AdaBoost for Concrete Type of Keywords Annotation

Wei-Chao Lin, Yan-Ze Chen and Shu-Yuan Chen

***Abstract*—The semantic gap presents an arduous task in semantic-based image retrieval investigations. In this paper, the author proposes the AdaBoost learning algorithm for large vocabulary classification. The main finding of this investigation is that using Gentle AdaBoost for image classification produced excellent results in terms of precision and the F-measure. With 190 concrete keywords categorisation, AdaBoost renders more keywords assignable and allows a significant improvement in all accuracy measures: precision, recall and F-measure. An AdaBoost vs. SVMs comparison showed that AdaBoost was an effective classifier using the one-versus-the-rest mode of operation**

***Index Terms*—Image annotation、Content-based image retrieval、AdaBoost learning algorithm**

I. Introduction

Visual information has become very important for many applications. Millions of people access digital images from the Internet, digital cameras and mobile phones each day. Retrieval of images is facilitated if the images are annotated with keyword descriptors of the image contents. Manual annotation of large image collections is both time consuming and expensive; as a result, there is currently substantial interest in automatic image annotation based on an analysis of the image contents by the computer. Content-Based Image Retrieval (CBIR) is a set of techniques for retrieving digital images based on their low-level image features, typically colour, texture and edges. However, the semantic gap between low-level features and high-level concepts is the major problem of CBIR.

Recently, a type of meta-heuristic principle, the AdaBoost learning algorithm, has been suggested for improving the performance of image processing. The AdaBoost learning algorithm is an emerging meta-algorithmic machine learning technique first developed by Freund & Schapire[1]. This technique has become popular in multimedia and visual pattern recognition. A small number of researchers have used AdaBoost as a classifier for image categorisation (e.g., Howe[2] ; Yuan et al.[3]; Zhang et al.[4]), although only up to 10 categories have been used (cf. TABLE). The main goal for this paper is to examine the success of AdaBoost with large numbers of controlled categories. The experiments employ Gentle AdaBoost for comparison with widely used supervised learning models, such as *k*-Nearest Neighbour (*k*-NN) and Support Vector Machines (SVMs) over concrete keywords from the Corel image collection.

The remainder of this paper is organised as follows. Section 2 reviews related work in the field of Content-based image retrieval (CBIR) and its challenges with respect to current approaches. Section 3 provides an overview of the AdaBoost learning algorithm, and then, Gentle AdaBoost is introduced. The main experiment is described in Section 4, where we compare classification results between Gentle AdaBoost, *k*-NN and SVMs for the Corel data set. Finally, section 5 summarises and discusses these system-centred experimental studies and provides ideas for future investigations.

II. Content-based image retrieval

Content-Based Image Retrieval (CBIR), which was proposed in the early 1990s, has been an active research area for over a decade. This field attempts to provide search methodology for retrieving images by the content of the images themselves. The main merits are the provision of the following: (i) the capability to support visual queries, (ii) friendly and intuitive querying for users and (iii) the creation of image content feature descriptions automatically [5].

An automatic image annotation system for CBIR is based on the extraction and indexing of an image's low-level features, which might include colour, texture and shape. The aim is to support visual queries in an intuitive way and to annotate images automatically with content descriptors. The process consists of three components, which are image segmentation, feature extraction and classification. Thus, to annotate an image with one or multiple keywords, an image is first segmented into a number of 'meaningful' regions by image segmentation algorithms. Then, low-level features are extracted from each of the segmented regions. As a result, the low-level features are used to represent or describe the local content of the image. Finally, some machine learning algorithms are used to learn these low-level feature representations for automatic image annotation. In other words, they are trained to recognise which low-level features correspond to which keywords (the keywords reflect high-level concepts).

Manuscript received December 8, 2017. This work was supported in part by the Ministry of Science and Technology of Taiwan under Contract MOST 105-2221-E-155 -060 -MY2.

Wei-Chao Lin is with the Department of Information Management, Chang Gung University, Taoyuan, Taiwan and Department of Thoracic Surgery, Chang Gung Memorial Hospital, Linkou, Taoyuan, Taiwan (corresponding author to provide phone: +886-3-2118800 #3770; fax: +886-3-2118020; e-mail: viclin@gap.cgu.edu.tw).

Yan-Ze Chen is with the Department of Computer Science and Engineering, Yuan Ze University, Chung Li, Taiwan (e-mail: s1056004@mail.yzu.edu.tw).

Shu-Yuan Chen is with the Department of Computer Science and Engineering, Yuan Ze University, Chung Li, Taiwan (e-mail: cschen@saturn.yzu.edu.tw).

Many related studies focus on proposing novel learning algorithms for better annotation accuracy. For example, in a representative study, Barnard et al.[6] cluster a number of segmented regions (blobs) of images that represent low-level features, and the words associated with these images in the training set are tokenised. A learning algorithm is used to learn a lexicon to link the blob tokens with the word tokens. A graph learning framework for image annotation is proposed by Liu et al.[7]. Specifically, a Nearest Spanning Chain method is used to construct the image-based graph, and the relationships between images and words are refined through the word-based graph, with the system learning to produce final annotations for each image. Tang et al.[8] focus on finding one prototype of an instance (represented by low-level features) for each concept (i.e., a keyword) and mapping the multiple instance features of every image into the feature space of the selected prototypes. Specifically, they integrate local and global instances of each image for annotation over 70 concepts (i.e., keywords). However, feature selection has been considered for improving image annotation accuracy. The aim of feature selection is to filter out unrepresentative features, which are likely to degrade the performance[9].

The semantic gap[10], which occurs in the translation of low-level features into high-level concepts, presents an arduous task to image annotation systems. This problem causes some images to not be assigned to their related keywords, as most CBIR systems can only work with small controlled vocabularies (usually less than 150 keywords). For this reason, users cannot be satisfied with a current system's performance. Therefore, deciphering this problem has been the main research focus in many recent investigations. This paper suggests using the AdaBoost learning algorithm for learning model reformulation to bridge the semantic gap problem.

III. ADABOOST LEARNING ALGORITHM

A. Overview

Boosting is a meta-algorithmic machine learning technique for improving supervised learning performance. The first Boosting system was developed by Schapire[11] and provided a simple procedure in the Probably Approximately Correct (PAC) learning framework [12]. Furthermore, Schapire provided support for the hypothesis of Kearns & Valiant [13]: that a weak learner could improve its performance when given filtered versions of the input data. Boosting is not an algorithmically constrained technique because it applies a template of classification algorithms and works in iterations, by sequentially reweighting instances of the training data and then taking the majority vote of the data in the training sequence to revise the weak learner. Based on this simple strategy, by incrementally complementing an initial set of weak learners to form a final strong learner, highly accurate prediction can be produced.

AdaBoost (Adaptive Boosting) is the most widely used boosting learning algorithm in visual pattern recognition and was used, for example, by Tieu & Viola[14] and Liu & Yuen[15]. AdaBoost adjusts adaptively to errors in the weak hypotheses and depends on instances of a previous learning model to weight the most important feature vector with the lowest error relative to the next learner in the sequence. The flowchart of AdaBoost is illustrated in Fig.1. The probability estimate p_t starts with the weights $w_{1,i}$ of the input training examples $(x_i, y_{1,i})$, then constructs a valued contribution $f_t(x)$ to update $w_{t,i}$ in the next sequence. Once all valued contributions have been defined, they are combined to produce a final hypothesis $F(x)$, which is a final strong learning model for a classifier.

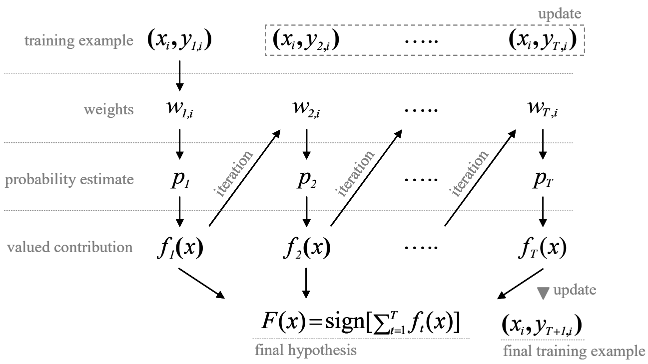


Fig. 1. Implemented flowchart of the AdaBoost learning algorithm.

B. Gentle AdaBoost

In the history of AdaBoost, there have been various modifications for algorithm improvement, such as AdaBoost.M1 [16], Real AdaBoost [17], Gentle AdaBoost [18] and Modest AdaBoost (Vezhnevets & Vezhnevets[19]). Among them, the Gentle AdaBoost (described in Algorithm 1) is a more robust and efficient version of AdaBoost and will be employed in this paper. This algorithm is a modified version of the Real AdaBoost. Friedman et al. (2000) inserted a logistic regression model to minimise the exponential loss of when updated through the Newton steps. However, Gentle AdaBoost omits the log-ratios update, which allows the regression function to be considered with a gentler alternative. Some experiments, such as the work of Lienhart et al.[20] and Martínez-Ponte[21] have verified that Gentle AdaBoost provides an excellent performance that outperforms other versions of the AdaBoost algorithm in terms of detected accuracy.

IV. EXPERIMENTS

In the literature, AdaBoost has been employed as a classifier for image classification, as listed in TABLE I, but was applied very infrequently and only for up to 10 categories of indexing.

A. Corel image collection

Corel is the most frequently used image collection in image retrieval research (e.g., Barnard et al.[6] and Tao et al.[22]) because it covers a wide range of different perspectives and the pictures have been taken by professional photographers. Corel supplies more than 800 topics, such as "antelope", "golf" and "London". Each category contains 100 images.

However, Corel is not a royalty-free data set. This project

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Algorithm 1. Gentle AdaBoost learning algorithm [18].

1. Input a training example $(x_i, y_{1,i})$, $i = 1,2,...,N$

   x_i is the feature vector value; $y_i \in \{1,-1\}$ for positive and negative examples; N is the total number of feature vectors.
2. Initial weights on the original training data

   $w_{1,i} = 1/N$, $i = 1,2,...,N$
3. Set $F(x) = 0$
4. Repeat for $t = 1,2,...,T$:

   (a) Obtain a probability estimate using weights $w_{t,i}$

   $$p_t(x) = P(y = 1 \mid x) \in [0,1]$$

   (b) Fit the regression function $f_t(x)$

   $$f_t(x) \leftarrow P(y = 1 \mid x) - P(y = -1 \mid x) \in R$$

   (c) Update $F(x) \leftarrow F(x) + f_t(x)$

   (d) Update weights

   $w_{t+1,i} \leftarrow w_{t,i} \exp[-y_i f_t(x_i)]$, and renormalise $\sum_{i=1}^{N} w_{t+1,i} = 1$
5. Output the final hypothesis

   $$F(x) = \text{sign}[\sum_{t=1}^{T} f_t(x)]$$

approaches (i) the four Corel Stock Photo Libraries[1] and (ii) the Corel Gallery 1,300,000[2] for data set organisation. The experiments aim to represent each group of images by a single keyword, to connect low-level features with their related high-level keywords. This structure enables the system to understand what types of features can be explained by specific categories of keywords in the training stage and then to assign relevant keywords to each test image. For this reason, the keyword assigned to each category depends on the original description by the publisher, Corel. Some of the descriptions from Corel were spread over more than one category, such as "pets" and "pets II" or "classic cars" and "classic automobiles". In such cases, only one of the overlapping categories is used. Altogether, there are 190 categories employed in the experiments. These categories are described by 190 individual keywords, one per category.

The WordNet[3] online lexical reference system was applied to discriminate the Corel keywords into concrete and abstract attributes. According to WordNet, a concrete keyword describes a type of physical entity object, such as "antelope", "bus" or "sky", while an abstract keyword involves a type of human activity and/or abstraction, such as "autumn", "golf" or "speed".

[1] A software review is available at: http://www.emsps.com/photocd/corelcds.htm

[2] A software review is available at: http://www.gtpcc.org/gtpcc/corelgallery.htm

[3] For more information about WordNet, please visit http://wordnet.princeton.edu/

B. Implementation

Five-fold cross-validation, based on 80 training and 20 testing images per category was used to avoid the problem of noisy estimates when the system operated with a small validation set to ensure that the results were credible.

The pre-processor includes image segmentation and feature extraction components. First, each image will be resized beforehand into a square, with a 128×128 pixel resolution, to standardise the size of the image frame, to speed up computer processing. Region-based segmentation (using *N*cut) and colour+texture features (HSV colour plus four levels of Daubechies-4 wavelet) were applied. Note that training data were presented and consisted of one feature vector for each image. Multiple segments were utilised (the experiments used 5 to 81 regions), and the central region provided the training information because an artist usually locates the main content in the central area of an image.

Gentle AdaBoost, *k*-NN and SVMs were the main methods used for pattern classification. This process involved the creation of distinct systems, as follows:

1. ***Gentle AdaBoost* system.** This system follows the learning algorithm illustrated in Algorithm 1 to produce a learning model for test image indexing. The system performs one-versus-the-rest classification for every single category, taken individually, to learn its special properties. For the input training example, the authors set x_i as the i^{th} feature vector value out of 19 dimensions, consisting of 3 dimensions of HSV colour features plus 16 dimensions of four-level Daubechies-4 wavelet texture features; $y_i \in \{1,-1\}$ labels positive and negative examples, and N is the total number of feature vectors (for example, N = 15,200 for 190 categories with 80 training images per category). In the experiments, all categories are learnt with the same number of iterations. Moreover, every category must have at least one new training example, as the number of feature vectors, $(x_i, y_{1,i}) \cap (x_i, y_{T+1,i})$, $y_{1,i} = y_{T+1,i} = 1$ must be at least one to mean that the analysis category has been learnt. In other words, the iterations stop when every category has some new training data. Fig.2 shows an example of learning that is achieved when iterations of Gentle AdaBoost are performed with 190 concrete categories, no segmentation and 81 regions. In this case, the authors set 150 and 175 iterations with no segmentation and 81 regions. The implementations were examined every 25 iterations. Although 149 iterations were needed for all categories to acquire new training examples with no segmentation, the authors used 150 iterations to strengthen the final strong learning model because more training examples were discovered.
2. ***K-NN* system.** The *k*-Nearest Neighbour (*k*-NN) is a well-known classifier in pattern classification. In several studies, such as Jain et al.[23] and Tsai & Lin [24], the value of k = 1 (*1*-NN) was chosen in experiments for system evaluation. The *1*-NN classifier can be conveniently used as a benchmark that enables an instinctive categorisation to classify low-level feature

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE I
THE LITERATURE ON ADABOOST THAT WAS EMPLOYED FOR IMAGE CLASSIFICATION.

Work	Pre-processing			Relevance feedback requests				Data set
	segment	feature	dimension	learning algorithm	categories	average accuracy	keywords per image	
Zhang et al. (2002)	B	C+E	19	Discrete AdaBoost	2	78.85%	1	Corel [train=5416 / test=5422]
Howe (2003)	G	C, C, C+T, P	128, 512, 19200, 46875	Real AdaBoost	5	20.99%	N/A	Corel [total=20100]
Yuan et al. (2007)	R	C+T+S	9	MI-AdaBoost	10	78.00%	1	Corel [train=500 / test=500]

Segment: 'B' means block-based segmentation; 'G' means global descriptors (no segmentation); 'R' means region-based segmentation.
Feature: 'C' means colour; 'E' means edge; 'P' means feature map via primitive filters; 'S' means shape; 'T' means texture.
'N/A' means that insufficient information was given within the cited paper.

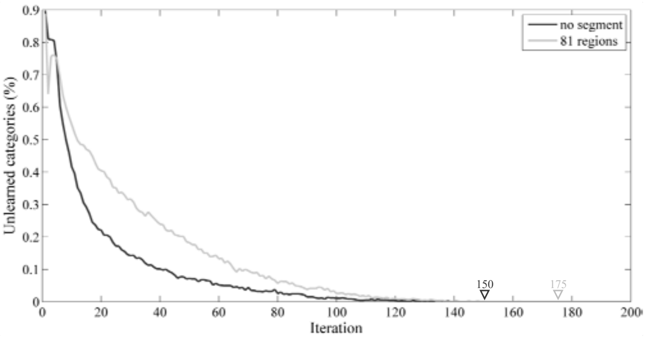


Fig. 2. Learning curves for the Gentle AdaBoost implementation: an example of 190 concrete categories with no segmentation and 81 regions.

vectors into conceptual categories through the training examples and their associated labels.

3. **SVMs system**. Support Vector Machines (SVMs) are also a popular classifier in information retrieval investigations. Most work (e.g., Hervé & Boujemaa[25] and Tsai et al.[26]) apply a one-against-all process, which is described as a one-versus-the-rest in the AdaBoost literature, to reduce the performance time. SVM*light* [27], which has been widely used in text and vision classification tasks (e.g. Hughes et al.[28] and Ke et al.[29]), with a polynomial kernel (c=1, d=3), was applied in the experiments.

To objectively assess system performance, the authors considered system-centred evaluation. *Precision*, *Recall* and *F-measure* [30] are the most common evaluation metrics in many information retrieval experiments. The approach taken in this paper is that retrieved images are images placed in a specific category by the system, and relevant images are identified as images in the same category in the annotated data set, Corel. Precision measures how many of the images retrieved have in fact been placed in the relevant category; recall measures the percentage of images assigned to a category out of the total number in that category in the original data set. Finally, the F-measure (β is set to 1 to produce an equal weight on precision and recall) is the weighted harmonic mean between the precision and the recall. The measure of *unassigned keywords* (U = *unassigned categories / total number of categories*) is also introduced here to show the percentage of keywords that are never assigned to a relevant image. This percentage can provide more understanding of retrieval success than can be obtained from the precision, recall and F-measure alone.

C. AdaBoost vs. k-NN and SVMs

Because a concrete noun that describes a type of physical entity is easier to use during system development (e.g., Barnard et al. [6] and Tsai et al. [26]), this study categorises up to 190 concrete keywords for the Corel data set, with no segmentation to select or examine a classifier, *k*-NN or SVMs. Fig.3 illustrates the experimental framework for this study.

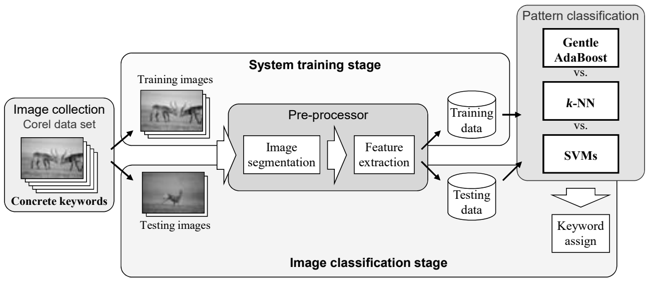


Fig. 3. Experimental framework for AdaBoost vs. *k*-NN and SVMs.

D. Results and comparison

TABLE II lists the experimental setup to evaluate the performance between *Gentle AdaBoost*, *k-NN*, *SVM^{light}* and *random guessing*. Fig.4 shows the experimental results for the assessment.

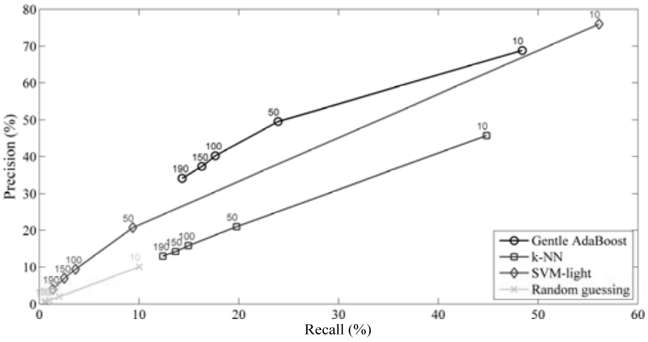
TABLE II
ASSESSMENTS FOR THE STUDY OF *GENTLE ADABOOST* VS. *K-NN* AND *SVM^{LIGHT}*.

	Assessment
Image collection	Corel
Keyword categories	10^{c}, 50^{c}, 100^{c}, 150^{c}, 190^{c} keywords
Image segmentation	no segmentation
Low-level feature	Colour+Texture
Pattern classification	*Gentle AdaBoost* vs. *k-NN* (k = 1) vs. *SVMs* (SVM^{light}, polynomial kernel: c = 1, d = 3) vs. *Random guessing*

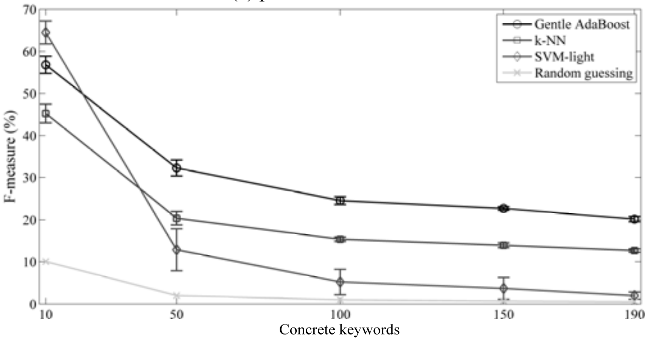
Keyword categories: 'C' means concrete.

The *Gentle AdaBoost* and *k-NN* systems surpassed the *random guessing* baseline. The *Gentle AdaBoost* system also outperformed *k-NN*, especially in terms of precision. On average, *Gentle AdaBoost* achieved almost double the increase in precision over *k-NN* (25.56% compared to 12.18%). This remained true when using larger numbers of keyword categories, with 23.11%, 28.58%, 24.43%, 23.13% and 21.18% improvement for 10, 50, 100, 150 and 190 concrete keywords, respectively. Moreover, the F-measure for *Gentle AdaBoost* was also superior to *k-NN* but was slightly decreased with 190 categories, where the improvement in the F-measure was 7.52% compared with 11.19% for the 50-category case. However, *Gentle AdaBoost* resulted in more keywords, which could not be assigned to their related images. The number of unassignable keywords

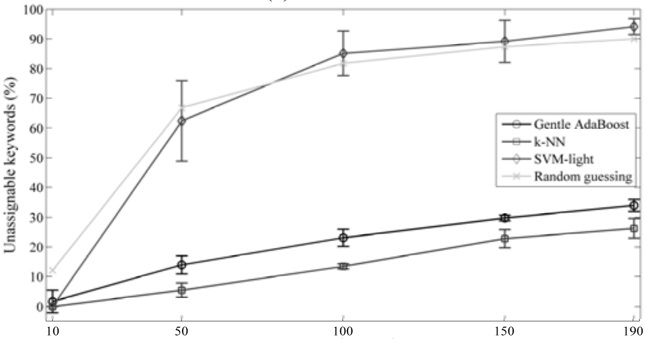
ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)



(a) precision vs. recall



(b) F-measure



(c) unassignable keywords

Fig. 4. Classification performances of *Gentle AdaBoost* vs. *k-NN* and *SVM^light^*: no segmentation over a range of vocabularies of concrete keywords.

was an average of 6.91% greater than that for *k-NN*.

The experimental results for AdaBoost vs. SVMs indicated that *Gentle AdaBoost* is better in this comparison. *SVM^light^* was better than *Gentle AdaBoost* for 10-category categorisation only, and, above this number of categories, *SVM^light^* was better by 7.19%, 7.70%, 7.68% and 1.7% in terms of precision, recall, F-measure and unassignable keywords, respectively. In addition, the *SVM^light^* system performance was very variable, as shown by the long accompanying cross-validation error bars.

The *k-NN* system had the advantage of saving time during training model creation. The *k*-NN classifier can classify an unknown image by directly computing the distance between the test image and the original training set. However, the amount of time consumed is more competitive between Gentle AdaBoost and *SVM^light^* because both systems require training model creation and a one-versus-the-rest operation. *SVM^light^* is very time-consuming for the 190-category implementation. This approach is much more time consuming compared to *Gentle AdaBoost*.

E. Discussion

The *overall* comparison of classification performances of the *Gentle AdaBoost*, *k-NN* and *SVM^light^* systems is shown in TABLE III.

Gentle AdaBoost was poorer than *k-NN* in terms of recall.

TABLE III
PERFORMANCE COMPARISON *OF* THE ADABOOST VS. *K*-NN AND SVM*^LIGHT^*

Accuracy	Recall	high	*k-NN* > *Gentle AdaBoost* > *random guessing*	low
	Precision	high	*Gentle AdaBoost* > *k-NN* > *random guessing*	low
	F-measure	high	*Gentle AdaBoost* > *k-NN* > *random guessing*	low
Unassignable keywords		less	*k-NN* > *Gentle AdaBoost* > *random guessing*	more
Time consumed		fast	*k-NN* > *Gentle AdaBoost*	slow

Nevertheless, *Gentle AdaBoost* showed excellent precision, more than twice that produced by *k-NN*. This is a significant improvement and an important discovery and is a unique contribution of this paper. Moreover, this good precision also means that the *Gentle AdaBoost* system performs well in terms of the F-measure.

In contrast to *k-NN*, *Gentle AdaBoost* causes more keywords to not be assigned to their related images. However, *Gentle AdaBoost* enables a greater accuracy of classification for those keywords that are assignable.

A similarity in the AdaBoost and SVMs methods arises because both systems involve training model creation and one-versus-the-rest operations. This study confirmed that *Gentle AdaBoost* was more economical in terms of computing costs compared to *SVM^light^*. In contrast, SVMs is shown to be good classifier for 10-category categorisation, even surpassing *k-NN* system performance. However, an ideally powerful image searching environment should operate with much larger vocabularies than those in the present examples.

V. CONCLUSIONS AND FUTURE WORK

The categorization ability of AdaBoost with different types of Corel concrete keywords was examined in this paper. Gentle AdaBoost was employed for comparison with widely used supervised learning models, *k*-NN and SVMs, to confirm that AdaBoost is a powerful classifier for image annotation.

AdaBoost vs. k-NN and SVMs, shows that AdaBoost produced significant improvements in terms of precision and F-measure for concrete keyword implementations. In other words, image annotation via AdaBoost can allow the retrieval of more relevant images for a search engine user. Additionally, the AdaBoost vs. SVMs comparison showed that AdaBoost was an effective classifier using the one-versus-the-rest mode of operation.

Altogether, the AdaBoost learning algorithm can be recommended as a supervised learning model for use in CBIR. However, there are two limitations of this investigation, as follows:

1. 190 categories are still a comparatively small annotation vocabulary in a real world searching environment;
2. The Corel data set was never originally designed as a test collection for a study of this sort and may be biased [31, 32].

In future investigations with AdaBoost, we plan to extend this vocabulary by using other stock image collections, such as the IAPR TC-12 Benchmark [31] and the Celtech Benchmarks (Celtech-101 [33] and Celtech-256 [34], and to confirm how this approach performs with other tag-based indexing systems.

REFERENCES

[1] Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55(1), 119-139.

[2] Howe, N. R. (2003). A closer look at boosted image retrieval. In Proceedings of the International Conference on Image and Video Retrieval (CIVR 2003), 24-25 July 2003 (pp. 109-114). Urbana-Champaign, IL, USA.

[3] Yuan, X., Hua, X.-S., Wang, M., Qi, G.-J., & Wu, X.-Q. (2007). A novel multiple instance learning approach for image retrieval based on adaboost feature selection. In Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2007), 2-5 July 2007 (pp. 1491-1494). Beijing, China.

[4] Zhang, L., Li, M., & Zhang, H.-J. (2002). Boosting image orientation detection with indoor vs. outdoor classification. In Proceedings of the 6th IEEE Workshop on Applications of Computer Vision (WACV 2002), 3-4 December 2002 (pp. 95-99). Orlando, FL, USA.

[5] Wu, J. K., Kankanhalli, M. S., Lim, J.-H., & Hong, D. (2000). Perspectives on Content-Based Multimedia Systems. London, Boston, Dordrecht: Springer.

[6] Barnard, K., Duygulu, P., Forsyth, D., De Freitas, N., Blei, D. M., & Jordan, M. I. (2003). Matching words and pictures. Journal of Machine Learning Research, 3, 1107-1135.

[7] Liu, J., Li, M., Liu, Q., Lu, H., & Ma, S. (2009). Image annotation via graph learning. Pattern Recognition, 42(2), 218–228.

[8] Tang, Jinhui, Li, H., Qi, G.-J., & Chua, T.-S. (2010). Image annotation by graph-based inference with integrated multiple/single instance representations. IEEE Transactions on Multimedia, 12(2), 131-141.

[9] Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. The Journal of Machine Learning Research, 3, 1157–1182.

[10] Datta, R., Joshi, D., Li, J., & Wang, J. Z. (2008). Image retrieval: ideas, influences, and trends of the new age. ACM Computing Surveys, 40(2), 5:1-60.

[11] Schapire, Robert E. (1990). The strength of weak learnability. Machine Learning, 5(2), 197-227.Kearns, M., & Valiant, L. G. (1988). Learning Boolean formulae or finite automata is as hard as factoring (Technical report, TR-14-88). Cambridge, MA, USA: Harvard University Aiken Computation Laboratory.

[12] Haussler, D. (1990). Probably approximately correct learning. In Proceedings of the 8th National Conference on Artificial Intelligence, 29 July-3 August 1990 (pp. 1101-1108).

[13] Kearns, M., & Valiant, L. G. (1988). Learning Boolean formulae or finite automata is as hard as factoring (Technical report, TR-14-88). Cambridge, MA, USA: Harvard University Aiken Computation Laboratory.

[14] Tieu, K., & Viola, P. (2004). Boosting image retrieval. International Journal of Computer Vision, 56(1), 17-36.

[15] Liu, C., & Yuen, P. C. (2010). Human action recognition using boosted EigenActions. Image and Vision Computing, 25(5), 825-835.

[16] Freund, Yoav, & Schapire, R. E. (1996). Experiments with a new boosting algorithm. In Proceedings of the 13th International Conference in Machine Learning (pp. 148–156). San Francisco, USA

[17] Schapire, Robert E, & Singer, Y. (1999). Improved boosting algorithms using confidence-rated predictions. Machine Learning, 37(3), 297-336.

[18] Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting. Annals of Statistics, 28(2), 337-407.

[19] Vezhnevets, A., & Vezhnevets, V. (2005). “Modest AdaBoost” - teaching AdaBoost to generalize better. In Proceedings of the Computer Graphics and Applications (GraphiCon ’05), 20-24 June 2005. Novosibirsk Akademgorodok, Russia.

[20] Lienhart, R., Kuranov, A., & Pisarevsky, V. (2003). Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In Proceedings of the 25th Pattern Recognition Symposium (DAGM ’03), 10-12 September 2003 (pp. 297-304). Magdeburg, Germany.

[21] Martínez-Ponte, I., Desurmont, X., Meessen, J., & Delaigle, J.-F. (2005). Robust human face hiding ensuring privacy. In Proceedings of the 6th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2005), 13-15 April 2005. Montreux, Switzerland.

[22] Tao, D., Tang, X., Li, X., & Rui, Y. (2006). Direct kernel biased discriminant analysis: a new content-based image retrieval relevance feedback algorithm. IEEE Transactions on Multimedia, 8(4), 716-727.

[23] Jain, A. K., Duin, R. P. W., & Mao, J. (2000). Statistical pattern recognition: a review. IEEE Transitions on Pattern Analysis and Machine Intelligence, 22(1), 4-37.

[24] Tsai, C.-F., & Lin, C.-Y. (2010). A triangle area based nearest neighbors approach to intrusion detection. Pattern Recognition, 43(1), 222-229.

[25] Hervé, N., & Boujemaa, N. (2007). Image annotation: which approach for realistic databases? In Proceedings of the 6th ACM International Conference on Image and Video Retrieval (CIVR 2007), 9-11 July 2007 (pp. 170-177). Amsterdam, Netherlands.

[26] Tsai, C.-F., McGarry, K., & Tait, J. (2003). Image classification using hybrid neural networks. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 28 July-1 August 2003 (pp. 431-432). Toronto, Canada.

[27] Joachims, T. (2001). Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms. London: Kluwer Academic Publishers.

[28] Hughes, M., Salway, A., Jones, G., & O’Connor, N. (2007). Analyzing image-text relations for semantic media adaptation and personalization. In Proceedings of the 2nd International Workshop on Semantic Media Adaptation and Personalization (SMAP 2007), 17-18 December 2007 (pp. 181-186). London, UK.

[29] Ke, G. S.-W., Oakes, M. P., Palomino, M. A., & Xu, Y. (2008). Comparison between SVM-Light, a search engine-based approach and the mediamill baselines for assigning concepts to video shot annotations. In Proceedings of the 6th International Workshop on Content-Based Multimedia Indexing (CBMI 2008), 18-20 June 2008 (pp. 381-387). London, UK.

[30] van Rijsbergen, C. J. (1979). Information Retrieval (2nd ed.). London: Butterworths.

[31] Grubinger, M., Clough, P., & Leung, C. (2006). The IAPR TC-12 Benchmark for visual information search. IAPR Newsletter, 28(2), 10-12.

[32] Tang, Jiayu, & Lewis, P. H. (2007). A study of quality issues for image auto-annotation with the corel data-set. IEEE Transactions on Circuits and Systems for Video Technology, 17(3), 384-389.

[33] Li, F.-F., Rod, F., & Pietro, P. (2004). Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. In Workshop on Generative-Model Based Vision. Washington DC, USA.

[34] Griffin, G., Holub, A., & Perona, P. (2007). Caltech-256 Object Category Dataset (Technical report, CNS-TR-2007-001). Pasadena, USA: California Institute of Technology.