# Fast re-ranking of visual search results by example selection

John Schavemaker[1], Martijn Spitters[1], Gijs Koot[1], and Maaike de Boer[12]

[1] TNO Technical Sciences, The Hague, The Netherlands
[2] Radboud University, Nijmegen, The Netherlands

**Abstract.** In this paper we present a simple, novel method to use state-of-the-art image concept detectors and publicly available image search engines to retrieve images for semantically more complex queries from local databases without re-indexing of the database. Our low-key, data-driven method for associative recognition of unknown, or more elaborate, concepts in images allows user selection of visual examples to tailor query results to the typical preferences of the user. The method is compared with a baseline approach using ConceptNet-based semantic expansion of the query phrase to known concepts, as set by the concepts of the image concept detectors. Using the output of the image concept detector as index for all images in the local image database, a quick nearest-neighbor matching scheme is presented that can match queries swiftly via concept output vectors. We show preliminary results for a number of query phrases followed by a general discussion.

**Keywords:** image retrieval, concept detectors, query expansion.

## 1 Introduction

Internet statistics show that visual information becomes more and more important online. All major social media applications, such as FaceBook and Twitter, allow sharing of photos and videos or are centered around photos, e.g., Instagram, Flickr, and Pinterest. Smart access to this ever expanding set of visual information has therefore become essential. One of the ways to provide smart access is through annotation of the material by adding tags. These tags can be used for search. Manually annotated tags, such as in Flickr, are not accurate because because they are often subjective. One way to automatically create tags is by concept detection based on deep-learning approaches such as in Google Image Captioning [16].

An issue with annotation is that people will expect different annotations of images depending on their interpretation of the world. For example, for the query 'dangerous animals' one may think that certain animals are dangerous while others think they are not. Traditional query-expansion techniques based on semantics may not be the solution for this issue since semantic databases are typically not contextual since they provide a general interpretation of the world.

The main contribution of this work is that we demonstrate a new, easy way to quickly retrieve results on a local image database by selecting visual examples as returned by popular web-based image search engines such as Bing. It provides an intuitive, visual interaction mode for the user to tailor and re-rank results to her/his preferences. Furthermore, it offers a simple way to get retrieval results for more complex queries even with concepts or adjectives for which the images are not annotated and indexed. This can be done without re-indexing of the database and without the use of semantic expansion of the query. We will illustrate that the use of image search as a means of 'visual query expansion' is an interesting alternative to semantic expansion solutions based on ConceptNet [14] allowing a fully functional image retrieval system for local databases that can deliver personalized answers to complex query phrases.

Our method is part of a larger application system, which is proposed in [11]. The application system is called GOOgle$^{TM}$ for Sensors (GOOSE) system and it is a general-purpose search engine conceived to enable any type of user to retrieve images and videos in real-time from multiple and heterogeneous sources and sensors. The proposed system especially focuses on cameras as sensors and aims at bridging the semantic gap between natural language queries that can be posed by a user and concepts that can be recognized by the concept detectors. The search engine allows users to pose natural language queries to retrieve corresponding images. User queries are interpreted using the Stanford Parser, semantic rules and the Linked Open Data source ConceptNet and further explained in [1]. The process of on-the-fly training of concept detectors is explained in [2]. This paper focuses only on the ranking and retrieval algorithm.

The outline of this paper is as follows. In the next section related work on the topic is discussed, Section 3 introduces our method for image retrieval based on examples. In Section 4 we show results of experiments we conducted, followed by conclusions and a discussion in Section 5.

## 2 Related Work

Semantic search in visual data often depends on pre-trained classifiers and object detectors for ranking the target data given the query. These pre-trained classifiers and object detectors are trained with annotations from various internet resources, such as image sharing platforms, e.g., Flickr [12], large-scale manually constructed image ontologies, e.g., ImageNet [5, 10] or public image search engines [6, 8].

With the explosive growth of digitally available visual data and countless possible labels of interest, the expensive process of annotating and training tailored detectors for unknown concepts does not seem sustainable. Several ways have been explored to automatically annotate images based on co-occurrence of visual and textual information on the internet [13, 15, 17, 18]. An example of a general-purpose large-scale system which learns new objects and relations from images is called NEIL [4]. This *Never Ending Image Learner* (NEIL) aims at developing the world's largest visual stuctured knowledge base with minimal human effort. NEIL queries Google Image Search to gather training examples for the objects, scenes and attributes in its ontology. The learned detectors and classifiers are subsequently applied to millions of images found on the web to learn relationships based on co-occurrence statistics.

Another relevant system is described in [3] and uses images sourced from Google to learn models for new objects on-the-fly. However, whereas their method actually computes descriptors for the retrieved images using well-known image encoding techniques like SIFT, and trains a linear SVM against a fixed set of negatives, we apply a much more basic method for *associative recognition* of unknown (or more elaborate) concepts in images. Therefore, our method can be considered as a low-key, scalable, data-driven way of retrieving images: we let our system look at examples of unknown concepts based on association with the concepts it knows.

Our approach is comparable to retraining strategies using one of the layers in a pretrained neural network, such as in [2]. Compared to using the abstract features in these layers, it has the advantage that the expansion is easily interpretable by the user, allowing the user to understand the search results and adjust the expansion, which can be very useful in real-life applications.

## 3 Method description

In this section we describe in more detail how our proposed image retrieval system works. This section is divided into two parts. In Section 3.1 we describe image indexing, and Section 3.2 goes into detail about our retrieval approach. Figure 1 shows a system overview.

### 3.1 Image representation and indexing

In order to retrieve images from a database, images need to be annotated and indexed. For that purpose we use the Python implementation of the Berkeley Caffe deep-learning framework [7] trained on the ILSVRC 2012 training set with 1000 image classes. For every image in the image database a $1 \times 1000$ concept support vector with detection scores for the different concepts is calculated and used as index key for future database retrieval. The support vector represents the support ($[0, 1]$) for every hypothesis that one of the $1,000$ concepts is presented in the image. Result of database indexing is an index that couples images to their corresponding support vectors.

### 3.2 Image database retrieval

Our approach for retrieving results for queries on the image database consists of the following steps:

1. Send natural language query as-is to a web-based image search engine such as Microsoft Bing, Google, or Yahoo by means of the API. In the experiments we use Bing because its API is easier to use in automatic scripts.

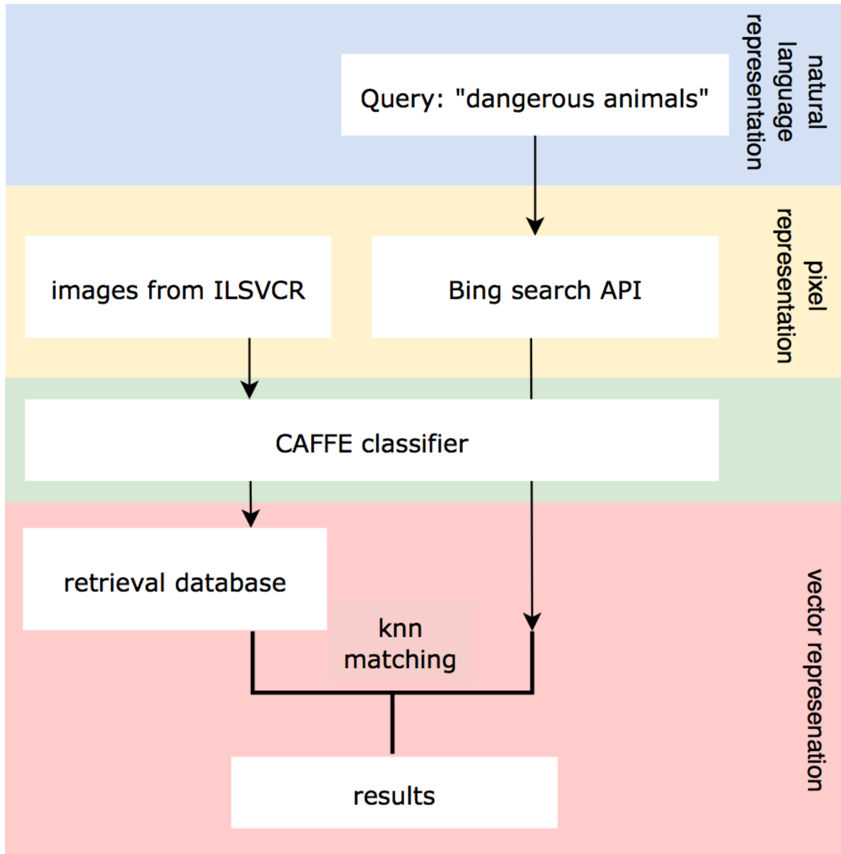


**Fig. 1.** System overview with in color marked the different representation layers: natural language, pixel, and concept vectors. The left path of the overview constitutes the image representation and indexing of Section 3.1, the right path is the retrieval path from Section 3.2 (steps 1 to 7).

2. Retrieve image search results by downloading the top $N$ results, in our experiments we use $N = 20$;
3. Let user interactively select the appropriate images from the downloaded search results (this step is optional);
4. Compute the concept support vector for every image in the user image selection;
5. Match the query set of concept support vectors with the local image database index using Euclidean nearest-neighbor (k-NN, with $k = 2$);
6. Rank matching results based on Euclidean matching distance;
7. Remove duplicate matching results stemming from different visual examples.

The user selection of visual examples may range from one to many images. More examples provide better coverage of variety in the query interpretation.

The K nearest-neighbor matching with the database index can be done brute force or with tree-based approximate nearest neighbor algorithms. The parameter K is set to $K = 2$ to make sure that not one example dominates the results by limiting results of one example to at most 2 matches. For example, for the query 'angry cats' in Figure 3, higher values of $K$ would probably lead to more similar results of Siamese cats as example #1, with a high support for the concept of Siamese Cat (0.97), may have many results close in distance.

Furthermore, steps $3 - 7$ can be repeated with alterations to user selection to quickly re-rank or renew search results. Note that only the nearest neighbor search in step 4 requires computing time that scales with the number of images in the database. This is an instance of the nearest neighbor problem, for which many optimizations are available, notably GPU implementations and approximation techniques [9]. With these algorithms, a neighbor search is still feasible within a second for tens of example images.

## 4 Experiments

In our experiments, we compare our 'visual query expansion' method with a 'textual query expansion' method. Our image retrieval system uses the web-based image search engine Bing to find the top 20 images relevant to the query and matches the support vectors for each of these images to the images in the database with the Euclidean nearest-neighbor distance metric. The results are displayed to the user and re-ranking is possible. An example of the Bing image search results on 'dangerous animals' is shown in Figure 2 and an example of the result of our visual query expansion method is in Figure 3. The 'textual query expansion' method uses ConceptNet to expand the query, as explained further in the next subsection.

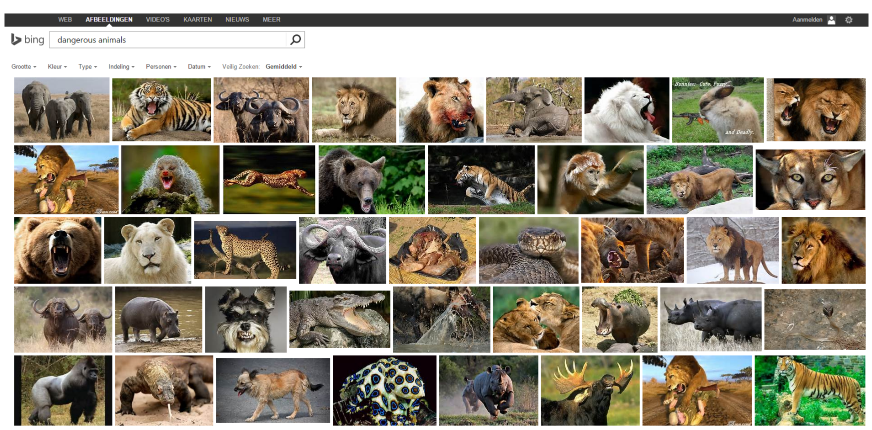


**Fig. 2.** Microsoft Bing image search results for query 'dangerous animals'.

In the experiments we use the following query phrases: ‘angry cats’, ‘beautiful dogs’, ‘cool old cars’, ‘dangerous animals’,‘healthy food’, and ‘ugly modern buildings’. These phrases are intentionally chosen because they emphasize subjectivity, while at the same time being a close enough generalization or specialization of one of the categories recognized by our instance of the Caffe classifier. For example, these categories include ‘snow leopard’ and ‘great white shark’, arguably related to the ‘dangerous animals’ query, but also some potentially ‘healthy foods’, such as ‘broccoli’ and ‘bananas’.

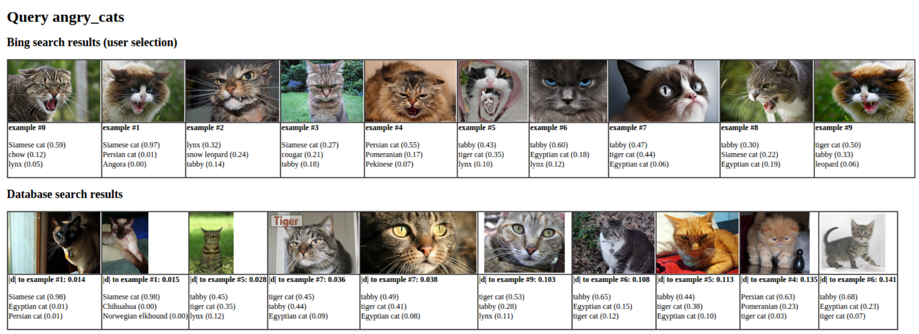


**Fig. 3.** Angry cats query: the first row shows the top 10 examples of the Bing search results. Every example shows the top 3 firing concepts of the concept detector. The second row shows the sorted database search results with their matching distance to one of the 10 examples. In addition the top 3 of concepts of the index is listed.

Evaluation is done on the evaluation set (named ‘val’) of ILSVRC 2012. This set is used as an example local image database. This set consists of $50,000$ images with different types of objects.

### 4.1 Textual Query Expansion

ConceptNet 5.3 [14] is used to find relevant concept detectors for the query. ConceptNet is automatically accessed through the REST API. First, the spaces between the words are replaced with a comma. Second, the association value between a concept detector label and the query is captured using

```
"http://conceptnet5.media.mit.edu/data/5.3/assoc/c/en/"
+ query + "?filter=/c/en/"
+ detectorLabel + "\&limit=1"
```

The association value between the query and the concept detector label is used as an indication of the relevance of the concept detector for the query. If the concept detector label could not be found in ConceptNet, the more general word, which is the last word in the label (*shark* in *white_shark*), is used to assign an association value. The association value between the query and the concept detector label is used as an indication of the relevance of the concept detector for the query. The captured concepts are converted to concept support vectors by creating a $1 \times 1000$ vector where the value of the corresponding concept is set to 1.0. In the Appendix, Table 1 lists for each query the concepts in the first expansion.

### 4.2 Results

In the Appendix, Figures 4 and 5 present database retrieval results for all example queries using our proposed method and results using textual query expansion.

For the 'angry cats' query, the visual query expansion shows results with only cats that mainly include Siamese, Tabby, and Tiger cats, just as the user-selected examples do. The textual query expansion results show mostly cats as well with the exception of some 'Madagascar cats' and cougars that both do not match the domestic cats from the user-selected examples. In general, the cats do not look that angry with the exception of the last cougar in the textual query expansion results.

A similar picture is shown for the query 'beautiful dogs'. Again, visual query expansion results are a good match and cover the user-selected examples of Samoyed, Pomeranian, Golden retriever and Maltese dogs. The textual query expansion results start with African hunting dogs, not resembling the domestic dogs in the user-selected examples, but include Maltese dogs as well.

The results for query 'cool old cars' show some more discrepancies between our proposed method and the textual query expansion. The concept 'car' is too ambiguous for expansion: the top results include 'freight car' and 'passenger car' or 'train' as well as car parts like car mirrors. The search results of our method show better results: mainly cool old cars with the exception of two modern race cars, due to example 'racer', and two modern convertibles, due to example 'convertible'.

The results of the textual query expansion of 'dangerous animals' show some errors including an amphibious vehicle, house finch and sea slug (see also Table 1). The results of our method capture better the concept of dangerous animals as selected in the examples including tiger and three of the the African big five, i.e., lion, buffalo and elephant.

The query 'healthy food' is also hard to expand correctly: the textual expansion includes food but pizza, bagel and meat loaf are in general not considered healthy. Our method results show for the first 6 ranked results good result that mimic the fruit and vegetables from the examples including the photo composition. The feather boa results are in error of course; these are examples that in an interactive session can be omitted by the user to re-rank better search results.

The last query 'ugly modern buildings' is the hardest one for both methods. The results in Figure 5 show Bing search examples that have low concept scores, except for library and mobile home, for most selected example and as a result some mediocre results for our method. Textual query expansions is mainly into household appliances and computer hardware and are completely off.

## 5 Conclusions and discussion

In this paper, we presented a novel and easy way to quickly access image databases by means of indexing images by concept detectors and finding good visual examples by querying web-based image search for examples. In this way the user has easy control on 'query expansion'. We have shown preliminary results that look promising for our proposed method. The experiments, however, are limited to a few queries and do not include a full quantitative evaluation of the method. One must also note that our queries have good coverage in the 1000 concepts of ILSVRC that include among others many cat and dog breeds. Queries that fall outside these pretrained concepts will probably be handled less well. The choice for Microsoft Bing as our primary search engine for the visual examples is also a pragmatic one (the search API is easier to script), other search engines may behave differently and are not investigated in this paper.

In future work, a full user study with an appropriate evaluation is necessary. An interesting point for the future would also be to provide both visual query expansion and textual query expansion data to the user and find out in which type of queries the visual method is preferred over the textual method and the other way around. Furthermore, it would be interesting to investigate whether one of the hidden layers of the pretrained Caffe ILSVRC neural network can be used to index images instead of the $1 \times 1000$ concept support vector. The hidden layer may contain more information that is relevant for the subjective part of the query. Retraining strategies such as [2] also use hidden layers as input.

## 6 Acknowledgments

This research was performed in the GOOSE project, which is jointly funded by the MIST research program of the Dutch Ministry of Defense and the AMSN enabling technology program.

## References

1. de Boer, M.H.T., Daniele, L., Brandt, P., Sappelli, M.: Applying semantic reasoning in image retrieval. In: ALLDATA 2015, The First International Conference on Big Data, Small Data, Linked Data and Open Data. pp. 69–74. IARIA (2015)
2. Bouma, H., Eendebak, P.T., Schutte, K., Azzopardi, G., Burghouts, G.J.: Incremental concept learning with few training examples and hierarchical classification. In: Proc. SPIE. vol. 9652 (2015)
3. Chatfield, K., Zisserman, A.: Visor: Towards on-the-fly large-scale object category retrieval. In: Computer Vision–ACCV 2012, pp. 432–446. Springer (2013)
4. Chen, X., Shrivastava, A., Gupta, A.: Neil: Extracting visual knowledge from web data. In: Computer Vision (ICCV), 2013 IEEE International Conference on. pp. 1409–1416. IEEE (2013)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR09 (2009)

6. Fergus, R., Fei-Fei, L., Perona, P., Zisserman, A.: Learning object categories from google's image search. In: Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on. vol. 2, pp. 1816–1823. IEEE (2005)
7. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093 (2014)
8. Li, L.J., Fei-Fei, L.: Optimol: automatic online picture collection via incremental model learning. International journal of computer vision 88(2), 147–168 (2010)
9. Muja, M., Lowe, D.G.: Flann, fast library for approximate nearest neighbors (2009)
10. 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 (2014)
11. Schutte, K., Bouma, H., Schavemaker, J., Daniele, L., Sappelli, M., Koot, G., Eendebak, P., Azzopardi, G., Spitters, M., de Boer, M., Brandt, P.: Interactive detection of incrementally learned concepts in images with ranking and semantic query interpretation. In: Proc. of 13th International Workshop on Content-Based Multimedia Indexing (CBMI) (2015)
12. Shi, Z., Yang, Y., Hospedales, T.M., Xiang, T.: Weakly supervised learning of objects, attributes and their associations. In: Computer Vision–ECCV 2014, pp. 472–487. Springer (2014)
13. Snoek, C.G.M., Worring, M., Koelma, D.C., Arnold W. M. Smeulders, M.: A learned lexicon-driven paradigm for interactive video retrieval. IEEE Transactions on Multimedia 9(2) (2007)
14. Speer, R., Havasi, C.: Representing general relational knowledge in ConceptNet 5. In: LREC. pp. 3679–3686 (2012)
15. Torralba, A., Fergus, R., Freeman, W.T.: 80 million tiny images: A large data set for nonparametric object and scene recognition. Pattern Analysis and Machine Intelligence, IEEE Transactions on 30(11), 1958–1970 (2008)
16. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. CoRR abs/1411.4555 (2014), http://arxiv.org/abs/1411.4555
17. Wang, X.J., Zhang, L., Jing, F., Ma, W.Y.: Annosearch: Image auto-annotation by search. In: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. vol. 2, pp. 1483–1490. IEEE (2006)
18. Zhang, R., Zhang, Z., Li, M., Ma, W.Y., Zhang, H.J.: A probabilistic semantic model for image annotation and multimodal image retrieval. In: Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on. vol. 1, pp. 846–851. IEEE (2005)


## Appendix: query results

In this appendix the results for the textual query expansion from Section 4.1 is presented in Table 1. The ranking results for the six different queries from Section 4 are presented in Figures 4 and 5. These results include results for our proposed method and results using textual query expansion.

**Table 1.** Textual query expansion, for every query the top 10 concepts are shown with associated weights.

| | |
|---|---|
| angry cats | Egyptian cat (1.0), Persian cat (1.0), Madagascar cat (1.0), Siamese cat (1.0), tiger cat (1.0), tabby (0.97), lynx (0.94), claw (0.89), cougar (0.81), jaguar (0.74) |
| beautiful dogs | Eskimo dog (1.0), Maltese dog (1.0), Bernese mountain dog (1.0), Greater Swiss Mountain dog (1.0), African hunting dog (1.0), toy poodle (0.98), standard poodle (0.98), miniature poodle (0.98), Walker hound (0.98), Afghan hound (0.98) |
| cool old cars | passenger car (1.0), sports car (1.0), freight car (1.0), car mirror (0.93), limousine (0.93), minivan (0.93), beach wagon (0.91), moving van (0.88), police van (0.88), fire engine (0.88) |
| dangerous animals | amphibious vehicle (0.94), great grey owl (0.93), sea slug (0.92), house finch (0.91), hippopotamus (0.91), red-backed sandpiper (0.89), little blue heron (0.89), echidna (0.87), gorilla (0.87), sulphur-crested cockatoo (0.87) |
| healthy food | pizza (0.86), bagel (0.75) meat loaf (0.74), French loaf (0.74), eggnog (0.74), corn (0.68), refrigerator (0.65), bakery (0.65), cheeseburger (0.64), chocolate sauce (0.63) |
| ugly modern buildings | monitor (0.60), desktop computer (0.51), hand-held computer (0.51), photocopier (0.51), sewing machine (0.50), cash machine (0.50), vending machine (0.50), electric fan (0.50), laptop (0.49), joystick (0.47) |

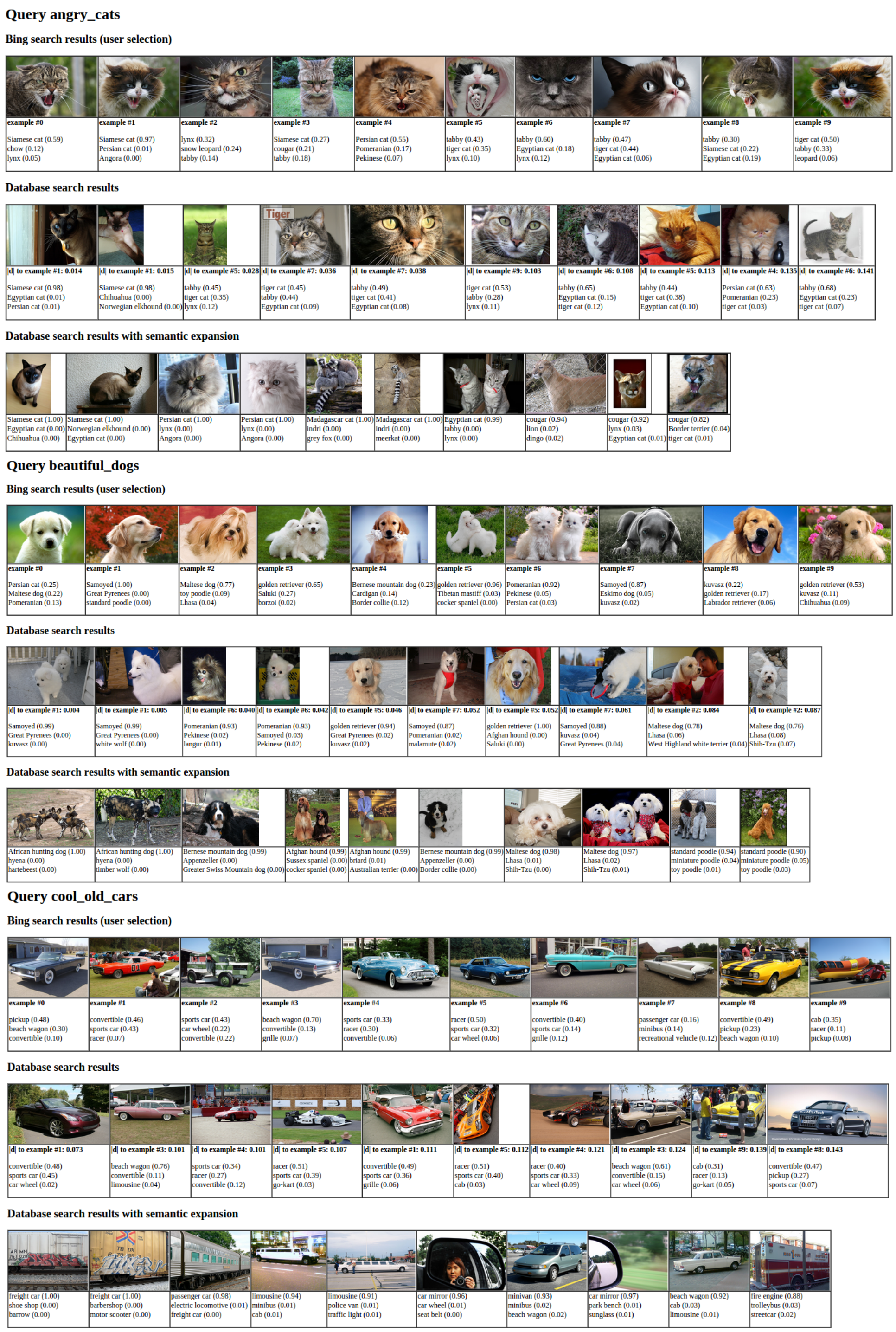


**Fig. 4.** From top to bottom: retrieval results on ILSVRC 2012 val set for queries ‘angry cats’, ‘beautiful dogs’, and ‘cool old cars ’. Top row shows the Bing search results (as selected by the user). Middle row shows the top 10 ranked results for the proposed method. Bottom row shows the top 10 ranked results using textual query expansion.

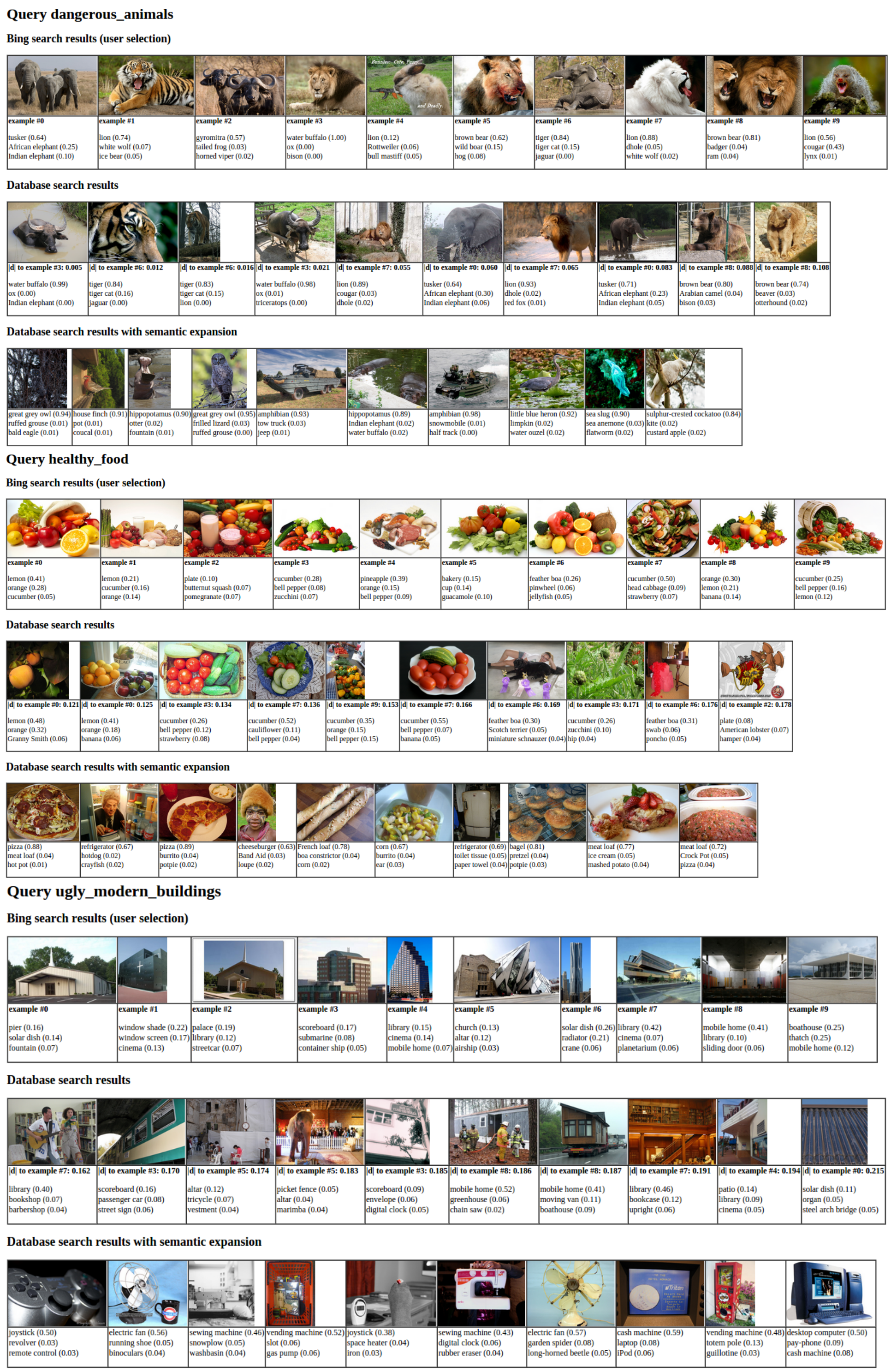


**Fig. 5.** Retrieval results on ILSVRC 2012 val set for query 'dangerous animals', 'healthy food', and 'ugly modern buildings'. Top row shows the Bing search results (as selected by the user). Middle row shows the top 10 ranked results for the proposed method. Bottom row shows the top 10 ranked results using textual query expansion.