

JOINT TEXTUAL AND VISUAL CUES FOR RETRIEVING IMAGES USING LATENT SEMANTIC INDEXING

Zoran Pečenović, Serge Ayer, and Martin Vetterli

Laboratory for Audio-Visual Communications
Swiss Federal Institute of Technology LAUSANNE
CH-1015 Lausanne, Switzerland

E-mail: {Zoran.Pecenovic, Serge.Ayer, Martin.Vetterli}@epfl.ch

ABSTRACT. In this article we present a novel approach of integrating textual and visual descriptors of images in a unified retrieval structure. The methodology, inspired from text retrieval and information filtering is based on Latent Semantic Indexing (LSI). It offers many advantages compared to standard approaches using simple data-base fields containing text and numerical visual descriptors. Whilst, compared to more elaborate constructs that model semantic content of the images, LSI offers less flexibility, it provides easier data acquisition and manipulation methods. It also offers novel query construction, result visualization and interaction means for exploratory and targeted query tasks.

We present an overview of the method and some preliminary results proving the applicability of the approach. In fact, starting from artificial data-sets and through increasing complexity up to real-world heterogeneous image collections, we illustrate the equivalence, and in many cases superiority of the proposed methodology.

1. INTRODUCTION

The advent of Internet search engines has spurred parallel efforts in building search engines for retrieving photographic and pictorial data. Today's Image Retrieval (IR) systems for heterogeneous collections suffer from the lack of unified indexing methods. The paradigms for querying images have diversified from query by keywords through query by example up to query by sketch. Several systems employ semantic annotation, deployed in parallel to visual content description (in numerical forms). However, they all use this semantic meta-data as separate query or a pre/post-filter for a visual query. Systems that model the semantic content of an image collection necessitate a quite complex database population phase where structured annotation is added through manual, and thus subjective, user intervention.

To our knowledge no system uses an approach where both visual and textual descriptors of image content are stored, managed and retrieved through a unified indexing structure. We present here a method, well known in text retrieval, extended and applied to image retrieval using both textual and visual cues in a coherent structure. Our approach allows the exploration of the relationships among these polymorphic descriptors, and in a certain way bridges the gap between purely syntactic visual content and semantic interpretation of the same content.

In the rest of this section we describe some related research efforts and try to highlight the differences with our approach, specifically in terms of functionality. Then we present a broad view of a framework for multimedia retrieval that we are developing and of which this approach is just a special instance. Section 2 is dedicated to the description of the method in its generic form giving some mathematical background and some references for further information. Section 3 will deal with the application to image/text data presenting: first the construction of a vocabulary for describing visual/textual image content, and then some discussion on alternative query paradigms and result visualizations. Section 4 is dedicated to experimental results with comparisons of performance and effectiveness. Section 5 will summarize our contribution and present some future research efforts.

1.1. Related research efforts. The problem of integrating textual and visual elements into queries for images has been addressed in several research efforts. In the WebSeek approach, Smith and Chang[SC97] use a model for the semantic annotation and classification methods to assign each image to a class. The queries can be executed through standard DBMS means (SQL, www interface). The two cues are however not managed in a uniform way. The QBIC project [FSN⁺95] has also addressed the issue in a similar way. The only related work we know of is by Cascia et al. [LC98] where the textual data is also processed by LSI, but the visual data is processed in standard vector

Keywords: joint text & image retrieval, latent semantic indexing, image segmentation, region characterization.
This work has been partly funded by grant #20-061493 of the Swiss National Fund for Scientific Research.

space fashion. The two cues are then combined into a single feature vector and treated by nearest neighbor search. For a more detailed survey of the literature we refer the reader to [SWS⁺00].

1.2. System overview. The project : *Content-based Image Retrieval and Consultation User-centered System* (CIRCUS) is an effort to develop a framework for image retrieval of distributed, heterogeneous, annotated image collections. It is constructed on top of a Client/Server architecture with an open and formally defined communication protocol MRML [MPM⁺00] which is available under the GNU Public License. This architecture allows for various user interfaces to connect to a set of retrieval servers implementing different methods or operating on different collections. It also provides means for automatic benchmarking and minimal effort meta-search engine construction. A series of user interaction paradigms, based on interactive visualizations and summarizing abilities have been presented in detail in [PDVP00]. Figure 1 illustrates this architecture and Figure 2 illustrates some user interaction abilities of the system as well as some sample results.

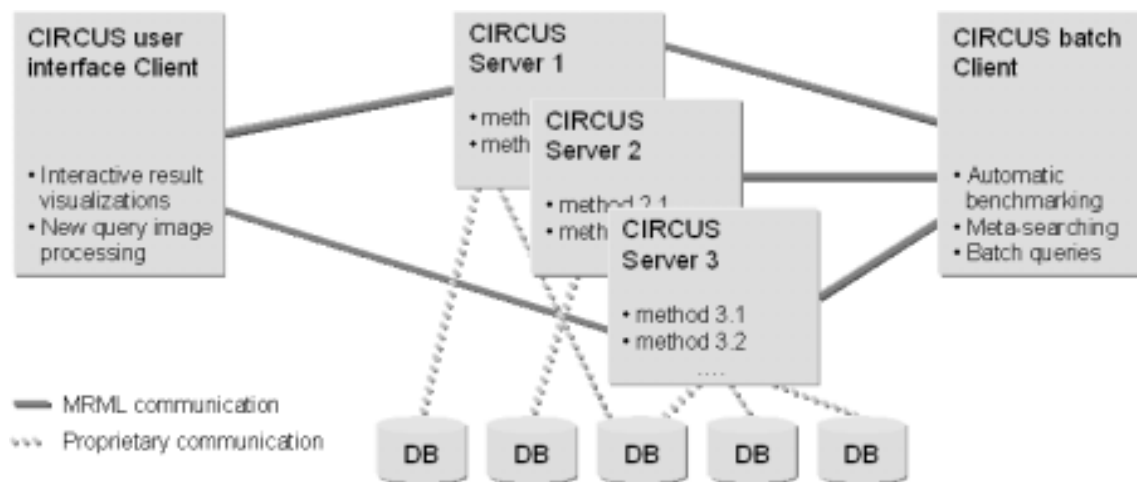


FIGURE 1. The components of the CIRCUS System.

2. AN OVERVIEW OF *Latent Semantic Indexing*

Latent Semantic Indexing was published in the early 1990's and patented, by Deerwester, Dumais et al. [DDF⁺90]. After successfully applying LSI to text-based documents researchers have started investigating its capabilities in other applications, like cross-lingual retrieval, information filtering or thesaurus construction. To our knowledge this is the first attempt to apply it to multimedia documents (annotated images).

The core idea of the method is to summarize the information in a collection of documents by approximating the term-document co-occurrence matrix. Term usage and co-occurrence contains implicit – or latent – semantic structure which is revealed, also implicitly, by a lower-rank approximation of the matrix. In other words terms that co-occur often and in many documents are considered to have a relationship of synonymy. Often, other terms carry several meanings decidable only according to the context. These two relationships are a cause for problems in standard information retrieval applications. LSI provides a way to identify them, moreover by an automatic procedure; so it also allows for the retrieval of relevant material that actually doesn't contain any occurrences of a query term. Similarly precision is increased since polysemic terms are identified. More detailed insight into why LSI achieves this kind of performance can be found in [Sto96].

2.1. Mathematical Background. We now give a brief overview of the technicalities involved with deploying LSI in any generic information retrieval environment.

Lets consider the term-document co-occurrence matrix \mathbf{A} based on a set of M terms and N documents. The dimensions $M \times N$ of this matrix are enormous¹. The actual entries $\{a_{ij}\}$ of \mathbf{A} are *idf*-weighted occurrence counts

¹However, it is essentially sparse since no document contains any significant percentage of the total number of terms, and vice versa no term (stop-words excluded) appears in all documents



Query by example and results displayed in a standard reading order by similarity.

Query by color proportions, a first (small) set of images is displayed underneath the query construction window.

FIGURE 2. One of the CIRCUS clients. User interactions and sample results.

of the terms in the documents. Any lower rank approximation of \mathbf{A} can be used, but the optimal approximation in any unitarily invariant norm is the truncated solution of the Singular Value Decomposition of \mathbf{A} :

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \simeq \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k^T,$$

where $\mathbf{\Sigma}$ is diagonal and sorted in decreasing order of magnitude of the singular values which constitute the diagonal. The subscript k denotes the restriction of a matrix to the the first k rows.

Once this approximation is computed (off-line) any user query considered as a very short document, can be expressed as an M -vector \mathbf{q} . The projected vector

$$\hat{\mathbf{q}} = \mathbf{q}^T\mathbf{U}_k\mathbf{\Sigma}_k^{-1}$$

of dimensionality k can be compared to similarly projected vectors representing the individual documents. Usually this comparison is done using the cosine of the angle between the query and document vectors. Conversely we can also project, instead of documents, terms into this same sub-space, using

$$\hat{\mathbf{t}} = \mathbf{t}\mathbf{V}_k\mathbf{\Sigma}_k^{-1}.$$

This two-way projection allows retrieval both of documents and terms relevant to a given query. Thus we also have a means to explicitly examine any term-term relationship, and test for synonymy and polysemy. This aspect is seldomly addressed by standard retrieval architectures.

The drawback of the method is its high computational cost, especially if updating of an LSI structure is to be applied frequently. In fact adding documents or terms implies a re-computation of the SVD of the large sparse matrix. Specific software methods for such computation exists ([BDL95]), and techniques for updating (approximately) rapidly varying collections have been investigated. We are not concerned with these issues, but with the transposition of the method to image retrieval. This, in fact, is the subject of the next section.

3. IMAGE (DE)COMPOSITION AND CHARACTERIZATION

Applying LSI to image retrieval could be considered from two distinct view points. First it can simply be applied for retrieving images solely based on the accompanying textual annotation. This approach translates to just applying a sophisticated method of text retrieval to image annotation, and ignores completely visual content of the images; although all the content might never have been annotated. The second view angle considers only visual aspects of the image, equating “terms” with numerical values representing certain global or local visual characteristics of the images. As above this approach ignores other data sources we might have available that associate semantics to

images as a whole or to certain of its parts. We opted for a third, novel, and in our opinion more powerful, approach : we jointly consider textual & visual data. “Terms” are thus *both* visually homogeneous parts of images characterized by color, texture and shape *and* semantic keyword annotations.

Segmentation, or identifying homogeneous regions associated with a single real object in an image from an unrestricted domain, is as yet an open problem. We use the Normalized Cut method proposed by Shi and Malik [SM00] because of its perceptually convincing results in various conditions.

Once an image has been segmented each region is characterized by color texture and shape. For color we use the moments of its 2-D chromaticity histogram. For texture we consider various statistics of the sub-bands of its wavelet decomposition. The shapes of the regions are characterized by moment invariants (with respect to scale, rotation and translation). The precision of any of these descriptions can be improved by using other state-of-the art techniques.

We have to differentiate different contexts for pursuing with the methodology. First of all we have to determine what the equivalent of a “term” in an image is. Then we have to decide how we identify a new region with an already known “term”. Finally we have to consider the annotation. These four concerns are addressed in the following sub-sections.

3.1. Defining the collection vocabulary. We have considered defining a “term” as an ordered triple:

$$(s, c, t) \in \mathcal{S} \times \mathcal{C} \times \mathcal{T} \quad \text{and} \quad s \neq \phi, \quad c \neq \phi \quad \& \quad t \neq \phi$$

where $s \in \mathcal{S}$ is a shape, $c \in \mathcal{C}$ a color and $t \in \mathcal{T}$ a texture. But this precludes equivalence classes of a quite simple type like ‘same car model under same viewing conditions but of different color’.

So our final choice was to introduce a redundant “term” definition considering each aspect separately and each combination of aspects also:

$$(s, c, t) \in (\mathcal{S} \cup \phi) \times (\mathcal{C} \cup \phi) \times (\mathcal{T} \cup \phi) \quad \text{and} \quad (s, c, t) \neq (\phi, \phi, \phi).$$

Thus a “term” is any combination of color, shape and texture properties. We can easily extend this definition to include textual annotation with a fourth element $w \in \mathcal{W}$, if it is available.

Global information about the image colors, textures or annotation not attached to a specific part of the image (author, location, atmosphere, impressions, etc.) also fits into this framework smoothly. It is naturally included as “terms” not carrying any shape description. This leads to a large, redundant set of features that exhibit much more structure than in normal word-document co-occurrence matrices. The only difference this induces is a faster decaying singular value distribution, but doesn’t invalidate the LSI model. Table 1 summarizes and illustrates the various types of terms we consider.

TABLE 1. Combinations of term properties (most have been omitted).

\mathcal{S}	\mathcal{C}	\mathcal{T}	\mathcal{W}	Description
•	•	•	•	A well formed term describing all aspects of a specific region.
•			•	Annotated shape.
•	•	•		A region of the image not annotated.
	•	•	•	Image of single texture (eg. a piece of fabric) with annotations.
	•			Global color information (cf. color proportion queries in Figure 2)
			•	Global annotation (eg. artist) but also all other keywords.

3.2. Known vs. unknown vocabulary. In order to construct a vocabulary for our set of images (equivalent of unique word lists in text) we must establish a classification of finite size for the visual features. Considering the color descriptions, we can quantize the color-space to any number of representative colors. We can achieve this either by selecting the perceptually most salient colors from a training set of images or pick a standard palette that exhibits maximum perceptual variety. Similarly for texture we can cluster the texture descriptors into a finite number of classes either by quantizing the space spanned by our wavelet sub-band statistics or using more elaborate classifications based on perceptual notions [MM98]. Textual descriptors naturally translate since they are discrete and a finite number of words is present in the database. The biggest problems occurs when we are dealing with shapes, since they offer basically an unlimited number of pixel configurations. To tackle this problem we divide and conquer in the following way:

- If the possible shapes come from a finite set of a-priori known classes \Rightarrow We take a representative image of each class and use it as a template for matching against each new identified region.

- The number of classes and their characteristics are unknown. Here we consider two sub-cases :
 - If we know how many classes how many shapes per class will be present in the database (like in catalogs) ⇒ We use class number/size constrained clustering to organize each of the classes.
 - If we know nothing about either class size, number of classes (most general case) ⇒ We must rely on the robustness of standard hierarchical clustering algorithms.

Again we emphasize that our goal was not the development of novel feature extraction or clustering techniques but a new image retrieval methodology. Once a decision is made about the terms that compose an annotated image document, we can apply the LSI methodology and explore new aspects of image retrieval.

3.3. Methodology benefits for IR. All the standard query types available in IR systems can be translated quite easily to query vectors in the LSI framework. For a query by example image we process it by segmentation, classification of regions and user annotation, then we construct a query vector in the same fashion as for the images populating the database. Color proportion queries can be expressed using the global color terms (cf. Table 1). Queries by keywords are expressed in the obvious way using either global or local terms. Queries by sketch are expressed similarly to queries by example. So this methodology covers the usual query paradigms in a generally efficient manner.

Other query mechanisms can be implemented, typically searching for specific regions and not whole images translates in our formulation to searching for shaped terms that are close to a query. In much a similar way we can search for visual terms that are associated with a specific textual attribute. For example “sky” should be related to bluish colors, cloudy textures, and usually have a roughly horizontally elongated shape. Similar queries could not be implemented with other feature-space oriented retrieval methodologies.

The results are generated by comparing the projected query vectors using any of a variety of metrics, eventually even metrics that adapt through user feedback. These results, documents and terms, are shipped to the interface which can display them in a variety of ways. Simple lists and trees can be displayed showing both images, and annotations. Alternatively images can be displayed in a “galaxy” using the first coordinates of the LSI projection. The closeness of the results (terms or images) indicates their relevance or similarity to the query and among each other. Showing the user this structure of the database, centered around the query location, helps her/him to target the search refinement of the next iteration in an interactive search task. Allowing a fluid and intuitive navigation of this space also gives the user the “feel” of the notion of similarity the system is employing. We have implemented such visualizations (Figure 3) [PDVP00] but have not as yet tested them with LSI based projections.

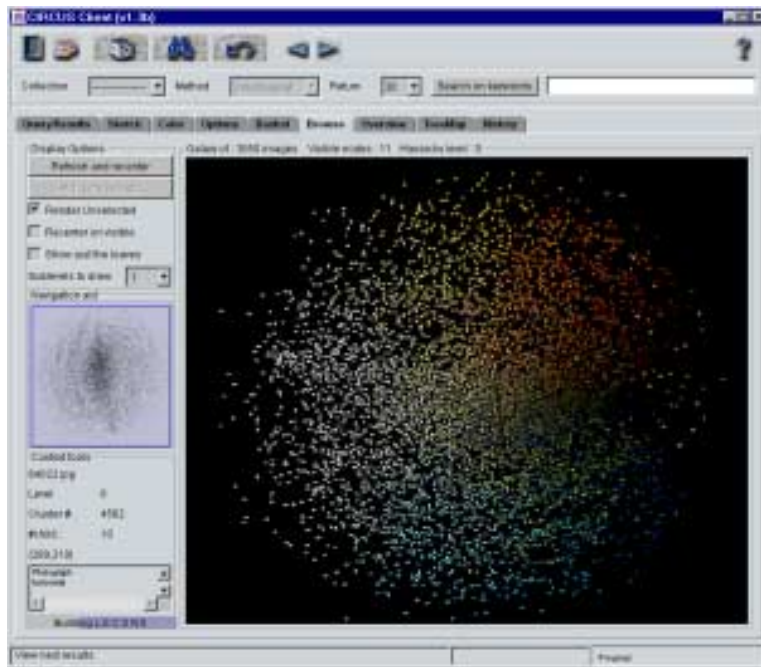


FIGURE 3. The images are projected in a 2-D space that preserves distances.

4. EXPERIMENTS AND RESULTS

This section presents some preliminary results and their interpretations. But, first we have to define some performance evaluation statistics used in the following subsections.

First consider a database having N documents and a query having ideally R relevant results in the database. After query processing the system returns S results in an ordered list. Define r the number of results, amongst the S , which are relevant.

Precision: is the ratio of relevant documents retrieved to all *retrieved* documents : precision = $\frac{r}{S}$.

Recall: is the ratio of relevant documents retrieved to all *relevant* documents : recall = $\frac{r}{R}$.

$n\%$ **Better than random:** Consider the case when recall= 1.0 meaning all the relevant results have been retrieved (in the worst case all documents have been retrieved $S = N$). Consider the relevant results after the first R positions and define l as the rank of the last relevant result ($l \geq R$). We say that a distribution of results is $n\%$ better than random if l falls within the $100 - n\%$ percentile of the distribution, had the remaining relevant documents been uniformly distributed among the remaining $S - R$ retrieved documents.

4.1. Proof of concept data-sets. The first and simplest data-set considered is composed of artificial images of 6 simple objects (eg. squares, triangles). The objects are placed on a uniform background with added Gaussian noise ($\sigma = 0.1$). The objects are colored in one of 16 colors and the same quantity of noise is added. Each images is composed by scaling, rotating and then pasting one to five objects at random positions. The annotation consist of words describing the shapes (eg. "small square", "large tilted rectangle"). Furthermore some artificial synonyms are used at random instead of the original words describing the shapes (eg. "square" = "stamp", "rectangle" = "field").

We construct the co-occurrence matrix by applying segmentation, followed by direct matching of shape moments to template moments. Color attributes are then also directly compared to the color palette used for building the database. The errors in segmentation and classification never produced more than 3% of miss-identified shapes.

We generate different sized databases with varying amount of pasted objects. We then pick a relevant group of images (eg. all triangles shape or annotation) and issue either textual, visual or combined queries. The Table 2 summarizes the results. The words in quotes "" denote textual queries whereas without quotes they denote a shape or visual query.

TABLE 2. Results for artificial databases.

$N = 200, M = 320, 817$ objects (square : $R = 27$)					$N = 300, M = 480, 1303$ objects (square : $R = 56$)				
	query terms	prec.	S^2	% BTR		query terms	prec.	S	% BTR
q1	"square"	0.79	34	95.9	q5	"square"	0.72	78	90.1
q2	square	0.87	31	97.7	q6	square	0.84	67	95.5
q3	"stamp"	0.6	45	89.6	q7	"stamp"	0.71	79	90.6
q4	"square" & square	0.87	31	97.7	q8	"square" & square	0.86	65	96.3
$N = 200, M = 320, 817$ objects (rectangle : $R = 34$)					$N = 300, M = 480, 1303$ objects (rectangle : $R = 62$)				
	query	prec.	S	% BTR		query	prec.	S	% BTR
q9	"rectangle"	0.76	45	93.4	q13	"rectangle"	0.77	81	92
q10	rectangle	0.83	41	95.8	q14	rectangle	0.85	73	95.4
q11	"field"	0.69	49	91	q15	"field"	0.72	86	89.9
q12	"rectangle" & rectangle	0.81	42	95.2	q16	"rectangle" & rectangle	0.83	75	94.5

We can note several interesting facts:

- First of all the method allows to retrieve effectively a large proportion of relevant material using either visual or textual cues. The slightly better performance of visual features probably comes from the larger number of visual "terms". In a more realistic data set the number of unique words will probably be larger than the number of unique visual features.
- The average precision on all queries is 0.78. We compare this to a simple retrieval method using weighted euclidean distance between shape moment invariants. This simple approach achieves average precision of 0.87 (again measured at 1.0 recall). At first inspection our results look worse. However if we consider only the queries involving pure shapes (q2,q6,q10,q14) our method achieves a precision of 0.85.

- On average the queries involving only textual annotation (q1, q3,q5,q7,q9,q13,q11,q15) achieve 0.76 precision, while effectively tackling the problem of synonymy (“square”- “stamp”, “rectangle” - “field”).

4.2. Increased complexity data-sets. The second data set consists of a similar construction of artificial images, but using more complex shapes, slightly more complex backgrounds and richer annotation. We have 20 template objects representing various snacks and 5 different textured backgrounds. Again we construct images by pasting different templates in different sizes, rotations and positions. The images are segmented and the shapes matched against the templates. With this set we achieve a maximum miss-classification rate of 6% for the shapes. Additionally we bias the generation of the documents so that $P(\text{hot dog}|\text{burger}) = P(\text{burger}|\text{hot dog}) = 0.75$. In this way we create a more profound notion of synonym including all aspect visual and textual. We investigate the precision of retrieving hot dogs when burgers where the catual query. Table 4.2 presents the results.

TABLE 3. Results for complex artificial databases.

$N = 300, M = 480, 1211$ objects (burger: 14, hot dog: 16, both: 11)							
	query	p(burger)		p(hot dog)			
q1	“burger”	0.73	0.53	(96%BTR)		q2	“hot dog”
q3	burger	0.77	0.34	(90%BTR)		q4	hot dog
q5	”burg.” & “hot”	0.71		0.74		q6	burg. & hot
							p(burger)
							p(hot dog)

The low precision values when we try to retrieve documents using synonym terms are however on average 92% better than if we had returned these documents in a random order. This fact alone justifies the application of LSI.

4.3. Real data-sets. The last data set is a real-world collection of 6100 images from the COREL CD set. These images are accompinied by a annotation of a few words which are not necessarily associated with objects visible on the image. Because of the size, number and complexity of the images we have not yet attempted to segment themand apply exactly the same method described above. Insted we consider the images as a single object, using the annotation as a global term (not associated to a shape of a region). This sort of approach was already investigated in our previous work [Pec97, PDAV98]. Several dense clusters have been identified during various experiments on this data-set. These were used as ground-truth for a query initiated with any keyword or sample image in the group. Table 4 illustrates the performance results and Figure 4 presents a sample result using both visual and textual query elements.

TABLE 4. Results for real-world database.

$N = 6100, M = 10807, 10807$ objects (scuba : 100, horse : 75, boat : 112)					
	query	precision			
q1	“scuba”	0.81		q2	scuba (average)
q3	“scuba” & scuba (best ¹)	0.84		q4	“scuba” & scuba (worst)
q5	“horse”	0.47		q6	horse (average)
q7	“horse” & horse (best)	0.69		q8	“horse” & horse (worst)
q9	“boat”	0.53		q10	boat (average)
q11	“boat” & boat (best)	0.71		q12	“boat” & boat (worst)

¹best/worst : the sample image yielding best/worst performance.

We can see that simply visual queries generally perform better than simply textual queries. This is probably due to the small size of the annotations in this database. On average precision increases when joint visual and textual queries are submitted (although choosing a bad sample leads to quite significant deccreas in performance). The scuba class of images is very dense both visually and textually and we see it chieves high accuracy. The least constrained category of horses has the worst performance, as could be explected.

Globaly the results are comparabale to published performance evaluations of other image retrieval methods. We cna thus say that LSI-based retrieval compares well with standard approaches. It offers on the other hand exciting new query abilities and effectively integrates textual *and* visual queries into a single flexible framework.



Query for “horse” and with sample in top left corner.

Query for “boat” and with sample in top left corner.

FIGURE 4. Results for joint textual and visual query elements. The white dot (added) shows images that don't have the keyword in their annotation although they do contain the referred object.

5. CONCLUSION

We have introduced a novel method for image retrieval that seamlessly integrates textual and visual descriptors in a single indexing structure. The benefits from this approach are threefold: First the interaction between visual (syntactic) and textual (semantic) descriptors of an image can be explored. Then the performance achieved is comparable to state-of-the-art retrieval systems. Finally the flexibility of query paradigms can be vastly enhanced. The only drawback of the approach is its computationally intensive nature.

Certain issues still remain open to investigation. Further testing of the method with real-world data accompanied by ground-truth information is necessary. Eventually one should employ more sophisticated methods for shape matching than moment invariants. Similarly a more robust clustering than hierarchical clustering should be applied in the general case. We plan to address some of these issues in the near future.

REFERENCES

- [BDL95] Michael W. Berry, Susan T. Dumais, and Todd A. Letsche, *Computational methods for intelligent information access*, Proceedings of Supercomputing'95 (San Diego, CA), ACM/IEEE, December 1995.
- [DDF⁺90] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman, *Indexing by latent semantic indexing*, Journal of the American Society for Information Science **41** (1990), no. 6, 391–407.
- [FSN⁺95] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Stelle, and P. Yanker, *Query by image and video content: The QBIC system*, Computer (1995), 23–32.
- [LC98] M et al La Cascia, *Combining textual and visual cues for content-based image retrieval on the world-wide web*, Proc. of the IEEE Workshop on Content-based Access of Image and Video Libraries, 1998, pp. 24–28.
- [MM98] W. Y. Ma and B. S. Manjunath, *A texture thesaurus for browsing large aerial photographs*, Journal of the American Society for Information Science **49(7)** (1998), 633–648.
- [MPM⁺00] Wolfgang Müller, Zoran Pecenov, Henning Müller, Stephane Marchand-Maillet, Thierry Pun, David Squire, Arjen P. De Vries, and Christoph Giess, *Mrm: An extensible communication protocol for interoperability and benchmarking of multimedia information retrieval systems*, SPIE Photonics East - Voice, Video, and Data Communications (Boston, MA, USA), nov 5–8 2000.
- [PDAV98] Zoran Pecenov, Minh Do, Serge Ayer, and M. Vetterli, *New methods for image retrieval*, ICPS'98, International Congress on Imaging Science, 1998.
- [PDVP00] Zoran Pečenović, Minh Do, Martin Vetterli, and Pearl Pu, *Integrated browsing and searching of large image collections*, International Conference on Visual Information Systems (Visual 2000) (Lyon, France), nov 2–4 2000.
- [Pec97] Zoran Pecenov, *Intelligent image retrieval using Latent Semantic Indexing*, Master's thesis, Swiss Federal Institute of Technology, Lausanne, Vaud, April 1997.
- [SC97] J.R. Smith and Shih-Fu Chang, *Visually searching the web for content*, IEEE Multimedia **4(3)** (1997), 12–20.
- [SM00] Jianbo Shi and J. Malik, *Normalized cuts and image segmentation*, PAMI **22(8)** (2000), 888–905.
- [Sto96] R.E. Story, *An explanation of the effectiveness of latent semantic indexing by means of a Bayesian regression model*, Information Processing & Management **32** (1996), no. 3, 329–344.
- [SWS⁺00] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, *Content-based image retrieval at the end of the early years*, PAMI **22(12)** (2000), 1349–1380.