

Cooperative classification of shared images

Claudio Cusano^a and Simone Santini^b

^aUniversità degli Studi di Milano-Bicocca, Viale Sarca 336, 20126 Milano, Italy

^bUniversidad Autónoma de Madrid, C/ Tomas y Valiente 11, 28049 Madrid, Spain

ABSTRACT

We propose a method for the semi-automatic organization of photo albums. The method analyzes how different users organize their own pictures. The goal is to help the user in dividing his pictures into groups characterized by a similar semantic content. The method is semi-automatic: the user starts to assign labels to the pictures and unlabeled pictures are tagged with proposed labels. The user can accept the recommendation or make a correction. We use a suitable feature representation of the images to model the different classes that the users have collected. Then, we look for correspondences between the criteria used by the different users which are integrated using boosting. A quantitative evaluation of the proposed approach is obtained by simulating the amount of user interaction needed to annotate the albums of a set of members of the flickr[®] photo-sharing community.

Keywords: Personal photography, automatic image annotation, content-based image analysis, social image retrieval

1. INTRODUCTION

One of the most remarkable social and technical by-products of the diffusion of the internet is the emergence of communities connected not by physical proximity but by common interests. While these utopic (in the original etymology of *οὐ – τοπος*: no place) communities have existed for a long time, from the medieval monastic orders to the scientific community to which this conference owes its existence, the internet has given them a stronger cohesion by providing the technical instruments for frequent communication. Socially, one fairly evident consequence of the internet has been, in a sense, the trivialization of the common interest around which communities gather. In other times, the survival of an utopic community required a considerable effort, and could be justified only by a common interest that its participants regarded as primary. With the advent of the internet, communities are created easily, and every one of us can be at the same time a member of many of them, often representing interests that we regard as superficial and of little importance. Technically — this is the aspect of interest here — these communities have created a great interest in peer-to-peer systems and in *social filtering*, a technical instrument to use the collective wisdom, so to speak, of the community to the advantage of each one of its members.

This paper will consider a community that is not held by very strong ties but to which many of us, at some time or another, belong: that of amateur photographers. We will consider a community of people interested in non-commercial photography who place their photographs in a suitable web server (flickr[®], or similar services) where it can be shared by a community of similarly interested people. One of the most common activities in which people engage when organizing pictures is that of classification: the pictures in the camera will be divided into thematically organized folders. The criteria that preside this organization are, of course, highly personal: in this case, what's good for the goose is not necessarily good for the gander. The same vacation photos that a person will divide in “Rhodos” and “Santorini” will be divided by someone else into “family”, “other people” and “places” or into “beach”, “hotel” and “excursion”, or in any other organization.

In this paper, we develop a system that will assist people in this classification task. Faithful to the principles of community based computing, we try to use the collective wisdom of the community in order to suggest to one of its member possible ways of classification. Briefly, when a person (we call this person the apprentice) start putting photos into carpets, the system will look at other users of the community and at the classifications they made. Members who agree with the classification made by the apprentice (except the wizards) will be used as classifiers to propose a classification of the apprentice's unclassified images.

Claudio Cusano: claudio.cusano@disco.unimib.it, Simone Santini: simone.santini@uam.es

1.1 Related Work

A number of commercial products is available for the management and organization of personal photo collections. In spite of being convenient and user friendly, these products still rely largely on manual annotation for browsing and retrieval. To overcome this limitation several automatic or semi-automatic content-based approaches have been proposed. A prototype system for home photo management and processing has been implemented by Sun et al.¹ Together with traditional tools, they included a function to automatically group photos by time, visual similarity, image class (indoor, outdoor, city, landscape), or number of faces (as identified by a suitable detector). Another system for managing family photos has been developed by Wenyin et al.² The system allows the categorization of photos into some predefined classes. A semi-automatic annotation tool, based on retrieval by similarity, is also provided. When the user imports some new images, the system searches for visually similar archived images. The keywords with higher frequencies in these images are used to annotate the new images. Keywords have to be confirmed or rejected in a successive retrieval-feedback process. Mulhem and Lim proposed the use of temporal events for organizing and representing home photos using structured document formalism.³ Retrieval and browsing of photos are based on both temporal context and image content, represented by the occurrence of 26 classes of visual keywords. Shevade and Sundaram presented an annotation paradigm that attempts to propagate semantic by using WordNet and low-level features extracted from the images.⁴ As the user begins to annotate images, the system creates positive and negative example sets for the associated WordNet meanings. These are then propagated to the entire database, using low-level features and WordNet distances. The system then determines the image that is least likely to have been annotated correctly and presents the image to the user for relevance feedback.

A common approach for automatic organization of photo albums consists in the application of clustering techniques to group images into visually similar sets. Manual post-processing is usually required to modify the clusters in order to match user's categorization. Information about time is often used to improve clustering by segmenting the album into events. Platt proposed a method for clustering personal images taking into account timing and visual information.⁵ Loui and Savakis described an event-clustering algorithm which automatically segments pictures into events and sub-events, based on date/time metadata information, as well as color content of the pictures.⁶ Li et al. exploited time stamps and image content to partition related images in photo albums.⁷ Key photos are selected to represent a partition based on content analysis and then collated to generate a summary. A semi-automatic technique has been presented by Jaimes et al.⁸ They used the concept of Recurrent Visual Semantics (the repetitive appearance of visually similar elements) as the basic organizing principle. They proposed a sequence-weighted clustering technique which is used to provide the user with a hierarchical organization of the contents of individual rolls of film. As a last step, the user interactively modifies the clusters to create digital albums.

Since people identity is often the most relevant information for the user, it is not surprising that several approaches have been proposed for the annotation of faces in family albums. Das and Loui used age/gender classification and face similarity to provide the user with the option of selecting image groups based on the people present in them.⁹

Another framework for semi-automatic face annotation has been proposed by Chen et al.¹⁰ In addition to the traditional face recognition features they used similarity search and relevance feedback on a set of color and texture features. Zhang et al. have reformulated the face annotation from a pure recognition problem to a problem of similar face search and annotation propagation.¹¹ Their solution integrates content-based image retrieval and face recognition algorithms in a Bayesian framework.

The idea of exploiting user correlation in photo sharing communities has been investigated by Li et al.¹² They proposed a method for inferring the relevance of user-defined tags by exploiting the idea that if different persons label visually similar images using the same tags, these tags are likely to reflect objective aspects of visual content. Each tag of an image accumulates its relevance score by receiving votes by neighbors (i.e. visually similar images) labeled with the same tag.

2. METHOD

We propose a method for the semi-automatic organization of personal photos. The method is content-based, that is, only pictorial information is considered. The method is applicable to non-visual information such as

keywords and annotations. In spite of the importance that these annotations may have for the determination of the semantics of images, we have decided to limit our considerations to visual information on methodological grounds, since this will give us a more immediate way of assessing the merits of the method in comparison with simple similarity search.

The goal is to help the user in classifying pictures dividing them into groups characterized by similar semantics. The number and the definition of these groups are completely left to the user. At the beginning all pictures are unlabeled, and the user starts to assign labels to them. After each assignment, the unlabeled pictures are tagged with proposed labels. The user can accept the recommendation or make a correction. In either case the correct label is assigned to the image and the proposed labels are recomputed. Unlabeled pictures are displayed sorted by decreasing confidence on the correctness of the suggestion, but the order in which the user process the images is not restricted. Provided a reasonable user interface is available, the labels proposed by the method can be confirmed very quickly, allowing for a rapid and convenient organization of the album.

At the beginning, we don't have any information on the criteria that the user is going to apply in partitioning his pictures. However, a huge library of possible criteria is available in photo-sharing communities. The users of these services are allowed to group their own images into sets and we can assume that these sets contain pictures with some characteristic in common. For instance, sets may contain pictures taken in the same location, or portraying a similar subject.

Our idea is to exploit the knowledge encoded in how a group of users (*wizards*, in the following) have partitioned their images, in order to help organize the pictures of a different user (the *apprentice*). The method is conceptually articulated in two parts. First, we use a suitable feature representation of the images of the wizards to model the different classes that they have collected, second, we look for correspondences between the (visual) criteria used in the wizards' classes and those that the apprentice is creating in order to provide advice. In other, somewhat oversimplistic words; if we notice that one of the classes that the apprentice is creating appears to be organized using criteria similar to those used in one or more wizard's classes, we use the wizards' classes as representative, and the unlabeled apprentice images that are similar to those of the wizard class are given the label of that class.

Consider a wizard, who partitioned his pictures into the C categories $\{\omega_1, \dots, \omega_C\} = \Omega$. These labeled pictures are used as a training set to train a classifier that consists of a classification function $g : X \rightarrow \Omega$ from the feature space X into the set of user defined classes. If the partition of the wizard exhibits regularities (in terms of visual content) that may be exploited by the classification framework, then g may be used to characterize the pictures of the apprentice as well. Of course, it is possible that the apprentice would like to organize his pictures into different categories. However, people tend to be predictable, and it is not at all uncommon that the sets defined by two different users present some correlation that can be exploited. To do so, we define a mapping $\pi : \Omega \rightarrow Y$ between the classes defined by the wizard and the apprentice (where $Y = \{y_1, \dots, y_k\}$ denotes the set of apprentice's labels). We allow a non-uniform relevance of the apprentice's images in defining the correlation with the wizard's classes. Such a relevance can be specified by a function w that assigns a positive weight to the images. Weighting will play an important role in the integration of the predictions based on different wizards, as described in Section 2.2. Let $Q(\omega_i, y_j)$ be the set of images to which the apprentice has assigned the label y_j , and that, according to g , belong to ω_i ; then π is defined as follows:

$$\pi(\omega) = \arg \max_{y \in Y} \sum_{x \in Q(\omega, y)} w(x), \quad \omega \in \Omega, \quad (1)$$

where a label is arbitrarily chosen when the same maximum is obtained for more than one class. That is, π maps a class ω of the wizard into the class of the apprentice that maximizes the cumulative weight of the images that g maps back into ω . If no apprentice image belong ω we define $\pi(\omega)$ to be the class of maximal total weight.

If we interpret w has a misclassification cost, our definition of π denotes the mapping which, when combined with g , minimizes the total misclassification error on the images of the apprentice:

$$\min_{\pi: \Omega \rightarrow Y} \sum_{x, y} w(x) (1 - \chi_{\{y\}}(\pi(g(x)))) , \quad (2)$$

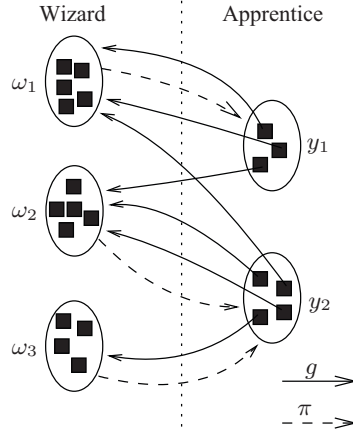


Figure 1. Example of definition of the mapping π between a wizard and the apprentice, assuming uniform weights. Since g maps into ω_1 two images of class y_1 and only one image of class y_2 , we have that $\pi(\omega_1) = y_1$. Similarly, $\pi(\omega_2) = \pi(\omega_3) = y_2$.

where the summation is taken over the pairs (x, y) of images of the apprentice with the corresponding labels, and where χ denotes the indicator function ($\chi_A(x) = 1$ if $x \in A$, 0 otherwise). Figure 1 depicts and summarizes how π is defined.

The composition $h = \pi \circ g$ directly classifies elements of X into Y . In addition to embedding the correlation between the wizard and the apprentice, the design of h enjoys a useful property: the part defined by g is independent of the apprentice, so that it can be computed off-line allowing for the adoption of complex (and hopefully accurate) machine learning models such as SVMs, neural networks, and the like; the part defined by π , instead, can be worked out very quickly since its computation is linear in the number of the images labeled by the apprentice and does not depend on the whole album of the wizard, but only on its partial representation provided by g . In this work, g is a k -nearest neighbor (KNN) classifier.

2.1 Image Description

Since we do not know the classes that the users will define, we selected a set of four features that give a fairly general description of the images: spatial color moments, color histogram, edge direction histogram, and a bag of features histogram.

Spatial color distribution is one of the most widely used feature in image content analysis and categorization.¹³ We divided each image into 7×7 blocks and computed the mean and standard deviation of the values of the color channels of the pixels in each block. We used the LUV color space, since moments in this color space are more discriminant than in other spaces, at least for image retrieval.¹⁴ This feature includes 294 components (six for each block).

Color moments are less useful when the blocks contain heterogeneous color regions. Therefore, a global color histogram has been selected as a second color feature. The RGB color space has been subdivided in 64 bins by a uniform quantization of each component in four ranges.

To describe the most salient edges we used a 8 bin edge direction histogram:¹⁵ the gradient of the luminance image is computed using Gaussian derivative filters tuned to retain only the major edges. Only the points for which the magnitude of the gradient exceeds a set threshold contribute to the histogram. The image is subdivided into 5×5 blocks, and a histogram for each block is computed (for a total of 200 components).

The fourth feature we considered is based on a bag-of-features representation.^{16–18} The basic idea is to select a collections of representative patches of the image, compute a visual descriptor for each patch, and use the resulting distribution of descriptors to characterize the whole image. In our work, the patches are the areas surrounding distinctive key-points and are described using the Scale Invariant Feature Transform (SIFT) which is invariant to image scale and rotation, and has been shown to be robust across a substantial range of

affine distortion, change in 3D viewpoint, addition of noise, and change in illumination.¹⁹ More in detail, we adopted the implementation described in²⁰ for both key-points detection and description. The SIFT descriptors extracted from an image are then quantized into “visual words”, which are defined by clustering a large number of descriptors extracted from a set of training images.²¹ The final feature vector is the normalized histogram of the occurrences of the visual words in the image.

2.2 Selecting and Combining Users

Since, there is no guarantee that the classes chosen by two different users have a sufficient correlation to make our approach useful we need several wizards and a method for the selection of those who may help the apprentice organize his pictures. The same argument may be applied to the features as well. Consequently, we treated the features separately instead of merging them into a single feature vector: given a set of pictures labeled by the apprentice, each wizard defines four different classifiers h , one for each feature considered. These classifiers need to be combined into a single classification function that will be then applied to the pictures that the apprentice has not yet labeled.

To combine the classifiers defined by the wizards we apply the multiclass variation of the Adaboost algorithm proposed by Zhu et al.²² In particular, we used the variation called Stagewise Additive Modeling using a Multi-class Exponential loss function (SAMME). Boosting defines the classifier as a weighted combination of weak classifiers which are trained on a training set of image/label pairs

Briefly, given a set of image/label pairs, the algorithm selects the best classifier and assign to it a coefficient. For each iteration the classifier is chosen by a weak learner. The weak learner we defined takes into account all the wizards and all the features. For each wizard u and each of the four features f , a KNN classifier $g_{u,f}$ has been previously trained. Given the weighted training sample, the corresponding mapping functions $\pi_{u,f}$ are computed according to (1); this defines the candidate classifiers $h_{u,f} = \pi_{u,f} \circ g_{u,f}$. The performance of each candidate is evaluated on the weighted training set and the best one is selected. The boosting procedure terminates after a set number T of iterations.

Given an image to be labeled, a score is computed for each class:

$$s_y(x) = \sum_{t=1}^T \alpha^{(t)} \chi_{\{y\}}(\bar{h}^{(t)}(x)), \quad y \in Y, \quad (3)$$

where $\bar{h}^{(t)}$ is the classifier selected at iteration t , and $\alpha^{(t)}$ is the corresponding weight. The combined classifier H is finally defined as the function which selects the class corresponding to the highest score:

$$H(x) = \arg \max_{y \in Y} s_y(x). \quad (4)$$

The combined classifier can be then applied to unlabeled pictures. According to,²² the a posteriori probabilities $P(y|x)$ may be estimated as:

$$P(y|x) = \frac{\exp \frac{s_y(x)}{k-1}}{\sum_{y' \in Y} \exp \frac{s_{y'}(x)}{k-1}}. \quad (5)$$

We used the difference between the two highest estimated probabilities as a measure of the confidence of the combined classifier. Unlabeled pictures can then be presented to the user sorted by decreasing confidence.

It should be noted that the output of the classifiers $g_{u,f}$ can be precomputed for all the images of the apprentice. Using the settings described in Section 3, the whole procedure is fast enough, on a modern personal computer, for real time execution.

In addition to exploiting the information provided by the wizards, we also considered a set of classifiers based on the contents of the apprentice’s pictures. They are four KNN classifiers, one for each feature. They are trained on the pictures already labeled and applied to the unlabeled ones. These additional classifiers are included in the boosting procedure as an alternative to the classifiers derived from the wizards.

The four KNN classifiers are also used as baseline classifiers to evaluate how much our method improves the accuracy in predicting classes with respect to a more traditional approach.

Table 1. Summary of the annotation performed by the 20 volunteers. For each album are reported the number of pictures and the names given to the classes into which the images have been divided.

Album	Size	Classes
1	328	animals, artefacts, outdoor, vegetables
2	261	boat, city, nature, people
3	182	close-ups&details, landscapes, railways, portraits&people, sunsets
4	251	buildings, flora&fauna, musicians, people, things
5	177	animals, aquatic-landscape, objects, people
6	188	animals, buildings, details, landscape, people
7	151	arts, city, hdr
8	182	buildings, hockey, macro
9	140	bodies, environments, faces
10	227	animals, beach, food, objects, people
11	371	animals, sea, sunset, vegetation
12	168	animals, flowers, horse racing, rugby
13	170	animals, concert, conference, race
14	209	aquatic, artistic, landscapes, close-ups
15	146	beach, calendar, night, underwater
16	134	animals, family, landscapes
17	158	animals, cold-landscapes, nature-closeups, people, warm-landscapes
18	156	buildings, landscape, nature
19	102	leaves&flowers, men-made, panorama, pets, trees
20	234	microcosm, panorama, tourism

3. EXPERIMENTAL RESULTS

To test our method we downloaded from flickr[®] the images of 20 users. Each user was chosen as follows: i) a “random” keyword is chosen and passed to the flickr[®] search engine; ii) among the authors of the pictures in the result of the search, the first one who organized his pictures into 3 to 10 sets is selected. In order to avoid excessive variability in the size of users’ albums, sets containing less than 10 pictures are ignored and sets containing more than 100 pictures are sub-sampled in such a way that only 100 random images are downloaded. Duplicates have been removed from the albums. The final size of users’ albums ranges from 102 to 371, for a total of 3933 pictures.

Unfortunately, some of the selected users did not organized the pictures by content: there were albums organized by time periods, by aesthetic judgments, and so on. Since, our system is not designed to take into account this kind of categorizations, we decided to reorganize the albums by content. To do so, we assigned each album to a different volunteer, and we asked him to label the pictures by content. The volunteers received simple directions: each class must contain at least 15 pictures and its definition must be based on visual information only. The volunteers were allowed to ignore pictures to which they were not able to assign a class (which usually happened when the obvious class would have contained less than 15 images). The ignored pictures were removed from the album for the rest of the experimentation. Table 1 reports the classes defined by the volunteers for the 20 albums considered. Thirteen volunteers decided to include the “ignored” class, and a total of 193 pictures have been ignored (4.3% of the whole dataset).

To quantitatively evaluate the performance of the proposed method we implemented a simulation of user interaction.²³ This approach effectively allows to evaluate objectively the methodology without taking into account the design and usability of the user interface. The simulation corresponds to the following process:

1. at the beginning all pictures are unlabeled;
2. a random picture is selected and annotated with the correct class;
3. until the whole album is annotated:

Table 2. Percentage of errors obtained by simulating user interaction on the 20 albums considered. The results are averaged over 100 simulations. For each album, the best performance is reported in bold. Standard deviations are reported in brackets.

Album	KNN classifier	Proposed method
1	30.4% (1.5)	27.9% (0.9)
2	30.3% (1.3)	26.6% (1.8)
3	51.3% (2.1)	45.1% (2.1)
4	55.5% (2.0)	54.0% (1.8)
5	54.6% (2.4)	54.2% (2.2)
6	48.0% (1.9)	46.5% (1.9)
7	24.7% (1.0)	27.1% (1.9)
8	12.3% (1.4)	13.5% (1.2)
9	43.5% (1.9)	45.4% (2.1)
10	31.4% (1.4)	32.1% (1.5)
11	27.1% (1.1)	24.4% (1.3)
12	20.7% (1.3)	23.9% (1.7)
13	17.6% (1.2)	16.2% (1.0)
14	52.2% (1.9)	51.2% (1.7)
15	4.6% (1.4)	4.5% (0.7)
16	32.6% (2.1)	27.3% (2.1)
17	35.2% (2.3)	34.2% (2.0)
18	36.2% (2.1)	32.9% (2.1)
19	57.0% (3.3)	60.0% (3.6)
20	21.6% (1.4)	18.8% (1.2)

- (a) the system is trained on already labeled pictures;
- (b) unlabeled pictures are classified;
- (c) the picture with the highest classification confidence is selected and annotated with the correct class (i.e. the class assigned to that picture by the volunteer).

As a measure of performance, we considered the fraction of cases in which the class proposed by the system for the picture selected in step 3c agrees with the annotation performed by the volunteer.

The simulation has been executed for the 20 albums considered. Each time an album corresponds to the apprentice and the other 19 correspond to the wizards. Since the final outcome may be heavily influenced by the random choice of the first picture, we repeated the simulation 100 times for each album.

We compared our wizard-based method with a KNN-based classification. The parameters of the method have been tuned on the basis of the outcome of preliminary tests conducted on ten additional albums annotated by the authors. The number of neighbors considered by the wizards and by the KNN classifiers has been set to 21 and 5, respectively; the number of boosting iterations has been set to 50.

Table 2 shows the average percentage of classification errors obtained on the 20 albums. For both the methods considered, there is a high variability in performance on the 20 albums, ranging from about 4% to 60% of misclassifications. Albums 8, 13, and 15 have been organized into classes which are easy to discriminate and obtained the lowest classification errors. It is interesting to note that these three albums have been the easiest to annotate manually as well (according to informal volunteers' feedback). The opposite happens for the albums to which correspond the highest classification errors (albums 4, 5 and 19).

The proposed method outperformed the KNN classifier on 14/20 albums. In some cases the improvement is barely noticeable, but in other cases it is significant, with a peak of more than 6% of decrease of misclassifications for album 3.

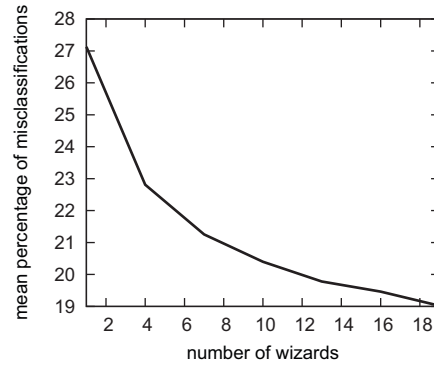


Figure 2. Mean percentage of misclassifications obtained on the 20 albums, varying the number of wizards considered.

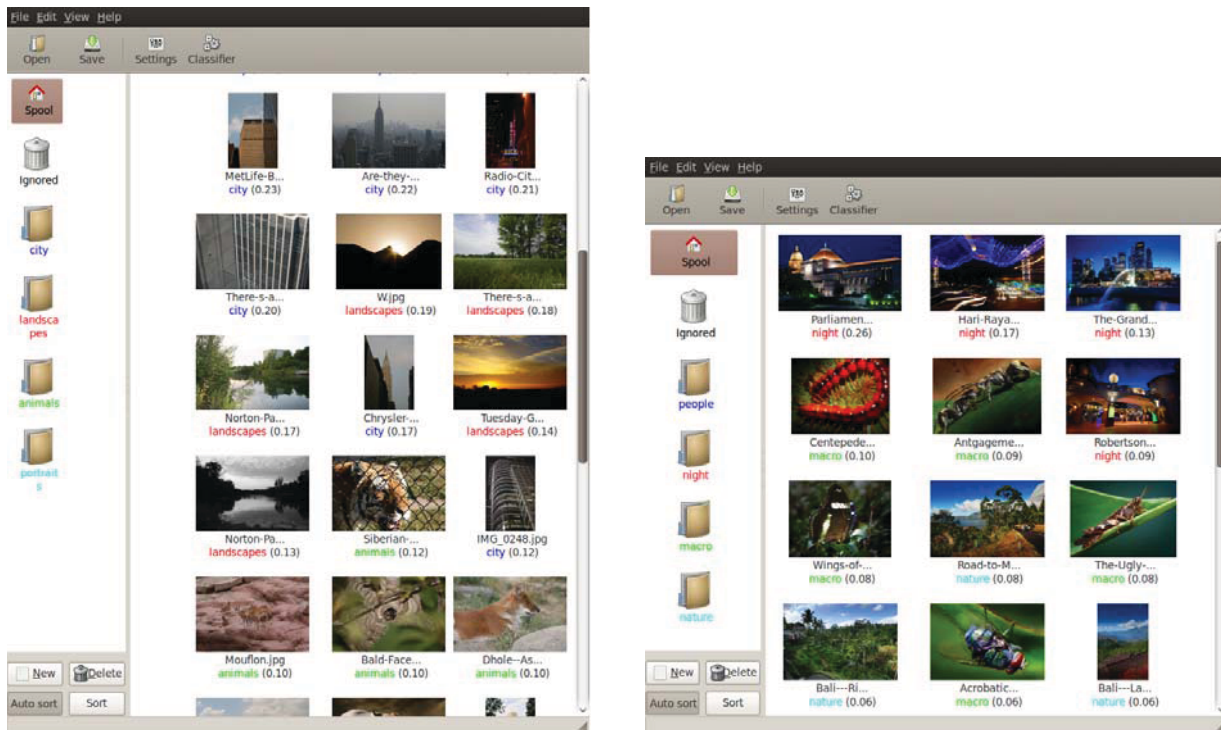


Figure 3. Screenshots of the image annotation tool. Each unlabeled picture is annotated with a class proposed by the system. Proposals are chosen among the classes defined by the user (represented by the folders on the left side). The confidence scores about the proposals are used to sort the images.

To verify the influence of the number of the wizards on classification accuracy, we repeated the simulations (using a wizards-only variant, without the baseline KNN), sampling each time a different pool of wizards. For each album, simulations are performed sampling 1, 4, 7, 10, 13, 16, and 19 wizards, and each simulation has been repeated 50 times (a different pool of wizard is randomly sampled each time). The plots in Figure 2 report the results obtained in terms of average percentage of misclassification errors. As expected, the error rate decreases as the number of wizards increases. The plots suggest that better performance may be obtained by considering more wizards.

Figure 3 shows two screenshots of a prototypical system which implements the proposed method.

4. CONCLUSIONS

In this paper, we described a content-based method for the semi-automatic organization of personal photo collections. The method exploits the correlations, in terms of visual content, between the pictures of different users considering, in particular, how they organized their own pictures. Combining this approach with a KNN classifier we obtained better results (measured on the pictures of 20 flickr[®] users) with respect to a traditional classification by similarity approach.

We believe that the performance of the method could be improved in several ways. For instance, the method could benefit from the adoption of more powerful machine learning techniques, such as Support Vector Machines. Since the training of wizard-based classifiers is performed off-line, this modification would not prevent real-time interaction. According to experimental results, performance could also be improved by considering a larger pool of wizards. In this work, we considered the apprentice and the wizards as clearly different characters. We plan to extend our approach to actual photo-sharing communities, where each user would be apprentice and wizard at the same time. However, in order to scale up to millions of wizards (the size of the user base of major photo-sharing websites) a method should be designed for filtering only the wizards that are likely to provide good advices. Moreover, we are considering to exploit additional sources of information such as keywords, annotations, and camera metadata.

Finally, we are investigating similar approaches, based on the correlation between users, for other image-related tasks such as browsing and retrieval.

REFERENCES

- [1] Sun, Y., Zhang, H., Zhang, L., and Li, M., “Myphotos: a system for home photo management and processing,” in [*Proceedings of the Tenth ACM International Conference on Multimedia*], 81–82 (2002).
- [2] Wenyin, L., Sun, Y., and Zhang, H., “Mialbum — a system for home photo management using the semi-automatic image annotation approach,” in [*Proceedings of the Eighth ACM International Conference on Multimedia*], 479–480 (2000).
- [3] Mulhem, P. and Lim, J., “Home photo retrieval: Time matters,” in [*Proceedings of the International Conference on Image and Video Retrieval*], 308–317 (2003).
- [4] Shevade, B. and Sundaram, H., “Vidya: an experiential annotation system,” in [*Proceedings of the ACM SIGMM Workshop on Experiential Telepresence*], 91–98 (2003).
- [5] Platt, J., “Autoalbum: clustering digital photographs using probabilistic model merging,” in [*Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries*], 96–100 (2000).
- [6] Loui, A. and Savakis, A., “Automated event clustering and quality screening of consumer pictures for digital albuming,” *IEEE Transactions on Multimedia* **5**(3), 390–402 (2003).
- [7] Li, J., Lim, J., and Tian, Q., “Automatic summarization for personal digital photos,” in [*Proceedings of the Fourth International Conference on Information, Communications and Signal Processing*], **3**, 1536–1540 (2003).
- [8] Jaimes, A., Benitez, A., Chang, S.-F., and Loui, A., “Discovering recurrent visual semantics in consumer photographs,” in [*Proceedings of the International Conference on Image Processing*], **3**, 528–531 (2000).
- [9] Das, M. and Loui, A., “Automatic face-based image grouping for albuming,” in [*Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*], **4**, 3726–3731 (2003).
- [10] Chen, L. and Hu, B., “Face annotation for family photo album management,” *International Journal of Image and Graphics* **3**, 1–14 (2003).
- [11] Zhang, L., Chen, L., Li, M., and Zhang, H., “Automated annotation of human faces in family albums,” in [*Proceedings of the Eleventh ACM International Conference on Multimedia*], 355–358 (2003).
- [12] Li, X., Snoek, C., and Worring, M., “Learning tag relevance by neighbor voting for social image retrieval,” in [*Proceeding of The first ACM International Conference on Multimedia Information Retrieval*], 180–187 (2008).
- [13] Vailaya, A., Figueiredo, M., Jain, A., and Zhang, H.-J., “Image classification for content-based indexing,” *IEEE Transactions on Image Processing* **10**, 117–130 (Jan 2001).

- [14] Furht, B., [*Handbook on Multimedia Computing*], ch. Content-based image indexing and retrieval, CRC Press, Inc. (1998).
- [15] Vailaya, A., Jain, A., and Zhang, H. J., “On image classification: city images vs. landscapes,” *Pattern Recognition* **31**(12), 1921–1935 (1998).
- [16] Zhang, J., Marszalek, M., Lazebnik, S., and Schmid, C., “Local features and kernels for classification of texture and object categories: A comprehensive study,” *International Journal of Computer Vision* **73**(2), 213–238 (2007).
- [17] Wallraven, C., Caputo, B., and Graf, A., “Recognition with local features: the kernel recipe,” in [*Proceedings of the Ninth IEEE International Conference on Computer Vision*], **1**, 257–264 (2003).
- [18] Grauman, K. and Darrell, T., “The pyramid match kernel: discriminative classification with sets of image features,” in [*Proceedings of the Tenth IEEE International Conference on Computer Vision*], **2**, 1458–1465 (2005).
- [19] Lowe, D. G., “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision* **60**(2), 91–110 (2004).
- [20] Vedaldi, A., “Sift++ a lightweight c++ implementation of sift.” <http://vision.ucla.edu/~vedaldi/code/siftpp/siftpp.html>.
- [21] Nister, D. and Stewenius, H., “Scalable recognition with a vocabulary tree,” in [*Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*], **2**, 2161–2168 (2006).
- [22] Zhu, J., Rosset, S., Zou, H., and Hastie, T., “Multiclass adaboost,” tech. rep., Stanford University (2005). Available at <http://www-stat.stanford.edu/~hastie/Papers/samme.pdf>.
- [23] Ivory, M. Y. and Hearst, M. A., “The state of the art in automating usability evaluation of user interfaces,” *ACM Computing Surveys* **33**(4), 470–516 (2001).