Grant Agreement Number: 257528

KHRESMOI

www.khresmoi.eu

D2.7: Report on results of the second evaluation phase

Deliverable number	*D2.7*
Dissemination level	*Public*
Delivery data	*30.6.2014*
Status	*Final*
Authors	*René Donner, Johannes Hofmanninger, Thomas Schlegl, Ljiljana Dolamic, Célia Boyer, Dimitris Markonis, Henning Müller, Georg Langs*

This project is supported by the European Commission under the Information and Communication Technologies (ICT) Theme of the 7th Framework Programme for Research and Technological Development.

Executive Summary

The retrieval performed in the prototypes created during the KHRESMOI project is in part based on image information. In this deliverable we report the quantitative evaluation of individual core components of image based retrieval. These components perform localization and identification of anatomical structures, disease relevant classification of regions in image data, content based image retrieval, or text based image retrieval, and finally relevance feedback. Integrated in the prototypes the components serve tasks raning from retrieval of three-dimensional radiological imaging data to search for images in online resources.

For each of these components, we briefly outline the method, and then focus on the quantitative evaluation of the component. For the clinical radiology prototype image analysis backend, we evaluate the three core components: (1) localization of anatomical structures, (2) the classification of tissue types, and (3) the retrieval of similar tissue types based on a user marked region of interest. For the frontend we evaluate relevance feedback, and for the professional- and general public prototypes we evaluate the text based image retrieval.

All components perform their role satisfactory in the prototypes, and contain substantial methodological novelty described in previous deliverables.

Table of Contents

List of Figures

Notation

$\mathbf{A}$	a 3×4 affinity matrix
$\mathbf{B}_j$	3D atlas volume
$\mathbf{I}_i$	Image or volume with index i. If it is 2D or 3D data will be clear from the context.
$\mathbf{I}_i \in \mathbb{R}^2$	2D data such as images.
$\mathbf{I}_i \in \mathbb{R}^3$	3D data such as volumes.
$\mathbf{I}_i(x)$	Value of image of volume at position x
$\mathbf{f}(x)$	Feature (vector) extracted at position x
$\mathbf{d}(\mathbf{I})$	Image descriptor (vector) for an entire image/volume
$p_{i,s}$	Identifier of a supervoxel s in image $\mathbf{I}_i$
$l(p_{i,s})$	Hidden labeling of a supervoxel
$\mathcal{T}_i$	Weak labels of an image

Abbreviations

ANN	Approximate Nearest Neighbour
API	Application Programming Interface
BoC	Bags of Colors
BoVW	Bags of Visual Words
CBIR	Content–based image retrieval
CEDD	Color and Edge Directivity Descriptor
CNN	Convolutional Neural Network
CouchDB	Cluster of unreliable commodity hardware DataBase
CRBM	Convolutional Restricted Boltzmann Machine
CSV	Comma–Separated Values
CT	Computed Tomography
DICOM	Digital Imaging and Communications in Medicine
DoG	Difference of Gaussians
E2LSH	Euclidean Locally Sensitive Hashing
EMEA	European Medicines Agency
Europarl	Europarl: A Parallel Corpus for Statistical Machine Translation
ezDL	easy Digital Libraries
FCTH	Fuzzy Color and Texture Histogram
GLCM	Gray Level Coocurrence Matrix
GUI	Graphical User Interface
HES-SO	University of Applied sciences, Western Switzerland
HoG	Histogram of Gradients
HTTP	Hypertext Transfer Protocol
ImageCLEF	Image Retrieval in the Cross Language Evaluation Forum
JSON	JavaScript Object Notation
K4E	Khresmoi for Everyone
LBP	Local Binary Patterns
LTRC	Lung Tissue Research Consortium
MAP	Mean Average Precision
MeSH	Medical Subject Headings
MRI	Magnetic Resonance (Imaging)
MUW	Medical University of Vienna
MySQL	My Structured Query Language
PACS	Picture archiving and communication system
ParaDISE	Parallel Distributed Image Search Engine
REST	Representational state transfer
ROI	Region of interest
SCA	Service Component Architecture
SIFT	Scale-Invariant Feature Transform
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TF–IDF	Term Frequency – Inverse Document Frequency
URL	Uniform Resource Locator

1 Introduction

The objective of workpackage 2 in the KHREMSOI project is to develop, evaluate, and integrate methods for image content based information retrieval. Image analysis methods are part of all prototypes developed in the project. In the prototype aimed at the general public, image retrieval methodology is driving the search for images relevant to the user query. In the prototype for expert users, retrieval uses image information itself to identify those images most similar to a query images among a large set of 2D imaging data. Finally, in the clinical radiology prototype, image analysis plays a central role, in retrieving similar cases, relevant to the diagnosis of a certain anomaly.

Evaluation results have been reported before in this project. In deliverable D2.3 we have evaluated feature extraction, matching and classification methods. In deliverable D2.4, we have detailed the algorithm for the localization of anatomical structures based on image patches, and reported its localization accuracy. We also presented methods for the mapping of large amounts of heterogeneous medical imaging data to a common reference space, in order to search across populations while focusing on specific anatomical locations. In deliverable D2.5 we explained the overall modular framework in which anatomical structure localization and identification is embedded, and in deliverable D2.6 we described the entire prototype framework.

In this deliverable we report the latest evaluation results of specific central components, that have either been adapted during the project, or for which new solutions became necessary. Specifically, we report quantitative experimental results on the following components:

1. Anatomical structure localization based on stochastic affine registration of fragments
2. Classification of anomalies based on deep-learning methodology
3. Retrieval of similar clinical radiology cases based on a query case
4. Relevance feedback
5. Text based image retrieval

For each of the components, we outline the methodological approach, describe the evaluation procedure, and then report and discuss the experimental results.

2 Localization

The localization of anatomical structures in a given image volume, and their mapping to an anatomical atlas, are of vital importance to the performance of the Khresmoi retrieval system. In the clinical radiology prototype, the user selects a region of interest (ROI) in a query volume, to trigger retrieval. First, the systems needs to know in which anatomical region this ROI is located to use the correct index of existing cases for retrieval. In the following we outline how this can be done in a general fashion for the radiological 3D data encoutered in clinical practice.

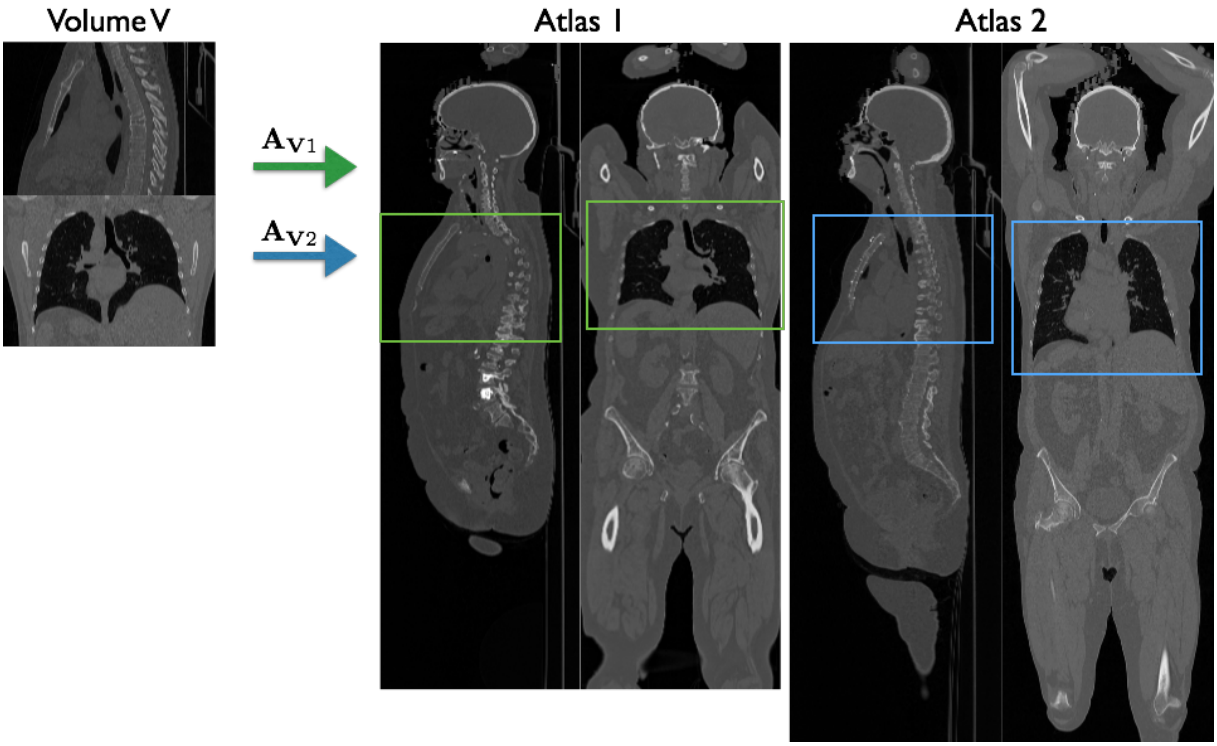


Figure 1: Example volume - a thorax CT, and two of the 3D fullbody atlases to which which the affine matching was performed

2.1 Method overview

We have developed a method that obtains a detailed labeling of the image volume with only the volume and an atlas provided as input data - no additional data or human intervention is required. The method is split into two parts: First, an affine matching of a query volume (e.g. a thorax CT) to the corresponding region in each atlas (fullbody CTs) is performed. Subsequently, a non-rigid registration is performed between the volume and the most similar atlas, yielding a detailed voxel-wise labling of the query volume, as depicted in Fig. 1.

2.1.1 Affine stochastic registration of volumes to atlases

The affine matching between two volumes is described by an affine matrix $\mathbf{A}$, encoding translation, rotation, scaling and shear. Given an atlas $\mathbf{B}_j \in 1,\ldots,J$, in our case one of a set of J fullbody CTs, and a volume $\mathbf{I}_i \in 1,\ldots,N$, we are interested in finding a matching transformation $\mathbf{A}_{ij}$ such that the difference $c(\mathbf{A}_{ij}) = \| \mathbf{I}_i - \mathbf{A}_{ij} \circ \mathbf{B}_j \|$ is minimal.

This matching objective is non-convex and is characterised by many local minima. For this reason a robust, stochastic approach to computing c is important. We chose Differential Evolution (DE) stochastic optimization [27] due to its fast convergence properties and simple algorithmic structure.

Differential Evolution stochastic optimization Differential Evolution (DE) is a member of the familiy of stochastic optimization aproaches popularized by the group of algorithms called

genetic algorithms. The basic idea is to use a set of candidate solutions, called the population $\mathcal{P}$, which are modified over the course of several iterations until the algorithm converges. The manner in which the individuals of the populatioin are changed is the main difference between genetic algorithms and Differential Evolution.

As the name implies, the main source of change in DE comes from the difference between the members of the population. Initially, each individual in the population is initiazed by drawing random samples from a uniform distribution matched to the problem domain. Then, in each iteration, for each individual $\mathbf{x}_m$, two other members $\mathbf{x}_n$ and $\mathbf{x}_o$ are randomly selected from $\mathcal{P}$. A hypothesis $\mathbf{x}'_m$ is constructed as

$$\mathbf{x}'_m = \mathbf{x}_m + \alpha(\mathbf{x}_o - \mathbf{x}_n), \tag{1}$$

with α generally set to 0.85. If the cost $c(\mathbf{x}'_m)$ is smaller than $c(\mathbf{x}_m)$ $\mathbf{x}_m$ is replaced in the population by $\mathbf{x}'_m$. This is repeated until the convergence of the best individual (with the lowest cost) or until a maximum number of iterations is reached. Despite this extremely simple update rule, DE manages to converge to a (potentially global) optimum even in challenging settings.

In the case of the affine matching the individuals are affine matrices. Initialized randomly, within sensible ranges for translation (spanning the entire atlas), rotation, scaling and shear, the individuals of the population quickly converge to the position in the atlas $\mathbf{B}_j$ which represents the anatomical region depicted in $\mathbf{I}_i$. Further iterations then refine the remaining parameters of the affine mapping, resulting in the best match $\mathbf{A}^*_{ij}$ with the associated cost $c(\mathbf{A}_{ij}*) = c(\mathbf{x}_m*)$ of the best individual. This optimization procedure is repeated for each atlas $\mathbf{B}_j$. Finally, the best overall match over all j, i.e. the affine mapping $\mathbf{A}^*$ represents the final result of the affine matching stage.

2.1.2 Non-rigid registration

The affine matrix $\mathbf{A}^*$ describes the single block from all of the atlases which is most similar to the volume $\mathbf{I}$. To further refine the correspondances between the atlas and the volume, we perform a non-rigid registration step. The inputs to this non-rigid registration are the volume $\mathbf{I}$ on the one hand, and the block specified by $\mathbf{A}^*$ from the corresponding atlas $\mathbf{B}$ on the other. Two state of the art registration frameworks were evaluated, namely *NiftyReg* and *Ezys*. The result of either of these methods are fine-grained, voxel by voxel mappings between the volume and the atlas. This allows to propagate the anatomical labels known for the atlas (annotations provided by medical experts) to the volume.

2.2 Evaluation experiments

The evaluation of the presented approach was performed using a set of annotated full body CTs as atlases, with volumes $\mathbf{I}$ cut out from these atlases. This allows to perform controlled numerical experiments.

Data The data set used for evaluation was a set of fullbody CTs provided by the Visceral project[1]. The data consisted of 7 standard CTs and 7 contrast enhanced CTs, with a field of

[1] http://www.visceral.eu/

view spanning from the lower extremities to the head. The resultion of the volumes was 580 by 580 inplane, with around 400 slices each. The volumes were downsampled to an isotropic resultion of 2mm. For each of the volumes a labeling annotation was performed by medical experts, specifying the major organs as well as selected smaller structures.

Set-up Two anatomical regions were chosen for the evaluation of the registration accuracy, namely the thorax and the abdominal region. All exeriments were run in a leave-one-out setup, i.e. from one of the full body volumes the thoratic region and the abdominal region were cut out, and all other fullbody volumes were used as atlases for this run, trying to match the thorax on the on hand and the abdomen on the other hand. We thus obtain registration results for 14 volumes for each of the two regions.

Tasks The task measured was the registration of the volume to the atlases, by finding first the most similar atlas using the affine stochastig matching approach, and then non-rigidly registering the volume to a block cut out from this atlas. This task is performed for both anatomical regions (thorax, abdomen) as well as two non-rigid registration frameworks (NiftyReg[2] and Ezys[3]).

Measures of quality As measure for estimating the performance of the registration we employ the DICE coefficient s, which allows to quantify the overlap of two regions. It is defined, when interpreting the regions as sets of voxels, as

$$s(X,Y) = \frac{2|X \cap Y|}{|X| + |Y|}, \tag{2}$$

with a maximum overlap for $s = 1$ and a minimum of no overlap at $s = 0$. Each of the volumes $\mathbf{I}$ is extracted from one of the fullbody volumes for which detailed annotations are available, and the block of these annoations $\mathbf{I}_{\text{groundtruth}}$ corresponding to $\mathbf{I}$ is used as the groundtruth for an optimal labeling. Thus the overlap for one result for one region $\mathcal{R}$ is computed as $s(\mathbf{I}_{\text{labeling}} = \mathcal{R}, \mathbf{I}_{\text{groundtruth}} = \mathcal{R})$.

2.3 Results

The results are depicted in Fig. 2, showing the whisker plot for the each of the anatomical regions and the two registration approaches (NiftiyReg and Ezys). As can be expected, both algorithms perform much better for large organs which high-contrast boundaries like the lung. While NiftyReg shows better performance for these cases, both methods exhibit difficulties at finding smaller and more diffuse structures, like the gallbladder.

Reiterating that in our case the task at hand is not a pixel-perfect segmentation but a localization of the organ in question, we find that the proposed approach works well enough to obtain these localization masks, with the overall median of the median DICE score per organ at 0.53.

[2] http://www.nitrc.org/projects/niftyreg/
[3] http://sourceforge.net/projects/ezys/

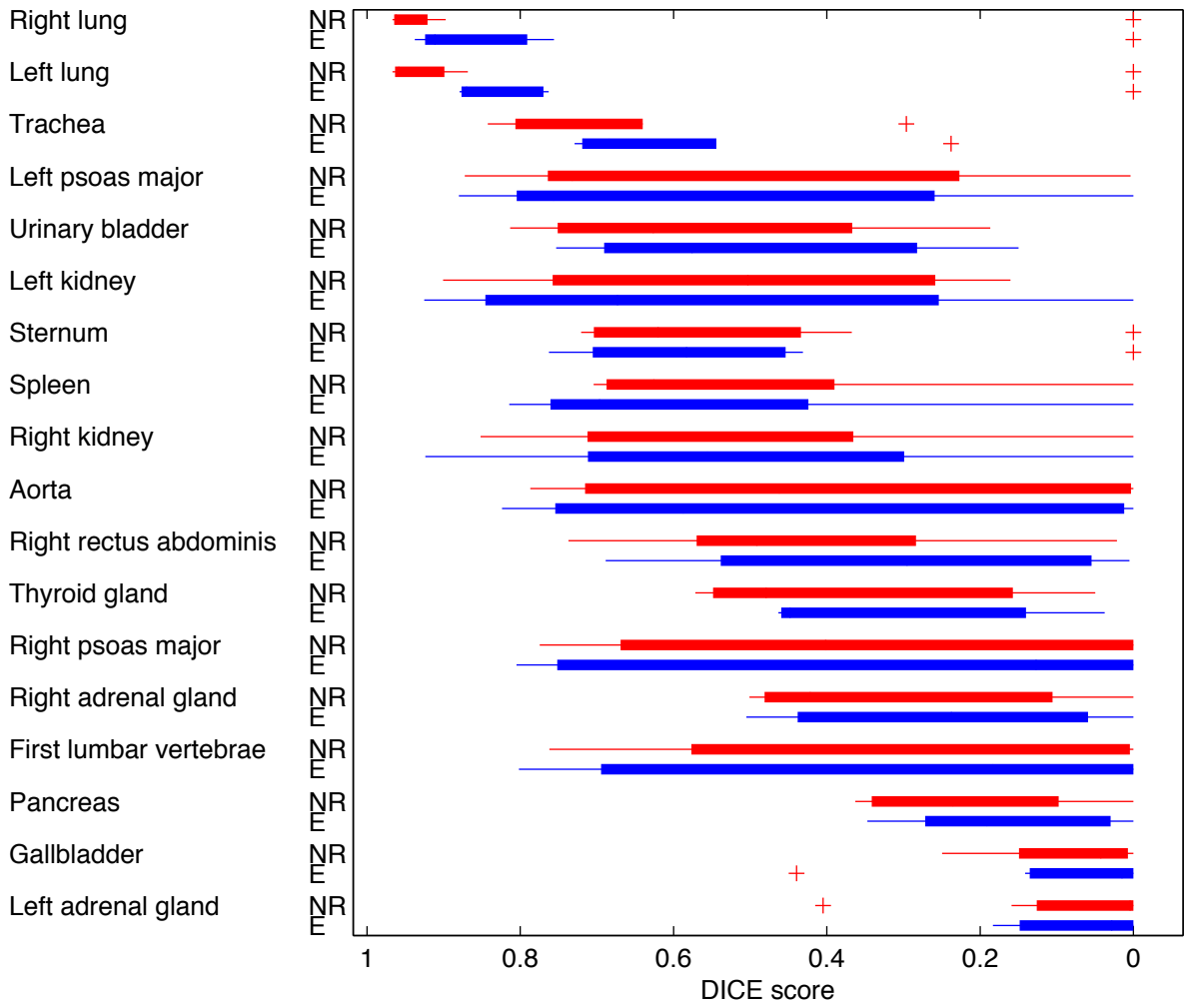


Figure 2: Evaluation results of the registration for the NiftiReg (NR, red) and Ezys (E, blue) registration algorithms as DICE coefficients.

2.4 Discussion

The labeling of a volume can be performed as soon as the volume is available to the systen. As such it can be labeled off-line, before any query makes use of the volume and its labeling. The computing time for a labelling is in the order of a few minutes depending of the specific hardware available. The primary goal of the outlines appraoch is to provide a general localization framework without the need for organ specific training and parameter optimization. The labeling accuracy is sufficent for a coarse segmentation of the imaging data, in order to trigger region specific indices for retrieval. Organ specific approaches can further improve accuracy of segmentation, or landmark localization (see previous deliverable D2.4 and publications e.g., [6]).

3 Classification

Disease diagnosis, treatment monitoring and treatment planning are important fields of application for computational medical image analysis methods such as image classification or object detection algorithms. In the medical domain, image classification is the computational task of finding an appropriate discrimination between different pathological tissues and between pathological and healthy tissues. Image classification algorithms can be built by learning a statistical model of variabilities of image features and their link to a set of classes (e.g., different disease patterns).

Here we evaluate and approach that learns both feature extractors and classification model based on labeled- and un-labeled data.

Image features describe the image information in relation to a given classification or object detection task. Raw image pixels are the simplest but least abstract variant of image features. Medical imaging devices produce large amounts of high-resolution data with high variability of visual patterns. Abstract feature representations enable the model to capture these high variabilities of visual patterns. Hand-engineered features require domain-specific expert knowledge and are neither offhand applicable for different classification tasks nor for images acquired with different imaging modalities. Deep Learning models are able to learn abstract representations of image patterns (features) in medical images in an unsupervised way, i.e. for the feature learning task no annotations are needed and thus large amounts of unlabeled domain-specific data can be used.

In contrast, learning representations tackling the classification task is done in a supervised way. Therefore large amounts of annotated training data is needed. The more annotated training data is available, the more natural variability in the data can be represented. Annotation of medical images is time consuming and does not scale well to large data. Thus often sufficient amounts of training data is not available.

It has been shown [7] that the performance of Deep Learning algorithms on supervised tasks can be improved via unsupervised pre-training which solely utilizes unlabeled data. Thus good classification results can be achieved even if a reduced amount of annotated data is available. Unlabeled medical image data is usually available in great quantities from a Picture Archiving and Communication System (PACS).

3.1 Method overview

Typically computational models, as for instance Support Vector Machines (SVMs), are learned based on relatively small data sets. Deep Learning algorithms, as for instance Autoencoders [12], Deep Boltzmann Machines [24] or Convolutional Neural Networks (CNNs) [8], are well suited for processing dataset with a very large sample size. Autoencoders and Deep Boltzmann Machines can be efficiently trained on relatively small image sizes (e.g. images from the MNIST dataset have a resolution of 28x28 pixels) [22]. In contrast, CNNs can also process larger image sizes (e.g., 256x256 or 512x512), which is characteristic for medical imaging.

CNNs are hierarchically structured by one ore more blocks stacked on top of each other, each consisting of a convolution layer followed by a max-pooling layer. In contrast to Autoencoders and Deep Boltzmann Machines, CNNs preserve the 2D structure of the input images. Convolutional Restricted Boltzmann Machines (CRBMs) are the central building blocks

of CNNs. They implement the convolution layers. CRBMs are undirected graphical models which are built-up by two layers, one visual layer and one hidden layer. The visual layer consists of a set of real-valued input units v of dimension $C \times C$ and the hidden layer consists of $\gamma = \{1, ..., \Gamma\}$ groups of hidden units h each of dimension $B \times B$. The activations of the hidden units are calculated by convoluting the input units with γ filter kernels. In the training phase the weights of the convolution filter kernels are learned via block Gibbs sampling. The positive phase computes the activations of the hidden units, given the visible units. The conditional distributions of the hidden units h in group γ is defined by

$$P(h_\gamma = 1|v) = \sigma\left((\tilde{W}_\gamma * v) + b\right) \tag{3}$$

The second phase, the negative phase, computes the activations of the visible units v, given the hidden units of all groups

$$P(v|h) = \sum_\gamma \left(W_\gamma * h_\gamma\right) + c. \tag{4}$$

The additive constants b and c are bias terms of the hidden and the visible layer respectively.

The $*$ operator denotes the convolution operation. The positive phase utilizes a vertically and horizontally flipped version of the filter matrix W. The flipped version of the filter matrix is denoted by $\tilde{W}$. Additionally, the positive phase includes a non-linear transformation which can be implemented via the sigmod function

$$\sigma(q) = \frac{1}{(1 + exp(-q))}. \tag{5}$$

The activations of the hidden units are pooled by the succeeding max-pooling layer, which introduces some kind of translation invariance. On top of the first CRBM a second CRBM can be stacked and so on. The activations of the visual units of the second CRBM are the resulting activations of the max-pooling layer of the first CRBM. The more CRBMs are stacked on top of each other the more abstract feature extractors can be realized. The first CRBM learns Gabor-like edge detectors from input image patches. CRBMs at higher layers are able to learn abstract representations of healthy or different types of pathological tissues.

The CRBM on the top of the stack is followed by one or more fully connected layer. The last layer is a classification layer, which for example implements softmax classification.

3.2 Evaluation experiments

Evaluation was performed on 380 clinical high-resolution lung CT scans. This dataset contains a subset from the Lung Tissue Research Consortium (LTRC) dataset [13]. Training and classification were exclusively done on 2D image patches $\mathbf{I}_i^p \in \mathbb{R}^2$ with patch width $m \times m$, extracted at position i from axial slices with image sizes of $n \times n$ of the image volumes $\mathbf{I}_i \in \mathbb{R}^3$, where $m << n$. In addition to each image volume the LTRC dataset also provides a corresponding annotation volume $\mathbf{A}_i \in \mathbb{R}^3$ with pixelwise annotations of anomalous lung tissue (ground glass opacity, reticular interstitial pattern, honeycombing, emphysema) and normal lung tissue. The class label of image patch $\mathbf{I}_i^p$ at position i was calculated as the modal value of the class labels of all pixels in the annotation patch $\mathbf{A}_i^p$ centered at position i.

As has been shown by Erhan et al. [7], unsupervised pre-training helps to improve the performance of Deep Learning algorithms. To verify this for our domain the training of the deep model was done in two different ways:

1. Supervised training of the whole model without a preceding unsupervised pre-training phase.

2. Layer-wise unsupervised pre-training of the single CRBMs and subsequent supervised fine-tuning of the whole model.

We evaluated to what extend unsupervised pre-training of a Convolutional Neural Network can improve the classification results on real clinical lung data.

The pre-training step learns the parameters for the feature extractors. In this step the bias terms and the weights of the filter matrices of the single CRBMs are learned. Pre-training is done in an unsupervised way by minimizing the difference between input data and the representation of the data by the network. Thus large amounts of unlabeled data can be used. In the main phase of the training the parameters of the CRBMs are fine-tuned and the parameters of the fully-connected layers and the classification layer are learned as well. Fine-tuning is done in a supervised way by minimizing the misclassification error on labeled image patches. Thus, this training step was done in our case on the annotated dataset of medical images.

Image patches $\mathbf{I}_i^p$ were used for unsupervised pre-training. Supervised fine-tuning and evaluation of the classification accuracy was performed on the image patches $\mathbf{I}_i^p$ with their corresponding patches of pixelwise annotations $\mathbf{A}_i^p$. Image patches were classified into five distinct groups, i.e. every image patch got the label of healthy lung tissue or one of the four anomalous lung tissue labels provided by the LTRC dataset.

The pixel based misclassification error e was used as measure of quality of the fine-tuned model. This measure can not be used for unsupervised pre-training, because for unsupervised training labels are not available by definition. We define the misclassification error as amount of misclassified pixels divided by all classified pixels. Thus, the measure of quality can be rewritten the following way

$$e = 1 - d, \tag{6}$$

where d is the Dice's coefficient. The Dice's coefficient is a similarity measure over two sets V and W defined by

$$d = \frac{2|V \cap W|}{|V| + |W|}, \tag{7}$$

where V is the set of true class labels and W is the set of predicted class labels. The evaluation was done on two diferrent CNN architectures. The first *simple CNN* consists of 2 convolution layers, 1 fully-connected layer and a softmax classification layer. The second is a slightly *deeper CNN* with 3 convolution layers, 2 fully-connected layers and a softmax classification layer. Each fully-connected layer in both architectures consists of 1000 neurons.

3.3 Results

Two training scenarios were compared. In the first scenario, both architectures were pre-trained in an unsupervised way over 500 epochs. Subsequently, supervised fine-tuning of the model

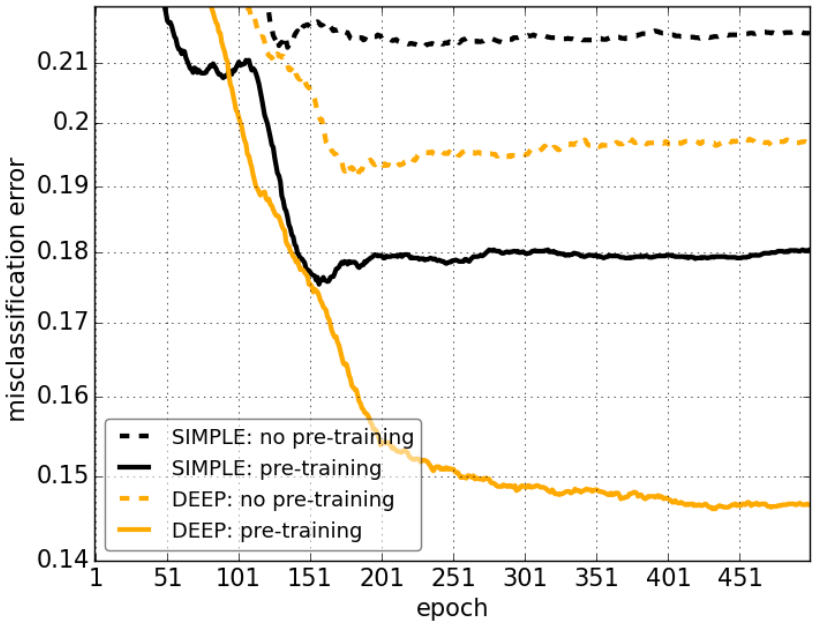


Figure 3: Misclassification error obtained by the fine-tuned CNN over training epochs.

Table 1: Comparison of the minimum misclassification error of the simple CNN and the deeper CNN.

	simple CNN	deeper CNN
no pre-trainnig	0.2127	0.1914
with pre-training	0.1754	**0.1460**
performance gain	0.0373	0.0454

parameters was performed. The number of tine-tuning epochs was set to 500. In the second scenario, solely fine-tuning of randomly initialized model parameters was performed over 500 epochs. We evaluated the effect of pre-training on the performance of the simple CNN and of the deeper CNN versus the performance of both architectures when no pre-trainnig was done.

The performance of both architectures is summarized in Table 1. Figure 3 shows the logarithmic misclassification error of both CNN architectures an both training scenarios as a function of fine-tuning epochs.

3.4 Discussion

The deeper CNN yields consistently better results in comparison with the simpler CNN. This is mainly because a larger number of convolutional layers allow to learn more abstract feature representations. Thus the network is able to capture latent discriminative information. Unsupervised pre-training improves the classification performance in both CNN architectures. Since the

unsupervised pre-training only adapts the filters of the feature extractors, this step helps primarily to learn domain specific feature extractors. In addition, we observe that pre-training leads to a steeper performance gain in the deeper setup. The additional pre-training of the deeper CNN reduces the misclassification error from 19.14% (at epoch 183) to 14.60% (at epoch 431), i.e. the performance gain is 4.54%. Training of the whole simple CNN without pre-training leads to a misclassification error of 21.27% (at epoch 134). Preceding unsupervised pre-training reduces the misclassification error by 3.73%. As unsupervised pre-training merely needs unlabeled data and this kind of data is commonly available in huge amounts, using deep learning methods in a clinical classification task should always include an unsupervised pre-training step.

4 Retrieval

The retrieval quality of CBIR-systems depends on all modules and parameters that can influence the ranking of the retrieved database entries. In order to judge the influence of a certain processing step on the retrieval result quality, a complete pipeline, starting from the definition of the information need to a result ranking metric has to be found and implemented in advance. This section describes the definition and extraction of an exemplary information need on real world data in order to evaluate different processing steps of a CBIR-system for medical imaging data. The evaluated CBIR-system relies on the extraction of separated regions like organs (e.g. Lung) and the oversegmentation of the images into supervoxels. All subsequent components, like feature extraction and indexing, deal with supervoxels as if they were small independent images.

However, at some point during retrieval, a ranking of entire images has to be generated on the basis of sorted lists of supervoxels has to be made. The following components in the indexing and retrieval pipeline were considered in our experiments:

- **Image Descriptors:** Different extraction methods and parameter variations.
- **Indexing Methods:** The influence of compression and search techniques compared to distance computation on the original vectors.
- **Result Ranking:** Comparison of different ways to rank complete medical images on the basis of retrieved segments (supervoxel) based on their distance to a set of queries.

Not considered in this evaluation are the segmentation of anatomical structures or the influence of oversegmentation algorithms and other preprocessing steps.

4.1 Problem Definition and Notation

In the following, a formal problem definition and description of the data is given.

For each image $\mathbf{I}_i$, there exists an oversegmentation of S_i segments called supervoxels where $p_{i,s}$ identifiers the supervoxel s in volume $\mathbf{I}_i$. The feature extraction and indexing pipeline operates on supervoxels rather than whole images. Let

$$\mathcal{M} = \{1,\ldots,M\} = \{p_{1,1},\ldots,p_{1,S_1},p_{2,1},\ldots,p_{2,S_2},p_{N,1},\ldots,p_{N,S_N}\} \tag{8}$$

be the set of all supervoxel identifiers in a database of N images. We assume, that each supervoxel represents a certain tissue property (e.g. 'healthy', 'groundglass', 'emphysema',...). Note, that this labeling is hidden and thus not given in real world data. However, this hidden labeling can be defined by a mapping function:

$$l : \mathcal{M} \rightarrow \{1, \ldots, T\} \tag{9}$$

In a real world dataset, there is a set of labels $\mathcal{T}_i = \{t_{i,1}, \ldots, t_{i,T_i}\} \subseteq \{1, \ldots, T\}$ given for each image, rather than a single label for each supervoxel. Thus, in contrast to the proper hidden labels for supervoxels, a weak labeling for images is given only:

$$\langle \{p_{i,1}, p_{i,2}, \ldots, p_{i,S_i}\}, \mathcal{T}_i \rangle \; \forall t_{i,j} \in \mathcal{T}_i : \exists p_{i,s} : l(p_{i,s}) = t_{i,j} \tag{10}$$

We are evaluating different similarity measures d which allows to judge the visual similarity of two supervoxels. We interpret d as a distance function, so that the similarity-correlation is modeled in an inverse way:

$$d(p_{i,s}, p_{j,b}) < d(p_{i,s}, p_{k,p}) \mid t_{i,s} = t_{j,b} \wedge t_{i,s} \neq t_{k,p} \tag{11}$$

Note that a similarity measure is a combination of image descriptor, metric and index (under assumption that the index implements an ANN search).

Furthermore, we also evaluate functions dv:

$$\forall j,k : dv(q, \mathbf{I}_j) < dv(q, \mathbf{I}_k) \mid (l(q) \in \mathcal{T}_j) \wedge (l(q) \notin \mathcal{T}_k). \tag{12}$$

Less formally, we are evaluating ranking functions, that allow to find images containing supervoxels with the same anomaly as a query supervoxel based on sorted lists of candidate supervoxels.

4.2 Methods overview

In the following we list the components and methods evaluated. In the following, the evaluated components of the indexing pipeline are recaptured concisely.

Following image descriptor methods applied on supervoxels are compared in this evaluation:

- **fernsBVWpca**: A bag-of-visual-words approach based on multi-scale three dimensional LBP features [2] and random vocabulary building by random ferns. Subsequent dimensionality reduction by PCA.
- **semanticProfile**: An image descriptor utilizing a weakly-supervised learning technique which is trained on volume labels extracted from radiological reports and fern based bag of visual word features (without PCA).
- **haar** Haar wavelets [4].
- **charalick**: Haralick features based on gray level cooccurence matrices [**?**].
- **patches**: Gray values extracted on a two dimensional patches.

- **saeembedding**

To retrieve a list of k-nearest-neighbors from the set of image descriptors in the database, three different lookup methods are compared:

- **euclid**: The calculation of the euclidean distance between the query vector and all vectors in the database
- **PQ**: The calculation of approximations of the euclidean distances using compact codes to increase retrieval speed and memory consumption by Product Quantization (PQ) and asymmetric distance computation [14].
- **PQSM**: A non-exhaustive version of the PQ-index, according to the Inverted File with Asymmentric Distance Computation (IVFADC) method proposed by Jegou et al. [14]. The used implementation however utilized a memory mapping functionality to read parts of the inverted file structure from secondary memory when required.

At retrieval time, the index returns lists of supervoxel-identifiers and their distances to the query. Different ways to rank images on the basis of these lists are compared. As a first step, the distances are scaled to a similarity measure by $sim(u) = e^{-u^2/(2\bar{u}^2)}$ where u is the (eventually approximated) euclidean distance between a query and a database supervoxel image descriptor and $\bar{u}$ is the mean of all distances retrieved. Based on this similarities, different ways to rank the images are implemented.

- **sumRanking1**: Ranking of volumes on the basis of summation of all retrieved supervoxel similarities for a volume.
- **sumRanking2**: Like sumRanking1. However, similarities, higher than a threshold are upscaled by a certain factor.
- **threshold**: Like sumRanking2, but the similarities are not upscaled and values below the threshold are not considered at all for volume ranking.
- **distance**: Ranking, based on the summation of similarity values. However, the values are rescaled according to their in-volume distance of the supervoxel center-coordinates to the supervoxels with the highest similarities in order to give favor to volumes with clusters of similar supervoxels.

4.3 Evaluation experiments

All experiments for evaluating the mentioned CBIR-system components were conducted on real world data of 327 oversegmented lung CT scans recorded in the daily routine of a hospital. The weak labels $\mathcal{T}_i$ of the images were extracted from radiological reports by means of Natural Language Processing (NLP). The example information needs were defined as query supervoxels with known anomaly. These supervoxels were manually extracted from the same real world dataset. The anomalies considered for retrieval were chosen with respect to: (1) their frequency of occurrence in the data-set and (2) their visual impact in a CT image. The frequencies of occurrence was considered to avoid a too limited set of relevant volumes, that may be biased

	# Queries	# Volumes
Emphysema	61	10
Ground-glass	63	6
Bulla	34	6
Atelectasis	39	8
Total	197	30

Table 2: The number of queries extracted for each anomaly. The second column shows the number of different volumes of which the queries have been extracted from.

to imaging quality or just little variation of appearance. Some diagnostic labels extracted from radiological reports may only be meaningful in a wider context of medical history and diagnosis. Thus, only anomalies were a direct connection between diagnostic label and visual appearance in the CT image can be drawn are considered. A radiologist suggested Emphysema, Ground-glass, Bulla and Atelectasis for these experiments. Table 2 lists the number of queries extracted for each anomaly and the number of volumes they have been extracted from. It is infeasible to evaluate all parameter respectively method variations of the considered pipeline. Thus, we decided to look at each component separately. A promising candidate of a processing step is used to vary parameters or methods of the subsequent module. For each evaluation round we produced (1) qualitative results by visualizing the nearest neighbor supervoxels for a query and (2) quantitative results by plotting the mean precision at different ranks. However, as no ground truth is available in the data, a result supervoxel is considered as relevant if it is from a volume carrying the same weak label as the label of the query supervoxel. More formally, if $p_{i,s}$ is a result supervoxel and q a query with known anomaly

$$rel(p_{i,s}) = \begin{cases} 1 & \text{if } l(q) \in \mathcal{T}_i \\ 0 & \text{if } l(q) \notin \mathcal{T}_i \end{cases} \tag{13}$$

To judge the relevance of volumes Equation 13 is used as well.

4.4 Results

In the following, results for the anomaly 'Bulla' are shown only as a representative example. Parameter variations have been performed for different image descriptors. Figure 4 shows plots of mean precision at different ranks from 50 to 500 for different parameters of the charalick descriptor. Figure 5 shows plots of mean precision for different descriptors. To illustrate the retrieval results, 3 nearest neighbors for a query supervoxel are shown for each method. Figure 6 shows results for different indexing methods compared to euclidean distance and Figure 7 shows precision at k's from 5 to 100 for retrieved volumes by applying different ranking methods. For the volume ranking, groups of query supervoxels between 1 and 5 have been used and result lists of 10 000 nearest neighbors for each query supervoxel were retrieved. Figure 8 shows examples of nearest neighbors for a query supervoxel of each anomaly using charalick descriptors and PQSMlookup.

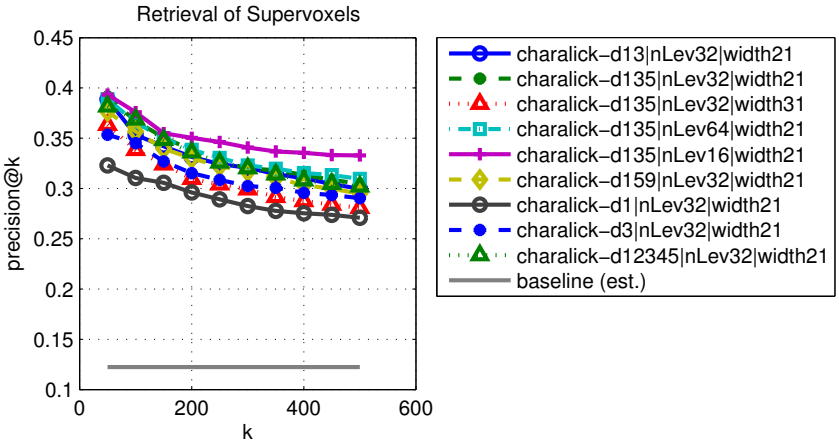


Figure 4: Plots of mean precision of 34 queries for different k from 50 to 500 and different image descriptors. On the right side some exemplary nearest neighbors for a single query are given. Baseline is the share of volumes with the label bulla in the dataset.

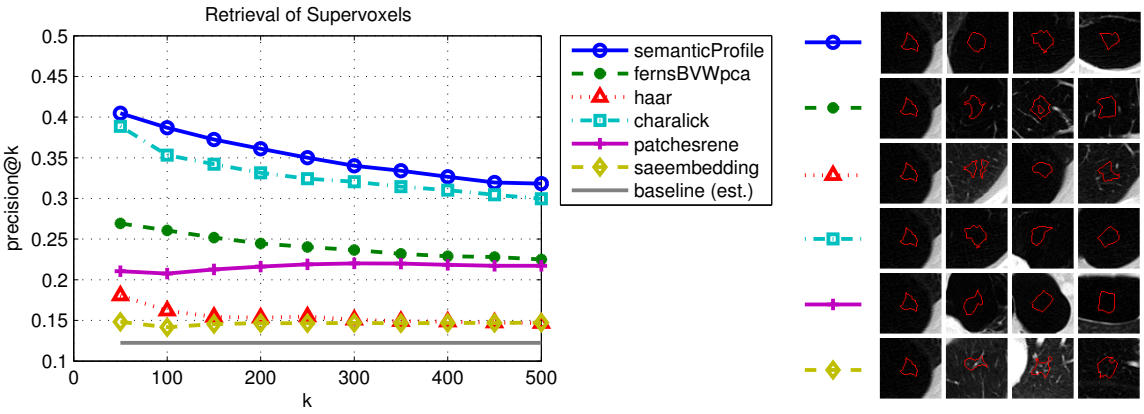


Figure 5: Plots of mean precision of 34 queries for different k from 50 to 500 and different image descriptors. On the right side some exemplary nearest neighbors for a single query are given. Baseline is the share of volumes with the label bulla in the dataset.

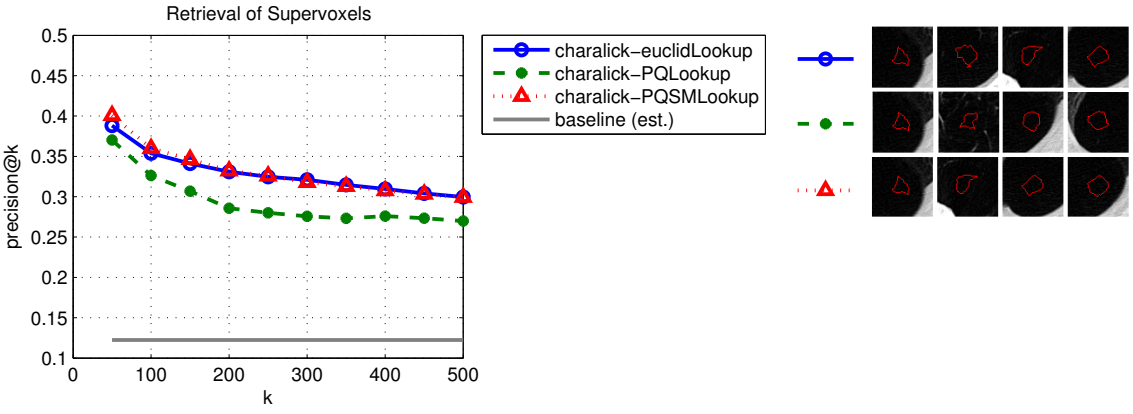


Figure 6: Plots of mean precision of 34 queries for different k from 50 to 500 and different indexing techniques. On the right side some exemplary nearest neighbors for a single query are given.

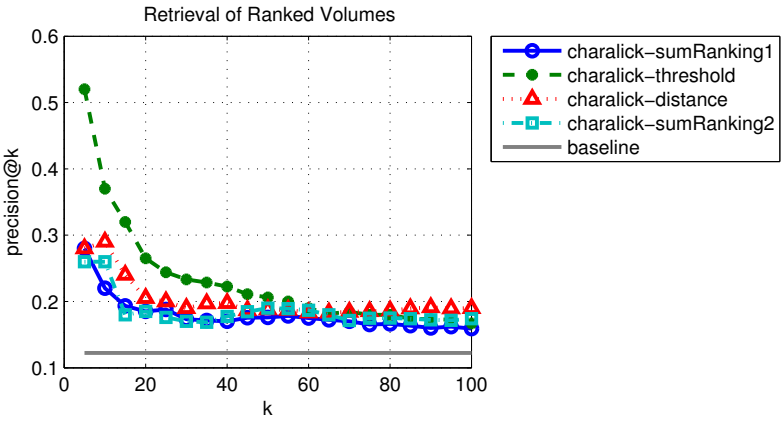


Figure 7: Plots of mean precision of 10 query groups for different k from 5 to 100 and different ranking metrics.

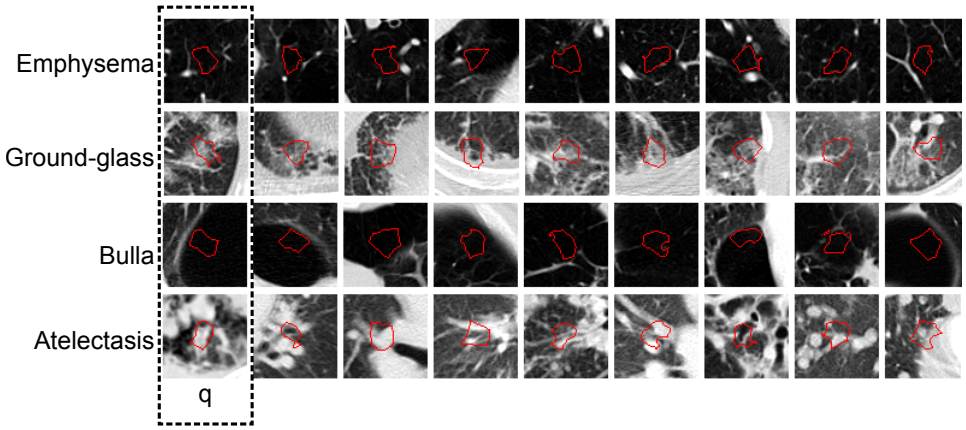


Figure 8: Examples for nearest neighbors for query voxels (first column) of different anomalies. Charalick descriptors and PQSM indexing was used.

4.5 Discussion

Several descriptors have been compared with respect to their ability to encode the appearance of different anomalies in lung CT images. Haralick features showed the consistent performance for the four anomalies. All of the considered descriptors are able to deliver visually similar voxel patches illustrated by visualizing nearest neighbor supervoxels. Product quantization is a suitable way to improve retrieval speed and reduce the memory footprint of the index. However, the non-exhaustive version PQSM allows better approximation than the exhaustive version while using the same code length. The reason is a preliminary quantization of the whole index and a subsequent approximation of the residual vectors. As the energy of the residuals is small compared to the original vectors, the approximation can be more accurate. However, the training and encoding step of the index building is more complex and costly. Ranking metrics were compared on the basis of charalick descriptors and PQSM lookups. Thresholding the supervoxel result lists on a certain metric of the distances showed improved performance compared to ranking methods which consider all result supervoxels.

5 Relevance feedback

A well known technique trying to improve search results by user interaction, called relevance feedback [23], is included in the KHRESMOI 2D image retrieval subsystem. Relevance feedback allows the user to mark results returned in a previous search step as relevant or irrelevant to refine the initial query. The concept behind relevance feedback is that though user may have difficulties in formulating a precise query for a specific task, they generally see quickly whether a returned result is relevant to the information need or not. This technique found use in image retrieval particularly with the emerge of content–based image retrieval (CBIR) systems [26, 28, 30].

Following the CBIR mentality, the visual content of the marked results is used to refine the initial image query. With the result images represented as a grid of thumbnails, relevance feedback can be applied quickly to speed up the search iterations and refine results. The first round of user–centered evaluation of KHRESMOI Radiology system showed that this method is intuitive and straightforward to learn [16].

Depending on whether the user manually provides the feedback to the system (e.g. by marking results) or the system obtains this information automatically (e.g. by log analysis) relevance feedback can be categorized as explicit or implicit. Moreover, the information obtained by relevance feedback can be used to affect the general behaviour of the system (long–term learning).

In this section we perform an empirical evaluation of different explicit, short–term relevance feedback techniques using visual content or text for medical image retrieval. Part of this section is included in [19] and has been accepted for publication.

5.1 Method overview

One of the most well known relevance feedback techniques is Rocchio's algorithm [23]. Its mathematical definition is given below:

$$\vec{q}_m = \alpha\vec{q}_o + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j \in D_{nr}}\vec{d}_j \tag{14}$$

where

- $\vec{q}_m$ is the modified query,
- $\vec{q}_o$ is the original query,
- D_r is the set of relevant images,
- D_{nr} is the set of non–relevant images and
- α, β and γ are weights.

Typical values for the weights are $\alpha = 1, \beta = 0.8$ and $\gamma = 0.2$. Rocchio's algorithm is typically used in vector models and also for CBIR. Intuitively, the original query vector is moved towards the relevant vectors and away from the irrelevant ones. By giving a weight to the positive and negative parts a problem of CBIR can be avoided that when more negative than positive feedback exists that also many releavnt images disappear from the results set.

Another technique that showed potential in image retrieval [11] is late fusion. Late fusion [5] is used in information retrieval to combine result lists. It can be applied for fusing multiple features, multiple queries and in multi–modal techniques. The concept behind this method is to merge the result lists into a single list while boosting common occurrences using a fusion rule.

For example, the fusion rule of the score–based late fusion method CombMNZ [25] is defined as:

$$S_{\texttt{combMNZ}}(i) = F(i) * S_{\texttt{combSUM}}(i) \tag{15}$$

where $F(i)$ is the number of times an image i is present in retrieved lists with a non–zero score, and $S(i)$ is the score assigned to image i. CombSUM is given by

$$S_{\texttt{combSUM}}(i) = \sum_{j=1}^{N_j} S_j(i) \tag{16}$$

where $S_j(i)$ is the score assigned to image i in retrieved list j.

Similarly, the fusion rule of the reciprocal rank fusion method [3] is given by:

$$S_{\texttt{RRF}}(i) = \sum_{j=1}^{N_j}\frac{1}{c + R_j(i)} \tag{17}$$

where c a constant and $R_j(i)$ the rank of the image in retrieved list j.

Most of the techniques use vectors either from the text or the visual models. However, it has been shown that approaches that use both text and visual information can outperform single–modal ones in image retrieval. We evaluate also the use of multi–modal information for relevance feedback to enhance the retrieval performance. As late fusion is applied on result lists, it is straightforward to use for combining results from visual and text queries.

5.2 Evaluation experiments

For evaluating the relevance feedback techniques the following experimental setup was followed: The n search iterations are initiated with a text query in iteration 0. The relevant results from the top k results of iteration i were used in the relevance feedback formulae of the iteration $i+1$ for $i = 0...n-2$.

The image dataset, topics and ground truth of ImageCLEF 2012 medical image retrieval task [20] were used in this evaluation. The dataset contains more than 300'000 images from the medical open access literature.

The image captions were accessed by the text–based runs and indexed with the Lucene[4] text search engine. Vector space model was used along with tokenization, stopword removal,stemming and Inverse document frequency-Term frequency weighting. The Bag–of–visual–words model described in [9] and the bag–of–colors model appearing in [10] were used for the visual modelling of the images. In multimodal runs, the fusion of the visual and text information is performed only for the text 1000 top results as in the evaluation of ImageCLEF only the top 1000 documents are taken into account in any case.

Five techniques were evaluated in this study:

1. **text**: text–based RF using vector space model. Word stemming, tokenization and stop word removal is performed in both text and multi–modal runs.

2. **visual_rocchio**: visual RF using Rocchio to fuse the relevant image vectors and CombMNZ fusion to fuse the original query's results with the visual ones.

3. **visual_lf**: visual RF using late fusion (and the CombMNZ fusion rule) to fuse the relevant image results and the original query results with the visual ones.

4. **mixed_rocchio**: multimodal RF using Rocchio to fuse the relevant image vectors and CombMNZ fusion to fuse the original query results with the relevant caption results and relevant visual results.

5. **mixed_lf**: multimodal RF using late fusion (and the CombMNZ fusion rule) to fuse the relevant image results and the original query results with the captions' results and relevant visual results.

[4] http://lucene.apache.org/

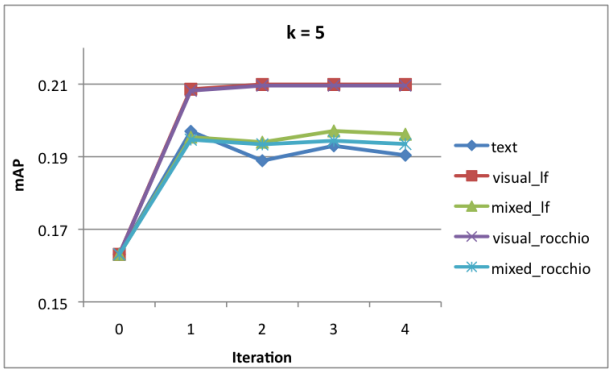


Figure 9: Mean average precision per search iteration for $k = 5$.

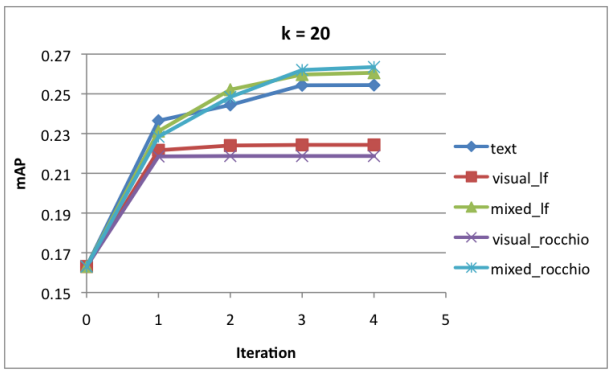


Figure 10: Mean average precision per search iteration for $k = 20$.

5.3 Results

The evaluation of the five techniques was performed for $k = 5, 20, 50, 100$ and $n = 5$. Results of the mean average presicion (MAP) of each technique per iteration are shown in Figures 9, 10, 11, 12.

Table 3 gives the best mAP scores of each run. The numbers in parentheses are the number of the iteration when this score was achieved. For scores that were the same in multiple iterations of the same run, the iteration closer to the first is used.

Table 3: Best mAP scores

Run	k = 5	k = 20	k = 50	k = 100
text	0.197 (1)	0.2544 (4)	0.3107 (3)	0.3349 (4)
visual_lf	0.2099 (2)	0.2243 (3)	0.2405 (4)	0.2553 (3)
visual_roc	0.2096 (2)	0.2187 (2)	0.2249 (3)	0.2268 (2)
mixed_lf	0.1971 (3)	0.2606 (4)	0.3079 (4)	0.3487 (3)
mixed_roc	0.1947 (1)	0.2635 (4)	0.3207 (4)	0.3466 (4)

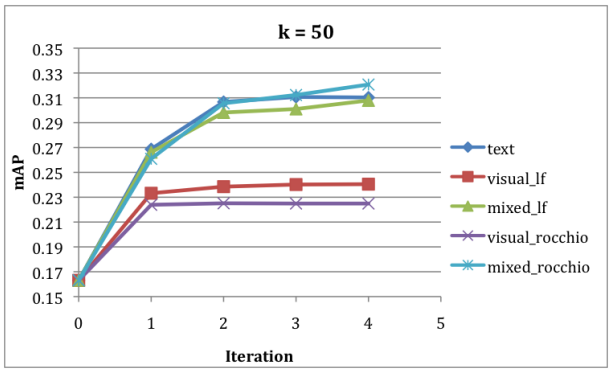


Figure 11: Mean average precision per search iteration for $k = 50$.

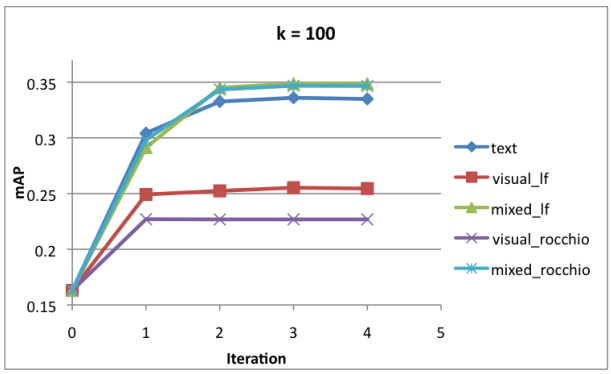


Figure 12: Mean average precision per search iteration for $k = 100$.

5.4 Discussion

All of the evaluated techniques improve retrieval after the initial search iteration. This demonstrates the potential of relevance feedback for refining medical image search queries.

Relevance feedback using only visual appearance models, even though improving the retrieval performance after the first iteration, performed worse than the text–based runs in most cases. Visual features still suffer from the semantic gap between the expressiveness of visual features and our human interpretation. Still, this shows their usefulness in image datasets where no or little text meta–data are available. Moreover, when combined with the text–information in the proposed method, they improve the text–only baseline.

The multi–modal runs provide the best results in all the cases except for case $k = 5$. Surprisingly, the visual runs perform slightly better than the text and the multi–modal approaches for this case. However, assuming independent and normal distributed average precision values the significance tests show that the difference is not statistically significant.

We consider the case $k = 20$ as the most realistic scenario since users do not often inspect more than 2 pages of results. Especially for grid–like result interface views, where each page can contain 20 to 50 results, we consider $k = 20$ more realistic than $k = 5$. In this case the multi–modal methods achieve the best performance with 0.2606 and 0.2635 respectively. Again, the significance tests do not find any signicance difference between the three best approaches. However, applying different fusion rules for combining visual and text information (such as

linear–weighting) could further improve the resuts of the mixed approaches.

It can be noted that as the k increases, the performance improvement also increases, highlighting the added value of relevance feedback. Larger values of k were not explored as this scenarios were judged as unrealistic.

In the visual runs using Rocchio for combining the visual queries is performing worse than late fusion. This comes in accordance with the findings in [9]. The reason behind this could be that the large visual diversity of relevant images in medicine and the curse of dimensionality cause the modified vector to behave as an outlier in the high dimensional visual feature space. In the mixed runs the difference between the two methods is not statistically significant with Rocchio performing slightly better than the late fusion.

Irrelevant results were ignored, as they often have little or no impact on the retrieval performance [21]. More importantly, the ground truth of the dataset used contains a much larger portion of annotated irrelevant results than relevant ones. This was considered to potentially simulate an unrealistic scenario, as users do not usually mark many results as negative examples. Having too many negative examples could also cause the modified vector to follow an outlier behaviour. Preliminary results confirmed this hypothesis, where the use of negative results for relevance feedback can decrease performance after the first iteration.

It should be noted that this is an automated relevance feedback experiment of positive only feedback and that in selective relevance feedback situations the retrieval performance is expected to perform even better. A larger number of steps could be investigated but this might be unrealistic, given the fact that physicians have little time and stop after a few minutes of search [18]. Often users will only test a few steps of relevance feedback at the most.

6 Text based image retrieval

In this section we present the results of the evaluation of the text based image retrieval system implemented within the K4E[5] search engine. The image retrieval system for which we are presenting the evaluation results here is described in details in [17] together with the system redesign.

6.1 Method overview

Even though the first evaluation of the above described system [15] has given promising results, users feedback has indicated the problems of low relevance of the retrieved images to the query. In the [17] Section 3.3.1, we have listed the issues detected that might have been the reason for this poor performance. It has been noticed that the collection of ImageCLEF 2011 Wikipedia retrieval task [29] is not the appropriate collection for evaluation of this system. When extracting image information from the web pages crawled by our system we are facing issues such as: slide shows, surrounding text not related to the image itself or image having different host than that of the webpage we are extracting from. These issues are not present in Wikipedia pages.

Due to the lack of suitable collection for this system evaluation, HON has opted for the creation of the test collection. The creation of the data collections used in this evaluation is described in next section.

[5] http://everyone.khresmoi.eu/hon-search/

6.1.1 Data collection

A total of 56 queries were used in this evaluation. These queries were chosen as the most frequent ones from the search engine logs. For creation of the testing collection HON has opted for crowd sourcing method (CrowdFlower[6]). Images related to the queries are retrieved and submitted to be evaluated by the "crowd". The results returned by the crowd sourcing were then used for the evaluation.

6.2 Evaluation experiments

The evaluation was performed in two phases. In the first phase, total of 2902 relevance judgements were obtained. These judgements were then used for the system evaluation. In order to measure retrieval performance, we have adopted the mean average precision (MAP) computed by trec_eval [1] based on maximum of 1000 retrieved items. Different features extracted from the web pages (e.g. title, alt, precedingtText, followingText, largePrecedingText, largeFollowingText) were used as fields to be searched in within the solr index. Different boost values combination on various filed were also tested.

After evaluating results obtained from this first evaluation phase, and based on the feedback obtained from it HON has proceeded with feature extraction redesign as described in [17], Section 3.3.2. In addition to the search field used in the first evaluation phase, the filed "pageTitle" was added. This field represents the title of the page from which the image was extracted.

The redesigned system has undergone a new evaluation. In this second evaluation phase the same set of queries was used as for the first phase. In this phase the "CrowdFlower" results contained 10753 judgement. The same evaluation approach as in the first phase was used.

6.3 Results

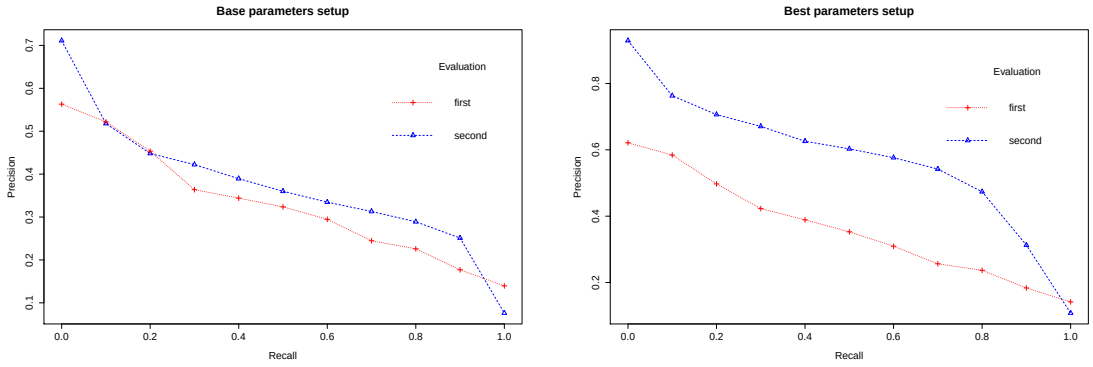
In this section we present the results of the evaluation described above. We present results obtained for total of 4 runs, base runs for both first and second evaluation phase, and best runs for both evaluation phases. For both base runs all available search fields were used, without score boosting.Thus for the first evaluation phase this represents: "title, alt, precedingtText, followingText, largePrecedingText, largeFollowingText". For the second evaluation phase the filds searched in are: "title, alt, precedingtText, followingText, largePrecedingText, largeFollowingText, pageTitle". The runs marked as "best" present the search filed and boost value parameter combination that resulted in highest overall map value. The "best" run in the first evaluation was the run with following parameters:"title alt^6 precedingText followingText largePrecedingText largeFollowingText", where alt^6 represents the boost value of 6 for the "alt" search field. In the second evaluation phase the "best" is the run with parameters:"title^2 alt^4 largePrecedingText largeFollowingText^2 pageTitle".

Table 4 brings the values of the Mean Average Precision for four runs previously described. In the Figures 13 and 14 we present the comparison of the different averaged 11-point precision/recall graphs across 56 queries.

[6] http://crowdflower.com/

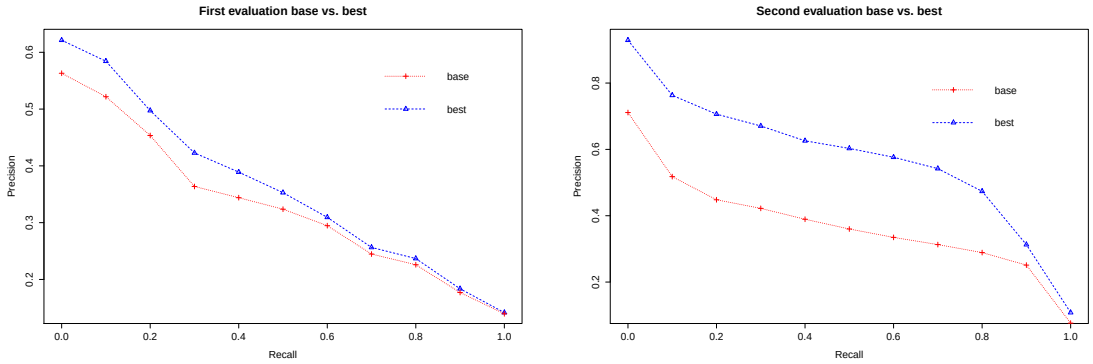
Table 4: K4E image retrieval evaluation: MAP scores

	Evaluation phase	
Run	first	second
base	0.3057	0.3529
best	0.3388	0.5590



(a) base runs (b) best runs

Figure 13: Precision/recall: first vs. second evaluation phase



(a) first evaluation phase (b) second evaluation phase

Figure 14: Precision/recall: best vs.base

6.4 Discussion

As it can be seen from the results presented in the previous section, in the first evaluation phase system was capable of achieving the MAP off 0.3507 for the base parameter setup. Adjusting the boost for various search filed, we have been able to gain 10%, namely the MAP of 0.3388 was achieved for the "best" parameter setup. At this point, we were unable to further ameliorate the performance of the system. Based on the results of the first evaluation phase, the algorithm of extraction of the features related to the images was changed. Another search field "pageTitle" was also added. Presented results show that these steps resulted in better retrieval performance.

Adjusting different parameters enabled achieving the MAP of 0.559 for the run “best” in second evaluation phase. It can be noticed that this “best” run in the second evaluation phase does not use “precedingtText” and “followingText” as field to search in. These fields, as described in [17] are attributed to the image they surround with out any discrimination, unlike “largePreceding-Text” and “largeFollowingText” for which the lexical distance to “title” and “alt” is calculated. Thus, it has been determined that usage of these two search fields hurts system performance.

The systems increase in performance between base and best runs in both evaluation phases is also evident from the precision/recall curves given in Figure 14. This is also the case for corresponding runs from different evaluation phases, given in Figure 13.

Based on the results obtained in this second evaluation phase, the image search system of the K4E search engine is adapted to use the parameters of the “best” run in the second evaluation phase.

7 Conclusion

This deliverable describes the results of the second evaluation phase of the KHRESMOI prototype, focused on methods and algorithms developed and integrated in workpackage 2. We performed quantitative evaluations of individual core components of the prototype, in order to allow readers to understand the role and accuracy of individual parts of the overall prototype. During the project several novel methods were introduced, ranging from the localization of anatomical structures, to feature extraction, and content based retrieval of imaging data. The methods were developed and improved through-out the project. Starting from the first evaluation phase, and the first round of user tests, methods were adapted, or re-thought based on the results obtained, once methods were integrated in a modular system, consisting of multiple parts. The results demonstrate that the individual components perform well, and are capable of high accuracy, stability, while covering the needs of a range of usecases. This deliverable is complimentary to the report on the user-tests that are performed to evaluate the entire prototype as a whole.

References

[1] C. Buckley and E. M. Voorhees. Retrieval system evaluation. *TREC. Experiment and evaluation in information retrieval*, pages 53–75, 2005.

[2] Andreas Burner, René Donner, Marius Mayerhoefer, Markus Holzer, Franz Kainberger, and Georg Langs. Texture bags: anomaly retrieval in medical images based on local 3d-texture similarity. In *Medical Content-Based Retrieval for Clinical Decision Support*, pages 116–127. Springer, 2012.

[3] Gordon V. Cormack, Charles L A Clarke, and Stefan Büttcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 758–759, New York, NY, USA, 2009. ACM.

[4] Adrien Depeursinge, Antonio Foncubierta-Rodríguez, Dimitri Van De Ville, and Henning Müller. Lung texture classification using locally–oriented Riesz components. In Gabor Fichtinger, Anne Martel, and Terry Peters, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2011*, volume 6893 of *Lecture Notes in Computer Science*, pages 231–238. Springer Berlin / Heidelberg, September 2011.

[5] Adrien Depeursinge and Henning Müller. Fusion techniques for combining textual and visual information retrieval. In Henning Müller, Paul Clough, Thomas Deselaers, and Barbara Caputo, editors, *ImageCLEF*, volume 32 of *The Springer International Series On Information Retrieval*, pages 95–114. Springer Berlin Heidelberg, 2010.

[6] René Donner, Bjoern H Menze, Horst Bischof, and Georg Langs. Global localization of 3d anatomical structures by pre-filtered hough forests and discrete optimization. *Medical image analysis*, 17(8):1304–1314, 2013.

[7] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. Why does unsupervised pre-training help deep learning? *The Journal of Machine Learning Research*, 11:625–660, 2010.

[8] Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980.

[9] Alba García Seco de Herrera, Dimitrios Markonis, Ivan Eggel, and Henning Müller. The medGIFT group in ImageCLEFmed 2012. In *Working Notes of CLEF 2012*, 2012.

[10] Alba García Seco de Herrera, Dimitrios Markonis, and Henning Müller. Bag of colors for biomedical document image classification. In Hayit Greenspan and Henning Müller, editors, *Medical Content–based Retrieval for Clinical Decision Support*, MCBR–CDS 2012, pages 110–121. Lecture Notes in Computer Sciences (LNCS), October 2013.

[11] Alba García Seco de Herrera, Dimitrios Markonis, Roger Schaer, Ivan Eggel, and Henning Müller. The medGIFT group in ImageCLEFmed 2013. In *Working Notes of CLEF 2013 (Cross Language Evaluation Forum)*, September 2013.

[12] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[13] DR Holmes III, BJ Bartholmai, RA Karwoski, V Zavaletta, and RA Robb. The lung tissue research consortium: an extensive open database containing histological, clinical, and radiological data to study chronic lung disease. In *The Insight Journal-2006 MICCAI Open Science Workshop*, 2006.

[14] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2011.

[15] Georg Langs, Joachim Ofner, Andreas Burner, René Donner, Henning Muller, Adrien Depeursinge, Dimitrios Markonis, Célia Boyer, Alexandre Masselot, and Nolan Lawson. D2.3: Report on results of the wp2 first evaluation phase. *Khresmoi project public deliverable*, 2012.

[16] Dimitrios Markonis, Frederic Baroz, Rafael Luis Ruiz de Castaneda, Celia Boyer, and Henning Müller. User tests for assessing a medical image retrieval system: A pilot study. In *MEDINFO 2013*, 2013.

[17] Dimitrios Markonis, René Donner, Ljiljana Dolamic, Roger Schaer, Georg Langs, Ceélia Boyer, and Henning Müller. D2.6: Report on and prototype of final image retrieval and analysis framework. *Khresmoi project public deliverable*, 2014.

[18] Dimitrios Markonis, Markus Holzer, Sebastian Dungs, Alejandro Vargas, Georg Langs, Sascha Kriewel, and Henning Müller. A survey on visual information search behavior and requirements of radiologists. *Methods of Information in Medicine*, 51(6):539–548, 2012.

[19] Dimitrios Markonis, Roger Schaer, and Henning Müller. Multi-modal relevance feedback for medical image retrieval. In *SIGIR 2014, Medical Information Retrieval (MedIR) Workshop*, 2014. accepted for publication.

[20] Henning Müller, Alba García Seco de Herrera, Jayashree Kalpathy-Cramer, Dina Demner Fushman, Sameer Antani, and Ivan Eggel. Overview of the ImageCLEF 2012 medical image retrieval and classification tasks. In *Working Notes of CLEF 2012 (Cross Language Evaluation Forum)*, September 2012.

[21] Henning Müller, Wolfgang Müller, David McG. Squire, Stéphane Marchand-Maillet, and Thierry Pun. Strategies for positive and negative relevance feedback in image retrieval. Technical Report 00.01, Computer Vision Group, Computing Centre, University of Geneva, rue G n ral Dufour, 24, CH–1211 Gen ve, Switzerland, January 2000.

[22] Jiquan Ngiam, Adam Coates, Ahbik Lahiri, Bobby Prochnow, Quoc V Le, and Andrew Y Ng. On optimization methods for deep learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 265–272, 2011.

[23] J. J. Rocchio. Relevance feedback in information retrieval. In *The SMART Retrieval System, Experiments in Automatic Document Processing*, pages 313–323. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1971.

[24] Ruslan Salakhutdinov and Geoffrey E Hinton. Deep boltzmann machines. In *International Conference on Artificial Intelligence and Statistics*, pages 448–455, 2009.

[25] Joseph A. Shaw and Edward A. Fox. Combination of multiple searches. In *TREC-2: The Second Text REtrieval Conference*, pages 243–252, 1994.

[26] David McG. Squire, Wolfgang Müller, Henning Müller, and Thierry Pun. Content–based query of image databases: inspirations from text retrieval. *Pattern Recognition Letters (Selected Papers from The 11th Scandinavian Conference on Image Analysis SCIA '99)*, 21(13–14):1193–1198, 2000. B.K. Ersboll, P. Johansen, Eds.

[27] Rainer Storn and Kenneth Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.

[28] Leonid Taycher, Marco La Cascia, and Stan Sclaroff. Image digestion and relevance feedback in the ImageRover WWW search engine. pages 85–94, 1997.

[29] Theodora Tsikrika, Adrian Popescu, and Jana Kludas. Overview of the wikipedia retrieval task at imageclef 2011. *Working Notes of CLEF 2011*, 2011.

[30] Matthew E.J. Wood, Neill W. Campbell, and Barry T. Thomas. Iterative refinement by relevance feedback in content–based digital image retrieval. pages 13–20, 1998.