Image Retrieval Using Natural Language and Content-Based Techniques

Kate Byrne
k.byrne@ed.ac.uk

Ewan Klein
ewan@inf.ed.ac.uk

Institute for Communicating and Collaborative Systems
School of Informatics
University of Edinburgh

ABSTRACT

This paper deals with the application of Natural Language Processing and Content-Based Image Retrieval to the practical problem of finding database images to answer user queries. The data collection used contains about 50,000 digital images linked to a large text database, but typically not having individual descriptive captions. A number of different techniques were explored, in particular the combination of traditional IR techniques with Named Entity Recognition, CBIR and relational database tools. Some methods combined well and others did not. Integrating normal database joins with external inverted text indexes built on NE-marked text worked well on the material. Combining these techniques with CBIR ones, that look at the physical characteristics of images in terms of colour, shape and so forth, was less successful; although circumstances were identified in which it is useful.

1. INTRODUCTION

In recent years many of our big public "heritage" collections — in museums, galleries, archives and libraries — have been made available for the first time to world-wide audiences over the Internet. Often the material includes large photographic collections and these have been digitised so that they can reach a much wider public than in the past. Developing image retrieval systems to access such collections is not always straightforward because the material was originally catalogued with different access methods in mind. This paper describes a short project on image retrieval using part of the National Monuments Record of Scotland (NMRS), an archive maintained by the Royal Commission on the Ancient and Historical Monuments of Scotland (RCAHMS) (see http://www.rcahms.gov.uk), a public body based in Edinburgh. The nature of the data meant that no single appropriate approach was obvious. Instead a range of methods was explored, including IR, IE, QA, CBIR (content-based image retrieval) and database techniques. The objective was

to assess whether they worked well in combination or not.

The image collection used is linked to a relational database containing both structured and semi-structured data, with a high proportion of free text fields — this is typical of the type of historical collection mentioned above. The domain in this case is archaeology and architectural history, and the material is mainly site-based: for example there may be a single principal text record describing an archaeological site or a building, with anything from a handful to several hundred related items attached, such as photographs, maps, excavation reports and so forth. The collection was started in 1908 and has grown steadily ever since. It was designed primarily for site-based access by specialists in the field: a researcher would start with a site report and consult boxes of related collection material. Individual captions for photographs were generally unnecessary or, if present, would often be of the form "view from west", "front view" or similar. Today's user is likely to be from a school, a local history society or the general public, and frequently wants a cross-site response, to a query such as "Show me the 18th Century houses near where I live". The sparseness of individual image caption data makes standard retrieval by caption impractical: 29% of the digital images available have no caption at all and a further 15–20% have only non-specific captions of the "view from west" type. Therefore this project used a selection of text fields from across the database as its source material, as well as the physical image content itself.

A set of just over 50,000 images was used, linked to a database of text amounting to approximately 3.9 million words. This was around 10–25% of the whole NMRS database (depending on how one measures its size: by number of site records, of archive items, or amount of text).

2. METHODOLOGY

There are two categories of approaches to image retrieval: those starting with the image content (CBIR) and those starting with the associated text. Veltkamp and Tanase [9] give a comprehensive survey of current CBIR applications, looking at 39 different systems in terms of features used, indexing data structures, matching techniques and so forth. Matches to a query image are found using colour distribution, texture, orientation of edges, and sometimes semantic features such as "indoors", "outdoors", "offensive", "be-
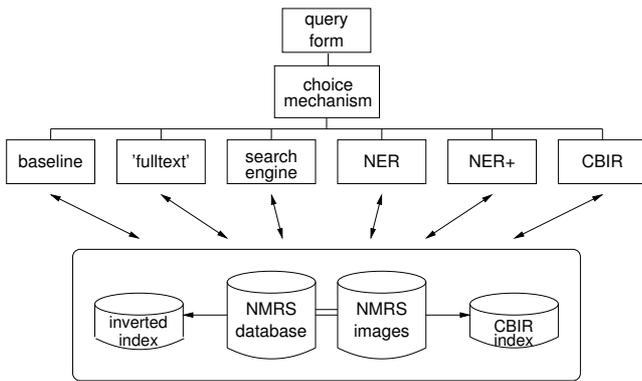
Figure 1: CANTRIP application

nign". The system used for this project was SIMPLIcity (Semantics-Sensitive Integrated Matching for Picture Libraries), which was kindly made available by its developers; see [11]. This is a sophisticated system using image semantics and designed to handle very large image collections.

Because it is a historical collection spanning almost a century of recording, a high proportion of the NMRS images are black and white. CBIR techniques such as colour histograms and edge matching struggle with monochrome images. Furthermore, they work best where the image collection can readily be divided into discrete categories, that also cover all the most frequent types of query required. Neither of these conditions really holds for the NMRS images. Therefore a primarily text-based approach was chosen, with CBIR as a supporting tool where appropriate.

Several different text based methods were used, added successively in layers. In order to test how they operated in combination, six versions of the image retrieval application, named CANTRIP, [1] were built, related as shown in Figure 1 and described below. To permit a blind evaluation, all six versions used an identical query interface so that the user could not distinguish one version from another.

2.1 Baseline Version

The textual data was loaded into a MySQL relational database using the table definitions of the parent NMRS database. A ten table subset of the NMRS was used, along with some ancillary lookup code lists. The baseline system used standard SQL select queries over left-anchored indexes, with wildcards being inserted between and after query terms. It did not search the larger free text fields as this simple indexing (requiring a match at the start of the field) would be ineffective on them. The query interface had five fields, each corresponding to a database field or concatenation of fields, and the baseline simply used these fields to construct an SQL query.

---

[1]The name was chosen as a nod to the existing Web-based query applications developed by RCAHMS: CANMORE (mainly for queries on site text) and CANMAP (for spatial queries). A "cantrip" is a piece of magic or mischief, in Scots dialect.

2.2 "Fulltext" Version

This version also used the relational database structure, but with "fulltext" inverted indexes on the main text fields. These indexes are a feature of MySQL (version 3.23 and later). Future versions of MySQL will permit configuration of the fulltext indexing, but in the version used this was not possible. The comparison with the "search engine" version is therefore indicative only, as they could not be set up identically. The aim of this version was to test the difference between inverted indexing that exploits the database structure as here, and that ignores it as in Section 2.3 below. It was anticipated that some of the shorter text fields would not provide enough material for a balanced index, so these fields were grouped together into concatenated indexes. For example, there are three separate classification fields per record, of up to 100 characters each; these were aggregated for indexing.

2.3 Search Engine Version

The "search engine" version and those following ignored the relational structure and indexed all the relevant text fields associated with each image as a single document. It used TF-IDF indexing, where the weight of each keyword is the product of its **term frequency**, f_{kd} (how often keyword k occurs in document d), and its **inverse document frequency** (inversely proportional to the number of documents in the whole corpus that contain k), defined as:

$$idf_k = log\left(\frac{(NDoc - D_k) + 0.5}{D_k + 0.5}\right)$$

where $NDoc$ is the number of documents in the corpus, and D_k is the number of documents that contain keyword k.

This formula for idf_k, smoothed to moderate extreme values and normalised with respect to the number of documents *not* containing the keyword, was used following Robertson and Sparck Jones, quoted in [1]. The term "keyword" signifies a cleaned token: after stop words have been removed, punctuation stripped and Porter stemming done.

The search engine tool used normalisation by document length in ranking the returned hits, where **document length** is defined as:

$$len(d) = \sqrt{\sum_{k \epsilon d}(f_{kd} \times idf_k)^2}$$

The aim is to allow for the fact that short documents inevitably contain fewer keywords than long ones, so preventing long repetitive documents always being ranked above short pithy ones. The indexing and search engine software was produced by adapting the suite of programs made available with Richard Belew's book [1].

The resulting inverted index contained just over 25,000 word types, a fairly average vocabulary figure for a corpus of this size. To run the search, text from all query fields in the CANTRIP interface was simply tokenised as described above and passed as a single string to the search engine.

2.4 NER Version

The "NER" version tested whether the plain IR method just described could be improved by first identifying Named Entities in the source data and query, and treating these as tokens. For this domain, the NEs included specialist archaeological and architectural terms like "chambered cairn" and "cruck-framed" as well as the usual "person", "organisation", "location" categories. In fact CANTRIP did not make use of categorisation, but this would be a useful area to explore further. NE recognition was carried out using a gazetteer of around 20,000 terms, built from various database fields including a free-standing hierarchical thesaurus provided by RCAHMS. There is scope here for further work, in hand-annotating the data and measuring the NER coverage actually achieved.

As expected, this index contained more terms — over 27,300 — but with lower frequency counts. The NER step was designed to find the longest available entity string in each case, and the components of many compounds occurred in their own right, accounting for the higher vocabulary count. NE terms found, in both the source data and the query, were replaced with regular expression patterns, so that variants on the representation of entities could be matched. For example, "Alexander J. Adie" became `a(lexander)?( j\.?)? adie`. Thus, no action to highlight entity strings is required from the user, and inexact string matches are possible.

2.5 Enhanced NER Version

The "Enhanced NER", or "NER+", version added three extra components to the previous one:

1. **Weighting of NEs:** NE terms in the inverted index were given a weight of 1.5 (arbitrarily chosen, but this seemed the best value after experimentation with others) as against 1.0 for normal entries. The theory behind this is that not only are the NEs supposed to be "content carriers" but they also require extra weighting to balance the fact that their frequency counts as compounds will be lower than if the constituents were counted separately. See [5] for discussion of ideas similar to this.

2. **Query expansion:** Query terms for "type of site" were compared against the thesaurus to find the canonical form for that term plus any entries for **preferred term** or **related term**.

3. **Database intersection:** A location pick-list field was included in the query form in order to test database intersection. Given a set of candidate answers based on running the whole query against the inverted index, the location code could then be used to rule out inappropriate ones. A coded field was used for simplicity, but this principle could be extended to other database fields — using the unstructured index to find a range of candidates quickly, and then narrowing it using the precision of SQL.

Ideally these three additions should have been tested separately, but eight versions of the application would have been unmanageable to evaluate.

2.6 CBIR Version

The final version used all the components just described, then intersected the results with CBIR ones. Because the CBIR system requires a seed image for the query, and in order to maintain the anonymity of the six versions, a query image had to be chosen arbitrarily. The top image from the NER+ version was used. However, because it would clearly be more sensible in practice to allow the user to choose the image for content matching, an extra facility was built into the interface permitting any chosen image to be matched. Two functions were provided: "Match Image *and* Query" and "Just Match Image". These functions were not part of the formal evaluation, but informal findings are discussed below.

The combination of results was done by simple intersection: the top 300 text-based hits (or however many were found, if fewer) were intersected with the same number of content-based matches to the top-ranked hit. The intersection method and numbers were chosen after a series of experiments, but there is room for more work here.

3. RESULTS

Evaluation was performed by two specialists from RCAHMS (an archaeologist and an architectural historian) and one non-specialist. A set of 16 test queries was provided by RCAHMS, with a balance of archaeology, architecture and industrial topics. Each evaluator used the same query wording and ran every query six times, once for each of the CANTRIP versions (which were in random order). The queries were all designed to produce a small result set, and the top 20 hits — or all the results if fewer than 20 — were examined for every run. Each result image was marked as either "relevant", "not relevant" or, if the result was unclear for some reason, "uncertain". Response time was also measured, as this would be important in a real application. Figure shows an example of the results screen. Clicking on a thumbnail image produces the pop-up window with a larger image and its associated text.

A larger number of queries, and possibly of evaluators, would have been preferable; but time was constrained and each evaluation session took several hours. According to the findings of [10], a set of 50 query topics would be needed for confidence in the comparisons of one retrieval method against another. It is hoped that future work will include more extensive trials. To partially address this shortcoming, the variance of the scores was analysed across evaluators and across queries, as detailed below.

The traditional measures, **precision** and **recall**, are only partially appropriate here.[2] Precision is easy to calculate, but with a ranked set one clearly wants to give more credit to a system that returns 10 good hits followed by 10 misses than one which puts all the misses first. Also, the judgment of relevance is inevitably subjective to some extent — for this reason the degree of agreement between evaluators was measured. Recall is notoriously difficult to calculate on a

---

[2]The standard definitions of precision and recall are referred to: $precision = tp/(tp + fp)$ or the proportion of selected items that were right; and $recall = tp/(tp + fn)$ or the proportion of the whole target that was found. (tp is true positives, fp is false positives and fn is false negatives.)

	Time (sec)	Precision	Recall[a]	Score	Accuracy
baseline	1.63	37.50%	62.50%	40.13	26.14%
fulltext	1.96	48.13%	100.00%	72.69	40.30%
search engine	3.17	29.38%	75.00%	48.63	29.84%
NER	5.00	31.56%	75.00%	39.06	25.68%
NER+	5.77	66.88%	100.00%	100.25	**52.28%**
CBIR	7.98	60.28%	36.11%	23.63	18.97%

Table 1: Summary of evaluation results for the six CANTRIP versions

[a]Over only 3 of the 16 result sets

large database because, as is discussed in for example [2], nobody knows exactly what's in it, and hence what is missing from the retrieved set. The frequently used solution (see for example [8]) of using a tiny subset of the database and comparing results against the whole of it, did not seem appropriate in this case because a TF-IDF index over a very small corpus would not be reliable. In 3 of the 16 queries there seemed a good case for believing that the successful systems had returned the entire result set available,[3] so recall percentages were calculated here; but in most cases it's impossible to estimate. The results are summarised in Table 1.

The score used (alongside precision and, where appropriate, recall) was designed to give credit for placement of correct answers within the ranked set and to impose a penalty for returning misses. The figures are shown in the column labelled "Score" in Table 1. Each correct image is given a positional score: 20 points for the first place, 19 for the second and so on down to 1 for the twentieth. For each wrong answer 1 point is deducted. The scores are normalised to produce the "Accuracy" column percentages by:

$$Accuracy = \frac{Score + 20}{230}$$

where $Score$ varies between $Min_{score} = 20 \times -1 = -20$ and $Max_{score} = \sum_{n=1}^{20} n = 210$.

A defect of this method is that in the small minority of cases where the target is fewer than 20 hits, a completely correct answer will achieve much less than the maximum score. For example, one of the test queries had a target of just three images, for which the highest score possible is 57 (20+19+18). The option of giving an arbitrary bonus award for 100% recall was considered, but seemed too unsafe. The recall figures are just not sufficiently reliable.

This scoring was used instead of a more standard method such as the Mean Reciprocal Rank score used, for instance, in work described in [7] and [3]. The MRR method, as is pointed out in [6], relies on there being a pre-existing set of relevance judgments to compare against, which was not possible here. The scoring used served its purpose, which was to compare the six application versions against each other on the criteria the systems were intended to meet; basically giving the best scores to correct answers returned

high in the list, whilst penalising over-answering. It has been pointed out that a preferable alternative might have been the "number-to-view" evaluation method described in [4], or the similar "precision at n" method which is standard in the IR field. Here scoring is based on the size of the result set needed in order to get a desired number of correct results, e.g. 20 results including 12 correct hits, 50 including 35 and such like. This method has the advantage of not requiring relevance judgments in advance across the entire database, but it does not give so much credit for ranking, looking only at precision within a set of a certain size.

The sets of evaluator marks were compared, noting any disagreements between them. Where two out of the three agreed, their mark was used. In the sole case (out of 1,147 marks) where there was a three-way split, "uncertain" was used. They disagreed on only 7% of the marks, most of the disagreements (60% of them) being where two of the three marked an image "wrong" and the third chose "uncertain". Of the 25 cases where the majority were "uncertain", all but one arose on the same query: "Cranes or bridges connected with the Arrol firm".[4] There were only 9 cases, less than 1% of the total, where the evaluators disagreed about a result being correct; generally it was easy to tell.

4. ANALYSIS
As Table 1 shows, the enhanced NER version performed best overall, by a significant margin. On the accuracy measure described above it doubled the baseline performance.

It was no surprise that the CBIR version did so badly. It had no choice but to work with the first image returned by NER+, and also the simple intersection of results didn't favour it. This is not the best way to use CBIR tools. There are no formal results to support this contention, but it was clear from using the system that CBIR *can* be a useful aid to text-based retrieval, but *only* where the user actually wants results that are visually similar. The technology works well with plans and maps, and fairly well with colour photographs. To give an example: suppose the user is interested in henges.[5] There are 114 digital image records where "henge" appears in the classification fields and 20 more where henges are mentioned in the text but not in the classification. This is too many to browse through with

[3]These 3 queries were for specific entities, such as "Mavisbank House". In that example the successful versions all returned the same small set of correct images and no others. The less succesful versions returned additional spurious images, such as "Mavisbank Quay", but no new correct ones.

[4]In this case the non-specialist marked items as wrong where Arrol's was not explicitly mentioned, but both the experts chose "uncertain" when they were fairly sure from the picture that the crane or bridge depicted *was* an Arrol one, even if the text didn't say so.

[5]Circular enclosures defined by a ditch and bank, sometimes with stone or timber settings inside.
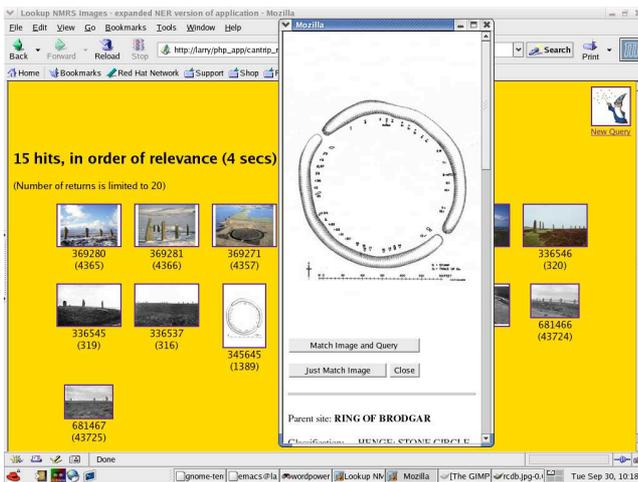
60

Figure 2: CANTRIP results screen

Figure 3: CBIR match on site plan

CANTRIP. Suppose further that the user is particularly interested in line drawings of henges: the NER+ version returns 7 of these amongst the first 20 images. If the user clicks on one of them and chooses the "Match Image and Query" option from the pop-up window, further line drawings of henges will be returned that were not in the original set. In this case the image is suitable for content-based matching, and the text query has a large pool of good hits, so the combination works well.

The "Just Match Image" function can also be useful. For example, a search for a specific site, such as "Ring of Brodgar", will bring back a heterogeneous collection of images of it, and in this case one of them is a site plan. Trying to "Match Image and Query" will fail because the set of good hits is very small and CBIR matching cannot expand it. However, a content-based match on the site plan, ignoring the text query, will return images of similar-looking site plans across the whole of Scotland. Figures 2 and 3 illustrate this point. It's not automatic that a similar site plan indicates a similar type of site, but this kind of match may be helpful and would be difficult to reproduce with a text query.

What was disappointing in the overall results was that the basic NER version performed so poorly; slightly worse than the baseline. This seemed surprising because it is closest in methodology to NER+ which did so much better. There isn't room here to present the investigation fully, but on further examination it was realised that a mistake had been made with the location data, resulting in the codes used not being translated to their text values in the data on which the TF-IDF index was built. Other location data was present and indexed, but the particular strings being searched for (such as "Perth and Kinross" or "Edinburgh, City of") were not necessarily present in the right form, though they were correctly marked as NEs if present. This skewed the results for location-based queries against the versions that did not directly use the location code field in the database. Half the test queries were location-based and half were not; when the location queries were taken out of the results the NER and NER+ versions came top, though average scores were lower.
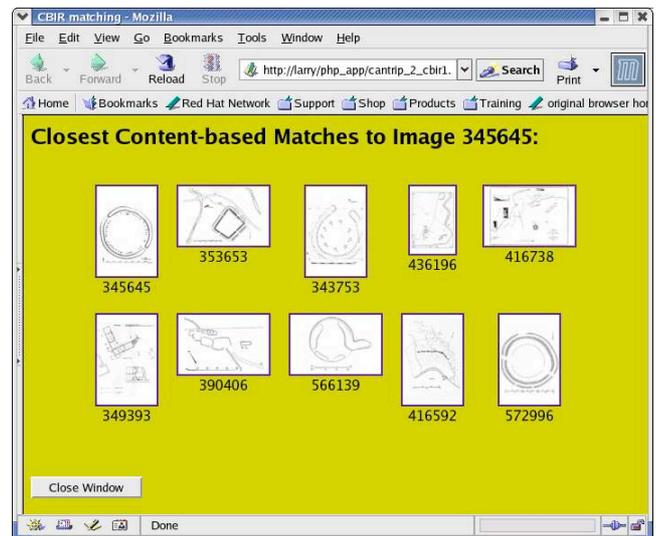
It's worth examining some of the individual queries. Two were examples of quite a common query type: information on a single entity ("Frank Matcham" and "Mavisbank House" respectively). As is to be expected, the two NER versions both get top marks here. Each recognises the query string as a single entity token and simply returns the documents containing it and no others. The search engine also finds all of the records, but fills the rest of the slots up to the limit with irrelevant items that contain one but not both of the query words. Another query, "cropmarks of Roman camps", is an example of where spotting an entity string did *not* help. The NER version results include records on a colliery called "Roman Camps mine" and on the "Roman Camp Hotel" in Callander. It's a failing of the NER method used that neither of these was identified in full as an entity in its own right, or of course they would have been ignored. This could certainly be improved on. Another query, "churches dedicated to St Columba", highlighted a problem with weighting; the NER+ version brought back churches with the wrong dedication. Both NER versions correctly identify "St Columba" as an entity pattern ((saint|st|s).?_columba'?s?). However, the query expansion used by NER+ turned "church" into a huge string of related terms: church ritual_buildings? bell_towers? burial[-_]?aisles? cathedral; (church|kirk)[-_]?yards? collegiate_church lych_gate round_towers? shrine; steeple. It is clear with hindsight that the weighting should have allowed for such expansion and prevented this term overwhelming the rest of the query, as it did in this case.

The figures given in Table 1 are of course averages across the whole query set. The variance of the scores for each system across the 16 queries was high, but a t test comparing NER+ with each of its rivals shows a statistically significant difference at a probability level of $\alpha = 0.0005$ in every case. However, a common-sense view is that each system performed well on some queries and poorly on others. This suggests a much larger set of test queries is needed to investigate strengths and weaknesses thoroughly; there is really

61

no such thing as an "average" query.

5. CONCLUSIONS

One of the clear findings is that broad-brush IR techniques are effective against this kind of semi-structured, text-rich relational database. It is clear that *if* the database fields include simple, cut-and-dried classification data items these will always provide the simplest way to get a definite "in or out" decision about candidates for the result set. The interesting thing about this data, and all similar text-rich datasets that have accreted over many years, is that such clear-cut classifications are often impossible. The combination of different techniques therefore seems promising:

- identify important entities and "concept terms" throughout the text;
- ignore the database structure and use statistical term weighting across the whole text to find candidate answers;
- weed out the spurious ones with the precision of SQL selection against the structured fields.

The application version that combined CBIR with text methods was not successful. This was not a surprise, for the reasons presented above. However, CBIR tools *were* effective when the image and query both met certain conditions.

The combination of NER with TF-IDF indexing worked well. NE identification has several advantages over simple string matching such as might be achieved by using quoted strings in a Web search engine query:

1. Inexact matches are possible; NER techniques code the entity itself and allow multiple representations.

2. If the processing of the source material can be done in advance, the performance of the search engine can be faster and its architecture less complicated than if it is responsible for finding signficant strings at query time.

3. Although not tested here, NE classification can help with disambiguation of entities from different categories.

4. Query expansion as used here depends on detecting NEs in the query. The system may be able to substitute a preferred term for that supplied by the user.

5. Single word entity terms can be handled as NEs, whereas a quoted string approach only works with compound terms.

6. Perhaps most significantly, no special knowledge of query technique is required from the user.

Although the pre-processing required by NLP methods may not be practical for IR applications such as Web searching, with vast and ever-changing source material, it seems a valid option for relatively fixed text databases such as the NMRS one and other similar archive resources. Very significant public resources have typically been used in constructing such archives over many decades. Improving public access to them is a useful task.

7. REFERENCES

[1] R. K. Belew. *Finding Out About: a Cognitive Perspective on Search Engine Technology*. Cambridge University Press, 2000.

[2] S. Flank. A layered approach to NLP-based information retrieval. In *Proceedings of the 36th ACL and the 17th COLING Conferences*, pages 397–403, Montreal, 1998.

[3] M. A. Greenwood and R. Gaizauskas. Using a named entity tagger to generalise surface matching text patterns for question answering. In *EACL03: 10th Conference of the European Chapter of the Association for Computational Linguistics*, Budapest, Hungary, April 2003. (Workshop on Natural Language Processing for Question-Answering).

[4] V. Harmandas, M. Sanderson, and M. D. Dunlop. Image retrieval by hypertext links. In *SIGIR 97*, Philadelphia PA, 1997. ACM.

[5] D. D. Lewis and K. Sparck Jones. Natural language processing for information retrieval. *Communications of the ACM*, 39(1):92–101, January 1996.

[6] K. Pastra, H. Saggion, and Y. Wilks. NLP for indexing and retrieval of captioned photographs. In *EACL03: 10th Conference of the European Chapter of the Association for Computational Linguistics*, Budapest, Hungary, April 2003.

[7] D. Ravichandran and E. Hovy. Learning surface text patterns for a question answering system. In *Proceedings of the 40th ACL Conference*, pages 41–47, Pennsylvania, 2002.

[8] T. Rose, D. Elworthy, A. Kotcheff, A. Clare, and P. Tsonis. ANVIL: a system for the retrieval of captioned images using NLP techniques. In *Challenge of Image Retrieval*, Brighton, UK, 2000.

[9] R. C. Veltkamp and M. Tanase. Content-based image retrieval systems: a survey. Technical Report UU-CS-2000-34, Utrecht University, Department of Computing Science, October 2000.

[10] E. M. Voorhees and C. Buckley. The effect of topic set size on retrieval experimental error. In *SIGIR 02*, Tampere, Finland, August 2002. ACM.

[11] J. Z. Wang, J. Li, and G. Wiederhold. SIMPLIcity: Semantics-sensitive Integrated Matching for Picture LIbraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9), September 2001.