Interactive Content-Based Image Retrieval with Deep Neural Networks

Joel Pyykkö and Dorota Głowacka(✉)

Department of Computer Science, HIIT, University of Helsinki, Helsinki, Finland
{joel.pyykko,dorota.glowacka}@cs.helsinki.fi

Abstract. Recent advances in deep neural networks have given rise to new approaches to content-based image retrieval (CBIR). Their ability to learn universal visual features for any target query makes them a good choice for systems dealing with large and diverse image datasets. However, employing deep neural networks in interactive CBIR systems still poses challenges: either the search target has to be predetermined, such as with hashing, or the computational cost becomes prohibitive for an online setting. In this paper, we present a framework for conducting interactive CBIR that learns a deep, dynamic metric between images. The proposed methodology is not limited to precalculated categories, hashes or clusters of the search space, but rather is formed instantly and interactively based on the user feedback. We use a deep learning framework that utilizes pre-extracted features from Convolutional Neural Networks and learns a new distance representation based on the user's relevance feedback. The experimental results show the potential of applying our framework in an interactive CBIR setting as well as symbiotic interaction, where the system automatically detects what image features might best satisfy the user's needs.

Keywords: Content-based image retrieval (CBIR) · Deep neural networks · Interactive systems · Exploratory search

1 Introduction

In recent years, image retrieval techniques operating on meta-data, such as textual annotations, have become the industry standard for retrieval from large image collections. This approach works well with sufficiently high-quality meta-data. However, with the explosive growth of image collections it has become apparent that tagging new images quickly and efficiently is not always possible. Secondly, even if instantaneous high-quality image tagging was possible, there are still many instances where image search by query is problematic. It might be easy for a user to define their query if they are looking for an image of a cat but how do they specify that the cat should be of a very particular shade of ginger with sad looking eyes. A solution to this is content-based image retrieval (CBIR) [7], especially in combination with relevance feedback [28] that actively

L. Gamberini et al. (Eds.): Symbiotic 2016, LNCS 9961, pp. 77–88, 2017.
DOI: 10.1007/978-3-319-57753-1_7

involves the user into the search loop and utilizes his knowledge in the iterative search process [1–3,10,17].

A variety of feature descriptors have been used for image representation in CBIR, such as color, edge, texture. Similarly in use have been local feature representations, such as the bag-of-words models [26] in conjunction with local feature descriptors (e.g. SIFT [20]). However, using such low-level feature representation may not be always optimal for more complex image retrieval tasks due to the semantic gap between such features and high-level human perception. Hence, in recent years there has been an increased interest in developing similarity measures specifically for such low-level feature image representation [5] as well as enhancing the feature representation in distance metric learning [24], which is the approach that we follow in this paper.

Over the past decade deep neural networks have seen many successful applications in various image recognition and classification tasks. These networks use multiple layers of non-linear transformations learning more abstract features from the input data. For example, Convolutional Neural Networks (CNNs) [19] have been shown to work extremely well in image classification tasks, such as the ImageNet competition [22], and they produce features that are highly descriptive for various image recognition tasks, even in tasks for which they were not trained, or higher level concepts, such as scenes [12,21]. However, the interactive nature of CBIR poses additional difficulties with regards to the application of deep neural networks, such as the responsiveness of the system – search engine response time exceeding 4 s already interferes with the user's experience [4]. Additionally, in interactive CBIR, the systems needs to learn what the user is interested in from a very small amount of feedback – at each search iteration users tend to indicate only a few images that they like or do not like [11,13].

Learning deep hierarchies for fast image retrieval was considered before by using autoencoders [18] or creating hash codes based on deep semantic ranking [27]. While both methods are fast, neither is flexible enough to learn the image target based on the small amount of relevance feedback obtained from the user. Wan et al. [24] is the first study to apply deep learning to learn a similarity measure between images in a CBIR setting. Unfortunately, no consideration was given to the time requirements of the learning task, which is an important aspect of an interactive retrieval systems. The reported training procedure uses entire datasets and the training itself can take days. Similarity learning can also be used to find new metrics between faces by maximizing the inter-class difference, while minimizing the inner-class difference [6,14], however, the method was not tested with a broader set of images and features. Two recent studies [8,25] took into consideration the training time requirements. However, their system setting relies on using thousands of images for training, which is too large for a user to tag over the span of a single search session. The system we describe in this paper needs only a small number of images tagged by the user through iterative relevance feedback in order for the system to be trained to find the target image(s).

Our focus in this paper is twofold. Our first goal is to show how to learn a definite representation of the user's target image(s) with only a few training

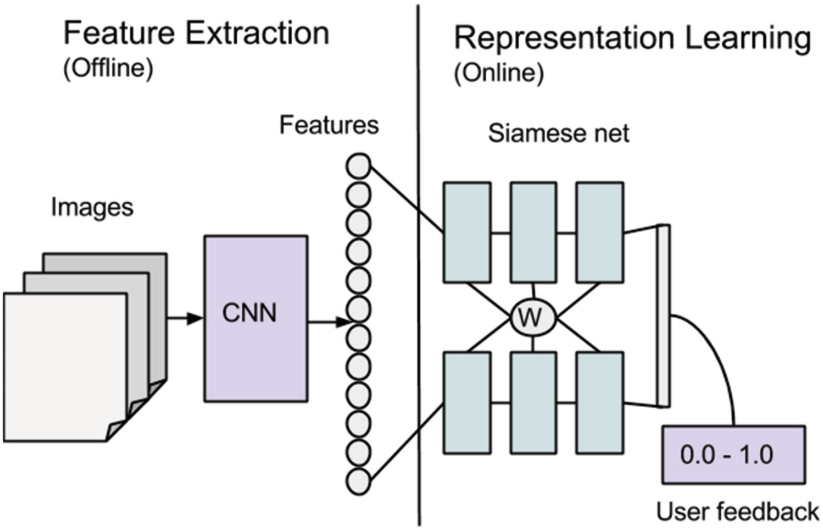


Fig. 1. The siamese architecture with the image feature preprocessing step. The online component accepts two feature vectors, one per image, and user feedback as the label.

examples. This is an important aspect of an interactive CBIR system that can gradually learn from user interaction and adjust to the changes in user's interests as the search progresses. Second, we aim to reduce the training time required for the system to be able to make new suggestions to the user to under 4 s. This will make the proposed system interactive and keep the user engaged in the search loop.

We use a specialized siamese architecture, originally used for face verification [6], that learns the similarity between two example images. This architecture utilizes pre-extracted features from Convolutional Neural Networks (CNNs) and learns a new distance representation based on the user's relevance feedback. Employing ready CNN features as the basis of the similarity learning speeds up the process considerably, while maintaining a broad set of features to prevent the user from getting stuck in a small area of the feature space. The speed of computing the distance metric and the fact that only a small set of examples is needed to learn it makes our framework easily applicable to an interactive retrieval setting, such as CBIR.

2 System Overview

The aim of our system is to assist the user in finding images that cannot be easily described using tags, such as an image of "beautiful sky". Thus, the system needs to learn what the target of the search is through relevance feedback obtained on the small number of images displayed at each iteration. As the user's final search

target may be an image containing any combination of features, our method utilizes a distance metric between images to learn what features or combination of features might be of interest to the user. This allows the system to learn a representation based on the user feedback on the presented images, and show the user more relevant images as the search progresses. The system differs from a classifier in that it does not predict which particular classes the user is interested in but instead tries to learn what features or combination of features might be of interest to the user.

The system adheres to a search procedure that can be briefly summarised as follows. The search starts with a random selection of images presented to the user. At each search iteration, the user is presented with k images and indicates which images are relevant to his search by clicking on them. The remaining images in the set of k images that did not receive any user feedback are treated as irrelevant. Based on this feedback, all the images in the dataset are re-ranked using a distance measure and the top k images are presented to the user at the next iteration. Images that were presented to the user so far are excluded from future iterations. If no images are selected as relevant by the user, we assume that all the presented images are irrelevant, and images that are maximally distant from the presented ones are shown to the user at the next iteration. The search continues until the user is satisfied with the presented images.

Below, we describe the feature extraction process and the architecture of the system in more details.

2.1 Feature Extraction

In order to obtain a good base representation, we use CNNs to extract image features. CNNs generate high quality classification results end-to-end from low, pixel-level data to image labels by utilizing deep non-linear architectures. The higher level features from these networks have been successfully used in tasks involving classification of images that were not in the initial training set. This can be achieved by retraining the features extracted from images to represent the area in the image space that corresponds to the user's interests. For our tests, we use features extracted with OverFeat [23] and relearn only the last few fully connected layers for the target representation. OverFeat is a publicly available CNN trained on the ILSVRC13 dataset [22], on which it achieved an error rate of 14.2%. ILVSRC13 contains 1000 object classes from a total of 1.2 million images. OverFeat has been shown to be successful at various image recognition tasks from fine-grained classification to generic visual instance recognition tasks [21]. The chosen features were a set of hidden nodes as the fully connected graph begins from layer 7 (19 within the architecture), totalling 4096 features. The images were shrunk and then cropped from all sides to produce images of equal size of 231×231 pixels. Table 1 shows the composition of the neural architecture used in our system.

Table 1. Composition of the neural architecture used in our system.

Layer	Input size	Output size	
$FC1$	4096	100	Fully connected layer
$ReLU$			Rectified Linear Unit
$FC2$	100	20	Fully connected layer
$ReLU$			Rectified Linear Unit
$Feat$	20	6	Final feature layer
CLF			Contrastive loss function

2.2 System Architecture

Our system employs the siamese architecture [6], which is used for learning similarities between images by labeling pairs of images as similar or dissimilar, and maximizing the distance between different image groups. We employ user relevance feedback to divide the presented images into the two classes, i.e. images with positive feedback (relevant class) and images with negative feedback (non-relevant class). The overview of the system's architecture can be seen in Fig. 2.

The siamese similarity metric aims to find a function that maps the input into a new space, where the target distance measure, such as Euclidean distance, may be used to determine the proximity of two data points. This similarity function, G, is parameterized with weights W, which the system tries to learn to form the similarity metric:

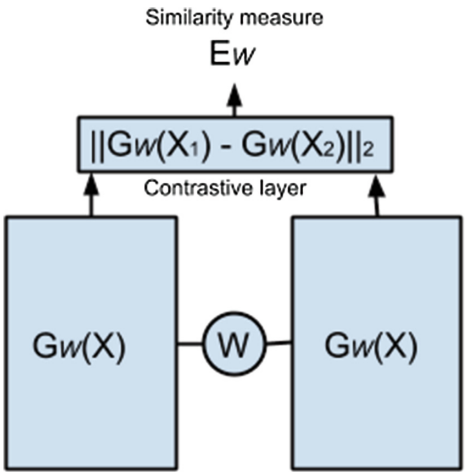


Fig. 2. The siamese architecture: two neural nets G that share the weights W as their parameters. They process the data for the contrastive layer, which outputs the similarity measure E_W.

$$E_W(X_1, X_2) = \|G_W(X_1) - G_W(X_2)\|,$$

where X_1 and X_2 are paired images.

This metric aims to minimize the intra-class similarity, in the case where X_1 and X_2 belong to the same class, and to maximize the inter-class similarity if X_1 and X_2 belong to different classes. The algorithm accepts a pair of observations, which when the loss function is minimized, minimizes or maximizes the similarity metric $E_W(X_1, X_2)$ depending on whether these observations belong to the same class.

The contrastive loss function used in the siamese architecture is:

$$\begin{gathered} L((W, Y, X_1, X_2)^i) = \\ (1 - Y)L_G(E_W(X_1, X_2)^i) + Y L_I(E_W(X, 1, X_2)^i), \end{gathered} \tag{1}$$

where $(Y, X_1, X_2)^i$ is the i-th sample, which is composed of a pair of images and a label (inter- or intra-class), L_G is the partial loss function for an intra-class pair, L_I is the partial loss function for an inter-class pair, and P is the number of training samples [6].

The siamese architecture (Fig. 1) can find an angle in the feature space that helps to distinguish between different aspects of the image, such as different position of the face or different facial expressions, making it an ideal choice for our application. An important aspect of this architecture is the fact that it generates a distance metric, which may be used to rank or generate dynamic relevance scores for all the images in a dataset.

3 Experiments

We conducted a set of simulation experiments to evaluate the applicability of the proposed systems in interactive CBIR. We identified the following aspects of the system's performance to be crucial:

1. The system needs to be trained with only a few training examples, i.e. at each search iteration, the user is presented with only a small number of images and often provides feedback to a subset of these, and the system needs to be able to "learn" what the user is looking for based on this limited feedback;
2. The search target maybe very concrete, e.g. "red rose", or very abstract, e.g. "happiness", and the system needs to support all types of searches with varying degrees of abstractness;
3. Training time has to be below 4 s for the system to be interactive.

3.1 Experimental Set-Up

We ran a number of simulations to assess the performance of our system. At each iteration, the system presents 10 images to the simulated user. The target of each search is a class of images with a given label, e.g. "dogs", and the simulated user "clicks" on relevant images from a given target class at each iteration, i.e. the

user feedback is 1 for images with a relevant label and 0 for the remaining images in the presented set. The number of relevant images in each iteration can vary from 0 to 10, depending on the number of relevant images in a given dataset and on the accuracy of the user throughout the search session. We assume that the user clicks only on images with the relevant label and that the user clicks on all the relevant images presented in a given iteration. To test whether the system can generalize, we also included as search targets images whose labels were not included in the training set. The search starts with a random selection of 9 images from a given test dataset plus one image with the label of the target class for a specific search – this setting allows us to ensure that all the simulation experiments have a comparable starting point. In summary, our system supports the user in finding an image that best matches their ideal target image(s) in the manner described below. In each iteration, k images from the database $\mathscr{D}$ are presented to the user and the user selects the relevant image(s) from this set, according to the following protocol:
For each iteration $i = 1, 2, \ldots$ of the search:

- Search engine calculates a set of images $\mathbf{x}_{i,1}, \ldots, \mathbf{x}_{i,k} \in \mathscr{D}$ to present to the user.
- If one or more of the presented images are of interest to the user, then the user clicks on them thus providing relevance score of 1 to the clicked images. All the remaining images in the presented set automatically receive relevance feedback of 0.
- If none of the presented images is of interest to the user, then the user proceeds to the next iteration and all the presented images automatically receive relevance feedback of 0.
- The search continues until the user finds their ideal target image.

We used Caffe [16] to produce the live network described above. The simulation experiments were run on a machine with an Intel Core $i5-4430$ CPU 3.00 $\times 4$ GHz and a GeForce GTX 660 Ti.

We used three different datasets (Fig. 3):

1. 1096 images from the MIRFlickr dataset [15] with various combination of the following labels: mammals, birds, insects, locomotives. This dataset allowed us to test whether the learned metric is able to generalize to abstract concepts. The arbitrary nature of these classes with regards to the model of the feature extractor is perfect to demonstrate the robustness of our system: the features extracted from the images may be widely different within each label class but as long as each label can be distinguished with a set of features, the system should be able to learn it.
2. Our own collection of 294 images of 6 different dog breeds, of which only four are included in the OverFeat classification list. This dataset allows us to test whether the model is able to learn the target in the presence of semantically related images, some of which are not included in original scope of features used for training. Such a scenario is quite common in a CBIR setting as the search gradually narrows down towards very specific set of image, e.g. the user

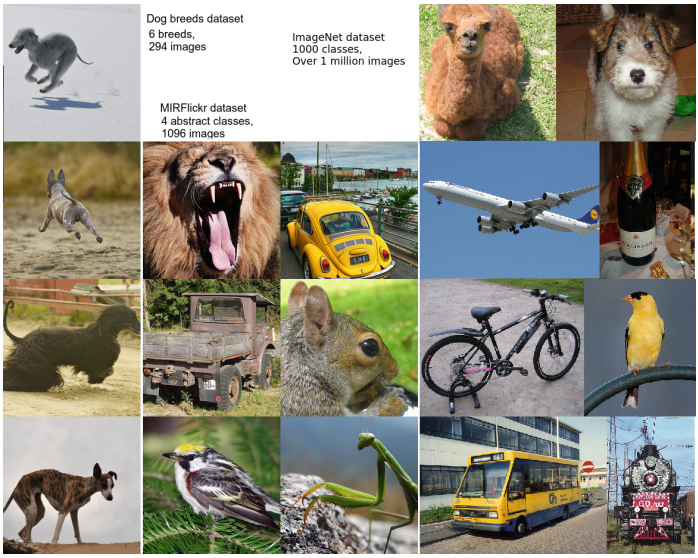


Fig. 3. Example images from the three datasets used in our experiments.

starts a search for images of dogs and gradually narrows down the search to images of black dogs with pointy ears and bushy tails.

3. 300 classes from the ImageNet dataset [22], totalling 385412 images. We used this dataset to show that even if the presented images could potentially lead to hundreds of different target images, the learned representation is still able to detect the relevant features and steer the search towards the most relevant images.

In the experiments with the ImageNet and MIRflickr datasets, we simulated 15 search iterations, which is the average number of iterations of a typical CBIR search session [9]. In the experiments with the dog breeds dataset, we simulated only 12 search iterations due to the small size of the dataset. This setting resulted in a gradually increasing training set, starting from 10 images at the beginning of the search session and gradually increasing by 10 images with each search iteration. This setting allowed us to test the robustness of our system with respect to a small number of training examples. All the reported results are averaged over 5 training runs for each of the existing classes in a given dataset.

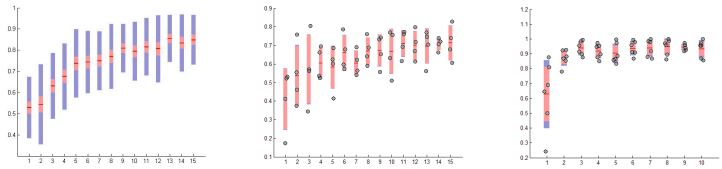
Before running the simulations, we conducted a number of experiments to configure our system and to learn what effect various networks parameters have on the overall performance. By varying the number of layers between one to three, we noticed smaller gains in the Imagenet dataset, while with the other datasets the accuracy improved when extra layers were added. We varied the number of training iterations and noticed no significant improvement after a thousand iterations. We settled for 1500 iterations for the final simulations. With these results, we chose a structure that takes at most 4 s to train,

while maximizing gains from the network structure. For the siamese architecture, the training time was already closer to 4 s with two hidden layers, thus we chose a smaller structure: the incoming 4096 image features are mapped first onto 100 features, then to 20, with the final mapping to 6 output values.

3.2 Experimental Results

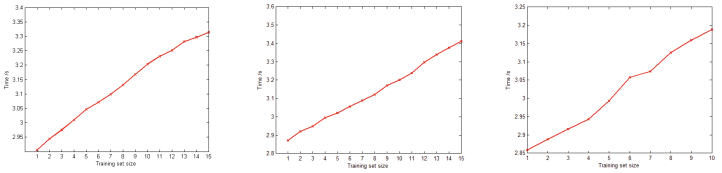
The aim of the experiments was to test whether the system is able to find the target image or class of images with a relatively small number of training examples and whether the training time for each iteration is short enough to allow the system to be used interactively. The test results are shown in Figs. 4 and 5. We show the F1 measure and the training time for each dataset. The system is able to retrieve relevant images from all the datasets within the first few iterations. Initially, the confidence intervals are wide, which reflects the uncertainty of the system with regards to the user's search target. However, as the search progresses and the system receives more and more training points and user feedback, the confidence intervals are getting narrower, indicating that the system is gradually zooming in on a specific area of the search space.

In Fig. 5 we show the average training time for each search iteration. For each dataset, the average duration of each search iteration is below the 4 s required to make the system interactive from the usability perspective. This is the case even when the number of the training datapoints grows with each iteration.



(a) ImageNet (b) MIRFlickr (c) dog breeds

Fig. 4. Test F1-scores (with confidence intervals) for each of the three datasets used in our experiments. The F-1 score increases with the number of iterations and thus more user feedback provided to the system.



(a) ImageNet (b) MIRFlickr (c) dog breeds

Fig. 5. Training times for the three datasets used in our experiments. For all the three datasets, the training time is less than 4 s

4 Conclusions

We presented a deep neural network framework for learning new representations in an online interactive CBIR setting. The experimental results show that it is possible to build CBIR systems that can dynamically learn the target from very limited user feedback. The system allows users to conduct searches for abstract concepts even though the system may not have been initially trained with abstract image classes. This aspect is also of high importance for symbiotic interactive systems, which can automatically detect what type of images the user might be looking for without the need on the part of the user to specify beforehand what image features would best satisfy their needs. We show that it is possible to produce near-instant image metrics with only a few training examples. Previous studies show that CNNs are able to abstract and discriminate beyond their original use. The descriptive value of the original features was not diminished by the small training set size used in our system, which is a promising step for using these in a CBIR setting.

The average duration of a search iteration with our pipeline is close to the 4 s required in interactive systems, and can be further reduced with more fine tuning of the system and improved hardware. In the future, we are planning to run more extensive simulation experiments as well as conduct extensive user studies to test the system for its applicability in various search scenarios. Additionally, decreasing the sampling size and parallelizing the framework with GPUs are the next steps in our system's development. The goal is to reduce the processing speed to below 3 s in a system that is able to converge to the target image in a user study within a reasonable number of iterations.

Acknowledgments. The work was supported by The Finnish Funding Agency for Innovation (projects Re:Know and D2I) and by Academy of Finland (project COIN).

References

1. Ahukorala, K., Medlar, A., Ilves, K., Glowacka, D.: Balancing exploration and exploitation: empirical parameterization of exploratory search systems. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, CIKM 2015, pp. 1703–1706. ACM, New York (2015)
2. Athukorala, K., Głowacka, D., Jacucci, G., Oulasvirta, A., Vreeken, J.: Is exploratory search different? A comparison of information search behavior for exploratory and lookup tasks. J. Assoc. Inf. Sci. Technol. **67**(11), 2635–2651 (2015)
3. Athukorala, K., Medlar, A., Oulasvirta, A., Jacucci, G., Glowacka, D.: Beyond relevance: adapting exploration/exploitation in information retrieval. In: Proceedings of the 21st International Conference on Intelligent User Interfaces, IUI 2016, pp. 359–369. ACM, New York (2016)
4. Brutlag, J.D., Hutchinson, H., Stone, M.: User preference and search engine latency. In: JSM Proceedings, Qualtiy and Productivity Research Section (2008)
5. Chechik, G., Sharma, V., Shalit, U., Bengio, S.: Large scale online learning of image similarity through ranking. J. Mach. Learn. Res. **11**, 1109–1135 (2010)
6. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: Proceedings of CVPR (2005)

7. Datta, R., Li, J., Wang, J.: Content-based image retrieval: approaches and trends of the new age. In: Multimedia Information Retrieval, pp. 253–262. ACM (2005)
8. Gao, X., Hoi, S.C., Zhang, Y., Wan, J., Li, J.: SOML: sparse online metric learning with application to image retrieval (2014)
9. Głowacka, D., Hore, S.: Balancing exploration-explotation in image retrieval. In: Proceedings of UMAP (2014)
10. Głowacka, D., Ruotsalo, T., Konuyshkova, K., Athukorala, K., Kaski, S., Jacucci, G.: Directing exploratory search: reinforcement learning from user interactions with keywords. In: Proceedings of the 2013 International Conference on Intelligent User Interfaces, IUI 2013, pp. 117–128. ACM, New York (2013)
11. Głowacka, D., Shawe-Taylor, J.: Content-based image retrieval with multinomial relevance feedback. In: Proceedings of ACML, pp. 111–125 (2010)
12. Gong, Y., Wang, L., Guo, R., Lazebnik, S.: Multi-scale orderless pooling of deep convolutional activation features. CoRR, abs/1403.1840 (2014)
13. Hore, S., Tyrvainen, L., Pyykko, J., Glowacka, D.: A reinforcement learning approach to query-less image retrieval. In: Jacucci, G., Gamberini, L., Freeman, J., Spagnolli, A. (eds.) Symbiotic 2014. LNCS, vol. 8820, pp. 121–126. Springer, Cham (2014). doi:10.1007/978-3-319-13500-7_10
14. Hu, J., Lu, J., Tan, Y.-P.: Discriminative deep metric learning for face verification in the wild. In: Proceedings of CVPR (2014)
15. Huiskes, M.J., Lew, M.S.: The MIR flickr retrieval evaluation. In Proceedings of MIR (2008)
16. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: convolutional architecture for fast feature embedding (2014). arXiv:1408.5093
17. Kangasrääsiö, A., Głowacka, D., Kaski, S.: Improving controllability and predictability of interactive recommendation interfaces for exploratory search. In: Proceedings of the 20th International Conference on Intelligent User Interfaces, IUI 2015, pp. 247–251. ACM, New York (2015)
18. Krizhevsky, A., Hinton, G.E.: Using very deep autoencoders for content-based image retrieval. In: Proceedings of ESANN (2011)
19. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
20. Lowe, D.G.: Object recognition from local scale-invariant features. In: ICCV, pp. 1150–1157 (1999)
21. Razavian, A.S., Azizpour, H., Sullivan, J., Carlsson, S.: CNN features off-the-shelf: an astounding baseline for recognition. CoRR, abs/1403.6382 (2014)
22. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C.: Fei-Fei., L.: Imagenet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (2014)
23. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: integrated recognition, localization and detection using convolutional networks. In: Proceedings of ICLR (2014)
24. Wan, J., Wang, D., Hoi, S.C.H., Wu, P., Zhu, J., Zhang, Y., Li, J.: Deep learning for content-based image retrieval: a comprehensive study. In: Proceedings of MM (2014)
25. Wan, J., Wu, P., Hoi, S.C.H., Zhao, P., Gao, X., Wang, D., Zhang, Y., Li, J.: Online learning to rank for content-based image retrieval. In: Proceedings of the 24th International Conference on Artificial Intelligence, IJCAI 2015, pp. 2284–2290. AAAI Press (2015)

26. Yang, J., Jiang, Y.-G., Hauptmann, A.G., Ngo, C.-W.: Evaluating bag-of-visual-words representations in scene classification. In: Multimedia, Information Retrieval, pp. 197–206 (2007)
27. Zhao, F., Huang, Y., Wang, L., Tan, T.: Deep Semantic ranking based hashing for multi-label image retrieval. ArXiv e-prints, January 2015
28. Zhou, X., Huang, T.: Relevance feedback in image retrieval: a comprehensive review. Multimedia Syst. **8**(6), 536–544 (2003)