An Adaptive Image Content Representation and Segmentation Approach to Automatic Image Annotation

Rui Shi[1], Huamin Feng[1, 2], Tat-Seng Chua[1], Chin-Hui Lee[3]

[1] School of Computing, National University of Singapore, Singapore
{shirui, fenghm, chuats}@comp.nus.edu.sg
[2] Beijing Electronic Science & Technology Institute, 100070, China
[3] School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA
chl@ece.gatech.edu

Abstract. Automatic image annotation has been intensively studied for content-based image retrieval recently. In this paper, we propose a novel approach to automatic image annotation based on two key components: (a) an adaptive visual feature representation of image contents based on matching pursuit algorithms; and (b) an adaptive two-level segmentation method. They are used to address the important issues of segmenting images into meaningful units, and representing the contents of each unit with discriminative visual features. Using a set of about 800 training and testing images, we compare these techniques in image retrieval against other popular segmentation schemes, and traditional non-adaptive feature representation methods. Our preliminary results indicate that the proposed approach outperforms other competing systems based on the popular Blobworld segmentation scheme and other prevailing feature representation methods, such as DCT and wavelets. In particular, our system achieves an F_1 measure of over 50% for the image annotation task.

1 Introduction

Recent advances in computer, telecommunications and consumer electronics have brought forward a huge amount of images to a rapidly growing group of users. With the wide spread use of Internet, more and more digital images are now available on the World-Wide Web (WWW). Thus effective tools to automatically index images are essential in order to support applications in image retrieval. In particular, image annotation has become a hot topic to facilitate content-based indexing of images.

Image annotation refers to the process of automatically labeling the image contents with a predefined set of keywords representing image semantics. It is used primarily for image database management. Annotated images can be retrieved using keyword-based search, while non-annotated images can only be found using image-based analysis techniques, which are still not very accurate nor robust. Thus automatic image annotation (AIA) aims to invest a large amount of preprocessing efforts to annotate the images as accurately as possible to support keyword-based image search.

Recent studies [14] suggest that users are likely to find it more useful and convenient to search for images based on text annotations rather than using visual-based features.

Most current automatic image annotation (AIA) systems are composed of three key modules: image component decomposition, image content representation, and content classification. A general framework of AIA is shown in Figure 1. The image component decomposition module decomposes an image into a collection of sub-units, which could be segmented regions, equal-size blocks or the entire image. The image content representation module models each content unit based on a feature representation scheme. Finally, the image content classification module computes the association between unit representations and textual concepts and assigns appropriate high-level concepts to the sub-image. Hence we need to answer the following questions:

1) What image components should be used as image analysis units?
2) How to develop better feature representation to model image contents?
3) How to describe the relationships between image components so as to build the relationships between these components and high-level annotations?

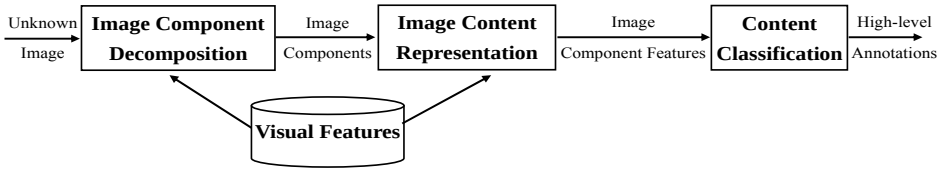

Figure 1 The general framework of Automatic Image Annotation (AIA) system

To address these three problems, most recent research focused on Problems 1 and 3. For Problem 1, three kinds of image components are often used as image analysis units in most CBIR (content-based image retrieval) and AIA systems. In [11, 18], the entire image was used as a unit. Only global features can be used to represent images. Such systems are usually not effective since the global features cannot capture the local properties of an image well. Some recent systems use segmented regions as sub-units in images [3, 5]. However, the accuracy of segmentation is still an open problem. To some extent, the performance of annotation or retrieval depends on the results of segmentation. As a compromise, several systems adopt fixed-size sub-image blocks as sub-units for an image [12, 19]. The main advantage is that block-based methods can be implemented easily. In order to compensate for the drawbacks of block-based method, hierarchical multi-resolution structure is employed [23]. When compare with region-based methods, the block-based methods often result in worse performance.

A lot of research work has also been conducted to tackle Problem 3. In [23], 2-D MHHMs are used to characterize the relationship between sub-image blocks. This model explores statistical dependency among image blocks across multiple resolution levels as well as within a single resolution. In [17], the relationship between image regions with spatial orderings of regions is considered by using composite region templates. In [8], cross-media relevance models are used to describe the association between segmented regions and keywords.

As far as we know, not much research has been done to address Problem 2. Conventional CBIR approaches employ color, simple texture, and statistical shape fea-

tures to model image content [11, 16, 18]. It is well known that such low-level content features are inadequate, and thus the resulting CBIR system generally has low retrieval effectiveness. Moreover, the retrieval effectiveness depends largely on the choice of query images, and the diversity of relevant images in the database. In order to ensure high accuracy, special purpose systems tend to rely on domain-specific features, such as the use of face detectors and face recognizers to look for images of people [11]. Since a single, fixed content representation is unlikely to meet all the needs of different applications, the challenge here is to explore adaptive content representation schemes to support a wide range of classification tasks.

In this paper we propose an adaptive and effective representation that has a high content representational power beyond the traditional low-level features, such as color, texture and shapes. In particular, we extend the adaptive matching pursuit (MP) features [2, 10] to model the texture content of images. In conjunction with these adaptive features, we also propose a two-level segmentation method to partition the image content into more appropriate units. We adopt the SVM-based classifiers employed in [7] to perform image annotation. We evaluated the proposed framework on an image annotation task using a set of about 800 training and testing images selected from the CorelCD and PhotoCD image collections. Preliminary results showed that the adaptive approach outperformed other competing systems based on the popular Blobworld segmentation scheme and other prevailing feature representation methods, such as DCT and wavelets. In particular, our system achieves an F_1 measure of over 50% on the image annotation task.

The rest of the paper is organized as follows. In Section 2 we introduce the proposed framework with adaptive MP features and two-level segmentation. In Section 3 we discuss experimental results and compare our performance with other systems. Finally we conclude our findings in Section 4.

2 Adaptive Image Content Representation

As discussed earlier, almost all the existing systems used a combination of color, texture and statistical shape features to model the visual contents of images [11, 16, 18]. These features have been found to be too low-level to adequately model the image content. Because the discrimination power of these visual features is usually quite low, they are effective only in matching highly similar images, and often fail if there exist diversity in relevant images, or when the query is looking for object segments within the images. These problems point to the need to develop adaptive scheme, in which feature representation can be adapted to suit the characteristics of the images to be modeled. Here we propose the adaptive texture features based on matching pursuit (MP) [2, 10], to be used in conjunction with color histogram, to represent the image content. We do not use the shape feature as it is often unreliable and easily affected by noise. As a part of this work, we also propose a two-level segmentation method to segment the image content into meaningful units. In the following, we introduce the proposed adaptive MP features and the two-level segmentation method.

2.1 Adaptive Texture Features Based on Matching Pursuit

Tuceryan and Jain [20] identified five major categories of features for texture identification: statistical, geometrical, structural, model-based, and signal processing features. In particular, signal processing features, such as DCT, wavelets and Gabor filters, have been used effectively for texture analysis in many image retrieval systems [11, 15, 17]. The main advantage of signal processing features is that they can characterize the local properties of an image very well in different frequency bands. However, specific images usually contain a lot of local properties that need to be characterized individually. In order to facilitate adaptive image representation, we borrow the concept from matching pursuit [2, 10] and employ a combination of DCT and three wavelets as the basis functions to construct an over-complete dictionary in our system. The three wavelets chosen are Haar, Daubechies and Battle/Lemarie, which are often used for texture analysis [20, 23]. We did not use the Gabor filters because they are non-orthogonal and expensive to construct.

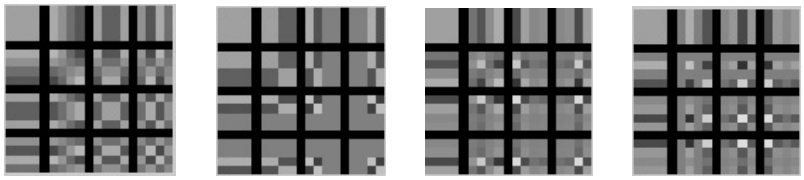

Figure 2 All the basis functions employed in our system

The basis functions for DCT and different types of wavelets are shown in Figure 2. It can be seen that these basis functions have different abilities to represent the details of images using different local properties. For example, images with sharp edges such as modern buildings are better modeled using Haar wavelets; signals with sharp spikes are better analyzed by Daubechies' wavelets; whereas images with soft edges such as clouds are better modeled using DCT basis functions. Thus given a band of basis functions for DCT and different wavelets, we should be able to find a representation that best matches the image content.

The basic idea behind wavelet and DCT transforms is similar. In DCT, a signal is decomposed into a number of cosines of different frequency bands; whereas in wavelet transform, a signal is decomposed into a number of chosen basis functions. To extract the MP features, we divide the image into fixed size blocks of 4x4 pixels. All these basis functions are then partitioned into 16 frequency bands in the horizontal, vertical and diagonal directions of DCT and wavelet transforms.

The algorithm for adaptive MP texture feature extraction can be described as follows. Let $F=f(x,y)$, $(1\le x\le N, 1\le y\le N, N=4)$, where F is an 2D image block and $f(x,y)$ denotes the intensity value at location (x,y). We first transform the 2D image block F into a vector I, column by column or row by row, with $I=(f(1,1), f(1,2),\ldots, f(N,N))^T$, $N = 4$. Thus, there are a total of $N\times N$ elements in vector I. We construct all the basis functions in the over-complete dictionary [2, 10] in a similar manner.

Next, we assumed that the signal space Ω is an $R^{N\times N}$ Hillbert space, with an inner product $\langle\cdot,\cdot\rangle$, and an induced norm denoted as $\| I \| = \langle I,I\rangle^{1/2}$. Assume that $D \subset \Omega$ is

a dictionary of M basis functions in Ω, with $D = \{w_1, w_2, \ldots, w_M\}$. Without loss of generality, it is assumed that $\| w_j \| = 1$ for every basis function (or word), w_j. Then, the feature extraction algorithm can be described in the following steps:

1) Set $I_0 = I, j = 1$.
2) Compute $w_j = \arg\max_{w \in D} \langle I_{j-1}, w \rangle^2$.
3) Let $a_j = \langle I_{j-1}, w_j \rangle$ and $I_j = I_{j-1} - a_j w_j$.
4) Repeat Steps 2 and 3 with $j \leftarrow j+1$ until $\| I_j \|^2 < \varepsilon$, a pre-fixed threshold.
5) Compute the energy value for each frequency band by the coefficient vector $(a_1, a_2, \ldots, a_M)$ which is obtained in Steps 1-4. Thus the 16-dimension vector of energy value is used as adaptive MP texture feature for each image block.

Comparing with the conventional wavelet-based texture features, the main advantages of our adaptive MP texture features are that they are efficient and provide adaptive reflection of local texture properties. This is because we are able to obtain the most appropriate representation for an image with the fewest significant coefficients through matching pursuit.

2.2 An Adaptive Two-Level Image Segmentation Method

In [22], it is argued that an internal spatial/frequency representation in human vision system is capable of preserving both global information and local details. In order to emulate human vision perception, we propose a two-level segmentation method to segment an image into meaningful regions by taking advantage of the adaptive MP features introduced above. Compared with the traditional multi-resolution and hierarchical segmentation methods [5, 6], our algorithm does not need to build complex segmentation schemes, including hierarchy structure for organizing and analyzing image content, and criteria for growing and merging regions. In addition, as the segmentation problem is task dependent, we consider segmentations at different levels as different tasks. Since the segmentation from global level to local level is a process of gradual refinement, we therefore use different features and methods for segmentations at different levels, as shown in Figure 3.

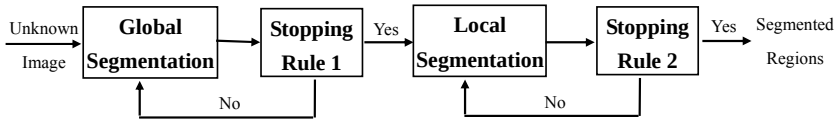

Figure 3 The process of our two-level segmentation method

The image is first partitioned into a large number of small 4x4 blocks. At the global level, we extract mostly global features such as color, texture and position for each block. Here, color is a 3-dim feature vector depicting the average (L, u, v) color components in a block. We apply DCT transform to the L component of the image block and extract a 3-dim texture vector, consisting of the energy values of the frequency bands in the horizontal, vertical and diagonal directions. Finally, we append the center position of the block (x,y) to the feature vector. We perform global segmentation using GMM (Gaussian Mixture Model), which has been used extensively

and successfully in automatic speech and speaker recognition [9] to model non-Gaussian speech features. In order to reduce the complexity in estimating the GMM parameters, we adopt a diagonal covariance matrix with the same variance for all elements. In the meantime, we employ the MDL principle [13] to determine the number of mixture components, and use it as the Stopping Rule 1 in Figure 3. The number of mixture components used ranges from 2 to 4.

For local segmentation, we use employ the adaptive MP features as discussed in Section 2.1 as it better model the content of global regions. For the 19-dim MP feature vector, three are the average of (L, u, v) color components in a 4x4 block, and the other 16 are the adaptive MP texture features. We adopt the *K*-means algorithm to perform local segmentation. Since the *K*-means algorithm does not specify the number of clusters, we adaptively choose *k* by gradually increasing its value *(k=1, 2 or 3)* until the distortion $D(k) - D(k-1)$ is below a threshold, with

$$D(k) = \sum_{i=1}^{N} \min_{1 \le j \le k} (x_i - \hat{x}_j)^2 \tag{1}$$

where $\{x_i : i = 1, \ldots, N\}$ are the N observations, and $\{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_k\}$ are the k group means. Eq. (1) also serves as the basis for Stopping Rule 2 in Figure 3.

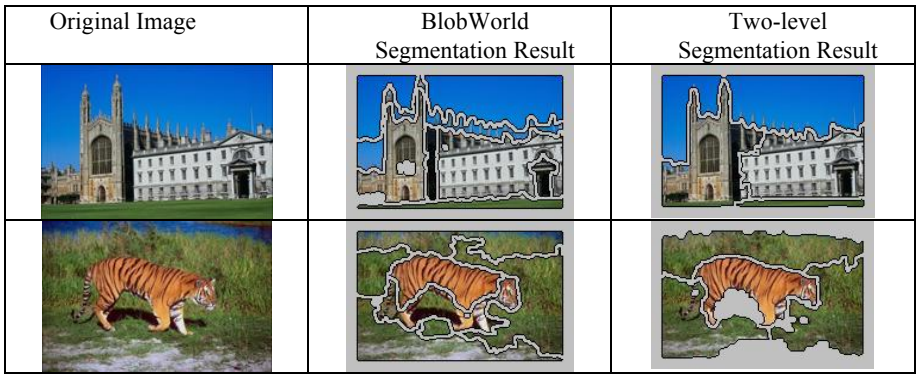

Figure 4 Some segmentation results by Blobworld and our two-level segmentation method

Figure 4 compares two segmentation results obtained using the proposed two-level scheme and Blobworld. It can be seen that our segmentation method tends not to over-segment the regions as in the case for Blobworld. There are two possible reasons. First, we need to estimate fewer parameters. The estimates tend to be more accurate with less training sample than those based on the full covariance matrix used in the Blobworld algorithm. Second, our method segments an image into regions in two steps by considering both global and local features, thus it can reduce over-segmentation.

3 Testing and Results

In order to evaluate the discrimination power of the new adaptive features and the effectiveness of our two-level segmentation method, we designed two sets of experiments based on about 800 images, chosen from the Corel and PhotoCD image collec-

tions. These images were first segmented into regions by using Blobworld and our two-level segmentation method. For our method, we used the second (or local) level of segmentation as the basis for annotation. After segmentation, we obtained over 4,000 segmented regions by Blobworld, and over 5,000 second level regions by our two-level segmentation method. To prepare the ground truth, we manually tagged each resulting segmented region using one of 23 concepts as listed in Table 1. The concepts are chosen based on the hierarchical concepts described in TGM I (Thesaurus for Graphics Materials) [24]. We chose these concepts based on the following two criteria: (a) the concept has concrete visual signatures; and (b) we are able to gather sufficient number of images for training and testing. When manually tagging the segmented regions, we came across many fragmented and meaningless regions because of the problems of segmentation methods. For such regions, we simply tagged them as "none". During testing, when "none" is detected, we simply discard it.

We employed SVM with RBF kernels [4] to perform image and sub-image classification as is done in [7]. In the training stage, we train a binary SVM model for each concept. In the testing stage, we pass the un-annotated segmentation of an image through all the models, and assign only the concept that corresponds to the model giving the highest positive result to the segment. Thus the training is based on segmented regions; while testing is done at the image level. We performed 10 fold cross validation by randomly choosing 80% of images from the corpus for training, and the remaining 20% for testing.

Table 1 The list of 23 concepts used to annotate images.

animals, vehicles, beaches, mountains, meadows, buildings, transportation facilities, office equipments, food, clouds, sky, snow, sunrises/sunsets, grasses, trees, plants, flowers, rocks, clothing, people, water, none, unknown

Experiment 1 aims to evaluate the effectiveness of our adaptive MP features, independent of the segmentation method used. We therefore used only the segmentations produced by the popular Blobworld method to perform image annotation. We tested the performance of the systems by combining color histogram with different texture models, including DCT, Haar wavelet, Daubechies' wavelet, Battle/Lemarie wavelet, and our adaptive MP features, as shown in Table 2. The top four rows of Table 2 provide baseline benchmarks. It can be seen that our MP features produced the best image annotation results in terms of both recall and F_1 measures. The precision results are slightly worse than those obtained with the other competing features. This indicates that the discrimination power of the adaptive MP features can still be enhanced with an adaptive segmentation algorithm.

Experiment 2 was designed to test the effectiveness of the adaptive MP features using the proposed two-level segmentation method. We used a combination of global feature, color histogram and different texture features (same as in Experiment 1) to model the content. The global features are available only in our two-level segmentation method (see Section 2.2); but not in Blobworld method. The global feature used here is a 6-dim vector, consisting of 3 for Luv mean and 3 for DCT textures. The results of Experiment 2 are listed in Table 3, which again shows that the use of adaptive MP texture features give the best image annotation performance as compared to

all other feature combinations. In fact, it produces the best F_1 measure of over 0.5 when using both two-level segmentation and adaptive MP feature extraction.

Tables 2 and 3 also demonstrate that the proposed two-level segmentation scheme is clearly superior to the single-level Blobworld method, because the results in Table 3 are consistently better than those in Table 2 in all cases, especially for recall and F_1.

Table 2 Results of Experiment 1 based on Blobworld (single-level) segmentation.

Feature Type	Recall	Precision	F_1
Luv hist + [DCT]	0.2859	0.3867	0.3287
Luv hist + [Haar]	0.2907	0.3972	0.3357
Luv hist + [Daube]	0.2791	0.3910	0.3257
Luv hist + [Battle]	0.2901	0.3920	0.3334
Luv hist + MP features	0.3544	0.3836	0.3684

Table 3 Results of Experiment 2 based on our two-level segmentation.

Feature Type	Recall	Precision	F_1
Global features+Luv hist+[DCT]	0.512	0.4022	0.4505
Global features+Luv hist+[Haar]	0.5181	0.4079	0.4564
Global features+Luv hist+[Daube]	0.5124	0.4104	0.4558
Global features+Luv hist+[Battle]	0.519	0.4132	0.4601
Global features+Luv hist + MP features	0.5642	0.454	0.5031

4 Conclusion

In this paper, we proposed a novel adaptive content representation scheme with two key components: (a) adaptive matching pursuit feature extraction for texture; and (b) adaptive two-level segmentation method. We compared our proposed methods with popular single-level segmentation, like Blobworld, and conventional feature representations, based on color and DCT or wavelets. The results indicate that our proposed adaptive approach outperformed other competing methods in most combinations, especially when the two-level segmentation algorithm is employed. In particular, our combined scheme achieved an F_1 measure of over 50%. The overall results suggest that some of the difficulties in content-based image retrieval and automatic image annotation could be mitigated by jointly considering both the segmentation and feature representation issues. It will also be worth looking into simultaneous segmentation and classification, commonly done in the state-of-the-art automatic speech and speaker recognition systems.

Our future research is focused in two directions. First, we will refine our adaptive approach and test it on a large collection of image sets. Second, we will extend the proposed techniques to handle video and web-based multimedia contents.

References

[1] Y. A. Aslandogan and C. T. Yu, "Multiple evidence combination in image retrieval: Diohenese searches for people on the web," *ACM SIGIR'2000*, Athens, Greece, 2000.

[2] F. Bergeaud and S. Mallat, "Matching pursuits of images," Proc. *IEEE ICIP'95*, Vol. 1, pp. 53-56, Washington DC, Oct 1995.
[3] C. Carson, M. Thomas, S. Belongie, J.M. Hellerstein and J. Malik, "Blobworld: A system for region-based image indexing and retrieval," *Proc. Int'l Conf. Visual Information System*, 1999.
[4] C.-C. Chang and C.-J. Lin, "Training nu-support vector classifiers: theory and algorithms," *Neural Computation*, 13 (9), pp. 2119-2147, 2001.
[5] Y. Deng, B. S. Manjunath, and H. Shin, "Color image segmentation," in *Proc. IEEE CVPR*, 1999.
[6] P. Duygulu and F. Y. Vural, "Multi-Level image segmentation and object representation for content based image retrieval," *SPIE Electronic Imaging 2001*, *Storage and Retrieval for Media Databases*, January 21-26, 2001, San Jose, CA.
[7] H.M. Feng and T.S. Chua, "A Bootstrapping Approach to Annotating Large Image Collection". ACM SIGMM International Workshop on Multimedia Information Retrieval. Berkeley, Nov 2003. 55-62.
[8] J. Jeon, V. Lavrenko and R. Manmatha, "Automatic image annotation and retrieval using cross-media relevance models," *ACM SIGIR'03*, July 28-Aug 1, 2003.
[9] C.-H. Lee, F. K. Soong and K. K. Paliwal, *Automatic Speech and Speaker Recognition: Advanced Topics*, Kluwer Academic Press, 1996.
[10] S.G. Mallat and Z.F. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. on Signal Processing*, 41 (12), pp. 3397-3415, 1993.
[11] B. Manjunath and W. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837-842, Aug 1996.
[12] Y. Mori, H. Takahashi and R. Oka, "Image-to-word transformation based on dividing and vector quantizing images with words," In *Proc. of First International Journal of Computer Vision*, 40(2): 99-121, 2000.
[13] J. Rissanen, "Modeling by shortest data description," *Automatica*, 14:465-471, 1978.
[14] K. Rodden, "How do people organize their photographs?" In *BCS IRSG 21st Ann. Colloq. on Info. Retrieval Research*, 1999.
[15] M. Shenier and M. Abedel-Mottaleb, "Exploiting the JPEG compression scheme for image retrieval," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18 (8), 849-853, 1996.
[16] J.R. Smith and S.F. Chang, "VisualSeek: A fully automated content-based query system," *ACM Multimedia*, 1996.
[17] J.R. Smith and C.S. Li, "Image classification and querying using composite region templates," *Journal of Computer Vision and Image Understanding*, 2000.
[18] M. Swain and D. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11-32, 1991.
[19] M. Szummer and R.W. Picard, "Indoor-outdoor image classification," *IEEE Intl Workshop on Content-based Access of Image and Video Databases*, Jan 1998.
[20] M. Tuceryan and A.K. Jain, "Texture analysis," Handbook Pattern Recognition and Computer Vision, Chapter 2, pp. 235-276, World Scientific, 1993.
[21] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Trans. on Image Processing*, 4 (11), pp. 1549-1560, 1995.
[22] R. D. Valois and K.D. Valois, "Spatial Vision," New York: Oxford, 1988.
[23] J.Z. Wang and J. Li, "Learning-based linguistic indexing of pictures with 2-D MHMMs," *Proc. ACM Multimedia*, pp. 436-445, Juan Les Pins, France, Dec 2002.
[24] http://www.loc.gov/rr/print/tgm1/