.89	79.11	83.74
	$LBP_{16,2}$	49.64	63.33	68.01	71.12	96.53	99.30	99.40	100	65.56	77.34	80.76	83.13
	MBP	51.16	59.36	65.13	73.55	84.64	95.03	99.11	99.66	63.77	73.07	78.60	84.64
	ILBP	**52.09**	**66.21**	**71.85**	**76.17**	92.01	96.72	98.76	99.98	66.52	78.55	**83.18**	**86.46**
	F-LBP	46.42	59.36	65.09	71.84	96.33	99.01	99.99	100	62.65	74.22	78.85	83.62
	LTP	43.92	55.21	61.57	69.70	99.18	99.98	100	100	60.88	71.14	76.22	82.15
	ILTP	41.10	53.60	60.31	71.35	98.02	99.21	100	100	57.91	69.60	75.25	83.28
	BGC_1	46.93	59.07	63.44	70.24	85.57	94.75	97.70	99.62	60.61	72.77	76.93	82.39
	BGC_2	47.81	59.40	64.49	71.84	95.75	98.42	99.96	100	63.78	74.09	78.40	83.62
	BGC_3	39.64	53.17	58.55	67.23	99.75	100	100	100	56.74	69.42	73.86	80.41
	CLBP	47.36	56.99	63.34	72.25	82.25	97.11	99.04	99.67	60.11	71.83	77.26	83.77
	CSLBP	38.63	54.46	61.63	71.27	**99.97**	**100**	**100**	**100**	55.73	70.52	76.26	83.22
	D-LBP	46.55	60.38	64.93	72.84	90.99	97.42	98.48	99.98	61.59	74.56	78.26	84.28
	ID-LBP	45.37	56.52	63.64	68.75	97.51	98.31	99.96	100	61.92	71.78	77.77	81.48
Transform-based approach	Wavelet	57.75	67.59	73.41	79.26	98.06	98.86	99.23	99.98	72.91	80.02	84.39	88.42
	Fourier	32.01	48.34	56.78	68.02	99.79	100	100	100	48.49	65.18	72.43	80.97
	Gabor	60.10	68.98	73.76	80.74	96.25	97.36	99.67	99.86	74.13	80.75	84.67	89.29
	Radon	44.78	54.08	59.21	70.92	96.68	99.71	99.99	100	61.21	70.13	74.21	82.99
	Contourlet	**60.36**	**71.20**	**76.50**	**81.58**	94.83	98.44	99.21	99.55	73.76	**82.63**	**86.39**	89.68
	Gist	59.26	69.13	72.96	79.21	**99.95**	**100**	**100**	**100**	**74.37**	81.74	84.37	**89.81**
Model-based approach	MRF	19.67	49.86	57.24	67.44	**100**	**100**	**100**	**100**	32.87	66.54	72.81	80.56
	Fractal	28.26	57.95	63.78	74.31	95.96	99.98	100	100	43.67	73.37	77.89	85.26
	Auto-reg.	**56.86**	**68.56**	**72.33**	**77.68**	97.22	98.92	98.89	100	**71.76**	**80.99**	**83.55**	**87.44**
Structural-based approach	ACF	20.41	39.01	47.17	58.92	**100**	**100**	**100**	**100**	33.89	56.13	64.10	74.15
	Edge det.	32.45	49.18	56.50	68.03	99.97	99.99	99.98	100	49.00	65.93	72.20	80.98
	Morphology	**55.34**	**63.27**	**67.76**	**75.93**	97.47	99.96	100	100	**70.60**	**77.49**	**80.78**	**86.32**

To summarise the results obtained, and to compare the effectiveness of different texture-based feature categories, the average of F-scores at the Top-1 on the CLEF_IP, ITESOFT, and MTDB datasets were calculated for each category, and the results are presented in Figure 4. From Figure 4 it is evident that texture features in the transform-based category consistently performed better on all three datasets compared to the other categories. Moreover, in line with this finding, we noted that the Gist descriptor, in particular, showed more consistency in the retrieval process and provided the best retrieval results on all three datasets compared to the other feature extraction methods. To get an idea of the performance of the Gist operator, a precision and recall curve of the retrieval results obtained from the ITESOFT dataset is shown in Figure 5. The precision and recalls were computed based on different values of γ (or threshold τ). From the results shown in Figure 5, it can be noted that the Gist features provided quite high precision (83% to 97%) when recalls varied between 100% and 48%.

Feature extraction methods in other categories provided inconsistent results on different datasets. However, there were some exceptions, for example, the GLTCS from the first group of statistical approaches, variations of LBP and BGC from the second group of statistical approaches, the autoregressive method from the model-based approaches and the morphology-based method from the structural-based approaches showed more consistency in providing retrieval results compared to the other feature extraction methods. Therefore, it is recommended that transform-based texture features,

in general, can be applied on scenarios where a DIR system is required to deal with document images with diverse content, size, and format (colour/grey/binary).



	Statistical-based (1st)	Statistical-based (2nd)	Transfom-based	Model-based	Structural-based
CLEF_IP	54.46	73.81	75.31	55.52	62.32
ITESOFT	61.04	65.65	74.72	72.87	61.43
MTDB	50.11	61.92	67.61	49.43	50.61

Figure 4. Comparison of the average F-scores at Top-1 obtained based on different categories of texture-based features applied for DIR on the CLEF_IP, ITESOFT and MTDB datasets

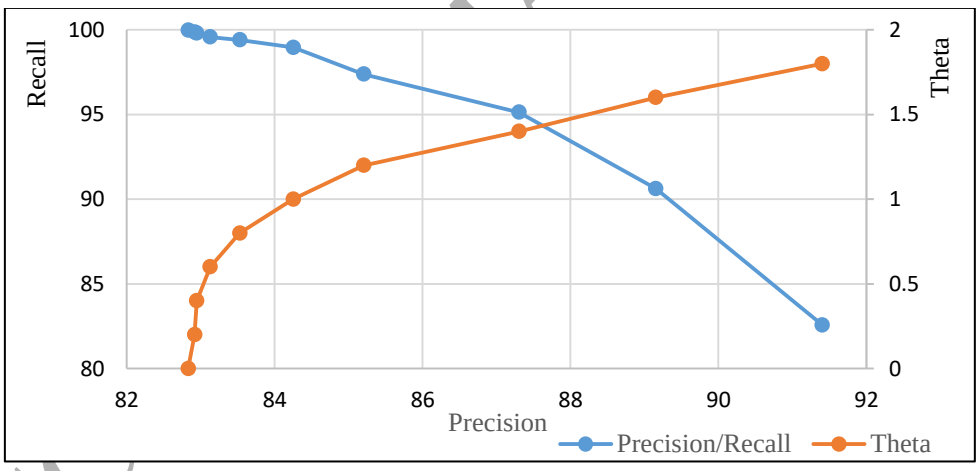


Figure 5. The precision and recall curve obtained from the Gist descriptor on the ITESOFT dataset with different thresholds.

Furthermore, to illustrate the behaviour of different feature extraction methods in relation to the retrieval results obtained for each class, the methods performing the best in their category were selected. The Top-1 retrieval results on the ITESOFT dataset with 12 classes were then computed, and the results are shown in Table 4. A graphical representation of the results based on F-scores is also provided in Figure 6. As is demonstrated in Figure 6 and Table 4, considering the F-scores for comparison, the Gist operator performed well on most of the classes except Class 7 (C7), Class 10 (C10), and Class 11 (C11), where the ILBP, GLTCS, and morphology features provided the best performance in these three classes, respectively. This observation may lead to the use of a feature fusion strategy by considering transform and statistical-based features that may result in better document retrieval results.

Table 4. The Top-1 results obtained from the texture-based features performing well on all classes of the ITESOFT dataset.

Method	Metric	Class Label											
		C1	**C2**	**C 3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C12**
GLTCS	Precision	73.72	76.58	23.08	53.73	42.31	53.85	**82.35**	81.55	45.45	**72.73**	97.96	34.15
	Recall	99.78	100	100	100	100	100	93.21	100	98.08	100	100	100
	F-score	84.79	86.74	37.50	69.90	59.46	70.00	87.44	89.84	62.12	**84.21**	98.97	50.91
ILBP	Precision	78.21	82.25	31.88	54.85	41.83	60.71	80.21	85.68	50.00	43.18	97.96	36.31
	Recall	100	**100**	100	**100**	99.55	100	100	**100**	100	98.02	**100**	100
	F-score	87.77	90.26	48.34	70.84	58.90	75.56	**89.02**	92.29	66.67	59.95	98.97	53.28
Gist	Precision	**90.36**	**85.56**	**54.88**	80.20	**81.41**	**75.82**	73.61	**90.09**	**62.19**	69.70	97.96	**94.44**
	Recall	**100**	99.55	**100**	98.69	**100**	**100**	**100**	98.02	**100**	**100**	**100**	**100**
	F-score	**94.94**	**92.03**	**70.87**	**88.49**	**89.75**	**86.25**	84.80	**93.89**	**76.69**	82.14	98.97	**97.14**
Auto-reg.	Precision	82.63	73.86	40.51	75.13	79.33	35.44	74.77	81.98	41.50	36.36	98.47	80.20
	Recall	100	99.79	98.44	98.57	100	100	100	96.20	100	100	100	100
	F-score	90.49	84.89	57.39	85.27	88.47	52.34	85.57	88.52	58.66	53.33	99.23	89.01
Morphology	Precision	82.69	79.14	43.59	**84.13**	80.00	46.15	41.67	78.95	40.00	18.18	**100**	76.19
	Recall	99.62	100	100	92.20	100	95.83	100	95.69	100	100	100	100
	F-score	90.37	88.36	60.71	87.98	88.89	62.30	58.82	86.51	57.14	30.77	**100**	86.49

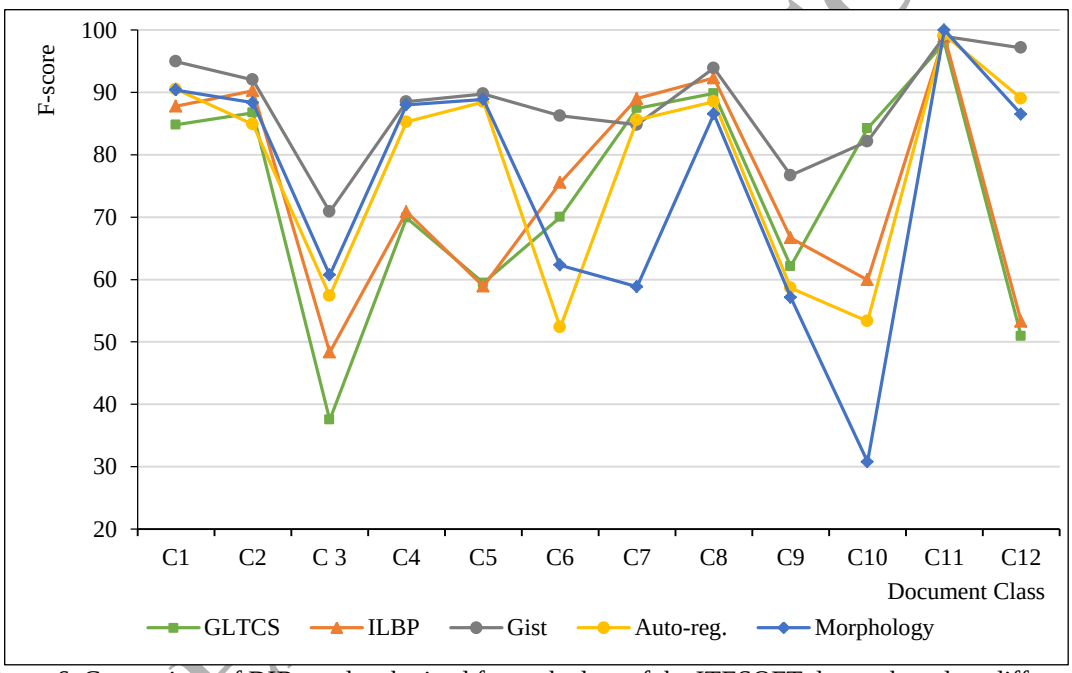


Figure 6. Comparison of DIR results obtained for each class of the ITESOFT dataset based on different texture-based features (from each category)

4.3. *Impact of Different Resolutions on Texture-based DIR Results*

To investigate the impact of different resolutions on texture-based DIR, document images with three different resolutions (high, low and thumbnail size) from the MTDB were considered for training purposes, and DIR results were obtained according to each resolution.

The DIR results obtained when the system was trained with high-resolution samples are presented in Table 5. From Table 5 it is noted that in the first group of statistical approaches, the GLRLM performed better compared to the other methods in the group with an F-score of 68.21% at the Top-10. The standard LBP method showed the highest F-score results in the Top-1 to Top-10 in the second group of statistical-based approaches. The Gabor wavelet method from the transform-based approaches provided the highest F-scores with the F-scores of 70.72% and 81.72% at Top-1 and Top-10, respectively. In the model-based and structural-based approaches, the auto-regressive method and

morphology performed well in their respective groups providing high F-scores compared to the other methods in their categories.

Table 5. Comparison of the results obtained by four texture-based categories on the MTDB when the DIR system was trained with only high-resolution document images and tested on the remainder of the dataset.

	Metric	**Precision**				**Recall**				**F-score**			
Category	**Method**	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)
Statistical-based approach (1st group)	GLCM	20.75	33.00	37.66	48.79	56.49	81.60	84.63	93.85	30.35	47.00	52.13	64.21
	GLRLM	18.70	**31.18**	**39.98**	**52.15**	**97.06**	98.37	98.14	98.56	31.36	**47.36**	**56.82**	**68.21**
	GLDM	22.79	23.63	29.63	47.87	96.21	**100**	**100**	**100**	**36.85**	38.22	45.72	64.75
	GLTCS	**23.23**	26.57	30.33	45.80	79.06	92.81	98.04	99.92	35.90	41.31	46.32	62.81
Statistical-based approach (2nd group)	LBP	**46.66**	**59.82**	**66.62**	**78.92**	75.82	92.62	90.89	89.21	**57.77**	**72.69**	**76.88**	**83.75**
	$LBP_{4,1}$	22.98	25.06	31.16	53.88	88.57	94.08	99.71	99.89	36.49	39.58	47.48	70.00
	$LBP_{8,1}$	21.12	25.30	30.92	53.41	96.08	99.93	100	100	34.63	40.38	47.24	69.63
	$LBP_{12,1.5}$	24.15	27.59	30.12	55.29	84.43	94.69	98.97	99.52	37.98	42.97	49.29	70.56
	$LBP_{16,2}$	24.13	35.35	38.17	64.92	89.79	90.18	91.42	92.15	38.03	51.16	54.89	75.69
	MBP	20.87	27.73	31.07	50.97	92.44	97.29	98.99	99.84	34.05	43.16	47.29	67.49
	ILBP	22.02	26.31	29.92	47.96	93.28	99.87	99.37	99.96	35.62	41.65	45.99	64.82
	F-LBP	22.83	23.16	30.20	50.32	92.69	93.36	95.83	99.35	35.93	37.12	45.93	66.81
	LTP	20.08	26.72	30.25	47.87	96.16	96.80	98.87	99.94	33.23	41.87	46.32	64.73
	ILTP	22.02	26.31	29.92	47.96	93.28	99.87	99.37	99.96	35.62	41.65	45.99	64.82
	BGC_1	21.75	30.92	31.85	52.12	99.92	99.98	99.50	99.98	35.72	47.23	48.26	68.52
	BGC_2	20.96	27.50	31.08	55.00	97.86	98.89	100	100	34.58	43.03	47.42	70.96
	BGC_3	21.10	25.09	30.98	51.35	90.35	94.33	99.63	99.92	34.21	39.64	47.27	67.84
	CLBP	19.94	23.02	31.45	46.91	97.30	98.34	99.04	99.96	33.09	37.30	47.74	63.85
	CSLBP	18.18	25.70	29.77	45.85	**99.95**	**100**	**100**	**100**	30.76	40.89	45.88	62.86
	D-LBP	23.09	30.16	30.99	51.41	84.83	96.89	98.37	99.40	36.30	46.00	47.14	67.77
	ID-LBP	21.41	23.33	31.49	53.23	93.24	99.98	98.91	100	34.83	37.83	47.77	69.47
Transform-based approach	Wavelet	44.17	50.30	52.99	57.52	85.51	94.67	96.03	97.73	58.73	66.79	68.37	73.42
	Fourier	18.08	23.55	29.59	47.60	**99.94**	**99.98**	**100**	**100**	30.62	38.13	45.67	64.49
	Gabor	**62.31**	**64.43**	**64.70**	**70.58**	81.77	90.98	94.55	97.03	**70.72**	**75.44**	**76.82**	**81.72**
	Radon	34.70	46.00	53.00	63.73	89.52	94.66	96.97	98.49	50.01	61.91	68.54	77.39
	Contourlet	40.62	54.16	59.92	69.43	87.47	97.73	98.48	99.54	55.48	69.69	74.50	81.80
	Gist	52.05	60.57	61.42	66.42	86.09	92.27	95.21	99.03	64.87	73.13	74.67	79.51
Model-based approach	MRF	30.54	45.61	52.88	61.99	92.70	95.25	96.59	96.47	45.95	61.69	68.35	75.48
	Fractal	33.53	47.11	53.28	64.03	**99.75**	**99.92**	**100**	**100**	50.18	64.03	69.52	78.07
	Auto-reg.	**38.16**	**49.08**	**54.25**	**64.84**	95.08	97.16	99.30	99.72	**54.47**	**65.21**	**70.17**	**78.58**
Structural-based approach	ACF	14.27	27.29	34.20	53.05	**100**	100	100	100	24.97	42.88	50.96	69.33
	Edge det.	22.79	36.14	43.15	60.09	75.43	88.00	91.21	90.53	35.00	51.23	58.58	72.24
	Morphology	**57.62**	**59.39**	**60.44**	**67.09**	82.62	90.84	91.37	95.76	**67.89**	**71.83**	**72.21**	**78.90**

Table 6 presents the experimental results on low-resolution samples of the MTDB method. In the first statistical group, the GLTCS with F-scores of 56.56% and 80.59% at Top-1 and Top-10 provided the highest values in most of the cases in the group. In the second statistical group, LTP at Top-1, with 64.54% and ID-LBP at Top-10 with 80.40% demonstrated the highest F-scores. Among the transform-based approaches, the Gist operator at Top-1 (67.27%) and Contourlet at Top-10 (85.62%) provided the highest F-score results in the transform-based approaches as well as other methods in Table 6. The auto-regressive method and morphology from two other categories provided better F-scores at Top-1 compared to the other methods in their own categories.

Table 6. Comparison of the results obtained by four texture-based categories on the MTDB when the DIR system was trained with only low-resolution document images and tested on the remainder of the dataset.

	Metric	Precision				Recall				F-score			
Category	**Method**	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)
Statistical-based approach (1st group)	GLCM	26.83	35.56	36.67	51.52	43.52	78.87	89.10	**98.10**	33.20	49.02	51.96	67.56
	GLRLM	29.86	37.48	39.74	52.27	66.41	**97.70**	**97.71**	97.59	49.73	45.74	54.18	68.07
	GLDM	41.25	52.01	58.90	68.60	82.15	86.99	89.78	88.60	54.93	65.10	71.13	77.33
	GLTCS	**42.63**	**59.26**	**67.06**	**72.23**	**84.00**	87.91	90.86	91.15	**56.56**	**70.80**	**77.17**	**80.59**
Statistical-based approach (2nd group)	LBP	59.01	64.27	63.83	71.13	70.47	76.24	82.72	88.15	64.24	69.74	72.06	78.73
	$LBP_{4,1}$	47.32	61.65	64.41	76.66	70.39	82.24	85.89	86.77	56.59	70.47	73.62	81.40
	$LBP_{8,1}$	48.11	57.64	61.55	68.26	68.83	82.28	87.44	87.45	56.64	67.79	72.25	76.67
	$LBP_{12,1.5}$	51.89	60.69	63.79	69.48	74.28	79.45	83.85	86.12	61.10	68.81	72.46	76.91
	$LBP_{16,2}$	48.33	59.73	60.08	63.35	66.34	76.78	84.57	89.06	55.62	67.74	70.50	77.69
	MBP	33.82	36.74	40.24	57.90	87.46	**89.22**	**85.66**	**93.09**	48.77	52.04	54.76	71.39
	ILBP	39.49	61.83	69.40	77.64	81.82	75.10	72.62	81.54	53.27	67.82	70.98	79.54
	F-LBP	48.86	62.16	67.15	69.97	76.26	85.89	80.21	88.89	59.56	72.12	73.10	78.30
	LTP	52.67	61.58	65.02	76.25	83.30	82.41	78.96	80.06	**64.54**	70.49	71.31	78.11
	ILTP	39.49	61.83	69.40	**77.78**	81.82	75.10	72.62	81.54	53.27	67.82	70.98	79.54
	BGC_1	72.02	**73.53**	68.60	67.73	43.80	60.91	68.15	79.44	54.47	68.07	68.37	73.12
	BGC_2	**72.15**	73.37	59.22	69.53	52.14	62.70	74.67	86.00	60.53	67.70	66.05	76.90
	BGC_3	48.15	63.00	66.31	66.70	69.98	81.26	77.54	88.66	57.05	70.98	71.48	76.12
	CLBP	45.11	61.63	66.77	74.30	81.62	84.78	84.77	90.06	58.10	71.37	74.70	81.42
	CSLBP	43.75	55.41	62.09	68.52	79.12	84.47	87.08	85.67	56.35	66.92	72.49	76.14
	D-LBP	55.86	63.66	66.72	73.21	70.73	82.51	84.45	87.10	62.42	71.87	74.55	79.56
	ID-LBP	32.09	64.65	**70.89**	72.96	**88.18**	85.64	86.80	89.54	47.05	**73.68**	**78.04**	**80.40**
Transform-based approach	Wavelet	46.28	51.61	52.95	58.03	80.89	90.49	94.35	96.70	59.56	65.90	67.92	72.95
	Fourier	26.06	33.25	42.39	54.70	90.49	92.05	94.51	92.99	40.47	48.85	58.53	68.88
	Gabor	53.80	58.14	59.92	64.54	84.08	88.47	92.01	97.28	65.62	70.17	72.58	77.60
	Radon	36.25	48.90	54.53	65.83	**90.64**	94.00	96.97	98.37	51.79	64.33	69.81	78.88
	Contourlet	54.11	**66.50**	**72.04**	**75.42**	83.82	**94.66**	95.94	**99.01**	65.77	**78.12**	**82.29**	**85.62**
	Gist	**55.57**	61.16	62.22	70.26	85.20	89.99	93.36	98.56	**67.27**	72.83	74.67	82.04
Model-based approach	MRF	32.08	34.97	38.03	48.94	**95.61**	**97.36**	98.18	98.33	48.04	51.45	54.82	65.36
	Fractal	43.09	**52.16**	**58.56**	**67.85**	86.61	92.23	**98.63**	**99.31**	57.55	**66.63**	**73.49**	**80.62**
	Auto-reg.	**44.86**	49.97	54.78	64.34	89.54	96.86	97.99	98.74	**59.77**	65.93	70.27	77.91
Structural-based approach	ACF	13.13	26.76	36.53	49.51	**89.15**	93.00	**95.32**	92.07	22.89	41.56	52.82	64.39
	Edge det.	37.99	52.91	**58.91**	**69.18**	72.29	85.73	87.32	88.20	49.80	65.44	70.36	77.54
	Morphology	**49.33**	**54.08**	58.20	66.63	84.77	**94.18**	94.69	97.17	**62.37**	**68.71**	**72.09**	**79.05**

Table 7 shows the experimental results on thumbnail size samples of the MTDB dataset. In the first statistical group, the GLTCS method demonstrated the highest performance in the group. In the second statistical group, $LBP_{16,2}$ and F-LBP obtained the highest F-scores in the group. By applying transform-based approaches on thumbnail size document images, the Contourlet method provided the highest precision and F-score at Top-1 to Top-10, respectively. In model-based and structural-based approaches, the auto-regressive and morphology methods performed well, similar to the results using high and low-resolution images for training.

The experimental results (F-scores at Top-1), considering different resolutions for training, demonstrated that the statistical texture feature extraction approaches, except the GLRLM method, provided better retrieval results when the proposed retrieval system was trained using low-resolution document images compared to high-resolution. There was no consistency in the results obtained from the transform-based approaches when the system was trained on high-resolution, low-resolution, and thumbnail size document images. However, the overall results were better when the DIR system was trained using low-resolution images. In model-based approaches, training the system using high-resolution document images provided a higher F-score, except in the Markov random method that

provided a better F-score using low-resolution and thumbnail size images for training. The structural-based approaches usually demonstrated better results on high-resolution images.

Table 7. Comparison of the results obtained by four texture-based categories on the MTDB dataset, the DIR system was trained with only thumbnail size document images and tested on the rest of the dataset.

	Metric	**Precision**				**Recall**				**F-score**			
Category	**Method**	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)	Top-1 (%)	Top-3 (%)	Top-5 (%)	Top-10 (%)
Statistical-based approach (1st group)	GLCM	25.08	37.65	37.43	46.87	64.07	86.96	91.08	98.38	36.05	52.55	53.06	63.49
	GLRLM	13.52	26.06	36.51	52.80	**99.94**	**100**	**100**	**100**	23.81	41.34	53.49	69.11
	GLDM	**43.37**	54.97	63.23	71.96	89.74	91.00	92.25	92.26	**58.47**	68.53	75.03	80.86
	GLTCS	41.60	**56.34**	**65.88**	**75.25**	88.38	90.93	91.82	92.38	56.57	**69.57**	**76.72**	**82.94**
Statistical-based approach (2nd group)	LBP	53.41	64.77	70.13	79.29	72.77	89.47	87.84	88.56	61.60	75.15	77.99	83.67
	$LBP_{4,1}$	40.69	57.16	64.91	74.55	91.14	91.83	92.31	92.56	56.26	70.48	76.22	82.59
	$LBP_{8,1}$	46.83	60.08	67.50	78.00	88.40	91.10	92.34	92.32	61.23	72.41	77.99	84.56
	$LBP_{12,1.5}$	49.98	60.79	67.51	70.44	**91.38**	**91.85**	**92.37**	**93.91**	64.62	73.16	78.00	81.64
	$LBP_{16,2}$	52.77	65.08	70.91	78.08	90.70	90.56	89.54	88.19	**66.72**	**76.41**	79.14	82.83
	MBP	47.29	58.51	65.16	75.60	85.39	89.28	88.85	88.45	60.87	70.69	75.18	81.52
	ILBP	32.30	50.86	58.45	69.03	90.18	84.22	85.31	91.99	47.57	63.42	69.37	78.88
	F-LBP	46.59	64.32	**70.94**	**80.21**	90.91	**91.85**	92.31	91.40	61.61	75.66	**80.16**	**85.44**
	LTP	38.68	52.26	57.79	69.27	79.63	77.77	79.31	78.44	52.07	62.51	66.86	73.57
	ILTP	32.30	50.86	58.45	69.03	90.18	84.22	85.31	91.99	47.57	63.42	69.37	78.88
	BGC_1	**65.31**	**66.07**	68.35	69.32	61.69	79.37	81.23	83.64	63.45	69.03	70.25	75.81
	BGC_2	50.10	63.13	68.63	79.21	86.54	90.44	88.78	87.40	63.46	74.36	77.41	83.10
	BGC_3	47.20	64.05	69.26	78.84	82.57	89.81	89.55	86.35	60.07	74.77	78.11	82.42
	CLBP	47.52	62.46	70.36	77.92	83.16	84.61	87.20	86.88	60.48	71.87	77.88	82.16
	CSLBP	19.46	37.99	47.65	64.10	84.22	77.30	84.88	87.17	31.62	50.94	61.03	73.87
	D-LBP	44.40	53.89	61.49	68.31	86.38	89.90	90.61	91.87	58.65	67.39	73.26	78.36
	ID-LBP	37.09	49.63	55.69	64.60	89.09	91.09	91.25	91.55	52.38	64.25	69.17	75.75
Transform-based approach	Wavelet	45.86	37.74	40.62	50.17	42.58	70.57	74.50	97.45	44.16	49.18	52.57	66.24
	Fourier	26.71	37.12	41.76	48.58	79.73	93.18	91.95	95.30	40.01	53.09	57.43	64.36
	Gabor	42.73	52.11	54.82	59.23	82.09	91.42	95.45	97.46	56.21	66.38	69.64	73.68
	Radon	25.94	41.84	49.91	61.70	**99.54**	**99.78**	**100**	**100**	41.16	58.95	66.59	76.31
	Contourlet	**48.64**	**61.92**	**68.22**	**75.22**	89.90	96.91	95.73	98.61	**63.13**	**75.56**	**79.67**	**85.34**
	Gist	48.41	58.12	52.81	55.22	65.37	81.75	90.94	97.79	55.62	67.94	66.82	70.58
Model-based approach	MRF	19.89	29.32	37.93	57.18	95.61	97.36	98.18	98.33	32.93	45.07	54.72	72.31
	Fractal	28.07	**47.13**	**50.78**	**58.84**	53.22	74.70	88.14	97.01	36.76	57.80	64.44	**73.25**
	Auto-reg.	**30.88**	44.11	49.74	57.08	**99.69**	**98.75**	**99.98**	**99.98**	**47.15**	**60.98**	**66.43**	72.67
Structural-based approach	ACF	13.41	28.29	34.31	44.08	92.33	93.57	94.84	95.54	23.42	43.44	50.39	60.32
	Edge det.	23.03	35.52	43.11	**59.42**	77.17	87.80	88.21	90.29	35.47	50.57	57.92	71.68
	Morphology	**29.18**	**42.94**	**48.91**	55.80	**99.97**	**100**	**100**	**100**	**45.17**	**60.08**	**65.69**	**71.63**

To summarise the results and discussion on the impact of different resolutions on DIR, the average F-scores at the Top-1 level from three resolutions are presented in Figure 7. Contrary to our expectations, training the DIR system using low-resolution document images demonstrated a higher mean and quartile F-scores compared to the results obtained from the system trained using high-resolution document images. It is also worth mentioning that training the system using low-resolution and thumbnail-size samples provided results with higher precision compared to the system trained using high-resolution samples. However, recall percentages obtained from the system trained using high-resolution documents were higher than the results obtained using low-resolution and thumbnail size document images for training.

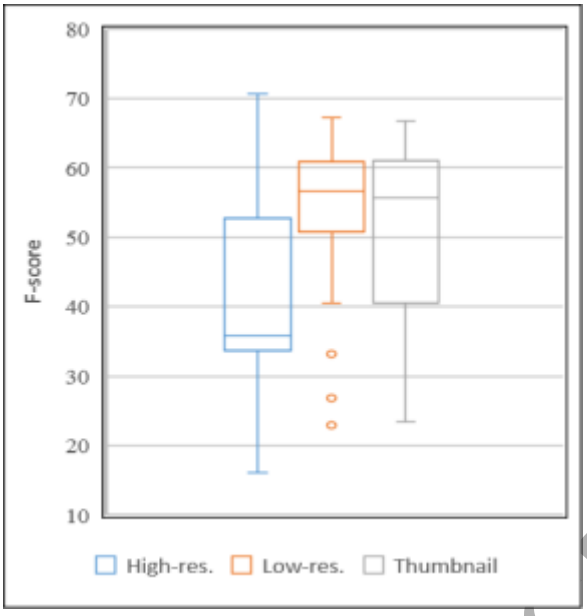


Figure 7. Comparison of the F-scores obtained at Top-1 for DIR using different document image resolutions of the MTDB.

4.4. *Comparison of Computing Times and Number of Features*

The number of extracted features and time complexity for extracting features from a high-resolution document image are critical for any DIR system. Therefore in this study, the number of features and the computing time were calculated for each feature extraction method. The experimental results were conducted using a Dell laptop with 16 Gigabytes memory (RAM) and a processor (CPU) of Core i5-5300U. All the texture-based feature extraction methods and DIR systems were implemented using MATLAB R2017a on Windows 7. A single document image of a large size from the MTDB was used for this experiment, and the number of features and computing time for extracting the features were calculated for each method as presented in Table 8.

It is apparent from Table 8 that the GLRLM is the fastest algorithm in terms of computing time among the first group of statistical-based feature extraction methods, although the GLCM, with only 4 features, needs the smallest memory space for storing the features and creating a knowledge base. In the second group of statistical approaches, the $LBP_{4,1}$ method with 15 features and the CSLBP with the lowest computing time, were the best in the group in relation to the number of features and computing time. In the transform-based category, extracting features using the Fourier transform required more computing time but took the smallest memory space compared to other transform-based techniques. In contrast, Contourlet feature extraction was the fastest method in the group. The fractal and edge detection methods provided a lower number of features for the retrieval process; however, the auto-regressive and morphology methods were the fastest for extracting features in the model-based and structural-based approaches. In summary, the model-based feature extraction methods were the fastest texture-based feature extraction methods compared to the other approaches. The transform-based feature extraction approaches with around half a second computing time were also fast, making them suitable for document image retrieval, as they performed well in terms of the retrieval results.

Table 8. The number of extracted features and computational time of texture-based feature extraction methods

Category	Method	Number of Features	Computing Time (seconds)
Statistical-based approach (1st group)	GLCM	**4**	1.4540
	GLRLM	7	**0.5251**
	GLDM	1024	0.9589
	GLTCS	24	1.4949
Statistical-based approach (2nd group)	LBP	256	0.9882
	$LBP_{4,1}$	**15**	0.6946
	$LBP_{8,1}$	59	1.5323
	$LBP_{12,1.5}$	135	3.3937
	$LBP_{16,2}$	243	5.0382
	MBP	511	1.7615
	ILBP	511	1.0471
	F-LBP	30	0.7632
	LTP	512	1.4830
	ILTP	1024	2.1140
	BGC_1	255	0.9470
	BGC_2	225	1.2762
	BGC_3	255	1.0995
	CLBP	256	0.8357
	CSLBP	16	**0.4051**
	D-LBP	16	0.5607
	ID-LBP	16	0.4788
Transform-based approach	Wavelet	140	0.6823
	Fourier	**26**	0.7296
	Gabor	120	0.3758
	Radon	581	0.2812
	Contourlet	258	**0.0537**
	Gist	513	0.5611
Model-based approach	MRF	12600	0.0946
	Fractal	**18**	0.0900
	Auto-reg.	90	**0.0092**
Structural-based approach	ACF	50	1.6032
	Edge det.	**4**	1.1239
	Morphology	90	**0.0404**

5. Conclusions and Future Work

The present study makes several noteworthy contributions to the application of texture-based feature extraction methods for document image retrieval. Comparison of the impacts of twenty-six texture-based features from four different categories including statistical, transform, model, and structural-based features for DIR was attempted. Different characteristics of each texture-based feature extraction technique, in terms of computing time and the number of features, were also studied. The experimental results proved that the transform-based texture features generally provided higher document retrieval results compared to the other categories of texture features used for DIR. Specifically, the Gist operator consistently resulted in quite high DIR results when employed on different datasets. In addition, the experimental analysis performed in this research work confirmed that the resolution of document images could influence texture-based document image retrieval results.

It is recommended that initially further research needs to focus on the creation of a larger, more complex, unstructured, and multi-lingual document dataset, and then evaluate the available DIR methods using this large dataset. Furthermore, the use of other classifiers, especially deep learning technology, can be taken into consideration in order to improve DIR results in future research. In addition, prior knowledge about the document classes may also be integrated into the feature set to further improve the retrieval performance. Another direction for future research can be to use advanced

human-computer interaction techniques to include human interaction data at different stages of DIR, including digitisation, pre-processing, feature extraction, and classification for optimising the retrieval process.

Acknowledgment

The authors would like to thank V.P. D'Andecy of the ITESOFT Company for providing access to the ITESOFT dataset. We are also thankful to the anonymous reviewers for providing valuable comments, which highly improved the quality of this paper.

References

Alaei, A., Roy, P. P., & Pal, U. (2016). *Logo and seal based administrative document image retrieval: a survey*. Computer Science Review, 22, pp. 47-63.

Alaei, F., Alaei, A., Blumenstein, M., & Pal, U. (2016a). *A brief review of document image retrieval methods: Recent* advances. Paper presented at the International Joint Conference on.Neural Networks (IJCNN), pp. 3500-3507.

Alaei, F., Alaei, A., Blumenstein, M., & Pal, U. (2016b). *Document image retrieval based on texture features and similarity fusion.* Paper presented at the International Conference on the Image and Vision Computing New Zealand (IVCNZ), pp. 1-6.

Alaei, F., Alaei, A., Pal, U., & Blumenstein, M. (2016c). *Document image retrieval based on texture features: a recognition-free approach.* Paper presented at the International Conference on Digital Image Computing: Techniques and Applications (DICTA), pp. 1-7.

Alaei, F., Alaei, A., Pal, U., & Blumenstein, M. (2017). *Fast local binary pattern: application to document image retrieval.* Paper presented at the Image and Vision Computing New Zealand (IVCNZ), pp. 1-6.

Beylkin, G. (1987). *Discrete Radon transform*. IEEE transactions on acoustics, speech, and signal processing, 35(2), pp. 162-172.

Blake, A., Kohli, P., & Rother, C. (2011). *Markov random fields for vision and image processing*: Mit Press.
Brigham, E. O., & Morrow, R. (1967). *The fast Fourier transform*. IEEE spectrum, 4(12), pp. 63-70.

Brigham, E. O., & Morrow, R. (1967). The fast Fourier transform. IEEE spectrum, 4(12), 63-70.

Canny, J. (1986). *A computational approach to edge detection*. IEEE transactions on pattern analysis and machine intelligence (6), 679-698.

Cesarini, F., Marinai, S., & Soda, G. (2002). *Retrieval by layout similarity of documents represented with MXY trees Document Analysis Systems*. Springer, pp. 353-364.

Chellappa, R., & Chatterjee, S. (1985). *Classification of textures using Gaussian Markov random fields*. IEEE transactions on acoustics, speech, and signal processing, 33(4), pp. 959-963.

Chen, J., & Jain, A. K. (1988). *A structural approach to identify defects in textured images.* Paper presented at the International Conference on Systems, Man, and Cybernetics (ICSMC), pp. 29-32.

Chen, K., Wei, H., Hennebert, J., Ingold, R., & Liwicki, M. (2014). *Page segmentation for historical handwritten document images using color and texture features.* Paper presented at the International Conference on.Frontiers in Handwriting Recognition (ICFHR), pp. 488-493.

Conners, R. W., & Harlow, C. A. (1980). *A theoretical comparison of texture algorithms*. IEEE transactions on pattern analysis and machine intelligence (3), pp. 204-222.

Costa, A. F., Humpire-Mamani, G., & Traina, A. J. M. (2012). *An efficient algorithm for fractal analysis of textures.* Paper presented at the Conference on Graphics, Patterns and Images (SIBGRAPI), pp. 39-46.

Daisy, M. M. H., TamilSelvi, S., & Prinza, L. (2012). *Gray scale morphological operations for image retrieval.* Paper presented at the International Conference on Computing, Electronics and Electrical Technologies (ICCEET), pp. 571-575.

Dey, S., Nicolaou, A., Llados, J., & Pal, U. (2016). *Local binary pattern for word spotting in handwritten historical document.* Paper presented at the Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR), pp. 574-583.

Dixit, U. D., & Shirdhonkar, M. (2018). *Face-based Document Image Retrieval System*. Procedia Computer Science, 132, pp. 659-668.

Do, M. N., & Vetterli, M. (2002). *Contourlets: a directional multiresolution image representation.* Paper presented at the International Conference on Image Processing (ICIP), 1, pp. 357-360.

Fernández, A., Álvarez, M. X., & Bianconi, F. (2011). *Image classification with binary gradient contours*. Optics and Lasers in Engineering, 49(9), 1177-1184.

Fisher, Y. (2012). Boobk on "*Fractal image compression: theory and application*." Springer Science & Business Media.

Gonzalez, R. C., & Woods, R. E. (2005). Book on "*Digital Image Processing*": Prentice-Hall of India Pvt. Ltd.

Gordo, A., Gibert, J., Valveny, E., & Rusiñol, M. (2010). *A kernel-based approach to document retrieval.* Paper presented at the International Workshop on Document Analysis Systems, pp. 377-384.

Griffin, L. D. (2000). *Mean, median and mode filtering of images.* Paper presented at the Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, pp.295-3004.

Haar, A. (1910). *On the theory of orthogonal function systems.* Mathematische Annalen, 69(3), pp. 331-371.

Hafiane, A., Seetharaman, G., & Zavidovique, B. (2007*). Median binary pattern for textures classification Image Analysis and Recognition.* Springer, pp. 387-398.

Hangarge, M., Veershetty, C., Pardeshi, R., & Dhandra, B. (2016). *Gabor Wavelets Based Word Retrieval from Kannada Documents.* Procedia Computer Science, 79, pp. 441-448.

Haralick, R. M., Shanmugam, K., & Dinstein, I. H. (1973). *Textural features for image classification*. IEEE Transactions on Systems, Man and Cybernetics (6), pp. 610-621.

Heikkilä, M., Pietikäinen, M., & Schmid, C. (2009). *Description of interest regions with local binary patterns.* Pattern Recognition, 42(3), pp. 425-436.

Heilbronner, R. P. (1992). *The autocorrelation function: an image processing tool for fabric analysis.* Tectonophysics, 212(3-4), pp. 351-370.

Jafari-Khouzani, K., & Soltanian-Zadeh, H. (2005). *Radon transform orientation estimation for rotation invariant texture analysis.* IEEE transactions on pattern analysis and machine intelligence, 27(6), pp. 1004-1008.

Jin, H., Liu, Q., Lu, H., & Tong, X. (2004). *Face detection using improved LBP under bayesian framework.* Paper presented at International Conference on the Image and Graphics (ICIG), pp. 306-309.

Joshi, M. S., Bartakke, P. P., & Sutaone, M. (2009). *Texture representation using autoregressive models.* Paper presented at International Conference on the Advances in Computational Tools for Engineering Applications (ACTEA), pp. 386-390.

Kato, Z., Zerubia, J., & Berthod, M. (1999). *Unsupervised parallel image classification using Markovian models.* Pattern Recognition, 32(4), pp. 591-604.

Katsigiannis, S., Keramidas, E. G., & Maroulis, D. (2010). *A contourlet transform feature extraction scheme for ultrasound thyroid texture classification.* International Journ Engineering Intelligent Systems Electrical Engineering Communications, (IJEIS)18(3), pp. 171-189.

Kim, J. K., & Park, H. W. (1999). *Statistical textural features for detection of microcalcifications in digitized mammograms*. IEEE transactions on medical imaging, 18(3), 231-238.

Kise, K., Yin Wuotang, & Matsumoto, K. (2003). *Document image retrieval based on 2D density distributions of terms with pseudo relevance feedback.* Paper presented at the International Conference on Document Analysis and Recognition, pp. 488-492.

Kumar, J., Ye, P., & Doermann, D. (2014). *Structural similarity for document image classification and retrieval.* Pattern Recognition Letters, 43, pp.119-126.

Kumar, K. S., Kumar, S., & Jawahar, C. (2007). *On segmentation of documents in complex scripts.* Paper presented at the International Conference on Document Analysis and Recognition(ICDAR), pp. 1243-1247.

Lee, T. S. (1996). *Image representation using 2D Gabor wavelets*. IEEE transactions on pattern analysis and machine intelligence, 18(10), pp. 959-971.

Li, J., Fan, Z.-G., Wu, Y., & Le, N. (2009). *Document Image Retrieval with Local Feature Sequences*. International Conference on Document Analysis and Recognition (*ICDAR*), pp. 346-350.

Li, W., Zhang, D., & Xu, Z. (2002). *Palmprint identification by Fourier transform.* International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI), 16(04), pp. 417-432.

Lu, Y., & Do, M. N. (2006). *A new contourlet transform with sharp frequency localization.* Paper presented at the International Conference on Image Processing (ICIP), pp.1629-1632.

Mallat, S. G. (1989). *A theory for multiresolution signal decomposition: the wavelet representation*. IEEE Transactions on Pattern Analysis and Machine Intelligence,11(7), pp. 674-693.

Manish H Bharati, J Jay Liu, & John F MacGregor. (2004). *Image texture analysis: methods and comparisons.* Chemometrics and intelligent laboratory systems, 72(1), pp. 57-71.

Marinai, S., Miotti, B., & Soda, G. (2011). *Digital Libraries and Document Image Retrieval Techniques: A Survey*. Springer, (375), pp. 181-204.

Materka, A., & Strzelecki, M. (1998). *Texture analysis methods–a review.* Technical university of lodz, institute of electronics, COST B11 report, Brussels, pp. 9-11.

Mehri, M., Héroux, P., Gomez-Krämer, P., & Mullot, R. (2017). *Texture feature benchmarking and evaluation for historical document image analysis.* International Journal on Document Analysis and Recognition (IJDAR), pp. 1-35.

Mehri, M., Nayef, N., Héroux, P., Gomez-Krämer, P., & Mullot, R. (2015). *Learning Texture Features for Enhancement and Segmentation of Historical Document Images.* Paper presented at the International Workshop on Historical Document Imaging and Processing, pp. 47-54.

Melter, R. A. (1987). *Some characterizations of city block distance.* Pattern Recognition Letters, 6(4), pp. 235-240.

Nanni, L., Brahnam, S., & Lumini, A. (2010). *A local approach based on a Local Binary Patterns variant texture descriptor for classifying pain states.* Expert Systems with Applications, 37(12), pp.7888-7894.

Nanni, L., Lumini, A., & Brahnam, S. (2010). *Local binary patterns variants as texture descriptors for medical image analysis.* Artificial intelligence in medicine, 49(2), pp. 117-125.

Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). *Multiresolution gray-scale and rotation invariant texture classification with local binary patterns* IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(7), pp. 971-987.

Oliva, A., & Torralba, A. (2001). *Modeling the shape of the scene: A holistic representation of the spatial envelope.* International journal of computer vision, 42(3), pp. 145-175.

Patel, D., & Stonham, T. (1992). *Texture image classification and segmentation using RANK-order clustering.* Paper presented at the Pattern Recognition, 3, pp. 92-95.

Patil, M. S. S., Junnarkar, M. A., & Gore, M. D. (2014). *Study of Texture Representation Techniques.* Emerging Trends and Technology in Computer Science, 3(3), pp. 267-274.

Pirlo, G., Chimienti, M., Dassisti, M., Impedovo, D., & Galiano, A. (2014). A Layout-Analysis Based System for Document Image Retrieval! Mondo Digitale, pp, 2-17.

Piroi, F., Lupu, M., Hanbury, A., & Zenz, V. (2011). *CLEF-IP 2011: Retrieval in the Intellectual Property Domain.* Paper presented at the CLEF (notebook papers/labs/workshop).

Rogers, D. J., & Tanimoto, T. T. (1960). *A computer program for classifying plants.* Science, 132(3434), pp. 1115-1118.

Sauvola, J., & Kauniskangas, H. (1999). *MediaTeam document database II.* A CD-ROM collection of document images, University of Oulu Finland.

Schouten, B. A., & de Zeeuw, P. M. (1999). *Feature extraction using fractal codes.* Paper presented at the International Conference on Advances in Visual Information Systems, pp. 483-493.

Srinivasan, G., & Shobha, G. (2008). *Statistical texture analysis.* Paper presented at the Proceedings of world academy of science, engineering and technology, pp. 1264-1269.

Tan, X., & Triggs, B. (2010). *Enhanced local texture feature sets for face recognition under difficult lighting conditions.* IEEE Transactions on Image Processing, 19(6), pp.1635-1650.

Tomita, F., & Tsuji, S. (2013). *Computer analysis of visual textures* (Vol. 102): Springer Science & Business Media.

Van de Wouwer, G., Scheunders, P., & Van Dyck, D. (1999). *Statistical texture characterization from discrete wavelet representations. IEEE Transactions on Image Processing, 8*(4), pp. 592-598.

Vargas, J. F., Ferrer, M. A., Travieso, C., & Alonso, J. B. (2011). *Off-line signature verification based on grey level information using texture features.* Pattern Recognition, 44(2), pp. 375-385.

Wu, Q., Wang, J., Yang, C., Cui, G., & Yang, W. (2016). *Target recognition by texture segmentation algorithm.* Expert Systems with Applications, 46, pp. 394-404.

Wu, X., & Sun, J. (2009). *An effective texture spectrum descriptor.* Paper presented at the International Conference on Information Assurance and Security, pp. 361-364.

Zhenhua Guo, Lei Zhang, & Zhang, D. (2010). *A completed modeling of local binary pattern operator for texture classification.* IEEE Transactions on Image Processing, 19(6), pp. 1657-1663.