Learning Neural Networks for Multi-label Medical Image Retrieval Using Hamming Distance Fabricated with Jaccard Similarity Coefficient

Asim Manna[0000−0001−7617−9762] and Debdoot Sheet[0000−0001−9046−149X]

Indian Institute of Technology Kharagpur, 721302, West Bengal, India
{asimmanna17@kgpian,debdoot@ee}.iitkgp.ac.in

Abstract. Deep neural hashing (DNH) has demonstrated its effectiveness in content-based medical image retrieval (CBMIR) for efficient nearest-neighbor search in large image datasets. It learns a hash function to generate hash codes from the images. Conventional pairwise DNH methods are inadequate for multi-label CBMIR as they do not incorporate between the Hamming distance (HD) of hash codes and the Jaccard similarity coefficient (JSC) of label sets for an image pair. This work introduces a JSC-based loss function called adaptive HD loss (AHDL) for learning HD between hash pairs using a deep neural network to retrieve multi-label medical images. AHDL helps the model assign an appropriate HD between a pair of hash codes based on their image similarity level. We also adopt pairwise multi-label classification loss to generate unique features for each class combination. Experiments are demonstrated on the publicly available NIH chest X-ray dataset. Our method achieves 3.98% higher normalized discounted cumulative gain compared to the state-of-the-art method for a top-100 image retrieval task.

Keywords: Content-based medical image retrieval · Deep neural hashing network · Jaccard coefficient · Hamming distance · Pairwise similarity

1 Introduction

Content-based medical image retrieval (CBMIR) offers clinicians valuable evidence for assessing cases with similar symptoms or pathological representations [3,22]. Feature extraction and ranking of images based on the similarity of a query image are two critical steps in CBMIR. With the rapid growth in medical imaging, efficient and accurate retrieval of relevant information from large databases remains a challenge in routine radiology workflows [18,8]. The interpretation of medical images often relies on the expertise of professionals. The utilization of case reports limited to medical images as diagnostic benchmarks has been shown to affect expert interpretation significantly. CBMIR is capable of retrieving similar cases of medical images for supplementary analysis to bridge variations across diagnoses by experts [8,27].

The task of multi-label medical image retrieval is challenging because it entails assigning multiple labels or categories to a given medical image [11,25]. Consequently, there is a growing focus on multi-label CBMIR tasks due to their practicality in real-life scenarios. As an instance, an organ in the human body is affected by multiple pathologies simultaneously. The similarity associated with various pathology in medical images could have been very subtle, so it is necessary to use advanced feature extraction approaches for more comprehensive analysis [7]. Challenges in multi-label medical image retrieval include the ability to accurately capture complex relationships represented by different labels, address class imbalance issues, and developing efficient retrieval algorithms capable of handling large-scale datasets [11,13]. These challenges create difficulties for traditional retrieval methods, thereby necessitating the development of advanced algorithms to enhance the retrieval performance.

Deep neural hashing (DNH) methods [21,23] have emerged as promising solutions for medical image retrieval. These methods involve encoding medical images into compact binary codes while preserving similarity relationships in the Hamming space [4]. It learns advanced hash functions that can generate hash codes from images. Hashing is used for feature extractors by transforming the images into binary representations. These hash codes facilitate efficient nearest-neighbor searches within extensive image datasets. After generating hash codes from images, the Hamming distance (HD) is often used to measure the semantic similarity between images. HD between a pair of hash codes is finite and inversely proportional to the similarity of the images they represent. Therefore, incorporating HD effectively into the ranking algorithm ensures that similar images are prioritized higher in the retrieval list, enhancing the overall performance of the retrieval system.

In the context of multi-label medical images, a pair of images may exhibit comorbid pathologies as well as distinct pathologies. So, HD between generated hash codes of a multi-label image pair should depend on the proportion of shared labels out of the total possible labels for the pair. An example regarding this is illustrated in Figure 1. There are three possible labels: Atelectasis, Effusion, and Infiltration. Consider three images $\mathbf{x}_i, \mathbf{x}_j,$ and $\mathbf{x}_k$ with their label sets denoted as $\mathbf{y}_i = \{1, 0, 0\}, \mathbf{y}_j = \{1, 1, 0\},$ and $\mathbf{y}_k = \{0, 1, 1\}$ respectively. The value 1 in a label set indicates that the pathology is associated with this image, while 0 indicates otherwise. The number of all possible labels, shared labels, and Jaccard similarity coefficient (JSC) [2] between an image pair $\mathbf{x}_i, \mathbf{x}_j$ are denoted as $n_{ij}^{(1)}(= |\mathbf{y}_i \cup \mathbf{y}_j|), n_{ij}^{(2)}(= |\mathbf{y}_i \cap \mathbf{y}_j|),$ and $\frac{n_{ij}^{(2)}}{n_{ij}^{(1)}}$ respectively. In this example, $n_{ij}^{(1)} = 2, n_{ij}^{(2)} = 1, n_{jk}^{(1)} = 3, n_{jk}^{(2)} = 1$. Since, $\frac{n_{ij}^{(2)}}{n_{ij}^{(1)}} > \frac{n_{jk}^{(2)}}{n_{jk}^{(1)}}$, the preferable scenario is that the HD between hash codes for $\mathbf{x}_j, \mathbf{x}_k$ is higher than the HD between hash codes for $\mathbf{x}_i, \mathbf{x}_j$. In the other words, for each unique combination of $n_{ij}^{(1)}$ and $n_{ij}^{(2)}$, the HD should vary. Indeed, as the JSC between the label sets of an image pair increases, indicating a higher degree of similarity in their pathology manifestations, the HD between the corresponding hash pair should decrease. This implies that the JSC-based similarity measurement has the capability to delineate nuanced multi-level semantic similarities. So, there is a need for an advanced learning hash function capable of establishing a one-to-one relationship between the HD of hash pairs and the JSC. Such a function would effectively capture the intricate relationships between label similarities and HD, facilitating more accurate representation and retrieval of multi-label medical images.

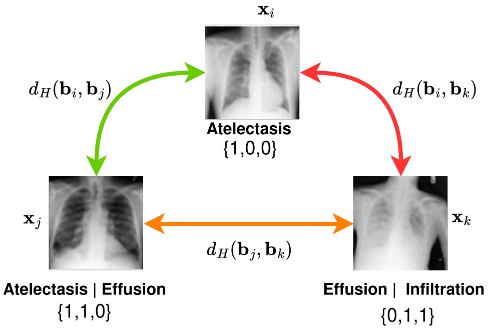


Fig. 1: An example overview of our objective using three images $\mathbf{x}_i$ (Atelectasis), $\mathbf{x}_j$ (Atelectasis, Effusion), $\mathbf{x}_k$ (Effusion, Infiltration). Here $\frac{n_{ij}^{(2)}}{n_{ij}^{(1)}} > \frac{n_{jk}^{(2)}}{n_{jk}^{(1)}} > \frac{n_{ik}^{(2)}}{n_{ik}^{(1)}} = 0$, where each ratio is calculated from corresponding image label sets. In this scenario, $d_H(\mathbf{b}_i, \mathbf{b}_j) < d_H(\mathbf{b}_j, \mathbf{b}_k) < d_H(\mathbf{b}_i, \mathbf{b}_k)$ will be followed, where $\mathbf{b}_i, \mathbf{b}_j, \mathbf{b}_k$ are the hash codes corresponding to each of the images.

In this work, we propose a loss function to foster similarity learning between a pair of images, utilizing the JSC as a metric. Adaptive HD loss (AHDL) is employed to assign suitable Hamming distance (HD) between a pair of hash codes according to their image similarity level based on the value of the JSC. Besides HD learning, semantic classification is another significant learning objective to learn hash representation from images. We adopt pairwise multi-label classification loss to generate unique features for each different label combination. The main contributions of this work are summarized as follows:

- Method of learning hash codes using a neural network in order to retrieve images contextually sensitive to their semantic similarity of multiple pathologies imaged in an organ.
- To the best of our knowledge, no existing work learns hash codes that simultaneously consider both the HD and the JSC for multi-label CBMIR. In this work, we develop an advanced learning hash function that establishes a one-to-one relationship between the HD of hash pairs and the JSC of the label set of the corresponding image pairs.

- A loss function designed to generate appropriate hash codes so that accurate HD is based on the similarity levels between a pair of images.
- The commonly used metrics, normalized discounted cumulative gain (nDCG), average cumulative gains (ACG), and wighted mean average precision (wMAP) are utilized in order to measure the retrieval performance of the proposed method.

The paper is organized as follows. The prior art of DNH for image retrieval is presented in Section 2. The proposed method is introduced in Section 3. Experimental details are discussed in Section 4. Results and discussions are presented in Section 5. This work is concluded in Section 6.

2 Prior Art

In this section, we will primarily discuss some related works in this domain, including methods related to medical image retrieval and DNH for multi-label image retrieval. The methods for image retrieval utilizing hashing can be broadly categorized into two categories: data-independent hashing (DIH) and data-dependent hashing (DDH).

DIH refers to a hashing technique where the hash functions are generated without relying on the specific characteristics or content of the data being hashed. This technique does not utilize any labeled data or information for the hashing process [24]. Local sensitive hashing (LSH) is a widely used DIH technique that employs randomized projections or permutations to design different hash functions, aiming to return the identical codes for similar data items with high probability [24,14].

The rapid expansion of data coupled with the advancement of deep neural networks (DNN) [20] is reshaping the landscape of various fields, including image retrieval. DNH has gained significant attention in recent times due to the advancements in deep neural networks (DNN) [20] for image representation. These methods incorporate DNN into the process of constructing binary codes for images. By combining the strengths of deep learning with the computational efficiency and storage benefits of hashing techniques, DNH-based methods effectively map the image representation space learned by deep models into a binary space. DNH techniques are suitable for large datasets, which is a common requirement in the medical field where medical images are continually being generated and stored. The concept of DNH [24,19,26] is introduced with the help of DNN to build a hash function that effectively leverages data distributions and incorporates information regarding class labels present in a dataset. Its objective is to ensure that the nearest neighbor of any pattern in the space of hash codes closely resembles the neighboring patterns in the original space [24]. Preserving data similarity in the Hamming space is the primary objective of majority of learning-based data-dependent hashing techniques [9,1,28]. Several studies have previously concentrated on supervised hashing-based image retrieval using the similarity matrix and quantization loss [29]. HashNet proposes a method to learn non-smooth binary activation in order to generate binary hash codes from

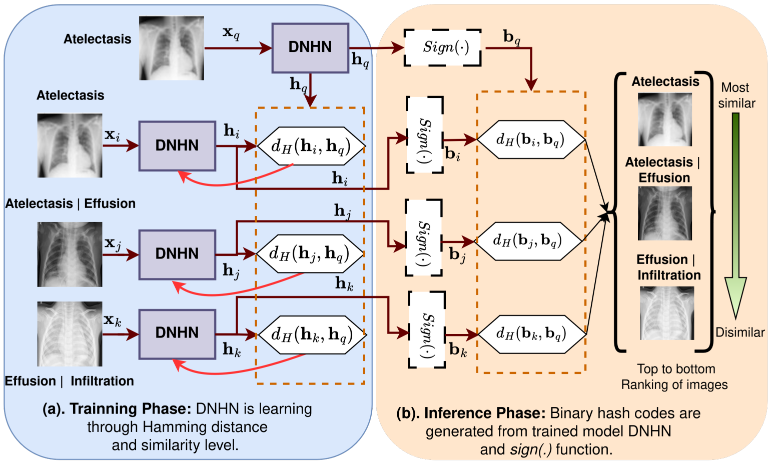


Fig. 2: The figure on the left illustrates the training process of a deep neural hashing network (DNHN), where the network learns by the HD between real-valued hash codes $(\mathbf{h}_q, \mathbf{h}_i, \mathbf{h}_j, \mathbf{h}_k)$. We aim to learn correct order of $d_H(\mathbf{h}_i, \mathbf{h}_q), d_H(\mathbf{h}_j, \mathbf{h}_q), d_H(\mathbf{h}_k, \mathbf{h}_q)$ through the Jaccard coefficient, where $d_H(\cdot)$ represents HD between two hash codes. Conversely, the figure on the right demonstrates the generation of binary hash codes $(\mathbf{b}_q, \mathbf{b}_i, \mathbf{b}_j, \mathbf{b}_k)$ achieved by applying the $sign(\cdot)$ function. The order of the HD is $d_H(\mathbf{b}_i, \mathbf{b}_q) < d_H(\mathbf{b}_j, \mathbf{b}_q) < d_H(\mathbf{b}_k, \mathbf{b}_q)$ and ideal image ranking with respect to $\mathbf{x}_q$.

imbalanced similarity data [5]. Deep Cauchy hashing (DCH) model utilizes a pairwise cross-entropy loss based on the Cauchy distribution to generate binary hash codes [4]. OrthoHash is based on one loss, eliminating the need for balancing coefficient tuning in various losses [12]. OrthHash generates center hash codes [15] using Bernoulli distributions. Attention-based triplet hashing (ATH) network [10] is an end-to-end system designed to learn low-dimensional hash codes that preserve the categorization, region of interest, and small-sample information. Multi-scale triplet hashing [6] and deep semantic ranking hashing based on self-attention (DSHA) [28] offer an effective and scalable solution by leveraging multi-scale information and triplet loss to achieve accurate and efficient retrieval of medical images. The previous learning-based hashing methods [31] are able to only take care of $n_{ij}^{(1)}$ but do not properly incorporate the information between $n_{ij}^{(1)}, n_{ij}^{(2)}$, and the HD between hash codes. This affects the ranking in the image retrieval list and, therefore, retrieval performance. Images with the same similarity level can be more finely differentiated using the JSC. However, a learning method that generates hash codes considering both the HD and the JSC has not been properly developed yet. In this work, we introduce a pairwse learning approach to generate hash codes from image pairs while accounting for the JSC between the label sets of these multi-label medical image pairs. Our method fulfills the following requirements. (i) generate unique features for different combinations of pathology present in images. (ii) HD between a pair of hash codes of multi-label images depends on the number of match pathologies. The overview of our approach and CBMIR using DNH is illustrated in Figure 2.

3 Proposed Methodology

3.1 Problem statement

Consider a training set of images represented as $\mathbf{X}_{\mathfrak{T}} = \{\mathbf{x}_1^{\mathfrak{T}}, \mathbf{x}_2^{\mathfrak{T}}, \dots, \mathbf{x}_i^{\mathfrak{T}}, \dots, \mathbf{x}_{U_1}^{\mathfrak{T}}\}$. $\mathbf{y}_i^{\mathfrak{T}} \in \{0,1\}^L$ represents the label set of image $\mathbf{x}_i^{\mathfrak{T}} \in \mathbb{R}^{M \times N}$, where L denotes the number of possible labels in the dataset. Consider a non-linear hash function $F : \mathbb{R}^{M \times N} \mapsto \{-1,1\}^K$ such that each $\mathbf{x}_i^{\mathfrak{T}} \in \mathbb{R}^{M \times N}$ is an image to be hashed into a K-length binary hash code $\mathbf{b}_i^{\mathfrak{T}} \in \{-1,1\}^K$.

Let, the number of all possible labels and shared labels between an image pair $\mathbf{x}_i^{\mathfrak{T}}, \mathbf{x}_j^{\mathfrak{T}}$ are denoted as $n_{ij}^{(1)}(= |\mathbf{y}_i^{\mathfrak{T}} \cup \mathbf{y}_j^{\mathfrak{T}}| \neq 0), n_{ij}^{(2)}(= |\mathbf{y}_i^{\mathfrak{T}} \cap \mathbf{y}_j^{\mathfrak{T}}|)$ respectively. Our aims to learn $F(\cdot)$ following a supervised learning approach such that $d_H(\mathbf{b}_i^{\mathfrak{T}}, \mathbf{b}_j^{\mathfrak{T}}) \leq d_H(\mathbf{b}_i^{\mathfrak{T}}, \mathbf{b}_k^{\mathfrak{T}})$ if and only if $\frac{n_{ij}^{(2)}}{n_{ij}^{(1)}} \geq \frac{n_{ik}^{(2)}}{n_{ik}^{(1)}}$, where $\mathbf{b}_i^{\mathfrak{T}} = F(\mathbf{x}_i^{\mathfrak{T}})$ and $d_H(\cdot)$ represents the HD between two hash codes of length K. The idea is that if the number of common labels between a pair of images is more, then the HD between a pair of generated hash codes should be less.

3.2 Hash code generation

The hash code generation process from images can be expressed through $F(\cdot)$ using the following equations,

$$\mathbf{b}_i^{\mathfrak{T}} = sign(\mathbf{h}_i) = F(\mathbf{x}_i^{\mathfrak{T}}) \tag{1}$$

$$\mathbf{h}_i = Tanh(\mathtt{fc_h}(\mathbf{z}_i)) \tag{2}$$

$$\mathbf{z}_i = \mathtt{net_e}(\mathbf{x}_i^{\mathfrak{T}}) \tag{3}$$

where, $\mathbf{h}_i \in [-1,1]^K$ and $\mathbf{b}_i^{\mathfrak{T}} \in \{-1,1\}^K$ represent real valued and binary hash codes respectively. The $sign(\cdot)$ function [4] is employed to convert the real valued hash to a binary hash representation. $\mathtt{net_e}(\cdot)$ represents the CNN-based feature encoding function is given by,

$$\begin{aligned} \mathtt{net_e}(\cdot) \mapsto\ & \mathtt{Conv2D:64c11w4s2p} \rightarrow \mathtt{ReLU} \rightarrow \mathtt{MaxPool2D:3w2s} \\ & \rightarrow \mathtt{Conv2D:192c5w1s2p} \rightarrow \mathtt{ReLU} \rightarrow \mathtt{MaxPool2D:3w2s} \\ & \rightarrow \mathtt{Conv2D:384c3w1s1p} \rightarrow \mathtt{ReLU} \rightarrow \mathtt{Conv2D:256c3w1s1p} \rightarrow \mathtt{ReLU} \\ & \rightarrow \mathtt{Conv2D:256c3w1s1p} \rightarrow \mathtt{MaxPool2D:3w2s} \rightarrow \mathtt{Flatten} \end{aligned} \tag{4}$$

$\mathtt{fc_h}(\cdot)$ is the real valued hash generating fully connected layers.

$$\mathtt{fc_h}(\cdot) \mapsto \mathtt{Linear:4096} \rightarrow \mathtt{ReLU} \rightarrow \mathtt{Linear:K} \tag{5}$$

(2) and (3) are used during training to avoid the vanishing gradient [4] challenge faced in (1) on account of the $sign(\cdot)$ function. Thus, we utilize $\mathbf{h}_i$ and $\mathbf{b}_i^{\mathfrak{T}}$ during training and inference respectively.

During the training process, we utilize two distinct loss functions: adaptive HD loss (AHDL) and pairwise multi-label classification loss (PMCL). The purpose of the PMCL is to create distinctive features for various combinations of pathologies present in the images. Meanwhile, AHDL is applied to generate hash codes, ensuring that the Hamming distance between these codes appropriately reflects the similarity levels between images. Our method is trained in an end-to-end manner, in which image feature learning and HD learning via hash codes from image pairs are performed simultaneously.

3.3 Adaptive Hamming distance loss (AHDL)

The idea of designing adaptive HD loss for multi-label image retrieval is that HD between hash codes of a image pair should be depended on the number of total possible pathology ($n_{ij}^{(1)} = |\mathbf{y}_i^{\mathfrak{T}} \cup \mathbf{y}_j^{\mathfrak{T}}|$) and the number of common pathology ($n_{ij}^{(2)}$) $= |\mathbf{y}_i^{\mathfrak{T}} \cap \mathbf{y}_j^{\mathfrak{T}}|$ between $\mathbf{x}_i^{\mathfrak{T}}$ and $\mathbf{x}_j^{\mathfrak{T}}$. When $\frac{n_{ij}^{(2)}}{n_{ij}^{(1)}}$ is increased HD should be less and vice-versa. Let, $\mathbf{h}_i, \mathbf{h}_j$ be the real valued hash codes of this image pair $\mathbf{x}_i^{\mathfrak{T}}$ and $\mathbf{x}_j^{\mathfrak{T}}$ respectively. Here are some constraints specified for this image pair,

1. $0 \leq n_{ij}^{(2)} \leq n_{ij}^{(1)} \leq L$ and $n_{ij}^{(1)} \neq 0 \ \forall i, j$.
2. $n_{ij}^{(1)} = 1$ implies $\mathbf{y}_i^{\mathfrak{T}} = \mathbf{y}_j^{\mathfrak{T}}$. In this scenario, $n_{ij}^{(2)} = n_{ij}^{(1)} = 1 \ \forall i, j$.
3. $0 \leq d_H(\mathbf{h}_i, \mathbf{h}_j) \leq K, \ \forall i, j$.

The adaptive HD between $\mathbf{h}_i, \mathbf{h}_j$ is based on the value of $n_{ij}^{(1)}, n_{ij}^{(2)}$ and defined by,

$$D_H^{(n_{ij}^{(1)}, n_{ij}^{(2)})}(\mathbf{h}_i, \mathbf{h}_j) = L_{HD}^{(n_{ij}^{(1)})}(\mathbf{h}_i, \mathbf{h}_j)[n_{ij}^{(2)}] \tag{6}$$

$$L_{HD}^{(n_{ij}^{(1)})}(\mathbf{h}_i, \mathbf{h}_j) = \left[K, \left\lfloor \frac{(n_{ij}^{(1)} - 1)K}{n_{ij}^{(1)}} \right\rfloor, \left\lfloor \frac{(n_{ij}^{(1)} - 2)K}{n_{ij}^{(1)}} \right\rfloor, \ldots, 0\right] \tag{7}$$

The HD between any hash pair ranges from 0 to K. Given that $0 \leq n_{ij}^{(2)} \leq n_{ij}^{(1)}$, the number of possible values for $n_{ij}^{(2)}$ is $(n_{ij}^{(1)} + 1)$. $(n_{ij}^{(1)} + 1)$ approximately equidistant points are selected from the interval $[0, K]$ using the floor function. We then store these HD in a descending order list, denoted as $L_{HD}^{(n_{ij}^{(1)})}(\mathbf{h}_i, \mathbf{h}_j)$ in (7). $L_{HD}^{(n_{ij}^{(1)})}(\mathbf{h}_i, \mathbf{h}_j)$ is a descending order list of HD between $\mathbf{h}_i, \mathbf{h}_j$ based on the value of $n_{ij}^{(1)}$. From (6), we can observe that as the number of shared levels increases, the value of $D_H^{(n_{ij}^{(1)}, n_{ij}^{(2)}))}(\mathbf{h}_i, \mathbf{h}_j)$ decreases. When $n_{ij}^{(2)} = n_{ij}^{(1)} = 1$, $L_{HD}^{(1)}(\mathbf{h}_i, \mathbf{h}_j) = [K, 0]$ implies $D_H^{(1,1)}(\mathbf{h}_i, \mathbf{h}_j) = 0$. An illustrative example is depicted in Table 1. When $n_{ij}^{(1)} = 3$, the possible values of $n_{ij}^{(2)}$ are $0, 1, 2, 3$.

Table 1: An example of computing HD based on similarity level of an image pair. Here $L = 3$ and $K = 16$. $L_{HD}^{(n_{ij}^{(1)})}(\mathbf{h}_i, \mathbf{h}_j)$ is the list of HD for given value of $n_{ij}^{(1)}$. $D_H^{(n_{ij}^{(1)},n_{ij}^{(2)})}(\mathbf{h}_i, \mathbf{h}_j)$ is the HD, collected from the list $L_{HD}^{(n_{ij}^{(1)})}(\mathbf{h}_i, \mathbf{h}_j)$ for given specific value of $n_{ij}^{(2)}$.

$\mathbf{y}_i^{\mathfrak{T}}$	$\mathbf{y}_j^{\mathfrak{T}}$	$n_{ij}^{(1)}$	$n_{ij}^{(2)}$	$L_{HD}^{(n_{ij}^{(1)})}(\mathbf{h}_i, \mathbf{h}_j)$	$D_H^{(n_{ij}^{(1)},n_{ij}^{(2)})}(\mathbf{h}_i, \mathbf{h}_j)$
$\{1, 0, 1\}$	$\{0, 1, 0\}$	3	0	$[16, 10, 5, 0]$	16
$\{1, 1, 1\}$	$\{1, 0, 0\}$		1		10
$\{1, 1, 1\}$	$\{1, 1, 0\}$		2		5
$\{1, 1, 1\}$	$\{1, 1, 1\}$		3		0
$\{1, 0, 0\}$	$\{0, 1, 0\}$	2	0	$[16, 8, 0]$	16
$\{1, 1, 0\}$	$\{1, 0, 0\}$		1		8
$\{1, 1, 0\}$	$\{1, 1, 0\}$		2		0
$\{1, 0, 0\}$	$\{1, 0, 0\}$	1	1	$[16, 0]$	0

Then we get, $L_{HD}^{(3)}(\mathbf{h}_i, \mathbf{h}_j) = [16, 10, 5, 0]$ from (7). If there is no shared label i.e, $n_{ij}^{(2)} = 0$ then $D_H^{(3,0)}(\mathbf{h}_i, \mathbf{h}_j) = 16$. Similarly, $D_H^{(3,1)}(\mathbf{h}_i, \mathbf{h}_j) = 10$, $D_H^{(3,2)}(\mathbf{h}_i, \mathbf{h}_j) = 5$, $D_H^{(3,3)}(\mathbf{h}_i, \mathbf{h}_j) = 0$. The value of $D_H^{(n_{ij}^{(1)},n_{ij}^{(2)}))}(\mathbf{h}_i, \mathbf{h}_j)$ is distinct for each unique combination of $n_{ij}^{(1)}$ and $n_{ij}^{(2)}$. The AHDL for image pair on $\mathbf{X}_{\mathfrak{T}}$ is computed as,

$$J_1 = \sum_{\mathbf{x}_i^{\mathfrak{T}}, \mathbf{x}_j^{\mathfrak{T}} \in \mathbf{X}_{\mathfrak{T}}} \log\left(\cosh\left(\frac{D_H^{(n_{ij}^{(1)},n_{ij}^{(2)})}(\mathbf{h}_i, \mathbf{h}_j) - d_H(\mathbf{h}_i, \mathbf{h}_j)}{K}\right)\right) \tag{8}$$

where the predicted HD $d_H(\mathbf{h}_i, \mathbf{h}_j)$ is defined by,

$$d_H((\mathbf{h}_i, \mathbf{h}_j) = \frac{K}{2}(1 - \cos(\mathbf{h}_i, \mathbf{h}_j)) \tag{9}$$

The above formula describes the relationship between cosine similarity and normalized Euclidean distance for hash codes $\mathbf{h}_i$ and $\mathbf{h}_j$ of length K, where $cos(\mathbf{h}_i, \mathbf{h}_j)$ represents the cosine similarity between the two hash codes. The above loss in (8) is deduced from absolute value $|\cdot|$. Given a real number x, we can write,

$$|x| \approx \log(\cosh(x)) \tag{10}$$

$D_H^{(n_{ij}^{(1)},n_{ij}^{(2)})}(\mathbf{h}_i, \mathbf{h}_j)$ and $d_H(\mathbf{h}_i, \mathbf{h}_j)$ respectively can be considered as ground truth and predicted HD for this image pair. The idea is based on the value of $n_{ij}^{(1)}$ and $n_{ij}^{(2)}$; this loss forces the predicted HD to closely trail the ground truth HD.

3.4 Pairwise multi-label classification loss (PMCL)

The PMCL on a train set $\mathbf{X}_{\mathfrak{T}}$ is defined by,

$$
\begin{aligned}
J_2 = - \sum_{\mathbf{x}_i^{\mathfrak{T}}, \mathbf{x}_j^{\mathfrak{T}} \in \mathbf{X}_{\mathfrak{T}}} \Bigg\{ \sum_{l=1}^{L} \Bigg(\mathbf{y}_{il}^{\mathfrak{T}} \log(\sigma(\hat{\mathbf{y}}_{il}^{\mathfrak{T}})) + (1 - \mathbf{y}_{il}^{\mathfrak{T}}) \log(1 - \sigma(\hat{\mathbf{y}}_{il}^{\mathfrak{T}})) \Bigg) \\
+ \Bigg(\mathbf{y}_{jl}^{\mathfrak{T}} \log(\sigma(\hat{\mathbf{y}}_{jl}^{\mathfrak{T}})) + (1 - \mathbf{y}_{jl}^{\mathfrak{T}}) \log(1 - \sigma(\hat{\mathbf{y}}_{jl}^{\mathfrak{T}})) \Bigg) \Bigg\}
\end{aligned}
\tag{11}
$$

where ground truth labels $\mathbf{y}_{il}^{\mathfrak{T}}, \mathbf{y}_{jl}^{\mathfrak{T}} \in \{0, 1\}$ indicates whether the l-th label is present in samples $\mathbf{x}_i^{\mathfrak{T}}$ and $\mathbf{x}_j^{\mathfrak{T}}$. $\sigma(\cdot)$ is sigmoid function. $\hat{\mathbf{y}}_i^{\mathfrak{T}}, \hat{\mathbf{y}}_j^{\mathfrak{T}}$ are the predicted classes of $\mathbf{x}_i^{\mathfrak{T}}, \mathbf{x}_j^{\mathfrak{T}}$ respectively. These are obtained from the classification network $\texttt{net}_{\texttt{c}}(\cdot)$. $\texttt{net}_{\texttt{c}}(\cdot)$ is defined as,

$$
\texttt{net}_{\texttt{c}}(\cdot) \mapsto \texttt{Linear : 4096} \to \texttt{ReLU} \to \texttt{Linear : L} \tag{12}
$$

Since multi-label classification serves as a fundamental loss function to generate distinctive features for various combinations of pathologies present in the images, obtaining accurate feature vectors $\mathbf{z}_i$ and $\mathbf{z}_j$ is crucial for deriving accurate representations $\mathbf{h}_i$ and $\mathbf{h}_j$ for an image pair. For this purpose, we utilize $\texttt{net}_{\texttt{c}}(\cdot)$, applying the PMCL loss to these feature vectors, which are the output of $\texttt{net}_{\texttt{e}}(\cdot)$.

3.5 Overall loss

The overall loss is computed as,

$$
J = \lambda_1 J_1 + \lambda_2 J_2 \tag{13}
$$

where, λ_1 and λ_2 is scale hyperparameters. Minimizing J, thereby updating the parameters of $\texttt{net}_{\texttt{e}}(\cdot)$, $\texttt{net}_{\texttt{c}}(\cdot)$, and $\texttt{fc}_{\texttt{h}}(\cdot)$, which enable us to achieve our objectives. The overall training procedure is illustrated in Figure 3.

4 Experiments

4.1 Experimental setup

We have implemented a modified AlexNet architecture [17] to build the $\texttt{net}_{\texttt{e}}(\cdot)$ and $\texttt{fc}_{\texttt{h}}(\cdot)$. $\texttt{net}_{\texttt{c}}(\cdot)$ comprises of two linear layers inherited from $\texttt{net}_{\texttt{e}}(\cdot)$. Adam optimizer [16] is used to learn the parameters of the above three networks. The weight decay parameter and batch size are set to 5×10^{-3} and 512, respectively. The training was initialized with a learning rate of 1×10^{-4}, and then the learning rate scheduler was used with patient 40 and factor 0.4. The values of hyperparameters $\lambda_1 = 1$ and $\lambda_2 = 1.5$ in (13). λ_1 and λ_2 are chosen through hyperparameter tuning using random search within the range $[0, 5]$ with a step size of 0.5. The experiments are conducted on a server equipped with 2× Intel Xeon 4110 CPUs, 12 × 8 GB DDR4 ECC Reg. RAM, 2 × 4 TB HDD, 4× Nvidia GTX 1080Ti GPUs, each with 11 GB DDR5 RAM, and Ubuntu 20.04 LTS operating system. The algorithms are implemented using Python 3.9 with PyTorch 1.11 and CUDA 11.2.

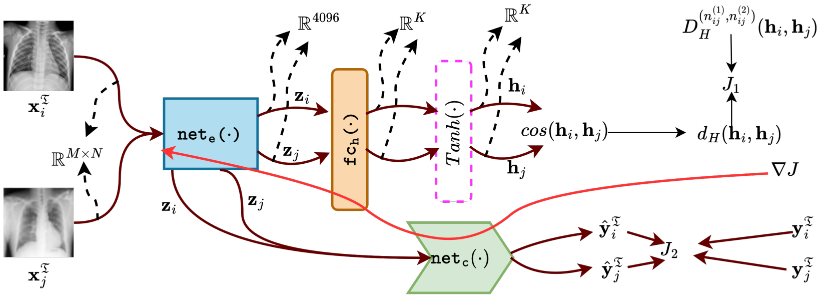

$\mathbf{x}_i^{\mathfrak{T}}$
$\mathbb{R}^{M\times N}$
$\mathbf{x}_j^{\mathfrak{T}}$
$\mathtt{net_e}(\cdot)$
$\mathbb{R}^{4096}$
$\mathbf{z}_i$
$\mathbf{z}_j$
$\mathbf{z}_i$
$\mathbf{z}_j$
$\mathtt{fc_h}(\cdot)$
$\mathbb{R}^K$
$\mathbb{R}^K$
$Tanh(\cdot)$
$\mathbf{h}_i$
$\mathbf{h}_j$
$cos(\mathbf{h}_i, \mathbf{h}_j)$
$D_H^{(n_{ij}^{(1)}, n_{ij}^{(2)})}(\mathbf{h}_i, \mathbf{h}_j)$
J_1
$d_H(\mathbf{h}_i, \mathbf{h}_j)$
∇J
$\mathtt{net_c}(\cdot)$
$\hat{\mathbf{y}}_i^{\mathfrak{T}}$
$\hat{\mathbf{y}}_j^{\mathfrak{T}}$
J_2
$\mathbf{y}_i^{\mathfrak{T}}$
$\mathbf{y}_j^{\mathfrak{T}}$

Fig. 3: An overview of training procedure of our method. Given an image pair $\mathbf{x}_i^{\mathfrak{T}}, \mathbf{x}_j^{\mathfrak{T}}$ along with their corresponding pair label sets $\mathbf{y}_i^{\mathfrak{T}}, \mathbf{y}_j^{\mathfrak{T}}$, we first compute the ground truth HD $D_H^{(n_{ij}^{(1)}, n_{ij}^{(2)})}(\mathbf{h}_i, \mathbf{h}_j)$. Next, we pass the generated feature vectors $\mathbf{z}_i$ and $\mathbf{z}_j$ from the network $\mathtt{net_e}(\cdot)$ through both the networks $\mathtt{fc_h}(\cdot)$. The real-valued hash vectors $\mathbf{h}_i$ and $\mathbf{h}_j$ are then obtained after applying $Tanh(\cdot)$ on the outputs of $\mathtt{fc_h}(\cdot)$. Subsequently, utilizing (8), we calculate the loss terms J_1. J_2 is computed using (11) based on the predicted class by $\mathtt{net_c}(\cdot)$. Minimizing the total loss J allows for the optimization of the weights of $\mathtt{net_e}(\cdot)$, $\mathtt{fc_h}(\cdot)$, and $\mathtt{net_c}(\cdot)$.

4.2 Dataset

We have utilized the publicly available NIH Chest X-ray database [1], which consists of 112,120 frontal-view X-ray images from 30,805 unique patients. Each image is associated with one or more of the 14 common thoracic pathologies identified from the accompanying radiological reports. We have selected 51,480 images depicting the 13 most frequent pathologies, including Atelectasis, Consolidation, Infiltration, Pneumothorax, Edema, Emphysema, Fibrosis, Effusion, Pneumonia, Pleural thickening, Cardiomegaly, Nodule, and Mass. These images are divided into three non-overlapping image sets: training set ($\mathbf{X}_{\mathfrak{T}}$), gallery set ($\mathbf{X}_G$), and query set ($\mathbf{X}_Q$), where $|\mathbf{X}_{\mathfrak{T}}| = 38,610$, $|\mathbf{X}_G| = 10,296$ and $|\mathbf{X}_Q| = 2,574$.

4.3 Evaluation metrics

The metrics most commonly used by multi-label CBMIR methods are: normalized discounted cumulative gain (nDCG), average cumulative gain (ACG), and weighted mean average precision (wMAP) [23]. These metrics provide a comprehensive evaluation by assessing ranking quality, overall relevance, and precision-recall balance respectively. During training, we use the normalized relevance score (i.e., JSC). During evaluation, we use only the relevance score, adhering to the same strategy as existing methods on multi-label image retrieval [31].

[1] https://www.kaggle.com/datasets/nih-chest-xrays/data

Normalized discounted cumulative gain (nDCG) In order to compute $nDCG@p$ for top-p retrieval, first we need to calculate $DCG@p$ for top-p retrieval. The mathematical formulation for $DCG_q@p$ of query image $\mathbf{x}_q^Q$ is given by,

$$DCG_q@p = \sum_{r=1}^{p} \frac{2^{R_q(r)} - 1}{log_2(r+1)} \tag{14}$$

where relevance score $R_q(r)(= |\mathbf{y}_q^Q \cap \mathbf{y}_r^G|)$ represents the number of pathologies between $\mathbf{x}_q^Q \in \mathbf{X}_Q$ and $\mathbf{x}_r^G \in \mathbf{X}_G$ are matched.

We normalize this by dividing it with the maximally achievable value or Ideal DCG (iDCG). Finally to obtain,

$$nDCG_q@p = \frac{DCG_q@p}{iDCG_q@p} \tag{15}$$

where $iDCG_q@p = DCG_q@p$ of ideal ranking or best possible ranking.

Finally,

$$nDCG@p = \frac{1}{|\mathbf{X}_Q|} \sum_{\mathbf{x}_q^Q \in \mathbf{X}_Q} nDCG_q@p \tag{16}$$

Average cumulative gains (ACG) The $ACG@p$ metric quantifies the cumulative similarity between a query image and the $top - p$ retrieved images. It is calculated by summing the similarities between the query image and each of the $top - p$ retrieve in the retrieval list. $ACG@p$ can be formulated as,

$$ACG@p = \frac{1}{|\mathbf{X}_Q|} \sum_{\mathbf{x}_q^Q \in \mathbf{X}_Q} \sum_{r=1}^{p} \frac{R_q(r)}{p} \tag{17}$$

Weighted mean average precision (*wMAP*) The $wMAP$ can be formulated by,

$$wMAP = \frac{1}{|\mathbf{X}_Q|} \sum_{\mathbf{x}_q^Q \in \mathbf{X}_Q} \left(\frac{\sum_{r=1}^{p} \delta(R_q(r) > 0) ACG@r}{\sum_{r=1}^{p} (\delta(R_q(r) > 0))} \right) \tag{18}$$

where $\delta(\cdot)$ is the indicator function.

5 Results and Discussions

5.1 Comparison with existing methods

Since our method is based on pairwise learning, we compare it with four recent state-of-the-art (SOTA) pairwise multi-label DNH methods IDHN [31], DCH [4], OrthoHash [12], CSQ [30], HSDH [1]. The same network i.e., AlexNet, is used for a fair comparison. The parameters of SOTA models are

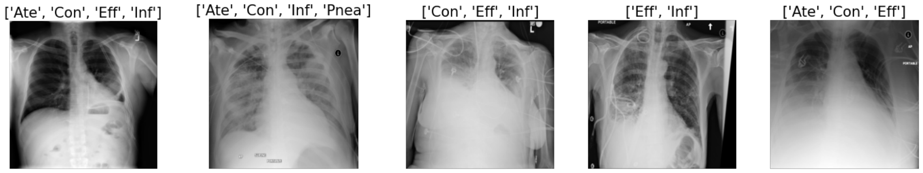
Table 2: Comparison of nDCG and ACG with pairwise deep hashing methods for different hash code lengths. - indicates that it is not evaluated since it is not applicable for the specific hash code length.

Method	$nDCG$@100				ACG@100			
	16	32	48	64	16	32	48	64
IDHN [31]	0.5900	0.5931	0.5797	0.5955	0.3283	0.3620	0.3323	0.3158
DCH [4]	0.5916	0.6139	0.6005	0.6084	0.3330	0.3550	0.3363	0.3383
OrthoHash [12]	0.5905	0.5947	0.6244	0.5916	0.3294	0.3387	0.3463	0.3256
CSQ [30]	0.5905	0.6215	-	0.6194	0.3308	0.3780	-	0.3624
HSDH [1]	0.5920	0.5917	0.6059	0.5910	0.3330	0.3326	0.3416	0.3309
Ours	**0.6318**	**0.6363**	**0.6426**	**0.6362**	**0.3874**	**0.3869**	**0.4028**	**0.3930**

Table 3: Comparison of $wMAP$ with the SOTA for different hash code lengths.

Method	$wMAP$			
	16	32	48	64
IDHN [31]	0.3619	0.4016	0.3776	0.3705
DCH [4]	0.3671	0.4174	0.3903	0.3990
OrthoHash [12]	0.3626	0.3713	0.4318	0.3644
CSQ [30]	0.3642	0.4377	-	0.4319
HSDH [1]	0.3674	0.3660	0.3895	0.3773
Ours	**0.4572**	**0.4600**	**0.4767**	**0.4664**

selected according to those specified in their respective original publications. We evaluate four variants of our proposed methods with four different hash code lengths $K = \{16, 32, 48, 64\}$. The comparison results for nDCG and ACG are presented in Table 2, while those for $wMAP$ are presented in Table 3. These demonstrate the substantial superiority of our proposed method over SOTA methods. CSQ achieves the best results for $K = 32$ hash code length across all SOTA compared. Our method shows an improvements over CSQ, with an approximately improvement of $4.13\%, 1.48\%, 1.68\%$ for hash code lengths $K = \{16, 32, 64\}$ respectively in terms of $nDCG$@100. Furthermore, our method demonstrates relative increases of $5.66\%, 0.89\%, 3.06\%$ in ACG@100, and $9.30\%, 3.23\%, 3.45\%$ in $wMAP$ for hash code lengths $K = \{16, 32, 64\}$ respectively over CSQ. CSQ is not applicable for $K = 48$. Notably, our method achieves $1.82\%, 5.65\%, 4.49\%$ higher $nDCG$@100, ACG@100, and $wMAP$ than OrthoHash for $K = 48$ respectively. Our method surpasses HSDH with improvements of 3.98%, 5.44%, and 8.98% in $nDCG$@100, ACG@100, and $wMAP$, respectively, for $K = 16$. Our method demonstrates comparable or even superior performance with shorter binary codes. For instance, our method with a 16-bit binary code significantly outperforms all other hashing methods by large margins for each metric. These results highlight the effectiveness of our proposed method for retrieving multi-label medical images.



(a) $\mathbf{x}_1^{\mathfrak{T}}(\mathbf{h}_1)$ (b) $\mathbf{x}_2^{\mathfrak{T}}(\mathbf{h}_2)$ (c) $\mathbf{x}_3^{\mathfrak{T}}(\mathbf{h}_3)$ (d) $\mathbf{x}_4^{\mathfrak{T}}(\mathbf{h}_4)$ (e) $\mathbf{x}_5^{\mathfrak{T}}(\mathbf{h}_5)$

Fig. 4: The above figures depict five random images selected from the training set and generated hash codes, with their respective labels annotated. Atelectasis ('Ate'), Pneumonia ('Pnea'), Consolidation ('Con'), Effusion ('Eff'), Infiltration ('Inf').

5.2 Learning of Hamming distances during training

We consider five randomly picked training images $\mathbf{x}_1^{\mathfrak{T}}, \mathbf{x}_2^{\mathfrak{T}}, \mathbf{x}_3^{\mathfrak{T}}, \mathbf{x}_4^{\mathfrak{T}}, \mathbf{x}_5^{\mathfrak{T}}$ along with their hash codes $\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3, \mathbf{h}_4, \mathbf{h}_5$. These images are illustrated in Figure 4 with their labels. Here $\frac{n_{12}^{(2)}}{n_{12}^{(1)}} = \frac{3}{5}, \frac{n_{13}^{(2)}}{n_{13}^{(1)}} = \frac{3}{4}, \frac{n_{14}^{(2)}}{n_{14}^{(1)}} = \frac{2}{4} = \frac{1}{2}, \frac{n_{15}^{(2)}}{n_{15}^{(1)}} = \frac{3}{4}$. The ground truth Hamming distances are calculated using (6) and (7), while the predicted Hamming distances are calculated using (9). The detailed calculations are presented in Table 4. It can be observed that the predicted Hamming distance ($d_H(\cdot)$) closely approximates the ground truth Hamming distance ($D_H(\cdot)$) between all pairs of hash codes for $K = \{16, 32\}$. For $K = 48$, the distances are close except for the hash pair $(\mathbf{h}_1, \mathbf{h}_4)$, and for $K = 64$, except for $(\mathbf{h}_1, \mathbf{h}_2)$, and $(\mathbf{h}_1, \mathbf{h}_4)$. The mean and standard deviation of the ground truth HD and predicted HD are presented in Table 4. These results show that smaller hash codes exhibit less disparity between these HD values. These findings confirm that our approach is particularly effective when employing shorter hash code lengths for these five images that are analyzed.

5.3 Top retrieved images

In Figure 5, we present retrieved images obtained using our approach. Images exhibiting more similarity in pathologies with the query image are prioritized to appear at the top of the ranking. This verifies the capability of our method to maintain multi-level similarity, thereby retrieving images that offer a higher level of similarity for enhanced assessment support.

Table 4: The comparison between ground truth HD ($D_H(\cdot)$) and predicted HD ($d_H(\cdot)$) for five randomly selected images (depicted in Figure 4) across different hash code lengths. The last column indicates the mean and standard deviation (SD) between these HD.

	Hash code pairs	$(\mathbf{h}_1, \mathbf{h}_2)$	$(\mathbf{h}_1, \mathbf{h}_3)$	$(\mathbf{h}_1, \mathbf{h}_4)$	$(\mathbf{h}_1, \mathbf{h}_5)$	mean ± SD
$K = 16$	ground truth HD	6	4	8	4	5.5 ± 1.65
	predicted HD	5.16	3.74	7.78	4.04	5.18 ± 1.59
$K = 32$	ground truth HD	12	8	16	8	11 ± 3.31
	predicted HD	13.29	9.71	13.67	7.55	11.05 ± 2.54
$K = 48$	ground truth HD	19	12	24	12	16.75 ± 5.06
	predicted HD	17.47	15.78	21.24	25.33	19.95 ± 3.67
$K = 64$	ground truth HD	25	16	32	16	22.25 ± 6.72
	predicted HD	17.27	14.45	17.10	13.23	15.51 ± 1.72

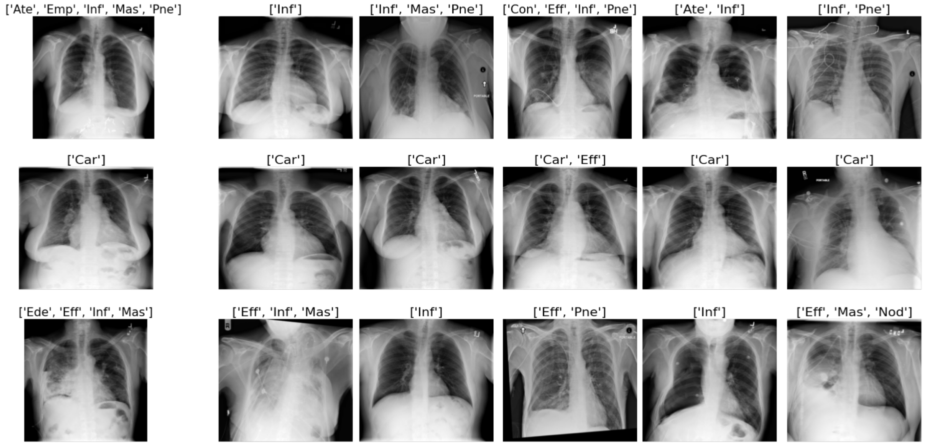


Fig. 5: Qualitative results for our method. The images on the left side represent the query images, while those on the right side depict the top-5 retrieved images. Atelectasis ('Ate'), Pneumonia ('Pnea'), Consolidation ('Con'), Effusion ('Eff'), Infiltration ('Inf'), Pneumothorax ('Pne'), Mass ('Mas'), Cardiomegaly ('Car'), Edema ('Ede').

6 Conclusion

In this work, we have developed an effective CBMIR system tailored for large-scale multi-label CBMIR. Our approach leverages DNH, which learns HD between hash codes to generate image-specific hash codes. We design a loss function for effective HD learning using the JSC between image labels. Through extensive experiments conducted with a publicly available multi-label medical image datasets, our proposed method demonstrated superior performance compared to existing methods. It effectively learns features and hash codes, enhancing the performance of multi-label CBMIR. This work provides insight into the advantageous nature of leveraging the complementarity between image labels for hash learning and their HD.

References

1. Alizadeh, S.M., Helfroush, M.S., Müller, H.: A novel siamese deep hashing model for histopathology image retrieval. Expert Syst. Appl. **225**, 120169 (2023)
2. Bag, S., Kumar, S.K., Tiwari, M.K.: An efficient recommendation generation using relevant jaccard similarity. Inf. Sciences **483**, 53–64 (2019)
3. Cai, T.W., Kim, J., Feng, D.D.: Content-based medical image retrieval. In: Biomed. Inf. Technol., pp. 83–113. Elsevier (2008)
4. Cao, Y., Long, M., Liu, B., Wang, J.: Deep cauchy hashing for hamming space retrieval. In: Proc. Conf. Comp. Vision Pattern Recognit. pp. 1229–1237 (2018)
5. Cao, Z., Long, M., Wang, J., Yu, P.S.: Hashnet: Deep learning to hash by continuation. In: Proc. Conf. Comp. Vision Pattern Recognit. pp. 5608–5617 (2017)
6. Chen, Y., Tang, Y., Huang, J., Xiong, S.: Multi-scale triplet hashing for medical image retrieval. Computers Biol. Medicine **155**, 106633 (2023)
7. Chen, Z., Cai, R., Lu, J., Feng, J., Zhou, J.: Order-sensitive deep hashing for multimorbidity medical image retrieval. In: Proc. Med. Image Comput. Comput. Assisted Intervention. pp. 620–628. Springer (2018)
8. Das, P., Neelima, A.: An overview of approaches for content-based medical image retrieval. Int. J. Multimedia Inf. Retrieval **6**(4), 271–280 (2017)
9. Doan, K.D., Yang, P., Li, P.: One loss for quantization: Deep hashing with discrete wasserstein distributional matching. In: Proc. Conf. Comp. Vision Pattern Recognit. pp. 9447–9457 (2022)
10. Fang, J., Fu, H., Liu, J.: Deep triplet hashing network for case-based medical image retrieval. Med. Image Anal. **69**, 101981 (2021)
11. Guo, X., Duan, J., Gichoya, J., Trivedi, H., Purkayastha, S., Sharma, A., Banerjee, I.: Multi-label medical image retrieval via learning multi-class similarity. Available at SSRN 4149616 (2022)
12. Hoe, J.T., Ng, K.W., Zhang, T., Chan, C.S., Song, Y.Z., Xiang, T.: One loss for all: Deep hashing with a single cosine similarity based learning objective. Advances Neural Inf. Process. Syst. **34**, 24286–24298 (2021)
13. Hou, D., Zhao, Z., Hu, S.: Multi-label learning with visual-semantic embedded knowledge graph for diagnosis of radiology imaging. IEEE Access **9**, 15720–15730 (2021)
14. Indyk, P., Motwani, R.: Approximate nearest neighbors: towards removing the curse of dimensionality. In: Proc. ACM Symp. Theory Cmput. pp. 604–613 (1998)
15. Jose, A., Filbert, D., Rohlfing, C., Ohm, J.R.: Deep hashing with hash center update for efficient image retrieval. In: IEEE Int. Conf. Acoust. Speech and Signal Proc. pp. 4773–4777. IEEE (2022)
16. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. In: Int. Conf. Learn. Representations. San Diega, CA, USA (2015)
17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances Neural Inf. Process. Syst. **25** (2012)

18. Kumar, A., Kim, J., Cai, W., Fulham, M., Feng, D.: Content-based medical image retrieval: a survey of applications to multidimensional and multimodality data. J. Digit. Imag. **26**, 1025–1039 (2013)
19. Liu, H., Wang, R., Shan, S., Chen, X.: Deep supervised hashing for fast image retrieval. In: Proc. Conf. Comp. Vision Pattern Recognit. pp. 2064–2072 (2016)
20. Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., Alsaadi, F.E.: A survey of deep neural network architectures and their applications. Neurocomputing **234**, 11–26 (2017)
21. Luo, X., Wang, H., Wu, D., Chen, C., Deng, M., Huang, J., Hua, X.S.: A survey on deep hashing methods. ACM Trans. Knowl. Discovery Data **17**(1), 1–50 (2023)
22. Müller, H., Deserno, T.M.: Content-based medical image retrieval. In: Biomed. Image Process., pp. 471–494. Springer (2010)
23. Rodrigues, J., Cristo, M., Colonna, J.G.: Deep hashing for multi-label image retrieval: a survey. Artif. Intell. Rev. **53**(7), 5261–5307 (2020)
24. Singh, A., Gupta, S.: Learning to hash: a comprehensive survey of deep learning-based hashing methods. Knowl. Inf. Syst. **64**(10), 2565–2597 (2022)
25. Sorower, M.S.: A literature survey on algorithms for multi-label learning. Oregon State University, Corvallis **18**(1), 25 (2010)
26. Su, S., Zhang, C., Han, K., Tian, Y.: Greedy hash: Towards fast optimization for accurate hash coding in cnn. Advances Neural Inf. Process. Syst. **31** (2018)
27. Tagare, H.D., Jaffe, C.C., Duncan, J.: Medical image databases: A content-based retrieval approach. J. Amer. Med. Inform. Assoc. **4**(3), 184–198 (1997)
28. Tang, Y., Chen, Y., Xiong, S.: Deep semantic ranking hashing based on self-attention for medical image retrieval. In: Int. Conf. Pattern Rec. pp. 4960–4966. IEEE (2022)
29. Wang, J., Zhang, T., Sebe, N., Shen, H.T., et al.: A survey on learning to hash. IEEE Trans. Pattern Anal. Mach. Intell. **40**(4), 769–790 (2017)
30. Yuan, L., Wang, T., Zhang, X., Tay, F.E., Jie, Z., Liu, W., Feng, J.: Central similarity quantization for efficient image and video retrieval. In: Proc. Conf. Comp. Vision Pattern Recognit. pp. 3083–3092 (2020)
31. Zhang, Z., Zou, Q., Lin, Y., Chen, L., Wang, S.: Improved deep hashing with soft pairwise similarity for multi-label image retrieval. IEEE Trans. Multimedia **22**(2), 540–553 (2019)