
**International Journal of Nanotechnology**

ISSN online: 1741-8151 - ISSN print: 1475-7435
https://www.inderscience.com/ijnt

# A dynamic programming approach for accurate content-based retrieval of ordinary and nano-scale medical images

Jinhong Sun, Liang Qi, Yinglei Song, Junfeng Qu, Mohammad R. Khosravi

**DOI:** 10.1504/IJNT.2023.10056470

**Article History:**

| | |
|---|---|
| Received: | 05 February 2021 |
| Last revised: | 14 April 2021 |
| Accepted: | 06 May 2021 |
| Published online: | 31 May 2023 |

# A dynamic programming approach for accurate content-based retrieval of ordinary and nano-scale medical images

Jinhong Sun, Liang Qi and Yinglei Song

School of Electronic and Information Sciences,
Jiangsu University of Science and Technology,
Zhenjiang, 212003, China
Email: 1446557714@qq.com
Email: 7978779@qq.com
Email: syinglei2013@163.com

Junfeng Qu

Department of Computer Science and Information Technology,
Clayton State University,
Morrow, GA 30260, USA
Email: junfengqu@clayton.edu

Mohammad R. Khosravi*

Department of Computer Engineering,
Persian Gulf University,
Bushehr, 999067, Iran
Email: m.khosravi@mehr.pgu.ac.ir
*Corresponding author

**Abstract:** Recently, with the explosive growth in the number of available medical images generated by medical imaging systems, content-based retrieval of medical images has become an important method for the diagnosis and study of many diseases. Most existing methods find medical images similar to a given one based on the extraction and comparison of crucial image features. However, similarity values computed with low level visual features of an image generally do not match the similarity obtained from human observation well. The overall performance of these methods is thus often unsatisfactory. This paper proposes a dynamic programming approach for content-based retrieval of medical images. The approach represents an image with three different histograms that contain both crucial intensity and textural features of the image. The similarity between two images is evaluated with a dynamic programming approach that can optimally align the peaks in the corresponding histograms from both images. Experiments show that the proposed approach is able to generate retrieval results with high accuracy. A comparison with state-of-the-art approaches for content-based medical image retrieval shows that the proposed approach can achieve higher retrieval accuracy in both ordinary and nano-scale medical images. As a result, higher retrieval accuracy may lead to more reliable results for the diagnosis and treatment of many diseases. The proposed approach is thus potentially useful for improving the reliability of many applications in health informatics.

**Keywords:** medical image retrieval; similarity; alignment of histograms; dynamic programming; intensity features; textual features.

**Reference** to this paper should be made as follows: Sun, J., Qi, L., Song, Y., Qu, J. and Khosravi, M.R. (2023) 'A dynamic programming approach for accurate content-based retrieval of ordinary and nano-scale medical images', *Int. J. Nanotechnol.*, Vol. 20, Nos. 1/2/3/4, pp.75–97.

**Biographical notes:** Jinhong Sun is currently a graduate student in the School of Electronics and Information Science at Jiangsu University of Science and Technology, China. Her research interests include information hiding and medical image processing.

Liang Qi received his PhD in Electrical Engineering from Jiangsu University, China in 2018. He is currently with Jiangsu University of Science and Technology, China. His research interests include algorithms for image processing, machine learning, intelligent control and its applications in mechanical engineering.

Yinglei Song received his PhD in computer science from the University of Georgia, USA in 2006. He worked as an Assistant Professor of Computer Science at the University of Maryland Eastern Shore, USA from 2007 to 2012. He is currently with Jiangsu University of Science and Technology, China. His research interests include algorithms for image processing, machine learning, bioinformatics and data mining.

Junfeng Qu received his PhD in Computer Science from the University of Georgia, USA in 2006 and joined the Department of Information Technology and Computer Science at Clayton State University as an Assistant Professor in that year. He is currently working as a Professor of Computer Science at the Clayton State University, USA. His research interests include image processing, information hiding, machine learning and bioinformatics.

Mohammad R. Khosravi received the BSc, MSc and PhD degrees in Electrical Engineering with expertise in communications and signal processing from Shiraz University of Technology, Shiraz, Iran, in 2013, 2015, and 2020, respectively. He is currently with the Department of Computer Engineering, Persian Gulf University, Bushehr. His main interests include statistical signal and image processing, medical bioinformatics, radar imaging and satellite remote sensing, and computer communications.

## 1 Introduction

In the past decades, the rapid development of various medical imaging systems has led to the generation of a large number of medical images. In practice, these medical imaging data can provide useful information for the diagnosis, study and treatment of many diseases [1,2]. For example, a database that contains the medical images for different cases of a disease can be constructed. For images collected from a new case of the same disease, the database can be searched to find images similar to them. The information of these similar images can probably provide insights into the diagnosis and treatment of the

new case [3]. In such applications, the similarity of two medical images is often automatically evaluated by a computer program based on the contents of them. Content-based medical image retrieval (CBMIR) thus has become an important problem in medical image processing.

A generalised problem of CBMIR is the problem of content-based image retrieval (CBIR). CBIR searches databases for images similar to a given image based on the similarities between the given image and images in the databases. The similarity of two images is generally computed with an algorithm and is crucial for the accuracy of retrieval. To improve the accuracy of CBIR, a large number of approaches have been proposed to accurately compute the similarity of two images [4–11]. In Sikha and Soman [4], visual saliency based features are combined with texture and color features for similarity evaluation. The work in Raghavan and John [5] proposes a hybrid approach that utilises enhanced first type of pessimistic covering based lower approximation multi-granular rough sets for CBIR. In Ghodratnama and Abrishami [6], a new feature weighting approach is combined with C-means clustering to improve the accuracy of CBIR. In Al-Mohamade et al. [7], a novel multiple query retrieval approach based on visual feature discrimination is proposed for computing the similarity of two images. The work in Nikolaos et al. [8] proposes a regularised discriminative deep metric learning method to improve the accuracy of CBIR. The proposed method learns a representation that can encode the latent generative factors for each image class. In Buvana et al. [9], a number of different features, including the histogram of oriented gradients (HOG) features, local binary pattern (LBP) and gray-level co-occurrence matrix (GLCM) are combined to improve the retrieval accuracy. In Ng et al. [10], it is shown that lower layers in convolutional neural networks (CNNs) are generally more important than the last layers. A novel approach is thus proposed for extracting convolutional features from different layers in CNNs for CBIR. In Tzelepi and Tefas [11], a model retraining method is proposed to obtain more efficient convolutional representations for CBIR. The feature representations are computed from the activations of the convolutional layers in a deep CNN model with max-pooling.

Numerous computational methods have been developed for CBMIR [12–18]. Currently, most of the existing computational methods come from three different categories. They are methods based on local features, visual words dictionary and deep learning techniques respectively [19].

For methods based on local features, the crucial local features are first obtained from images and the similarity between two images is evaluated based on the values of their local features. A distance metric is often utilised to evaluate the similarity between the feature vectors of two images. For example, in Srinivas et al. [20], mean and variance of pixel intensities are used as local features to evaluate the similarities of images. In Tao [21], PA-AP chest images are retrieved with a multi-level learning-based algorithm that uses local features of images. In Shamna et al. [22], topic and location models are used to construct an automatic medical image retrieval system. In Ahmed [23], a new retrieval method based on relevance feedback is proposed for CBMIR. The feedback mechanism utilises the voting values of each class in the image repository. In Qazanfari et al. [24], CBMIR is performed with a short term learning approach based on a collection of low level image features. The work in Kashif et al. [25] improves the retrieval accuracy of lung diseases with a new approach that combines local ternary pattern (LTP), local phase

quantisation (LPQ), and discrete wavelet transform. In Renita and Christopher [26], a new approach based on grey wolf optimisation-support vector machine (GWO-SVM) is developed for CBMIR. In Shinde et al. [27], a flexible directional filter bank (DFB) is proposed for CBMIR. The parameters of the Hermite transform in the DFB can be adaptively tuned for improved retrieval accuracy.

Methods based on visual words dictionary use the bag-of-visual-words (BoVW) technique to generate features for a medical image. In Zhang et al. [28], a retrieval approach based on a pruned dictionary is proposed to obtain the features of medical images for the evaluation of similarities. In Shamna et al. [29], visual spatial word matching is used in the proposed CBMIR framework to compare medical images. The work in Banerjee et al. [30] proposes a retrieval method that combines high level semiconductor words with low level image features to achieve improved accuracy. In Karamti et al. [31], a vectorisation technique is combined with a pseudo-relevance model to construct a framework for CBMIR. In Torjmen-Khemakhem and Gasmi [32], a relevance feedback approach is proposed for CBMIR, the approach selects the most important context features for evaluating the similarities of medical images. The work in Nair et al. [33] combines canonical correlation analysis (CCA) and fuzzy C means clustering to improve the computational efficiency of CBMIR.

Recently, deep learning based approaches have been extensively applied to the retrieval of medical images. Deep learning approaches provide neural network based models that can effectively describe the relationships between the visual local features and the semantic features of a medical image [34]. A large number of deep learning based methods thus have been proposed for CBMIR. For example, in Sun et al. [35], a novel deep learning method is proposed to retrieve medical images with improved accuracy in integrated RIS/PACS. In Cai et al. [36] and Qayyum et al. [37], deep learning frameworks based on convolutional neural networks (CNNs) are developed for CBMIR. The CNNs used in the proposed frameworks are trained for the classification of medical images. Medical images are retrieved with the features and the classification results generated with the CNNs. The work in Yang et al. [38] considers the multi-domain medical image retrieval problem and constructs a deep learning based single multi-domain medical image retrieval (MIR) model to effectively integrate the knowledge from multiple specialist MIR models. The work in Haripriya and Porkodi [39] proposes a parallel deep convolutional neural network (PDCNN) model to combine deep convolutional neural network (DCNN) features, low level content features, high level semantic features and compact features together to achieve improved accuracy for CBMIR applications.

Although significant progresses have been achieved in the development of computational approaches for accurate CBIR and CBMIR, a number of challenges remain. Firstly, although deep learning based methods have provided an effective approach to the extraction of accurate high level semantic features of an image, the semantic gap between the high-level semantic features and low level local features of images remains a major obstacle to the further improvement in retrieval accuracy [40]. Secondly, due to the fact that the feature extraction methods in many state-of-the-art (SOTA) approaches for CBIR are initially designed for image classification, the scalability of these approaches on various target datasets is often not satisfactory and cannot be guaranteed [41]. Thirdly, many feature extraction methods often generate high dimensional feature vectors for computing the similarity of two images. A large amount

of computation time is usually required for the computation and comparison of such feature vectors. A trade-off between computational efficiency and retrieval accuracy therefore exists for most of the SOTA approaches for CBIR [42]. New approaches are thus desirable to overcome the challenges and further improve the performance of CBIR. In particular, certain important fine details in images may need to be considered collectively when retrievals of nano-scale medical images are performed, an approach that can accurately compare such fine details in two medical images is thus crucial for improving the accuracy of CBMIR.

In this paper, we develop a new approach for accurate content-based retrieval of medical images. Due to the fact that more accurate results of retrieval would lead to improved diagnosis accuracy in health informatics [43,44], the proposed approach is aimed at improving the retrieval accuracy. The approach is unsupervised and compares both the low level and high level features of medical images with a dynamic programming approach. For each image, three different histograms are obtained to represent both the low level and high level features. To compute the similarity of two images, all possible alignments for each pair of corresponding histograms are considered and the one that minimises the distance between the two histograms is selected with dynamic programming. The similarity between the corresponding histograms is computed from their optimal alignment. The overall similarity value of two images is a weighted combination of the similarity values obtained on the three histogram pairs associated with the images.

A histogram provides a simple representation of both high level and low level features in an image. Specifically, a histogram describes the statistical distribution of a given local feature within an image and can be considered as a feature vector that contains both high level and local information in the image. An appropriate comparison of histograms constructed from two images can thus generate important information on their similarity. The proposed approach uses a dynamic programming method to compute an optimal alignment of the structures in two histograms. The resulting optimal alignment compares high level features and low level ones together and can potentially alleviate problems that may arise from the semantic gap. Due to this fact, the proposed approach may lead to a more accurate evaluation of similarity.

On the other hand, it is clear that the size of a histogram only depends on the possible values of the given local feature and is independent of the size of an image. The proposed approach can thus be applied to various target datasets without the scalability problem. In addition, the dynamic programming procedure can be efficiently performed in constant time. This fact implies that the computational efficiency can be guaranteed in cases where the histogram of each image in a searched database has been computed in a preprocessing stage.

The proposed approach is tested with the Kvasir dataset [45] and the hela dataset [46]. The Kvasir dataset is a dataset of ordinary medical images and the hela dataset is composed of nano-scale medical images. The performance of the approach in both precision and recall is compared with that of SOTA methods for CBMIR. Experimental results show that, compared with other SOTA methods, the proposed approach is able to achieve improved search accuracy for retrieval on both ordinary and nano-scale medical image datasets. The experimental results thus suggest that the proposed approach is able to capture crucial low level and high level features of a medical image and it thus can

possibly be applied for CBMIR tasks in large databases of both ordinary and nano-scale medical images.

## 2 The proposed approach

The proposed approach computes the similarity of two images in two steps. In the first step, all pixels in both images are processed and two local features are extracted for each pixel with two linear convolutional operators. Since previous research on CNN has shown that local features obtained with linear convolutional operators can effectively improve the accuracy of CBMIR [35–39], histograms obtained from such local features may contain crucial information in an image. For each image, three histograms are constructed to describe the statistical distributions of the two local features and intensities of all pixels in the image.

In the second step, each histogram constructed from one image is compared with the corresponding one from the other image. The goal of the comparison is to recognise and align similar structures in two histograms to the largest extent. A similarity score is then computed based on the result of alignment. Since similar structures may appear in different positions in two histograms, the comparison is performed with a dynamic programming approach similar to the algorithm used for optimal sequence alignment in bioinformatics [47].

Dynamic programming methods have been extensively used to efficiently compute optimal solutions for combinatorial optimisation problems from a variety of areas, such as speech recognition [48], image processing [49] and bioinformatics [50] etc. For example, dynamic programming methods have been extensively used in bioinformatics for the analysis, comparison and alignment of biological sequences [50]. A dynamic programming approach is recently proposed in Li et al. [49] for real-time enhancement of colour images. In Li et al. [51], dynamic programming is utilised for real-time classification of brain tumours. In general, dynamic programming methods can efficiently find exact solutions for many optimisation problems and can thus guarantee both the accuracy and computational efficiency for many image processing applications in practice.

### *2.1 Extraction of image features*

Let $I$ be the intensity map of a medical image with $p$ rows and $q$ columns. For each pair of integers $i$ and $j$ that satisfy $1 \le i \le p$ and $1 \le j \le q$, $I(i, j)$ denotes the intensity of the pixel in row $i$ and column $j$. The intensity value of a pixel is generally an integer between 0 and 255. For each given intensity value $l$ between 0 and 255, $C(I,l)$ is the number of pixels with an intensity value of $l$ in $I$. $R_i(I,l) = C(I,l) / pq$ is thus the ratio of pixels with an intensity value of $l$ in $I$. A histogram $H_m$ can be constructed for $I$ by combining the values of $R_i(I,l)$'s for all possible values of $l$. Specifically, $H_m$ is defined as follows.

$$H_m = \bigcup_{0 \le l \le 255} \{R_i(I,l)\} \tag{1}$$

For the pixel at row $i$ and column $j$ in $I$, four different values $D_l(i,j)$, $D_r(i,j)$, $D_u(i,j)$ and $D_d(i,j)$ for intensity difference can be computed as follows if both $1 < i < p$ and $1 < j < q$ hold.

$$D_l(i,j) = | I(i,j) - I(i,j-1) | \tag{2}$$

$$D_r(i,j) = | I(i,j) - I(i,j+1) | \tag{3}$$

$$D_u(i,j) = | I(i-1,j) - I(i,j) | \tag{4}$$

$$D_d(i,j) = | I(i+1,j) - I(i,j) | \tag{5}$$

Consider the values of $D_l(i,j)$, $D_r(i,j)$, $D_u(i,j)$ and $D_d(i,j)$ for all integer pairs $(i,j)$ that satisfy $1 < i < p$ and $1 < j < q$. It is clear that each value is an integer between 0 and 255, a histogram $H_d$ that describes the probability for each integer between 0 and 255 to appear in these values can be obtained. Specifically, let $R_v(I,l)$ denote the probability for integer $l$ ($0 \le l \le 255$) to appear in these values, $H_d$ can be constructed as follows.

$$H_d = \bigcup_{0 \le l \le 255} \{R_v(I,l)\} \tag{6}$$

Another integer feature value can be defined for the pixel at row $i$ and column $j$ in $I$, if all neighbouring pixels that surround the pixel are considered. Let $M_n(i,j)$ be the mean intensity value of all pixels that surround the pixel at row $i$ and column $j$. If both $1 < i < p$ and $1 < j < q$ hold, $M_n(i,j)$ can be computed with equation (7) as follows.

$$M_n(i,j) = \left\lfloor \frac{\sum_{s=-1}^{1} \sum_{t=-1}^{1} I(i+s, j+t) - I(i,j)}{8} \right\rfloor \tag{7}$$

The difference between $M_n(i,j)$ and $I(i,j)$ is an important local feature for the pixel. Let $D_n(i,j)$ be the difference and $D_n(i,j)$ can be computed as follows.

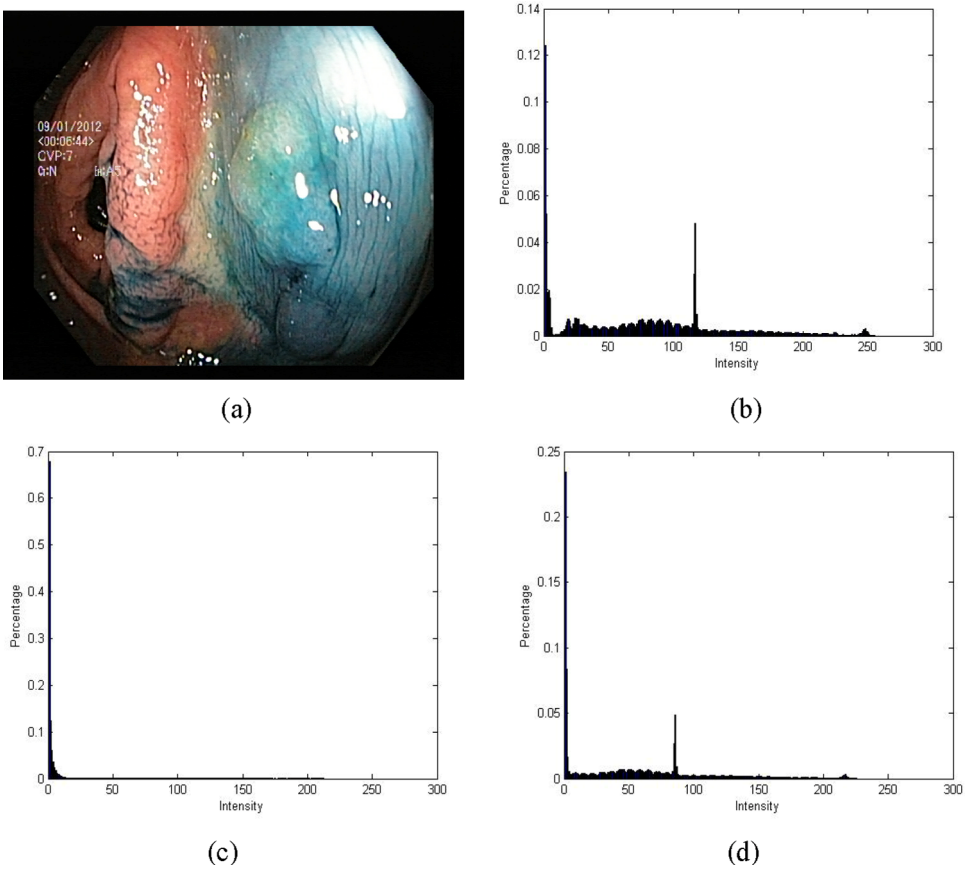
$$D_n(i,j) = | I(i,j) - M_n(i,j) | \tag{8}$$

Since $0 \le D_n(i,j) \le 255$ holds for any integer pair $(i,j)$ that satisfies $1 < i < p$ and $1 < j < q$, a histogram $H_n$ can be constructed to model the distribution of $D_n(i,j)$'s for all integer pairs $(i,j)$ that satisfy $1 < i < p$ and $1 < j < q$. For each integer $l$ between 0 and 255, $R_n(I,l)$ is the probability for $l$ to appear as the value of $D_n(i,j)$ for a pixel at $(i,j)$. $H_n$ can be obtained as shown in equation (9).

$$H_n = \bigcup_{0 \le l \le 255} \{R_n(I,l)\} \tag{9}$$

It is evident that $H_m$ represents the high level intensity features of a medical image, while both $H_n$ and $H_d$ represent combinations of high level features and low level textural features. Figure 1 provides an example of a medical image and the histograms obtained based on the example image.

**Figure 1** (a) shows an example of a medical image while (b)–(d) show the histograms obtained with equations (1), (6) and (9) respectively (see online version for colours)




## 2.2 *Alignment of histograms*

The similarity of two histograms can be compared with an alignment approach similar to the method used for the alignment of biological sequences in bioinformatics. Let $H_1$ and $H_2$ be two histograms. Each of $H_1$ and $H_2$ contains the probability value for each integer between 0 and 255. To simplify the notation, we use $H_i(l)$ ($i = 1, 2$) to denote the probability value associated with integer $l$ in $H_i$. An *alignment* $A(H_1, H_2)$ between $H_1$ and $H_2$ is a set of integer pairs defined as follows.

$$A(H_1, H_2) = \{(l_1, m_1), (l_2, m_2), ...(l_k, m_k)\} \tag{10}$$

where $k$ is the number of aligned integer pairs in the alignment and $0 \le l_s < l_t \le 255$ and $0 \le m_s < m_t \le 255$ hold for all integers $s$ and $t$ that satisfy $1 \le s \le t \le k$.

An alignment selects $k$ probability values from each histogram and form $k$ pairs based on the selected probability values. The probability values in each pair are considered to be related, a distance measure can be computed for an alignment based on

the paired probability values in the alignment. The distance measure $d(A)$ for the alignment defined in equation (10) can be computed as follows.

$$d(A) = \sum_{u=1}^{k} (H_1(l_u) - H_2(m_u)) \ln \frac{H_1(l_u)}{H_2(m_u)} + 2(256-k)P \tag{11}$$

where $P > 0$ is the penalty value for an unpaired probability value. The penalty term is intended to increase the distance between two histograms when there are unpaired probability values in the alignment. Clearly, the factor $2(256-k)$ in the term is the number of unpaired probability values in the alignment. A larger number of unpaired probability values in an alignment thus would lead to a larger distance value for the alignment.

A closer analysis of the first term in equation (11) shows that, in the case where $H_1(l_u) > H_2(m_u)$, $H_1(l_u) - H_2(m_u) > 0$ and $\ln H_1(l_u) / H_2(m_u) > 0$ both hold, the product of them is a positive number. On the other hand, when $H_1(l_u) < H_2(m_u)$, $H_1(l_u) - H_2(m_u) < 0$ and $\ln H_1(l_u) / H_2(m_u) < 0$ both hold, the product of them remains positive. It is thus clear from equation (11) that $d(A) \geq 0$ holds for any alignment $A(H_1, H_2)$ and $d(A) = 0$ holds only in the case where $k = 256$ and $H_1(l_u) = H_2(m_u)$ holds for each integer $u$ between 1 and $k$. In other words, the distance measure of an alignment defined in equation (11) is nonnegative and its value is zero only when all probability values are paired in the alignment and all paired probability values in the alignment are identical. The distance defined in equation (11) thus provides a valid measure for the similarity between two aligned histograms.

### *2.3 Alignment with dynamic programming*

Given two histograms $H_1$ and $H_2$, a large number of different alignments exist between them. To evaluate the similarity between $H_1$ and $H_2$, the alignment with the minimum distance can provide crucial information on probability values that are most likely related in $H_1$ and $H_2$. The paired probability values in the alignment can thus be utilised to compute a similarity value for $H_1$ and $H_2$.

For a given nonnegative integer $x$ between 0 and 255, $H_i[0, x]$ ($i = 1, 2$) denotes the probability values associated with all integers between 0 and $x$ in $H_i$. Based on the distance measure defined in equation (11), a dynamic programming algorithm can be employed to compute the alignment that minimises the distance measure. Specifically, two $256 \times 256$ tables $M$ and $T$ are maintained for the dynamic programming. $M$ is the table used to store the intermediate results during the dynamic programming and $T$ stores the information needed to trace back and obtain the paired probability values. For integers $x$ and $y$ that satisfy $0 \leq x \leq 255$ and $0 \leq y \leq 255$, $M(x, y)$ stores the minimum distance between the probability values in $H_1[0, x]$ and those in $H_2[0, y]$. In the case where $x > 0$ and $y > 0$ both hold, one of the following four cases holds for an alignment between probability values in $H_1[0, x]$ and those in $H_2[0, y]$.

1 None of $H_1(x)$ and $H_2(y)$ are paired in the alignment.

2 $H_1(x)$ is paired but $H_2(y)$ is not paired in the alignment.

3 $H_2(y)$ is paired but $H_1(x)$ is not paired in the alignment.

4 $H_1(x)$ is paired with $H_2(y)$ in the alignment.

Based on the above four cases, since the distance between two alignments is computed with equation (11), $M(x,y)$ is related to $M(x-1,y-1)$, $M(x-1,y)$ and $M(x,y-1)$ with the following recursion relation when $x>0$ and $y>0$ both hold.

$$M(x,y)=\min\begin{Bmatrix} M(x-1,y-1)+2P \\ M(x-1,y)+P \\ M(x,y-1)+P \\ M(x-1,y-1)+s(x,y) \end{Bmatrix} \tag{12}$$

where $P$ is the penalty for the generation of an unpaired probability value and $s(x,y)$ can be computed as follows.

$$s(x,y)=(H_1(x)-H_2(y))\ln\frac{H_1(x)}{H_2(y)} \tag{13}$$

In the case where $x=0$ and $y\geq 0$, $M(0,y)$ can be computed with equation (14).

$$M(0,y)=\min\begin{Bmatrix} (y+2)P \\ yP+s(0,y) \end{Bmatrix} \tag{14}$$

where $s(0,y)$ can be computed with equation (13). Similarly, in the case where $x>0$ and $y=0$, $M(x,0)$ can be computed with equation (15).

$$M(x,0)=\min\begin{Bmatrix} (x+2)P \\ xP+s(x,0) \end{Bmatrix} \tag{15}$$

When $x>0$ and $y>0$, the value of $T(x,y)$ is set to indicate which value on the right hand side of equation (12) is assigned to $M(x,y)$. Similarly, when $x=0$ and $y\geq 0$, $T(0,y)$ is set to indicate which value on the right hand side of equation (14) is assigned to $M(0,y)$; in the case where $x>0$ and $y=0$, $T(x,0)$ is set to indicate which value on the right hand side of Equation (15) is assigned to $M(x,0)$. The information in $T$ is used later for the determination of paired probability values in the optimal alignment.

After all elements in have been determined, the algorithm uses a tracing back procedure to determine the probability values that have been paired in the optimal alignment. The tracing back procedure starts with the information in $T(255,255)$ and computes the paired probability values based on the following steps.

1 Set $a=255$ and $b=255$, initialise set $A$ to be empty;

2 Include pair ($H_1(a)$, $H_2(b)$) into set $A$ if the value of $T(a,b)$ suggests that a pair is formed between $H_1(a)$ and $H_2(b)$ in the optimal alignment.

3 If one of $a$ and $b$ is 0, go to step 5.

4 Update the values of $a$ and $b$ based on the value of $T(a,b)$, then go to step 2.

5 Output $A$ as the set of paired probability values.

### *2.4 Evaluation of similarity*

The similarity between two histograms $H_1$ and $H_2$ can be evaluated based on the optimal alignment $A(H_1, H_2)$ obtained from the dynamic programming approach. Let $(l_1, m_1)$, $(l_2, m_2)$, …, $(l_k, m_k)$ be the pairs of probability values in $A(H_1, H_2)$, two vectors $V_1$ and $V_2$ with $k$ components can be constructed from $A(H_1, H_2)$ as shown in equations (16) and (17).

$$V_1 = (H_1(l_1), H_1(l_2), ..., H_1(l_k)) \tag{16}$$

$$V_2 = (H_2(m_1), H_2(m_2), ..., H_2(m_k)) \tag{17}$$

The similarity of $V_1$ and $V_2$ can be evaluated based on their inner product as follows.

$$l(V_1, V_2) = \frac{\sum_{t=1}^{k} H_1(l_t) H_2(m_t)}{\sqrt{\sum_{i=1}^{k} H_1^2(l_i)} \sqrt{\sum_{j=1}^{k} H_2^2(m_j)}} \tag{18}$$

where the numerator is the inner product of $V_1$ and $V_2$, the denominator is the product of the norms of $V_1$ and $V_2$. Geometrically, $l(V_1, V_2)$ is the cosine of the angle between $V_1$ and $V_2$. $l(V_1, V_2)$ is thus a positive number between 0 and 1. $l(V_1, V_2)$ is 1 when $V_1$ and $V_2$ are vectors along the same direction, which implies that there is a common factor between each pair of the corresponding components in $V_1$ and $V_2$. The aligned part of two histograms thus have the same structure in the case where $l(V_1, V_2)$ is 1. In addition, a lower value of $l(V_1, V_2)$ implies a larger angle between $V_1$ and $V_2$, which indicates a lower similarity between the aligned parts of the histograms.

In addition to the similarity between $V_1$ and $V_2$, the total percentages of aligned probability values in the alignment is also considered in the evaluation of the similarity between $H_1$ and $H_2$. Let $p_a(H_1)$ and $p_a(H_2)$ denote the total percentages of aligned probability values in and respectively. It is clear that $p_a(H_1)$ and $p_a(H_2)$ can be computed as follows.

$$p_a(H_1) = \sum_{i=1}^{k} H_1(l_i) \tag{19}$$

$$p_a(H_2) = \sum_{i=1}^{k} H_2(m_i) \tag{20}$$

The similarity between and can thus be evaluated based on $p_a(H_1)$, $p_a(H_2)$ and $l(V_1, V_2)$ as shown in equation (21).

$$s(H_1, H_2) = \frac{(p_a(H_1) + p_a(H_2)) l(V_1, V_2)}{2} \tag{21}$$

For two images $I_1$ and $I_2$, three histograms are obtained for each of them to represent the high level and low level texture features together. Let $H_m^1$, $H_d^1$ and $H_n^1$ be the histograms obtained on $I_1$ and $H_m^2$, $H_d^2$ and $H_n^2$ be the corresponding histograms obtained on $I_2$. The similarity between and can be evaluated as follows.

$$t(H_1, H_2) = w_1 s(H_m^1, H_m^2) + w_2 s(H_d^1, H_d^2) + w_3 s(H_n^1, H_n^2) \quad (22)$$

where $s(H_m^1, H_m^2)$, $s(H_d^1, H_d^2)$ and $s(H_n^1, H_n^2)$ are similarity values computed based on the dynamic programming approach presented in Subsection 2.3 and equation (21); and positive constant relative weight values that satisfy the following constraint.

$$w_1 + w_2 + w_3 = 1 \quad (23)$$

### *2.5 Computational complexity*

Given two images $I_1$ and $I_2$ that need to be compared for a similarity value, it is straightforward to see that the computation of the three histograms for each image needs linear computation time. Specifically, the amount of computation time needed to preprocess the images to obtain the histograms that are to be aligned is $O(p_1 q_1 + p_2 q_2)$, where $p_1$ and $q_1$ are the numbers of rows and columns in $I_1$, $p_2$ and $q_2$ are the numbers of rows and columns in $I_2$.

The dynamic programming approach needs constant computation time since only 256 different intensity values are considered for each histogram. The computation time needed to compute the similarity of two images is thus $O(1)$ if the histograms of both images have been available. In practice, the histograms of medical images in a database can be computed and stored in the database for retrieval. The retrieval process can directly use these histograms and thus only need to perform the dynamic programming approach to compute similarity values. The proposed approach is thus computationally efficient.

## 3 Experimental results

The proposed approach has been implemented into a computer program DCBMIR in MATLAB and its performance is tested with the Kvasir dataset [45] and the hela dataset [46]. The hela dataset is a dataset of nano-scale medical images. Both datasets consist of images from a number of different classes. For all experiments, DCBMIR is applied to a dataset to search for images that are similar to a given image from the dataset. To evaluate the accuracy of the search results, images from the same class are considered to be similar and should have similarity values higher than those obtained on images from different classes. The performance of DCMIR is compared with that of a number of SOTA methods that utilise different models for medical image retrieval. These methods include RFRM [23], CNNSH [36], DCNN [37], VPDML [8], DMD [4], GWOSVM [26] and PDCNN [39]. RFRM is a retrieval method based on relevance feedback. In RFRM, a number of important images are labelled as positive feedback and some insignificant ones are labelled as negative feedback. The retrieval results are refined based on the labelling of these images. CNNSH, DCNN, VPDML and PDCNN are deep learning based retrieval methods. GWOSVM retrieves medical images with a model based on support vector machine and DMD uses a dynamic mode decomposition framework for retrieval.

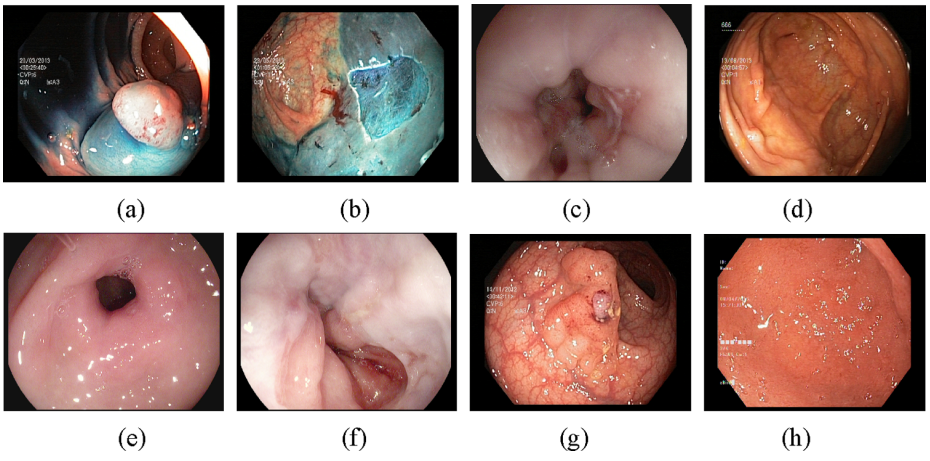
### *3.1 Experiments on the Kvasir dataset*

The Kvasir dataset contains 4000 medical images in total and the images in the dataset are classified into 8 different classes. Each class in the dataset contains 500 images. A detailed description of the images in the dataset is shown in Table 1. Figure 2 shows an example image for each class in the dataset. For each class, 10 images are randomly selected and each selected image is queried in a collection of images from the dataset with the proposed approach to find the images that are most similar to it.

**Table 1** Some information on the Kvasir dataset used for the testing of the proposed approach

| *Class number* | *Class name* | *Number of images* | *Image size* |
|---|---|---|---|
| 1 | Dyed-lifted-polyps | 500 | 720 × 576 |
| 2 | Dyed-resection-margins | 500 | 720 × 576 |
| 3 | Esophagitis | 500 | 1280 × 1024 |
| 4 | Normal-cecum | 500 | 720 × 576 |
| 5 | Normal-pylorus | 500 | 1280 × 1024 |
| 6 | Normal-z-line | 500 | 1280 × 1024 |
| 7 | Polyps | 500 | 720 × 576 |
| 8 | Ulcerative-colitis | 500 | 720 × 576 |

**Figure 2** (a)–(h) are the examples of images from classes 1–8 in the Kvasir dataset respectively (see online version for colours)



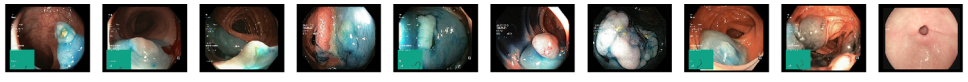

DCBMIR returns a number of images that have the highest similarity values with the queried one. The number of images returned is considered to be an integer parameter $c$ and can be set by the user. The performance of the proposed approach on the Kavsir dataset is compared with that of RFRM [23], CNNSH [36], DCNN [37], VPDML [8], DMD [4], GWOSVM [26] and PDCNN [39]. The accuracy of each approach is evaluated based on precision and recall. The precision of a search result is the percentage of related images that have been included in the search result. The recall of search result is the percentage of related images in the returned images. The precision $P_r$ and recall $R_e$ for a given result can be computed based on equations (24) and (25).

$$P_r = \frac{C}{T_r} \tag{24}$$

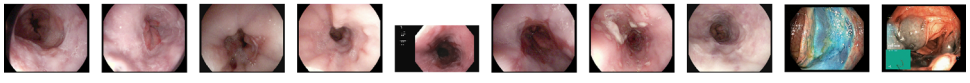
$$R_e = \frac{C}{N} \tag{25}$$

where $C$ is the number of images related to the queried image in the result, $T_r$ is the total number of images related to the queried image and $N$ is the number of images in the result.

**Figure 3** Examples of the searched results generated by the proposed approach on the Kvasir dataset: (a) shows the 10 images returned for the query of an image from class 1 and (b) shows the 10 images returned for the query of an image from class 3 (see online version for colours)



(a)



(b)

**Table 2** The mean and standard deviations (std.) of precision obtained by DCBMIR on each class in the Kvasir dataset when 500, 1000, 1500, 2000, 2500, and 3000 images are generated in a retrieval result

| *Class* | | $c = 500$ | $c = 1000$ | $c = 1500$ | $c = 2000$ | $c = 2500$ | $c = 3000$ |
|---|---|---|---|---|---|---|---|
| 1 | Mean | 0.7558 | 0.9117 | 0.9417 | 0.9667 | 0.9775 | 0.9833 |
| | Std. | 0.1287 | 0.0948 | 0.0752 | 0.0481 | 0.0423 | 0.0296 |
| 2 | Mean | 0.6675 | 0.8175 | 0.8800 | 0.9100 | 0.9325 | 0.9550 |
| | Std. | 0.1301 | 0.1280 | 0.1250 | 0.1119 | 0.0977 | 0.0724 |
| 3 | Mean | 0.6258 | 0.7108 | 0.7642 | 0.8025 | 0.8300 | 0.8608 |
| | Std. | 0.2755 | 0.2872 | 0.2711 | 0.2390 | 0.2234 | 0.2015 |
| 4 | Mean | 0.7250 | 0.8367 | 0.8875 | 0.9400 | 0.9675 | 0.9858 |
| | Std. | 0.1899 | 0.1475 | 0.1196 | 0.0889 | 0.0558 | 0.0243 |
| 5 | Mean | 0.4000 | 0.4800 | 0.5325 | 0.5875 | 0.6575 | 0.7125 |
| | Std. | 0.2395 | 0.2526 | 0.2571 | 0.2559 | 0.2687 | 0.2554 |
| 6 | Mean | 0.6206 | 0.6813 | 0.7138 | 0.7362 | 0.7650 | 0.7963 |
| | Std. | 0.2623 | 0.2735 | 0.2761 | 0.2757 | 0.2738 | 0.2657 |
| 7 | Mean | 0.6650 | 0.8425 | 0.8825 | 0.9175 | 0.9675 | 0.9825 |
| | Std. | 0.0844 | 0.1290 | 0.1217 | 0.0990 | 0.0613 | 0.0467 |
| 8 | Mean | 0.6217 | 0.8083 | 0.8842 | 0.9242 | 0.9608 | 0.9800 |
| | Std. | 0.1968 | 0.2244 | 0.1992 | 0.1417 | 0.0924 | 0.0534 |
| Overall | Mean | 0.6352 | 0.7611 | 0.8108 | 0.8481 | 0.8823 | 0.9070 |
| | Std | 0.1071 | 0.1354 | 0.1349 | 0.1306 | 0.1191 | 0.1053 |

Figure 3 shows two examples of the query results generated by DCBMIR. In Figure 3(a), an image from class 1 is queried in the Kvasir dataset and nine out of 10 returned images are from class 1. In Figure 3(b), an image from class 3 is queried and eight out of 10 returned images are from class 3. Tables 2 and 3 show the mean and standard values of precision and recall of DCBMIR respectively on each class in cases where a different number of images are generated in the retrieval results. The number of generated images (parameter $c$) is set to be 500, 1000, 1500, 2000, 2500, and 3000 in the experiments. It can be seen from the tables that the precisions become higher and the recalls become lower when more images are generated in the retrieval result, Figure 4 shows the relationships between the precisions and recalls obtained with DCBMIR and the other SOTA retrieval methods. It is clear from the figure that DCBMIR outperforms the other SOTA retrieval methods in accuracy on the Kvasir dataset since a higher overall precision can be achieved by DCBMIR under a given value of recall.

**Table 3** The mean and standard deviations (std.) of recall obtained by DCBMIR on each class in the Kvasir dataset when 500, 1000, 1500, 2000, 2500, and 3000 images are generated in a retrieval result

| *Class* | | $c = 500$ | $c = 1000$ | $c = 1500$ | $c = 2000$ | $c = 2500$ | $c = 3000$ |
|---|---|---|---|---|---|---|---|
| 1 | Mean | 0.7558 | 0.4558 | 0.3139 | 0.2417 | 0.1955 | 0.1639 |
| | Std. | 0.1287 | 0.0474 | 0.0251 | 0.0120 | 0.0085 | 0.0296 |
| 2 | Mean | 0.6675 | 0.4088 | 0.2933 | 0.2275 | 0.1865 | 0.1592 |
| | Std. | 0.1301 | 0.0640 | 0.0417 | 0.0280 | 0.0195 | 0.0724 |
| 3 | Mean | 0.6258 | 0.3554 | 0.2547 | 0.2006 | 0.1660 | 0.1435 |
| | Std. | 0.2755 | 0.1436 | 0.0904 | 0.0598 | 0.0447 | 0.2015 |
| 4 | Mean | 0.7250 | 0.4183 | 0.2958 | 0.2350 | 0.1935 | 0.1643 |
| | Std. | 0.1899 | 0.0737 | 0.0399 | 0.0222 | 0.0112 | 0.0243 |
| 5 | Mean | 0.4000 | 0.2400 | 0.1775 | 0.1469 | 0.1315 | 0.1187 |
| | Std. | 0.2395 | 0.1263 | 0.0857 | 0.0640 | 0.0537 | 0.2554 |
| 6 | Mean | 0.6206 | 0.3406 | 0.2379 | 0.1841 | 0.1530 | 0.1327 |
| | Std. | 0.2623 | 0.1368 | 0.0920 | 0.0689 | 0.0548 | 0.2657 |
| 7 | Mean | 0.6650 | 0.4213 | 0.2942 | 0.2294 | 0.1935 | 0.1638 |
| | Std. | 0.0844 | 0.0645 | 0.0406 | 0.0248 | 0.0123 | 0.0467 |
| 8 | Mean | 0.6217 | 0.4042 | 0.2947 | 0.2310 | 0.1922 | 0.1633 |
| | Std. | 0.1968 | 0.1122 | 0.0664 | 0.0354 | 0.0185 | 0.0534 |
| Overall | Mean | 0.6352 | 0.3805 | 0.2703 | 0.2120 | 0.1765 | 0.1512 |
| | Std. | 0.1071 | 0.0677 | 0.0450 | 0.0327 | 0.0238 | 0.0175 |

Table 4 compares the highest precisions DCBMIR and the other methods can achieve on each class in the dataset. It can be seen from the Table that DCBMIR outperforms all other methods on classes 1, 4, 7, and 8 and it achieves the highest overall precision. The overall precision of DCBMIR is higher than that of RFRM. The results in Table 6 also show that RFRM significantly outperforms DCBMIR in precision on class 6. This fact suggests that the feedback mechanism employed in RFRM can probably be

combined with the proposed dynamic programming approach to further improve the accuracy of DCBMIR.

**Figure 4** The relationships between precisions and recalls for DCBMIR (the proposed) and the other SOTA retrieval methods on the Kvasir dataset; the horizontal axis is the recall and the vertical axis is the precision (see online version for colours)

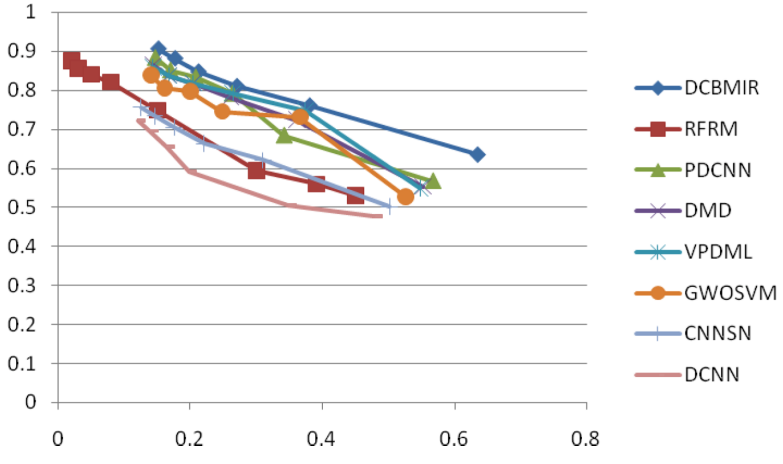



**Table 4** The highest precisions obtained by DCBMIR and RFRM on each class

| *Method* | *Class 1* | *Class 2* | *Class 3* | *Class 4* | *Class 5* | *Class 6* | *Class 7* | *Class 8* | *Overall* |
|---|---|---|---|---|---|---|---|---|---|
| DCBMIR | **0.9833** | 0.9550 | 0.8608 | **0.9858** | 0.7125 | 0.7963 | **0.9825** | **0.9800** | **0.9070** |
| RFRM | 0.9000 | 0.9000 | 0.8500 | 0.8500 | **0.8500** | **0.9000** | 0.8500 | 0.8000 | 0.8625 |
| PDCNN | 0.9527 | 0.9211 | 0.8753 | 0.9423 | 0.7343 | 0.7531 | 0.9621 | 0.9239 | 0.8831 |
| DMD | 0.9316 | **0.9643** | 0.8531 | 0.9015 | 0.7123 | 0.7434 | 0.9476 | 0.8686 | 0.8653 |
| VPDML | 0.9215 | 0.9372 | **0.8621** | 0.8954 | 0.7263 | 0.7154 | 0.9323 | 0.8834 | 0.8592 |
| GWOSVM | 0.9026 | 0.9137 | 0.8333 | 0.8642 | 0.8061 | 0.7043 | 0.9121 | 0.7717 | 0.8385 |
| CNNSN | 0.8465 | 0.7847 | 0.7462 | 0.7331 | 0.7011 | 0.6872 | 0.8215 | 0.7333 | 0.7567 |
| DCNN | 0.7823 | 0.7557 | 0.7754 | 0.6743 | 0.6529 | 0.6642 | 0.7539 | 0.7125 | 0.7214 |

### *3.2 Experiments on the hela dataset*

There are in total 862 nano-scale medical images in the hela dataset and the dataset contains 10 different classes. The number of images in a class ranges from 73 to 98. Table 5 shows the detailed information on the image classes in the dataset. An example image for each class in the dataset is shown in Figure 5. For each image in the dataset, a search is performed with DCBMIR in the dataset for images with $c$ highest similarities to the image. Figure 6 shows two examples of the query results generated by DCBMIR. Figure 6(a) shows the query result of an image from class 1, the first nine returned images are from class 1. Figure 6(b) shows the query result of an image from class 10 and the first eight returned images are from class 10.

**Figure 5** (a)–(j) are the examples of images from classes 1–10 in the hela dataset respectively

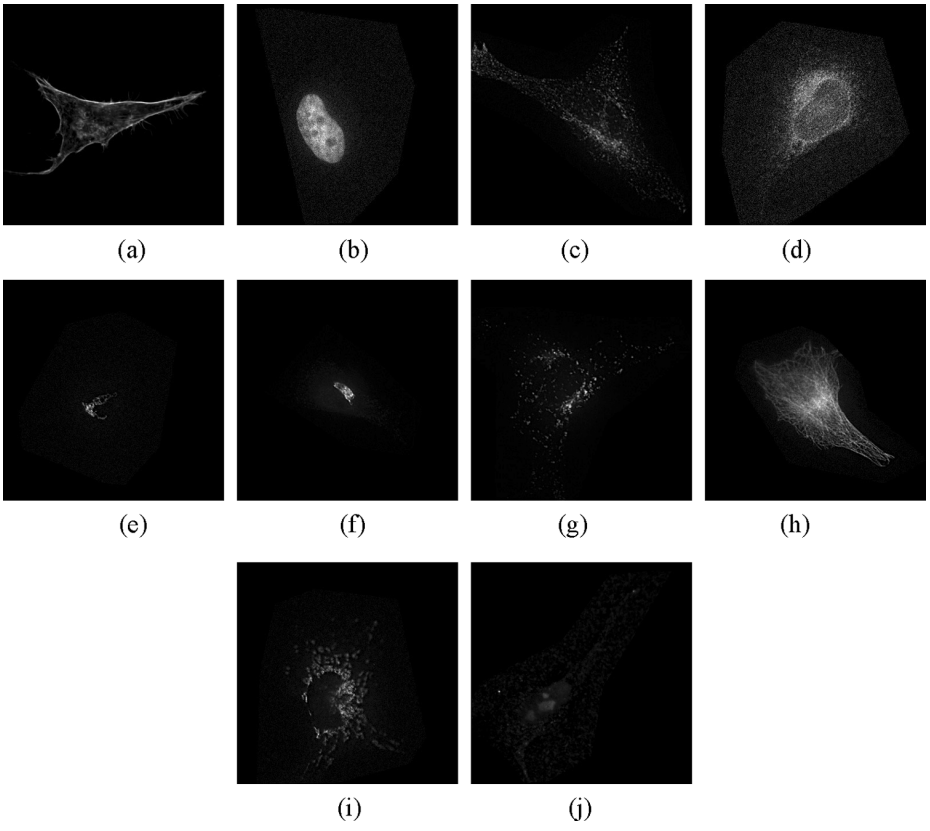
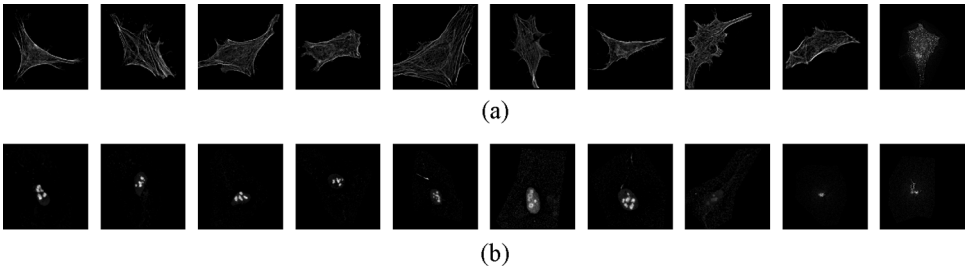



**Figure 6** Examples of the searched results generated by the proposed approach on the hela dataset: (a) shows the 10 images returned for the query of an image from class 1, where the first 9 images are from class 1 and the last image is from class 3 and (b) shows the 10 images returned for the query of an image from class 10, where the first 8 images are from class 10 and the last two images are from class 5




The values of parameter $c$ are then selected to be 100, 200, 300, 400, 500 and 600 respectively. Table 6 shows the mean and standard deviations of precisions obtained on each class in the dataset for different values of $c$. The mean and standard deviations of recalls on each class for different values of $c$ are shown in Table 7. The relationships between the precisions and recalls obtained with all methods on the hela dataset are shown in Figure 7. The figure clear shows that the overall accuracy of DCBMIR on the

hela dataset is higher than that of the other SOTA retrieval methods since DCBMIR can achieve a higher precision for a given value of recall.

**Table 5** Some information on the hela dataset used for the testing of the proposed approach

| *Class number* | *Class name* | *Number of images* | *Image size* |
|---|---|---|---|
| 1 | actin | 98 | 382 × 382 |
| 2 | dna | 87 | 382 × 382 |
| 3 | endosome | 91 | 382 × 382 |
| 4 | er | 86 | 382 × 382 |
| 5 | golgia | 87 | 382 × 382 |
| 6 | golgpp | 85 | 382 × 382 |
| 7 | lysosome | 84 | 382 × 382 |
| 8 | microtubules | 91 | 382 × 382 |
| 9 | mitochondria | 73 | 382 × 382 |
| 10 | nucleolus | 80 | 382 × 382 |

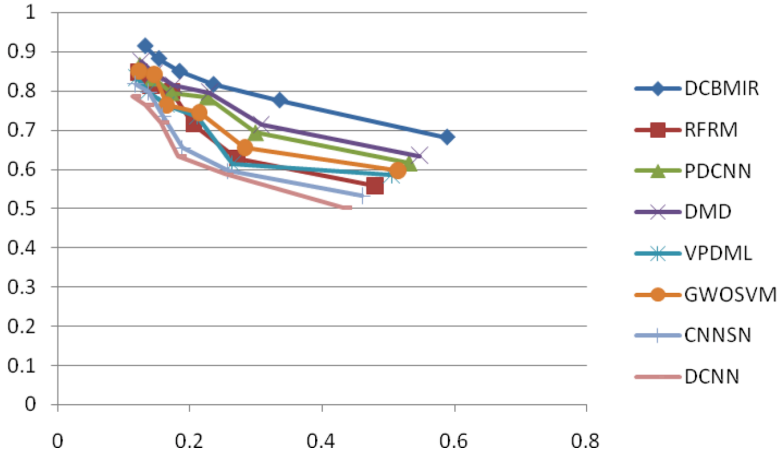
**Table 6** The mean and standard deviations (std.) of precision obtained by DCBMIR on each class in the hela dataset when 100, 200, 300, 400, 500, and 600 images are generated in a retrieval result

| *Class* | | $c = 100$ | $c = 200$ | $c = 300$ | $c = 400$ | $c = 500$ | $c = 600$ |
|---|---|---|---|---|---|---|---|
| 1 | Mean | 0.8300 | 0.8967 | 0.9222 | 0.9422 | 0.9589 | 0.9778 |
| | Std. | 0.1072 | 0.0748 | 0.0576 | 0.0418 | 0.0277 | 0.0181 |
| 2 | Mean | 0.7500 | 0.7901 | 0.8000 | 0.8300 | 0.8899 | 0.8901 |
| | Std. | 0.1581 | 0.1197 | 0.1247 | 0.1160 | 0.1370 | 0.1370 |
| 3 | Mean | 0.6300 | 0.7500 | 0.7800 | 0.8600 | 0.8701 | 0.9201 |
| | Std. | 0.2359 | 0.2273 | 0.2150 | 0.1647 | 0.1567 | 0.0919 |
| 4 | Mean | 0.6650 | 0.8050 | 0.8675 | 0.8975 | 0.9175 | 0.9525 |
| | Std. | 0.1688 | 0.1711 | 0.1236 | 0.0968 | 0.0755 | 0.0583 |
| 5 | Mean | 0.6100 | 0.7000 | 0.7233 | 0.7617 | 0.8017 | 0.8600 |
| | Std. | 0.1101 | 0.0882 | 0.1037 | 0.1042 | 0.0938 | 0.0763 |
| 6 | Mean | 0.6601 | 0.7451 | 0.7650 | 0.7801 | 0.8151 | 0.8350 |
| | Std. | 0.1150 | 0.1117 | 0.0944 | 0.1085 | 0.1081 | 0.0883 |
| 7 | Mean | 0.6475 | 0.7676 | 0.7951 | 0.8125 | 0.8450 | 0.8800 |
| | Std. | 0.1618 | 0.1882 | 0.1961 | 0.2015 | 0.1870 | 0.1517 |
| 8 | Mean | 0.6450 | 0.7700 | 0.8500 | 0.8600 | 0.8800 | 0.9050 |
| | Std. | 0.1165 | 0.1378 | 0.1434 | 0.1370 | 0.1418 | 0.1301 |
| 9 | Mean | 0.7300 | 0.8000 | 0.8600 | 0.9000 | 0.9550 | 0.9901 |
| | Std. | 0.1337 | 0.1106 | 0.0876 | 0.0667 | 0.0497 | 0.0211 |
| 10 | Mean | 0.6486 | 0.7330 | 0.7972 | 0.8500 | 0.8858 | 0.9314 |
| | Std. | 0.0886 | 0.1004 | 0.1148 | 0.1167 | 0.1028 | 0.0758 |
| Overall | Mean | 0.6816 | 0.7758 | 0.8160 | 0.8494 | 0.8819 | 0.9142 |
| | Std. | 0.0676 | 0.0533 | 0.0583 | 0.0557 | 0.0527 | 0.0502 |

**Table 7** The mean and standard deviations (std.) of precision obtained by DCBMIR on each class in the hela dataset when 100, 200, 300, 400, 500, and 600 images are generated in a retrieval result

| *Class* | | $c = 100$ | $c = 200$ | $c = 300$ | $c = 400$ | $c = 500$ | $c = 600$ |
|---|---|---|---|---|---|---|---|
| 1 | Mean | 0.8134 | 0.4393 | 0.3013 | 0.2309 | 0.1880 | 0.1597 |
| | Std. | 0.1051 | 0.0367 | 0.0188 | 0.0103 | 0.0054 | 0.0029 |
| 2 | Mean | 0.6525 | 0.3437 | 0.2320 | 0.1805 | 0.1549 | 0.1290 |
| | Std. | 0.1375 | 0.0521 | 0.0362 | 0.0252 | 0.0238 | 0.0198 |
| 3 | Mean | 0.5733 | 0.3413 | 0.2366 | 0.1957 | 0.1583 | 0.1395 |
| | Std. | 0.2147 | 0.1035 | 0.0653 | 0.0375 | 0.0285 | 0.0139 |
| 4 | Mean | 0.5719 | 0.3462 | 0.2487 | 0.1930 | 0.1578 | 0.1366 |
| | Std. | 0.1452 | 0.0736 | 0.0354 | 0.0208 | 0.0130 | 0.0083 |
| 5 | Mean | 0.5307 | 0.3045 | 0.2098 | 0.1656 | 0.1395 | 0.1247 |
| | Std. | 0.0958 | 0.0384 | 0.0301 | 0.0227 | 0.0164 | 0.0110 |
| 6 | Mean | 0.5610 | 0.3166 | 0.2167 | 0.1657 | 0.1385 | 0.1183 |
| | Std. | 0.0978 | 0.0475 | 0.0268 | 0.0230 | 0.0184 | 0.0125 |
| 7 | Mean | 0.5439 | 0.3223 | 0.2226 | 0.1706 | 0.1420 | 0.1232 |
| | Std. | 0.1359 | 0.0790 | 0.0549 | 0.0423 | 0.0314 | 0.0213 |
| 8 | Mean | 0.5870 | 0.3504 | 0.2578 | 0.1957 | 0.1602 | 0.1372 |
| | Std. | 0.1060 | 0.0627 | 0.0435 | 0.0312 | 0.0258 | 0.0197 |
| 9 | Mean | 0.5329 | 0.2920 | 0.2093 | 0.1642 | 0.1394 | 0.1205 |
| | Std. | 0.0976 | 0.0404 | 0.0213 | 0.0122 | 0.0072 | 0.0026 |
| 10 | Mean | 0.5189 | 0.2931 | 0.2126 | 0.1700 | 0.1417 | 0.1242 |
| | Std. | 0.0709 | 0.0402 | 0.0306 | 0.0234 | 0.0165 | 0.0100 |
| Overall | Mean | 0.5885 | 0.3349 | 0.2347 | 0.1832 | 0.1520 | 0.1313 |
| | Std. | 0.0877 | 0.0427 | 0.0287 | 0.0211 | 0.0155 | 0.0124 |

**Figure 7** The relationships between precisions and recalls for DCBMIR (the proposed) and the other SOTA retrieval methods on the hela dataset; the horizontal axis is the recall and the vertical axis is the precision (see online version for colours)



## 4 Conclusions

In this paper, a new approach is developed for accurate content-based retrieval of medical images. The proposed approach utilises three different histograms to represent the high level features and the low level textural features of a medical image together. Three pairs of histograms can thus be formed for two medical images. The structures of each pair of histograms are compared and the largest substructure shared by the two histograms is recognised. Such shared substructures may provide crucial information on the low level and high level features shared by both images. The similarity of two medical images is thus evaluated based on the identified shared substructures. The recognition of the largest substructure shared by two histograms is performed by optimally aligning the histograms. The optimal alignment can be efficiently computed with a dynamic programming approach. The optimal alignments of the corresponding histograms are used to obtain the largest shared substructure and a similarity value is computed for the two images based on the identified largest shared substructure. Experimental results obtained on two benchmark datasets show that the proposed approach is able to outperform SOTA approaches for CBMIR on both ordinary and nano-scale medical images.

It is clear that more experiments are needed in future work to provide a more thorough evaluation of the retrieval accuracy of the proposed approach. In addition, more image features that can represent the most important aspects of a medical image can probably be extracted if deep learning based technique can be employed. The statistical properties of such features can be described by histograms and the alignment between such histograms with the proposed approach can probably significantly improve the retrieval accuracy. However, the training of the parameters in such a model cannot be performed with a gradient based iterative method and is thus computationally intensive. New computational methods are probably needed to improve the computational efficiency of training for a retrieval approach that combines the proposed approach and deep learning based methods. In addition, retrieval methods that are more service-oriented may significantly extend its areas of applications in practice. The combination of the proposed method with deep learning based feature extraction techniques, fog computing [52] and service-oriented computing [53] thus might be an important direction for our future efforts to further improve the performance of the proposed approach.

## References

**1** Owais, M., Arsalan, M., Choi, J. and Park, K.R. (2019) 'Effective diagnosis and treatment through content-based medical image retrieval (CBMIR) by using artificial intelligence', *J. Clin. Med.*, Vol. 8, No. 4, p.462.

**2** Chakraborty, C. (2017) 'Chronic wound image analysis by particle swarm optimization technique for tele-wound network', *Wireless Pers. Commun.*, Vol. 96, pp.3655–3671.

**3** Sung, S.F., Lee, P.J., Hsieh, C.Y. and Zheng, W.L. (2020) 'Medication use and the risk of newly diagnosed diabetes in patients with epilepsy: a data mining application on a healthcare database', *J. Organ. End User Comput.*, Vol. 32, No. 2, pp.93–108.

**4** Sikha, O.K. and Soman, K.P. (2021) 'Dynamic mode decomposition based salient edge/region features for content based image retrieval', *Multimed. Tools Appl.*, Vol. 80, pp.15937–15958.

**5** Raghavan, R. and John, S.K. (2021) 'A hybrid approach towards content-based image retrieval for colored images using enhanced first type of pessimistic covering based lower approximation multi-granular rough sets', *Evol. Intell.*, Vol. 14, pp.1135–1143.

**6** Ghodratnama, S. and Abrishami, M.H. (2021) 'Content-based image retrieval using feature weighting and C-means clustering in a multi-label classification framework', *Pattern Anal. Appl.*, Vol. 24, pp.1–10.

**7** Al-Mohamade, A., Bchir, O. and Ben Ismail, M.M. (2020) 'Multiple query content-based image retrieval using relevance feature weight learning', *J. Imaging*, Vol. 6, No. 1, p.2.

**8** Nikolaos, P., Alexandros, I., Moncef, G. and Anastasios, T. (2020) 'Variance-preserving deep metric learning for content-based image retrieval', *Pattern Recognit. Lett.*, Vol. 131, pp.8–14.

**9** Buvana, M., Muthumayil, K. and Jayasankar, T. (2021) 'Content-based image retrieval based on hybrid feature extraction and feature selection technique pigeon inspired based optimization', *Ann. Romanian Soc. Cell Biol.*, Vol. 25, No. 1, pp.424–443.

**10** Ng, J.Y., Yang, F. and Davis, L.S. (2015) 'Exploiting local features from deep networks for image retrieval', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Boston, MA, USA, pp.53–61.

**11** Tzelepi, M. and Tefas, A. (2018) 'Deep convolutional learning for content based image retrieval', *Neurocomputing*, Vol. 275, pp.2467–2478.

**12** Rahman, M.M., Antani, S.K. and Thoma, G.R. (2011) 'A learning-based similarity fusion and filtering approach for biomedical image retrieval using SVM classification and relevance feedback', *IEEE Trans. Inf Technol Biomed.*, Vol. 15, No. 4, pp.640–646.

**13** Ponciano-Silva, M. *et al.* (2013) 'Does a CBIR system really impact decisions of physicians in a clinical environment?', *Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems*, pp.41–46.

**14** Quellec, G., Lamard, M., Cazuguel, G., Cochener, B. and Roux, C. (2012) 'Fast wavelet-based image characterization for highly adaptive image retrieval', *IEEE Trans. Image Process.*, Vol. 21, No. 4, pp.1613–1623.

**15** Jiji, G.W. and Raj, P.S.J.D. (2015) 'Content-based image retrieval in dermatology using intelligent technique', *IET Image Proc.*, Vol. 9, No. 4, pp.306–317.

**16** Zhang, F., Song, Y., Cai, W., Hauptmann, A.G., Liu, S., Pujol, S., Kikinis, R., Fulham, M.J., Feng, D.D. and Chen, M. (2016) 'Dictionary pruning with visual word significance for medical image retrieval', *Neurocomputing*, Vol. 177, pp.75–8.

**17** Roy, V., Shukla, P.K., Gupta, A.K., Goel, V., Shukla, P.K. and Shukla, S. (2021) 'Taxonomy on EEG artifacts removal methods, issues, and healthcare applications', *J. Organ. End User Comput.*, Vol. 33, No. 1, pp.19–46.

**18** Mizotin, M., Benois-Pineau, J., Allard, M. and Catheline, G. (2012) 'Feature-based brain MRI retrieval for Alzheimer disease diagnosis', *19th IEEE International Conference on Image Processing*, Lake Buena Vista, FL, USA., pp.1241–1244.

**19** Singh, R.K., Patidar, K., Kushwah, R. and Chouhan, S. (2017) 'Content based image retrieval system techniques: A review and analysis', *ACCENTS Transactions on Image Processing and Computer Vision*, Vol. 3, No. 7, pp.8–12.

**20** Srinivas, M., Naidu, R.R., Sastry, C.S. and Mohan, C.K. (2015) 'Content based medical image retrieval using dictionary learning', *Neurocomputing*, Vol. 168, pp.880–895.

**21** Tao, Y. (2010) *Multi-level Learning Approaches for Medical Image Understanding and Computer-Aided Detection and Diagnosis*, Virginia Polytech. Inst. State Univ., Blacksburg, VA, USA, Tech. Rep., April.

**22** Shamna, P., Govindan, V.K. and Abdul Nazeer, K.A. (2019) 'Content based medical image retrieval using topic and location model', *J. Biomed. Inform.*, Vol. 91, p.103112.

**23** Ahmed, A. (2020) 'Implementing relevance feedback for content-based medical image retrieval', *IEEE Access*, Vol. 8, pp.79969–79976.

**24** Qazanfari, H., Hassanpour, H. and Qazanfari, K. (2017) 'A short-term learning framework based on relevance feedback for content-based image retrieval', *Proc. 3rd Iranian Conf. Intell. Syst. Signal Process. (ICSPIS)*, December, Shahrood, Iran, pp.136–140.

**25** Kashif, M., Raja, G. and Shaukat, F. (2020) 'An efficient content-based image retrieval system for the diagnosis of lung diseases', *J. Digit. Imaging*, Vol. 33, pp.971–987.

**26** Renita, D.B. and Christopher, C.S. (2020) 'Novel real time content based medical image retrieval scheme with GWO-SVM', *Multimed. Tools Appl.*, Vol. 79, pp.17227–17243.

**27** Shinde, A., Rahulkar, A. and Patil, C. (2020) 'New flexible directional filter bank by tuning Hermite transform parameters for content based medical image retrieval', *IET Image Proc.*, Vol. 14, No. 11, pp.2403–2416.

**28** Zhang, F., Song, Y., Cai, W., Hauptmann, A.G., Liu, S., Pujol, S., Kikinis, R., Fulham, M.J., Feng, D.D. and Chen, M. (2016) 'Dictionary pruning with visual word significance for medical image retrieval', *Neurocomputing*, Vol. 177, pp.75–88.

**29** Shamna, P., Govindan, K.V.K. and Nazeer, A.A. (2022) 'Content-based medical image retrieval by spatial matching of visual words', *J. King Saud Univ.-Comput. Inf. Sci.*, Vol. 34, No. 2, pp.58–71.

**30** Banerjee, I., Kurtz, C., Devorah, A.E., Do, B., Rubin, D.L. and Beaulieu, C.F. (2018) 'Relevance feedback for enhancing content based image retrieval and automatic prediction of semantic image features: application to bone tumor radiographs', *J. Biomed. Inform.*, Vol. 84, pp.123–135.

**31** Karamti, H., Tmar, M., Visani, M., Urruty, T. and Gargouri, F. (2018) 'Vector space model adaptation and pseudo relevance feedback for content-based image retrieval', *Multimedia Tools Appl.*, Vol. 77, No. 5, pp.5475–5501.

**32** Torjmen-Khemakhem, M. and Gasmi, K. (2019) 'Document/query expansion based on selecting significant concepts for context based retrieval of medical images', *J. Biomed. Inform.*, Vol. 95, p.103210.

**33** Nair, L.R., Subramaniam, K. and Venkatesan, G.K.D.P. (2020) 'An effective image retrieval system using machine learning and fuzzy c-means clustering approach', *Multimed. Tools Appl.*, Vol. 79, pp.10123–10140.

**34** Bengio, Y., Courville, A.C. and Vincent, P. (2013) 'Representation learning: a review and new perspectives', *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 35, No. 8, pp.1798–1828.

**35** Sun, Q., Yang, Y., Sun, J., Yang, Z. and Zhang, J. (2017) 'Using deep learning for content-based medical image retrieval', *Proc. SPIE Med. Imag. Imag. Informat. Healthcare, Res., Appl.*, Vol. 10138, p.1013812.

**36** Cai, Y., Li, Y., Qiu, C., Ma, J. and Gao, X. (2019) 'Medical image retrieval based on convolutional neural network and supervised hashing', *IEEE Access*, Vol. 7, pp.51877–51885.

**37** Qayyum, A., Anwar, S.M., Awais, M. and Majid, M. (2017) 'Medical image retrieval using deep convolutional neural network', *Neurocomputing*, Vol. 266, pp.8–20.

**38** Yang, F., Yubao, L. and Jiebo, L. (2020) *Universal Model for Multi-Domain Medical Image Retrieval*, arXiv:2007.08628.

**39** Haripriya, P. and Porkodi, R. (2021) 'Parallel deep convolutional neural network for content based medical image retrieval', *J. Ambient Intell. Hum. Comput.*, Vol. 12, pp.781–795.

**40** Qi, C., Shi, C., Xu, J., Wang, C. and Xiao, B. (2017) 'Spatial weighted fisher vector for image retrieval', *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, Hong Kong, China, pp.463–468.

**41** Zheng, L., Yang, Y. and Tian, Q. (2018) 'SIFT meets CNN: a decade survey of instance retrieval', *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 40, No. 5, pp.1224–1244.

**42** Cao, Y., Long, M., Wang, J., Zhu, H. and Wen, Q. (2016) 'Deep quantization network for efficient image retrieval', *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI)*, Phoenix, AZ, USA, pp.3457–3463.

**43** Li, Y. and Wang, X. (2020) 'Seeking health information on social media: a perspective of trust, self-determination, and social support', *J. Organ. End User Comput.*, Vol. 30, No. 1, pp.1–22.

**44** Kishor, A., Chakraborty, C. and Jeberson, W. (2021) 'A novel fog computing approach for minimization of latency in healthcare using machine learning', *Int. J. Interact. Multimed. Artif. Intell.*, Vol. 6, No. 7, pp.7–17.

**45** Pogorelov, K., Randel, K.R., Griwodz, C., Eskeland, S.L., Lange, T.D., Johansen, D., Spampinato, C., Dang-Nguyen, D., Lux, M., Schmidt, P.T., Riegler, M. and Halvorsen, P. (2017) 'KVASIR: a multi-class image dataset for computer aided gastrointestinal disease detection', *Proc. 8th ACM Multimedia Syst. Conf.*, June, Taipei, Taiwan, pp.164–169.

**46** Boland, M.V. and Murphy, R.F. (2001) 'A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of hela cells', *Bioinformatics*, Vol. 17, pp.1213–1223.

**47** Needleman, S.B. and Wunsch, C.D. (1970) 'A general method applicable to the search for similarities in the amino acid sequence of two proteins', *J. Mol. Biol.*, Vol. 48, No. 3, pp.443–453.

**48** Malik, M., Malik, M.K., Mehmood, K. and Makhdoom, I. (2021) 'Automatic speech recognition: a survey', *Multimed. Tools Appl.*, Vol. 80, pp.9411–9457.

**49** Li, G., Rana, M.N.A., Sun, J., Song, Y. and Qu, J. (2020) 'Real-time image enhancement with efficient dynamic programming', *Multimed. Tools Appl.*, Vol. 79, Nos. 41–42, pp.30883–30903.

**50** Song, Y., Liu, C., Huang, X., Malmberg, R.L., Xu, Y. and Cai, L. (2006) 'Efficient parameterized algorithms for biopolymer structure-sequence alignment', *IEEE ACM Trans. Comput. Biol. Bioinform.*, Vol. 3, No. 4, pp.423–432.

**51** Li, G., Sun, J., Song, Y., Qu, J., Zhu, Z. and Khosravi, M.R. (2021) 'Real-time classification of brain tumors in MRI images with a convolutional operator-based hidden Markov model', *J. Real-Time Image Process.*, Vol. 18, pp.1207–1219.

**52** Amit, K., Chinmay, C. and Wilson, J. (2021) 'Reinforcement learning for medical information processing over heterogeneous networks', *Multimed. Tools Appl.*, Vol. 80, pp.23983–24004.

**53** Bhadoria, R.S. and Chaudhari, N.S. (2020) 'Pragmatic sensory data semantics with service-oriented computing', *J. Organ. End User Comput.*, Vol. 31, No. 2, pp.22–36.