Two-stage Automatic Image Annotation Based on Latent Semantic Scene Classification

Hongwei Ge[1], Kai Zhang[1], Yaqing Hou[1], Chao Yu[1], Mingde Zhao[2], Zhen Wang[1], Liang Sun[1]
[1]*College of Computer Science and Technology, Dalian University of Technology,* Dalian, China
[2]*School of Computer Science, McGill University,* Montréal, Canada
hwge@dlut.edu.cn

***Abstract*—The rapid growth of multimedia content makes existing automatic image annotation techniques difficult to satisfy the demands of real-world applications. In this paper, we propose a two-stage automatic image annotation algorithm (TAIA) based on latent semantic scene classification. In the offline training phase, the hidden connectivity of labels is firstly excavated by a directed-weighed graph based on label co-occurrence relation matrix, and then the latent scene categories are detected among the labels by using nonnegative matrix factorization. Further, we propose a multi-view extreme learning machine (MELM) to learn the probability that the multiple visual feature maps to the semantic scenes. In the online annotation phase, the image to be annotated is fed to the scene classifier MELM to identify its relevant scenes. Then k-nearest neighbor based annotator is conducted on the relevant scenes to predict labels for the unannotated images. The TAIA is formulated in such a framework so that the relationship between labels and semantic scenes is fully considered, and the hard classification problem is solved. The experimental results on multiple datasets have demonstrated that the proposed framework TAIA is both effective and efficient.**

I. Introduction

Automatic image annotation (AIA) has been one of the hot research topics in the field of computer vision. It is not only beneficial for information retrieval, but also attractive for information management, since it helps converting the conversion from content-based searching to text-based searching [1]–[4]. AIA aims to learn the association from the visual features to the predefined concepts of labels, which are the semantic concepts for the underlying image information.

The existing approaches for AIA can be categorized into two classes: the learning-based methods and the retrieval-based methods. The learning-based methods include the generative models, the discriminative models and the deep learning models. The generative models consider images as instances which are sampled from a specific statistical distribution and try to maximize the generative likelihood of image features and labels [5]. The shortcomings of the generative models are unguaranteed optimization of the prediction and the high computing cost caused by the complex algorithms. The discriminative models generally treat the AIA problem as a multi-class classification problem and have shown promising performance with higher accuracy [4], [6]–[8]. However, since the label only reflects local semantic concepts, the fact that the image-level features are used in the process of annotation makes the classifier unable to learn the pattern between image features and labels well. Moreover, the bad extendability and high training cost remain bottlenecks of the algorithm [1]. The deep learning models generally use deep neural networks to obtain robust visual features or exploit the side information [9], [10]. Wu *et al.* employed global CNN-based features to represent images and attempted to use a limited number of tags to cover as much semantic image information as possible [9]. Although the deep learning models improve the performance of AIA significantly, the problems of how to improve efficiency and how to control the training process still remain unanswered [11].

The retrieval-based methods try to utilize the similarities of visual features between the new images and the images in the labeled database. The essence of the methods is to provide the candidate list of labels for the query image directly based on the labeled images with complete and effective label information. Generally, this kind of methods partitions the existing image set into clusters to utilize the joint distribution of the image labels with respect to the visual features [12]. Researches on this method are dedicated to enhance the traditional KNN on its two major problems: the ignorance for the semantic connectivity and the calculation burden for the comparisons when applying to real-world problems.

Recently, the community detection techniques that take into account the relation of labels and communities have shown promising results [13], [14]. Bracamonte *et al.* used community detection for clustering labels of images, with each community representing a collection of interconnected concepts [13]. Maihami *et al.* retrieved a cluster of neighbor images for the query image, and labeled it based on community detection techniques within the cluster [14]. To some extent, these approaches solve the problems of huge labels in real worlds and inappropriate mapping between labels and image-level features. There are still some limitations in these methods. First, the mapping from labels to communities is regarded as hard classification problem, that is, one label only belongs to one community, which violates the one-to-many relationship between labels and communities. Second, the problems of how to use multi-view features to train community classifiers effectively and how to improve the efficiency of the algorithm still remain unanswered.

The existing drawbacks in AIA motivate us to find better models for interpreting the relationship between labels and semantic scenes, enhancing the flexibility of label classi-

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

fication for overlapped scene partition and alleviating the training costs. In this paper, we propose a two-stage automatic image annotation framework (TAIA) based on semantic scene classification. The TAIA aims to excavate the relationship between labels and latent semantic scenes and mitigate the problem of hard classification of labels. The contributions of this paper are as follows:

1) A non-negative matrix factorization (NMF) based scene detection process is proposed to excavate the interactive patterns of the labels corresponding to the latent semantic scenes in the images. The innovative idea is that the detected semantic scenes are not necessarily disjoint, i.e., the same label may belong to different scenes. The soft classification of labels is achieved through the mapping probability of labels to different scenes.
2) A multi-view extreme learning machine (MELM) is designed, which decreases the training costs while ensuring competitive accuracy rate. By learning a mapping from visual features to semantic scenes, a new image will be linked to several relevant scenes.

II. Proposed Method

The proposed framework TAIA consists of two major phases, i.e., the offline training phase and the online annotation phase. The framework is presented in Fig. 1.

The training phase includes three major processes: semantic scene detection, sample mapping and scene classifier learning. In the process of semantic scene detection, we firstly build a relation matrix R to represent the co-occurrence of labels. Then a NMF based operator works on the matrix for finding probabilistic mapping from labels to semantic scenes. In the process of sample mapping, each sample is assigned to latent scenes using a label based probabilistic mapping. In the process of scene classifier learning, we train a MELM classifier to learn the mappings from multi-view visual features to image scenes.

In the annotation phase, the MELM classifier provides the probabilities for the query image belonging to semantic scenes. We take the training samples in two most related semantic scenes as the related image subset, with which a specialized KNN method will be conducted.

A. TAIA: Offline Training Phase

The offline training phase of TAIA contains three major processes, which will be explained in details in the following parts of this subsection:

1) Semantic Scene Detection: detecting semantic scenes of labels using non-negative matrix factorization.
2) Sample Mapping: mapping the images from the training set to overlapping scenes with reference to the detected semantic scenes.
3) Scene Classifier Learning: learning the mappings from multi-view visual features to image scenes using the proposed multi-view extreme learning machine.

1) Semantic Scene Detection: We aim to find the latent patterns of the labels and create semantic scenes of labels that contain the mutual underlying semantic information. Actually, we consider the scene detection as community detection of a complicated network. To detect the interaction patterns of the semantic labels, the labels should be expressed properly. In this paper, we adopts an oriented graph expression with the co-occurrences. A relation matrix R that corresponds to such oriented graph is employed, with element r_{ij} representing the relation between label l_i and l_j. The element r_{ij} is calculated using the following formula:

$$r_{ij} = P(l_i|l_j) = \frac{N(l_i, l_j)}{N(l_j)} \tag{1}$$

where $N(l_i, l_j)$ is the number of samples labeled with both l_i and l_j, $N(l_j)$ is the number of samples labeled with l_j.

With the obtained relation matrix R, we employ a non-negative matrix factorization based scene detection method for finding appropriate scenes of the labels. The NMF method detects the desired overlapping scenes within randomized iterations by optimizing the differences between R and WSW^T:

$$\min_{W,S\geq 0} \|R - WSW^T\|_F^2 \tag{2}$$

where $W \in \mathbb{R}^{m\times k}$ is a matrix representing the memberships of the m labels to the k scenes, and $S_{k\times k}$ is a matrix representing the correlations within each scene. Then we update the objective function by update rules in directed graph as in [15] until it keeps invariant. After the normalization operation of W by introducing a diagonal matrix, the entries in W can be taken as the probability of a certain label belonging to a certain scene.

There is a critical parameter k in such method since the number of scenes has crucial impact on the calculation of the matrices. The appropriate value of k is of the necessity to guarantee the overall detection quality. We introduce the modularity and dispersion coefficient to determine k.

The value of modularity can be calculated using the following formula:

$$\begin{aligned} M &= \frac{1}{\sum_{i,j}^{m} r_{ij}} \sum_{i,j}^{m} [\Phi(i,j)(r_{ij} - \frac{\sum_{k=1}^{m} r_{ik} \sum_{k=1}^{m} r_{jk}}{\sum_{i,j}^{m} r_{ij}})] \\ \Phi(i,j) &= \begin{cases} 1, & l_i \text{ and } l_j \text{ inside same community} \\ 0, & l_i \text{ and } l_j \text{ inside different communities} \end{cases} \end{aligned} \tag{3}$$

Larger the modularity, weaker the connectivity among communities. The modularity for semantic scenes detected in our paper indicates how well the scene detection is done. As the number of semantic scenes increases, the modularity increases gradually. However, when the number of scenes reaches a certain value, if we continue to increase the number of scenes, the empty scenes will emerge and the efficiency of the detection process will deteriorate. Thus, we let k increase incrementally and record the properties of the detected scenes until the one empty scene appears.

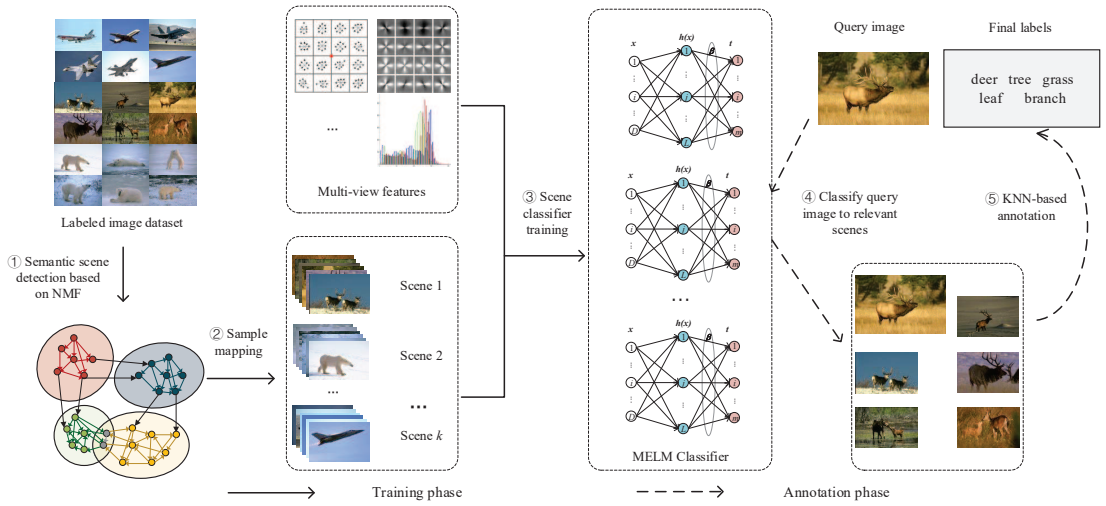


Fig. 1: The framework of the proposed method TAIA. Our method consists of two phases: the training phase represented by solid arrow and the annotation phase represented by dotted arrow. In the training phase, the latent semantic scenes are detected based on NMF and the mappings from visual features of images to latent semantic scenes are learnt. In the annotation phase, the query image will be labeled in the most related semantic scenes using a specialized KNN method.

The dispersion coefficient [16] that reflects the stability of the NMF method is defined as follows:

$$\rho = \frac{1}{n^2} \sum_{i=1}^{m} \sum_{j=1}^{m} 4 \times (C_{ij} - \frac{1}{2})^2 \tag{4}$$

where C is a consensus matrix whose elements reflect the probability that labels i and j belong to the same scene. The range of values for ρ is $[0, 1]$. If label i and label j belong to the same scene, NMF should partition them into the same scene with high probability by selecting the appropriate k, and vice versa.

In this paper, we devised a k selection mechanism based on the empirical patterns. We let k increase incrementally and calculate the corresponding dispersion coefficient and modularity until the empty scenes appear. We try to select the k with relative small value when the upward trend of the values of the two parameters M and ρ becomes smooth and steady. Once the value of k is determined, we can select the best probability matrix W with the maximum modularity. The pseudo code of semantic scene detection is shown in Algorithm 1.

2) Sample Mapping: After the detection, the training samples will be projected into the detected semantic scenes. It is worth noting that the semantic scenes may also be overlapping. Given a training sample $\langle x, \boldsymbol{y} \rangle$, where $\boldsymbol{y}$ is a vector representing the annotated labels, we can calculate the probability of the image belonging to a certain scene S_i using the formula:

$$p(S_i \mid \boldsymbol{y}) = \frac{\boldsymbol{y} \cdot W_{(:,i)}}{\sum_{j=1}^{k} \boldsymbol{y} \cdot W_{(:,j)}} \tag{5}$$

where $W_{(:,i)}$ is the i-th column of matrix W.

For the whole training set $\langle X, Y \rangle$, we can obtain the probability matrix as $P = V \cdot (Y \cdot W)$, where V is a normalized diagonal matrix. Using the probabilities of sample mapping, we can project the samples in a one-to-many style to the semantic scenes with corresponding probabilities. Even mapping one sample into one scene can still solve the problem of hard classification for labels since a label may belong to different scenes. In the experiments, we map the samples into the scenes with the highest corresponding probabilities. The

Algorithm 1: Semantic Scene Detection

Input: T (set of training samples), R_{DM} (number of runs to sample dispersion and modularity)
Output: W (probabilistic matrix representing the allocation of labels)

```
R = generateRelationMatrix(T);
//Detect Community
f = true; t = 0;
while f do
    t = t + 1;
    W_t = ∅;
    for i = 1 : R_DM do
        W_t^i = normalize(NMF(R, t));
        W_t = W_t ∪ W_t^i;
    M_t = modularity(R, W_t, R_DM);
    ρ_t = dispersion(W_t, R_DM);
    //Check the emergency of empty scenes
    f = detectEmptyScenes();
for k = 1 : t do
    if k is appropriate then
        break;
//pick W with the maximum modularity in W_k
W = pickW(W_k);
```

Algorithm 2: Scene Classifier Learning

Input: T (set of training samples), P (probability matrix obtained in the sample mapping process)
Output: Ω (learnt classifier MELM)
F = extractFeatures(T);
$E = \emptyset$;
for *each view feature* f_i *in* F **do**
  e_i = trainELM(P, f_i);
  $E = E \cup \{e_i\}$;
$\boldsymbol{\theta}$ = randWeight();
$\boldsymbol{\theta}$ = optimizeWeight($\boldsymbol{\theta}$, P, E, F, 'SHADE');
//Assemble the classifier
Ω = assembleELM(E, $\boldsymbol{\theta}$);

samples inside each semantic scene have strong intra-class semantic relations.

3) Scene Classifier Learning: As we have discussed above, we aim to label the new images with the help of the images of the similar scenes. Through the former processes, the semantic scenes are constructed. Then a scene classifier needs to be learned for projecting the new images to the related scenes. Thus, we propose an ensemble classifier with the capability of handling multi-view visual features.

The ensemble scene classifier is based on the extreme learning machine. It trains in analytical style. Therefore, it is fast with promising precision. For multi-view learning, the ensemble scene classifier is designed by integrating multiple ELMs, with each ELM for a single view feature. A leveraged result C_{final} referencing the multiple ELMs will be used as the final decision, which can be calculated using the following formula:

$$C_{final} = \sum_{v=1}^{V} \theta_v C_v = \boldsymbol{\theta} \cdot C \tag{6}$$

where C_v is the result of the ELM classifier corresponding to v-th view visual feature and $\boldsymbol{\theta}$ is the weight vector of the ELM classifiers. To obtain an appropriate weight vector, the parameter $\boldsymbol{\theta}$ is optimized by using a differential evolution (DE) variant SHADE [17], which is a numerical optimization operator that has demonstrated promising performance. The objective function of the optimization is designed as follows:

$$L(\boldsymbol{\theta}) = \|\boldsymbol{\theta} \cdot C - P\| + \lambda \|\boldsymbol{\theta}\|^2 \tag{7}$$

where P is the probability matrix obtained in the sample mapping process and λ is the parameter constraining the sparsity of $\boldsymbol{\theta}$.

The pseudo code for the learning process is presented in Algorithm 2.

B. TAIA: Online Annotation Phase

Using the semantic scenes and the classifier MELM constructed by the former processes, the annotation of a query image is conducted within the image cluster from related semantic scenes. Specifically, the multi-view visual features of the query image are extracted, and then the probabilities for the image belonging to different scenes are obtained by MELM. Combining the samples in the most relevant semantic scenes as a subset of training database, the annotation for the query image is conducted in the subset based on the specialized KNN. The pseudo code of the proposed TAIA is presented in Algorithm 3.

Algorithm 3: TAIA Framework

Input: T (training set), S (image set to be labeled), β (number of scenes for an image to be labeled within), K (number of nearest neighbours in KNN), N (number of labels for an image)
Output: S (labeled image set)
//Training Phase
W = detectScenes(T);
P = mapSamples(T, W);
F = extractFeatures(T);
Ω = trainClassifier(F, P);
//Annotation Phase
for *each* $s_i \in S$ **do**
  f_i = extractFeatures(s_i);
  Φ = pickScenes(f_i, Ω, β);
  s_i = specializedKNN(s_i, f_i, Φ, F_Φ, K, N);

The differences between the specialized KNN and the traditional KNN are twofold. First, in order to utilize the connectivity between labels and to avoid huge number of comparisons, the specialized KNN execute on the relevant semantic scenes of the query image. On the other hand, the traditional KNN execute on the entire training set. The samples in the relevant semantic scenes are less influenced by noise since their strong semantic connectivity. Moreover, there are fewer samples in the relevant scenes than in the entire training set, thus the specialized KNN can avoid huge number of comparisons. Second, in specialized KNN, the evaluation functions for different kinds of features varies. As recommended in a recent analytical studies [18], we use L_2 distance to quantify the differences in Gist feature, L_1 distance for the color histogram, and χ^2 for SIFT. The above differences are normalized in the range $[0, 1]$. The mean value of the differences is taken as the final distance among samples. Finally, the labels of the K nearest neighbors are collected, and the N most high frequent labels are selected to label the query image.

We boost the KNN algorithm by using KD Tree in the experiments. The time complexity of tree building for all semantic scenes is $O(Sn_s log_2 n_s)$, where S is the number of scenes and n_s is the number of training samples in each scene. The time complexity of annotation is $O(UNlog_2 n_s)$, where U is the number of untagged images and N is the number of most relative scenes. Therefore, the total time complexity of annotation based on KNN is $O(Sn_s log_2 n_s + UNlog_2 n_s)$.

III. Experimental Studies

In this section, we conduct experiments to test the effectiveness of the proposed TAIA, and compare the results with those of the state-of-the-art methods. Also, the key properties of the components of the TAIA are investigated.

A. Experimental Settings

The experiments in this section were executed on an Ubuntu PC with an Intel CPU (i5-2400, @3.1GHz) and 8GB RAM.

The settings and the details for the implemented version of TAIA are presented in Table I with corresponding descriptions.The experiments are conducted on Corel-5K [19], and IAPR-TC12 [20] datasets. The mean precision (P), mean recall (R), F1-score, the number of total labels with a recall greater than zero ($N+$) and the average precision AP [2] are selected as evaluation metric.

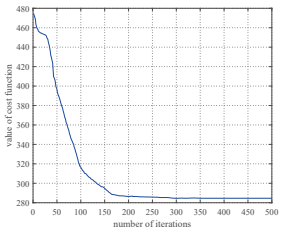
B. Tests on Corel-5K

TABLE II: Comparison results on Corel5K.

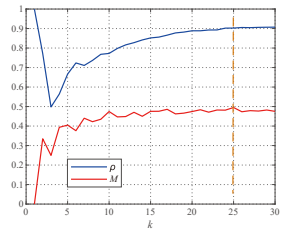
Method	P	R	$F1$	$N+$
SML [21]	23	29	25.7	137
GS [22]	30	33	31.4	146
MRFA [23]	31	36	33.3	172
TagProp(sigmaML) [18]	33	42	37.0	160
2PKNN [12]	**39**	40	39.5	177
RIA(dictionary) [24]	30	29	30.0	138
RIA(rare-first) [24]	32	35	32.0	139
RMLF [25]	29.7	32.6	31.1	-
LJNMF [26]	35.5	43	39.1	-
SEM [27]	36.5	48.4	41.6	173
CDNI [28]	29.8	32.1	30.9	162
OPSL [29]	37.4	**49.3**	42.5	177
TAIA	38.4	48.6	**42.9**	**177**

1) Results for Annotation Performance: In the experiment, we use a total of 15 publicly available features extracted by [18] to represent an image and train our classifier. As for KNN-based labeling, the number of the nearest neighbor is taken as 20 and the top 5 relevant labels are selected for annotation. Table II presents the experimental results, and the best results are marked in bold. It can be seen from Table II that the proposed TAIA yields the ranked 1st $F1$ value, ranked 2nd precision, ranked 1st recall and $N+$ value. The TAIA adopts an "image-scene-label strategy", i.e., an image can be assigned to many scenes, and a label may belongs to many scenes. This strategy may decrease the possibility of a correct label being missed. We also employ KNN-based method to handle imbalanced class distributions in the annotation phase. Therefore, TAIA achieves a higher recall rate. As the NMF based method in TAIA adopts an unsupervised setting, it will assign some labels to an incorrect scene occasionally, thus resulting in a relative lower precision.

2) Convergence of NMF and Parameter k Selection: Fig. 2 (a) presents the convergence graph of the scene detection algorithm based on NMF. It can be observed that after about 300 iterations, the cost function converges. The convergence pattern may vary under different train cases, but under most cases, the algorithm converges in 500 iterations.



(a) convergence



(b) modularity M and dispersion coefficient ρ

Fig. 2: The convergence, modularity and dispersion coefficient on Corel5K.

The scene detection algorithm relies on the parameter k, the number of the latent semantic scenes. We choose that value using the selection mechanism based on the empirical patterns. Fig.2 (b) presents the change curves in dispersion coefficient and modularity. We can see that when $k > 25$, the modularity and the dispersion coefficient curves become relatively stable and smooth. If we continue to increase k, empty scenes will emerge. The dispersion coefficient declines from 1 firstly and then increases with the increase of k. The dispersion coefficient represents the possibility of a label being distributed to a scene, demonstrating the stability of an algorithm. We expect to select the k with relative small value when the upward trend of the values of the two parameters M and ρ becomes smooth and steady. Thus we have taken the value of k as 25. The W with the highest modularity is picked out as the probability matrix from labels to scenes.

3) Results for Scene Detection: In the experiments, the Corel5K dataset is used to investigate the effectiveness for semantic scene detection and the related performance of the proposed TAIA. Using the W obtained by the aforementioned operations, the results of scene detection is presented in Table III. The mapping between the label and the scene is obtained by conducting the unsupervised NMF. This mechanism assigns labels that have strong interactions to the same scene. As we can see from Table III, most of the relationships between labels and semantic scenes are satisfactory. As relationships between labels and scenes are described by probability, we still have the probabilities to obtain weak label-scene couples. This results in a small proportion of incorrect labels (marked in italic and red) for some scenes.

C. Tests on IAPR-TC12

1) Results for Annotation Performance: To make fair comparison, five of the fifteen features extracted by [18] including Gist (512D), DenseHue (100D), HarrisHue (100D), DenseSift (1000D), HarrisSift (1000D) are used. The number of the nearest neighbors is taken as 20 and the top 5 relevant labels of the images are selected for annotation. Table V presents the results. The proposed TAIA creates a one-to-many way for the

TABLE I: The parameter settings and the corresponding descriptions.

Name	Value	Description
I_{max}	500	Number of iterations for the approximation of NMF
k	details in the experiments	Number of the semantic scenes for a certain test case
R_{DM}	50	Number of independent runs for NMF, used for sampling dispersion coefficient and modularity
L	1500	Number of hidden nodes in the hidden layer in each ELM
λ	0.3	Sparsity control parameter for weight optimization
β	2	Number of relevant image scenes for a query image

TABLE III: Results of scene detection on Corel5K.

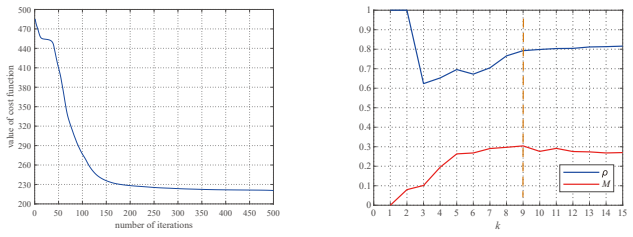
Scene	Corresponding Semantics	Labels according to the highest probabilities
S_1	livestock	vineyard, fence, field, horse, mare, foal
S_2	wildlife	head, forest, cat, tiger, bengal, lynx
S_3	water	sea, coral, anemone, fish, ocean, *basket*, reef, crab
S_4	racing	car, track, *wall*
S_5	flora	leaf, flower, *close-up*, plants, vines, tulip, petal, needle

TABLE IV: Results of scene detection on NUS-WIDE.

Scene	Semantics	Labels according to the highest probabilities
S_1	architecture	town, street, cityscape, nighttime, buildings, tower, window, house
S_2	mountains	glacier, valley, mountain, snow, rocks, frost, waterfall, plants
S_3	sky	sky, clouds
S_4	wildlife	map, tiger, cat, bear, animal, dog, fox, elk, leaf, birds
S_5	scenery	moon, fire, rainbow, statue, sand, sun, temple, food, sign, flags, tree, castle
S_6	sea	ocean, surf, whales, beach, boats, coral, lake, water, harbor
S_7	transport	airport, plane, military, vehicle, train, railroad, cars, road
S_8	human activity	book, soccer, tattoo, sports, person, protest, dancing, wedding, running, swimmers, police

TABLE VI: Results of scene detection on IAPR-TC12.

Scene	Semantics	Labels according to the highest probabilities
S_1	transport	bike, cycling, cyclist, helmet, road, short, car, hand, side
S_2	furnishing	corner, corridor, couch, curtain, floor, hammock, orange
S_3	nature	bloom, branch, bush, cliff, creek, fern, forest, garden, grass, hut, jungle, trunk, vegetation
S_4	landscape	cloud, hill, house, meadow, people, roof, sky, tree
S_5	human life	boy, child, desk, girl, glass, hair, hat, jacket, man, woman, shirt, sweater, trouser, tourist



(a) convergence

(b) modularity M and dispersion coefficient ρ

Fig. 3: The convergence, modularity and dispersion coefficient on IAPR-TC12.

TABLE V: Comparison results on IAPR-TC12.

Method	AP	$F1$
MLKNN [6]	0.294	0.151
NBVT [7]	0.283	0.154
TVSA-cur [30]	N/A	0.184
TVSA-prev [30]	N/A	0.181
LCMKL-G [31]	0.279	0.169
LCMKL-M [31]	0.321	0.180
SEM [27]	0.372	0.238
CDNI [28]	0.346	0.217
OPSL [29]	0.384	**0.253**
TAIA	**0.386**	0.244

TABLE VII: Comparative results for the effectiveness of the semantic scene based annotation on Corel-5K.

Method	P	R	$F1$	$N+$
KNN	23	24	23.4	113
LCKNN	35	39	36.9	144
TAIA	**38.4**	**48.6**	**42.9**	**177**

mappings from labels to semantic scenes by using NMF based scene detection, which avoids the missing of labels in the top two scenes. It can be seen that the proposed TAIA yield the best average precision and the ranked 2nd $F1$ value.

2) Convergence of NMF and Parameter k Selection: Fig. 3 presents the convergence curve and the change curves in

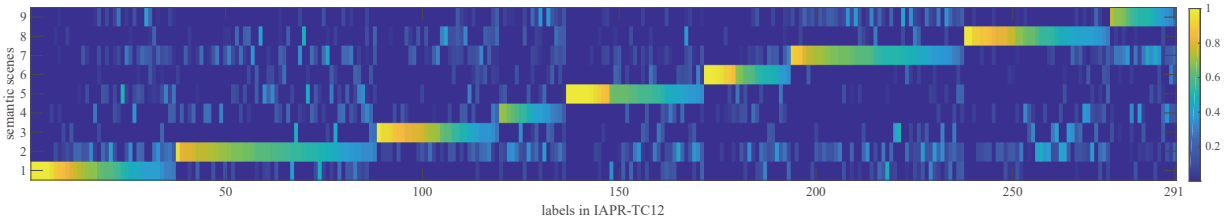


Fig. 4: The probability that each label in each scene on IAPR-TC12.

TABLE VIII: Comparative Results for MELM and MKL-SVM.

	IAPR-TC12			
	$F1$	AP	Train Time	Classification Time per Image
TAIA-MKL	0.176	0.347	38038.98s	0.495s
TAIA-MELM	0.244	0.386	50.52s	2.65×10^{-4}s

dispersion and modularity on the IAPR-TC12 dataset. It can be observed that Fig. 3 exhibits the same shape with those of Fig. 2. Fig. 3 (a) illustrates that the cost function converges after about 250 iterations, and we set the value of k as 9 from Fig. 3 (b) according to the strategy described previously.

3) Results for Scene Detection: In Tabel VI, the results of the scene detection on the IAPR-TC12 are presented. It can be seen that the labels fit the semantic scenes for most cases, and TAIA can obtain the satisfactory correspondence relations between labels and semantic scenes. Fig. 4 shows the probability that each label belongs to each semantic scene on IAPR-TC12 dataset, which also demonstrates that our algorithm can solve the hard classification problem of labels.

D. Component Analysis

1) Effectiveness for Semantic Scene based Annotation: To validate the effectiveness for semantic scene based annotation, TAIA is compared with the KNN based labeling on the whole training set (denoted as "KNN") and the KNN labeling on the semantic communities detected using the approach in [2] (denoted as "LCKNN"). In KNN and LCKNN, the other processes are same with the TAIA. Table VII presents the comparative results. It can be observed that TAIA and LCKNN achieve better performance than KNN, which demonstrates the effectiveness of semantic scene based labeling. Also, the performance of TAIA is better than LCKNN, which demonstrates the advancements of non-negative matrix factorization.

2) Analysis on MELM: In this part, we analyze the efficiency and effectiveness of the MELM and discuss the influences of the MELM structure on the performance.

First, we analyze the efficiency and effectiveness of the proposed MELM by comparing it with the MKL-SVM [2]. For the fairness of the comparison, the original implemented version of TAIA is compared with the version whose classifier is replaced using the MKL-SVM. The comparative results for AP, $F1$ and the time needed for training and classification are presented in Table VIII. As we can see from the table, no significant differences can be observed on the metrics for the proposed MELM and the popular MKL. However, the significant differences in training time and classification time have demonstrated the advancement of the proposed MELM.

The structure of the extreme learners influences the global performance. Fig. 5 presents the changes in performance along the changes of the number of hidden nodes in the ELM. From the plots, we can see that in a certain scale, the larger the number of hidden nodes, the better the performance. However, the differences in performance is quite small when the number of nodes increases to a certain number, thus the sensitivity is in a degree low. When the number of nearest neighbors is 20 and the number of hidden nodes is 1500, the proposed method TAIA achieves best performance.

IV. Conclusion

Inspired by the related contributions and aiming to address the current challenges in terms of effectiveness and efficiency, in this paper, we have proposed a two-stage automatic image annotation framework based on latent semantic scene classification (TAIA). In the offline training phase of TAIA, NMF-based semantic scene detection creates overlapping scenes for labels to solve the hard classification problem. A multi-view classifier MELM based on extreme learning machine with fast training speed is proposed to reduce training and classification time. In the online annotation phase of TAIA, the query image to be labeled is classified into the most relevant semantic scenes by the trained classifier and tagged by a specialized KNN, which utilizes the semantic interactions between the semantic labels and reduces the number of comparisons. The experiments have shown that the proposed TAIA achieves the competitive results in comparison with the state-of-the-arts in terms of effectiveness and efficiency.

Acknowledgment

The authors are grateful to the support of the National Natural Science Foundation of China (61976034,61572104), the National Key R&D Program of China (2018YFB1600600),and the Dalian Science and Technology Innovation Fund under Grant 2019J12GX035.

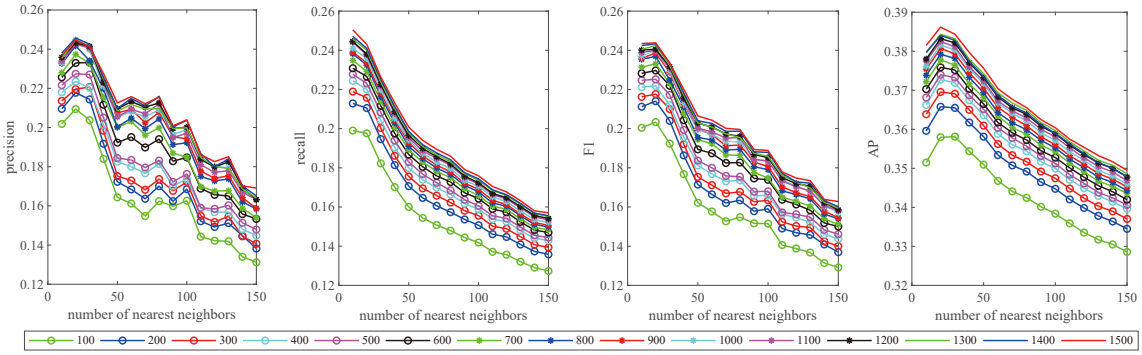


Fig. 5: The demonstration for the sensitivity for the ELM structure of MELM on IAPR-TC12. The differences in performance is quite small when the number of nodes increases to a certain number.

REFERENCES

[1] M. Hu, Y. Yang, F. Shen, L. Zhang, H. T. Shen, and X. Li, "Robust web image annotation via exploring multi-facet and structural knowledge," *IEEE Transactions on Image Processing*, vol. 26, no. 10, pp. 4871–4884, Oct. 2017.

[2] Y. Gu and X. Qian and Q. Li and M. Wang and R. Hong and Q. Tian, "Image Annotation by Latent Community Detection and Multikernel Learning," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3450–3463, Nov. 2015.

[3] Z. Y. Shi, Y. X. Yang, T. M. Hospedales, and T. Xiang, "Weakly-supervised image annotation and segmentation with objects and attributes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2525–2538, Dec. 2017.

[4] M. Jiu and H. Sahbi, "Nonlinear deep kernel learning for image annotation," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 1820–1832, Apr. 2017.

[5] M. M. Kalayeh, H. Idrees, M. Shah, "NMF-KNN: image annotation using weighted multi-view non-negative matrix factorization," in *IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, USA, 2014, pp. 184C191.

[6] M. Zhang and Z. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.

[7] M.-L. Zhang, J. M. P. na, and V. Robles, "Feature selection for multi-label naive bayes classification," *Information Sciences*, vol. 179, no. 19, pp. 3218–3229, 2009.

[8] M. Zand, S. Doraisamy, A. A. Halin, and M. R. Mustaffa, "Visual and semantic context modeling for scene-centric image annotation," *Multimedia Tools and Applications*, vol. 76, no. 6, pp. 8547–8571, 2017.

[9] B. Y. Wu, F. Jia, W. Liu, and B. Ghanem, "Diverse Image Annotation," in *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, 2017, pp. 6194–9202.

[10] H. Hu , G. T. Zhou , Z. Deng , Z. Liao , G. Mori, "Learning structured inference neural networks with label relations," in *IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, USA, 2016, pp. 2960–2968.

[11] Q. M. Cheng, Q. Zhang, P. Fu, C. H. Fu, and S. Li, "A survey and analysis on automatic image annotation," emphPattern Recognition, vol. 79, pp. 242–259, Jul. 2018.

[12] Y. Verma and C. V. Jawahar, "Image annotation using metric learning in semantic neighbourhoods," in *European Conference on Computer Vision*, Florence, Italy, 2012, pp. 836–849.

[13] T. Bracamonte, A. Hogan, and B. Poblete, "Applying community detection methods to cluster tags in multimedia search results," in *IEEE International Symposium on Multimedia*, Taichung, Taiwan, 2017, pp. 467–474.

[14] V. Maihami and F. Yaghmaee, "Automatic image annotation using community detection in neighbor images," *Physica A Statistical Mechanics and Its Applications*, vol. 507, pp. 123–132, 2018.

[15] F. Wang, T. Li, X. Wang, S. Zhu, and C. Ding, "Community discovery using nonnegative matrix factorization," *Data Mining and Knowledge Discovery*, vol. 22, no. 3, pp. 493–521, May. 2011.

[16] H. Kim and H. Park, "Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis," *Bioinformatics*, vol. 23, no. 12, pp.1495–1502, 2007.

[17] R. Tanabe and A. Fukunaga, "Success-history based parameter adaptation for differential evolution," in *IEEE Congress on Evolutionary Computation*, Cancun, Mexico, 2013, pp. 71–78.

[18] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid, "TagProp: Discriminative metric learning in nearest neighbor models for image auto-annotation," in *IEEE 12th International Conference on Computer Vision*, Kyoto, Japan, 2009, pp. 309–316.

[19] Corel Coporation, "Corel5k: The Corel database for content based image retrieval," Corel Discovery Center, Tech. Rep., 2009.

[20] M. Grubinger, P. Clough, H. Müller, and T. Deselaers, "The IAPR benchmark: A new evaluation resource for visual information systems," in *Proceedings of International Conference on Language Resources and Evaluation*, Genoa, Italy, 2006.

[21] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos, "Supervised learning of semantic classes for image annotation and retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 394–410, Mar. 2007.

[22] S. Zhang, J. Huang, Y. Huang, Y. Yu, H. Li, and D. N. Metaxas, "Automatic image annotation using group sparsity," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, USA, 2010, pp. 3312–3319.

[23] Y. Xiang, X. Zhou, T. S. Chua, and C. W. Ngo, "A revisit of generative model for automatic image annotation using markov random fields," in *IEEE Conference on Computer Vision and Pattern Recognition*, Miami, USA, 2009, pp. 1153–1160.

[24] J. Jin and H. Nakayama, "Annotation order matters: Recurrent image annotator for arbitrary length image tagging," in *International Conference on Pattern Recognition*, Cancun, Mexico, 2016.

[25] Y. Yao, X. Xin, and P. Guo, "A rank minimization-based late fusion method for multi-label image annotation," in *International Conference on Pattern Recognition*, Cancun, Mexico, 2016, pp. 847–852.

[26] R. Rad and M. Jamzad, "Automatic image annotation by a loosely joint non-negative matrix factorisation," *IET Computer Vision*, vol. 9, no. 6, pp. 806–813, 2015.

[27] Y. C. Ma, Y. J. Liu, Q. Xie, and L. Li, "CNN-feature based automatic image annotation method," Multimedia Tools and Applications, vol. 78, pp. 3767-3780, 2019.

[28] Y. Maihami and F. Yaghmaee, "Automatic image annotation using community detection in neighbor images," Physica A, vol.507, pp. 123-132, 2018.

[29] Z. Xue, G. R. Li and Q. M. Huang, "Joint multi-view representation and image annotation via optimal predictive subspace learning," Information Sciences, vol.451-452, pp. 180-194, 2018.

[30] Y. Gu, H. Xue, and J. Yang, "Cross-modal saliency correlation for image annotation," *Neural Processing Letters*, vol. 45, no. 3, pp.777–789, Jun. 2017.

[31] F. Zhong and L. Ma, "Image annotation using multi-view non-negative matrix factorization and semantic co-occurrence," in *IEEE Region 10 Conference*, Singapore, 2016, pp. 2453–2456.