Modelling Crowdsourcing Originated Keywords within the Athletics Domain

Zenonas Theodosiou and Nicolas Tsapatsoulis

Dept. of Communication and Internet Studies,
Cyprus University of Technology,
31 Archbishop Kyprianos Str, CY-3036, Limassol
{zenonas.theodosiou,nicolas.tsapatsoulis}@cut.ac.cy

Abstract. Image classification arises as an important phase in the overall process of automatic image annotation and image retrieval. Usually, a set of manually annotated images is used to train supervised systems and classify images into classes. The act of crowdsourcing has largely focused on investigating strategies for reducing the time, cost and effort required for the creation of the annotated data. In this paper we experiment with the efficiency of various classifiers in building visual models for keywords through crowdsourcing with the aid of Weka tool and a variety of low-level features. A total number of 500 manually annotated images related to athletics domain are used to build and test 8 visual models. The experimental results have been examined using the classification accuracy and are very promising showing the ability of the visual models to classify the images into the corresponding classes with the highest average classification accuracy of 74.38% in the purpose of SMO data classifier.

Keywords: Crowdsourcing Annotation, Keyword Modelling, Image Classification

1 Introduction

Image tagging helps search engines to better retrieve desired images in response to text queries. Automatic image annotation has concentrated on the difficulty of relating high-level human interpretations with low-level visual features. The interpretation inconsistency between image descriptors and high-level semantics is known as the semantic gap [1] or the perceptual gap [2]. This is due to the fact that the visual image features extracted from an image cannot be automatically translated reliably into high-level semantics [3]. A manually annotated set of multimedia data is used to train a system for the identification of joint or conditional probability of an annotation occurring together with a certain distribution of multimedia content feature vectors [4]. Different models and machine learning techniques are developed to learn the correlation between image features and textual words from the examples of annotated images and then apply the learned correlation to predict words for unseen images [5].

On the other hand, manual image tagging annotation is an extremely difficult and elaborate task and cannot always be considered as correct due to visual information that always lets the possibility for more individual interpretation and ambiguity [6]. Crowdsourcing [7] has magnetized the interest of several researchers, since it is a very attractive solution to the problem of cheaply and quickly acquiring annotations and has a potential to improve evaluation of information retrieval systems by scaling up relevance assessments and creating test collections with more complete judgments [8]. Amazon Mechanical Turk [9] opened a new way of satisfying the need for large collections of human-annotated data as presented in the recent past [10] by extending the interactivity of crowdsourcing tasks using more comprehensive user interfaces and micro-payment mechanisms.

In this paper we get the advantage of the availability of a large dataset related to the athletics domain created during the FP6 Boemie project [11] and deal with the experimental evaluation of the efficiency of various low-level features and data classifiers in modelling crowdsourcing originated keywords. A set of images was annotated by 15 users using a predefined set of keywords. Images sharing a common keyword are grouped together and used to create the visual model which corresponds to this keyword. Eight different keyword models are created using low-level features and tested with the aid of well known data classifiers. We have used publicly available tools for the computation of the low level features [12], [13] and the model creation (the Weka tool [14]) and classified the images into 8 keyword classes.

This paper is organized as follows: Section 2 gives a detailed description of the method we have followed to create the dataset and build the visual models while Section 3 presents and discusses the experimental results. Finally, conclusions are drawn and further work hints are given in Section 4.

2 Method Overview

This section presents the method we have followed to model the crowdsourced keywords. It consists of 3 main steps: the dataset creation, the feature extraction, and the keyword modelling. The overall procedure is illustrated in Fig. 1.

2.1 Dataset Creation

The crowdsourcing annotation was based on a randomly selected set of 500 images taken from a large dataset created during the BOEMIE project [11]. The dataset was manually annotated by 15 users using the MuLVAT annotation tool [15] with the aid of a structured xml dictionary consists of 33 different keywords. For our experiments we have selected 8 representative keywords and for each keyword, 50 images that were annotated from more than 5 annotators with this keyword were grouped together to create a set of 8 different groups of images (Fig. 2).

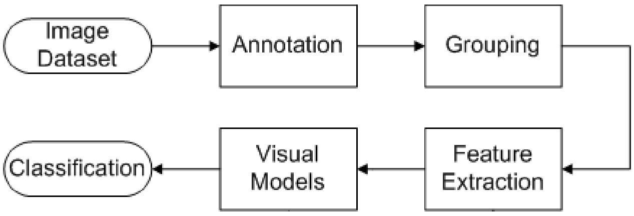

Fig. 1. The flowchart of overall method.

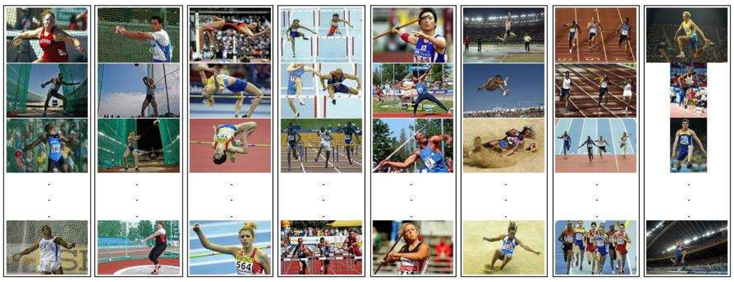

Fig. 2. The set of image groups.

2.2 Feature Extraction

Among the possible low-level features that can be extracted from the image groups, we have chosen the following popular and widely used features:

Histogram of Gradients Features. The HOG features exploit the idea that local object appearance can be described by the distribution of intensity gradients or edge directions. The image is divided into small connected regions, called cells. For each cell, a histogram of gradient directions or edge orientations within this cell is compiled. For the implementation of HOG, each pixel within the cell casts a weighted vote for an orientation-based histogram channel. For the current study we have used the implementation proposed in [13] with the aid of 25 rectangular cells and 9 bins histogram per cell. The 16 histograms with 9 bins were then concatenated to make a 225-dimensional feature vector.

Scale-Invariant Feature Transform features. SIFT transforms image data into scale-invariant coordinates relative to local features and performs a set of features that are not affected by object scaling and rotation. Key points are detected as the maxima of an image pyramid built using difference-of-Gaussians.

The multi-scale approach results in features that are detected across different scales of images. For each detected key-point, a 128 dimensional feature vector is computed describing the gradient orientations around the key-point. The strongest gradient orientation is selected as reference, thus giving rotation invariance to SIFT features. For our experiments each SIFT vector is quantized into a 100-dimensional feature vector using k-means clustering.

MPEG-7 features. MPEG-7 visual descriptors include the color, texture and shape descriptor. A total of 22 different features are included, nine for color, eight for texture and five for shape. The dominant color features include color value, percentage and variance and require especially designed metrics for similarity matching. Furthermore, their length is not known a priori since they are image dependent (for example an image may be composed from a single color whereas others vary in color distribution). The previously mentioned difficulties cannot be easily handled in machine learning schemes, therefore we decided to exclude these features for the current experimentation. The texture browsing features (regularity, direction, scale) have not been included in the description vectors since in the current implementation of the MPEG-7 experimentation model [12] the corresponding descriptor cannot be reliably computed (it is a known bug of the implementation software). The scalable color and shape descriptor features have been also excluded because vary depending on the form of an input object and can not be used for the holistic image description. Among all MPEG-7 descriptors only the Color Layout (CL), Color Structure (CS), Edge Histogram (EH) and Homogenous Texture (HT) descriptors are used in our experiments. The combination of the selected descriptors creates a 186-dimensional feature vector.

2.3 Keyword Modelling

To overcome the multiclass classification problem and facilitate effective and efficient learning, each keyword is treated as a separate binary classification problem. We have followed the one-against-rest approach [16] and we have built a total number of 8 models, one for each keyword. The feature vectors of each keyword class were split into two groups, called the training (80%) and testing (20%) set. Each model is trained and tested between one class and the 7 other classes. The training and testing set for each model contain the feature vectors of the corresponding keyword class and the same number of randomly selected feature vectors of the the rest 7 classes. Keywords models were created using Weka tool [14]. Since statistical methods have their limitations, particularly in relation to distributional assumptions and to the restrictions on data input, we have decided to use artificial intelligence classifiers such as Support Vector Machines (SVM) and Decision Trees (DT).

2.4 Support Vector Machines.

SVM separates the classes with a decision surface that maximizes the margin between the classes. The surface is often called the optimal hyperlane and the data points closer to the hyperplane are called support vectors. For our experiments we decided to use two of the state of the art implementations of the Support Vector Machines (SVMs), the SMO [17] and the LibSVM [18]. They have been reported in several publications as the best performing machine learning algorithms for a variety of classification tasks. The performance of SVM classifiers can vary significantly with variation in parameters of the models. During training we experimented with different parameters and kernels and for each kernel we built models for several combinations of the parameters, with the Pearson VII universal and polynomial kernel performing better than the others for the SMO and LibSVM classifier respectively.

2.5 Decision Trees.

Unlike other classification approaches that use a set of features jointly to perform classification in a single decision step, the decision tree is based on a multistage or hierarchical decision scheme or a tree like structure. The tree is composed of a root node (containing all data), a set of internal nodes (splits) and a set of terminal nodes (leaves). Each node of the decision tree structure makes a binary decision that separates either one class or some of the classes from the remaining classes. The processing is generally carried out by moving down the tree until the leaf node is reached. Turning to the classifiers, Random Forest [19] and Logistics Model Tree (LMT) [20] have been employed to model the keywords.

3 Experimental Results

We used the dataset and keyword modelling process described in the previous Section to examine the performance and effectiveness of the created models: “Discus”, “Hammer”, “High Jump”, “Hurdles”, “Javelin”, “Long Jump”, “Running”, and “Triple Jump”. Fig. 3, 4, 5, 6 show the accuracy of correctly classified instances for all classes using the various data classifiers. The results shown in these figures were examined under three perspectives: First, in terms of the efficiency of the various classifiers in modelling crowdsourced keywords, second in terms of the efficiency of the low-level features to create accurate visual models and third, in terms of the ability of the created models to classify the images into the corresponding classes.

The efficiency of the training algorithms is examined through the effectiveness of the created models, the time required to train the models and the robustness to the variation of learning parameters. The SMO and Random Forest classifiers require by far the lower time and effort to create an effective model, while LMT is the slowest classifier among all. In the case of SMO and Random Forest the learning takes no more than a few seconds for the majority of the keyword models. Furthermore, the fluctuation in classification performance during parameters tuning is significantly lower than that of the LibSVM and LMT. There is a significant difference on the performance of the models created using the individual classifiers. It is evident from Table 1 that SMO is the most reliable classifier with a total average classification accuracy equal to 74.38%. The LibSVM and Random Forest give the same average classification accuracy with the difference that LibSVM performs better for the HOG while the Random Forest performs better for the MPEG-7 features. The most disappointing classifier is the LMT which has the worst average classification accuracy values. The best classification accuracy was occurred in the case of SMO classifier using MPEG-7 features and the worst in the case of LMT classifier using SIFT features.

Concerning the efficiency of the various low-level features, the experimental results indicate that the MPEG-7 features perform better than HOG and SIFT. The classification accuracy obtained using these features is quite good in comparison with the other two and has average classification accuracy values in the range of 71.25%- 81.25%, with the lowest value given by LMT and the highest by SMO classifier respectively. The second more reliable low-level features for modeling keywords are the HOG that can obtain the maximum average classification accuracy value of 72.5% in the purpose of the SMO classifier. The most disappointing classification performance is achieved by the SIFT which can obtain the maximum value of 75% only for “Discus”, “Hammer” and “Hurdles” classes.

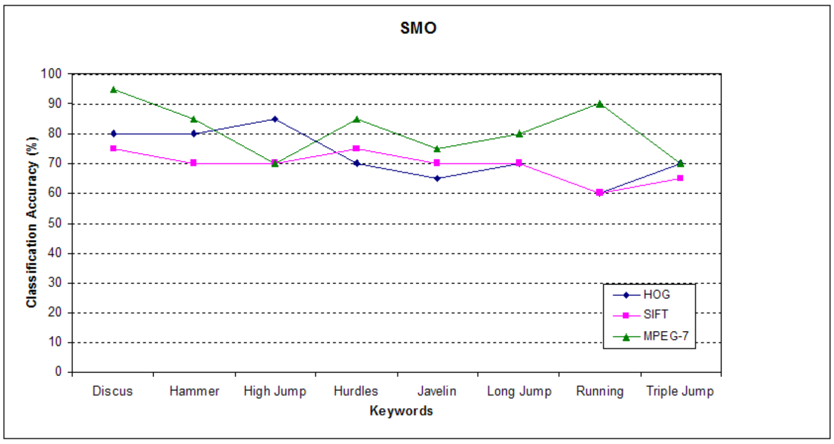

Fig. 3. Accuracy of the correctly classified instances using the SMO data classifier.

Nearly all models are able to classify the images into the corresponding classes with classification accuracy in the range of 55%-95%. The best efficiency is per-

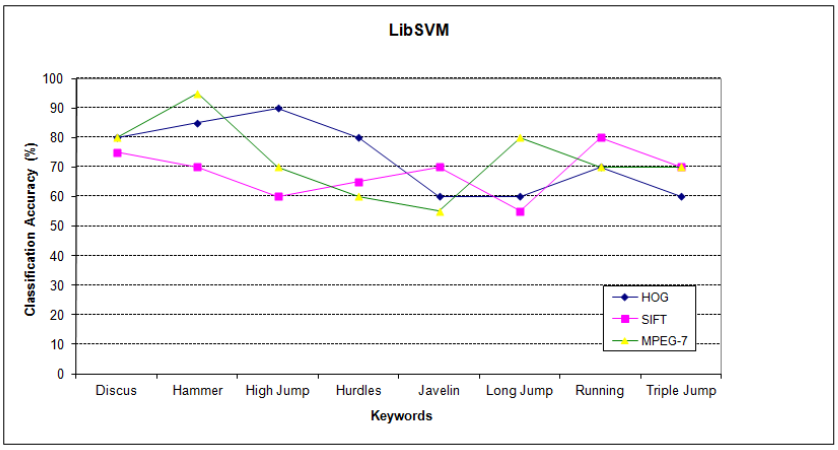

Fig. 4. Accuracy of the correctly classified instances using the LibSVM data classifier.

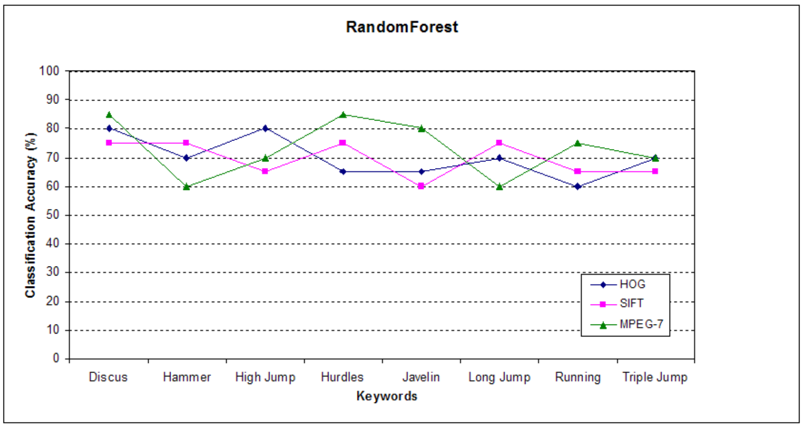

Fig. 5. Accuracy of the correctly classified instances using the Random Forest data classifier.

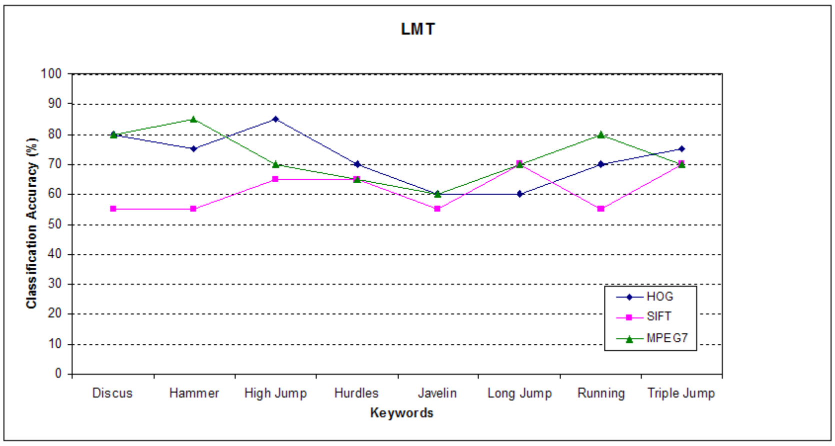

Fig. 6. Accuracy of the correctly classified instances using the LMT data classifier.

ceived when testing models created by images having objects with a well defined shape such as "Discus" and "Hurdles". As a consequence, the worst results are occurred when testing the "Running" and "Triple Jump" because the content of images belong to these keywords has many similarities with the content of images belong to other keywords.

Table 1. Average classification accuracy values (%) for the different classifiers.

Classifier	***HOG***	***SIFT***	***MPEG-7***	***Overall***
SMO	72.5	69.38	81.25	74.38
LibSVM	73.13	68.13	71.25	70.83
Random Forest	70.0	69.38	73.13	70.83
LMT	71.88	61.25	72.5	68.54

4 Conclusions and Future Work

We present an experimental evaluation of modelling crowdsourcing originated keywords within the athletics domain. Specifically, 8 different keywords were modeled using various low-level features and data classifiers. According to our experimental results, nearly all created models can classify the images into the 8 classes with medium to high classification accuracy. Although there is a significant variation on the efficiency of the various classifiers with the SMO having the highest performance, a great improvement can achieved when the MPEG-7 features are used. Our future perspectives involve the evaluation of the proposed method on larger and different datasets as well as the experimentation of additional training algorithms and other classifications schemes. In addition, the efficiency of more low-level features in creation of visual models will be investigated.

Acknowledgements: This work falls under the Cyprus Research Promotion Foundation's Framework Programme for Research, Technological Development and Innovation 2009-2010 (DESMI 2009-2010), co-funded by the Republic of Cyprus and the European Regional Development Fund, and specifically under Grant *ΠΕΝΕΚ*/0609/95.

References

1. Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence. 22(12), 1349–1380, (2000)
2. Jaimes, A., Christel, M., Gilles, S., Sarukkai, R., Ma, W.-Y.: Multimedia information retrieval: what is it, and why isn't anyone using it?. In: Proc. of the 7th ACM SIGMM International Workshop on Multimedia Information Retrieval (MIR '05). 3–8, (2005)
3. Datta, R., Joshi, D., Li, J., Wang, J. Z.: Image retrieval: Ideas, influences, and trends of the new age. In ACM Computing Surveys. 40(2), 1–60, (2008)
4. Athanasakos, K., Stathopoulos, V., Jose, J.: A Framework for Evaluating Automatic Image Annotation Algorithms. Advances in Information Retrieval. LNCS, vol.5993, pp. 217-228 (2010)
5. Zhang, P., Zhang, Z., Li, M., Ma, W. Y., Zhang, H. J.: A probabilistic semantic model for image annotation and multi-modal image retrieval. Multimedia Systems. 12(1), 27–33, (2006)
6. Volker, T., Thom, A., Tahaghoghi, S. M. M.: Modelling human judgment of digital imagery for multimedia retrieval. IEEE Transactions on Multimedia. 9(5), 967–974, (2007)
7. Howe, J.: Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business. Crown Publishing Group. (2008)
8. Kazai, G., Kamps, J., Koolen, M., Milic-Frayling, N.: Crowdsourcing for book search evaluation: Impact of quality on comparative system ranking. In: Proc. of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM Press, New York NY, (2011)
9. Amazon Mechanical Turk - Artificial Artificial Intelligence, `http://www.mturk.com`
10. Eickhoff, C., de Vries, A. P.:How Crowdsourcable is Your Task?. In Proc. of the Workshop on Crowdsourcing for Search and Data Mining. Hong Kong, China, (2011)
11. BOEMIE - Bootstrapping Ontology Evolution with Multimedia Information Extraction, `http://www.boemie.org`
12. MPEG-7 Visual Experimentation Model (XM), Version 10.0, ISO/IEC/JTC1/SC29/WG11, Doc. N4063, (2001)
13. Ludwig, O., Delgado, D., Goncalves, V., Nunes, U.: Trainable Classifier-Fusion Schemes: An Application To Pedestrian Detection. In: Proc. of 12th International IEEE Conference On Intelligent Transportation Systems. 1, 432–437, (2009)

14. Witten, I. H., Frank, E.: Data Mining: Practical machine learning tools and techniques. 2nd Edition, Morgan Kaufmann, San Francisco, (2005)
15. Theodosiou, Z., Kounoudes, A., Tsapatsoulis, N., Milis, M.: MuLVAT: A Video Annotation Tool Based on XML-Dictionaries and Shot Clustering. In: Proc. of the 19th International Conference on Artificial Neural Networks: Part II. 913–922, (2009)
16. Tax, D. M. J., Duin, R .P .W.: Using two-class classifiers for multiclass classification. In: Proc. of 16-th International conference of Pattern Recognition. 124–127, (2002)
17. Platt, J.: Fast Training of Support Vector Machines using Sequential Minimal Optimization. Advances in Kernel Methods-Support Vector Learning, B. Schoelkopf, C. Burges, and A. Smola, eds., MIT Press, (1998)
18. Fan, R.-E., Chen, P.-H., Lin, C.-J.: Working set selection using the second order information for training SVM. Journal of Machine Learning Research. 6, 1889–1918, (2005)
19. Breiman, L.:Random Forests. Machine Learning. 45(1), 5–32, (2001)
20. Landwehr, N., Hall, M., Frank, E.:Logistic Model Trees. Machine Learning. 59(1), 161–205, (2005)