Neurocomputing 102 (2013) 176–183

Contents lists available at SciVerse ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Semantic concept detection for video based on extreme learning machine

Bo Lu [a,*], Guoren Wang [a], Ye Yuan [a], Dong Han [b]

[a] School of Information Science and Engineering, Northeastern University, Shengyang 110004, People's Republic of China
[b] National Marine Data and Information Service, Tianjin 300171, People's Republic of China

ARTICLE INFO

Available online 5 June 2012

Keywords:
Extreme learning machine
ELM classifier
ELM-OAA classifier
Multi-modality
Probability-based fusion
Semantic concept detection
Contextual correlation
Single hidden layer feedforward networks

ABSTRACT

Semantic concept detection is an important step in concept-based semantic video retrieval, which can be regarded as an intermediate descriptor to bridge the semantic gap. Most existing concept detection methods utilize Support Vector Machines (SVM) as concept classifier. However, there are several drawbacks of using SVM, such as the high computational cost and large number of parameters to be optimized. In this paper we propose an Extreme Learning Machine (ELM) based Multi-modality Classifier Combination Framework (MCCF) to improve the accuracy of semantic concept detection. In this framework: (i) three ELM classifiers are trained by exploring three kinds of visual features respectively, (ii) a probability-based fusion method is then proposed to combine the prediction results of each ELM classifier, (iii) we integrate the prediction results of ELM classifier with the information of contextual correlation among concepts to further improve the accuracy of semantic concept detection. Experiments on the widely used TRECVID datasets demonstrate that our approach can effectively improve the accuracy of semantic concept detection and achieve performance at extremely high speed.

1. Introduction

With the amount of huge multimedia collections increasing at a tremendous speed, there is an urgent need to develop automatic retrieval systems for content-based video retrieval. Traditional content-based approaches often only use low-level features of the video material have proven their limitations in the face of the so-called semantic gap between the low-level features of video shots and the high-level semantic representation of the video material [1]. Recently, semantic concept detection has been proposed to solve this challenging problem, as it can be regarded as an intermediate descriptor to bridge the aforementioned semantic gap [2–5]. Using semantic concepts such as *car*, *outdoor*, *mountain* are used to comprehensively characterize the meaning of the video content. Detecting the presence or absence of these concepts in video shots can be leveraged to obtain effective results for semantic video retrieval [6,7].

The task of semantic concept detection then is to analyze the semantic representations of video shots. Most existing methods of the semantic concept detection [8–11] use traditional Support Vector Machines (SVM) as the concept classifier (or detector). The general approach to learn the concept classifier is based on generic visual features with manually labeled video shots. The output of the concept classifiers indicates the probability of a target concept presence in given video shots. The probability of the presence of certain semantic concepts are determined by the confidence of SVM output, in other words, the distance between the shot sample and decision boundary in the visual feature space. The decision boundary is determined to be the one for which the margin between the positive and negative video shot samples is maximized. Although the semantic concept detection based on SVM classifier enhances practical performance of content-based video retrieval to some extent, the accuracy of semantic concept detection still needs to be improved further. Meanwhile, the SVM classifier usually suffers from the high computational cost and the large number of parameters to be optimized.

To solve these issues, we apply Extreme Learning Machine (ELM) [12,13] as the concept classifier for semantic concept detection of video shots. ELM is a new algorithm based on Single Hidden Layer Feedforward Networks (SLFNs) [14,15] which may not be neuron like [14,16]. Meanwhile, ELM has fewer optimization constraints due to its special separability property for classification, and tends to achieve better generalization performance at very high learning speed as compared to traditional SVM. In addition, ELM classifier is good at addressing the multi-categories classification problem [20,21,29]. In this paper, we propose an Extreme Learning Machine based Multi-modality Classifier Combination Framework (MCCF) to improve the accuracy of the semantic concept detection, as shown in Fig. 1. Firstly, three ELM classifiers are trained using three different visual features namely *color*, *edge*, and *texture* respectively. Besides the

[*] Corresponding author.
E-mail address: mrcooler1982@gmail.com (B. Lu).

0925-2312/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2012.02.043

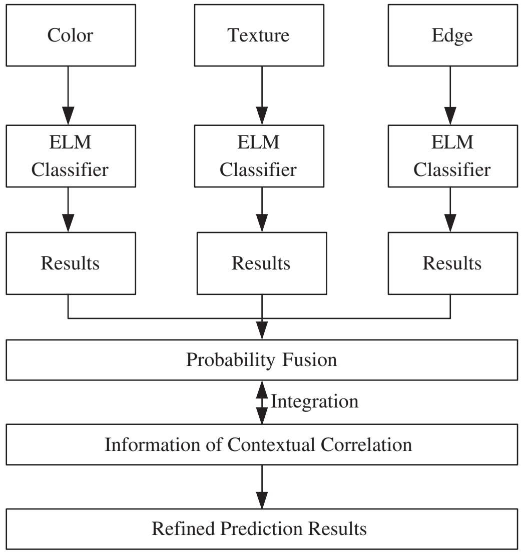

Fig. 1. Overview of extreme learning machine based multi-modality classifier combination framework (MCCF).

classical multi-class ELM methods are used as concept classifier for multi-categories classification, specifically, we also explore the One-Against-All (OAA) [17] method which can decompose a multi-class problem into a set of two-class problem. Secondly, we combine the prediction results of each ELM classifier based on a probability fusion method. The final result is an interpretable posterior probability of the concept relevance with respect to the given shot, which represents the probability of the occurrence of concept in that particular video shot. In addition, we consider the information of contextual correlation amongst concepts, which is propitious to infer the relationship between independent concepts for improving the semantic concept detection. We finally integrate the prediction results of ELM classifier with the information of contextual correlation among concepts to further improve the accuracy of semantic concept detection.

The rest of the paper is organized as follows. Section 2 briefly introduces the data representation in training phase and visual feature extraction procedure for video shots. In section 3, we describe the ELM based Multi-modality classifier combination framework. Section 4 reports the performance of our framework and give experiment results. We finally conclude the paper in Section 5.

2. Preliminaries

In this section, we briefly introduce the data representation in semantic concept detection, followed by a description of the feature extraction for video shots.

2.1. Data representation

For the purpose of semantic concept detection, a video is usually segmented into a sequence of shots. Let $S=\{s_1,s_2,\ldots,s_n\}$ be the training set which comprises of n shots; the indices of the shots are assigned based on their temporal orders in a video, e.g., s_{t-1} is the shot previous to s_t, and s_{t+1} is the shot following s_t. Let $C=\{c_1,c_2,\ldots,c_m\}$ be the concept lexicon, which contain m semantic concepts contained within the training set. To train the concept classifiers, each shot in the training set is manually annotated in advance with a set of corresponding labels. Let $L=\{L_{s_1},L_{s_2},\ldots,L_{s_n}\}$ be the set of labels of n shots. Due to m concepts must be labeled for each shot; label L_{s_t} corresponding to shot s_t is defined as an m-dimensional vector $[l_{s_t}^{c_1},l_{s_t}^{c_2},\ldots,l_{s_t}^{c_m}]^T$, in which $l_{s_t}^{c_i}$ can be either 1 or 0, which denotes whether concept c_i is present in shot s_t or not respectively. In the training phase, a set of features is extracted from each shot to characterize the visual properties of the annotated concept. The set of features contain information on *color*, *edge* and *texture*. Let $\{f_{s_1},f_{s_2},\ldots,f_{s_n}\}$ be the set of the visual features for training concept classifiers, where f_{s_t} is the visual feature extracted from shot s_t. Then, the concept classifier d_{c_i} for predicting concept c_i can be trained from the features and the manually labeled ground truth. To predict an unlabeled shot s_u given the corresponding the visual feature f_{s_u}, each trained concept classifier d_{c_i} provides a detection score in the range [0, 1] as the probability that concept c_i is present in testing shot s_u.

2.2. Visual feature extraction

To train the ELM classifier, we employ three different visual features namely *color*, *edge*, and *texture* respectively. These visual features are extracted by Grid Color Moment (GCM), Edged Direction Histogram (EDH), and Gabor Filters (GBR) respectively [18]. Furthermore, because the ELM classifiers work best with features which are roughly in the same dynamic range, which is not typically the case with the raw features that we have chosen. To address this issue, we have normalized the features using statistical normalization: using the sample mean μ and standard deviation σ to adjust each feature dimension to have zero mean and unit standard deviation.

Given a visual feature vector with n dimensions $f_i=(f_{i_1},f_{i_2},\ldots,f_{i_n})$, the sample mean vector μ and the standard deviation vector σ are calculated as:

$$\mu=\frac{1}{N}\sum_{i=1}^{N}f_i, \quad \sigma=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(f_i-\mu)^2} \tag{1}$$

where N is the number of features. Then, the normalized feature vector is given by

$$f_i'=\frac{(f_i-\mu)}{\sigma} \tag{2}$$

3. Multi-modality classifier combination framework based on ELM

In this section, we describe the main steps of our multi-modality classifier combination framework. Firstly, the ELM classifier for the multi-categories classification problem and multiple binary ELM classifiers based on OAA decomposition method are introduced. Secondly, a probability fusion method is proposed to combine the prediction results of each ELM classifier corresponding to the uni-modal feature. Finally, contextual semantic correlation among semantic concepts is described, which we integrate with the prediction results of the ELM classifier to further improve the accuracy of semantic concept detection.

3.1. ELM classifier for multi-categories classification

ELM is a new algorithm based on Single-Hidden Layer Feedforward Networks (SLFNs) [14,15]. Compared with traditional SVM, ELM not only tends to reach the smallest training error but also the smallest norm of the output weights. ELM is not very sensitive to user specified parameters and has fewer optimization

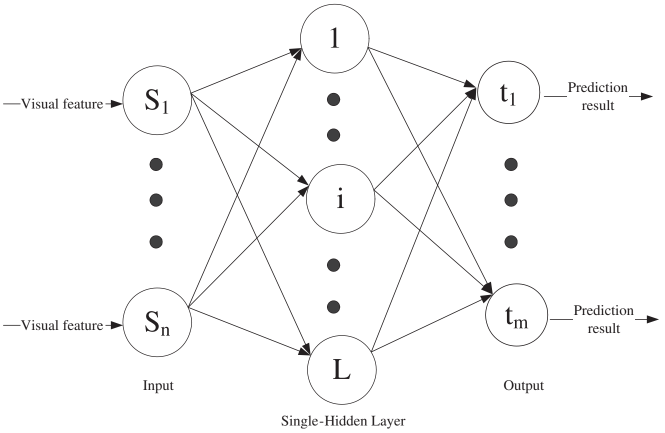

Fig. 2. Example of ELM classifier with multiple output nodes for multi-categories classification.

constraints [19]. In addition, ELM tends to provide good generalization performance at extremely high learning speeds.

For the multi-categories classification problem, ELM classifier [20] uses a network of multiple output nodes equal to the number of pattern classes m, as shown in Fig. 2. For each training shot sample s_i, the target output t_i is an m-dimensional vector $(t_1,t_2,\ldots,t_m)^T$.

The learning procedure of the ELM classifier is given below. For N arbitrary distinct shot samples $(s_i,t_i) \in R^n \times R^m$, if an SLFN with L hidden nodes can approximate these N samples with zero error, then we have

$$\sum_{i=1}^{L} \beta_i G(a_i,s_j,b_i) = t_j, \quad j=1,\ldots,N \tag{3}$$

where a_i is the weight vector connecting the ith hidden node and the input nodes, β_i is the output weight linking the ith hidden node to the output node, and b_i is the threshold of the ith hidden node. $G(a_i,s_j,b_i)$ is the output of the ith hidden node with respect to the input s_j. In our simulations, sigmoid activation function of hidden nodes are used, we formulate it as $G(a_i,s_j,b_i) = 1/(1+\exp(-a_i \cdot s_j + b_i))$. Under these conditions, Eq. (3) can be written compactly as

$$H\beta = T \tag{4}$$

where

$$H = \begin{bmatrix} G(a_1,s_1,b_1) & \cdots & G(a_L,s_1,b_L) \\ \vdots & \cdots & \vdots \\ G(a_1,s_N,b_1) & \cdots & G(a_L,s_N,b_L) \end{bmatrix}_{N\times L} \tag{5}$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L\times m}, \quad T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N\times m} \tag{6}$$

While computing, the determination of the output weights β is estimated by the smallest norm least-squares solution and defined as

$$\hat{\beta} = H^{\dagger} T \tag{7}$$

where $H^{\dagger}$ is the Moore–Penrose generalized inverse [22] of the hidden layer output matrix H. The original algorithm of ELM proposed by Huang et al. [13] contains three steps as follows:

Algorithm ELM: Given a training set $\aleph = \{(s_i,t_i) | s_i \in R^n, t_i \in R^m, i=1,\ldots,N\}$, activation function $G(x)$, and hidden node number L, then training the ELM classifier takes several steps:

*Step*1: Randomly assign hidden node parameters (a_i,b_i), $i=1,\ldots,L$.
*Step*2: Calculate the hidden layer output matrix H.
*Step*3: Calculate the output weight $\hat{\beta} = H^{\dagger} T$.

For a training shot sample s, the single ELM classifier produces m outputs represented by

$$L(s) = \sum_{i=1}^{L} \hat{\beta}_i G(a_i,s,b_i) \tag{8}$$

Here, $L(s) = [L_1(s),\ldots,L_m(s)]^T$ denote the output function of the m output nodes. Then, the concept label which the shot s belongs to is determined by the output node with the largest prediction value, which can be defined as

$$\tilde{L}(s) = \arg \max_{i=1,\ldots,m} L_i(s) \tag{9}$$

3.2. ELM-OAA for multiclass classification

Generally, multi-label classification problem based on ELM is decomposed into multiple binary classification problems according to the inherent concept label correlation among the training data. Rong et al. [17] proposes a major decomposition method, which is called the One-Against-All (OAA) approach. In our alternative approach, the OAA decomposition method is used and each binary classifier based on the ELM algorithm, we name it as ELM-OAA for simplicity. Specifically, in this section, each classifier for uni-modal feature is trained based on the ELM-OAA approach. We will introduce the ELM-OAA classifier as follows.

In the ELM-OAA classifier, the m-label classification problem is implemented by m binary ELM classifiers, each of which is trained independently to classify one of the m labels with the training data relevant to each concept. Each binary ELM has the same input data, however, the target output is different. For the xth binary ELM classifier, the target output data need to be decomposed into two subsets: one labeled as 1 for all the xth class samples, which represent the shot with xth concept label. The other labeled as -1 for all the samples belonging to the other concept labels. The xth binary ELM classifier has one output node with an output value y_x. When a training shot sample s is input, the ELM-OAA classifier produces the corresponding m prediction values from the m binary classifiers. For the m-label classification problem, the number binary ELM classifiers required by ELM-OAA is m, for the reason that the ELM-OAA can be regarded as a combination of m SLFNs which share the same hidden nodes, as shown in Fig. 3. As illustrated in Fig. 3, the same hidden nodes can be randomly generated for all the binary ELM classifiers. m groups of independent training data are obtained according to the m concept label to train the m binary classifiers. The m output neurons represent the output of m binary classifiers and thus they are independently trained. In the SLFNs, the weight β^x connecting the xth binary classifier output neuron with the shared hidden nodes is learned without affecting other weights. We define the weight vector β^x as $[\beta_1^x,\ldots,\beta_L^x]^T$. According to the theory of ELM as mentioned above, β^x then can be calculated as

$$\beta^x = (H^x)^{\dagger} t^x \tag{10}$$

where t^x is defined as $t^x = [t_1^x,\ldots,t_{N^x}^x]^T$, N^x is the training data size for the xth binary classifier.

H^x is defined as

$$H^x = \begin{bmatrix} G(a_1,s_1^x,b_1) & \cdots & G(a_L,s_1^x,b_L) \\ \vdots & \cdots & \vdots \\ G(a_1,s_{N^x}^x,b_1) & \cdots & G(a_L,s_{N^x}^x,b_L) \end{bmatrix}_{N^x\times L} \tag{11}$$

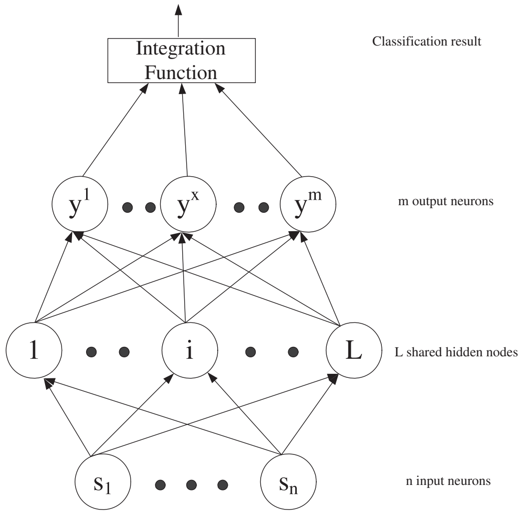

Fig. 3. Example of SLFNs of OAA-ELM with shared hidden neurons.

Since the multiple outputs from the SLFNs corresponding to the each binary ELM classifier represent collective outputs for the m classes, thus, we need to integrate them into the final class label for the training shot sample s. Here the loss based decoding method [23] is used to integrate the output of each binary ELM classifier. In this approach, the chosen class label is most consistent with the output y_i, which means the total loss for sample (s_i,t_i) is the minimum for all the class labels. The total loss on the training shot sample s is defined as

$$D(M,y_i(s)) = \sum_{x=1}^{m} \Theta(M(i,x)y^x(s)) \tag{12}$$

where i represents the index of the output nodes. M is an $m \times m$ matrix, the diagonal elements are $+1$ and the other elements are -1. $y^x(s)$ is output value for the xth binary classifier and Θ is the exponential loss function. Based on the Eq. (12), the final output for the training sample s is defined as

$$\tilde{y}(s) = \underset{i=1,\ldots,m}{\arg\min} D(M,y_i(s)) \tag{13}$$

3.3. Probability fusion method of multi-modality

In the task of semantic concept detection, for a concept c_i, a training video shot is s_i labeled with $l_{s_i}^{c_i}$, where $l_{s_i}^{c_i} \in \{0,1\}$. A value for $l_{s_i}^{c_i}=1$ signifies the concept c_i is present in shot s_i, while on the other hand, $l_{s_i}^{c_i}=0$ means the concept c_i is not present in shot s_i. For simplicity, we assume the feature for shot s_i is a feature vector f_{s_i} and discuss probabilistic fusion of the classifier results for each feature.

As mentioned before in Section 3.2, we learn three ELM-OAA classifiers for each visual feature. However, we hope to combine the prediction results of the ELM-OAA classifiers on each unimodal feature to obtain a fused prediction score. To predict the semantic concepts in an unlabeled shot sample s_u, given the corresponding single feature $\{f_{s_u}^k\}^K$ which is extracted from a set of K (K=3) different features; each trained ELM-OAA classifier provides a prediction value on [0,1] which serves as the posterior probability $P(t_i|\{\tilde{y}\}^K)$. Here, we assume that different features are conditionally independent and define the posterior probability corresponding to the K different features as

$$P(\{\tilde{y}\}^K|t_i) = \prod_{k=1}^{K} p(\{\tilde{y}\}^k|t_i) \tag{14}$$

According to the Bayes theory $P(A|B) = p(A)/p(B)P(B|A)$, we come to the probability fusion of multiple features:

$$P(t_i|\{\tilde{y}\}^K) = \frac{p(t_i)}{p(\{\tilde{y}\}^K)} P(\{\tilde{y}\}^K|t_i) \tag{15}$$

where $P(\{\tilde{y}\}^K) = \prod_{k=1}^{K} p(\tilde{y})^k$ and the prior probability $P(t_i)$ is assumed to be a uniform distribution. From Eqs. (14) and (15), we define the fused result as

$$P(t_i|\{\tilde{y}\}^K) = \frac{P(t_i)}{P(\{\tilde{y}\}^K)} \prod_{k=1}^{K} p(\{\tilde{y}\}^k|t_i) \tag{16}$$

In a similar way, for the prediction results of three ELM classifiers, we define the fused result as

$$P(t_i|\{\tilde{L}\}^K) = \frac{P(t_i)}{P(\{\tilde{L}\}^K)} \prod_{k=1}^{K} p(\{\tilde{L}\}^k|t_i) \tag{17}$$

3.4. Inference of contextual correlation

Generally, the information for contextual correlation can be used to infer semantic concepts. The contextual correlation describes the relations between independent semantic concepts, where some concepts often co-occur within the same shot and other concepts are mutually exclusive in most shots. For the semantic concept c_t, the probability of presence of c_t in unlabeled shot s_u can be inferred by selecting the most relevant concepts from the orthogonal semantic concept space [26]. Note that, in our previous work [26], we constructed an orthogonal semantic space by exploring the relationship of concepts. All concepts are mapped onto the semantic space, which is described in detail in [26].

Here, the selected concepts can be regarded as the condition of the query concept occurring in shot. For example, if the selected related concepts are *car* and *outdoor* for a given concept *road*, the conditional probability of concept *road* can be described as:

$$\begin{aligned} P(road) &= P(road|car=1 \cap outdoor=1)P(car=1 \cap outdoor=1) \\ &= P(road|car=1 \cap outdoor=1)P(car=1)P(outdoor=1) \end{aligned}$$

Furthermore, we can obtain the conditional probability that a semantic concept occurs given a corresponding condition. We further derive the inferred probability of concept c_t occurring in unlabeled shot s_u, which is defined as:

$$P(l_{s_u}^{c_t}) = \sum_k P(l_{s_u}^{c_t}|\partial(\varphi_k,s_u)=1)P(\partial(\varphi_k,s_u)=1) \tag{18}$$

where φ_k is a conditional set (the conditional set includes the related concept for the given concept), and

$$\partial(\varphi_k,s_u) = \begin{cases} 1 & \text{shot } s_u \text{ satisfies all of the conditions of } \varphi_k \\ 0 & \text{otherwise} \end{cases}$$

By exploring the prediction score of ELM classifier and the contextual correlation between concepts, we can use them to further improve the accuracy of semantic concept detection. The inferred probability with contextual correlation is defined as $P(l_{s_u}^{c_t}|R_{c_t}^{ctx})$. Given the selected m concepts, we can calculate the inferred probability of c_t occurring in s_u using the following equation:

$$P(l_{s_u}^{c_t}|R_{c_t}^{ctx}) = \sum_{k=1}^{m} (P_k \cdot P(\partial(\varphi_k,s_u)=1)) \tag{19}$$

where $P(\partial(\varphi_k,s_u)=1)$ is the probability that unlabeled shot s_u satisfies the condition φ_k.

As we know from above, the concept detector outputs a prediction score for the concept c_t occurring in any unlabeled shot s_u. Next, we have to further integrate the prediction results of the ELM classifier and the information from contextual correlation. We use $\widehat{P^{c_t}_{s_u}}$ to indicate the refined score, which should satisfy the contextual correlation and approximate the prediction score for each concept in each shot. Therefore, we simultaneously take both factors into account to obtain a refined score.

When the given query concept c_t is present in an unlabeled shot s_u, the energy term of c_t can be defined as:

$$E^{c_t}_{s_u} = \begin{cases} \|\widehat{P^{c_t}_{s_u}} - P(t_i|\{\tilde{y}\}^K)\|^2 + \frac{\lambda}{2}\|\widehat{P^{c_t}_{s_u}} - P(l^{c_t}_{s_u}|R^{ctx}_{c_t})\|^2 & \text{if classifier is ELM} - \text{OAA}, \\ \|\widehat{P^{c_t}_{s_u}} - P(t_i|\{\tilde{L}\}^K)\|^2 + \frac{\lambda}{2}\|\widehat{P^{c_t}_{s_u}} - P(l^{c_t}_{s_u}|R^{ctx}_{c_t})\|^2 & \text{if classifier is ELM.} \end{cases} \tag{20}$$

Here the weighting parameter λ can be adjusted according to the concept, because we observed that the reliability of contextual correlation varies from concept to concept. Let $\omega^{c_t}_d$ and $\omega^{c_t}_c$ be the cross validation AP obtained using the concept detector and the contextual correlation for concept c_t, respectively. We set $\lambda = \omega^{c_t}_c / \omega^{c_t}_d$. In our experiments, since APs of most concepts range from 0.3 to 0.5, we set $\omega^{c_t}_d = 0.45$ for all concepts; $\omega^{c_t}_c$ of the contextual correlation is estimated from the annotations of the ground truth.

Then, based on $E^{c_t}_{s_u}$ in Eq. (20), the potential integration function is formed by summing all energy produced by each concept for each testing shot, which is defined as:

$$\widehat{p^{c_t}_{s_u}} = \sum_{t=1}^{m}\sum_{i=1}^{n} E^{c_t}_{s_u} \tag{21}$$

where n is the total number of shots in the test set. We can obtain the final refined scores by solving Eq. (21).

4. Performance evaluation

4.1. Experiment datasets

To evaluate the performance of the proposed approach, we conduct experiments on the benchmark TRECVID [24] datasets TV05 and TV06 respectively. We use the TV05 development set as the training corpus and performed the evaluations on the TV06 test set. TV05 and TV06 are composed of broadcast news videos in English, Chinese and Arabic. TV05 contains 85 h of videos, which includes 137 videos and 45,765 shots. TV06 contains 150 h of videos, which includes 259 videos and 79,484 shots. We use the concept labels of the TV05 dataset from *Columbia374* [27], which has a lexicon of 374 semantic concepts. The optimization is performed independently on each video for simultaneously labeling of all concepts and shots. We describe performance on the 20 officially evaluated concepts, as shown in Fig. 4.

To compare the effectiveness of our proposed framework MCCF, the concept detectors of *Columbia374* and *VIREO374* [28] are utilized; here we name them as SVM-col374 and SVM-VIR374 respectively. Because concept classifier of *columbia374* and *VIREO374* are both associated with three SVM classifiers trained with visual features: color, texture and edge respectively, the structure of both classifiers is similar to our proposed framework MCCF.

We conduct all experiments in MATLAB 7.0 environment running on a Intel Core-Duo 2.8 GHz CPU with 2 GB memory and a 500 GB hard disk.

4.2. Evaluation criteria

The evaluation criteria here is the mean average precision (MAP), which is the mean of average precision (AP) of each concept. In words, the AP is the sum of the precision at each relevant concept in the rank list divided by the total number of relevant shots in the collection. AP is defined as below:

$$AP = \frac{1}{W}\sum_{j=1}^{T}\frac{W_j}{j} \times I_j \tag{22}$$

where T is the total number of shots in the test set, and W represents the number of relevant shots. For any given index j, let W_j be the number of relevant shots among the top j shots. Let $I_j = 1$ if the jth shot is relevant and 0 otherwise.

4.3. Semantic concept detection by multi-modality features

To test the effectiveness of our proposed framework, we apply the ELM, ELM-OAA, SVM-VIR374 and SVM-col374 classifiers using three visual features, namely *color*, *edge*, and *texture* respectively to detect semantic concept. For the cross validation in SVM-col374 and SVM-VIR374, we split the training data into two parts, one for training and the other for the cross validation runs. Table 1 shows the MAP performance of the aforementioned four classifiers. As shown in Table 1, our proposed framework performs best for 14 out of 20 concepts. Specifically, the ELM-OAA classifier based MCCF improves the MAP by 15.6%, ELM classifier based MCCF improves the MAP by 15.8%, which doubles that the performance of the SVM-col374 and SVM-VIR374. As can be observed in Table 1, to our surprise, MCCF(ELM) and MCCF(ELM-OAA) achieve similar testing accuracy. Moreover, the consistent MAP improvements of MCCF on all features demonstrate that the ELM is a good generic classifier.

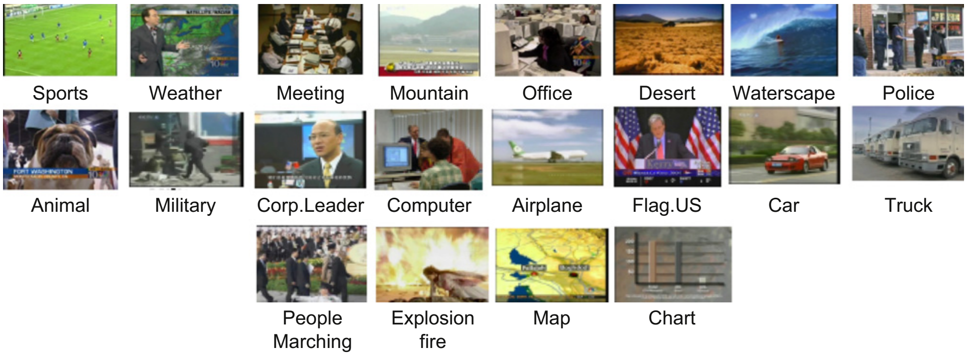

Fig. 4. Example of keyframes of 20 officially evaluated concepts used in TV 06.

4.4. Semantic concept detection by integrating the multi-modality features with the information of contextual correlation

In this section, we combine the information of contextual correlation with the set of visual features to test the effectiveness of MCCF. In MCCF, the multi-modality fusion is accomplished by the probability fusion method described in Section 3.3. Here, the inference of contextual relationships between the semantic concepts is used and based on that, we integrate the prediction results of the concept classifier and the information on contextual correlation to obtain a refined score. For SVM-col374 and SVM-VIR374, the late fusion of averaging individual SVM outputs of the three features is exploited. Fig. 5 shows that the MAP of the four kinds of concept detectors for 20 concepts in the official evaluation of the TRECVID 2006 benchmark. Overall, MCCF(ELM) and MCCF(ELM-OAA) achieve a MAP of 0.376, which implies approximately 11.7% improvements over SVM-col374 and SVM-VIR374 with MAP of 0.254. Specifically, for 12 out of 20 concepts, MCCF(ELM) outperforms SVM-col374 and SVM-VIR374 with a considerable margin, and for 14 out of 20 concepts, MCCF (ELM-OAA) is outperforming SVM-col374 and SVM-VIR374. This manifests that the probability fusion with proper Bayesian theoretical foundations is more suitable than the average SVM fusion. Meanwhile, we integrate the information of contextual correlation with prediction score of classifier, which further improves the accuracy of semantic concept detection.

Table 1
Performance comparison with all four concept classifiers on multi-modality features.

Concept	SVM-VIR374	SVM-col374	MCCF(ELM)	MCCF(ELM-OAA)
Sports	0.397	0.431	**0.450**	**0.445**
Weather	0.386	0.357	**0.453**	**0.459**
Meeting	0.322	0.327	**0.350**	**0.356**
Office	0.045	0.025	**0.079**	**0.086**
Mountain	0.214	**0.229**	0.237	0.220
Desert	0.048	0.056	**0.061**	**0.059**
Waterscape	0.139	**0.145**	0.132	0.128
Police_Security	0.016	**0.019**	0.017	0.015
Military	0.019	0.122	**0.135**	**0.134**
Animal	0.005	0.004	**0.007**	**0.007**
Computer_TV	0.256	0.273	**0.279**	**0.289**
Corporate	0.037	0.005	**0.056**	**0.047**
Airplane	0.051	0.050	**0.052**	**0.054**
Car	0.176	0.183	**0.189**	**0.191**
Flag_US	0.295	**0.316**	0.287	0.291
Truck	0.080	**0.082**	0.079	0.082
People_March	0.059	0.060	**0.065**	**0.068**
Explosion_Fire	0.160	0.165	**0.167**	**0.169**
Map	0.300	0.294	**0.338**	**0.341**
Chart	0.200	**0.204**	0.179	0.181
MAP	0.160	0.167	**0.180**	**0.179**
Improvement	6.9%	7.3%	15.8%	15.6%

To further confirm whether MCCF can outperform SVM-col374 and SVM-VIR374 in MAP improvements, we perform the significance test on the MAP results of 20 concepts. The test is to determine whether the MAP improvements by MCCF over SVM-col374 and SVM-VIR374 are significant, or whether they are just due to chance. Note that, here we only contrast with SVM-col374, because its MAP is much better than SVM-VIR374 for the 20 evaluated concepts. There are several types of significance test methods available. Here, we exploit the *t*-test [25] to measure the significance of MAP improvements. The *t*-test is an accurate indicator in measuring difference in mean average precision. The output of *t*-test is actually a signal–noise ratio, which is defined as:

$$t=\frac{mean(AP_{MCCF})-mean(AP_{SVM-col374})}{\sqrt{\mathrm{var}(AP_{MCCF})+\mathrm{var}(AP_{SVM-col374})}} \tag{23}$$

where $mean(AP_{MCCF})$ and $\mathrm{var}(AP_{MCCF})$ are the mean and variance of APs by MCCF. Intuitively, the larger the *t*-value is the smaller the probability of chance. By substituting the APs of 20 concepts into Eq. (23), we have $t=5.1$. By looking up the *t*-test table, we find that the $P(chance|t=5.1)\approx 0.05$. In other words, there is 0.05 probability that the MAP improvements by MCCF is by chance.

In addition, MCCF (include the ELM and ELM-OAA classifiers) and SVM-col374 yield different MAP scores, their relative performances on multiple concepts are quite different. This demonstrates that the ELM and SVM generally leads to different results from the relative complexity of multiple concepts. One major factor of concept complexity is found to be the number of positive training samples available. By correlating the MAP (in Fig. 5) with the number of positive samples available (in Fig. 6) for each concept, we observe that the concepts with a large number of positive samples tend to deliver more satisfactory MAP.

4.5. Complexity analysis

The video corpus tends to consist of enormous amount of keyframes of video shot. Datasets of training includes 35,647 in

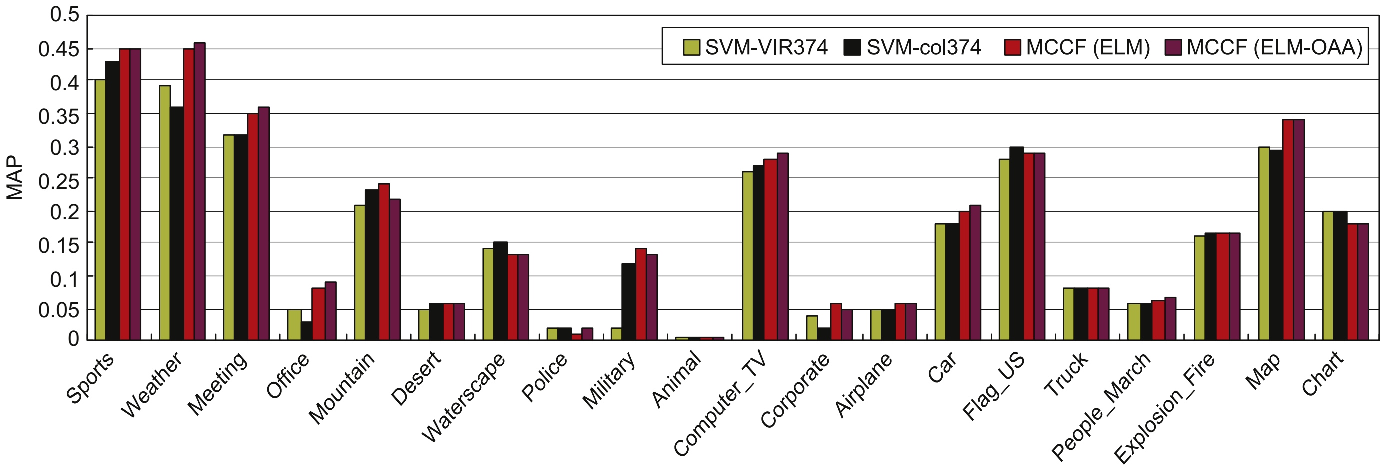

Fig. 5. MAPs of the MCCF(ELM) and the MCCF(ELM-OAA) by integrating the information of contextual correlation with multi-modality features as compared to the SVM-VIR374 and the SVM-col374 for 20 concepts in the official evaluation of the TRECVID 2006 benchmark.

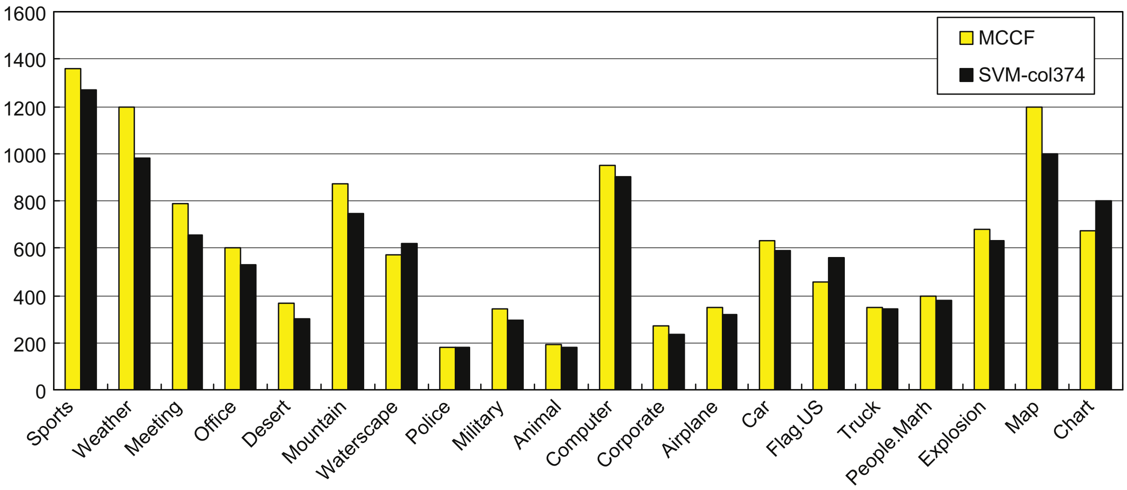

Fig. 6. The number of positive training samples available for each concept.

Table 2
Comparison of training and testing time (s) of MCCF, SVM-VIR374 and SVM-col374.

Datasets	Class	SVM-VIR374 SVM		SVM-col374 SVM	
		Training(s)	Testing(s)	Training(s)	Testing(s)
TV05/06	20	269.75	47.86	260.47	43.79
		MCCF		MCCF	
		ELM		ELM-OAA	
		7.54	1.78	35.68	13.56

TV05 and 19,562 in TV06 for testing. Thus, the algorithmic complexity of a classifier is critical. In our experiments, the computational complexity depends on the feature dimensionality, the number of hidden nodes and support vectors. Here, the dimensionality of feature d equals 256. Table 2 gives the training and testing times for all classifier methods. From this we can conclude that the ELM classifier by far is the fastest approach. Both SVM-VIR374 and SVM-col374 require in similar amounts of time. For the ELM and ELM-OAA classifier, they run 39 and 7 times faster than SVM-VIR374 and SVM-col374 respectively. Moreover, the running time of ELM classifier is lower than that of the ELM-OAA classifier. However, the ELM-OAA requires a similar or lower amount of training time as compared to ELM when the number of pattern class is not larger than 10. Because the ELM-OAA decomposes the multiclass classification problem into multiple binary classifiers, each binary classifier requires smaller number of hidden nodes.

5. Conclusions

In this paper, we proposed a multi-modality classifier combination framework based on Extreme Learning Machine (ELM) to improve the accuracy of semantic concept detection from video frames. To achieve better prediction accuracy, firstly three ELM classifiers are trained using three visual features respectively. Secondly, a probability based fusion method is proposed to combine the prediction results of each ELM classifier. We also consider the information from contextual correlation among concepts and infer the relationship between concepts. Finally, we integrate the prediction score from ELM classifier and the information derived from the contextual correlation amongst concepts to further improve the accuracy of the semantic concept detection. Experiments show that our approach can achieve a good performance, while requiring far lower running times for the classifier than other existing methods.

Acknowledgments

This research are supported by National Science Fund of China Key Program (Grant No. 60933001), NSFC (Grant No. 61100024), NSFC (Grant No. 61173030), National Basic Research Program of China (973, Grant No. 2011CB302200-G) and National Science Fund for Distinguished Young Scholars (Grant No. 61025007).

References

[1] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, R. Jain, Content based image retrieval at the end of the early years, IEEE Trans. PAMI 22 (2000) 1349–1380.
[2] W. Adams, G. Iyengar, G.Y. Lin, M. Naphade, C. Neti, H. Nock, Semantic indexing of multimedia content using visual, audio and text cues, EURASIP J. Adv. Signal Processing 2003 (2) (2003) 170–185.
[3] S.F. Chang, W.Y. Ma, A. Smeulders, Recent advances and challenges of semantic image/video search, in: Proceedings of ICASSP, 2007.
[4] L. Kennedy, S.F. Chang, A reranking approach for context-based concept fusion in video indexing and retrieval, in: Proceedings of CIVR, 2007.
[5] C.G.M. Snoek, M. Worring, Multimodal video indexing: a review of the state-of-the-art, Multimedia Tools Appl. 25 (1) (2005) 5–35.
[6] C.G.M. Snoek, B. Huurnink, L. Hollink, M. de Rijke, G. Schreiber, M. Worring, Adding semantics to detectors for video retrieval, IEEE Trans. Multimedia 9 (5) (2007) 975–986.
[7] J.R. Smith, M. Naphade, A. Natsev, Multimedia semantic indexing using model vectors, in: Proceedings of ICME, 2003.
[8] M.R. Naphhade, J.R. Smith, On the detection of semantic concepts at TRECVID, in: Proceedings of ACM International Conference on Multimedia, 2004, pp. 660–667.
[9] T.-S. Chua, S.-Y. Neo, Y.-Y. Zheng, H.-K. Goh, Y. Xiao, S. Tang, M. Zhao, Trecvid-2006 by nus-i2r, in: TREC Video Retrieval Evaluation Proceedings, 2006.
[10] S.F. Chang, W. Hsu, L. Kennedy, L. Xie, A. Yanagawa, E. Zavesky, D.-Q. Zhang, Columbia university trecvid-2005 video search and high-level feature extraction, in: TREC Video Retrieval Evaluation Proceedings, 2006.
[11] C.G.M. Snoek, M. Worring, J.C. van Gemert, J.-M. Geusebroek, A.W.M. Smeulders. The challenge problem for automated detection of 101 semantic concepts in multimedia, in: Proceedings of ACM International Conference on Multimedia, 2006, pp. 421–430.
[12] G.B. Huang, Q.Y. Zhu, C.K. Siew, Extreme learning machine: a new learning scheme of feedforward neural networks, in: Proceedings of IEEE International Joint Conference on Neural Networks, 2004, pp. 985–990.
[13] G.B. Huang, Q.Y. Zhu, C.K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (2006) 489–501.
[14] G.B. Huang, L. Chen, Convex incremental extreme learning machine, Neurocomputing 70 (2007) 3056–3062.
[15] G.B. Huang, Q.Y. Zhu, K.Z. Siew, P. Saratchandran, N. Sundararajan, Can threshold networks be trained directly? IEEE Trans. Circuits Syst. II 53 (3) (2006) 187–191.

[16] G.B. Huang, L. Chen, Enhance random search based incremental extreme learning machine, Neurocomputing 71 (2008) 3460–3468.

[17] H.J. Rong, G.B. Huang, Y.-S. Ong, Extreme learning machine for multi-categories classificaiton applications, in: Proceedings of IEEE International Joint Conference on Neural Networks, 2008, pp. 1709–1713.

[18] A. Yanagawa, W. Hsu, S.-F. Chang, Brief Descriptions of Visual Features for Baseline TRECVID Concept Detectors. Columbia University ADVENT Technical Report 219-2006-5, 2006.

[19] G.B. Huang, X.-J. Ding, H. Zhou, Optimization method based extreme learning machine for classification, Neurocomputing (2010).

[20] G.-B. Huang, H. Zhou, X. Ding, R. Zhang, Extreme learning machine for regression and multi-class classification, IEEE Trans. Systems Man Cybern. B: Cybern. PP (99) (2011).

[21] Q. He, C. Du, Q. Wang, F. Zhuang, Z. Shi, A parallel incremental extreme SVM classifier, Neurocomputing 74 (2011) 2532–2540.

[22] D. Serre, Matrices: Theory and Applications, Springer, New York, 2002.

[23] E.L. Allwein, R.E. Schapire, Y. Singer, Reducing multiclass to binary: a unifying approach for margin classifiers, J. Mach. Learn. Res. (2001) 113–141.

[24] A.F. Smeaton, P. Over, W. Kraaij, Evaluation campaigns and TRECVid, in: Proceedings of MIR, 2006.

[25] G. Cormack, T. Lynam. Validity and power of T-test for comparing map and gmap, in: SIGIR, 2007, pp. 753–754.

[26] B. Lu, G. Wang, X. Gong, Multi-information fusion for uncertain semantic representations of videos, CIKM (2010) 1609–1612.

[27] A. Yanagawa, S.-F. Chang, L. Kennedy, W. Hsu, Columbia University's Baseline Detectors for 374 lscom Semantic Visual Concepts, Columbia University ADVENT Technical Report #222-2006-8, March 2007.

[28] Y.-G. Jiang, C.-W. Ngo, J. Yang, Towards optimal bag-of-features for object categorization and semantic video retrieval, in: CIVR, 2007.

[29] Q. Liu, Q. He, Z. Shi, Extreme support vector machine classifier, PAKDD, 2008, pp. 222–233.

Bo Lu received the B.S. degree and M.S. degree in computer science from Northeastern University, China in 2006 and 2008 respectively. Currently, he is a Ph.D. candidate at computer science department of the Northeastern University. His main research interests are concept-based video retrieval, semantic concept detection and cross-media retrieval.

Guoren Wang is a lecturer at the Northeastern University, China. He received his B.E., M.E. and Ph.D. degrees from the Northeastern University in 1988, 1991, 1996, respectively. His research interests include machine learning, data mining, data management, bioinformatics and multimedia technology. He has published more than 80 papers in international conferences and journals.

Ye Yuan received the B.S., M.S. and Ph.D. degrees in computer science from the Northeastern University, China, in 2004, 2007 and 2011, respectively. He is now an associate professor with the College of Information Science and Engineering in Northeastern University, China. His research interests include graph databases, probabilistic databases, data privacy-preserving and cloud computing.

Dong Han, born in 1981, is an engineer of National Marine Data And Information Service. His research interests include high performance computing, parallel computing, Internet services and cloud computing.