Multi-modal Image Retrieval for Search-based Image Annotation with RF

Petra Budikova
Masaryk University
Brno, Czech Republic
budikova@fi.muni.cz

Michal Batko
Masaryk University
Brno, Czech Republic
batko@fi.muni.cz

Pavel Zezula
Masaryk University
Brno, Czech Republic
zezula@fi.muni.cz

***Abstract*—Search-based annotation methods can be used for proposing descriptive keywords to users who need to annotate images e.g. in image stock databases. From the annotation output, users select keywords which they want to assign to the given image. The selected keywords can serve as a relevance feedback for additional annotation refinement. In this paper, we study the possibilities of exploiting the annotation relevance feedback, which is a novel problem that has not been systematically addressed yet. In particular, we focus on the subtask of utilizing the feedback for the retrieval of related annotated images that are subsequently used for mining of candidate keywords. We select three multi-modal search techniques that can be applied to this problem, implement them within a state-of-the-art search-based annotation system, and experimentally evaluate their usefulness for annotation quality improvement.**

***Index Terms*—image annotation, relevance feedback, multi-modal image retrieval**

I. Introduction

Automatic image annotation is an important tool for keyword-based retrieval, which remains the most natural way of accessing multimedia information. Depending on the application that will be using the annotations, the descriptive keywords may be chosen from a narrow or wide set of eligible keywords (usually denoted as the *target vocabulary* of the annotation task). Narrow target vocabularies are typical for domain-specific applications such as art or medical image retrieval, whereas in general web image search the target vocabulary contains the whole vocabulary of a given language. Clearly, the difficulty level of the annotation task grows with the size of the target vocabulary.

In recent years, we could observe big advances of classification methods based on convolutional neural networks, which achieve very good results on some types of annotation tasks [1]. However, the applicability of these methods is limited to medium-sized target vocabularies, for which reliable training data can be obtained. For large vocabularies, a complementary approach of search-based annotation was proposed, which exploits big amounts of unreliable web data to estimate the most probable descriptive keywords [2], [3].

The basic principle of the search-based image annotation is depicted in Fig. 1. We attempt to annotate an unlabeled image by propagating labels of similar photos that are available in web image collections or social networks. In the time of query execution, a content-based search is initiated to retrieve images that are visually similar to the picture being annotated, and the textual metadata of the resulting images is used to form the annotation. The underlying assumption is that a significant portion of visually similar photos should be semantically related to the image that is being analyzed. The use of web data instead of dedicated training collections significantly lowers the barriers of building an annotation system. The search-based annotation also needs no learning phase and scales well to large vocabularies. Although the current precision of search-based annotation methods is not satisfactory for fully automated use, they can be used as a tag-hinting tool in applications that require broad-vocabulary annotations.

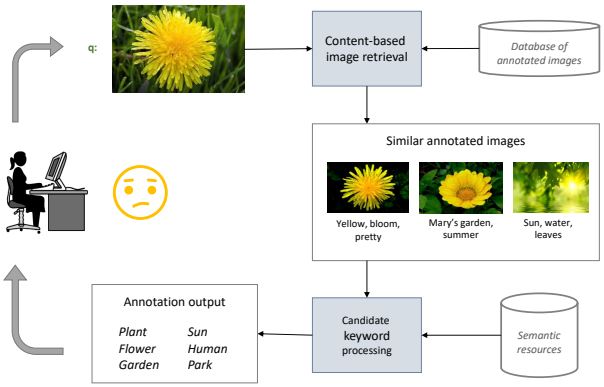

Fig. 1. Search-based annotation without RF.

In this paper, we consider a particular use case of tag-hinting for image stock sites: whenever a user wants to upload his or her photos to the image stock, he or she has to accompany the images with descriptive keywords that are used for keyword searching in the stock. To maximize the findability of their photos, the users are motivated to provide a lot of descriptive keywords. Today, the keywords are mostly entered manually, which is time- and effort-demanding. To minimize the manual work, the search-based annotation can be employed to get a set of candidate keywords from which the user will choose the relevant ones (Fig. 1).

By selecting relevant keywords for the image stock, the user also gives a relevance feedback to the annotation system. A

This work was supported by the Czech national research project GA16-18889S.

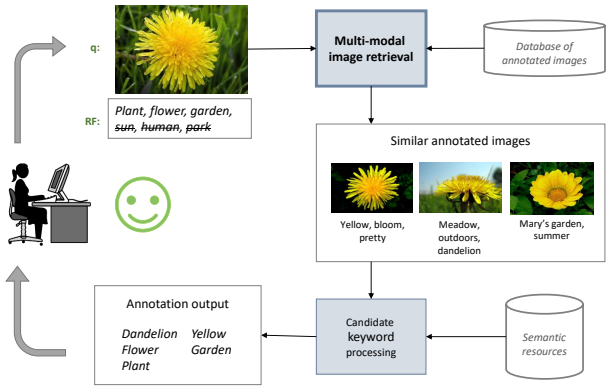

Fig. 2. Search-based annotation with RF; the feedback is exploited in the image retrieval phase.

logical next step towards providing richer annotations is to extend the annotation process into multiple iterations, where the user's choices of relevant keywords serve as additional input for the next annotation cycle. This explicit relevance feedback allows the system to take into account the user's individual needs and preferences. In the future, a similar kind of additional information can be also provided by pseudo-RF or dedicated classifiers for selected target keywords.

As depicted in Fig. 1, a search-based annotation process consists of two distinct phases: 1) retrieval of similar images, and 2) processing of similar images' descriptions. Relevance feedback can be exploited in any (or both) of these phases. In this paper, we focus on exploiting annotation RF in the image search phase. Using information about (ir)relevance of keywords suggested as annotations in the first annotation round, we attempt to select a set of images that are similar to the query image both visually and semantically. If we succeed, the similar images should provide more relevant information to the second annotation phase, which in turn should produce more precise annotation keywords (Fig. 2).

Searching for images that are visually and semantically similar to a text-and-image query is actually an instance of the multi-modal search problem, which has been addressed by many previous works. Therefore, we can capitalize on existing research in this area and try to utilize methods that have shown promising results. At the same time, we need to keep in mind that image searching for interactive annotation has some specifics. In particular, it is not important whether all returned images are relevant because the user never sees them; as long as the majority of images is relevant, the label transfer algorithm should give good results. On the other hand, the retrieval efficiency is very important since we need to search large data collections while the user is waiting.

The specific contributions of the paper are the following: we select three promising algorithms for text-and-visual image retrieval and implement them within the MUFIN Image Annotation tool, a state-of-the-art search-based annotation system [4]. In particular, we employ standard CBIR with text-based re-ranking of results, the complementary approach of text-search with content-based re-ranking, and finally a very recent technique that employs neural networks to map keywords into the space of visual descriptors. Apart from comparing these three orthogonal approaches, we study in detail various parameters that influence the RF processing.

The rest of the paper is organized as follows. First, we survey existing works on image annotation with RF and explore several related fields of multimedia information retrieval. Next, we formalize the task of image annotation with relevance feedback and introduce some basic terminology. In Section IV, we describe three orthogonal methods of multi-modal image retrieval that can be used in search-based image annotation. In Section V, we evaluate the effectiveness and efficiency of these methods on real-world data and examine the influence of different parameters. In the last section, we summarize our findings and outline possible future work.

II. Related Work

To the best of our knowledge, RF for search-based image annotation has not been systematically studied yet. The few existing works on this topic [5], [6] focus on exploiting pseudo-RF information – they assume that the top-ranking keywords produced by a given annotation method are likely to be relevant and use this assumption as additional input for annotation refinement. The pseudo-RF information is exploited in both annotation phases and the influence of various RF-processing parameters is not discussed. Instead, these studies focus mainly on estimating the probability that a given top-ranking keyword is indeed relevant.

The possibility of enhancing search results using RF has been studied extensively in other IR areas including text search [7] or content-based image search [8], [9]. These works provide us with valuable insights into the general trends of RF collecting and processing, but are not directly applicable to our problem. In text (image) retrieval, the feedback information is of the same data type as the query and the main challenge lies in creating a new similarity model that combines the multiple positive/negative examples provided as feedback. In image annotation, however, there are two different modalities involved: the query is an image, but the annotation output and the RF information are textual. This issues new challenges to the RF processing.

Exploiting annotation RF for the retrieval of semantically relevant similar images is more closely related to multi-modal image searching that is often applied on collections of images with textual metadata. A multi-modal query (provided by user or via a pseudo-RF mechanism) can be processed in a number of different ways [10], [11]; highly popular are late fusion approaches that implement the search as a two-phase process where one modality is used to retrieve a set of candidate objects and the other for re-ranking. This allows fast and sufficiently precise searching without need to implement any specialized indexing structures. In text-and-visual searching, the text modality is typically used for the selection of candidates [12] but in some situations, it may be more advantageous to begin with the visual modality [13].

Another similar task is the cross-modality searching where the query is issued in a different modality than is contained in the database to be searched; for instance, a user may want to search a database of unannotated images using a text query. To facilitate this, it is necessary to transform the query and/or database objects into a common representation. Very recently, it has been shown that convolutional neural networks can be trained to transform a query provided in one modality into the domain of a different modality. In particular, the neural network described in [14] allows to transform text queries into the domain of visual descriptors.

III. Formal Problem Definition

An annotation task is defined by a binary query image q and a target vocabulary V of eligible keywords. For the tag-hinting application, we assume that V comprises the whole English vocabulary. As elaborated in [4], we find it most fitting to model the solution as a function $f_A : Image \times Keyword \rightarrow [0; 1]$, which for each keyword $c \in V$ computes the probability that c is relevant for q. The n keywords with the highest probability are shown to the user as the annotation result.

After the *initial annotation* of the input image, users can provide relevance judgements to some/all of the suggested keywords. These keywords and their relevance scores form the *annotation RF (ARF)*, which is passed as an additional input to the *feedback loop* of the annotation method. In this paper, we adopt a very general model where the relevance score can be any real value between 0 (irrelevant) and 1 (highly relevant). Formally, $ARF = \{(c, rel_c)|c \in V, rel_c \in [0; 1]\}$. We are aware that this level of detail is too high for any real user-interaction scenario but in experiments a detailed simulated RF allows us to gain better insight into the behavior of RF processing methods. In the following text, we will be using the term *negative text queries* for keywords with relevance score 0, keywords with non-zero scores will be denoted as *positive text queries*. The size of the annotation RF is not set, because we are interested in finding out the influence of RF size on result quality. The original query image q together with the ARF form the *extended annotation query* Q.

As discussed earlier, in search-based annotation there are two basic phases of the annotation process: the retrieval of similar images from a suitable database D of well-annotated images, and the processing of descriptions of similar images. The intermediate result of the first phase is a set of annotated images from D, which will be denoted as *query neighbors* (QN). The query neighbors are submitted to the second annotation phase that performs semantic analysis of the neighbors' descriptions and determines the final annotation. Let k^{QN} be the number of the query neighbors; k^{QN} is unknown to the user and subject to parameter tuning.

In this paper, we focus on RF processing in the first annotation phase. Using the query image and the annotation RF, we attempt to select a set of query neighbors that are similar to the query image both visually and semantically. The quality of the query neighbors cannot be evaluated directly since it is a hidden component of the annotation process that has no clear information value for the user. Instead, we need to consider the final annotation result after the second, unchanged annotation step is performed. The task we want to solve can be thus formulated as follows: using the extended annotation query Q, select query neighbors such that the quality of annotations produced by the second annotation phase is improved.

IV. Our Approach

In this section, we focus on the retrieval of query neighbors in the feedback loop of a search-based annotation system. To fully exploit the information available in ARF, we need to replace the content-based retrieval module of the annotation tool by a multi-modal retrieval module, as illustrated in Figures 1 and 2.

As discussed in Section II, there are several possible ways to combine text and image clues in image retrieval. Until recently, the most common approach was the asymmetric late fusion, when one modality (text or image) is used to retrieve a set of candidate images and the other modality is utilized for re-ranking of the candidates. In the last years, convolutional neural networks have been developed that allow to map the domain of one modality onto the domain of another one. In the following sections, we propose three techniques that apply these ideas to annotation RF processing.

A. CBIR with text re-ranking

In this method, the visual image similarity serves as the primary modality which is utilized to retrieve a set of candidate images $Cand^{Vis}$ from database D. The textual similarity between image descriptions and the ARF is only evaluated for objects in $Cand^{Vis}$ to determine a new ranking of the candidate objects. In particular, the final ranking of $Cand^{Vis}$ is determined by a combined object score, which is computed as the average of three normalized partial object scores provided by (1) the visual similarity to the query image q, (2) the text similarity to the positive text queries, and (3) the text dissimilarity to the negative text queries. Finally, the k^{QN} top-ranking objects are selected as the query neighbors and passed to the second annotation phase.

The size of the candidate set, denoted as k^{Cand}, is an important parameter of any re-ranking method. Clearly, there need to be at least k^{QN} objects in $Cand^{Vis}$ but larger candidate sets are typically used. Large candidate sets allow to compare more objects with both modalities, but the processing costs of the re-ranking phase grow linearly with the candidate set size. Also, the retrieval of the larger candidate set is more expensive, although with efficient CBIR techniques the costs grow sub-linearly with respect to the candidate set size. The effects of the k^{Cand} on both annotation quality and processing costs will again be studied in the experimental section.

From the efficiency point of view, it is worth noticing that a content-based retrieval of images visually similar to the query image q is evaluated in both the initial annotation and the feedback loop. The only difference is the number of images required – in the initial annotation, only k^{QN} objects are needed, whereas for the re-ranking in the feedback loop more

images are usually considered. To avoid repeated evaluation of the same visual query, it is possible to retrieve k^{Cand} similar objects during the initial annotation and keep them cached for the feedback loop(s). This way, the feedback processing can be quite efficient, with no need to access the whole database D in the feedback loop. On the other hand, the success of both the initial annotation and the feedback loop is highly dependent on the quality of the visual similarity evaluation, which is the main weakness of this method.

B. Text search with CBIR re-ranking

In the second method, we again follow the principle of mono-modal candidate set retrieval and multi-modal re-ranking. However, the primary modality is now the text similarity between the ARF and the text queries, which is utilized to retrieve a candidate set $Cand^{Text}$. The candidate set is again re-ranked by a combination of three partial rankings – by visual similarity to the query image q, by text similarity to the positive text queries, and by text dissimilarity to the negative text queries. The parameters of the feedback loop processing are the same as in the previous method and will be analyzed in the experimental section.

In this method, different modalities are utilized for candidate set retrieval in the initial annotation and the feedback loop. The caching of candidate objects from initial annotation therefore cannot be used, however the text retrieval is very efficient so this is not an important issue. By using the text search for candidate set identification, we can access semantically relevant objects that could be missed by content-based queries. On the other hand, images with relevant content but without the specific keywords from ARF cannot be included in $Cand^{Text}$.

C. NN-based transformation of keyword feedback into the space of visual descriptors

Our last method was inspired by recent advances in convolutional neural networks (CNN), in particular the work [14] where a CNN was trained to transform user-provided text queries into the domain of visual descriptors. Using such transformation, it is possible to submit a text query to standard CBIR and search for images that are semantically similar to the text query but may not be annotated by exactly the same words. Intuitively, the same approach could also work for text queries derived from annotation feedback.

The authors of [14] used the original AlexNet [1] model as the base for the visual descriptor. Since our CBIR uses the later VGG [15] model, we have repeated the process of the CNN learning using the Profiset [16] collection as the source of annotated images for the training. In particular, a random subset of 1 million images with at least ten-words annotation was used. Since the Profiset images are described by keywords without order, we have used the bag-of-words variant of the approach. We will denote the descriptor produced by the retrained CNN as Text2Vis.

The adjusted model allows us to transform a set of keywords into a visual descriptor. The extended annotation query can be thus replaced by three visual descriptors from the same domain: the original visual query descriptor extracted in a standard way, and two Text2Vis descriptors derived from positive and negative text queries, respectively. These descriptors can be utilized in two modes to select the query neighbors for annotation mining. In the *re-ranking mode*, a set of candidate objects is again retrieved, using either the original query image descriptor or the Text2Vec descriptor of the positive RF keywords as the query. The candidates are subsequently re-ranked by a combination of (1) the similarity to the visual query descriptor, (2) the similarity to the Text2Vis descriptor of the positive text queries, and (3) the dissimilarity to the Text2Vis descriptor of the negative text queries. In the *direct mode*, we take advantage of the fact that the visual and Text2Vec descriptors are vectors from the same space, and try to combine them into a compound vector which could be used to retrieve the query neighbors directly. Since such combinations have not been previously studied, we experiment with several primitive combination methods, in particular average, minimum, and maximum. For the construction of the compound query vector, only the positive examples are used, i.e. the original visual descriptor and the Text2Vis descriptor of the positive text queries.

The theoretical strength of the transformation into the visual descriptor domain is the possibility to retrieve semantically related objects that are not annotated by matching keywords. However, there are challenges that have not been studied yet. The combination of the descriptors computed by the CNN is an open question, since the vectors do not represent defined properties of the image but rather some internal knowledge of the CNN which is a "black box". Moreover, one of our descriptors represents a negative example derived from a set of non-relevant keywords. This set is likely to be diverse and have different properties than the relevant query annotation that was considered in the CNN training.

V. Experimental Evaluation

To determine the usefulness of the proposed RF processing methods, we performed a thorough experimental evaluation with real-world data. In this section, we first introduce the infrastructure and data used in experiments, and specify the evaluation methodology. Then, we present the most interesting results and discuss them.

A. Implementation and Test Data

To be able to measure the quality of the proposed methods, we implemented them within a state-of-the-art image annotation system and obtained real-world annotation queries from a popular image stock site.

1) Implementation: The state-of-the-art MUFIN Image Annotation system [4] is used as the baseline in our experiments. The implementation was enriched by a new module for multi-modal image retrieval that is exploited in the feedback loop of the annotation process. As discussed earlier, all visual similarity evaluations utilize the cutting-edge VGG [15] descriptors. The content-based image retrieval engine exploits the PPP-Codes technique [17], which allows efficient evaluation

of kNN queries. The technique was implemented in Java using the Metric Similarity Search Implementation Framework (MESSIF) [18]. Text-retrieval utilizes the Lucene search engine [19] with standard cosine distance for text similarity evaluation. For the semantic analysis of the query neighbors' descriptions (the second phase of the annotation process), the ConceptRank technique [4] is employed.

All the experiments were conducted on a single machine with 8 Intel Xeon cores, 32GB RAM, and two 160GB SSD disks where all the data were stored. The only exception is the Text2Vec transformation, for which another machine with Quadro K1200 GPU card was used.

2) Data Collection: The search-based annotation paradigm relies on extracting information from large amounts of web data. Therefore, we need a suitable database D of annotated images from which the visual neighbors can be selected. In our experiments, we used the Profiset data – a collection of 20 million photos with rich keyword annotations that were downloaded from the Profimedia image-stock site[1] and are available for research purposes [16]. As detailed in Section IV, a subset of the Profiset collection was also used for the training of the CNN that produces the VGG descriptors.

3) Queries and Ground Truth: From the Profiset collection, we selected 160 images as test queries. Out of these, 80 photos were selected from Promedia logs of popular queries, another 80 were chosen randomly from images sold in a two-year period. Each query needs to be described by 30 relevant keywords. The query images were removed from the Profiset collection, so there is no overlap between the test queries and the annotated images that can be retrieved as query neighbors.

To evaluate annotation tasks with unlimited vocabularies, we should ideally use a ground truth of all English keywords relevant for a given image. Unfortunately, it is not feasible to collect such a ground truth, since there may be literally a thousand words describing each picture. However, during our previous research of image annotation we had organized manual relevance assessments for the same set of query images and large sets of possible descriptive keywords suggested by diverse annotation methods [4]. As a result, we now have a *partial ground truth* for the 160 queries, where for each image there are 200-300 keywords with multiple relevance assessments provided by different people. Each relevance assessment is a value from the interval [0;1] with the same semantics as we defined for the annotation RF (0 – irrelevant; 1 – highly relevant). By averaging the relevance assessments, we obtain a relevance score for each query-keyword pair. This partial GT allows us to compute lower- and upper-bounds on the precision of newly proposed annotation methods – for lower bounds (lb), we simply assume that all keywords with unknown relevance are irrelevant, while for upper bounds (ub) we assume them to be relevant.

4) Simulating Relevance Feedback: In our experiments, we do not collect the relevance feedback from real users but simulate it using our ground truth assessments. Simulating the ARF not only saves the time and effort needed to collect the feedback but also allows us to experiment with various RF properties, such as the number of evaluated keywords.

[1]http://www.profimedia.com

TABLE I
OVERVIEW OF TESTED PARAMETERS

parameter	tested values
ARF properties	
# of positive RF keywords	1, 2, 3, 5, 7, 10, all
# of negative RF keywords	0, 1, 2, 3, 5, 7, 10, all
selection of keywords for RF	FIRST, LAST, RANDOM, ALL
RF information type	binary, multivalued
RF processing parameters	
candidate set size (k^{Cand})	100, 200, 500, 1000, 2000
# of query neighbors (k^{QN})	50, 70, 100, 150, 200

In each experiment, the systems first computes the initial (fully automatic) annotation. We have a complete ground truth for the initial annotations, so each initial keyword has a relevance score. We consider keywords with relevance score at least 0.5 eligible for positive feedback, whereas keywords with lower scores can be selected as negative examples. We use four methods of selecting a given number of relevant/irrelevant keywords from the initial annotation, denoted as FIRST, LAST, RANDOM, and ALL. The FIRST method takes the given number of (ir)relevant keywords from the beginning of the initial annotation, while the LAST method searches the initial annotation from the end. The FIRST method thus gives feedback about the keywords that were proposed as the most probable in the initial annotation, wheres the LAST method provides information about keywords that the system was less certain about, which may describe some smaller detail or a background information, or use an uncommon terminology. As the third option, we consider a random selection of the feedback keywords. Finally, the ALL method takes all relevant keywords from the initial annotation.

B. Evaluation Methodology

In this section, we formulate the particular objectives of our evaluation and discuss suitable evaluation measures.

1) Objectives of Experimental Evaluation: The quality of annotation with RF intuitively depends on two factors: 1) the quality of the feedback information itself, and 2) the quality of the feedback processing. In the experiments, we want to compare the three proposed ARF processing methods and determine their suitability for different types of the ARF. Furthermore, we are interested in the influence of different parameters of the ARF processing methods that were mentioned in Section IV. We will also focus on the efficiency of individual solutions, which is vital for interactive annotation.

In Table I, we provide an overview of parameters that are common to all tested methods, and the values that were used in experiments. The parameters in the first group determine the properties of the simulated feedback. We experiment with different numbers of positive and negative feedback words, and compare four methods of choosing them. We also consider binary (relevant/irrelevant) RF values as well as the multivalued relevance assessments discussed in Section III. The second

part of Table I deals with two important parameters of ARF processing methods that follow the search-and-rerank scheme: the number of images retrieved by the first coarse search, and the number of top-ranking images selected after the re-ranking.

2) Evaluation Measures: As stated earlier, we are interested in both effectiveness and efficiency of our methods. For efficiency, we use the straightforward measure of average query processing time. With effectiveness, the situation is a bit more complicated. As discussed e.g. in [7], standard accuracy measures such as precision can be misleading because some of the RF loop results are known in advance to be relevant and artificially increase the precision. This makes it difficult to compare an initial result to a result after the RF loop, or two experiments where the feedback sizes were different. On the other hand, the final precision remains the most important quality for the user. Since there is no generally accepted quality measure for RF processing, we use a combination of several commonly used metrics. In particular, we evaluate the mean precision of the whole annotation (MP), the mean precision of non-RF keywords only (MP-new), and the average number of relevant keywords that were lost/gained as compared to the initial result (RW-lost, RW-gained). For all these measures, we compute the lower- and upper-bounds using our partial GT.

C. Evaluation Results

The best-performing MUFIN Image Annotation settings (detailed in [11]) were used to obtain an initial annotation of each query image with the mean precision of 58.76 %. From the initial annotations, we generated different types of ARF and submitted it to each of our three methods to obtain an improved set of 30 descriptive keywords. The following sections analyze the performance of individual solutions.

1) CBIR with text re-ranking (CBIR+TextRank): Using the combination of content-based search and text re-ranking, we were able to achieve new result precision of 73.8 % (lower bound), which is a 15 % improvement over the initial annotation. This means that at least 4.5 new relevant keywords were identified on average within each 30-keyword annotation. A more detailed view on the numbers of new relevant keywords is provided in Figure 3. The upper bound on MP is 83.5 %, which would correspond to almost 25 % improvement and 7 new relevant keywords on average. The combination of parameters that produced the best result is detailed in Table II.

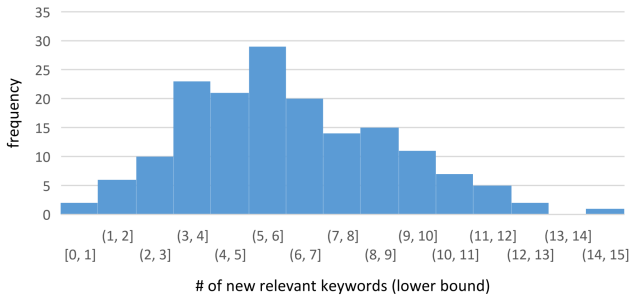

Fig. 3. Histogram of the number of new relevant keywords (CBIR+TextRank, optimal settings).

As we expected, the quality of the new annotation result is strongly influenced by the properties of the ARF, in particular by its size. In Figure 4, we can see how the annotation precision grows with the number of positive keywords contained in ARF. We can also observe that the RF selection method LAST produces the best results. This is consistent with observations reported from RF experiments in related fields (e.g. [8]) – the system gains the most information from relevance scores of the keywords that had lower confidence in the initial annotation. The last point of the graph shows the precision that can be achieved if all correct keywords are marked in the initial annotation (ARF selection method ALL). We have plotted this point for RF size 30 which is the theoretical maximum of the number of relevant kewyords; however, on average there are only about 17 relevant keywords in each result. When all relevant words are selected, we can automatically mark the remaining words as negative and exploit them as well in the RF processing, which will result in significant precision improvement depicted by the green star point.

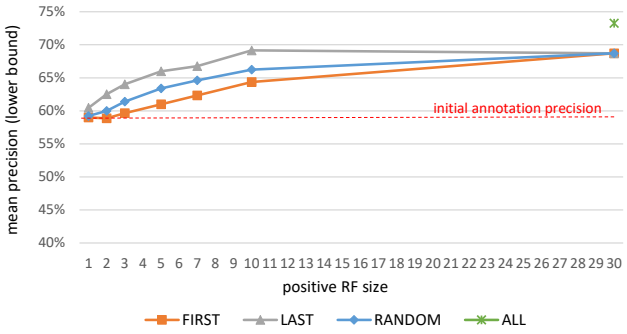

Fig. 4. Influence of positive RF size on precision (CBIR+TextRank)

In Figure 5, we take a closer look at the importance of positive and negative RF keywords. It reveals that positive keywords are more informative than the negative ones – for a fixed ARF size, it is always better to have just positive keywords in the ARF than a combination of positive and negative examples. The only reasonable use of negative feedback is thus the above-mentioned situation of full positive feedback where the negative keywords can be determined automatically.

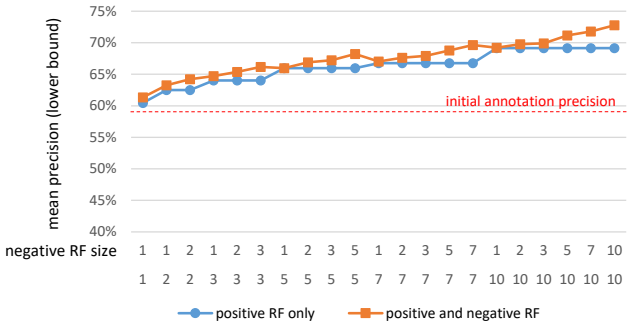

Fig. 5. Importance of positive and negative RF keywords (CBIR+TextRank, RF type LAST)

TABLE II
BEST RESULTS ACHIEVED BY INDIVIDUAL METHODS

Method	MP (lb/ub) [%]	MP-new (lb/ub) [%]	RW-lost	RW-gained (lb)	Time [s]	ARF properties	RF processing params
CBIR+TextRank	73.8 / 83.5	46.1 / 66.0	0	4.5	1.6	RF type:ALL RF size: 30 multivalued assessments	k^{Cand} = 500 k^{QN} = 100
Text+CBIRRank	72.2 / 79.1	40.0 / 59.2	0	4.0	4.5	RF type:ALL RF size: 30 multivalued assessments	k^{Cand} = 2000 k^{QN} = 100
Text2Vec	69.9 / 78.2	38.6 / 56.8	0	3.3	3.1	RF type:ALL RF size: 30 multivalued assessments	k^{Cand} = 1000 k^{QN} = 100

We also compared the effects of using multivalued and binary ARF. The multivalued ARF performed slightly better but the difference was only about 0.5 % of MP.

Finally, let us look at the influence of the candidate set size (k^{Cand}) and the query neighbor count (k^{QN}). In Figure 6, we can observe that the best results were achieved for candidate set sizes of 500-1000 objects. If a smaller candidate set was used, there were not enough text-relevant images that could be identified by the re-ranking. On the other hand, the biggest candidate set we tested ($k^{Cand} = 2000$) already contained too much visual noise. Regarding k^{QN}, we can see that for larger candidate sets the optimal number of query neighbors is 100-200 objects. Since both k^{Cand} and k^{QN} influence the annotation processing costs, it is preferable not to use the maximum values. Therefore, $k^{Cand} = 500$ and $k^{QN} = 100$ have been selected as the optimal settings.

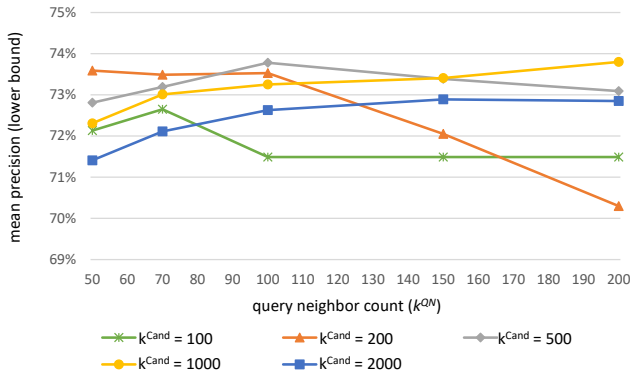

Fig. 6. Influence of k^{Cand} and k^{QN} on precision (CBIR+TextRank, RF type ALL)

2) Text search with CBIR re-ranking (Text+CBIRRank): The second tested method uses the opposite approach – the candidate set is obtained via text search engine and the visual descriptor is used for re-ranking. The best result achieved by this method has mean precision between 72.2 % (lower bound) and 79.1 % (upper bound), which represents a 13 % to 20 % improvement. The method was able to introduce from three to six new relevant keywords on average. The best-performing parameters (see Table II) were similar to the CBIR+TextRank method except for the candidate set size, which was bigger.

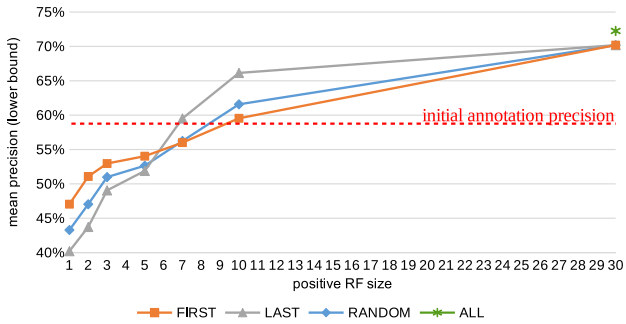

Fig. 7. Influence of positive RF size on precision (Text+CBIRRank)

Figure 7 shows the dependence of the MP on the ARF size. We can see that a small text query retrieves a diverse collection of candidates from which the re-ranking cannot select good query neighbors. This is caused by the fact that the Profiset images are described by keywords only, which makes it difficult for the search engine to determine annotation-helpful images given only a few (one to three) words. However, as the number of keywords increases, the precision improves. We can also observe that the keywords taken from the beginning of the initial annotation (RF type FIRST) work better when the RF size is small but the less-obvious keywords (RF type LAST) achieve higher precision for bigger RF sizes. This is probably caused by the fact that the LAST keywords are more restrictive, thus the text search result will be more compact. This is also supported by the fact that there is a rather steep improvement of the precision when the full RF is used.

We have also experimented with the negative RF, which can be used by the text search engine to filter out items described by some of the negative keywords. We have observed similar positive influence of adding the negative RF as with the CBIR+TextRank method. However, for a fixed RF size it is again better to provide only positive keywords. When the full positive RF is selected (RF type ALL), we can again automatically use the remaining keywords as the negative RF, which increases the mean precision from 70.1 % to 72.2 %.

Regarding the number of candidate objects k^{Cand}, the situation is slightly different than with the previous method. The best results were achieved for the candidate set size of 2000 objects. For k^{Cand} up to 500, the CBIR re-ranking

was not able to find sufficiently coherent groups of images, which resulted in poor overall precision that was much lower than the initial annotation precision. On the other hand, with $k^{Cand} = 1000$ we achieved only about 2 % lower precision than with $k^{Cand} = 2000$. Since the computational costs are significantly smaller for $k^{Cand} = 1000$, we can recommend this value if efficiency is a concern.

3) NN-based transformation of ARF (Text2Vec): Finally, let us look at the possibilities of using the NN-based transformation of the text query into a visual descriptor. In this case, we were able to achieve 69.9–78.2 % mean precision (lower/upper bound), which improves the original annotation by 11–19 %. The optimal parameters were similar to the previous methods.

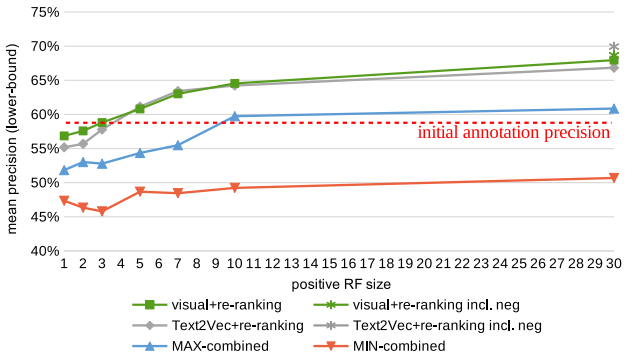

Fig. 8. Influence of positive RF size on precision (Text2Vec)

In Figure 8, the influence of the RF size and various modes of the Text2Vec utilization is depicted. The "visual+re-ranking" mode uses the query-image visual descriptor for content-based retrieval of candidates, which are re-ranked by the similarity to the query-image visual descriptor, similarity to the positive Text2Vec descriptor, and dissimilarity to the negative Text2Vec descriptor. The "Text2Vec+re-ranking" mode is similar, only the positive Text2Vec descriptor is used for the retrieval of candidates. These two approaches provide similar quality, the "visual+re-ranking" being slightly better for smaller RF sizes and "Text2Vec+re-ranking" for richer textual data. The "MIN-combined" and "MAX-combined" approaches use the minimum and maximum functions to combine the original query-image visual descriptor with the Text2Vec of the positive RF keywords. However, these simple combinations yield rather poor results, so a better way of combining the descriptors should be researched in future.

Experiments with the negative feedback have shown that it is not helpful in this method. The negative Text2Vec descriptor can only be used in the re-ranking phase where it should push the images similar to the negative keywords to the end of the list, but obviously the Text2Vec transformation of the negative keywords was not very successful.

The k^{Cand} parameter exhibits an expected behavior similar to the previous methods. Smaller candidate sets do not provide enough objects for the re-ranking to have significant impact, while the largest candidate set we considered is already too noisy. The best results were thus obtained for $k^{Cand} = 1000$.

The overall performance of the Text2Vec approach is rather disappointing. We have hoped that the Text2Vec transformation will be able to retrieve semantically related objects that are not described by the exact keywords contained in the RF. However, this was not confirmed by the experiments. We suspect the problem lies in the process of the Text2Vec neural network training, which is quite complex and has many variables, so a further study in this area will be necessary.

4) Efficiency: In Figure 9, we present the average annotation processing times. We can observe that the Text+CBIRRank method is very fast for smaller candidate sets but its costs sharply grow with the candidate set size. This is mainly due to the fact that the text search engine needs to retrieve the bulky visual descriptors for the re-ranking phase. We have also observed that the text search costs grow with the RF size, which is an expected behavior since more lists need to be intersected.

The Text2Vec transformation is rather expensive because the neural network needs to be utilized to transform the keywords. Since the model runs on a different machine with a GPU card, also the network transfer cost is included. The actual CBIR search time is thus shifted by the time needed to compute the Text2Vec transformation. However, for the biggest candidate set the text-search is the most costly.

As explained in Section IV-A, the efficiency of the CBIR+Rank method can be improved by caching the CBIR results from the original annotation. The effects of the caching are shown by the last curve in Figure 9. With this optimization, less than one second is needed to process the RF loop even for the biggest candidate set.

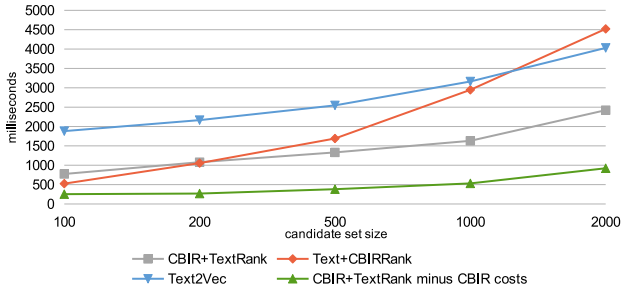

Fig. 9. Efficiency of the various ARF methods with respect to the size of the candidate set measured by wall-clock time

D. Discussion

From the experimental results, we can conclude that the RF can be successfully used to improve the quality of search-based annotations. Let us first summarize our findings regarding the ARF properties. The quality of the results improves with the growing RF size, which is no surprise. If we assume the photo-stock website scenario where users want as many keywords as possible, we can expect them to select *all* relevant words from the primary annotation for the RF. The remaining keywords can be automatically marked as irrelevant. This full ARF allows us to obtain the best results in the RF

Initial annotation:

Flower, plant, nature, red, close, pink, flowering, seasons, flowers, leaf, spring, background, Christmas, petals, white, color, nobody, bloom, period, tree, natural, beauty, orchid, photo, green, autumn, plants, bouquet, indoors, illustration

After RF loop:

Christmas, tree, indoors, plant, red, green, decoration, season, decorated, leaves, *baubles*, traditions, advent, *lights*, single, festivals, *herb*, *poinsettia*, *shrub*, holiday, *fir*, *traditional*, *ball*, *objects*, shot, medicinal, *poppy*, food, winter, interiors

Fig. 10. Comparison of annotation result before and after the RF loop (CBIR+TextRank with optimal parameter settings). Relevant keywords are underlined, RF keywords are highlighted in bold, keywords in italics are not evaluated in our partial GT.

loop. If (in another application) the users are not willing/able to mark all relevant keywords, they should mark as many positive examples as possible, preferably starting from the least probable candidates. Negative feedback is significantly less useful, therefore it is not advisable to ask users to spend time marking the non-relevant examples. It is also not worthwhile to require multivalued relevance assessments, as the performance improvement over binary assessments is negligible.

Out of the tested methods, the highest lower-bound precision of 73.8 % was achieved by the CBIR with textual re-ranking (CBIR+TextRank), which corresponds to 15 % improvement over the primary annotation. The annotation enrichment is illustrated in Figure 4. The text-search with visual re-ranking (Text+CBIRRank) ranked as second, achieving comparable results for larger RF sizes. The Text2Vec method was the worst of our three methods, being 4 % worse than the CBIR+TextRank. Furthermore, only the CBIR+TextRank method was able to consistently improve the result even for the smallest RF. The other two methods needed at least five keywords in order to enrich the annotation.

The most surprising result was the low performance of the Text2Vec transformations. The modes that try to combine the visual and Text2Vec descriptor together provided worse results than the initial annotation. However, only simple minimum and maximum functions were tried, so as a future work, we will focus on designing better combination methods. Our experiments also revealed that the Text2Vec transformation utilizes only a subset of the Profiset vocabulary, so we plan to refine the NN training process to include a wider set of keywords. We can also enrich the learning process by employing some semantic information (e.g. WordNet synonyms).

From the efficiency point of view, even with the most expansive parameters all annotations could be computed under five seconds. If we utilize a simple caching mechanism, we can lower the time of the best-performing CBIR+TextRank method below one second, which is considered an online response.

VI. Conclusions

We have studied the possibilities of exploiting the image annotation relevance feedback, which is a novel problem that has not been systematically addressed yet. We have presented three multi-modal search techniques that can be applied to this problem, implemented them, and conducted numerous experiments. Our best-performing method increased the overall precision of the annotation by 15 %. We have also analyzed the influence of different RF properties on the result quality.

In the future, we would like to tackle the problem of pseudo-relevance feedback, where the user input is not necessary. For that, we plan to employ tailored CNN-based classifiers that can provide highly-confident feedback keywords from a limited vocabulary (due to the necessity of training). Furthermore, we want to focus on exploiting the annotation relevance feedback also in the second annotation phase, i.e. during the mining of keywords from the query neighbors.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1106–1114.

[2] X. Li, L. Chen, L. Zhang, F. Lin, and W. Ma, "Image annotation by large-scale content-based image retrieval," in *ACM International Conference on Multimedia*, 2006, pp. 607–610.

[3] X.-J. Wang, L. Zhang, and W.-Y. Ma, "Duplicate-search-based image annotation using web-scale data," *Proceedings of the IEEE*, vol. 100, no. 9, pp. 2705–2721, 2012.

[4] P. Budíková, M. Batko, and P. Zezula, "ConceptRank for search-based image annotation," *Multimedia Tools Appl.*, vol. 77, no. 7, pp. 8847–8882, 2018.

[5] C. Cui, J. Ma, T. Lian, Z. Chen, and S. Wang, "Improving image annotation via ranking-oriented neighbor search and learning-based keyword propagation," *JASIST*, vol. 66, no. 1, pp. 82–98, 2015.

[6] T. Mensink, J. J. Verbeek, and G. Csurka, "Trans media relevance feedback for image autoannotation," in *British Machine Vision Conference (BMVC 2010)*, 2010, pp. 1–12.

[7] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*. Cambridge University Press, 2008.

[8] X. S. Zhou and T. S. Huang, "Relevance feedback in image retrieval: A comprehensive review," *Multimedia Syst.*, vol. 8, no. 6, pp. 536–544, 2003.

[9] B. Thomee and M. S. Lew, "Interactive search in image retrieval: a survey," *IJMIR*, vol. 1, no. 2, pp. 71–86, 2012.

[10] P. K. Atrey, M. A. Hossain, A. El-Saddik, and M. S. Kankanhalli, "Multimodal fusion for multimedia analysis: a survey," *Multimedia Syst.*, vol. 16, no. 6, pp. 345–379, 2010.

[11] P. Budikova, M. Batko, and P. Zezula, "Fusion strategies for large-scale multi-modal image retrieval," *T. Large-Scale Data- and Knowledge-Centered Systems*, vol. 33, pp. 146–184, 2017.

[12] Y. Jing and S. Baluja, "VisualRank: Applying PageRank to large-scale image search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, pp. 1877–1890, 2008.

[13] J. M. dos Santos, J. M. B. Cavalcanti, P. C. Saraiva, and E. S. de Moura, "Multimodal re-ranking of product image search results," in *35th European Conference on IR Research (ECIR 2013)*, 2013, pp. 62–73.

[14] F. Carrara, A. Esuli, T. Fagni, F. Falchi, and A. M. Fernández, "Picture it in your mind: generating high level visual representations from textual descriptions," *Inf. Retr. Journal*, vol. 21, no. 2-3, pp. 208–229, 2018.

[15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[16] P. Budikova, M. Batko, and P. Zezula, "Evaluation platform for content-based image retrieval systems," in *International Conference on Theory and Practice of Digital Libraries (TPDL 2011)*, 2011, pp. 130–142.

[17] D. Novak and P. Zezula, "PPP-codes for large-scale similarity searching," *Trans. Large-Scale Data- and Knowledge-Centered Systems*, vol. 24, pp. 61–87, 2016.

[18] M. Batko, D. Novak, and P. Zezula, "MESSIF: Metric similarity search implementation framework," in *1st DELOS Conference*. Springer, 2007, pp. 1–10.

[19] M. McCandless, E. Hatcher, and O. Gospodnetić, *Lucene in Action: Covers Apache Lucene V. 3. 0*, ser. Manning Pubs Co Series. Manning, 2010.