Bringing Semantics in Word Image Retrieval

Praveen Krishnan and C. V. Jawahar
Center for Visual Information Technology, IIIT Hyderabad, INDIA
Email: praveen.krishnan@research.iiit.ac.in, jawahar@iiit.ac.in

Abstract—**Performance of the recognition free approaches for document retrieval, heavily depends on the exact or approximate matching of images (in some feature space) to retrieve documents containing the same word. However, the harder problem in information retrieval is to effectively bring semantics into the retrieval pipeline. This is further challenging when the matching is based on visual features. In this work, we investigate this problem, and suggest a solution by directly transferring the semantics from the textual domain. Our retrieval framework uses (i) the language resources like WordNet and (ii) an annotated corpus of document images, to retrieve semantically relevant words from a large word image database. We demonstrate the method on two languages — English and Hindi, and quantitatively evaluate the performance on annotated word image databases of more than a Million images.**

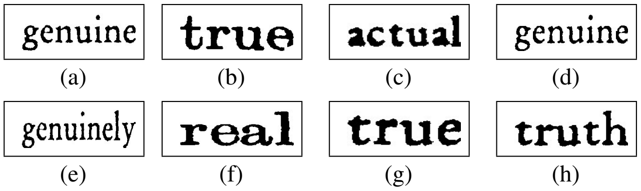
Keywords—Word Image Retrieval, Semantic Indexing, Bag of Words

I. Introduction

Recognition free methods have emerged as successful paradigms to retrieve relevant information when the query is a word or even a phrase [1], [2]. They model the problem as an efficient feature matching scheme in a large database of document images. This category of solutions (often called word spotting) is seriously limited by the ability to match in a feature space. Focus of research in this direction has been to design novel features and representations (e.g. profile features [3], BoW histograms [1], [2], GFG [4]), appropriate distance functions (eg., Euclidean [5], DTW [3], Earth Movers Distance [6]) and efficient indexing schemes (eg. inverted index [1], LSH [5]).

We observe that there are two complementary directions to this research which could make these word spotting solutions more useful and powerful. (1) Semantics has to be brought into the word spotting pipeline to retrieve visually dissimilar but semantically similar word images. For this linguistic resources available in textual form, need to be effectively combined with the feature based techniques. (2) A word spotting system should be "trainable" with a smaller collection of labeled examples. This will make the system capable of extending the performance to a larger collection of unlabeled data. In this work, we make an attempt in these two directions. We model the problem of semantic retrieval by transferring the semantics from the textual domain to the feature domain. Our retrieval framework uses (i) the language resources like WordNet and (ii) an annotated corpus of document images, to retrieve semantically relevant words from a document image database.

Semantics have always played a key role in information extraction and retrieval solutions. Users express the need of information by formulating a query to a search engine, which often fails to retrieve truely relevant information. This is mainly because of the vocabulary mismatch between the query and the documents present in the corpus. The key aspect of a semantic indexing algorithm is to allow the user to search and retrieve results using concepts which are latent or hidden inside a query or a document. In text domain, there are many algorithms such as Latent Semantic Indexing (LSI) [7], Probabilistic Latent Semantic Indexing (PLSI) [8] and Latent Dirichlet Allocation (LDA) [9] which represent the document in a latent space to extract the semantics.



(a) (b) (c) (d)
(e) (f) (g) (h)

Fig. 1: Given a sample query "genuine", we retrieve its semantically related words like "true", "actual" etc.

There are also few works in the field of semantic indexing for document images. They all primarily extend the notion of semantic indexing from text to a feature space. In [10], the authors use LSI in the visual feature space. They represent documents as a collection of terms which are the quantized form of actual word images using Geometric Feature Graph (GFG) [4]. Meshesha and Jawahar [11], proposed a morphological matching scheme for word images which can retrieve semantically related words which are visually similar, except the changes in the prefix and suffix regions. Here, the semantic relationship is limited to some of the word form variations. All these methods define semantics in terms of visual similarity instead of linguistic similarity. In this work, we define a broader notion for semantics using "synonyms". We focus on semantic access to the individual pages or document image with a better definition of semantics. We start with a compact semantic relationships of words using WordNet [15] and validate the semantic retrieval in the feature space. In short, we do not aim at learning semantic relationships from large textual or image corpus, but enable the semantic retrieval by exploiting the semantic knowledge available in textual representations, which could even be hand coded.

Our objective is to extend the recognition free retrieval solution by exploiting the advances that have taken place in the textual domain. However, transferring the results available in the text domain to the image (feature) domain is non-trivial. For this purpose, we use a small collection of annotated documents. This allows seamless transfer of results from textual domain to image domain. For the verification of the approach, we consider a linguistic resource (WordNet) available in textual form and demonstrate how this can help in

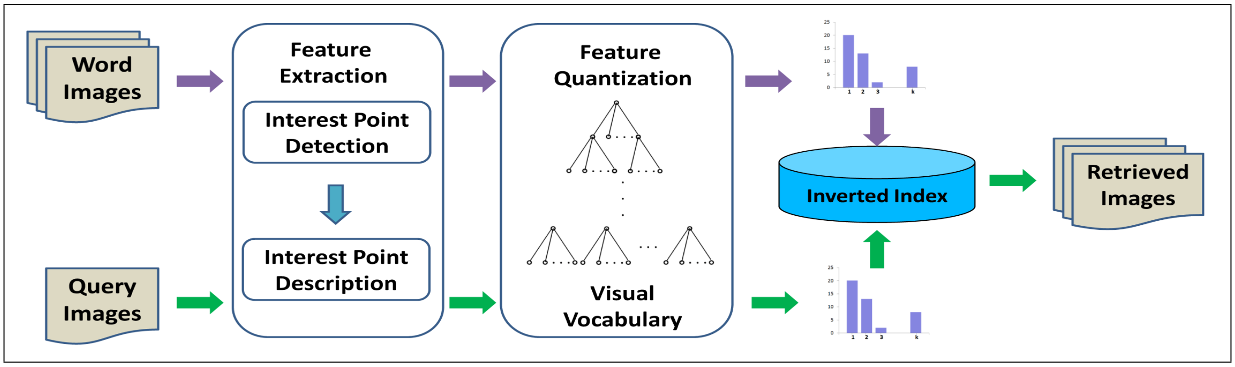


Fig. 2: Overview of BoW based retrieval and indexing. The top half shows the offline process of indexing of word images present in the corpus. The bottom half shows the retrieval of a query image from the inverted index.

retrieving semantically related words even when a recognizer is not available. For example, Fig. 1 demonstrates one such example of the retrieved images using the proposed method. The actual query image is "genuine" as shown in (a) and its semantically related retrieved results are shown in (b-h). Thus, our method retrieves semantically related words and enables more meaningful document retrieval. We validate our method on two different languages — English and Hindi. We quantitatively and qualitatively measure the performance in Section IV.

II. BoW Based Document Image Retrieval

Document image retrieval methods could be divided into two broad categories. The first category is called the recognition based approaches, where an Optical Character Recognition (OCR) solution converts the document image into text which can be indexed. Many of the semantic retrieval schemes popular in textual domain are directly applicable in this category. The second category is called as the recognition free approach or Word Spotting [3] where the document images are represented using visual features and a comparison is done with the help of an appropriate distance metric. Bag of Words (BoW) representation has emerged as a very powerful scheme for word spotting [1] [2]. BoW model, which was originally designed for textual domain has been adapted for natural images by learning a dictionary/vocabulary from a set of examples [12]. This method is effectively used for image retrieval in many domains.

In BoW model, each word image is represented by an unordered set of non-distinctive visual words present in the document, regardless of the spatial order. Visual words are often a quantized description of a reliable feature representation (eg. SIFT). The set of these discrete visual features is called vocabulary. The vocabulary comprises of all visual words present in the corpus, which is learnt offline in an unsupervised manner. Every word image is finally represented with the help of frequency of occurrences (histogram) of the terms in the vocabulary as $[h_1, \dots, h_k]$ where h_i is the number of occurrences of the i^{th} visual word in the image and k (which is set empirically) is the vocabulary size. Since the number of interest/key points vary between images (due to size, scale etc.), the BoW histograms are normalized to have a unit L1 norm. We have used Harris corner detector for interest point detection which is shown to be useful in image representations [13]. Vocabulary creation is done using computationally efficient Hierarchical K Means (HKM) [14] clustering. Fig. 2 shows the typical procedure of indexing and retrieval using BoW based representations. As the offline step, visual vocabulary is computed and all the word images present in the corpus are indexed. As the online step, the query image is converted into histogram of visual words and inverted index returns the similar word images. More details on how the BoW model can be used for document retrieval can be seen in [1] and [2].

One of the main limitations of BoW is the absence of semantic knowledge in the pipeline. BoW operates purely in the visual domain where there are no explicit relations of human level semantics linked with the visual words. While in the text domain, some level of semantic knowledge can be recovered using linguistic resources. Note that a direct LDA or pLSA in this feature space does not result in the human perceivable semantics, which is one of our objectives.

III. Word Image Semantic Indexing

There has been a great interest in the textual community in building lexical databases such as WordNet [15] which stores the lexical and semantic relations between the words of a language. Inspired by WordNet created for English, many other language groups have designed its WordNet to support language projects of their interest. In this section, we explain the proposed method for associating semantic relationships among the word images. We use a small subset of the entire corpus as ground truth (GT) where we explicitly know the text annotation of a word image along with its BoW representation. We use an inverted index which captures the semantic relationships between the word images present in the GT. We call this a semantic index. Essentially when we search a query in the form of histogram of visual words, the semantic index returns the visually similar word images present in the GT along with the images of the synonym words. The retrieved list contains the true occurrences along with some wrong images which are visually similar to the query but semantically very different. In other words, the synonym set of these retrieved results could be totally incoherent visually. Under such circumstances, it is very important to filter out the results and pick the correct group for that particular query. This is done in the group assignment stage as explained in the next section.

A. Group Assignment

The retrieved results computed from the semantic index is divided into groups. Each group contains different instances/images of the same word called as the parent image along with its synonyms called as the child images. In order to associate an unknown query image to the parent of one of the groups, we use a SIFT [16] based geometric verification. We divide the images into three parts (set empirically) horizontally. Each part is given to a Harris corner detector [13] and the corresponding SIFT descriptors are extracted. A matching score as shown in Equation 1 is obtained between the corresponding parts of query image (Q) and the parent image (I). Scores across all the parts are combined using Equation 2. Here $Score_{full}$ corresponds the SIFT matching score for the entire word image while $Score_i$ corresponds to the i^{th} sub-part of the images.

$$Score(I, Q) = \frac{\#MatchPoints}{\#SIFT(I) + \#SIFT(Q)} \tag{1}$$

$$TotalScore(I, Q) = Score_{full} + \frac{1}{3}\sum_i Score_i \tag{2}$$

We iteratively do this process for top-n groups and select the group with highest score.

B. Query Formulation

The group assignment module returns a set S of possible synonym word images. A naive way to proceed to next step will be to search using each image in S and merge the final results. A more effective and cleaner method is to formulate the queries in such a way that each query will symbolize a different synonym, font, style and presence of degradation. In this way we can also reduce the number of queries to be expanded by only preserving the discriminative ones. Here we use a K-means algorithm to cluster the word images in its visual vocabulary space. Each word image is represented as a k dimensional term vector where k is the vocabulary size. The term vector is sparse with only few of the visual words being active. The number of clusters is chosen according to the number of unique synonyms present in S. Each cluster will give a representation of the query for all the word images assigned to it. Hence we modify the original set S into a reduced set R which is a more compact representation.

C. Query Expansion & Re-Ranking

Query expansion [17] is a process of modifying the original query to improve the performance of search and retrieval system. This is usually done by expanding the query to contain more information in terms of synonyms, morphological variations and spelling normalization etc. Finally the expanded words are re-queried to the system by giving appropriate weights. In this work we are focusing on the retrieval of synonyms word images along with similar images of the original query. As mentioned in the previous section, the query formulation module will return discriminative queries $(Q_1, Q_2 ... Q_{|R|})$ and each one of these will be queried in the BoW index built on the entire corpus. Let us denote S_{mn} as the retrieval score for I_m^{th} retrieved image in Q_n^{th} retrieval list. Each of the retrieved list need to be merged into one single ranked list which should be sorted in a way so that the top images are the correct matches of individual retrieved lists. The task is not trivial as we do not know the true matches of the individual lists and also setting one global threshold for all retrieved lists will not help much. To resolve this we apply a Longest Common Sub-sequence (LCS) based re-ranking [2] for the word images of each lists and combine it together using the corresponding scores obtained. LCS based re-ranking enforces spatial constraints on the visual terms being matched. We use a similarity score [2] shown in Equation 3 and the scores are normalized across all expanded queries.

$$Sim(I_m, Q_n) = \lambda Cover(I_m, Q_n) + (1 - \lambda) Config(I_m, Q_n) \tag{3}$$

$$Cover(I_m, Q_n) = \frac{\sum_{i \in I_m \cap Q_n} w_i}{\sum_{j \in I_m} w_j} \tag{4}$$

$$w_i = \frac{1}{log(f_i + 1)} \tag{5}$$

$$Config(I_m, Q_n) = \frac{\sum_{i \in LCS(I_m, Q_n)} w_i}{\sum_{j \in Q_n} w_j} \tag{6}$$

Here $\lambda \in [0, 1]$ in Equation 3 is a weighting parameter between the coverage and configuration scores as given in Equation 4 and 6. The coverage score gives a higher weightage to visual words which occur less frequently compared to the ones which have higher frequency in the corpus. This is shown in Equation 5 where w_i is the weight of the i^{th} visual word in the visual vocabulary. The concept of giving weights is analogous to the removal of stop words in the text domain where the stop words give misleading information about that document. To calculate the configuration score as given in Equation 6 the visual words are projected on the X axis of the image plane. The LCS between the query (Q_n) and retrieved image (I_m) is extracted and $Config$ score is calculated. LCS uses an dynamic programming based algorithm to return the longest subsequence common to all sequences between the pair of input strings. The modified scores for each retrieved image is given as shown in Equation 7.

$$QES_{mn} = \beta \times S_{mn} + (1 - \beta) \times Sim(I_{mn}, Q_n) \tag{7}$$

Here $\beta \in [0, 1]$ is a weight factor, QES_{mn} is the query expansion score which combines the original retrieval score with the similarity score. The final ranked list is a naive fusion across all retrieved images sorted by query expansion scores.

IV. Results & Discussions

In this section, we demonstrate the utility of the semantic retrieval and demonstrate its importance for searching in document images. We show our results on English and Hindi language documents. We use a dataset of 0.25M words in English and 1.5M words in Hindi, details are given in Table I. To evaluate the results quantitatively, the entire corpus is annotated using [18] along with its ground truth. The semantic index is created for each language using a subset of the entire corpus (approximately 20%). We use the corresponding WordNets also for these languages. In WordNet, the words are grouped into sets of synonyms called synsets which are related to other synsets through well known lexical and semantic relationships. Some of these relations are Hypernymy (superset relation), Antonymy (opposite meaning) etc.

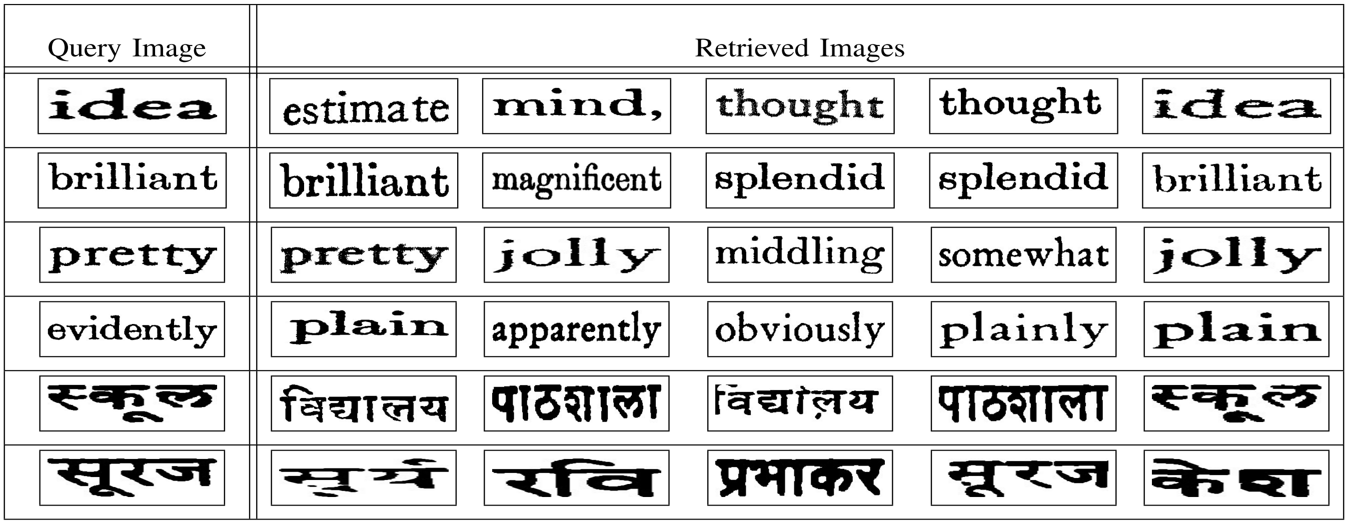


Fig. 3: Qualitative Results: On left column query image is shown and the right column shows the images from Top-25 retrieved results in a ranked order.

TABLE I: Datasets used for the experimentation.

Language	Annotation	#Books	#Pages	#Words
English	Yes	5	900	0.25M
Hindi	Yes	33	4500	1.5M

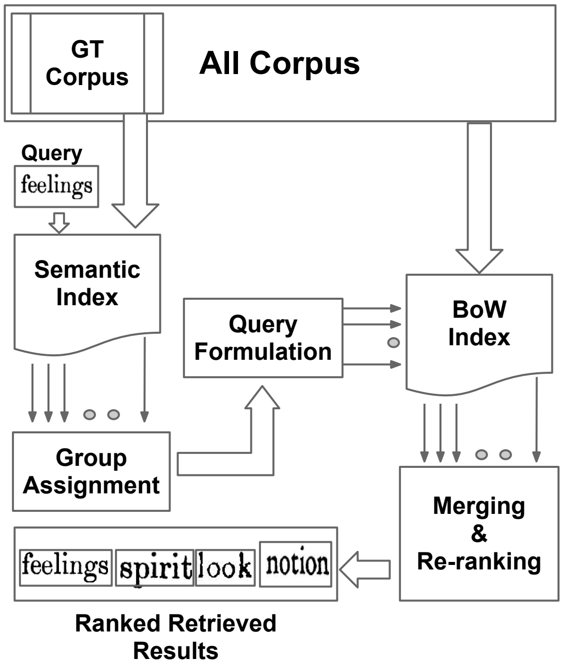


Fig. 4: Search architecture showing the indexing and retrieval. The input to the system is shown at upper left side as the query image given to semantic index. The results are given in a ranked manner.

TABLE II: Performance statistics of the proposed method on English and Hindi datasets.

Dataset	#Query	Proposed Method	
		mAP	mPrec@10
English	70	65.10	94.69
Hindi	80	50.40	88.20

Table II shows the quantitative performance of the proposed solution. Queries considered for the experimentation, are mostly content words. Performance is measured in terms of mean average precision (mAP) and mean precision@10 (mPrec@10). The mAP is the mean of the area under the precision-recall curve for all the queries while mPrec@10 shows how accurate the top 10 results are. These are validated across a ground truth prepared from the occurrences of query word and its synonyms in the annotated corpus. As shown in the Table II, the proposed method gives a mAP of 65.10% for English and 50.40% for Hindi. The mPrec@10 result for English is 94.69% and 88.20% for Hindi. On an average, queries for both languages have 5 synonyms each. Fig. 5 shows precision v/s recall curve on test queries. We do not observe 100% recall since we have limited the number of retrieved list to a lower value. It is also observed that Hindi queries give poor performance when compared to English. Hindi script is complex in appearance. Another reason for the poorer performance for Hindi is that the size of Hindi database was much larger than English. Fig. 3 shows some qualitative results retrieved for the given query images. For every query image we show the synonym word images retrieved along the same word image. We also observe that there is some amount of print variation and degradation in the retrieved images which shows the robustness of BoW system along with re-ranking.

A. *Search Architecture*

Fig. 4 shows the entire architecture of the retrieval solution. The GT corpus is a small subset of the entire corpus. As an offline process, we created the semantic index on the GT corpus with the help of WordNet [19] [15]. The entire corpus of word images is represented as histogram of visual words and indexed into a BoW index. The BoW is calculated on the visual vocabulary size of $k = 29K$, which is fixed empirically.

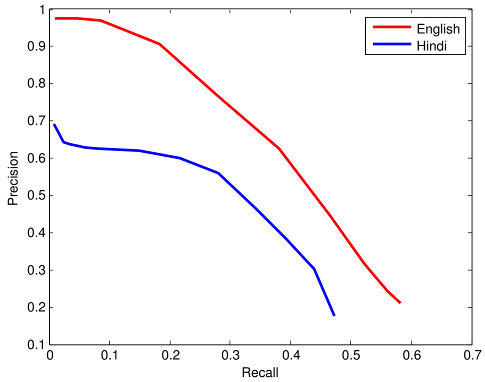
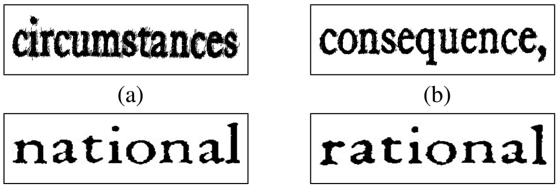


Fig. 5: Precision Recall curve for test queries of English and Hindi languages.

In online processing, query image is given to the semantic index, which returns multiple sets of word images. The group assignment module filters it and picks the most relevant group to be associated. We further refine the word images of the selected group by extracting/creating the discriminate queries using the query formulation module. Each of the formulated query is given to the BoW index created on the entire corpus. Each of the individual results are merged into one list which contains the similar word images to the query image and its synonym images from the entire corpus. The current work uses WordNet as the source of linguistic knowledge. To learn corpus specific semantics, one would require a statistical language model in the pipeline. Our method is still applicable and it will be handled by the semantic index in a similar fashion.

We have used Lucene [20], a popular open source search engine for all the indexing purposes. Lucene creates optimized indexes, which are split into sub-indexes and can be searched independently. This helps to maintain constant searching time even for large index sizes. The index size for English and Hindi datasets are around 1.2GB and 3.2GB respectively.



(a)

(b)

(c)

(d)

Fig. 6: Failure Case Images. On left (a & c) we show the query image and right images (b & d) show the respective incorrect assignments.

B. *Failure Cases*

The proposed method relies on two critical operations, group assignment and merging of expanded results. However the problem arises due to high degree of visual similarity between the query image and the retrieved images. Fig. 6(a) shows an example of a failure case, where the parent image has been wrongly assigned to (b). This can also happen if the query image is not present in GT/Semantic index. To certain extent, we can use the retrieval scores to know whether group assignment has succeeded or not. In case of low scores, we avoid query expansion and do a normal query retrieval on BoW index. The second challenge comes in robust sorting of the final merged list as shown in Fig. 6(c & d), where both images are highly similar. We also noticed that when the size of database increases (as seen in Hindi), the mAP of the retrieval solution shows some amount of drop. This could be solved with an additional post processing.

V. Conclusion

In this paper, we have introduced an architecture to incorporate semantics into the retrieval pipeline of document images. We bring semantics to the recognition free retrieval solutions. We demonstrate a simple practical framework for transferring the textual knowledge to the visual domain. This is done by exploiting the linguistic resources such as WordNet and an annotated corpus. We also demonstrate our method on two languages English and Hindi with mean Precision@10 around 90%.

Acknowledgment

This work was partly supported by Ministry of Communication and Information Technology, Government of India.

References

[1] R. Shekhar and C. V. Jawahar, “Word Image Retrieval Using Bag of Visual Words,” in *DAS*, 2012.

[2] I. Z. Yalniz and R. Manmatha, “An Efficient Framework for Searching Text in Noisy Document Images,” in *DAS*, 2012.

[3] T. M. Rath and R. Manmatha, “Word Spotting for Historical Documents,” *IJDAR*, 2007.

[4] G. Harit, R. Jain, and S. Chaudhury, “Improved Geometric Feature Graph: A Script Independent Representation of Word Images for Compression, and Retrieval,” in *ICDAR*, 2005.

[5] A. Kumar, C. V. Jawahar, and R. Manmatha, “Efficient Search in Document Image Collections,” in *ACCV*, 2007.

[6] S. Marinai, B. Miotti, and G. Soda, “Using Earth Mover’s Distance in the Bag-of-Visual-Words Model for Mathematical Symbol Retrieval,” in *ICDAR*, 2011.

[7] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, “Indexing by Latent Semantic Analysis,” *JASIS*, 1990.

[8] T. Hofmann, “Probabilistic Latent Semantic Indexing,” in *SIGIR*, 1999.

[9] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent Dirichlet Allocation,” *JMLR*, 2003.

[10] S. Banerjee, G. Harit, and S. Chaudhury, “Word Image Based Latent Semantic Indexing for Conceptual Querying in Document Image Databases,” in *ICDAR*, 2007.

[11] M. Meshesha and C. V. Jawahar, “Matching Word Images for Content-based Retrieval from Printed Document Images,” *IJDAR*, 2008.

[12] J. Sivic and A. Zisserman, “Video Google: A Text Retrieval Approach to Object Matching in Videos,” in *ICCV*, 2003.

[13] C. Schmid, R. Mohr, and C. Bauckhage, “Evaluation of Interest Point Detectors,” *IJCV*, 2000.

[14] D. Nister and H. Stewenius, “Scalable Recognition with a Vocabulary Tree,” in *CVPR*, 2006.

[15] G. A. Miller, “WordNet: A Lexical Database for English,” *Commun. ACM*, 1995.

[16] D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints,” *IJCV*, 2004.

[17] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[18] C. V. Jawahar and A. Kumar, “Content-level Annotation of Large Collection of Printed Document Images,” in *ICDAR*, 2007.

[19] P. Bhattacharyya, “Indowordnet,” in *LREC*, 2010.

[20] “Lucene,” http://lucene.apache.org/.