Design and implementation of a Content-based medical image retrieval (CBMIR) system for radiology images

Izborni projekt - Strojno učenje, RITEH diplomski studij Računarstva, 2021

Student: Dario Barać, mentor: izv. prof. dr. sc. Ivan Štajduhar

***Abstract*—In this study, a Content-based image retrieval system for radiology images of various anatomical regions and imaging modalities was designed and implemented. Three retrieval methods were evaluated: BoVW, retrieval with HOG descriptors and retrieval with CNN-extracted descriptors. The configuration for each method was chosen empirically by comparing results on a validation image set. Retrieval with CNNs pretrained on ImageNet outperformed both BoVW and HOG retrieval. Re-ranking retrieved images with a different method improved results in almost all cases. Out of all methods evaluated on the test set, CNN retrieval (ShuffleNet + average pooling for dimensionality reduction) with HOG re-ranking performed the best.**

***Index Terms*—CBIR, CBMIR, radiology, BoVW, HOG, SIFT, CNN, image retrieval**

I. Introduction

Finding relevant images in large image databases can be challenging. The traditional approach is searching based on keywords. This requires manual image annotation by humans, which is a time-consuming process. Content-based image retrieval (CBIR) is an alternative approach which attempts to index images automatically using visual features such as image texture, shape or color distribution. Relevant images are retrieved by extracting visual features from a given query image and comparing them with the features of images in the database. CBIR research started in the early 1990s. Since then, a variety of methods have been develeloped and applied to many problems, such as landmark recognition or general reverse image search engines.

CBMIR (Content based medical image retrieval) systems apply CBIR methods to assist in medical diagnosis, education and research. The goal of this paper is to design and implement a CBMIR system for radiology images and evaluate the systems performance.

II. Background

A. General CBIR framework and steps

The scope of CBIR systems (their purpose and image domain), effects of sensory and semantic gaps and options for each of the necessary computational steps (image processing, feature extraction, similarity measurement) are discussed in "Content-Based Image Retrieval at the End of the Early Years" [1], a review of 200 papers from the 1990s.

A short summary of parts of the paper is given in this subsection.

1) Scope: The first step in designing a CBIR system is defining its scope. CBIR systems usually have one of three purposes:

- Target search: finding more images of the exact object as found in the query image.
- Category search: finding images of objects that belong to the same class as the object in the query.
- Search by association: finding images that are in some way related to the query. Relevance feedback by the user is often used to iteratively refine the search results.

A clear purpose helps us choose features to consider and to define measures of simliarity (eg. similarity measures for target search should be strict to ensure that we find the same object as in the image). The domain of images under consideration can be narrow (limited variability in its appearance), broad (unlimited and unpredictable variability) or somewhere in between. In the case of narrow domains, image semantics are usually clearly defined and often unique. On the other hand, images from broad domains usually have only partial (or many possible) semantic descriptions - there is a large gap between computed feature descriptions and the descriptions given by humans (the semantic gap).

Another source of ambiguity is the sensory gap - the gap between real life objects and their description captured by sensors (multiple 3D scenes can be mapped into the same 2D image, loss of information by occlusion, clutter and illumination). Explicit knowledge about the problem domain is used to alleviate the sensory gap.

2) Image processing: CBIR systems rely on useful descriptions of image content. The first step needed to acquire these descriptions is image processing. The goal is to enhance relevant information in images and reduce image aspects that are not relevant (eg. image noise). This is done by transforming each pixel based on color, shape or textures in its local neighbourhood. These operations can be used to generate image descriptions that are invariant to a variety of

conditions (such as lightning or viewing angle) and carry more object specific information. Descriptions that are invariant to many different conditions can lose the ability to discriminate between important diferences in images, so the degree of invariance should be chosen carefully.

3) Feature extraction: The next step after image processing is feature extraction: selecting relevant points in images and computing their feature values. It can be useful to divide images into parts before computing features (eg. segmentation based on color or texture homogeneity). Several classes of features are frequently used: accumulating features, salient features and signs. Accumulating features agregate information over a selected area (eg. a global color histogram for the entire image). Salient features are stored only for the most conspicuous regions in a partitioned image. Signs are symbols (eg. a character or icon) which have a strong relation to the meaning of an image. The presence of a sign in an image implies its meaning with a high probability. Features can optionally be stored in a graph with spatial relations between them, which can make matching images of the same object easier.

4) Similarity definition: To find images similar to a query image it is necessary to define a similarity measure. There are multiple options for this. One is learning conditional probabilites of possible semantic interpretations for given feature vectors and comparing them for different images. Another option is measuring distance (often Euclidean) between the feature vectors. If features are computed for salient points only, images can be said to be similar when there are enough feature vectors with low distance. If spatial relations between image features are stored, the structures can be compared for different images.

5) Query formation: There are multiple options for defining a query. For example a user could draw a shape outline or give one or multiple example images. Optionally, the user could also select relevant features types (eg. shape or texture) and specify the degree and type of invariance that is appropriate. The query answer is usually one or multiple images, ranked by similarity to the input. User feedback can be used to refine results.

6) System aspects: For large datasets computational performance cannot be ignored, especially if real-time system response is required. To avoid computing the similarity of the query image with all images in the database, images can be indexed by their calculated feature vectors, for example by storing them in a k-d tree. A potential problem is the degradation of performance when the feature space is high-dimensional.

III. Related work

A. Content-based medical image retrieval

Reference [2] is a 2017 review of approaches for large scale medical image retrieval. The paper describes challenges (system efficiency and accuracy), opportunities in the medical field (computer-aided diagnostics, visual pattern exploration) existing applications, datasets and methods for feature extraction, indexing and search. According to the paper, many medical retrieval systems use hand-crafted local feature descriptors. SIFT [3], SURF (Speeded up robust features) [4] and LBP (Local binary patterns) [5] and frequently used, often with Bag-of-Visual-Words (BoVW) [6] models for feature quantization. Global feature descriptors such as GIST [7], HOG [8] or color histograms are also used for medical image retrieval. The alternative to hand-crafted using features is learning useful feature representations from the dataset. For example, a convolutional stacked autoencoder has been used for feature learning for brain MR image registration [9].

To avoid comparing the query image feature vector with all feature vectors in the database, hierarchical clustering and vocabulary trees [10] have been used in large scale medical image retrieval systems to improve retrieval efficiency. Another method used for image indexing is feature hashing: feature vectors are hashed into short binary codes, which can be efficiently compared (for finding the nearest neighbours for an image) by calculating the Hamming distance. The hashing functions can be data-independent [11] or learned on a given set of images [12]. The dimensionality of large feature vectors can also be reduced with Principal component analysis (PCA) [13], or by pruning redundant words from visual vocabularies with probabilistic latent semantic analysis (PLSA) [14]. After finding the nearest neighbours of the query image in the database, re-ranking the retrieved images can lead to more accurate results. For example re-ranking can be done by comparing images using different types of features or by asking the user for relevance feedback. For evaluation of systems, metrics such as precision, recall and mean average precision (mAP) are often used.

B. CBMIR applied to radiology

For texture features, Fourier transforms, wavelets and Gabor filters have been used in radiology applications [15]. Edges, contours, polylines and polygonals have been used to describe shape features. In [16], the authors used Radon features and barcodes in combination with support vector machines (SVMs) for x-ray image retrieval. In [17], Gabor Wavelets, Gray Level Co-occurrence Matrix (GLCM), Radon Transforms, Orientation Histograms and Fourier Descriptors were used on regions of interest selected by radiologists for the retrieval of spine vertebrae irregularity images. Many content based image retrieval systems made before 2011 are mentioned in [15].

IV. Problem statement and system description

A. Problem

The goal of this project is to design and implement a Content-based image retrieval system for radiology images. The purpose of the system is category search, ie. the system should retrieve images from the database which belong to the same class as the query image, wrt. the anatomical region in the image (eg. chest images), the imaging modality (eg. MR images) or both (eg. MR images of the head).

Anatomical region	Train dataset count	Test dataset count
Chest	5123	1587
Urinary tract	5000	1961
Colon	3672	1011
Abdomen	3351	758
Head	3239	555
Cerebrala	3062	879
Breast	1104	432
L-spine	944	65
C-spine	877	128
Knee	505	55
Shoulder	477	68
Heart	369	69
Arm	337	15
Pelvis	306	118
Extremity	292	70
Foot	282	63
Neck	260	59
Ankle	231	16
Hand	202	15
Spine	162	22
Whole body	77	40
Lower extrem	70	5
T-spine	42	12
Extrem	16	6

TABLE I: Number of images per anatomical region in the train and test datasets

Imaging modality	Train dataset count	Test dataset count
RF	5123	1587
XA	5000	1961
CT	3672	1011
MR	3351	758
CR	3239	555
NM	3062	879

TABLE II: Number of images per imaging modality in the train and test datasets

The system should take as its input a single query image and the value for the k parameter, which is the number of images the system should retrieve from the database and return to the user. After comparing the query image with the images stored in the systems database by using a selected distance measure, the system should return the top-k most similar images to the user.

B. Dataset

The dataset used in this study consists of 38000 radiology images of 24 anatomical regions and 6 modalities. The imaging modalities are Radio Fluoroscopy (RF), X-Ray Angiography (XA), Computed Tomography (CT), Magnetic Resonance (MR), Computed Radiography (CR) and Nuclear Medicine (NM). 30000 images are used for training, algorithm parameter selection and validation, and the remaining 8000 images are used for testing. All images are grayscale and have a fixed size of 256×256. The number of images of each anatomical region in the training and test sets is shown in table I. The number of images per imaging modality is shown in table II.

C. System description

Three methods were used for image retrieval: a Bag-of-visual-words model with SIFT descriptors, HOG (Histogram-of-Oriented-Gradients) and convolutional neural networks. These methods are described in the next section. In general (regardless of the method used), the system operates in two phases: offline and online. The offline phase is necessary for extracting and storing vectors for representing images from the training set (the set of images which can be retrieved). In the online phase, the user provides a query image and the system returns similar images. Optionaly, retrieved images can be reranked with another method. Both phases are described in figure 1.

The dataset image descriptors computed in the offline phase are simply stored in an array and loaded into memory at runtime. In the online phase, after computing the query image descriptor, the distance between the query descriptor and all loaded dataset image descriptors is computed. After sorting the dataset images by distance, the top-k most similar image IDs are returned to the user. Since the number of images in the training set is not too large (30000), it is possible to do this with an interactive retrieval time (under 1s). For larger datasets more advanced indexing techniques should be used.

V. BoVW with SIFT features

A. Local feature extraction with SIFT

The first step for using the BoVW model is local feature extraction. In this study SIFT (Scale-invariant feature transform) was used for local feature detection and extraction.

To detect scale-invariant features, SIFT uses a Difference-of-Gaussians (DoG) keypoint detector, which is an approximation of the Laplacian-of-Gaussian computed with a discrete Gaussian image pyramid. The image pyramid is generated by bluring and downsampling the original image multiple times. Image keypoints are selected by finding local extrema in the DoG image pyramid and discarding points which are on edges or have low-contrast. Each keypoint has a characteristic scale and orientation, which are used for computing the SIFT descriptor.

The SIFT descriptor is generated by computing gradient directions for pixels in the area surrounding the keypoint, dividing the area into blocks, calculating histograms of gradient directions for each block, and finally, concatenating the histograms. The keypoint scale and orientation are used during these steps to make sure the descriptors are scale and rotation invariant. Detailed steps for SIFT keypoint detection and descriptor computation can be found in [18].

B. Visual vocabulary learning

Since many SIFT features are found in each image (50-350 were found per train dataset image), it would be computationaly expensive to directly compare features between all images in the dataset and the query image. The BoVW model quantizes all local features found in an image, aggregates them and describes the image with a single fixed-length vector. To quantize local SIFT features it is necessary to first build the

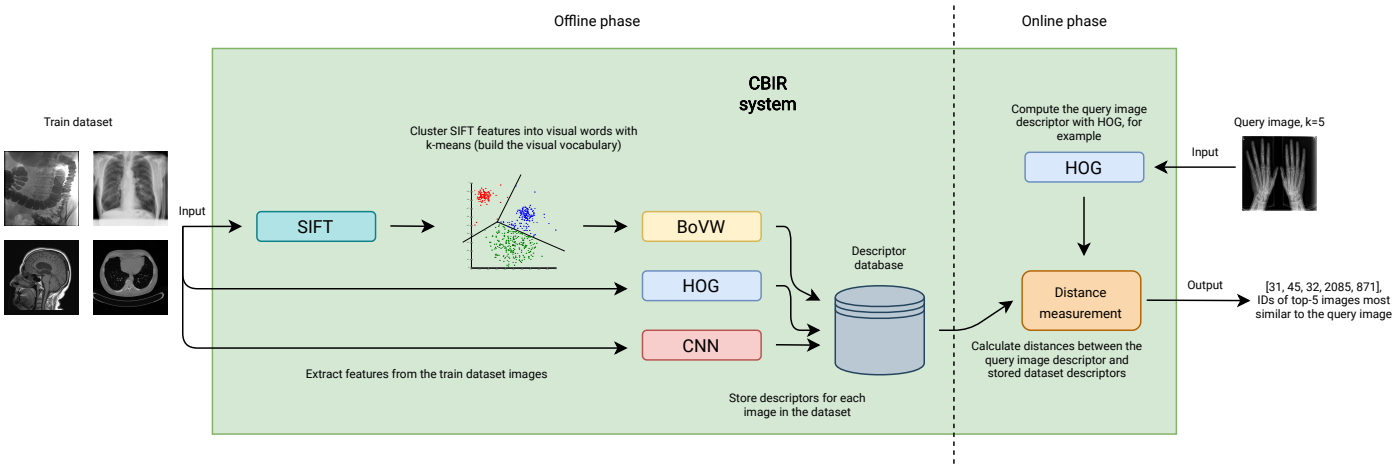

Fig. 1: The image retrieval system:
The system operates in two phases: offline and online. In the offline phase, BoVW, HOG and CNN descriptors for the training set images are extracted and stored. Before computing BoVW descriptors, local SIFT features are extracted and clustered with k-means for building the visual vocabulary. In the online phase, the user provides a query image and the system returns similar images. Images are retrieved by measuring the distance between the computed query image descriptor (BoVW, HOG or CNN) and the stored dataset image descriptors.

visual vocabulary. This is done by clustering all SIFT features extracted from the train dataset by running k-means, with $k = n_{words}$ (the size of the vocabulary). The vocabulary is the set of n visual words ie. the n most representative SIFT descriptors from the dataset, which are selected by taking the cluster centroids generated by k-means.

C. BoVW image descriptor computation

After visual vocabulary learning, the BoVW descriptors for each image can be generated. First, each SIFT descriptor extracted from an image is quantized to the closest visual word. This is done by measuring the Euclidean distance between the descriptor and each visual word (k-means cluster centroid), and assigning the ID of the closest visual word to the SIFT descriptor.

The BoVW descriptor is a histogram of visual word occurences in an image. The dimensionality of the descriptor is the same as the visual vocabulary size (n_{words}). The descriptor is computed by starting with zeros in all positions in the histogram, and then incrementing the position of the assigned visual word ID for each SIFT descriptor. Each image contains only a subset of all visual words, so the BoVW descriptor is a sparse vector.

D. TF-IDF descriptor re-weighting

To improve performance, the BoVW descriptors are re-weighted with TF-IDF [19] (term frequency–inverse document frequency). The re-weigthing aims to increase the bin magnitude for meaningful words (words which rarely occur), and reduce the bin magnitude of words which are not meaningful. Each histogram bin in the descriptor is re-weigthed as follows:

$$t_{id} = \frac{n_{id}}{n_d} log \frac{N}{n_i} \tag{1}$$

- t_{id}: histogram bin of word i for image d
- n_{id}: occurences of word i in image d
- n_d: number of word total occurences in image d
- n_i: number of images that contain word i
- N: number of images

E. Distance measure

For retrieving image, the cosine distance measure [19] is used to compute the distance between the query image descriptor and the stored descriptors of dataset images:

$$d_{cos}(\mathbf{x}, \mathbf{x}) = 1 - \frac{\mathbf{x}^T \mathbf{y}}{||\mathbf{x}||||\mathbf{y}||} \tag{2}$$

The cosine distance ignores the magnitude between the vectors and considers the cosine of the angle between them.

VI. Retrieval with HOG descriptors

The HOG (Histogram-of-Oriented-Gradients) descriptors are computed once globaly for the whole image. The first step is computing the gradient direction and magnitude for each pixel in the image. After computing the gradient (the vector of partial x and y image derivatives):

$$\nabla f(x,y) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} f(x+1,y) - f(x-1,y) \\ f(x,y+1) - f(x,y-1) \end{bmatrix} \tag{3}$$

the magnitude and direction of the gradient can be calculated as follows:

$$g = \sqrt{g_x^2 + g_y^2} \tag{4}$$

$$\theta = arctan(g_y / g_x) \tag{5}$$

For computing the descriptor, the image is divided into a grid of $size_{cell} \times size_{cell}$ sized cells. For each cell, a histogram of

gradient orientations of pixels is computed. The histograms have n_{bins} bins each. Then, individual cell histograms for blocks of 2×2 cells are concatenated into a single vector and normalized. The HOG descriptor is the concatenation of block vectors in each image position. Block positions are determined by sliding the block across the image with stride equal to $stride_{block}$. If stride is equal to cell size, then the block overlap, which can result in better performance.

A. Distance measure

For computing the distance between HOG image descriptors L_1 distance was used, since it showed better performance than L_2 (Euclidean):

$$d_1(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_1 \tag{6}$$

VII. Retrieval with CNN descriptors

Convolutional neural networks (CNNs) are a class of neural networks frequently used for computer vision tasks such as image classification or object detection. In general, for image classification, CNNs architectures consist of various blocks of convolutional and pooling layers for feature extraction, and one or more fully connected layers at the top of the network.

For image retrieval, the fully connected layers at the top can be removed. Then, image descriptors can be computed by using images as the network input and taking the flattened output of the final convolutional (or pooling) layer [20].

In this study, several CNN architectures were tested in such a manner. Image retrieval performance was tested for ResNet18 [21], ResNet34 [21], AlexNet [22], SqueezeNet [23] and VGG16 [24]. The models are all pretrained on the ImageNet dataset and were not additonally trained. For some models, an average pooling layer was added at the end to reduce the dimensionality of the final output vector.

A. Distance measure

For computing the distance between CNN image descriptors L_2 (Euclidean) distance was used:

$$d_2(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_2 \tag{7}$$

VIII. Methodology

To find appropriate algorithm parameters for BoVW and HOG, select the CNN architecture and compare the performance between the methods, a subset of train dataset images was used for evaluating. This validation dataset consists of 20% of images of each anatomical region (5991 in total). The validation images were not processed in the offline stage (their descriptors were not stored).

A. Evaluation metrics

1) mean top-k Precision (mP@k): For evaluating and visualy comparing the performance between different methods and algorithm parameters, mean top-k precision ($mP@k$) for k values ranging from 1 to 10 was calculated. (k is the number of images to retrieve).

The metric is computed for given k as follows:

$$mP@k = \frac{1}{n_{imgs}} \sum_{i=1}^{n_{imgs}} \frac{\sum_{j=1}^{k} rel(i,j)}{k} \tag{8}$$

$rel(i,j)$ is a binary indicator equal to 1 if the j-th retrieved image for the i-th query image is relevant, and 0 otherwise. An retrieved image is said to be relevant if both the imaging modality and the anatomical region of the query image and the retrieved image are the same (as an alternative, only the anatomical region or the imaging modality can be checked). For the mean precision metric, we first find the precision for each query image (the number of relevant retrieved images divided by k), and then take the mean of precision scores for all query images.

2) mean Average Precision (mAP): To summarize mean top-k precision scores for multiple k values (1,..,k_{max}), mean Average Precision (mAP) was calculated as described in [2]:

$$\begin{aligned} mAP &= \frac{1}{n_{imgs}} \sum_{i=1}^{n_{imgs}} \frac{1}{k_{max}} \sum_{k=1}^{k_{max}} precision(k) \\ &= \frac{1}{n_{imgs}} \sum_{i=1}^{n_{imgs}} \frac{1}{k_{max}} \sum_{k=1}^{k_{max}} \frac{\sum_{j=1}^{k} rel(i,j)}{k} \end{aligned} \tag{9}$$

First, retrieval precision for individual query images is measured for each k value. Then, average precision for each query image is calculated. Finally, the mean of average precision scores for each query image is taken.

B. Experiments and algorithm parameter selection

1) BoVW: A total of 4152087 SIFT descriptors were extracted from the train dataset images. The appropriate visual vocabulary size for BoVW was determined empirically by evaluating the model on the validation image set. Vocabulary sizes of 250, 500, 1000, 2000, 3000 and 4000 were compared. For vocabulary learning, k-means was run for 50 iterations on the SIFT descriptors, with 5 retries for every vocabulary size. Increasing the number of iterations did not result in a performance improvement. The comparison can be found in figure 2. Vocabulary size (n_{words}) 3000 performed the best.

After that, RootSIFT [25], a simple modification of SIFT descriptors was tested with the chosen vocabulary size (3000). For RootSIFT, the SIFT descriptors are modified by first L_1 normalizing them, and then taking the square root for each element in the descriptor. For $n_{words} = 3000$, using RootSIFT descriptors resulted in a 2.2% performance increase for ($mP, k = 1$). Since using RootSIFT improved the performance, more vocabulary sizes were tested (3000, 3500 and 4000). Details are in figure 3. $n_{words} = 3500$ performed the best for low k. For higher k values, $n_{words} = 4000$ was slightly better.

2) HOG: For HOG retrieval, it is necessary to choose values for $size_{cell}, size_{block}, stride_{block}$, and n_{bins} (the number of bins in each cell histogram). First, 6 combinations of cell size, block size and block stride (overlapping and non-overlapping) with a fixed $n_{bins} = 9$ was tested. The

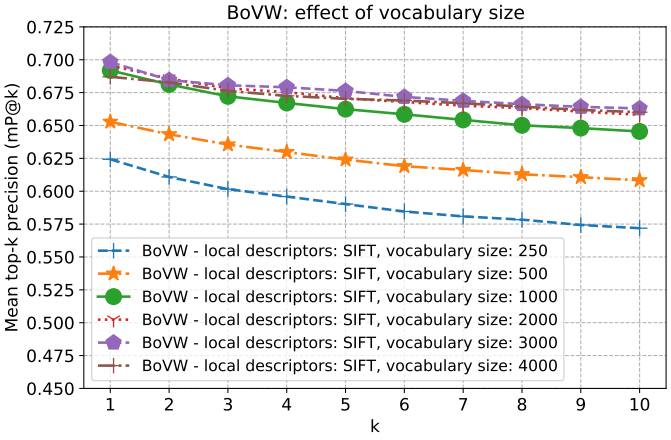

Fig. 2: Performance comparison for BoVW vocabulary sizes: Size over 3000 did not improve the performance for BoVW with SIFT descriptors.

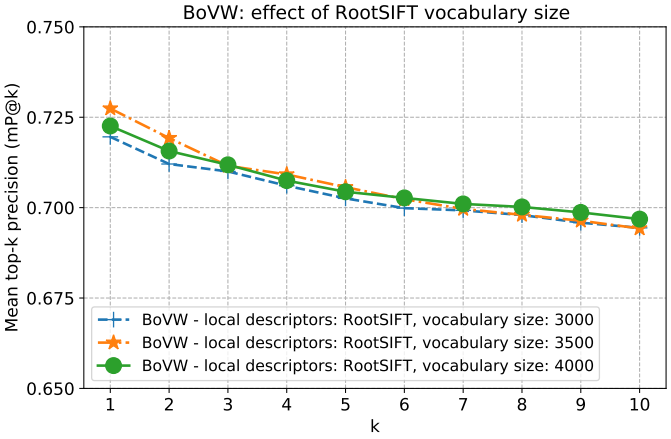

Fig. 3: Performance comparison for BoVW vocabulary sizes: For lower k, $n_{words} = 3500$ had the best performance. For higher k, $n_{words} = 4000$ was the best.

performance was the best for $size_{cell} = 64$, $size_{block} = 128$, and $stride_{block} = 64$ (overlapping blocks). The comparison is in figure 4.

After that, the effect of changing n_{bins} was tested. Using 15 histogram bins performed the best. Details are in figure 5.

3) CNN retrieval: Performance of retrieval with CNN-extracted descriptors was compared for ResNet18, ResNet34, AlexNet, SqueezeNet and VGG16. The models are pretrained on ImageNet and require 3-channel images as input, so the grayscale channels were stacked before input. SqueezeNet with an 5×5 average pooling layer to reduce output dimensionality performed the best. The comparison is in figure 6.

4) Retrieval result re-ranking: Images retrieved with one method (eg. BoVW) can be re-ranked with another method (eg. HOG). This is done by first retrieving k_{prep} images with the first method, sorting the retrieved images according to the distance of descriptors computed with the second method, and finally returning the top-k results with the lowest distance to

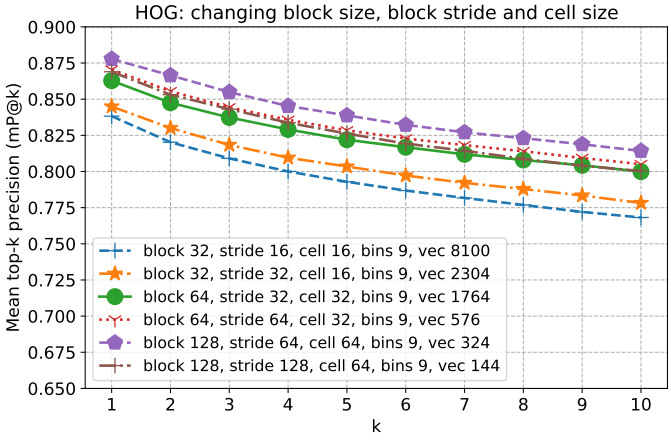

Fig. 4: Performance comparison for multiple combinations of HOG parameters: block size, block stride and cell size. The number of bins was fixed (9). HOG descriptor dimensionality depends on the aforementioned parameters and was also stated in the legend (vec). The performance was the best for $size_{cell} = 64$, $size_{block} = 128$, and $stride_{block} = 64$ (overlapping blocks).

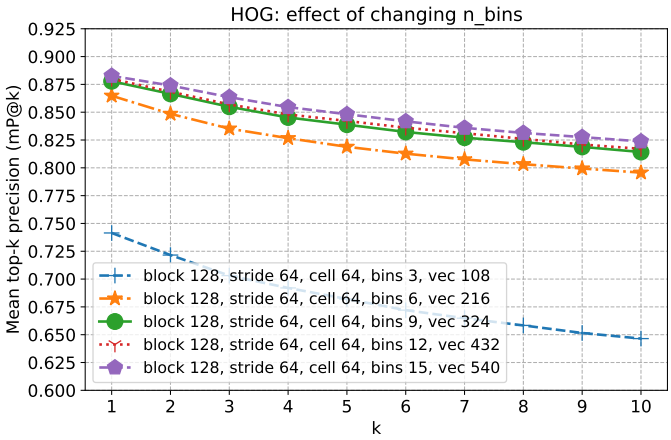

Fig. 5: Performance comparison for different histogram sizes: $n_{bins} = 15$ performed the best.

the query image descriptor.

Result re-ranking was tested for various combinations of the retrieval method, the re-ranking method and k_{prep} values, which have an influence on the final performance. The results can be seen in figure 7 and 8. In most cases, re-ranking improved the performance. The performance improvement was the largest for images retrieved with BoVW. Only CNN retrival was not improved with re-ranking.

IX. Results on the test dataset

For the final evaluation on the test dataset, the configuration with the best performance on the validation set was chosen for each retrieval method. The best re-ranking configuration was also included. The following configurations were chosen:

- BoVW: $n_{words} = 4000$, RootSIFT local features

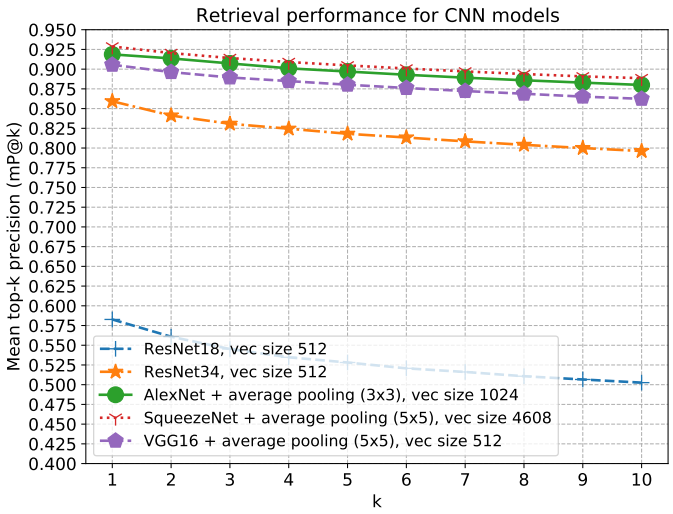

Fig. 6: Performance comparison for retrieval with CNN-extracted image descriptors: output of the final convolutional (or pooling) layer is used as the descriptor. The performance was the best for SqueezeNet with an 5×5 average pooling layer on top.

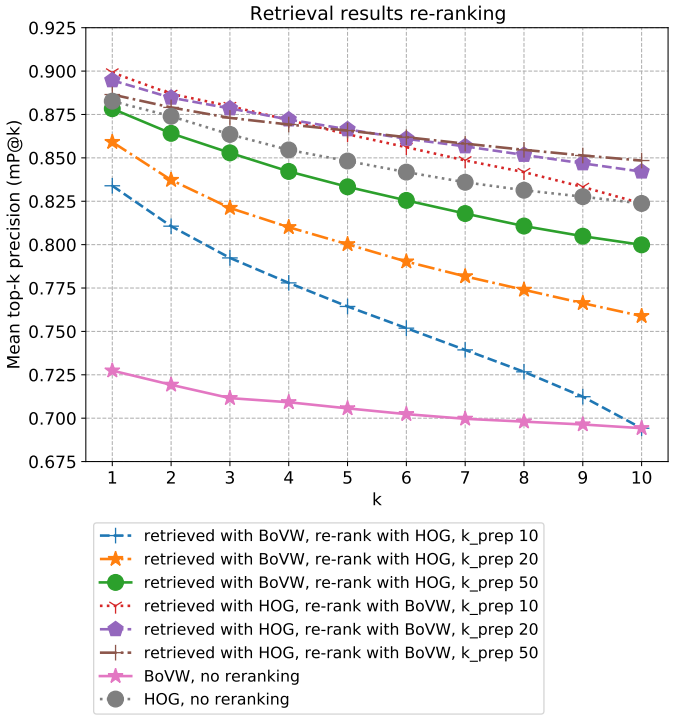

Fig. 7: Performance comparison for result re-ranking with different k_{prep} values. Re-ranking with HOG causes a significant performance increase for images retrieved with BoVW. Higher k_{prep} values work better. Re-ranking with BoVW after retrieving with HOG also improves results in comparison to HOG without reranking.

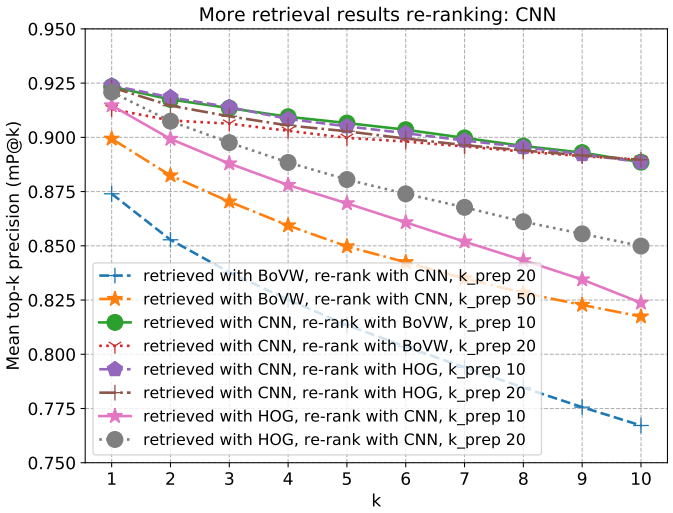

Fig. 8: Re-ranking with CNN as the retrieval method or as the re-ranking method. CNN re-ranking images retrieved with BoVW or HOG increases the performance. Re-ranking with BoVW or HOG after retrieving with CNNs does not outperform CNN retrieval without re-ranking.

Retrieval method	mAP (an. region)	mAP (modality)	mAP (both)
BoVW	0.7097	0.8576	0.7022
HOG	0.8249	0.9121	0.8146
CNN	0.8845	**0.9654**	0.8821
CNN + re-ranking	**0.8855**	0.9635	**0.8835**

TABLE III: Performance comparison on the test set (mAP)

- HOG: $size_{cell} = 64$, $size_{block} = 128$, $stride_{block} = 64$ (overlapping blocks), $n_{bins} = 15$
- CNN: SqueezeNet + 5×5 average pooling for extracting the image descriptors
- Re-ranking: Retrieval with CNN, re-ranking with HOG, $k_{prep} = 10$

The results for mean top-k precison ($k = 1, ..., 10$) can be found in figure 9. Like on the validation set, CNN retrieval performed the best, followed by CNN retrieval with HOG reranking, HOG and finally BoVW.

Mean top-k precison (mP) scores for multiple k values can be summarized by calculating the mean-Average-Precison (mAP). mAP scores (for $k = 1, ..., 10$) were calculated by considering the anatomical region, the imaging modality and both for determining the relevance of the retrieved images. The results are in table III. For imaging modality CNN retrieval performed the best (0.9654). For the anatomical region and for both, CNN with HOG re-ranking was the best (0.8855 and 0.8835). Examples of retrieved images for each anatomical region can be found in appendix A.

The time required in the offline phase for each method is in table IV. The time necessary to retrieve images with each method is in table V. HOG retrieval requires the least amount of time. The time was measured on a AMD Ryzen 5 2600 @ 3.400GHz CPU with 16GB RAM.

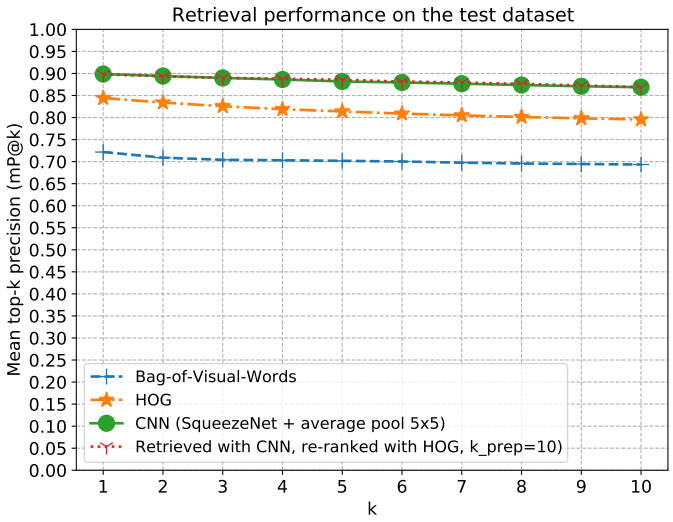

Fig. 9: Performance comparison on the test dataset: mean top-k precison for $k = 1, ..., 10$. Retrieved images were considered relevant if both the anatomical region and the imaging modality of the query image and the retrieved image were the same. For top-1 retrieval, SqueezeNet performed the best. For higher k values, reranking with HOG was slightly better.

Method	Offline phase duration (minutes)
BoVW	141 min total (SIFT - 6, vocab - 121, descriptors - 14)
HOG	1 min (extract descriptors)
CNN	22 min (extract descriptors)

TABLE IV: Time required for the offline phase for each retrieval method

Method	Retrieval time (ms)
BoVW	334ms
HOG	53ms
CNN	339ms
CNN + HOG re-ranking	341ms

TABLE V: Mean retrieval time of 8000 query images

X. Conclusion

In this study, a Content-based image retrieval system for radiology images of various anatomical regions and imaging modalities was designed and implemented. Three retrieval methods were evaluated: BoVW, retrieval with HOG descriptors and retrieval with CNN-extracted descriptors.

The configuration for each method was chosen empirically by comparing results on a validation image set. BoVW was tested with SIFT and RootSIFT local features. The use of RootSIFT descriptors improved the results. For retrieval with CNN descriptors, five architectures pretrained on the ImageNet dataset were evaluated. ShuffleNet provided the best performance. Retrieval with CNNs outperformed both BoVW and HOG retrieval. The CNN models were not additionaly trained. Fine-tuning the convolutional networks for classifying the anatomical region or modality of radiology images could potentially further improve the retrieval results. Aditional architectures could be tested as well.

Retrieving images with one method and then re-ranking them according to another method was also tested. Re-ranking improved the results in almost all cases. BoVW retrieval showed the most significant performance increase after re-ranking. Out of all methods evaluated on the test set, CNN retrieval (ShuffleNet with 5×5 average pooling) with HOG re-ranking performed the best.

In the future, more retrieval methods could be tested. Some examples are BoVW alternatives for aggregating local features into fixed-length vectors such as VLAD [26] and Fisher Vectors [27], or additional global descriptors, for example GIST. BoVW could also be tested with other local features, such as SURF.

References

[1] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta and R. Jain, "Content-based image retrieval at the end of the early years," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, 2000.

[2] Li, Zhongyu, Zhang, Xiaofan, Müller, Henning, Zhang, Shaoting. "Large-scale Retrieval for Medical Image Analytics: A Comprehensive Review", Medical Image Analysis, vol 43., 2018.

[3] Lowe, D. "Distinctive Image Features from Scale-Invariant Keypoints." International Journal of Computer Vision 60, 2004.

[4] Bay, H., Tuytelaars, T., Van Gool, L., "SURF: Speeded up robust features", Computer Vision-ECCV, 2006.

[5] Ojala, T., Pietikäinen, M., Harwood, D. "A comparative study of texture measures with classification based on featured distributions". Pattern Recognition, 29, 1996.

[6] Sivic, J., Zisserman A., "Video Google: a text retrieval approach to object matching in videos." Proceedings Ninth IEEE International Conference on Computer Vision, 2003.

[7] Oliva, A. and A. Torralba. "Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope." International Journal of Computer Vision 42, 2001.

[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2005.

[9] Wang S., Kim M., Wu G., Shen D. "Scalable High Performance Image Registration Framework by Unsupervised Deep Feature Representations Learning", Deep Learning for Medical Image Analysis, 2017.

[10] D. Nister and H. Stewenius, "Scalable Recognition with a Vocabulary Tree," IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2006

[11] B. Kulis and K. Grauman, "Kernelized locality-sensitive hashing for scalable image search," IEEE 12th International Conference on Computer Vision, 2009

[12] Y. Gong and S. Lazebnik, "Iterative quantization: A procrustean approach to learning binary codes," CVPR 2011

[13] Tian Guangjian, Fu Hong, Feng David Dagan Feng, "Automatic medical image categorization and annotation using LBP and MPEG-7 edge histograms", 2008.

[14] Foncubierta, A., García Seco de Herrera A., Müller, H., "Medical Image Retrieval using Bag of Meaningful Visual Words: Unsupervised visual vocabulary pruning with PLSA", 2013.

[15] Akgül Ceyhun, Rubin Daniel, Napel Sandy, Beaulieu Christopher, Greenspan Hayit, Acar, Burak. "Content-Based Image Retrieval in Radiology: Current Status and Future Directions", Journal of digital imaging : the official journal of the Society for Computer Applications in Radiology. 24., 2011.

[16] Zhu Shujin, Tizhoosh Hamid, "Radon Features and Barcodes for Medical Image Retrieval via SVM", 2016.

[17] Mustapha Aouache, Hussain Aini, Samad Salina, Zulkifley Mohd Asyraf, W Zaki W Mimi Diyana,"Design and development of a content-based medical image retrieval system for spine vertebrae irregularity", Biomedical engineering online, 2015.

[18] Rey-Otero, Ives and M. Delbracio. “Anatomy of the SIFT Method.” Image Process. Line 4 : 370-396., 2014.
[19] Bag of Visual Words for Finding Similar Images, Cyrill Stachniss, 2020.
[20] Razavian, A. et al. “Visual Instance Retrieval with Deep Convolutional Networks.” CoRR, 2015.
[21] He, Kaiming et al. “Deep Residual Learning for Image Recognition.” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
[22] Krizhevsky, A. et al. “ImageNet classification with deep convolutional neural networks.” Communications of the ACM 60, 2012.
[23] Iandola, Forrest N. et al. “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and ¡1MB model size.” ArXiv abs/1602.07360, 2016.
[24] Simonyan, K. and Andrew Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition.” CoRR abs/1409.1556, 2015.
[25] Arandjelović, R. and Andrew Zisserman. “Three things everyone should know to improve object retrieval.” 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012.
[26] Jégou, H. et al. “Aggregating local descriptors into a compact image representation.” 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010.
[27] Perronnin, F. et al. “Large-scale image retrieval with compressed Fisher vectors.” 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010.

APPENDIX

A. Appendix A: Example retrieved images for each anatomical region

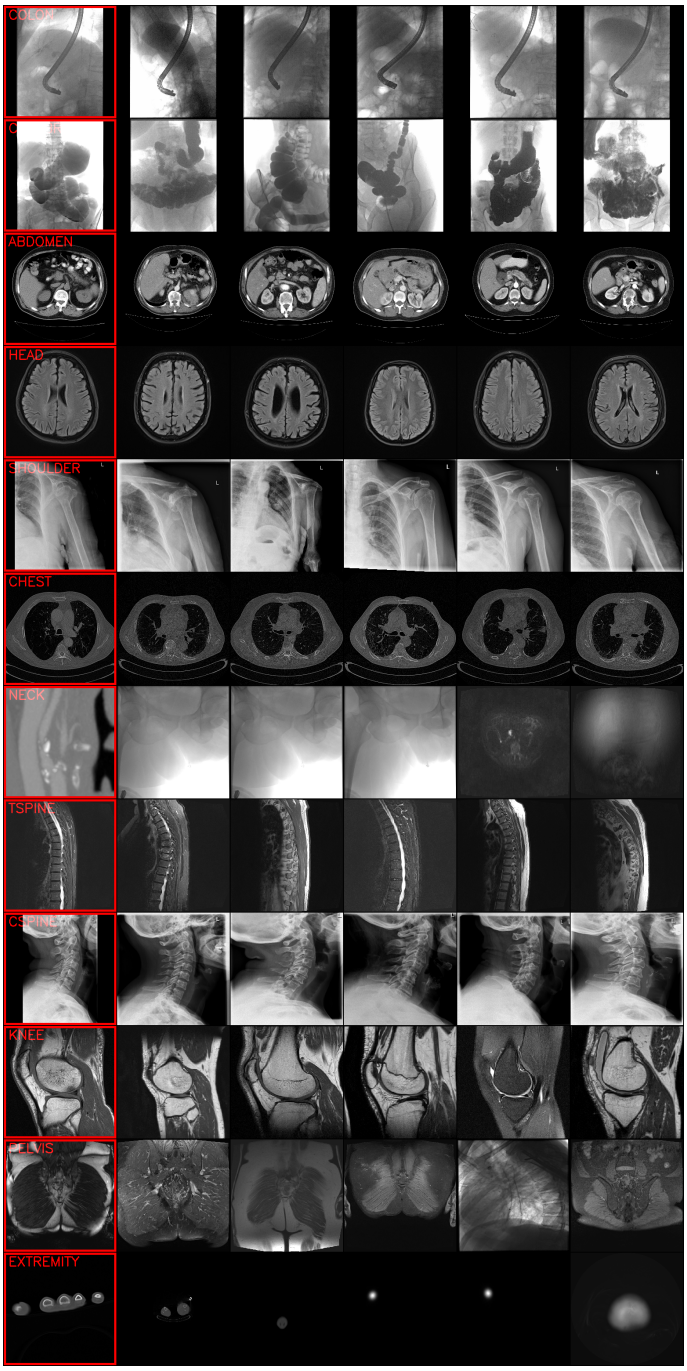

Fig. 10: Retrieved images for the first 12 anatomical regions

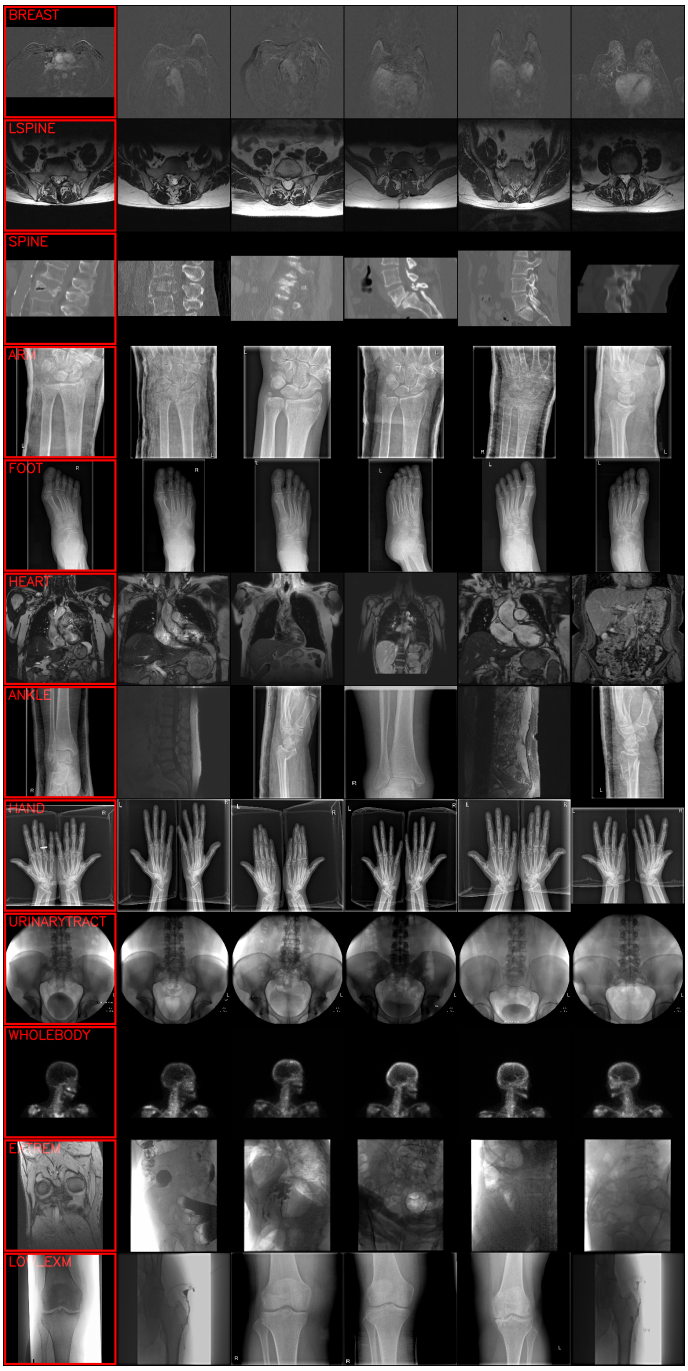

Fig. 11: Retrieved images for the last 12 anatomical regions

One random query image was chosen for each anatomical region in the dataset. The leftmost images with the red frame are the query images. The remaining 5 images in each row are the top 5 most similar images retrieved with the best performing retrieval method (SqueezeNet + 5×5 average pooling).