Labeling Images by Integrating Sparse Multiple Distance Learning and Semantic Context Modeling

Chuanjun Ji[1], Xiangdong Zhou[1], Lan Lin[2], and Weidong Yang[1]

[1] School of Computer Science, Fudan University, China
{10210240023,xdzhou,wdyang}@fudan.edu.cn
[2] School of Electronics and Information, Tongji University, China
linlan@tongji.edu.cn

Abstract. Recent progress on Automatic Image Annotation (AIA) is achieved by either exploiting low level visual features or high level semantic context. Integrating these two paradigms to further leverage the performance of AIA is promising. However, very few previous works have studied this issue in a unified framework. In this paper, we propose a unified model based on Conditional Random Fields (CRF), which establishes tight interaction between visual features and semantic context. In particular, Kernelized Logistic Regression (KLR) with multiple visual distance learning is embedded into the CRF framework. We introduce L_1 and L_2 regularization terms into the unified learning process for the distance learning and the parameters penalty respectively. The experiments are conducted on two benchmarks: Corel and TRECVID-2005 data sets for evaluation. The experimental results show that, compared with the state-of-the-art methods, the unified model achieves significant improvement on annotation performance and shows more robustness with increasing number of various visual features.

Keywords: Automatic Image Annotation, multiple distance learning, semantic context, alternating optimization.

1 Introduction

Automatic Image Annotation (AIA) has been an appealing research topic for almost a decade. The challenge originates from the so called "semantic gap", namely the mismatch between image semantics and visual perception. A great deal of research efforts have been devoted to bridge the semantic gap. Both low level visual features and high level semantics are explored in previous literatures [1–10].

In recent years, most impressive works of AIA can be categorized into two general classes. The first class is exploring visual feature learning techniques, such as feature selection [11], which combines multiple visual features to enhance annotation performance. TagProp [12] obtains competitive result by using multiple similarity measurements learning. The second class is the semantic context

A. Fitzgibbon et al. (Eds.): ECCV 2012, Part IV, LNCS 7575, pp. 688–701, 2012.
© Springer-Verlag Berlin Heidelberg 2012

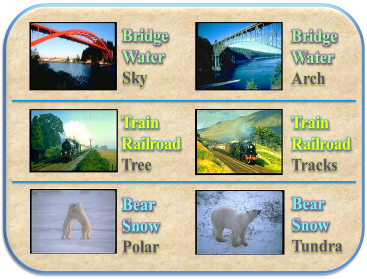


Fig. 1. Illustrative images with annotations for semantic context from Corel dataset

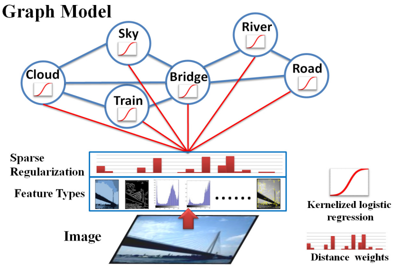


Fig. 2. Framework of the proposed Kernelized Conditional Random Fields

modeling technique [13–16]. In the scenario of AIA, semantic context refers to contextual relationships between concepts that co-occur frequently. For example, “bridge” and “water” often appear in the same image. Intuitively, for images annotated with “bridge”, it is more likely to observe “water”. Some illustrative images with human annotated keywords from [1] are presented in Figure 1. Probabilistic graphical model is adopted for semantic context modeling to boost the performance of AIA [13]. Integrating these two paradigms to further leverage the performance of AIA seems very promising. However, very few previous works have studied this issue in a unified framework.

In this paper, we propose the Kernelized Conditional Random Fields (KCRF), a unified model that integrates semantic context modeling and sparse multiple distance learning with tight interaction between them. To the best of our knowledge, our work is the first attempt to integrate these two paradigms in a unified framework for AIA. Within the unified framework, semantic contextual information is directly utilized in learning the optimal multiple feature combination, while at the same time the visual feature combination yields powerful support for modeling the semantic context.

Our KCRF model is built on semantic level to capture the relationships between semantic keywords. Figure 2 illustrates the graph structure and framework of our model. In the graph model, sites (nodes) represent concepts and edges refer to the interactions between them. To explore multiple visual feature learning, we introduce KLR [17] into the site potential. Our kernel function is based on a weighted sum of distances of multiple visual features. The parameter set of our unified model is made up of distance weights (visual parameters) and CRF parameters (semantic context parameters). Different from previous layered approaches [11, 18] that separate feature learning from image labeling, our multiple distance learning and CRF parameter estimation are conducted simultaneously subjecting to one unified object function, resulting in close interactions between the two paradigms. A pairwise L_1 and L_2 regularization term is introduced into the unified object function. Specifically, we impose L_1 regularization on the distance weight vector to obtain sparse distance combination, which makes our model more robust when dealing with increasing number of visual features. On the other hand, the semantic context parameters are penalized by L_2

regularization. We use an alternating optimization approach to estimate the optimal distance weights and CRF parameters iteratively.

To evaluate our model, we conduct experiments on Corel [1] and TRECVID-2005 datasets. Comparing with the state-of-the-art approaches, such as non-contextual methods and semantic context modeling methods, our model achieves the best performance on these two datasets with significant improvement over the others. Particularly, the experimental results show that, with increasing number of visual features, our model is more robust.

The rest of the paper is organized as follows: Section 2 reviews some related work. Section 3 presents the model setting. Section 4 and Section 5 detail the alternating parameter estimation and model inference respectively. Section 6 presents the experiment setup, and Section 7 provides the experimental results. Section 8 concludes the paper.

2 Related Work

Most of the previous AIA work[19, 2, 3] can be considered as propagating semantic concepts from training images to unlabeled images based on visual similarity. This idea is further developed by JEC [11] and TagProp [12]. Both methods focus on exploring optimal combination of multiple distances based on K-nearest neighbor (KNN) technique. In [11], the authors also tried to introduce L_1 regularization for feature selection in logistic regression. However, due to the separation of feature learning from image labeling, the logistic regression model does not outperform the JEC model using equally weighted combination of various distances. Subsequently, TagProp [12] adopts metric learning in KNN and gives out more competitive result.

Another remarkable technique is semantic context modeling. Feng and Manmatha [15] use Markov Random Fields (MRF) and propose a framework for image and video retrieval using discrete image features. Xiang et al. [13] adapted MRF for semantic context modeling in AIA. Song et al. [16] propose the Contextualized Support Vector Machine, which employs contextual information to adjust the classification hyperplane.

Considering the effectiveness of semantic context modeling technique and optimal combination of visual features, it is a rational attempt to integrate them into one consistent framework to achieve better performance. MMCRF [18] tries to make use of multiple visual features under Conditional Random Fields framework, but the feature weights are learned independently from the image labeling. Wang et al. [20] propose a Bi-relational Graph (BG) that combines the data graph connecting images and the label graph connecting concepts through label assignments. Different from previous work, our model integrates semantic context modeling and sparse multiple distance learning by using Kernel Logistic Regression in CRF framework. Rather than resorting to a layered approach as in [11], our sparse multiple distance learning and CRF parameter estimation are conducted simultaneously subjecting to one unified object function.

3 Kernelized Conditional Random Fields

In this section we present our Kernelized Conditional Random Fields model. Detailed description of the kernelized site potential and edge potential is described subsequently.

3.1 General Conditional Random Fields

Conditional Random Fields (CRF) [21] uses discriminative models for the nodes and the interactions between nodes. Let $G = (S, E)$ be a graph with site set $S = \{1, 2, ..., m\}$ and edge set E. Let $\mathbf{y} = \{y_1, y_2, ..., y_m\}$ be a set of random variables indexed by S, and $\mathbf{x} \in \chi$ be the feature vector of observed data. Then $(\mathbf{y}, \mathbf{x})$ is said to be a conditional random field if, when conditioned on $\mathbf{x}$, the random variable y_i obey the Markov property with respect to the graph: $P(y_i|\mathbf{x}, \mathbf{y}_{S-\{i\}}) = P(y_i|\mathbf{x}, \mathbf{y}_{\mathcal{N}_i})$, where $S - \{i\}$ is the set of all nodes in G except node i, $\mathcal{N}_i$ is the set of neighbors of node i in G, and $\mathbf{y}_\Omega$ represents the set of labels on nodes in the set Ω. The conditional distribution over the labels $\mathbf{y}$ given $\mathbf{x}$ is defined as,

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z} exp\bigg(\sum_{i \in S} A(y_i, \mathbf{x}) + \sum_{i \in S} \sum_{j \in N_i} I(y_i, y_j, \mathbf{x}) \bigg), \tag{1}$$

where Z is a normalizing constant called the partition function, and -A and -I are the site potential and edge potential respectively. Notice that in this paper we only consider cliques of order up to two.

3.2 Kernelized Site Potential

AIA can be considered as a binary classification problem on each site of the CRF model, i.e., $y_i \in \{-1, +1\}$ represents the absence/presence of the i^{th} concept. Hence we model site potential using a local discriminative classifier which outputs the probability of label y_i conditioned on the observation $\mathbf{x}$ on site i ignoring its neighboring sites. In order to facilitate the use of multiple image features within the context modeling framework, we employ kernelized logistic regression (KLR) [17], the nonlinear kernelized variant of logistic regression, to model the local class posterior. Given training set $\mathcal{T} = \{(\mathbf{x}^n, \mathbf{y}^n)\}_{n=1}^N$, the posterior of label y_i is defined as,

$$P(y_i|\mathbf{x}, \boldsymbol{\alpha}_i) = \frac{1}{1 + exp(-y_i f(\mathbf{x}, \boldsymbol{\alpha}_i))}, \tag{2}$$

where

$$f(\mathbf{x}, \boldsymbol{\alpha}_i) = \sum_{m=1}^{N} \alpha_i^m K(\mathbf{x}, \mathbf{x}^m), \tag{3}$$

N is the number of training images, $\boldsymbol{\alpha}_i = (\alpha_i^1, \alpha_i^2, ..., \alpha_i^N)^T$ is the parameter for site i and kernel K is the dot product matrix in a feature space. The construction of kernel will be explained in section 3.4. Finally the site potential is modeled as,

$$A(y_i, \mathbf{x}) = u_i log(P(y_i|\mathbf{x}, \boldsymbol{\alpha}_i)) = u_i log(\frac{1}{1 + exp(-y_i f(\mathbf{x}, \boldsymbol{\alpha}_i))}), \tag{4}$$

where u_i is the parameter controlling the contribution of site potential to the overall conditional distribution. Larger value of u_i indicates stronger effect of site potential. We use a spherical Gaussian prior with expectation value 1 for u_i, which will be described later. Note that the logarithm transformation ensures that our model degenerates into KLR if $u_i = 1$ and the edge potential in Eq.1 is set to zero.

3.3 Edge Potential

Using a linear discriminative model, we define edge potential as,

$$I(y_i, y_j, \mathbf{x}) = v_{ij} y_i y_j P(y_j|\mathbf{x}), \tag{5}$$

where v_{ij} is the parameter on edge (i, j) to be estimated, and $P(y_j|\mathbf{x})$ is the conditional probability of label y_j given observation $\mathbf{x}$. The edge potential is designed to favor identical labels at a pair of sites. When $v_{ij} > 0$, equal values of y_i and y_j will raise the conditional probability Eq.1 with confidence $P(y_j|\mathbf{x})$, while different value will cause punishment. In our experiment we use kernel logistic regression [17] described in Section 3.2 to generate $P(y_j|\mathbf{x})$ before the training procedure of our model. For each label $y_j (j = 1, ..., m)$, a KLR model is learned and then used to obtain $P(y_j|\mathbf{x})$. Hence, $P(y_j|\mathbf{x})$ can be regarded as a constant in Eq.5.

3.4 Kernel Construction

Through the use of Kernel our model is able to utilize multiple visual features yielding stronger support to capture semantics. Specifically, a Gaussian radial basis kernel is used on distance metric,

$$K(\mathbf{x}, \mathbf{x}') = exp(-d_{\mathbf{w}}(\mathbf{x}, \mathbf{x}')/2\sigma^2), \tag{6}$$

where σ is the width of the Gaussian kernel. The distance metric $d_w(\mathbf{x}, \mathbf{x}')$ is defined as a weighted sum of distances of image $\mathbf{x}$ and $\mathbf{x}'$ on different features,

$$d_{\mathbf{w}}(\mathbf{x}, \mathbf{x}') = \sum_{t=1}^{T} w_t d_t(\mathbf{x}, \mathbf{x}'), \tag{7}$$

where T denotes the number of features, $d_t(\mathbf{x}, \mathbf{x}')$ is the distance on the t^{th} feature, and $\mathbf{w} = (w_1, w_2, ..., w_T)$ is the feature weight vector. A larger value of w_t indicates higher importance of the corresponding feature, whereas a non-relevant feature will be assigned with zero value. As a result, the whole parameter set of our unified model consists of two parts, i.e., the conventional CRF parameters $\{(\boldsymbol{\alpha}_i, u_i, v_{ij})_{j \in N_i}\}_{i \in S}$ on sites and edges, and the feature weight vector $\mathbf{w}$.

3.5 Concept Graph

The concept graph of our model is constructed based on concept co-occurrence in the training set $\mathcal{T} = \{(\mathbf{x}^n, \mathbf{y}^n)\}_{n=1}^N$, where $\mathbf{x}^n$ denotes the n^{th} image, $\mathbf{y}^n = (y_1^n, y_2^n, ..., y_m^n)$ is the corresponding label vector with $y_i^n \in \{-1, +1\}$ indicating the absence or presence of the i^{th} concept, and N is the size of the training set. If two keywords appear in the same training image, they are treated as associated, and an edge between them is added to the graph $G = (S, E)$. Accordingly, the neighborhood of site i is defined as $\mathcal{N}_i = \{j | j \in S \wedge (i, j) \in E\}$. We extract a subgraph from G for every site to capture the semantic relationship more precisely. The subgraph contains only the site in concern and its neighboring sites as well as all the edges connecting them.

4 Alternating Parameter Estimation

Maximum likelihood is a widely used approach for CRF parameter estimation. But the computation of the partition function in Eq.1 is a generally NP-hard problem. To avoid this, we resort to the pseudo-likelihood scheme, which uses a factored approximation on every site such that

$$P(\mathbf{y}|\mathbf{x}) \approx \prod_{i \in S} P(y_i | \mathbf{y}_{\mathcal{N}_i}, \mathbf{x}) = \prod_{i \in S} \frac{1}{Z_i} exp\Big(A(y_i, \mathbf{x}) + \sum_{j \in \mathcal{N}_i} I(y_i, y_j, \mathbf{x}) \Big). \tag{8}$$

Then the negative log pseudo-likelihood on the training set $\mathcal{T}$ is defined as,

$$L = -\sum_{n=1}^{N} \sum_{i \in S} \Big\{ u_i F(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i) + \sum_{j \in \mathcal{N}_i} v_{ij} y_i^n y_j^n P(y_j^n | \mathbf{x}^n) - log Z_i^n \Big\} + R_{CRF} + R_{\mathbf{w}}, \tag{9}$$

where $F(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i)$ is defined as,

$$F(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i) = log(1/\{1 + exp(-f(\mathbf{x}^n, \boldsymbol{\alpha}_i) y_i^n)\}), \tag{10}$$

where $f(\mathbf{x}^n, \boldsymbol{\alpha}_i)$ is defined in Eq.3. The partition function for site i on the n^{th} observation is,

$$Z_i^n = \sum_{y_i^n} z(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i), \tag{11}$$

where $z(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i)$ is defined as

$$z(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i) = exp\{u_i F(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i) + \sum_{j \in \mathcal{N}_i} v_{ij} y_i^n y_j^n P(y_j^n | \mathbf{x}^n)\}. \tag{12}$$

$R_{\mathbf{w}}$ and R_{CRF} are pairwise regularization terms on feature weight $\mathbf{w}$ and the CRF parameters $\{(\boldsymbol{\alpha}_i, u_i, v_{ij})_{j \in N_i}\}_{i \in S}$ respectively. As these two parts of parameters have different effect on our model, we impose different kinds of penalty on them. Specifically, to prevent our AIA model from overfitting, we use L_2 regularization for R_{CRF}. L_1 regularization is adopted for $R_{\mathbf{w}}$ to perform sparse multiple distance learning, which encourages non-relevant feature's weight to be zero.

4.1 Alternating Parameter Estimation Procedure

An alternating procedure is proposed for parameter estimation. In CRF parameter estimation stage, the algorithm fixes $\mathbf{w}$ and optimizes $(\boldsymbol{\alpha}_i, u_i, v_{ij})_{j\in N_i}$, while in the sparse multiple distance learning stage, with fixed $(\boldsymbol{\alpha}_i, u_i, v_{ij})_{j\in N_i}$, it searches for the optimal $\mathbf{w}$. At each stage of the algorithm, the regularization term of fixed parameters is omitted, as it remains constant through the optimization process. Consequently, the object functions of each stage differ slightly with regularization terms. Detailed description of the object functions will be given in subsequent sections. Before the training process, for each site i, we build a training set $T_i = \{(\mathbf{x}^n, \mathbf{y}^n)\}_{n=1}^{N_i}$ from the original training set T by randomly selecting more balanced positive and negative samples. In our experiments the parameter estimation procedure converge rapidly after two or three times of parameters update alternations.

4.2 CRF Parameter Estimation

To optimize $\{(\boldsymbol{\alpha}_i, u_i, v_{ij})_{j\in N_i}\}_{i\in S}$, we fix $\mathbf{w}$ and omit the corresponding regularization. The estimation task is then reduced to the same problem as learning CRF parameters. Since there are no shared parameters among all sites, $(\boldsymbol{\alpha}_i, u_i, v_{ij})_{j\in N_i}$ can be trained per site. The negative log pseudo-likelihood of site i is,

$$L_i = -\sum_{n=1}^{N}\left\{u_i F(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i) + \sum_{j\in\mathcal{N}_i} v_{ij} y_i^n y_j^n P(y_j^n|\mathbf{x}^n) - logZ_i^n\right\} + R_{CRF}^i. \quad (13)$$

In practice, edge parameters tend to be overestimated that we need to penalize them more. Hence we introduce piecewise L_2 regularization terms on $\boldsymbol{\alpha}_i$, u_i and v_{ij} respectively,

$$R_{CRF}^i = \frac{\lambda_1}{2}\boldsymbol{\alpha}_i^T K \boldsymbol{\alpha}_i + \frac{\lambda_2}{2}\|u_i - 1\|^2 + \frac{\lambda_3}{2}\sum_{j\in\mathcal{N}_i}\|v_{ij}\|^2, \quad (14)$$

where K is the kernel matrix calculated using Eq.6 on the Training set T, λ_1, λ_2 and λ_3 are constants controlling the strength of the penalty, which are chosen empirically. Notice that the regularization term on $\boldsymbol{\alpha}_i$ is the same as KLR. The regularization for u_i forces it to stay around 1. The derivatives of Eq.13 with respect to $\boldsymbol{\alpha}_i$ equals to

$$\frac{\partial L_i}{\partial \boldsymbol{\alpha}_i} = -u_i K \mathbf{M}_i + \lambda_1 K \boldsymbol{\alpha}_i, \quad (15)$$

where $\mathbf{M}_i = (M_i^1, M_i^2, ..., M_i^N)^T$ is a coefficient vector with each of its component defined as

$$M_i^n = \frac{y_i^n}{1 + exp(y_i^n f(\mathbf{x}^n, \boldsymbol{\alpha}_i))} - \frac{1}{Z_i^n}\sum_{y_i^n} z(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i)\frac{y_i^n}{1 + exp(y_i^n f(\mathbf{x}^n, \boldsymbol{\alpha}_i))}. \quad (16)$$

The derivatives of Eq.13 with respect to u_i is

$$\frac{\partial L_i}{\partial u_i} = -\sum_{n=1}^{N} \left\{ F(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i) - \frac{1}{Z_i^n} \sum_{y_i^n} z(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i) F(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i) \right\} + \lambda_2(u_i - 1). \tag{17}$$

By differentiating Eq.13 with respect to v_{ij}, we will get

$$\frac{\partial L_i}{\partial v_{ij}} = -\sum_{n=1}^{N} \left\{ y_j^n P(y_j^n, \mathbf{x}^n)(y_i^n - \frac{1}{Z_i^n} \sum_{y_i^n} y_i^n z(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i)) \right\} + \lambda_3 v_{ij}. \tag{18}$$

To minimize Eq.13 we set its derivatives Eq.15, Eq.17 and Eq.18 to zero. Eq.13 is concave when λ_1, λ_2 and λ_3 are given and can be easily minimized using a projected gradient algorithm.

4.3 Sparse Multiple Distance Learning

At this stage we fix the CRF parameters and optimize the feature weight vector $\mathbf{w}$. Regularization term on CRF parameters is left out. We penalize $\mathbf{w}$ with L_1 regularization. The object function becomes,

$$L_{\mathbf{w}} = -\sum_{n=1}^{N} \sum_{i \in S} \left\{ u_i F(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i) + \sum_{j \in \mathcal{N}_i} v_{ij} y_i^n y_j^n P(y_j^n | \mathbf{x}^n) - log Z_i^n \right\} + C \sum_{t=1}^{T} |w_t|, \tag{19}$$

where C is the coefficient controlling the level of sparsity of $\mathbf{w}$. In practice it is chosen empirically. As the absolute value function is not differentiable at the zero value point, solving optimization problem Eq.19 is harder than solving differentiable optimization problems. Here we take the sub-gradient [22] of the second term in Eq.19 with respect to w_t at zero,

$$\frac{\partial L_{\mathbf{w}}}{\partial w_t} = -\sum_{n=1}^{N} \sum_{i \in S} \left\{ u_i \frac{y_i^n g(\mathbf{x}^n, \boldsymbol{\alpha}_i)}{1 + exp(y_i^n f(\mathbf{x}^n, \boldsymbol{\alpha}_i))} - \frac{1}{Z_i^n} \frac{\partial Z_i^n}{\partial w_t} \right\} + C\text{sign}(w_t), \tag{20}$$

where $\text{sign}(w_t) = 1$ if $w_t > 0$, $\text{sign}(w_t) = -1$ if $w_t < 0$, and $\text{sign}(w_t) = 0$ if $w_t = 0$, and $g(\mathbf{x}^n, \boldsymbol{\alpha}_i)$ is defined as,

$$g(\mathbf{x}^n, \boldsymbol{\alpha}_i) = \sum_{m=1}^{N} \alpha_i^m K(\mathbf{x}^n, \mathbf{x}^m)(-d_t(\mathbf{x}^n, \mathbf{x}^m)/2\sigma^2), \tag{21}$$

and the derivative of Z_i^n in Eq.20 is,

$$\frac{\partial Z_i^n}{\partial w_t} = \sum_{y_i^n} z(y_i^n, \mathbf{x}^n, \boldsymbol{\alpha}_i) u_i g(\mathbf{x}^n, \boldsymbol{\alpha}_i)(y_i^n - \frac{1}{1 + exp(-y_i^n f(\mathbf{x}^n, \boldsymbol{\alpha}_i))}). \tag{22}$$

Using the method in [22], we compute the pseudo-gradient of the L1 penalty to the extent that it does not change its sign. The limited memory BFGS algorithm is adopted to obtain the optimization of the weight parameters.

5 Model Inference

The inference problem of KCRF is to find the optimal label configuration $\mathbf{y}$ given an image $\mathbf{x}$:

$$\mathbf{y}^* \leftarrow \arg\max_{\mathbf{y}} P(\mathbf{y}|\mathbf{x}), \tag{23}$$

where $P(\mathbf{y}|\mathbf{x})$ is defined in Eq.1. The iterative conditional modes (ICM) algorithm is employed in our model. In the $(k+1)^{th}$ iteration, given the observation $\mathbf{x}$ and labels on neighboring sites $y^{(k)}_{\mathcal{N}_i}$ obtained in the last iteration, the algorithm sequentially updates each $y^{(k)}_i$ to $y^{(k+1)}_i$ that yields maximal conditional probability $P(y_i|\mathbf{y}^{(k)}_{\mathcal{N}_i}, \mathbf{x})$ defined in Eq.8. The update rule can be written as follows

$$y^{(k+1)}_i = \begin{cases} +1, \text{if } P(y_i = 1|\mathbf{y}^{(k)}_{\mathcal{N}_i}, \mathbf{x}) > P(y_i = -1|\mathbf{y}^{(k)}_{\mathcal{N}_i}, \mathbf{x}) \\ -1, \text{otherwise.} \end{cases} \tag{24}$$

The ICM algorithm starts with the initial configuration that all labels are set to be -1 and runs until convergence when two label vectors of consecutive iterations are the same. If it does not converge after 10 iterations, the process will be stopped. Ultimately it outputs the approximate result of the most probable label configuration of the observation.

6 Experiment Setup

6.1 Experimental Datasets

Our experiments are conducted on two commonly used datasets: **Corel 5k Dataset:** [1] is an important benchmark for AIA performance evaluation. It contains 5000 images, where 500 of them are used for testing and the rest for training. The whole vocabulary consists of 260 unique words with each image annotated with 1-5 keywords; **TRECVID-2005 Dataset** contains about 108 hours broadcast news, which can well represents the real world scenario. A total of 69,901 keyframes are extracted from these videos. It consists of 39 keywords. For computational efficiency, we select training images from 90 videos and testing images from the other 47 videos. For each keyword (concept), no more than 500 and 100 positive samples for training and testing respectively are included. Finally 6,657 keyframes are used for training and 1,748 keyframes for testing.

6.2 Feature Extraction

22 visual features are utilized in the experiments, where 15 feature provided by [12] are included. Apart from these features, we also extract Texture Co-occurrence, Scalable Color, HarrWavelet, Edge Histogram, Color Moments, Color Layout, and Color Correlogram according to MPEG7. All features except Gist [23] are L1-normalized. Following previous work on distance calculation, we use L2 metric for Gist, L1 for color histograms and $\chi 2$ for the rest.

6.3 Evaluation Measurements

For AIA performance evaluation, we use recall, precision and F1 measure. For a given query word w, let $|W_G|$ be the number of images with label w in the test set, $|W_M|$ be the number of annotated images by our model with the same label, then recall, precision and F1 are defined as $recall = \frac{|W_G \bigcap W_M|}{|W_G|}$, $precision = \frac{|W_C \bigcap W_M|}{|W_M|}$ and $F1 = \frac{2 \times recall \times precision}{recall + precision}$. We compute recall and precision for each keyword and then average them to measure the overall annotation performance. $F1$ is calculated with the derived mean recall and precision.

7 Experimental Results and Discussions

7.1 Performance Evaluation on Corel

In this section we evaluate the annotation performance of our method. TagProp [12] is chosen for comparison due to its state-of-the-art performance and adopting a metric learning approach. The code we use is provided by the authors. Different from TagProp, KCRF's multiple distance learning is embedded with semantic context, thus the resulted distance combination is expected to capture semantics more precisely. We conduct 9 rounds of experiments, where we start with 14 visual features and add 1 new feature incrementally in each subsequent round until all the 22 features are used in the last round. The F1 of all the 9 round experiments are given in Figure 3.

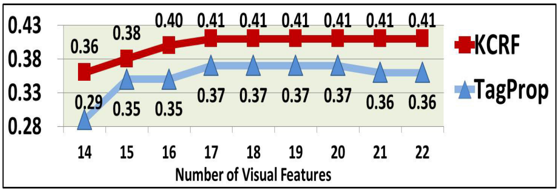


Fig. 3. F1 measure comparisons between KCRF and TagProp on Corel

It shows that KCRF outperforms TagProp in all cases, achieving the highest improvement of 24.1% in F1 score when 14 features are used, where KCRF gets 0.36 while TagProp gets 0.29. Annotation accuracy increases from 14 features to 17 features are observed for both models, while KCRF is more stable producing a smoother F1 score line. KCRF reaches the best F1 score of 0.41 with 17 features, leading to an improvement of 10.8% over TagProp, which also reaches its best F1 score of 0.37. F1 of KCRF remains the same afterward. But for TagProp model, performance decrease occurs when more than 20 features are used. The reason is that, the optimality of the distance weights is not guaranteed in TagProp, because it directly sets negative weight value to 0 to derive non-negative weight vector [12]. Unlike TagProp, KCRF introduces L_1 regularization to ensures sparsity of weight vector. Thus KCRF has higher stability with increasing number of features.

Table 1. Performance comparisons between KCRF and TagProp on Corel dataset. N+, Length, Recall, Precision, F1 and Zero-weight denote the number of keywords with non-zero recall value, average annotation length, average recall, average precision, f1 score and number of features with zero weight respectively.

Models	TagProp-14	KCRF-14	TagProp-18	KCRF-18	TagProp-22	KCRF-22
N+	140	183	160	190	158	189
Length	5	5.2	5	4.9	5	5.0
Recall	0.33	0.41	0.42	0.47	0.42	**0.48**
Precision	0.26	0.33	0.33	0.36	0.32	**0.36**
F1	0.29	0.36	0.37	0.41	0.36	**0.41**
Zero-weight	2	6	9	9	10	10

We present some detailed statistics of 3 rounds of experiments in Table 1. For the limit of page space, we cannot give out all results. Note that in Table 1, the suffixes "-14", "-18" and "-22" in the model name denote the number of features it uses. "KCRF-22" gives out the highest precision of 0.48 and the highest recall of 0.36.

7.2 Evaluation of the Unified Model

In this experiment we will clarify that, the performance improvements of KCRF given in previous sections are brought by integration of context modeling and multiple distance learning, rather than by either one of them individually. Thus we compare KCRF to these two separate methods: First, the candidate for multiple distance learning is obtained by removing context modeling from KCRF. Specifically, we set the edge potentials to 0 and it becomes Kernel Logistic Regression (KLR) with sparse multiple distance learning. We use KLR-l_1 to refer to it in following sections. For KLR-l_1, original KLR parameters and distance weights are also estimated in an alternating fashion. Second, sparse multiple distance learning is removed from our model, and we get the conventional Conditional Random Fields (CRF) as a representative for context modeling. The only difference between CRF and KCRF is the absence of multiple distance learning. Distances of different features are combined with equal weight one for CRF. The same distance metrics and kernel function are used for KCRF, KLR-l_1 and CRF. Experiment is conducted on the Corel dataset. Here all the 22 features are used. The annotation length of KLR-l_1 is fixed to be 5, while CRF and KCRF can decide the length automatically.

Experimental results are shown in Table 2. It can be observed that KCRF gives out significant performance superiority over KLR-l_1 and CRF. Specifically, the recall, precision and F1 for our unified model are 0.48, 0.36 and 0.41 respectively. It outperforms KLR-l_1 by 24% and CRF by 13.9% in F1. We also provide comparisons between distance weights of 22 features learned by KCRF and KLR-l_1 in Figure 4. From the figure, KCRF generates more sparse weight vector than KLR-l_1 and achieves better performance. It well demonstrates that the proposed unified model is able to find the optimal distances combination and

Table 2. Performance comparison with KLR-l_1 and CRF on Corel dataset

Models	KLR-l_1	CRF	KCRF
N+	157	166	**189**
Length	5	5.6	5.0
Recall	0.37	0.41	**0.48**
Precision	0.31	0.32	**0.36**
F1	0.33	0.36	**0.41**
zero-weight	6	0	10

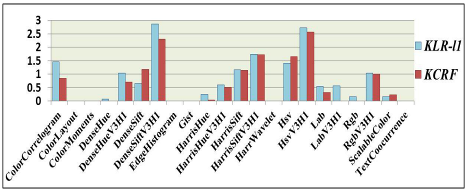
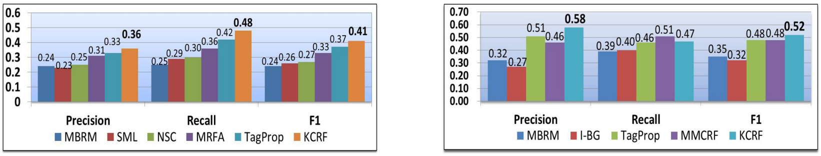


Fig. 4. Feature Weights of 22 Features Produced by KCRF and KLR-l_1

achieves better performance, which is not obtainable when using only the sparse multiple distance learning. Hence, the integration of sparse multiple distance learning and context modeling has significant advantages over the separated methods.

7.3 Performance Comparison on Corel

To further evaluate KCRF, we compare it to the TagProp [12], the semantic context modeling MRFA [13], and the other AIA methods such as MBRM [19], the supervised multi-class labeling (SML) [2], and the Nearest Spanning Chain (NSC) [3]. These models are representative techniques, and some of them achieve the stat-of-the-art performance so far. Figure 5(a) gives out the experimental results.



(a) Corel

(b) Trecvid

Fig. 5. Performance Comparison with Other Methods on Corel and Trecvid Datasets

It shows that our KCRF model has the best performance with significant improvement over the others. Specifically, the average recall, precision and F1 score of KCRF are 0.48, 0.36, and 0.41, realizing improvements in F1 score of 10.8% and 24.2% over TagProp and MRFA, which give out the second and the third highest F1 of 0.37 and 0.33, respectively. Figure 6 gives some examples of annotation results generated by KCRF and the corresponding ground-truth. It shows that the annotations of our model captures the semantics of images precisely.

7.4 Performance Comparison on TRECVID-2005

As MMCRF [18] also employs multiple visual features in CRF and achieves very competitive result on this dataset, we choose it for comparison. Besides,

Corel						
Ground Truth	Grass, Cars, Tracks	People, Flowers, Street, Vendor	Bear, Polar, Snow, Tundra	Tree, Flowers, House, Garden	Water, Boats, Harbor	Flowers, tulip, sky, tree
KCRF Annotation	Grass, Cars, Tracks, Prototype	People, Flowers, Village, Vendor	Bear, Polar, Snow, Tundra	Tree, Flowers, Garden, Cottage	Water, Boats, Harbor	Flowers, tulip
Trecvid-2005						
Ground Truth	Face, Meeting, Government-Lead er, Person	Animal, Outdoor, Sky, Waterscape_Waterfr ont	Corporate-Leader, Face, Office, Person	Boatship, Outdoor, Mountain, Sky, Waterscape_Waterfront	Building, Crowd, Face, Outdoor, Person, People-Marching, Walking_Running	Car, Face, Outdoor, Person, Road, Sky, Truck
KCRF Annotation	Corporate-Leader, Face, Meeting, Government-Lead er, Person	Animal, Outdoor, Sky, Waterscape _Waterfront	Corporate-Leader, Face, Meeting, Office, Person	Boat_ship, Sky, Natural-Disaster, Outdoor, Waterscape _Waterfront	Crowd, Face, Outdoor, People-Marching, Person, Walking_Running	Car, Military, Outdoor, Person, Truck

Fig. 6. Comparisons of KCRF annotation results with ground-truth annotations on Corel dataset and TRECVID-2005 dataset

MBRM [19], Tagprop [12] and the newly proposed BG model [20] are also included for comparison. Experimental result is given in Figure 5(b). It shows that our model also outperforms all the other methods with significant improvement. Specifically, KCRF gives out the highest F1 score of 0.52, realizing an improvement of 8.3% over TagProp and MMCRF, whose F1 scores are both 0.48. KCRF also achieves the highest precision of 0.58. Some annotation examples of KCRF are given in Figure 6 compared to the ground-truth. Specially, perfect match is reported in the second keyframes.

8 Conclusion

We propose a novel Kernelized Conditional Random Fields model for AIA problem. It integrates semantic context modeling and sparse multiple distance learning in a unified framework. We conduct the experiments on the Corel dataset and the TRECVID-2005 for evaluation. The experimental results show that through integrated learning of "visual" parameters and "semantic" parameters, our model is able to leverage the annotation performance significantly. Compared to the state-of-the-art metric learning based AIA work, KCRF is more robust and achieves higher annotation accuracy, especially with a bigger feature set.

Acknowledgments. This work was supported by the Natural Science Foundation of China under Grant No.61073002 and No.60773077.

References

1. Duygulu, P., Barnard, K., de Freitas, J.F.G., Forsyth, D.: Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part IV. LNCS, vol. 2353, pp. 97–112. Springer, Heidelberg (2002)
2. Carneiro, G., Chan, A., Moreno, P., Vasconcelos, N.: Supervised learning of semantic classes for image annotation and retrieval. IEEE PAMI 29 (2007)
3. Liu, J., Li, M., Ma, W., Liu, Q., Lu, H.: An adaptive graph model for automatic image annotation. In: ACM Workshop on Multimedia Information Retrieval (2006)

4. Wang, Y., Mori, G.: Max-margin latent dirichlet allocation for image classification and annotation. In: 22nd British Machine Vision Conference, BMVC (2011)
5. Zhou, X., Wang, M., Zhang, J., Zhang, Q., Shi, B.: Automatic image annotation by an iterative approach: Incorporating keyword correlations and region matching. In: ACM Int'l Conf. Image and Video Retrieval, CIVR (2007)
6. Li, L., Li, F.: What, where and who? classifying events by scene and object recognition. In: CVPR (2007)
7. Deng, J., Dong, W., Socher, R., Li, L., Li, K., Li, F.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
8. Tang, J., Hong, R., Yan, S., Chua, T.S., Qi, G.J., Jain, R.: Image annotation by knn-sparse graph-based label propagation over noisily-tagged web images. ACM Transactions on Intelligent Systems and Technology 2 (2011)
9. Qi, G., Hua, X., Rui, Y., Tang, J., Mei, T., Zhang, H.: Correlative multi-label video annotation. In: ACM SIGMM (2007)
10. Jiang, Y.G., Dai, Q., Wang, J., Ngo, C.W., Xue, X., Chang, S.F.: Fast semantic diffusion for large-scale context-based image and video annotation. IEEE Transactions on Image Processing 21 (2012)
11. Makadia, A., Pavlovic, V., Kumar, S.: A New Baseline for Image Annotation. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 316–329. Springer, Heidelberg (2008)
12. Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In: ICCV (2009)
13. Xiang, Y., Zhou, X., Chua, T., Ngo, C.: A revisit of generative model for automatic image annotation using markov random fields. In: CVPR (2009)
14. Rasiwasia, N., Vasconcelos, N.: Holistic context modeling using semantic co-occurences. In: CVPR (2009)
15. Feng, S., Manmatha, R.: A discrete direct retrieval model for image and video retrieval. In: CIVR (2008)
16. Song, Z., Chen, Q., Huang, Z., Hua, Y., Yan, S.: Contextualizing object detection and classification. In: CVPR (2011)
17. Roth, V.: Probabilistic Discriminative Kernel Classifiers for Multi-class Problems. In: Radig, B., Florczyk, S. (eds.) DAGM 2001. LNCS, vol. 2191, pp. 246–253. Springer, Heidelberg (2001)
18. Xiang, Y., Zhou, X., Liu, Z., Chua, T., Ngo, C.: Semantic context modeling with maximal margin conditional random fields for automatic image annotation. In: CVPR (2010)
19. Feng, S., Manmatha, R., Lavrenko, V.: Multiple bernoulli relevance models for image and video annotation. In: CVPR (2004)
20. Wang, H., Huang, H., Ding, C.: Image annotation using bi-relational graph of images and semantic labels. In: CVPR (2011)
21. Lafferty, J., McCallum, A., Pereira, F.: Conditonal random fields: Probabilistic models for segmenting and labeling sequence data. In: ICML (2001)
22. Andrew, G., Gao, J.: Scalable training of l1-regularized log-linear models. In: ICML (2007)
23. Oliva, A., Torralba, A.: The role of context in object recognition. Trends in Cognitive Sciences 11(12), 520–527 (2007)