Computer Vision and Image Understanding xxx (2008) xxx–xxx

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Visual word proximity and linguistics for semantic video indexing and near-duplicate retrieval

Yu-Gang Jiang *, Chong-Wah Ngo

Department of Computer Science, City University of Hong Kong, 83 Tat Chee Avenue, Hong Kong

ARTICLE INFO

Article history:
Received 31 October 2007
Accepted 5 October 2008
Available online xxxx

Keywords:
Visual ontology
Linguistic similarity
Soft-weighting
CEMD matching
Near-duplicate keyframe
Semantic concept

ABSTRACT

Bag-of-visual-words (BoW) has recently become a popular representation to describe video and image content. Most existing approaches, nevertheless, neglect inter-word relatedness and measure similarity by bin-to-bin comparison of visual words in histograms. In this paper, we explore the linguistic and ontological aspects of visual words for video analysis. Two approaches, soft-weighting and constraint-based earth mover's distance (CEMD), are proposed to model different aspects of visual word linguistics and proximity. In soft-weighting, visual words are cleverly weighted such that the linguistic meaning of words is taken into account for bin-to-bin histogram comparison. In CEMD, a cross-bin matching algorithm is formulated such that the ground distance measure considers the linguistic similarity of words. In particular, a BoW ontology which hierarchically specifies the hyponym relationship of words is constructed to assist the reasoning. We demonstrate soft-weighting and CEMD on two tasks: video semantic indexing and near-duplicate keyframe retrieval. Experimental results indicate that soft-weighting is superior to other popular weighting schemes such as term frequency (TF) weighting in large-scale video database. In addition, CEMD shows excellent performance compared to cosine similarity in near-duplicate retrieval.

1. Introduction

Retrieving and classifying video/image content according to their semantics is currently one of the most difficult challenges in computer vision, especially in the presence of within-class variation, occlusion, background clutter, pose and lighting changes. Recently, numerous approaches grounded on sparse local keypoint features are proposed to deal with these variations and have been shown to offer excellent performance. Keypoints are salient patches that contain rich local information about an image or a video frame. The most popular keypoint-based representation is *bag-of-visual-words* (BoW). In BoW, a visual vocabulary is generated through grouping similar keypoints into a large number of clusters and treating each cluster as a visual word. By mapping the keypoints back into the vocabulary, a histogram of visual words is constructed, which forms the feature clue for retrieval and classification.

Existing approaches with BoW mostly evaluate visual similarity by direct bin-to-bin comparison of visual word histograms. This comparison, nevertheless, neglects the linguistics and proximity of visual words. For example in Fig. 1, five visual words derived from three keyframes produce similarity scores equal to zero using histogram bin comparison. However, keyframes I_1 and I_2 are more alike as both visual words v_1 and v_2 depict the visual parts of wheel. In other words, v_1 and v_2 are linguistically related but this cue is buried under the bin-to-bin comparison. While intuitively quite appealing, such linguistics and proximity heuristics have so far been largely under-explored in the literature of video semantic analysis. Most approaches utilize visual words independently and evaluate their individual significances by schemes such as binary or term frequency (TF) weighting [1–4]. In these works, words are stored in separate bins and then compared bin-to-bin with measures such as Euclidean distance or histogram intersection. These approaches therefore cannot capture the inter-word relationship and fail to address the problem illustrated in Fig. 1.

In text information retrieval (IR), the semantic relatedness of text words has been widely explored through general purpose vocabularies such as the WordNet ontology [5]. For instance, the words "car" and "truck" should be more alike than "car" and "dog", as the common ancestor of "car" and "truck" ("motor vehicle") is much lower than that of "car" and "dog" ("object") in WordNet. Motivated by the text-based ontology which can be effectively utilized for describing word-to-word relationships, this paper proposes novel ideas exploring visual linguistics in two different approaches. First, we propose a soft-weighting scheme which captures the inter-word relatedness to evaluate the significance of words to a keypoint. In contrast to the traditional one-to-one keypoint-to-word mapping, the soft-weighting performs one-to-many

* Corresponding author.
E-mail addresses: yjiang@cs.cityu.edu.hk (Y.-G. Jiang), cwngo@cs.cityu.edu.hk (C.-W. Ngo).

1077-3142/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2008.10.002

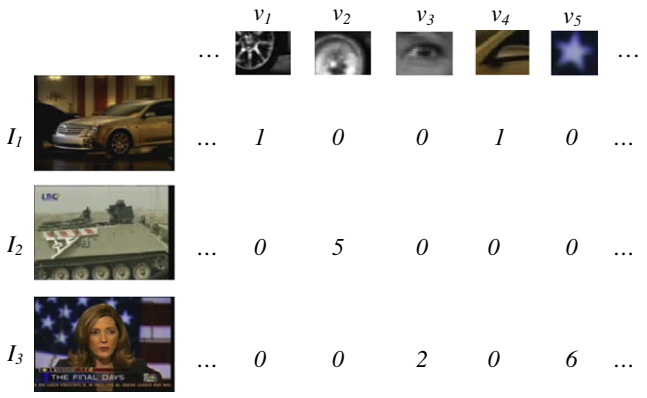

Fig. 1. Histogram representation using bag-of-visual-words.

mapping, while linguistically interpreting the relationship of multiple words in close proximity. The scheme still follows the convention of bin-to-bin comparison, but inter-word relatedness is inherently modeled during the construction of the word histogram. In the second approach, we explore a cross-bin word matching algorithm by constructing a visual-based ontology to capture the is–a relationship of visual words. The ontology is built on top of BoW, as illustrated in Fig. 2(b). Analogous to a text-based ontology such as the WordNet, the visual ontology captures the hyponym (is–a) relationship of visual words.

The two proposed approaches utilize the visual linguistics and proximity from different points of view. The soft-weighting scheme *implicitly* addresses the problem illustrated in Fig. 1. The similarity between I_1 and I_2 is narrowed by softly assigning the words v_1 and v_2 to both keyframes, after observing the proximity of both words in comparing to the extracted keypoints. In other words, the v_1 and v_2 bins of I_2 and I_1 are not empty and thus the linguistic relatedness is still captured during the bin-to-bin histogram comparison. The cross-bin matching approach, on the other hand, *explicitly* evaluates the linguistic relatedness of words. Any two visual words can be directly linked and matched by knowing their ontological relatedness. For instance, by traversing the ontology in Fig. 2(b), the linguistic similarity of different words (e.g., v_2 and v_4) can be rigorously defined based on the distance traveled ($v_2 \rightarrow e \rightarrow c \rightarrow v_4$), depth of their ancestor (node c at depth 1) in the ontology, and the probability of words seen. In this paper, we novelly formulate linguistic cross-bin matching under a constraint variant of the EMD (earth mover's distance) framework, namely CEMD. As an interesting note, the matching framework can be used together with soft-weighting scheme. By recording the soft weights as the signatures of words, a comprehensive version of CEMD can be derived for video analysis.

With the two proposed approaches, we study the effectiveness of visual linguistics for two challenging tasks: (1) semantic video indexing; and (2) near-duplicate keyframe retrieval. Intensive experiments on the TRECVID dataset are conducted to compare the proposed approaches with other existing methods. The rest of this paper is organized as follows. In Section 2, we briefly review the previous related works. Section 3 presents the main idea of constructing visual ontology and modeling linguistic similarity. Section 4 proposes the soft-weighting scheme and Section 5 presents the cross-bin matching by CEMD. Sections 6 and 7 describe our experimental results on video semantic analysis and near-duplicate retrieval, respectively. Finally, Section 8 concludes this paper.

2. Related works

A major challenge in the field of video/image retrieval and classification is building effective features that are invariant to a wide range of variations. The most popular image representation has been global features, which describe images by the overall distribution of color, texture, edge or other visual properties. Features like color histograms/moments and Gabor filters [6] belong to this category. To include spatial information, a keyframe is usually partitioned into either rectangular regions or segments of objects. Features computed from these regions/segments are then concatenated into a single feature vector for retrieval. Such region-based representation has been commonly adopted in tasks such as image annotation [7] and semantic video indexing [4,8,9]. Although popular, these global features face problems in capturing the intra-class geometric and photometric variations.

More recently, there has been a great deal of interest in classifying videos/images based on local keypoints. Keypoints are salient patches that contain rich local information that can be detected using various detectors such as difference of Gaussian (DoG) [10] and depicted by various descriptors such as SIFT (scale-invariant feature transform) [10]. Surveys of keypoint detectors and descriptors can be found in [11] and [12], respectively. There are two common ways of utilizing keypoint features. Keypoints can be matched directly in the descriptor feature space and the matching patterns [13] or the cardinality of matching pairs [14] could be used to estimate the video/image similarity. Alternatively, keypoints can be vector-quantized or clustered into a representation analogous to the bag-of-words representation commonly used in IR. There have been many works in the computer vision community using this vector-quantized keypoint feature, popularly referred to as BoW, for multimedia retrieval [15,16] and classification [1–3,17–19]. Recently, several studies have also been conducted to optimize the representation choices of BoW. These choices include the selection of keypoint detector and descriptor [1,2], clustering algorithm for generating vocabulary [20,21], vocabulary size (number of visual words) [1], and kernel utilized in supervised learning [2]. Nevertheless, none of the works studies the linguistics and proximity aspects of BoW for video analysis.

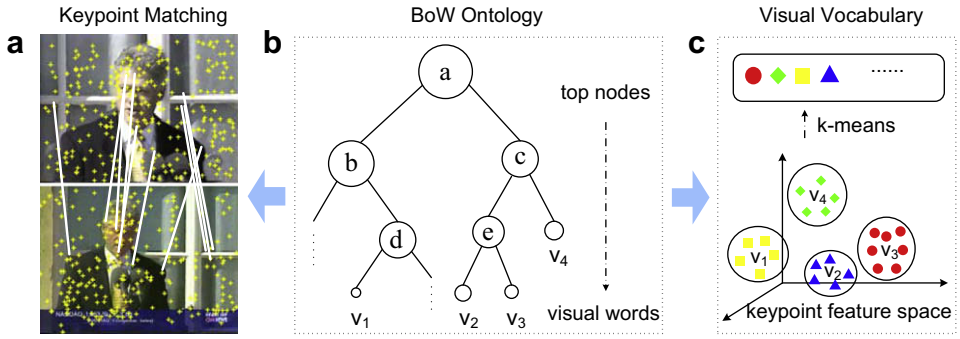

Fig. 2. Visual word ontology (b) as a bridge between expensive keypoint matching (a) and simple bin-to-bin word comparison (c).

Please cite this article in press as: Y.-G. Jiang, C.-W. Ngo, Visual word proximity and linguistics for semantic video indexing ..., Comput. Vis. Image Understand. (2008), doi:10.1016/j.cviu.2008.10.002

3. Modeling visual linguistics

In this section, we first describe the construction and utilization of visual word ontology for linguistic reasoning.

3.1. Bag-of-visual-words ontology

Linguistic reasoning is a useful feature for word disambiguation in IR. For example, the words "car" and "truck" are not matched by comparing characters, but can be semantically linked by "motor vehicle" through the is–a relationship in ontology. Building such an ontology for visual words allows the modeling of word-to-word similarity, while reducing the signal loss during quantization. Since visual words are the outcome of clustering, the is–a relationship can be explicitly extracted by considering the proximity among the clusters of visual words in keypoint feature space.

To mine the is–a relationship, we first employ *k*-means algorithms to cluster the set of given keypoints. The resulting clusters form a visual vocabulary, where each keypoint cluster is treated as a "visual word" in the vocabulary. With this BoW representation, a visual ontology is further constructed by adopting the agglomerate clustering algorithm to hierarchically group two words at a time in the bottom-up manner. Consequently, an ontology which hierarchically encodes the set of visual words in a binary tree is constructed. The leaves of the ontology are words while the internal nodes are ancestors modeling the is–a relationship of words. Note that the ancestors are also treated as words. Fig. 2(b) shows an example of the visual ontology. To model the information richness of a word, the number of keypoints in a node is kept during the construction of the ontology. This information hints at the probability of observing a word that could be utilized for linguistic reasoning. With this information, as shown in Fig. 2(b), each internal node can be viewed as a hyperball and the size of the hyperball, which increases when traversing the tree upward, models the population of visual words that are directly or indirectly related to a particular branch. The depth of a branch, on the other hand, hints at the specificity of visual words, which could also be utilized for a linguistic measure.

The visual word ontology could also be interpreted as a link between the traditional bin-to-bin based BoW comparison [1,2,17,20] and the keypoint-to-keypoint based matching [14,13]. With the ontology, the quantization loss in generating BoW as in Fig. 2(c) can be eliminated since the similarity of any two words can be measured. Furthermore, compared to keypoint matching, there are normally less words to match. For instance, there could be thousands of keypoints available for matching between two keyframes, as depicted in Fig. 2(a). The proposed approach can be viewed as an *extension* of visual vocabulary and an *efficient* version of keypoint matching, where cross-bin matching of words, instead of keypoints, is enabled.

3.2. Linguistic similarity of visual words

With the constructed BoW ontology, linguistic reasoning can be conducted by considering specificity, path length and information content (IC) of visual words. The specificity refers to the depth of a word in the tree. The deeper a word, the more specific the word. Path length, measured by the minimum number of links traverse from one word to the other, indicates the physical distance of two visual words in the ontology. IC is inversely proportional to the probability of a word being seen. Basically, higher probability means a word is frequently observed and less discriminative. Thus, the value of IC, which indicates the significance of a word, should be lower in this case. In the literature of linguistic computing, there exist various measures characterizing the similarity of words [22]. In this paper, by utilizing the BoW ontology, we explore the three most popular measures for reasoning the similarity of visual words.

3.2.1. Resnik

Resnik considers the IC of common ancestors for similarity measure [23]. Denote v_i and v_j as two visual words, Resnik is defined as

$$sim(v_i, v_j) = \mathrm{IC}(\mathrm{LCA}(v_i, v_j)), \tag{1}$$

where LCA is the lowest common ancestor of v_i and v_j. IC is quantified as the negative log likelihood of word probability. The probability is estimated by the percentage of keypoints in a visual hyperball. In the extreme case, the root node "*a*" in Fig. 2(b) has IC = 0 since $p(a) = 1$.

3.2.2. JCN

Resnik has the disadvantage that all words sharing one LCA have the same similarity, despite how far the distances between them. JCN deals with this problem by also considering the ICs of the compared words, defined as [24]

$$sim(v_i, v_j) = \frac{1}{\mathrm{IC}(v_i) + \mathrm{IC}(v_j) - 2 \cdot \mathrm{IC}(\mathrm{LCA}(v_i, v_j))}. \tag{2}$$

3.2.3. WUP

In addition to IC, WUP considers the path length and the depth of words to measure the linguistic similarity [25]:

$$sim(v_i, v_j) = \frac{2 \cdot \mathrm{depth}(\mathrm{LCA}(v_i, v_j))}{\mathrm{len}(v_i, v_j) + 2 \cdot \mathrm{depth}(\mathrm{LCA}(v_i, v_j))}, \tag{3}$$

where $\mathrm{len}(v_i, v_j)$ represents the minimum path length between the words v_i and v_j, and $\mathrm{depth}(\cdot)$ is the depth a word in the BoW ontology.

4. Soft weighting with visual linguistics

We first explore the BoW linguistics introduced in the previous section to *softly* weight the importance of visual words in a keyframe. Word (Term) weighting is a key technique in IR. Two major factors in term weighting are TF (term frequency) and IDF (inverse document frequency). The popular term weighting schemes in IR are summarized in Table 1. Binary weighting assigns 1 (or 0) to indicate the presence (absence) of a word, while TF measures the importance of a word by considering the frequency of its appearance in a document. TF-IDF further degrades the importance of a word if the word also frequently appears in other documents.

In current literature, existing works on visual-based BoW mostly migrate these weighting schemes directly. For example, TF-IDF is adopted in video google [17], binary weighting is employed in [1] for image classification, and most approaches use TF directly [2,3]. Furthermore, the assignment of a keypoint to visual word is normally conducted by measuring the distance between keypoint and cluster centroids. The assignment is basically a one-to-one mapping process, where a keypoint is mapped to a

Table 1
Weighting schemes for bag of visual/text words (t_i: the *i*th word, tf_i: term frequency of t_i; N: the total number of images/documents; n_i: the number of images/documents having word t_i.).

Name	Factors	Value for t_i
Binary	Binary	1 if t_i is present, 0 if not
TF	tf	tf_i
TF-IDF	tf, idf	$tf_i \cdot \log(N/n_i)$

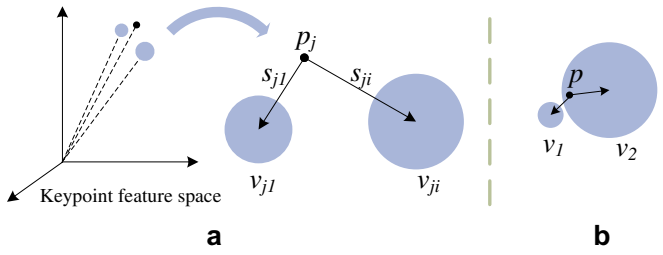

Fig. 3. Soft-weighting scheme: (a) keypoint p_j is *softly* assigned to multiple visual words v_{j1} and v_{ji}; (b) information content (cluster sizes of v_1 and v_2) is used to assess the importance of v_2 to keypoint p.

word with the closest distance to centroid. Visual words are derived directly from keypoint clustering, so the current weighting schemes present several problems. First, the size of visual vocabulary can govern the formation of clusters. By increasing the vocabulary size, two similar keypoints may reside in different clusters. The direct one-to-one keypoint-to-word assignment, which treats each visual word independently, overlooks the inherent linguistics among the visual words. Second, using the frequency of words (i.e., TF) for term weighting is not adequate. For example, two keypoints assigned to the same visual word are not necessarily equally similar to that visual word, meaning that their distances to the cluster centroid are different. TF, which assumes equal contribution for every keypoint assigned to a word, could over-estimate the significance of a word.

To tackle the aforementioned problems, we propose a *soft weighting* scheme to rigorously evaluate the significance of visual words in a keyframe. The new scheme takes into account two key steps: the assignment of keypoint-to-word is one-to-many, and the importance of an assigned word is governed by their linguistic relationship. The intuitive idea is that, the significances of visual words to a keyframe are *softly* weighted depending on the underlying similarity among keypoints and words. Let V as a visual vocabulary with n words. Each keypoint is assigned to k $(k < n)$ nearest visual words in V.[1] The *soft*-weight of a word v_m in an image, denoted as sf_m, is then measured as

$$sf_m = \sum_{i=1}^{k} \sum_{j=1}^{L_i} h(i,j) \times sim(j,m), \tag{4}$$

where L_i is the set of keypoints whose ith nearest neighbor is v_m. The measure $sim(j,m)$ represents the similarity between the keypoint $p_j \in L_i$ and the visual word $v_m \in V$. The $h(i,j)$ is a function to further quantify the importance of $sim(j,m)$, by modeling the relationship among p_j, its first and ith nearest visual words.

Fig. 3(a) illustrates the function $h(i,j)$. Suppose visual word v_{j1} is the nearest neighbor of keypoint p_j, the weight for v_{ji}, which is the ith nearest word of p_j, is measured by $h(i,j)$ defined as

$$h(i,j) = \left(\frac{s_{ji}}{s_{j1}}\right)^{\alpha}, \tag{5}$$

where s_{j1} and s_{ji} are the cosine similarities between keypoint p_j and visual words v_{j1} and v_{ji}, respectively. The parameter α is introduced to amplify the ratio of s_{ji} to s_{j1}. Obviously $\alpha > 0$ since the weight $h(i,j)$ should be monotonically increasing with the similarity s_{ji}.

Eq. 5 only considers the proximity between keypoints and words (cluster centroids), and thus cannot characterize the keypoint distributions such as cluster size. Linguistics can be utilized to adjust the weight $h(i,j)$ by introducing the IC of both words:

$$h(i,j) = \left(\frac{s_{ji}}{s_{j1}}\right)^{\alpha} \times \left(\frac{IC(v_{j1})}{IC(v_{ji})}\right), \tag{6}$$

where $IC(\cdot)$ represents the IC described in Section 3.2. The rationale of using IC can be explained by an example shown in Fig. 3(b), in which the keypoint p resides in the cluster of v_2 but having nearer distance to the centroid of v_1 compared to v_2. To increase the importance of v_2, the $h(i,j)$ in Eq. 6 considers the ratio of two words in terms of the size of their hyperballs.

Soft-weighting offers a new perspective of weighting visual words. One similar piece of work is a recent study by Agarwal et al. [26]. In [26], keypoints are estimated with posterior probabilities through a Gaussian mixture distribution learnt from the descriptor feature space. However, this approach is not scalable to a large dataset because learning the distribution from a huge amount of descriptors is computationally intensive. The visual word proximity is also implicitly modeled in [27,28] using a vocabulary tree, which is constructed by hierarchical quantization of keypoint features. The word weighting scheme of [27,28] is mainly based on TF or its combination with IDF. By assigning weights to both inner nodes and leaf nodes of the tree, the proximity of leaf nodes can also be partially inferred. Comparing [27,28] to soft-weighting, there are two major drawbacks. First, the estimation of word significance is not as accurate as ours because [27,28] are basically based on simple word counting. Second, [27,28] uses both inner and leaf nodes as visual words, resulting in a much higher dimensionality compared to our approach that only uses the leaf nodes. Furthermore, the linguistic aspect of visual words was not explored in these existing works.

5. Linguistic matching with CEMD

A special feature of soft-weighting is that the proximity and linguistics of visual words are determined offline during the process of keypoint-to-word assignment. The similarity of two keyframes can thus be efficiently computed via bin-to-bin comparison of visual words. The linguistic similarity of the visual words, however, is only partially exploited as the ontological factors such as ancestor relationship and path length are not characterized. In this section, we present a cross-bin word matching algorithm that fully utilizes the ontological relationship of visual words for measuring image similarity.

Based on the BoW ontology, two different visual words can always be matched by measuring their linguistic similarity. Consequently, given m words in a keyframe, there is $O(m^2)$ possible matching of words for comparing a keyframe pair. We adopt EMD [29,30] for matching two sets of visual words across bins. The ground distance of EMD is based on the linguistic measure, while the signatures of words are characterized by word weighting.

EMD measures the distance between two weighted point sets as a transportation problem [29]. A point set is normally referred to as a signature. EMD strives to find the minimum amount of "work" to transport the weights from one signature to the other. In BoW, a keyframe P is represented as a signature $P = \{(p_1, w_{p_1}), \ldots, (p_m, w_{p_m})\}$ of m words, where p_i indexes the p_ith visual word in the vocabulary, and w_{p_i} is the corresponding weight (signature). To match P with another keyframe Q of n words, the EMD is computed as

$$EMD(P,Q) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} d_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}}, \tag{7}$$

where the ground distance d_{ij} between word v_{p_i} and v_{q_j} is measured via the linguistic similarity such as JCN. The flow f_{ij}, representing the amount of weight transferred from v_{p_i} to v_{q_j}, is optimized during the transportation.

[1] Empirically we found $k = 4$ is a reasonable setting.

5.1. Constraint-based EMD

While the idea of adopting EMD for exploring linguistic similarity appears interesting, the approach suffers from speed inefficiency. Suppose the number of visual words in two signatures is m, the complexity of EMD is $O(m^3 \log m)$. Considering that there are generally tens to hundreds of visual words in a keyframe, the matching could be computationally intensive. Here, we propose a novel constraint matching by dividing the visual vocabularies into c visual chapters, and consequently enforcing EMD not to match words across chapters. This is equivalent to "distributive matching" where there are c EMDs being performed for each chapter, and then merged as a whole. The idea is based on the fact that visual words are cluster centers in keypoint feature space. Certain categories of visual words (e.g., *people* and *building*) are seldom matched, and thus can be ignored from EMD matching when speed is an issue to consider.

To learn the visual chapters of a vocabulary, we compute a flow matrix **F** by observing the accumulated flows (f_{ij}) of EMD over a set of training examples. EMD will basically create flows among similar words. The matrix **F** hints at the correlation among visual words, and each entry F_{ij} indicates the total sum of flows between two words v_i and v_j. By treating **F** as a similarity matrix, an undirected fully-connected graph is constructed over **F**, where nodes are words and edges represent similarities based on EMD flows. The normalized cut algorithm [31] is then employed to partition the graph into c disjoint sub-graphs. Each sub-graph is treated as a visual chapter of the vocabulary. The visual words with lower amount of flows are expected to stay in different chapters.

With the c chapters of words, the constraint-based EMD, namely CEMD, of two keyframes P and Q is performed by running EMD separately in each chapter and then combined as

$$CEMD(P,Q) = \sum_{i=1}^{c} \left(\frac{S_{P_i}}{S_P} + \frac{S_{Q_i}}{S_Q} \right) EMD(P_i, Q_i), \quad (8)$$

where S_{P_i} is the number of visual words that P has in chapter i, and similarly for S_{Q_i}. Note that $S_P = \sum_{i=1}^{c} S_{P_i}$ and $S_Q = \sum_{i=1}^{c} S_{Q_i}$. For two keyframes with m visual words, the speed of CEMD is improved to $O(c \times (m/c)^3 \log(m/c))$. While the computational complexity is the same as the original EMD, CEMD is practically more efficient because the constant $(1/c)^3$ eliminates many word comparisons.

6. Experiment I: Semantic video indexing

Semantic video indexing is also referred to as high-level feature extraction in TRECVID [32]. The aim is to annotate the semantic concepts of keyframes for video indexing. In TRECVID, this task is generally conducted in a diversified setting where the emphasis usually includes feature extraction, multi-modality fusion, and machine learning on a huge multimedia dataset. In this section, we only focus on the feature level testing. In particular, the performance of soft-weighting on BoW will be verified and compared with other weighting schemes.

6.1. Dataset and experimental setup

We use the TRECVID-2006 dataset, where the training and testing sets consist of 61,901 and 79,484 video shots, respectively. In the experiments, we test the 20 semantic concepts which are selected in the TRECVID-2006 evaluation [32]. The class labels of the training set are provided by LSCOM [33]. We use one keyframe per shot for experiments. Fig. 4 shows example keyframes of the 20 semantic concepts. These concepts cover a wide variety of types, including objects, indoor/outdoor scenes, people, events, etc. Note that this dataset is a multi-label dataset, which means each keyframe may belong to multiple classes or none of the classes, e.g. the example of concept *weather* in Fig. 4 also belongs to concept *map*.

In the experiments, we use DoG [10] to detect keypoints, and 128-dimensional SIFT descriptor [10] to describe keypoints. After generating a visual vocabulary with the k-means algorithm and encoding all the keyframes using BoW histograms, a two-class SVM classifier is trained for each semantic concept using the training set. We only experiment with soft-weighting in this task. As the size of the TRECVID dataset is huge, CEMD which performs expensive cross-bin matching is not tested mainly due to speed consideration. In addition, there is no proof showing that the EMD kernel (and thus CEMD) satisfies Mercer's condition [34]. Thus, instead of adopting CEMD as the kernel of SVM, we use the χ^2 kernel SVM [2] which is computationally efficient and theoretically valid. The parameters in the SVMs are optimized by grid search using cross validation.

The performance evaluation follows TRECVID's standard using the inferred average precision (InfAP) computed over the top

Fig. 4. Keyframe examples of 20 semantic categories in the TRECVID-2006 evaluation.

2000 retrieved shots. The InfAP is an approximation of the conventional average precision (AP). The main advantage of InfAP is that it can save lots of judging effort in generating ground-truth labels for a large test dataset [35].

6.2. Visual word weighting

We first study the sensitivity of the parameter α in Eq. 6. Fig. 5 plots the sensitivity curve. We can see that the curve appears to be stable as the value of α increases and the performance peaks at around 60. With reference to Fig. 3(b), this can be intuitively explained when observing that the distances among p, v_1 and v_2 are close to each other, so a larger value of α is required to amplify the similarity difference among them. This experiment verifies that by considering and amplifying the tri-wise relationship between keypoint and visual words, better performance can be expected. The α is not sensitive as long as the value is large enough to emphasize the similarity difference.

Table 2 compares the performance of four different weighting schemes. The proposed soft-weighting outperforms the other popular weighting schemes across different vocabulary sizes with large margins (improvement ranges from 18% to 150%). This confirms our claim that the visual words are correlated to each other, and by modeling the word linguistics and proximity in the soft-weighting, we can significantly boost the discriminative power of BoW. The vocabulary size is claimed as an important factor of BoW in many other studies [3,15]. Generally, smaller size is preferred to speed up the assignment of keypoints to visual words, which is a problem of nearest neighbor search. Most studies, nevertheless, claim that large vocabulary size can lead to better performance. In our experiment, when testing *binary* weighting, we observe that an appropriate size of vocabulary is about 10,000 (or even larger). An interesting and important observation is that when more sophisticated weighting schemes are employed, the impact of vocabulary size turns to be insignificant, especially for our soft-weighting scheme. This observation is explained by the advantages of the soft-weighting scheme discussed in Section 4. We consider this an important merit of soft weighting: less sensitivity to the size of vocabulary. The merit allows the use of smaller vocabulary size, while maintaining a comparable or even better performance to that of large vocabulary sizes.

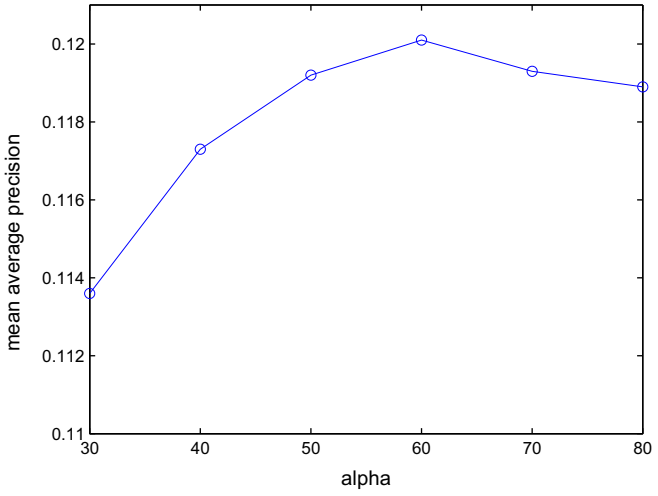

Fig. 5. Sensitivity to parameter α.

Table 2
Performance comparison (mean InfAP) of different word weighting schemes on the TRECVID-2006 dataset.

Vocabulary size	Weighting schemes			
	Binary	TF	TF-IDF	Soft-weighting
500	0.048	0.088	0.081	**0.120**
1000	0.076	0.082	0.078	**0.116**
5000	0.082	0.083	0.089	**0.107**
10,000	0.083	0.090	0.096	**0.113**

Table 3
Performance comparison of semantic video indexing on the TRECVID-2006 dataset.

Approach		Mean InfAP
BoW with soft-weighting		0.120
Local feature systems in TRECVID'06	Mediamill [4]	0.055
	UC Berkeley [36]	0.110
Top 3 performance of TRECVID'06	CMU [8]	0.159
	IBM [37]	0.177
	Tsinghua [9]	0.199

6.3. Performance comparison

With soft-weighting, local features alone exhibit excellent performance on video indexing. In this section, we further compare and analyze the performance of soft-weighted BoW with some state-of-the-art techniques presented in TRECVID-2006.

We first compare our BoW to the local feature approaches of Berkeley [36] and Mediamill [4] teams. The results are shown in Table 3. Compared to [36] who adopted keypoint matching, the soft-weighting gives better performance. In [36], point-to-point matching with exemplars is required. This process is computationally expensive. For instance, the number of comparisons per keypoint in a test sample is as high as 258,200 (1291×200), where 1291 exemplars and 200 keypoints per exemplar were used in their experiment. While for the soft-weighting, the number of comparisons per keypoint is only 500 for a vocabulary of 500 words. The BoW of [4] used late fusion to combine multiple keypoint detectors and descriptors. However, our results show that the soft-weighted BoW using DoG detector alone achieves a mean InfAP of 0.12, which already doubles that of [4]. Finally, we compare our results with the top-3 performance teams (CMU, IBM, Tsinghua) in the TRECVID-2006 evaluation [32]. These systems emphasized not only features, but also multi-modality fusion techniques and machine learning methods. As shown in Table 3, our BoW using local feature alone is comparable to those sophisticated systems such as CMU [8] and IBM [37]. CMU used both visual (color, texture, BoW) and text features, while IBM used global and localized color and textures, motion features, as well as text. Compared to Tsinghua [9] who emphasized rich features and rich classifiers (110 SVMs were used for each concept), our method is more efficient and can be easily scaled up to a thousand of semantic concepts. To the best of our knowledge, the proposed soft-weighted BoW is the best single visual feature for the TRECVID-2006 high-level feature extraction task.

7. Experiment II: Near-duplicate keyframe retrieval

In this section, we experiment the CEMD matching and soft-weighting for near-duplicate keyframe (NDK) retrieval. Near-duplicate keyframes are a group of keyframes similar to each other, but appear differently due to variations introduced during acquisition time, lens setting, lighting condition, editing operation, etc. Fig. 6 shows three pairs of NDKs that have undergone various changes. The task of NDK retrieval is to identify and search the set of near-duplicates for a given query keyframe.

Please cite this article in press as: Y.-G. Jiang, C.-W. Ngo, Visual word proximity and linguistics for semantic video indexing ..., Comput. Vis. Image Understand. (2008), doi:10.1016/j.cviu.2008.10.002

Fig. 6. Examples of near-duplicate keyframes: (a) different acquisition time; (b) lens variations; (c) video editing.

In this task, we begin by testing various settings of CEMD matching, including the choice of ground distance using different linguistic measures, and the choice of word signature (binary, TF, soft-weighting).

7.1. Datasets and experimental setup

We use the Columbia dataset [38] which contains 600 keyframes from the TRECVID-2004 benchmark. There are 150 near-duplicate pairs in this dataset, and we use all of them (300 duplicates) as queries for assessing retrieval performance. The evaluation is based on the probability of successful top-k retrieval [38], defined as $R(k) = N_c/N_a$ where N_c is the number of queries that find their near-duplicates in the top k list, and N_a is the total number of queries. To further strengthen our claim, we also use a larger dataset – the TRECVID-2006 test set containing a total of 79,484 keyframes in the experiments.

Again, we use DoG [10] as keypoint detector and SIFT [10] as the descriptor. For Columbia dataset, a visual vocabulary of 1000 words is built, associated with an ontology of 32 levels. For the TRECVID-2006 test set, we choose the vocabulary of 500 words, which performs best in the experiments of semantic video indexing. The depth of the associated ontology is 23.

7.2. Effect of linguistic similarity measure

First, we compare three linguistic similarity measures: JCN, Resnik (RES) and WUP, with the original EMD as the distance measure and TF as the weighting scheme. Fig. 7(a) shows the performance comparison of the three measures on Columbia dataset. Among them, JCN demonstrates the best performance for considering the ICs of visual words and their ancestor. Resnik, considering only the IC of the lowest common ancestor (LCA), loses the discriminative power as it assigns equal similarity to all the words sharing the same LCA. WUP, utilizing path length and depth, does not show an apparent advantage over JCN, while still performing better than Resnik. We investigate the results and find that this is mainly because the similarity of some words is close to 0 as long as their LCA is near the root (where depth(LCA) = 0), despite the distance between two words. Our finding indeed indicates that the ancestor relationship and ICs of words are the best resources to use in near-duplicate retrieval. To further justify the usefulness of the linguistic measure, we also compare the performances with EMD which uses Euclidean distance between words (cluster centroids) as the ground distance. As shown in Fig. 7(a), Euclidean is not better than JCN, but still outperforms WUP and Resnik. This probably indicates that word distance is an important factor that should not be ignored as in Resnik. While JCN and WUP do not account for word distance, the information can be indirectly inferred from the ICs and path length of words. JCN, when considering ICs of three parties (words and their

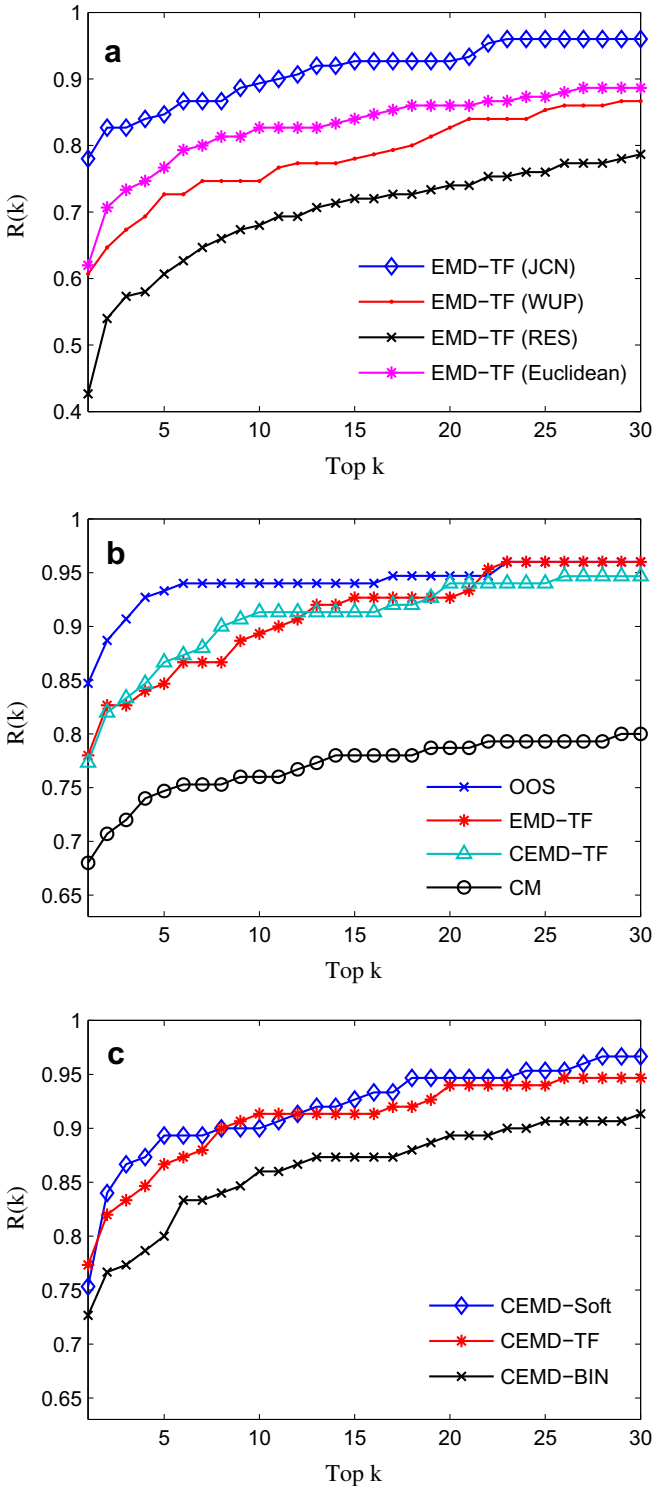

Fig. 7. NDK retrieval performance on Columbia dataset: (a) comparison of linguistic measures; (b) performance of CEMD versus EMD; (c) effect of word weighting on CEMD.

LCA), shows better performance. This is mainly because of the additional consideration of IC which infers cluster size, and LCA which infers the global view of inter-cluster distance and density.

7.3. Comparison of CEMD and EMD

Next, we compare the performance of CEMD and EMD with JCN as the linguistic measure. In CEMD, 300 keyframes are used for

Table 4
Per query NDK retrieval efficiency on the TRECVID-2006 test set.

Approach			Time
Keypoint based	OOS		5 h 44 min
Ontology based	EMD-	Soft	20 h 10 min
		TF	2 h 5 min
	CEMD-	Soft	19 min 58 s
		TF	2 min 59 s
Vocabulary based	COS-	Soft	51 s
		TF	50 s
Fusion	CEMD-TF + COS-Soft		3 min 10 s
Baseline	CM		22 s

training the visual chapters. To avoid over fitting, this training set is randomly obtained from the TRECVID-2005 dataset and is independent of the Columbia and TRECVID-2006 datasets. In our experiment, the vocabulary is empirically divided into eight chapters. Fig. 7(b) shows the performances of CEMD and EMD, in comparison with the one-to-one symmetric (OOS) matching of [13] and block-based color moment (CM). OOS, in contrast to our approach, adopts keypoint matching (without vocabulary) and thus is computationally slow. Nevertheless, since no quantization loss is involved, OOS can achieve the best possible performance of keypoint-based approach for this dataset [13]. CM, on the other hand, serves as a baseline to judge the performance improvement by CEMD and EMD. As shown in Fig. 7(b), the performance of CEMD is highly competitive to EMD. CEMD offers better retrieval rate for top-k ($k \leqslant 10$) list, despite the fact that CEMD is about 40 times faster than EMD. Compared with OOS, CEMD offers lower precision but faster speed (about 100 times, cf. Table 4).

7.4. Effect of word weighting

We experiment with the effects of three weighting schemes: soft-weighting, binary and TF on CEMD. For soft-weighting, the $h(i,j)$ is based on Eq. 5 since the CEMD has already incorporated the linguistic information in the ground distance measure. Fig. 7(c) shows the performances of CEMD with JCN based on three weighting schemes on Columbia dataset. The results show that soft (CEMD-Soft) outperforms other schemes. This confirms the fact that soft-weighting performs well not only for semantic video indexing, but also for NDK retrieval under the CEMD matching.

7.5. Performance on TRECVID-2006 test set

To further verify the performance, we conduct experiments on a larger dataset: the TRECVID-2006 test set containing nearly 80k keyframes. In addition to retrieval effectiveness, we also consider speed which is also a concern when searching duplicate copies in large-scale database. We compare five measures: CEMD-Soft, CEMD-TF, COS-Soft, COS-TF, and CEMD-TF + COS-Soft. Cosine similarity (COS) serves as a baseline for its popularity and efficiency in IR [39]. It is important to note that the soft-weighting, which softly assigns a keypoint to multiple words, sacrifices the advantage of sparse representation as in TF weighting (average number of non-

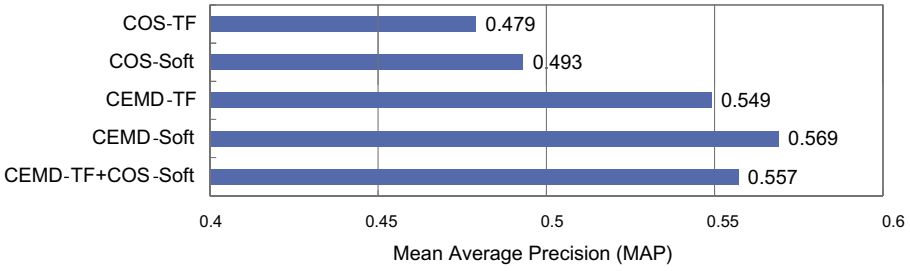

Fig. 8. NDK retrieval performance on the TRECVID-2006 test set.

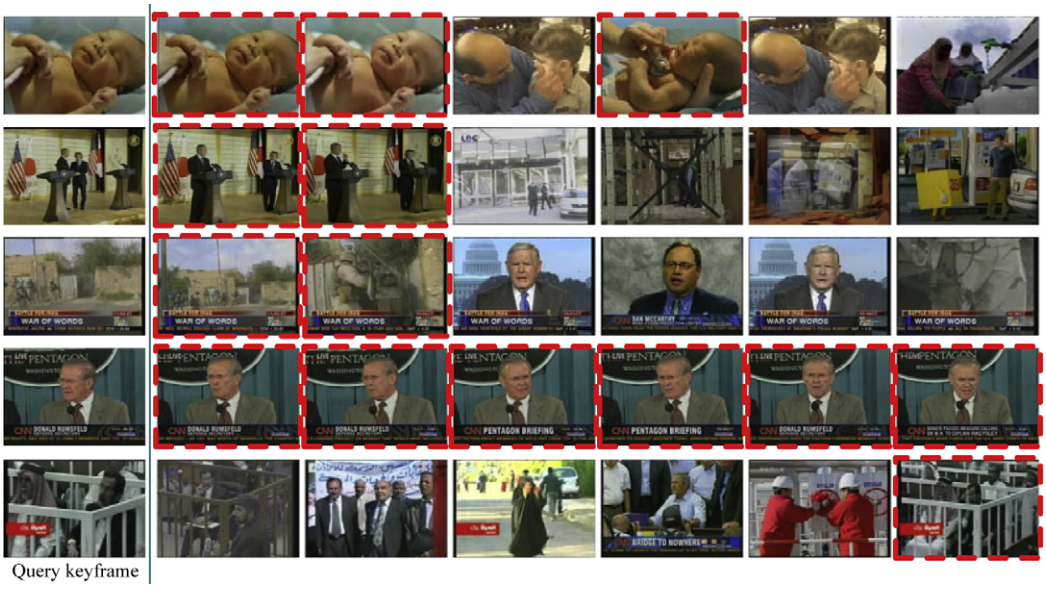

Fig. 9. Examples of near-duplicate keyframes retrieved by CEMD-Soft. The left most five examples are query keyframes, followed by the most similar retrieved keyframes. The true positives are marked in red boxes. (For interpretation of color mentioned in this figure the reader is referred to the web version of the article.)

Please cite this article in press as: Y.-G. Jiang, C.-W. Ngo, Visual word proximity and linguistics for semantic video indexing ..., Comput. Vis. Image Understand. (2008), doi:10.1016/j.cviu.2008.10.002

zero bins in BoW histogram increases from 153 to 340). As a result, the increase of non-zero bins in soft-weighting seriously affects the speed of EMD/CEMD. This situation is in contrast to Section 6, where soft-weighting does not affect the speed much because the running time complexity of χ^2 kernel is linear. The complexity of EMD/CEMD is cubic and thus can become much slower when soft-weighting scheme is adopted. For practical consideration, we also experiment a fusion strategy, CEMD-TF + COS-Soft, which takes the advantage of cross-bin word matching in sparse representation while still utilizing the power of soft-weighting under cosine similarity.

We experiment with 110 near-duplicate queries randomly found in the test set. Each approach returns the top-40 ranked keyframes for performance comparison. To evaluate the results, two assessors were invited to label the keyframes returned from the five tested approaches. The ground-truth is produced by pooling the near-duplicate keyframes labeled by assessors. We use AP over the top-40 lists as the evaluation criteria. Fig. 8 shows the mean average precision (MAP) of the five approaches over the 110 queries. CEMD outperforms cosine similarity by 14.6% (TF) and 15.5% (Soft). This again confirms the effectiveness of the proposed linguistics-based CEMD matching for near-duplicate retrieval. Meanwhile, the average fusion of CEMD-TF and COS-Soft offers comparable performance to CEMD-Soft, while with the advantage of speed efficiency. Fig. 9 shows the examples of near-duplicate keyframes retrieved by CEMD-Soft. These examples indicate the types of near-duplicate visual information being captured. Our approach could successfully retrieve the NDKs with variations such as color, lighting, scale, etc.

Table 4 lists average response time of querying a NDK in TRECVID-2006 test set. The response time includes the time to upload the features and save the results. The experiments are conducted on a Pentium-4 3 GHz machine. Overall, the proposed CEMD is significantly faster (40–60 times) than EMD, and TF is more efficient (7–10 times) than soft-weighting under EMD/CEMD. By fusing CEMD-TF and soft weighting, the speed is about the same as CEMD-TF but with better retrieval performance.

8. Conclusion and future work

We have presented our approaches in exploring the linguistics and proximity of visual words. On one hand, a soft-weighting scheme is proposed to softly and linguistically weight the significance of words by assigning one keypoint to multiple words. On the other hand, CEMD, which exploits the ontological relationship of visual words, is proposed for the cross-bin matching of visual words. The experimental results on video semantic indexing and near-duplicate retrieval show the advantages of incorporating linguistic and ontological relationships in word weighting and similarity measures of BoW.

Given the success of modeling visual linguistics as shown in our experiments, we plan to develop a more efficient method for cross-bin word matching. One possible direction is to apply the recently proposed diffusion distance [40]. In addition, while ontology is shown to be useful, in this paper we only explore the hyponym relationship of visual words. Other aspects such as synonymy and polysemy of visual words could be further studied to extend our current work.

Acknowledgments

The work described in this paper was fully supported by two grants from City University of Hong Kong (Project No. 7002241 and Project No. 7002112). The authors also thank Eric Zavesky for his help on polishing the language of this paper.

References

[1] E. Nowak, F. Jurie, B. Triggs, Sampling strategies for bag-of-features image classification, European Conference on Computer Vision (2006).
[2] J. Zhang, M. Marszałek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, International Journal of Computer Vision 73 (2) (2007) 213–238.
[3] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, IEEE Conference on Computer Vision and Pattern Recognition (2006).
[4] C.G.M. Snoek, J.C. van Gemert, Th. Gevers, B. Huurnink, D.C. Koelma, M. Van Liempt, O. De Rooij, K.E.A. van de Sande, F.J. Seinstra, A.W.M. Smeulders, A.H.C. Thean, C.J. Veenman, M. Worring, The mediamill TRECVID 2006 semantic video search engine, TRECVID Online Proceedings (2006).
[5] C. Fellbaum, WordNet: An Electronic Lexical Database, MIT Press, Cambridge, MA, 1998.
[6] B.S. Manjunath, W. Ma, Texture features for browsing and retrieval of image data, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (8) (1996) 837–842.
[7] K. Barnard, P. Duygulu, D. Forsyth, N. de Freitas, D. Blei, M.I. Jordan, Matching words and pictures, Journal of Machine Learning Research 3 (2003) 1107–1135.
[8] A.G. Hauptmann, M.-Y. Chen, M. Christel, W.-H. Lin, R. Yan, J. Yang, Multilingual broadcast news retrieval, in: TRECVID Online Proceedings, 2006.
[9] J. Cao, Y. Lan, J. Li, Q. Li, X. Li, F. Lin, X. Liu, L. Luo, W. Peng, D. Wang, H. Wang, Z. Wang, Z. Xiang, J. Yuan, W. Zheng, B. Zhang, J. Zhang, L. Zhang, X. Zhang, Intelligent multimedia group of Tsinghua University at TRECVID 2006 TRECVID, in: Online Proceedings, 2006.
[10] D. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.
[11] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, L. Van Gool, A comparison of affine region detectors, International Journal of Computer Vision 65 (1/2) (2005) 43–72.
[12] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (10) (2005).
[13] C.-W. Ngo, W.-L. Zhao, Y.-G. Jiang, Fast tracking of near-duplicate keyframes in broadcast domain with transitivity propagation, in: ACM International Conference on Multimedia, 2006.
[14] Y. Ke, R. Suthankar, L. Huston, Efficient near-duplicate detection and sub-image retrieval, in: ACM International Conference on Multimedia, 2004, pp. 869–876.
[15] J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Object retrieval with large vocabularies and fast spatial matching, IEEE Conference on Computer Vision and Pattern Recognition (2007).
[16] X. Wu, W.-L. Zhao, C.-W. Ngo, Near-duplicate keyframe retrieval with visual keywords and semantic context, in: ACM International Conference on Image and Video Retrieval, 2007.
[17] J. Sivic, A. Zisserman, Video google: a text retrieval approach to object matching in videos, in: International Conference on Computer Vision, 2003.
[18] Y.-G. Jiang, C.-W. Ngo, J. Yang, Towards optimal bag-of-features for object categorization and semantic video retrieval, in: ACM International Conference on Image and Video Retrieval, 2007.
[19] Y.-G. Jiang C.-W. Ngo, Bag-of-visual-words expansion using visual relatedness for video indexing, in: ACM SIGIR Conference on Research & Development on Information Retrieval, 2008.
[20] F. Jurie, B. Triggs, Creating efficient codebooks for visual recognition, in: International Conference on Computer Vision, 2005.
[21] F. Moosmann, B. Triggs, F. Jurie, Randomized clustering forests for building fast and discriminative visual vocabularies, in: Conference on Neural Information Processing Systems (NIPS), 2006.
[22] T. Pedersen, S. Patwardhan, J. Michelizzi, WordNet::Similarity – measuring the relatedness of concepts, in: National Conference on Artificial Intelligence (AAAI), 2004.
[23] P. Resnik, Using information content to evaluate semantic similarity in a taxonomy, in: International Joint Conferences on Artificial Intelligence, 1995.
[24] J.J. Jiang, D.W. Conrath, Semantic similarity based on corpus statistics and lexical taxonomy, in: Proc. of ROCLING X, 1997.
[25] Z. Wu, M. Palmer, Verb semantic and lexical selection in Annual Meeting of the ACL, 1994, pp. 133–138.
[26] A. Agarwal, B. Triggs, Hyperfeatures – multilevel local coding for visual recognition, in: European Conference on Computer Vision, 2006.
[27] D. Nister, H. Stewenius, Scalable recognition with a vocabulary tree, in: IEEE Conference on Computer Vision and Pattern Recognition, 2006.
[28] K. Grauman, T. Darrell, Approximate correspondences in high dimensions, in: Advances in Neural Information Processing Systems (NIPS), 2007.
[29] Y. Rubner, C. Tomasi, L.J. Guibas, The earth mover's distance as a metric for image retrieval, International Journal of Computer Vision 40 (2) (2000) 99–121.
[30] Y.-G. Jiang, C.-W. Ngo, Ontology-based visual word matching for near-duplicate retrieval, in: IEEE International Conference on Multimedia & Expo, 2008.
[31] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 888–905.
[32] TREC Video Retrieval Evaluation (TRECVID), Available from: <http://www-nlpir.nist.gov/projects/trecvid/>.

[33] LSCOM lexicon definitions and annotations, in: DTO Challenge Workshop on Large Scale Concept Ontology for Multimedia, Columbia University ADVENT Technical Report #217-2006-3, 2006.
[34] V. Vapnik, The Nature of Statistical Learning Theory, Springer, New York, 1995.
[35] J.A. Aslam, V. Pavlu, E. Yilmaz, Statistical method for system evaluation using incomplete judgments, in: ACM SIGIR Conference, 2006.
[36] S. Petrov, A. Faria, P. Michaillat, A. Berg, D. Klein, J. Malik, A. Stolcke, Detecting categories in news video using acoustic, speech, and image features, in: TRECVID Online Proceedings, 2006.
[37] M. Campbell, S. Ebadollahi, D. Joshi, M. Naphade, A. Natsev, J. Seidl, J.R. Smith, K. Scheinberg, J. Tesic, L. Xie, IBM research TRECVID-2006 video retrieval system, in: TRECVID Online Proceedings, 2006.
[38] D.-Q. Zhang, S.-F. Chang, Detecting image near-duplicate by stochastic attributed relational graph matching with learning, in: ACM International Conference on Multimedia, 2004.
[39] G. Salton, A. Wong, C.S. Yang, A vector space model for automatic indexing, Communications of the ACM 18 (1975) 613–620.
[40] H. Ling, K. Okada, Diffusion distance for histogram comparison, in: IEEE Conference on Computer Vision and Pattern Recognition, 2006.

Please cite this article in press as: Y.-G. Jiang, C.-W. Ngo, Visual word proximity and linguistics for semantic video indexing ..., Comput. Vis. Image Understand. (2008), doi:10.1016/j.cviu.2008.10.002