正準相関部分空間における再帰的ランダム・ウォークを用いた画像の自動タグ付け

Automatic Image Annotation as a Random Walk with Priors on the Canonical Correlation Subspace

ティモシー バヨール[*]　チャイジィ シュ[*]　インフイ ジョ[**]
Timothée Bailloeul　Cai-Zhi Zhu　Yinghui Xu

要　旨

本稿では，Panら[14)] によるGCapと呼ばれる手法をベースとする，グラフ処理に基づく画像の自動タグ付け手法を提案する．我々の手法（EGCap：Enhanced GCap）は，正準相関解析（CCA）を用いて画像空間中の意味的な乖離を縮小すると共に，テキスト空間中においてタグ同士を関連付ける新しい距離関数を定義する．この結果，画像レベルでのリンク誤りが減少し，システムから出力されるタグの一貫性が改善された．更に本手法では，タグ付けの精度向上が確認された．文書の逆出現頻度に基づくリンクの重み付け手法とCCA距離関数を導入する．本手法は単純で且つ安定しており，リアルタイムでのタグ付けが可能である．この要因として，画像の局所的二値パターン（Local Binary Pattern）特徴を利用していること，領域分割が不要であること，またCCAの射影により特徴ベクトルを削減していることが挙げられる．提案手法と他の最先端の手法とをCorelおよびFlickrのデータベースを用いて比較し，本手法の優位性を単語単位，画像単位，および処理時間の各評価指標を用いて示す．

ABSTRACT

In this paper, we present a graph-based scheme founded on the GCap method of Pan *et al.*[14)] to perform automatic image annotation. Our approach, namely enhanced GCap (EGCap), takes advantage of the canonical correlation analysis technique (CCA) to shorten the semantic gap in the image space and define a new metric in the text space to correlate annotations. As a result, linkage errors at the image level are decreased and the consistency of tags output by the system is improved. Besides, we introduce graph link weighting techniques based on inverse document frequency and CCA metric which are proved to enhance the annotation quality. Simple and self-consistent, the present approach achieves image annotation in real time due to the lightweight Local Binary Pattern image features used, the absence of image segmentation, and the reduced size of feature vectors after CCA projection. We test the proposed approach against top-grade state-of-the-art techniques on Corel and Flickr databases, and show the effectiveness of our method in terms of per-word, per-image and processing time performance indicators.

[*] リコーソフトウェア研究所（北京）有限公司
Ricoh Software Research Center (Beijing) Co., Ltd.

[**] 研究開発本部 ソフトウェア研究所
Software R&D Center, Research and Development Group

1. Introduction

The rapid expansion of the Web and the growing number of digital imaging sensors embedded in consumer-level products are producing larger and larger image repositories. As the mass of accumulated data is meant to be useful, successful image indexing and retrieval systems have become a necessity. Querying databases by an exemplar image has been one of the first techniques to search digital photographs. In such scenario, images of the database which are visually similar to the query are returned. This procedure is inconvenient for two reasons: i) it constrains the user to always have an image at hand. ii) the contents of retrieved images are different from these of the query because of the semantic gap occurring in the image feature space. A more promising alternative consists in indexing images automatically with high-level information keywords prior to the retrieval task. This way, the user can retrieve more semantically consistent images upon text query, which is more effective and suitable.

There are basically two ways to carry out such computerized image annotation. The first, called *text-based* approach, consists in mining keywords from any kind of textual information attached to the image to be indexed. The text metadata used comprehends the image file name, caption, and text wrapping the picture when the latter is embedded in a Web page. While such technique has been widely adopted by major Web search engines, it has two limitations: i) it is prone to indexing errors as the text metadata is not necessarily related to the contents depicted on the image. ii) it is constrained to the processing of photos incorporated into electronic documents, such as Web pages, and cannot process standalone images. The shortcomings of the text-based procedure are addressed by the alternative *image-based* approach. The latter is based on Computer Vision and statistical Machine Learning techniques to discover the intricate relationships among words and image features given a training set of human-annotated images. The trained statistical model is then able to predict a set of relevant keywords related to an image unseen in the learning phase. For nearly a decade, computer scientists have achieved progresses in this field since the pioneering work of Mori *et al.*[11]. However, automatic image annotation remains a challenging problem, and because of the moderate annotation accuracy and recall performance obtained, no fully operational system or product has emerged till now. For the past years, various approaches have been proposed, including methods based on machine bilingual translation[3], generative probabilistic models[1), 4)], graph theory[8), 14)] and multi-class classification[2]. Recently, more efforts were dedicated to achieve automatic image annotation in real time. In 12), Nakayama *et al.* disclosed one of the fastest annotation schemes of the moment. Implemented on an 8-core machine, their method achieves image annotation in 0.02 s and yields the best performance on the Corel5k data set designed in 3). The main contribution of their approach is the pre-processing of image features by a subspace method which maximizes the correlation between image and text information contained in the training set. The reduced size of image features after subspace projection fastens the image annotation procedure which is similar to that of Feng *et al.*[4]. In the following, we highlight the main contributions of the approach presented in this paper:

Firstly, we explore the benefit of the Local Binary Pattern (LBP) image features for the task of automatic image annotation. These features are attractive as they compactly embed various sources of information such as contrast and color and have a low computational cost.

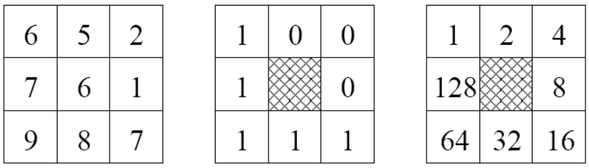
Secondly, we introduce the canonical correlation analysis (CCA) subspace technique to improve the graph-based scheme of Pan *et al.*[14]. The advantages of CCA are manifold. First, it reduces the semantic gap in the image feature space, which decreases errors in the "image-to-image" linkage. Second, in case we deal with weakly annotated[1] training data, CCA is most effective when considering image features at the whole image level. We consequently do not need to segment images, which saves processing time and simplifies the structure of the graph linking images and their annotations. The computational workload is also cut by the smaller size of feature vectors after projection in CCA space. Third, we use the CCA-projected text space to measure the similarity among annotations based on visual impression. It allows the creation of new "word-to-word" links in the graph to increase the consistency of words output by the annotation system.

Thirdly, we investigate graph link weighting techniques based on inverse document frequency and the distance in CCA-projected subspaces to improve the quality of annotations.

The remainder of this paper is organized as follows. In sections 2 and 3 we recall the background of LBP image features and the CCA technique. In section 4, we outline the shortcomings of the GCap method[14)] and propose counter-measures based on the random walk on the canonical correlation space. In section 5, we provide experimental results to assess the efficiency of the novelties exposed in this paper, and compare the proposed scheme to top-grade state-of-the-art techniques. We show the effectiveness of the proposed method in terms of per-word, per-image and processing time indicators, and conclude in section 6.

2. LBP image features

The LBP feature is a compact texture descriptor for which each comparison result between a center pixel and one of its surrounding neighbors will be encoded as a bit in a LBP code (see Fig.1).



6	5	2
7	6	1
9	8	7

1	0	0
1	▨	0
1	1	1

1	2	4
128	▨	8
64	32	16

Fig.1 LBP code computation for gray level image and an 8-neighbor window[9)]. The 8-bit pattern is 11110001, and the corresponding LBP code: 1+16+32+64+128=241.

LBP codes are computed for every pixel and accumulated into a histogram to represent the whole image. Statistical analysis can also be used to generate even shorter or compacter LBP histograms, such as rotation invariance (RI), uniformity (U2), or the combination of both (RIU2) (see 9) for more details). LBP is flexible as it can be enhanced by contrast, color or multi-scale information. In the proposed approach, we adopted the uniformed LBP (59 bins) to which contrast information - quantized into 5 levels - was added. Finally, we took advantage of color information by extracting the aforementioned LBP feature from three intra-channels (R,G,B) and three inter-channels (RG, RB, GB). The final LBP feature size is then $p = 59 \times 5 \times 6 = 1770$, which is high, but as we will see in the following, CCA can reduce it to a more reasonable size. According to our experience, the above LBP settings are a good balance between annotation performance and time complexity. As LBP extraction mainly requires addition operations, it can be achieved at a very high speed.

3. Canonical correlation analysis

Founded by Hotelling in 1936[6)], the canonical correlation analysis (CCA) is a technique which maximizes the correlation between two linearly dependent signals while minimizing their sizes. CCA and kernel-CCA were first used for the purpose of image

[1] In "weak annotation", keywords are attached to an image without localization information. It is opposite to region labeling.

annotation and retrieval by Hardoon *et al.* in 5).

3-1 Inputs of CCA

Assume we are given a training set of N pairs of observations $(\mathbf{x}_i, \mathbf{y}_i)$, where $\mathbf{x}_i \in \Re^p$ and $\mathbf{y}_i \in \Re^q$ refer to the image and annotation feature vectors of the ith training sample respectively. Later on, we assume that all feature vectors corresponding to the training images are stored in the columns of the sample matrix $\mathbf{X} \in \Re^{p\times N} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$.

The vector $\mathbf{y}_i \in \Re^q$ refers to the caption of the ith image of the training set Im_i. It is the output of a function mapping the caption of Im_i to an indicator variable $G_{w_k} \in \{0,1\}$ where w_k is a word of the annotation vocabulary V and with $G_{w_k} = 1$ if w_k is an annotation of Im_i, otherwise it equals 0. In such case, $\mathbf{y}_i = (G_{w1}, \ldots, G_{w_q})^t$ where q denotes the size of V. All annotation vectors are stored in the columns of the sample matrix $\mathbf{Y} \in \Re^{q\times N} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)$. Prior to the computation of the CCA projection model, one has to center the matrices $\mathbf{X}$ and $\mathbf{Y}$ and normalize them by the standard deviation across the samples as follows:

$$\tilde{\mathbf{Z}} = \left((\mathbf{z}_1 - \hat{\mathbf{m}}_z)(\hat{\boldsymbol{\sigma}}_z \mathbf{Id}_s)^{-1}, \ldots, (\mathbf{z}_N - \hat{\mathbf{m}}_z)(\hat{\boldsymbol{\sigma}}_z \mathbf{Id}_s)^{-1}\right) \quad (1)$$

where $\tilde{\mathbf{Z}} = \tilde{\mathbf{X}}$ or $\tilde{\mathbf{Z}} = \tilde{\mathbf{Y}}$, $\hat{\mathbf{m}}_z = \frac{1}{N}\sum_{i=1}^{N} \mathbf{z}_i \in \Re^s$ and $\hat{\boldsymbol{\sigma}}_z^2 = \frac{1}{N}\sum_{i=1}^{N} (\mathbf{z}_i - \hat{\mathbf{m}}_z)^2 \in \Re^s$ with $(z,s) = (x,p)$ or $(z,s) = (y,q)$, and $\mathbf{Id}_s$ is the $s \times s$ identity matrix.

3-2 CCA model calculation

CCA is to find pairs of canonical directions $\mathbf{W}_x$ and $\mathbf{W}_y$ that maximize the correlation ρ between the projections $\mathbf{W}_x^t \tilde{\mathbf{X}}$ and $\mathbf{W}_y^t \tilde{\mathbf{Y}}$. The definition of the correlation is recalled in equation (2):

$$\rho = \frac{\mathbf{W}_x^t \hat{\mathbf{C}}_{xy} \mathbf{W}_y}{\sqrt{\mathbf{W}_x^t \hat{\mathbf{C}}_{xx} \mathbf{W}_x \mathbf{W}_y^t \hat{\mathbf{C}}_{yy} \mathbf{W}_y}} \quad (2)$$

where $\hat{\mathbf{C}}_{xx}$, $\hat{\mathbf{C}}_{yy}$ and $\hat{\mathbf{C}}_{xy}$ are the unbiased estimations of the within-set and between-set covariance matrices. The maximization of ρ is realized by maximizing the covariance $\mathbf{W}_x^t \hat{\mathbf{C}}_{xy} \mathbf{W}_y$ given the constraints $\mathbf{W}_x^t \hat{\mathbf{C}}_{xx} \mathbf{W}_x = \mathbf{W}_y^t \hat{\mathbf{C}}_{yy} \mathbf{W}_y = 1$. It leads to the resolution of two symmetric eigenproblems which can be resolved successively, or by resorting to the Singular Value Decomposition (SVD) technique10). In the SVD-based approach we consider the following matrix:

$$\mathbf{M} = \hat{\mathbf{C}}_{xx}^{-1/2} \hat{\mathbf{C}}_{xy} \hat{\mathbf{C}}_{yy}^{-1/2} \in \Re^{p\times q} \quad (3)$$

$\mathbf{M}$ can be decomposed by SVD as follows:

$$\mathbf{M} = \mathbf{U}\mathbf{D}_m \mathbf{V}^t \quad (4)$$

where $\mathbf{U} \in \Re^{p\times p}$ and $\mathbf{V} \in \Re^{q\times q}$ are two orthonormal matrices and $\mathbf{D}_m$ is a pseudo-diagonal matrix so that $\mathbf{D}_m = diag(\rho_1^2, \ldots, \rho_m^2) \in \Re^{p\times q}$, with $m = \min(p,q)$. The diagonal elements of $\mathbf{D}_m$ - which are referred to as singular values sorted in decreasing order - correspond to the squared correlation values of the canonical directions. The CCA projection matrices are finally $\mathbf{W}_x = \hat{\mathbf{C}}_{xx}^{-1/2}\mathbf{U}$ and $\mathbf{W}_y = \hat{\mathbf{C}}_{yy}^{-1/2}\mathbf{V}$. The canonical directions - or eigenvectors - of both image and text spaces are stored as the columns of $\mathbf{W}_x$ and $\mathbf{W}_y$ respectively. As any subspace method, it is possible to only retain the canonical directions pertaining to the most significant m_0 correlation values and consequently decrease the size of the projected features.

3-3 Outputs of CCA

Given two CCA projection matrices $\mathbf{W}_x \in \Re^{p\times m_{0x}}$ and $\mathbf{W}_y \in \Re^{q\times m_{0y}}$, and some arbitrary image and annotation features $\mathbf{x}$ and $\mathbf{y}$, the projection in the image CCA subspace is achieved as follows: $\mathbf{x}_{proj} = \mathbf{W}_x^t \tilde{\mathbf{x}} \in \Re^{m_{0x}}$, while in the CCA text space, it takes the form: $\mathbf{y}_{proj} = \mathbf{W}_y^t \tilde{\mathbf{y}} \in \Re^{m_{0y}}$. Alternatively, the projection matrices $\mathbf{W}_x$ and $\mathbf{W}_y$ can be weighted by the correlation values contained in $\mathbf{D}$ so as to enhanced the main canonical directions.

4. Random walk with priors on canonical correlation subspace

4-1 Background and limitations of GCap method

The training model in GCap[14)] is represented by a 3-layer undirected graph: i) the word layer whose nodes correspond to the keywords of the annotation vocabulary. ii) the image layer whose nodes relate to the training images. iii) the image region layer whose nodes embody the sub-areas of the training images. Such regions are typically output by an image segmentation procedure, and ideally correspond to the image objects. In the training phase, the image nodes are connected to their respective annotations by an "image-to-word" link. Each image is connected to its regions by "image-to-region" links that establish parent-to-child relationships between the image and its sub-areas. Finally, image regions are connected to each other across the whole data set based on their visual similarity: each region is connected to its k_{im} nearest neighboring regions in the sense of a metric defined on the image feature space ("region-to-region" links). In the image annotation phase, an incoming test image is first appended to the training graph. This is realized by connecting each test image region to its k_{im} nearest neighbors of the trained graph. Then, a random walk with restart (RWR) – beginning from the test image node – is performed to determine the most relevant words to annotate the test image. The RWR can be regarded as a Markov chain transition process biased towards the starting node, and is modeled by the following iterative equation:

$$\mathbf{U}_{n+1} = (1-c)\mathbf{A}\mathbf{U}_n + c\mathbf{V} \qquad (5)$$

where $\mathbf{U}$ is the N_{node} -dimensional vector which represents the Markov transition state of all nodes of the graph (N_{node} is the total number of nodes in the graph). $\mathbf{A}$ is the adjacency - or Markov transition - matrix which is populated with ones when a link is realized in the undirected graph. c is the restart probability and the N_{node}-dimensional restart vector $\mathbf{V}$ is null, except at the entry of the test image where it is set to 1. When the values of vector $\mathbf{U}$ have converged according to equation (5), the entries related to the word nodes are sorted by decreasing order, and the annotations corresponding to the top-n values are chosen to annotate the test image. In the following, we list the limitations of the scheme of Pan *et al.*[14)].

4-1-1 Region-to-region linkage errors

The k_{im} nearest image region nodes in the graph are connected based on visual similarity. However, whatever the region similarity, all "region-to-region" links have the same constant weight. This is problematic as the k_{im}^{th} neighbor, which could be somehow very different from the region query, would have the same influence as other nearer neighbors. If the "farthest" neighbor is very dissimilar to the query, an incorrect link would be made, which decreases the annotation quality in the end. As a result, only small values of k_{im} can be used, which precludes from investigating long-range interactions at the region level. This fact is verified in 14) where the annotation accuracy starts to decline for values of k_{im} superior to 3.

The nearest neighbor search to connect regions is not scalable if a brute force approach is employed. As a consequence, it is desirable to resort to optimized nearest neighbor (ONN) search engines to fasten the process. In 13), the Approximate Nearest Neighbor (ANN) engine based on kd-tree data structure is used. In a kd-tree, L_n norm must be used to partition the k-dimensional feature space into hyper-rectangles to optimize the search. However, such L_n metric is incompatible with the multi-form image features often used in automatic image annotation. For instance, in 3), 13), and 14), color histograms, texture histograms and shape descriptors are merged together, which calls for more sophisticated pseudo-distances to compare images (entropy, chi-square, etc.). Since such pseudo-distances define manifolds that are inconsistent with kd-tree optimization, L_n norm must be used anyway to the detriment of the accuracy in the "region-to-region" linkage.

4-1-2 Bias in "image-to-word" linkage

Some word nodes might be connected to image nodes more often than others. Such terms relate to the most popular tags used in the training phase, and there is a natural bias towards them in the RWR process due to their prominence in the Markov transition matrix. This is an unwelcome effect as popular words would always be output by the annotation system, regardless the image to tag. Balancing the link weight values between frequently and seldom used tags is a possible counter-measure to this problem.

4-2 Proposed counter-measures

We address the aforementioned issues by resorting to the CCA technique to improve and simplify the construction of the graph which associates image and word nodes. The merit of introducing CCA in the graph lies in the two following aspects.

First, the distance in the CCA-projected image space will somehow tell us how visually and semantically similar image nodes are. Such characteristic - inherited from the high-level annotations of the training set - helps bridge the semantic gap and decreases linkage errors in the graph at the image level. Moreover, CCA enables the use of a Euclidean metric in the CCA-projected image space, which allows to fully take advantage of the speed capability of ONN search engines without lowering the accuracy of the linkage. Finally, as we deal with weakly tagged training images in this study, CCA best befits image and annotation information at a global level. Consequently, image features can be computed at the scale of the whole image and no segmentation is needed. This allows a significant cut in the computational workload and the simplification of the graph in which only image and word nodes are needed (the region nodes can be removed, see Fig. 3). As CCA is a subspace method, image features are also projected in a lower dimensional space. The population to be investigated in "image-to-image" linkage by the ONN search engine is then smaller and carried out in a low-dimensional space. It dramatically enhances the processing speed for training the graph and annotating new images.

Second, the distance in the CCA-projected text space will somehow tell us how similar words are based on their visual impression. Akin to the "image-to-image" linkage procedure, we connect each word node of the graph to its k_{txt} nearest neighbor word nodes based on a metric defined on the CCA-projected text space. This way, we intend to reinforce the consistency among the words of the vocabulary to output annotations which are semantically more similar in the end. Such process is different from that of 8) where dictionary-based prior knowledge is used to increase the consistency of the tags. In synonymy hierarchy like in WordNet, only closely related terms can be correlated. As a result, the words "water" and "boat", while being conceptually correlated, cannot be associated to each other using lexicon-based tools like WordNet. Instead, measuring the word similarity in the CCA-projected text space allows correlating concepts which are often depicted simultaneously on the training images. In Table 1, we provide some examples of word correlation in the CCA space based on the Corel5k data set described in section 5.

Table 1 Examples of "word-to-word" relationships in the CCA-projected text space built from Corel5k data set.

Query word	Top-3 nearest neighbors
boats	sailboats, lighthouse, ships
leaf	plants, branch, stems
horizon	trail, peaks, lake
plane	prop, helicopter, eagle

Fig.2 illustrates the benefits of CCA in both image and text spaces in order to narrow the semantic gap and construct a text metric to correlate annotations.

In order to investigate longer interaction ranges at the image and word levels, i.e. higher values of k_{im} and k_{txt}, we set the weight of "image-to-image" and "word-to-word" links as a decreasing function of the distance measured in the CCA-projected space:

$$w_{im\text{-}im}(i,j) = \exp\left(-\frac{d_{ij}^{im}}{\alpha}\right) \qquad (6)$$

$$w_{word-word}(i,j) = \exp\left(-\frac{d_{ij}^{txt}}{\beta}\right) \qquad (7)$$

where d_{ij}^{im} is the Euclidean distance between image i and j measured in the CCA-projected image space of dimension m_{0x}; d_{ij}^{txt} is the Euclidean distance between word i and j measured in the CCA-projected text space of dimension m_{0y}; α and β are parameters controlling the decrease rate of the functions.

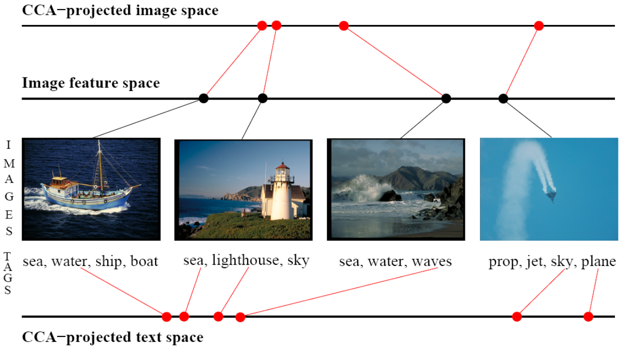


Fig.2 Illustration of projections in CCA image and text spaces.

To counter the bias of popular words learned in the training, we determine the weight between images and words by resorting to the Inverse Document Frequency (IDF) method. The latter sets a weight value to "image-to-word" links which is inversely proportional to number of times the considered word node is connected to image nodes. Consequently, in the graph building phase of the training, we set the weight between image i and word j as follows:

$$w_{im\text{-}word}(\,\mathrm{Im}_i\,,w_j\,) = (1-\gamma)+\gamma\left(\frac{Freq_{\max}-Freq_j}{Freq_{\max}-Freq_{\min}}\right) \qquad (8)$$

where γ is a parameter tuning the weight balance between popular and rare words, $Freq_{\max}$ is the occurrence frequency of the most popular word in the training set, $Freq_{\min}$ is the occurrence of the most seldom used word in the training set, and $Freq_j$ is the occurrence frequency of the j^{th} word of the annotation vocabulary. Note that in 8), Liu *et al.* also used IDF to improve the quality of automatic image annotation. Unlike our proposal, the IDF technique in 8) does not intervene directly at the graph level to alleviate the bias towards high frequency tags in the RWR process. It is rather used with exogenous linguistic knowledge derived from WordNet[15)] to filter out irrelevant tags from the outputs of a graph-based RWR procedure. As a result, it is a counter-measure to improve the consistency of the final output words, rather than a procedure to alleviate the problem of popular words learned in the training. The schema of Fig.3 summarizes the contributions disclosed in this paper.

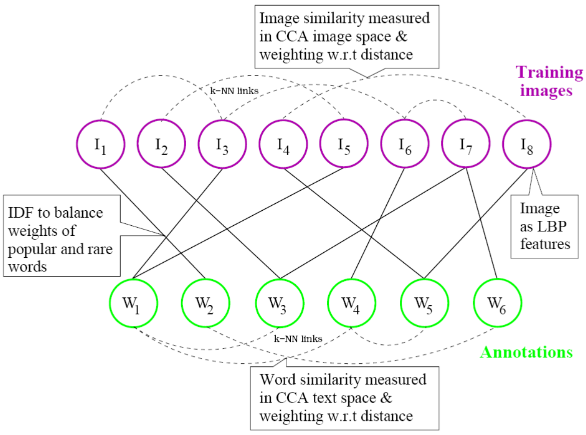


Image similarity measured in CCA image space & weighting w.r.t distance
k−NN links
Training images
I_1
I_2
I_3
I_4
I_5
I_6
I_7
I_8
IDF to balance weights of popular and rare words
Image as LBP features
W_1
W_2
W_3
W_4
W_5
W_6
k−NN links
Annotations
Word similarity measured in CCA text space & weighting w.r.t distance

Fig.3 Illustration of the enhanced graph to perform the RWR on canonical correlation space.

5. Experimental results

Two image data sets were used to assess the image auto-annotation quality of the present scheme against that of the state-of-the-art. First, we employed the Corel5k database[2] which is made of 5,000 color images annotated with 371 English keywords[3)] (the image size is 768×512 pixels). Typically, 4,500 images are used for training, while 500 images are used for testing. According to the Corel5k protocol, the schemes will be evaluated based on the first 5 tags they would assign to each test image. To evaluate per-word performance, the 260 keywords that are actually used in the ground truth of the testing set are considered. While Corel5k database has been the reference benchmark for computer scientists to compare their algorithms for the past years, it has the following limitations: i) images are taken by professional photographers, which is far from being the case in real world applications. ii) images are originally analog, then scanned. The digitization process always introduces noise. iii) the visual and semantic information contained in the images is not diverse. iv) the most outstanding problem is that the testing set is very similar to the training set. This makes the performance evaluation over-optimistic and biased.

Because of the aforementioned issues, more and more researchers carry out experiments on more challenging and diverse photos such as these available on the Web [7, 17]. As a result, we also built an image data set made of photos gathered from the Internet to evaluate the performance of our approach. Using the Flickr API, we downloaded 11,981 color photographs stored in Flickr's repository in 2007. 8,979 images are used for training while 3,002 photos are employed for testing and the annotation vocabulary is made of 298 English words. Per-word performance is evaluated based on the 292 words used in the testing ground truth. Unlike the Corel5k database, Flickr12k has the following features: i) images are taken by average camera users, i.e. non-expert photographers. ii) the visual and semantic diversity of the images is high. This is due to the random selection of images while acquiring them from Flickr. iii) there are errors and inconsistencies in the annotation ground truth as the users are not experts. iv) images are originally in the digital form. In the Flickr12k protocol, we decided to evaluate the proposed annotation scheme based on the first 15 tags assigned to each test image.

5-1 Performance indicators

In this article, traditional performance indicators are used. They are per-word precision&recall (eq. 9-10) and per-image precision&recall (eq. 11-12).

$$pw_{precision}(w) = 100 * \mathrm{Im}_C(w) / \mathrm{Im}_{AUTO}(w) \quad (9)$$

$$pw_{recall}(w) = 100 * \mathrm{Im}_C(w) / \mathrm{Im}_H(w) \quad (10)$$

where $\mathrm{Im}_C(w)$ is the number of images correctly annotated with word w by the system, $\mathrm{Im}_{AUTO}(w)$is the number of images annotated with word w by the system, and $\mathrm{Im}_H(w)$ is the number of human-annotated images with word w (ground truth).

$$pi_{precision}(\mathrm{Im}) = 100 * W_C(\mathrm{Im}) / W_{AUTO}(\mathrm{Im}) \quad (11)$$

$$pi_{recall}(\mathrm{Im}) = 100 * W_C(\mathrm{Im}) / W_H(\mathrm{Im}) \quad (12)$$

where $W_C(\mathrm{Im})$ is the number of words correctly assigned to image Im by the system, $W_{AUTO}(\mathrm{Im})$ is the number of words assigned to image Im by the system, and $W_H(\mathrm{Im})$ is the number of human-produced annotations for image Im (ground truth). In case a word w is never used by the image annotation scheme to annotate test images, its per-word precision is not computable as the denominator of equation (9) is zero.

[2] This publication includes images from the Corel Stock Photo Library images (Fig.2 and Fig.4) which are protected by the copyright laws of the U.S., Canada and elsewhere. Used under license.

By convention, we set the precision of the word to zero when such situation occurs. To derive the global per-word and per-image indicators, the individual scores are averaged over the whole annotation vocabulary and testing image set:

$$PW_{p/r} = \frac{1}{q}\sum_{i=1}^{q} pw_{precision/recall}(w_i) \qquad (13)$$

where q is the size of the annotation vocabulary.

$$PI_{p/r} = \frac{1}{N_{test}}\sum_{j=1}^{N_{test}} pi_{precision/recall}(\mathrm{Im}_j) \qquad (14)$$

where N_{test} is the number of images in the testing set. The following F1 measures are also used to tune parameters: $PW_{F1} = \frac{2PW_pPW_r}{PW_p + PW_r}$ and $PI_{F1} = \frac{2PI_pPI_r}{PI_p + PI_r}$.

We define n_{nzr} as the number of words with non-zero recall. Such performance indicator n_{nzr} indicates how many words were used for automatic annotation in the testing phase and represents the number of terms effectively learned by the system. Finally, we also measure the processing time t which is needed to annotate one image on the average over the testing set. The time t is reported in seconds on the basis of a one-core implementation[3]. It covers all operations needed to annotate an image, including the cost of feature extraction.

5-2 Cross-validation

In order to tune the parameters of the proposed scheme - namely c, k_{im}, k_{txt}, α, β, γ, m_{0x}, and m_{0y} - we carried out a 5-fold cross-validation procedure on the training set of Corel5k and Flickr12k databases. Given a training set, we selected 5 different validation sets made of 450 images each. Optimal parameters are the ones that maximize $PW_{F1} + PI_{F1}$ on the average computed over the 5 validation subsets.

5-3 Results

In Table 2, we compare the performance of the proposed scheme EGCap to that of the state-of-the-art on the Corel5k benchmark. We also provide results of EGCap with LBP features only, after CCA processing, and with all proposed novelties (CCA + eq. (6-7-8)). The baseline EGCap-LBP is similar to the GCap approach [14], except that there is no image segmentation and LBP features are used. We notice that CCA projection at the image level drastically improves performance indicators. Indeed, compared to EGCap-LBP, EGCap-LBP-CCA enhances by 40% to 50% per-word indicators, while it improves per-image scores by 26% (relative increase). EGCap-LBP-CCA outperforms state-of-the-art methods, except that of Nakayama *et al.* (Proba CCA, 12)) for which results are similar. The number of words learned by EGCap-LBP-CCA increased by 22 while it is lower than that of SML. Note that the processing speed of EGCap-CCA is higher than that of EGCap-LBP. This is due to the smaller size of image feature vectors after CCA projection $(m_{0x} = 60)$, which alleviates the computational burden in the NN search to append test images. The full configuration of EGCap (eq. (6-7-8)) allows improving performances compared to EGCap-LBP-CCA. The enhancement is however more subtle as it ranges from 4% to 12% regarding per-word and per-image scores. The most remarkable observation is the higher increase in per-word recall than in per-word precision. It would confirm that the proposed countermeasures not only enhance the accuracy by outputting more consistent tags, but also allow the use of more diverse words to annotate images (this fact is verified by the increase of n_{nzr}). Having a high per-word recall is especially interesting for the image search task as more relevant images would be returned upon keyword queries. Considering the PW_{F1} measure, the full configuration of EGCap outperforms the scheme of Nakayama *et al.*[12)] while having a similar processing time.

[3] EGCap was implemented in C++ on a 2.33 GHz CPU machine with 2 Go RAM (the ANN engine was used to fasten the "image-to-image" linkage).

The time complexity of the Proba CCA scheme is $O(N\times(m_{0x}+q))\approx 410N$ according to the parameters used in [12]. The complexity of EGCap is in $O(2\times N\times n_{iter}\times(\bar{n}+k_{im}+1))\approx 640N$, where n_{iter} is the number of iterations needed for the RWR to converge (20) and $\bar{n}$ is the average number of tags per training image (4 in Corel5k). We notice that the time complexity ratio between Proba CCA and EGCap corresponds to the difference in processing time reported in Table 2. Besides, EGCap can achieve faster annotation speed if k_{im} is reduced.

Table 2 Performance results of EGCap on Corel5k. Number of training images N = 4500, vocabulary size q = 371.

Method	Ref.	PW_p	PW_r	PI_p	PI_r	n_{nzr}	t (s)
EGCap	Eq. (6-7-8)	29	37	41	58	144	21
EGCap	LBP+ CCA	27	33	39	56	132	21
EGCap	LBP	18	23	31	44	110	27
Proba CCA	[12]	30	32	-	-	-	16
AGAnn	[8]	24	29	-	-	-	-
SML	[2]	23	29	-	-	137	1620
GCap	[14]	-	-	37	-	-	-

In Table 3, we compare the baseline of EGCap to its full configuration on the Flickr12k data set. As observed on Corel5k, the measures we proposed to enhance GCap are effective to improve the annotation quality. Note that the processing time is higher as there are more training image nodes in the graph. Besides, Flickr images have various sizes which are in general larger than that of Corel data.

Table 3 Performance results of EGCap on Flickr12k. Number of training images N = 8979, vocabulary size q = 298.

Method	Ref.	PW_p	PW_r	PI_p	PI_r	n_{nzr}	t (s)
EGCap	Eq. (6-7-8)	32	23	35	37	250	43
EGCap	LBP	27	11	29	29	182	47

5-4 Examples of image annotation

In Fig.4, we illustrate the benefits of EGCap on a few images extracted from Corel5k. In the second column, we notice that the joint use of IDF and "word-to-word" linkage is effective to lower the rank of the mistaken tag "water" which belongs to the most popular words learned in the training phase. Instead, the more relevant and correct tag "fox" is output. In column three, whatever the EGCap configuration used, only the tag "snow" is correct. However, note that irrelevant terms like "water", "sky", "grass" could be filtered out and more consistent labels belonging to the snowscape&animal topics were proposed by the most complete version of EGCap. In column four, we acknowledge that the tags of the full EGCap which do not match the ground truth are more relevant than these of EGCap-LBP and EGCap-LBP-CCA. In column five, there is a significant improvement between the baseline EGCap-LBP and the full version of EGCap where the top-3 suggested words are the same as these of the ground truth. Note that there is no significant difference with EGCap-CCA in this case, except that the latter output "tree" which seems less relevant than "frozen" regarding the information depicted on the image. In Fig.5, we illustrate examples of annotations on Flickr images. In column two, we observe that the full version of EGCap output tags related to flowers and colors, while these of the baseline scheme are all irrelevant. In column three, the irrelevant tag "water" could be moved to a lower position in the ranking order of the keywords while "street" could be successfully predicted. In column four, we show that paintings can also be handled by EGCap as the full version of the system could predict keywords consistent with the depicted contents, even if they do not match the ground truth (e.g. "graffiti, "portrait").

6. Conclusion

In this paper, we presented a simple and self-consistent graph-based method to achieve automatic image annotation. Based on the scheme of Pan *et al.*[14], EGCap takes advantage of the canonical correlation analysis (CCA) technique to shorten the semantic gap occurring in the image space, and define a metric in the text space to correlate words directly in the graph. The combination of the enhanced graph with some link weighting techniques dependent on IDF and CCA metric was shown to effectively improve the annotation quality by alleviating the bias towards popular words and enhancing the consistency of tags output by the system. The use of lightweight LBP features, the absence of segmentation and the reduced size of feature vectors after CCA projection allowed performing automatic image annotation in real time. Experimental results on Corel5k showed that EGCap reached top-level performance results with a processing speed similar to that of the fastest scheme of the moment. Future works will be dedicated to the evaluation of EGCap for the image retrieval task, the incorporation of less empirical link weighting functions, and the reduction of tuning parameters.

7. Acknowledgments

The authors would like to thank Tanakawho, Michael Cameron and Suzanne Limbert for providing the Flickr images displayed in Fig.5.

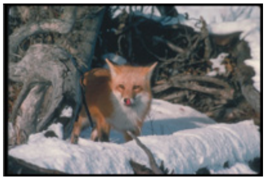
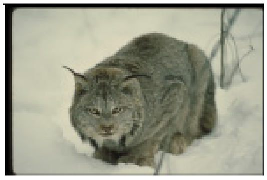
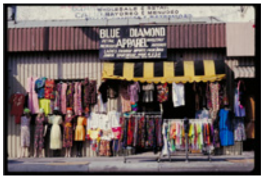
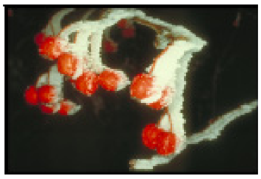
				
Ground truth	fox, snow, tree, wood	lynx, snow	buildings, clothes, shops, street	frost, fruit, ice
EGCap: LBP	festival, street, people, school, arctic	frost, tree, sky, grass, water	street, buildings, shops, people, tree	leaf, tracks, fruit, plants, cars
EGCap: LBP+CCA	water, rocks, grass, birds,tree	head, dog, snow, wolf, water	street, buildings, cars, mast, sign	fruit, frost, ice, snow, tree
EGCap: eq. (6-7-8)	birds, den, fox, water, tree	snow, head, bear, polar, tree	street, buildings, sign, cars, shops	frost, fruit, ice, frozen, snow

Fig.4 Examples of annotations output by EGCap on images extracted from the testing set of Corel5k database. The top-5 auto-tags are displayed.

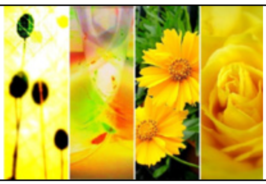
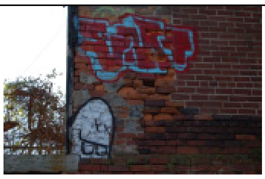
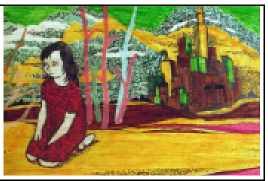
			
Ground truth	closeup, macro, color, collage, yellow	graffiti, street, art, streetart, Atlanta	drawing, painting, wood, city, art
EGCap: LBP	woman, water, white, summer, street	water, girl, art, white, summer	water, white, UK, summer, street
EGCap: eq. (6-7-8)	flower, macro, color, nature, green	art, city, water, street, white	art, family, portrait, city, graffiti

Fig.5 Examples of annotations output by EGCap on images extracted from the testing set of Flickr12k database. The top-5 auto-tags are displayed.

References

1) D. Blei and M. Jordan : Modeling annotated data, Proc. SIGIR'03, 2003.

2) G. Carneiro et al. : Supervised learning of semantic classes for image annotation and retrieval, IEEE Trans. Pattern Anal. Mach. Intell., 29(3), (2007), pp.394–410.

3) P. Duygulu et al. : Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary, Proc. European Conference in Computer Vision (ECCV'02), (2002).

4) S. L. Feng, R. Manmatha, and V. Lavrenko : Multiple Bernoulli relevance models for image and video annotation, Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'04), (2004).

5) D. Hardoon et al. : A correlation approach for automatic image annotation, Z. O. Li, X. and E. Li, Z., editors, Proc. of Second International Conference on Advanced Data Mining and Applications (ADMA), China, (2006), pp. 681–692.

6) H. Hotelling : Relations between two sets of variants, Biometrika, 28, (1936), pp.321–377.

7) J. Li and J. Z. Wang : Real-time computerized annotation of pictures, IEEE Trans. Pattern Anal. Mach. Intell., 30(6), (2008), pp.985–1002.

8) J. Liu et al. : An adaptive graph model for automatic image annotation, MIR'06:Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, ACM, (2006), pp.61–70.

9) T. Mäenpää : The local binary pattern approach to texture analysis - extensions and applications, PhD thesis, Infotech Oulu and Department of Electrical and Information Engineering, University of Oulu, (2003).

10) T. Melzer : Generalized Canonical Correlation Analysis for Object Recognition, PhD thesis, Vienna University of Technology, Institute of Automation, 2002.

11) Y. Mori, H. Takahashi, and R. Oka : Image-to-word transformation based on dividing and vector quantizing images with words, Proc. International Workshop on Multimedia Intelligent Storage and Retrieval Management, (1999).

12) H. Nakayama et al. : Ultra high speed image annotation/retrieval method by learning the conceptual relationship between images and labels, IEICE tech. report, volume 107 of PRMU2007-147, December, (2007), pp.65–70.

13) J.-Y. Pan et al. : Knowledge Discovery and Data Mining: Challenges and Realities with Real World Data, chapter Cross-modal Correlation Mining Using Graph Algorithms, Information Science Reference, June, (2006), p.274.

14) J.-Y. Pan et al. : GCap: Graph-based automatic image captioning, Proc. of the 4th International Workshop on Multimedia Data and Document Engineering (MDDE), in conjunction with Computer Vision Pattern Recognition Conference (CVPR'04), (2004).

15) M. Pucher : Performance evaluation of Wordnet-based semantic relatedness measures for word prediction in conversational speech, Proc. of Sixth International Workshop on Computational Semantics, Tilburg, Netherland, (2005).

16) H. Tong, C. Faloutsos, and J.-Y. Pan : Fast random walk with restart and its applications, Proc. of the Sixth International Conference on Data Mining, ICDM'06, Washington, DC, USA, IEEE Computer Society, (2006), pp.613–622.

17) R. Zhang et al. : A probabilistic semantic model for image annotation and multimodal image retrieval, Proc. of the Tenth IEEE International Conference on Computer Vision (ICCV'05), (2005).