Concurrent Multiple Instance Learning for Image Categorization

[†]Guo-Jun Qi, [‡]Xian-Sheng Hua, [‡]Yong Rui, [‡]Tao Mei, [†]Jinhui Tang, [‡]Hong-Jiang Zhang
[†]Department of Automation, University of Science and Technology of China
Huang Shan Road, No.4, Hefei, Anhui, 230027, China
{qgj, jhtang}@mail.ustc.edu.cn
[‡]Microsoft Research Asia
Microsoft Research Asia, 49 Zhichun Road, Beijing, 100080, China
{xshua, yongrui, tmei, hjzhang}@microsoft.com

Abstract

We propose a new multiple instance learning (MIL) algorithm to learn image categories. Unlike existing MIL algorithms, in which the individual instances in a bag are assumed to be independent with each other, we develop concurrent tensors to explicitly model the inter-dependency between the instances to better capture image's inherent semantics. Rank-1 tensor factorization is then applied to obtain the label of each instance. Furthermore, we formulate the classification problem in the Reproducing Kernel Hilbert Space (RKHS) to extend instance label prediction to the whole feature space. Finally, a regularizer is introduced, which avoids overfitting and significantly improves learning machine's generalization capability, similar to that in SVMs. We report superior categorization performances compared with key existing approaches on both the COREL and the Caltech datasets.

1. Introduction

With the proliferation of digital photography, automatic image categorization becomes increasingly important. In this paper, we define categorization as automatic classification of images into predefined semantic concepts (categories). Before a learning machine can perform classification, it needs to be trained first, and training samples need to be accurately labeled. The labeling process can be both time consuming and error-prone [17]. Fortunately, multiple instance learning (MIL) allows for coarse labeling at the image level, instead of fine labeling at pixel/region level, which significantly improves the efficiency of image categorization. [12] [3] [19].

In the MIL framework, there are two levels of training inputs: *bags* and *instances*. A bag is composed of multiple instances. A bag is labeled positive if at least one of its instances falls within the concept, and it is labeled negative if all of its instances are negative. The efficiency of MIL lies in the fact that during training, a label is required only for a bag, not the instances in the bag. In the case of image categorization, a labeled image (e.g., a "beach" scene) is a bag, and the different regions inside the image are the instances [12]. Some of the regions are background and may not relate to "beach", but other regions, e.g., sand and sea, do relate to "beach". If we exam more carefully, we can see that sand and/or sea do not appear independently in statistics, they tend to appear simultaneously in an image of "beach" frequently. Such an *co-existence* or *concurrency* can significantly boost the belief that an instance (e.g. the sand, the sea etc.) belongs to a "beach" scene. Therefore, in this "beach" scene, there exist *order*-2 *concurrent* relationship between the sea instance (region) and the sand instance (region). Similarly, in this "beach" scene, there also exist higher-order (*order*-4) concurrent relationship between instances, e.g., sand, sea, people, and sky.

To the best of our knowledge, all the existing MIL-based image categorization algorithms assume that the instances in a bag are independent and they have not explored such concurrent relations. Although this independence assumption significantly simplifies the modeling and computing procedure, it does not take into account the hidden information encoded in the semantic linkage among instances, as we described in the above "beach" example.

To address this problem, in this paper, we propose a novel *Concurrent MIL* (ConMIL) scheme to encode the inter-dependency between instances. ConMIL has three major contributions. First, ConMIL uses concurrent tensor to model the semantic linkage between the instances. In addition, based on the concurrent tensor, rank-1 supersymmetric non-negative tensor factorization (SNTF) [10] is applied to estimate the probability of each instance being relevant to a target category. Second, ConMIL formulates the label prediction processes in a regularization

1-4244-1180-7/07/$25.00 ©2007 IEEE

framework, which avoids overfitting, and significantly improves learning machine's generalization capability, similar to that in SVMs [5]. Third, ConMIL uses *Reproducing Kernel Hilbert Space* (RKHS) to extend predicted labels to the whole feature space based on the generalized representer theorem [15] to facilitate the testing process. In the experiment section, we will show that ConMIL achieves high classification accuracy on both bags and instances, is robust to different datasets, and is computationally efficient.

The rest of the paper is organized as follows. We review related work on MIL-based image categorization in Section 2. Section 3 gives detailed description of the proposed ConMIL algorithm, including the concurrent tensor and its factorization, the kernelization framework, as well as ConMIL's interesting relationship to existing MIL algorithms. Experimental results and comparisons on both COREL and Caltech are reported in Section 4. We give concluding remarks in Section 5.

2. Related Work

In this section, we will review representative MIL-based image categorization approaches. In general, they can be divided into two paradigms according to their classification levels (bag level vs. instance level). The bag-level approaches aim at predicting the bag label directly. For example, in [7], a standard support vector machine (SVM) is used to predict bag label with so-called Multi-Instance kernels which are designed for bags. DD-SVM [4] selects a set of prototypes from the local maxima of DD function, and then a SVM was trained based on the bag features summarized by these selected prototypes. In [3], bags are embedded into a feature space defined by instances and then 1-norm SVM is applied to construct bag classifiers. However, the bag-level approaches do not try to gain insight into instance label.

The instance-level approaches first attempt to infer the hidden instance label and then to predict the bag label. For example, Yang et al. [19] proposed an Asymmetric Support Vector Machine-based MIL algorithm (ASVM-MIL) by introducing asymmetric loss function for false positives and false negative to exploit the instance label while the diverse density (DD) approach [12][20] takes a scaling and gradient search algorithm to find the prototype points in the instance space with the highest DD value. However, both of these algorithms have not considered the relationship among instances when inferring their label. Furthermore, the DD-based algorithm is computationally expensive, because it searches for globally optimal points in the feature space, and overfitting may occur for the lack of a regularization term in the DD measure. Ray et al. [14] extended the DD framework, where they seek $p(y_i = 1|B_i = \{B_{i1}, B_{i2}, \cdots, B_{in}\})$, i.e., the conditional probability of the label of the i-th bag being positive, given the instances in the bag. They use the Logistic Regression (LR) algorithm to estimate the equivalent probability for an instance, $p(y_{ij} = 1|B_{ij})$, and then a combination function $softmax$ is used to combine $p(y_{ij} = 1|B_{ij})$ in a bag to estimate $p(y_i = 1|B_i)$:

$$\begin{aligned} p(y_i = 1|B_i) &= softmax_\gamma(S_{i1}, S_{i2}, \ldots, S_{in}) \\ &= \frac{\sum_j S_{ij} \exp(\gamma \cdot S_{ij})}{\sum_j \exp(\gamma \cdot S_{ij})} \end{aligned} \quad (1)$$

where $S_{ij} = p(y_{ij} = 1|B_{ij})$. The combining function encodes the multiple instance assumption in this MIL algorithm. In [17], MILBoost is proposed to adopt MIL into the AdaBoost framework, where the combination function Integrated Segmentation and Recognition (ISR) or noisy-or is used to combine instance labels into bag labels.

To summarize, regardless being bag-based or instance-based, all the existing MIL algorithms assume that the instances independently occur in an image and they do not take into account the hidden information encoded in the semantic linkage among instances. For example, DD function is based on statistical independency assumption of all instances [12], and mi/MI-SVM [1] did not consider the concurrent relations of instances when maximizing instance and bag margins. For the bag-level MIL, DD-SVM [4] dose not select the set of prototypes based on the instance interdependencies, and MILES [3] do not investigate the semantic relations among the instances either when embedding bags into a feature space defined by these instances. To address these issues, we propose the ConMIL algorithm, which will be described in the next section.

3. The Proposed Approach - ConMIL

In Section 3.1, we illustrate that the concurrent relations among instances can be naturally described in a concurrent hypergraph. In Section 3.2, we will develop a statistical measure of the concurrent relations quantitatively which will be used as entries in the proposed concurrent tensors. Based on the concurrent tensor representation, in Section 3.3, we will use tensor factorization for label inference. Finally in Section 3.4, we will formulate the label inferring processes into a kernelization framework.

Let B_i denote the i-th bag, B_i^+ a positive bag and B_i^- a negative one. Let bag set $\mathcal{B} = \{B_i\}$, positive bag set $\mathcal{B}^+ = \{B_i^+\}$ and negative bag set $\mathcal{B}^- = \{B_i^-\}$. Let $\mathcal{I}$ denote the set of instances and $n_I = |\mathcal{I}|$ the number of all instances. An instance $I_j \in \mathcal{I}$ $(1 \leqslant j \leqslant n_I)$ is denoted I_j^+ when it is positive and is denoted by I_j^- when negative. We also denote an instance by B_{ij} to emphasize it is the j-th instance in the i-bag. Let $p(I_j)$ and $p(B_{ij})$ denote the probability of I_j and B_{ij} being a positive instance respectively. $p(B_{ij})$ is equivalent to $p(y_{ij} = 1|B_{ij})$ in Eq. (1).

3.1. Concurrent Hypergraph Representation

Figure 1 illustrates an example of concurrent hypergraph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ for the category "beach" discussed in Section

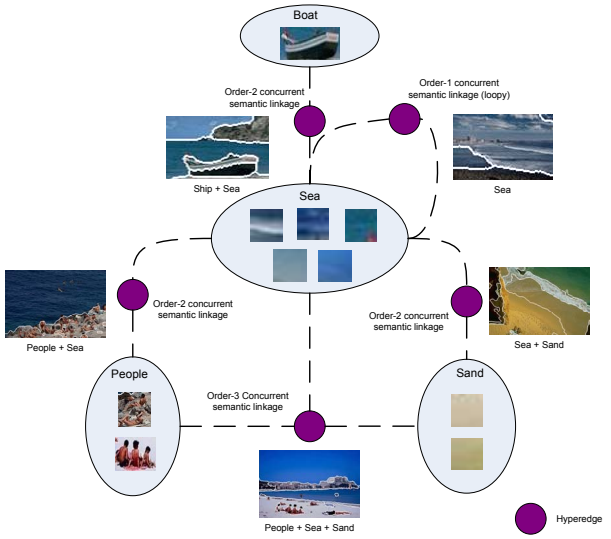

Figure 1. A concurrent hypergraph describing concurrent semantic linkage of different regions (instances) for a category "beach".

1, where $\mathcal{G}$ and $\mathcal{E}$ are the vertex and hyperedge set, respectively. As shown in Figure 1, the vertices in this hypergraph represent different instances and these instances are linked semantically by hyperedges to encode any order of concurrent relationships between instances in $\mathcal{G}$. A statistic quantity is associated with each hyperedge in $\mathcal{G}$ to measure these concurrent relationships which will be detailed in Section 3.2.

Based on the concurrent hypergraph $\mathcal{G}$, tensor and its corresponding algebra can naturally be used as a mathematical tool to represent and learn the concurrent relationship, and the tensor entries are associated with the hyperedges in $\mathcal{G}$. As to be detailed in following sections, with the tensor representation, rank-one super-symmetric non-negative tensor factorization (SNTF) [10] can then be applied to obtain $p(y_{ij} = 1|B_{ij})$, i.e., the probability of an instance B_{ij} being positive (cf. Section 2). Once the instance label is obtained, the bag label can be directly computed from the combination function (such as Eq. (1)) which encodes the MIL assumption in it.

Before we move further, we next will give a brief introduction to the tensor rank factorization. An n-order tensor $\mathcal{T}$ of dimension $[d_1] \times [d_2] \times \cdots [d_n]$, indexed by n indices $i_1, i_2, \ldots, i_n$ with $1 \leqslant i_j \leqslant d_j$, is of rank-1 if it can be expressed by the generalized outer product of n vectors: $\mathcal{T} = \boldsymbol{v}_1 \otimes \boldsymbol{v}_2 \otimes \cdots \otimes \boldsymbol{v}_n$, where $\boldsymbol{v}_i \in \mathcal{R}^{d_i}$. A tensor $\mathcal{T}$ is called super-symmetric when its entries are invariant under any permutation of their indices. For such a super-symmetric tensor, its factorization has a symmetric form: $\mathcal{T} = \boldsymbol{v}^{\otimes n} = \underbrace{\boldsymbol{v} \otimes \boldsymbol{v} \otimes \cdots \otimes \boldsymbol{v}}_{n \ \ terms}$. Some factorization algorithm has been proposed such as High-Order SVD (HOSVD) [10] which is an extent of Singular Value Decomposition (SVD), however it cannot guarantee the factorization convergence for high-order tensor. In this paper, we adopt a direct gradient descent based approach, as to be detailed in Section 3.3.

3.2. Concurrent Relations in MIL

As illustrated in Figure 1, in images labeled as a specific category (e.g. car, mountain, beach, etc.), there exists hidden information encoded in the concurrent semantic linkage among different regions (instances). This observation prompts us to incorporate these concurrent relations into the process of inferring probability $p(I_j)$. Therefore, we must firstly answer the question what is an appropriate statistic to measure such concurrent relations?

We use $p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n})$ to denote the probability of the concurrence of n instances $I_{i_1}, I_{i_2}, \cdots, I_{i_n}$ in the same bag labeled as a certain concept, where the notation "$\wedge$" means the logic operation "and". Given the bag set $\mathcal{B} = \{B_i\}$, we have the likelihood

$$\begin{aligned} & p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n}|\mathcal{B}) \\ & = \prod_i p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n}|B_i^+) \\ & \cdot \prod_j p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n}|B_j^-) \end{aligned} \tag{2}$$

Typically, the logic operation "$\wedge$" in Eq. (2) can be estimated by "min" [18], so we have

$$p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n}|B_i) = \min_k\{p(I_{i_k}|B_i)\} \tag{3}$$

Adopting a noisy-or model [12], the probability that not all points missed the target concept is

$$p(I_{i_k}|B_i^+) = p(I_{i_k}|B_{i1}^+, B_{i2}^+, \cdots) = 1 - \prod_j \left(1 - p(I_{i_k}|B_{ij}^+)\right) \tag{4}$$

and likewise

$$p(I_{i_k}|B_i^-) = p(I_{i_k}|B_{i1}^-, B_{i2}^+, \cdots) = \prod_j \left(1 - p(I_{i_k}|B_{ij}^-)\right) \tag{5}$$

Concatenating Eq. (2)∼(5) together, we have

$$\begin{aligned} & p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n}|\mathcal{B}) \\ & = \prod_i \min_k\{1 - \prod_j \left(1 - p(I_{i_k}|B_{ij}^+)\right)\} \\ & \cdot \prod_i \min_k\{\prod_j \left(1 - p(I_{i_k}|B_{ij}^-)\right)\} \end{aligned} \tag{6}$$

The causal probability of an individual instance on a potential target $p(I_{i_k}|B_{ij})$ can be modeled as related to the distance between them, that is $p(I_{i_k}|B_{ij}) = \exp(-\|B_{ij} - I_{i_k}\|^2)$.

As $p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n}|\mathcal{B})$ is the likelihood over the entire set $\mathcal{B}$ with $m = |\mathcal{B}|$ bags, and $p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n})$ is the probability that $I_{i_1}, I_{i_2}, \cdots, I_{i_n}$ occur at the same time in a positive bag while not in a negative bag, we have $[p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n})]^m = p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n}|\mathcal{B})$, then the concurrent probability can be estimated as

$$p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n}) = [p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n}|\mathcal{B})]^{\frac{1}{m}} \quad (7)$$

Consequently, $p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n})$ can be regarded a measure of n-order concurrent relations among $I_{i_1}, I_{i_2}, \cdots, I_{i_n}$.

3.3. High-Order Concurrent Tensor Inference

In this section, we represent concurrent relations in an order-n tensor form, and a rank-1 tensor factorization procedure can be utilized to derive $p(I_j)$, the probability of I_j being a positive instance. The concurrent relations measured by $p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n})$ are used as the entry of high order tensor. We name this tensor *concurrent tensor*denoted by $\mathcal{T}$ to denote this tensor. From the Eq. (6)(7), the entry of this tensor is given by

$$\begin{aligned} \mathcal{T}_{i_1,i_2,\cdots,i_n} &\triangleq p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n}) \\ &= \Big\{\prod_i \min_k\{1 - \prod_j (1 - p(I_{i_k}|B_{ij}^+))\} \cdot \\ &\quad \prod_i \min_k \Big\{\prod_j (1 - p(I_{i_k}|B_{ij}^-))\Big\}\Big\}^{\frac{1}{m}} \quad (8) \end{aligned}$$

where $1 \leqslant i_1, i_2, \ldots, i_n \leqslant n_I$. Since the bag label and the concurrent relation information have been incorporated into $\mathcal{T}$, this concurrent tensor is a supervised measure instead of an unsupervised affinity measure in other works [16].

Given the concurrent tensor $\mathcal{T}$, we wish to estimate $p(I_j)$. The desired probabilities form a nonnegative $1 \times n_I$ vector $\boldsymbol{P} = [p(I_1), p(I_2), \cdots, p(I_{n_I})]^T$, thus our goal is to find $\boldsymbol{P}$ given tensor $\mathcal{T}$.

As $p(I_{i_1} \wedge I_{i_2} \wedge \cdots \wedge I_{i_n})$ is equivalent to $\min\{p(I_{i_1}, I_{i_2}, \cdots, I_{i_n})\}$ according to logic operation " $\wedge$ ", considering the concurrent tensor definition (8), we have the following set of n_I^n equations with $1 \leqslant i_1, i_2, \cdots, i_n \leqslant n_I$:

$$\mathcal{T}_{i_1,i_2,\cdots,i_n} = \min\{p(I_{i_1}), p(I_{i_2}), \cdots, p(I_{i_n})\} \quad (9)$$

It is an over-determined problem to solve n_I unknown variables $p(I_j), 1 \leqslant j \leqslant n_I$, and it is computationally extensive to find an optimal solution to probability vector $\boldsymbol{P}$ if we exhaustively search it in n_I dimension space $\mathcal{R}^{n_I}$.

Alternatively, we can relax the non-differentiable operation "min" to a differentiable function, and then a gradient search algorithm can be adopted to give an efficient search for the optimal solution to $\boldsymbol{P}$. As discussed in [18], the logic "$\wedge$" can also been estimated by a kind of T-norm functions. The multiplication operation has been proven to be such an operator, and these two operators have the relation that the "min" operator is an upper bound of the "multiplication" operator:

$$p(I_{i_1}) \cdot p(I_{i_2}) \cdots p(I_{i_n}) \leqslant \min\{p(I_{i_1}), p(I_{i_2}), \cdots, p(I_{i_n})\} \quad (10)$$

Therefore, an alternative solution is to use "multiplication" to estimate the logic "$\wedge$"

$$\mathcal{T}_{i_1,i_2,\cdots,i_n} \doteq p(I_{i_1}) \cdot p(I_{i_2}) \cdots p(I_{i_n}) \quad (11)$$

In this form, the above set of n_I^n equations can be represented in a compact tensor form:

$$\mathcal{T} = \underbrace{\boldsymbol{P} \otimes \boldsymbol{P} \otimes \cdots \otimes \boldsymbol{P}}_{n \ terms} = \boldsymbol{P}^{\otimes n} \quad (12)$$

This equation can be translated to the fact that $\mathcal{T}$ should be a rank-1 super-symmetric tensor, and $\boldsymbol{P}$ can be calculated given the concurrent tensor $\mathcal{T}$. Eq. (12) is also an over-determined multilinear system with n_I^n equations like Eq. (11). This problem can be solved by a search for an optimal solution $\boldsymbol{P}$ to approximate the tensor $\mathcal{T}$ in light of least-squared criterion, and the obtained $\boldsymbol{P}$ can best reflect semantic linkage of instances contained in $\mathcal{T}$.

In order to find the best solution to $\boldsymbol{P}$, we consider the following least-squared problem:

$$\begin{aligned} \min_{\boldsymbol{P}} C(\boldsymbol{P}) &= \frac{1}{2}\|\mathcal{T} - \boldsymbol{P}^{\otimes n}\|_F^2 \quad (13) \\ s.t.\ \boldsymbol{P} &\geqslant 0 \end{aligned}$$

where $\|\cdot\|_F^2$ is the squared Frobenious norm as $\|\mathcal{K}\|_F^2 = \langle \mathcal{K}, \mathcal{K} \rangle = \sum_{i_1,i_2,\ldots,i_n} \mathcal{K}_{i_1,i_2,\ldots,i_n}^2$. Since the super-symmetric tensor dose not depend on the order of the indices, we can only store a single representative of each n-tuple and focus on the entries $i_1 \leqslant i_2 \leqslant \cdots \leqslant i_n$, this could save a great deal of memory to store the tensor $\mathcal{T}$.

The most direct approach is to form a gradient descent scheme. To that end, we derive the gradient function w.r.t. $\boldsymbol{P}$ at first. Following that the differential commutes with inner-product operation $\langle \cdot, \cdot \rangle$, i.e., $d\langle \mathcal{K}, \mathcal{K} \rangle = 2\langle \mathcal{K}, d\mathcal{K} \rangle$ and the identity $d(\boldsymbol{P}^{\otimes n}) = (d\boldsymbol{P}) \otimes \boldsymbol{P}^{\otimes(n-1)} + \cdots + \boldsymbol{P}^{\otimes(n-1)} \otimes (d\boldsymbol{P})$, we have

$$\begin{aligned} dC(\boldsymbol{P}) &= d\big(\frac{1}{2}\langle \mathcal{T} - \boldsymbol{P}^{\otimes n}, \mathcal{T} - \boldsymbol{P}^{\otimes n} \rangle\big) \\ &= \langle \mathcal{T} - \boldsymbol{P}^{\otimes n}, d(\mathcal{T} - \boldsymbol{P}^{\otimes n}) \rangle \\ &= \langle \boldsymbol{P}^{\otimes n} - \mathcal{T}, d(\boldsymbol{P}^{\otimes n}) \rangle \\ &= \langle \boldsymbol{P}^{\otimes n} - \mathcal{T}, (d\boldsymbol{P}) \otimes \boldsymbol{P}^{\otimes(n-1)} \\ &\quad + \cdots + \boldsymbol{P}^{\otimes(n-1)} \otimes (d\boldsymbol{P}) \rangle \quad (14) \end{aligned}$$

Then the partial derivative w.r.t. p_j (the j-th entry of $\boldsymbol{P}$) is:

$$\begin{aligned}\frac{\partial C(\boldsymbol{P})}{\partial p_j} &= \langle \boldsymbol{P}^{\otimes n} - \mathcal{T}, e_j \otimes \boldsymbol{P}^{\otimes(n-1)} + \cdots + \boldsymbol{P}^{\otimes(n-1)} \otimes e_j \rangle \\ &= \langle \boldsymbol{P}^{\otimes n}, e_j \otimes \boldsymbol{P}^{\otimes(n-1)} + \cdots + \boldsymbol{P}^{\otimes(n-1)} \otimes e_j \rangle \\ &\quad - \langle \mathcal{T}, e_j \otimes \boldsymbol{P}^{\otimes(n-1)} + \cdots + \boldsymbol{P}^{\otimes(n-1)} \otimes e_j \rangle \\ &= n \cdot P_j \cdot \|\boldsymbol{P}\|^{2(n-1)} - \sum_{r=1}^{n} \sum_{S/i_r} \mathcal{T}_{S_{i_r \leftarrow j}} \prod_{m \neq r} P_{i_m} \end{aligned} \tag{15}$$

where e_j is the standard vector $(0, 0, \ldots, 1, 0, \ldots, 0)$ with 1 in the j-th coordinate, and S represents an n-tuple index, S/i_r denotes $\{i_1, \cdots, i_{r-1}, i_{r+1}, \cdots, i_n\}$, $S_{i_r \leftarrow j}$ the set of indices S where the index i_r is replaced by j. Hence, we have the gradient function w.r.t. $\boldsymbol{P}$, that is

$$\nabla_{\boldsymbol{P}} C(\boldsymbol{P}) = \Big[\frac{\partial C(\boldsymbol{P})}{\partial p_1}, \frac{\partial C(\boldsymbol{P})}{\partial p_2}, \cdots, \frac{\partial C(\boldsymbol{P})}{\partial p_{n_I}}\Big]^T \tag{16}$$

Consequently, a direct gradient descent scheme could be applied to form an iterative algorithm of search for the best solution $\boldsymbol{P}$. However, this solution to $\boldsymbol{P}$ is limited to the available set of instances instead of the whole feature space. In the following section, we will develop an approach to extend the solution to the whole space in a natural way, i.e., find a function $p(x)$ defined on the whole feature space from RHKS to give the probability of any instance of being positive.

3.4. A Kernelization Framework

In this section, we will solve two problems. First, we extend the estimated posterior probability vector $\boldsymbol{P}$ by searching for an optimal function defined over the whole feature space on the basis of a kernelized representation of the objective problem Eq. (13). Second, in this kenelization form, a regularization term will be adopted to generate a regularized function $p(x)$ over feature space, which is able to avoid overfitting of the concurrent likelihood model and high-order concurrent tensor model.

To start, we rewrite the objective cost function in problem Eq. (13). Given the function $p(x)$, the probability vector $\boldsymbol{P}$ in Eq. (13) can be given as $\boldsymbol{P} = [p(I_1), p(I_2), \cdots, p(I_{n_I})]^T$ where $\{I_i\}_{i=1}^{n_I}$ is the training set. Therefore, the cost function in Eq. (13) can be rewritten as $C(p(x), \{I_i\}_{i=1}^{n_I}) = \frac{1}{2}\|\mathcal{T} - \boldsymbol{P}^{\otimes n}\|_F^2$. Note that, Different from Eq. (13), $C(p(x), \{I_i\}_{i=1}^{n_I})$ is defined as a function of $p(x)$ instead of vector $\boldsymbol{P}$, and this cost function will be minimized w.r.t. the function $p(x)$.

Secondly, as mentioned in 3.2, we use a multiplicative noisy-or model in the multiple instance setting, which is often sensitive to instances in negative bags for one well placed negative instance could bring DD to near zero. In addition, when the order of the concurrent tensor increases, the complexity of the tensor model also increases, which tends to overfit the concurrent likelihood in (6). To solve this issue, a regularization term $\Omega(\|p(x)\|_{\mathcal{H}})$ is introduced to control the complexity of the high-order tensor model by penalizing the RKHS norm to impose smoothness condition on possible solutions. Here $\mathcal{H}$ denotes RKHS, $\|\cdot\|_{\mathcal{H}}$ the norm in this Hilbert space, and $\Omega(\cdot)$ is a strictly monotonically increasing function. Combining the above two considerations, the final optimization problem can be written as

$$\begin{aligned}\min_{p(x)\in\mathcal{H}} F(p(x), \{I_i\}_{i=1}^{n_I}) &= C(p(x), \{I_i\}_{i=1}^{n_I}) + \lambda \cdot \Omega(\|p(x)\|_{\mathcal{H}}) \\ &= \frac{1}{2}\|\mathcal{T} - \boldsymbol{P}^{\otimes n}\|_F^2 + \lambda \cdot \Omega(\|p(x)\|_{\mathcal{H}}) \\ s.t.\ \ \boldsymbol{P} &= [p(I_1), p(I_2), \cdots, p(I_{n_I})]^T \\ & p(x) > 0 \end{aligned} \tag{17}$$

where λ is a parameter that trades off the two components.

Since the objective function $F(p(x), \{I_i\}_{i=1}^{n_I})$ is pointwise, which only depends on the value of $p(x)$ at the data points $\{I_i\}_{i=1}^{n_I}$, according to the generalized representer theorem [15], the minimizer $p^*(x)$ exists in RKHS and admits a representation of the form

$$p^*(\cdot) = \sum_{i=1}^{n_I} \alpha_i k(\cdot, I_i). \tag{18}$$

where $k(\cdot,\cdot)$ is a Mercer Kernel associated with RKHS $\mathcal{H}$.

Let $K = [k(I_i, I_j)]_{n_I \times n_I}$ denote $n_I \times n_I$ Gram matrix with a kernel function $k(I_i, I_j) = \exp\{-\frac{\|I_i - I_j\|^2}{2\sigma^2}\}$ (Gaussian Kernel) over instance features and coefficient vector $\alpha = [\alpha_1\ \alpha_2\ \cdots\ \alpha_{n_I}]^T$ in Eq.(18). Using $\Omega(\|p(x)\|_{\mathcal{H}}) = \frac{1}{2}\|p(x)\|_{\mathcal{H}}^2$ and substitute Eq.(18) into Eq. (17), we have the following optimization problem:

$$\begin{aligned}\min_{\alpha} F(\alpha) &= \frac{1}{2}\|\mathcal{T} - (K \cdot \alpha)^{\otimes n}\|_F^2 + \frac{1}{2}\lambda \alpha^T K \alpha \\ s.t.\ \ \alpha &> 0 \end{aligned} \tag{19}$$

To solve it, we derive the gradient of $F(\alpha)$ w.r.t. α:

$$\begin{aligned}\nabla_\alpha F(\alpha) &= \nabla_\alpha C(p(x), \{I_i\}_{i=1}^{n_I}) + \frac{1}{2}\lambda \nabla_\alpha (\alpha^T K \alpha) \\ &= K \cdot \nabla_{\boldsymbol{P}} C(\boldsymbol{P}) + \lambda K \cdot \alpha \end{aligned} \tag{20}$$

where $\nabla_{\boldsymbol{P}} C(\boldsymbol{P})$ is the gradient of cost function $C(p(x), \{I_i\}_{i=1}^{n_I})$ w.r.t. vector P derived in Eq. (15)(16).

With this obtained gradient, L-BFGS quasi-Newton method [11] is used to solve this optimization problem. By building up an approximation scheme through successive evaluation of the gradient in Eq. (20), L-BFGS can avoid the explicit estimation of the Hessian matrix. It has been proven L-BFGS has a fast convergence rate to learn the parameters α than traditional scaling learning algorithms.

4. Experiment

In this section, we will evaluate ConMIL along several dimensions. First, we will compare ConMIL with key existing MIL approaches in image categorization on the most

Figure 2. Three sample images (top row), corresponding segmented regions (middle row) and their probability(salience) map based on the estimated posterior probability $p(x)$ (bottom row) .

	Avg. AUC for COREL dataset
EM-DD [20]	0.775
DD-SVM [4]	0.858
ASVM-MIL [19]	0.836
ConMIL	0.916

Table 1. Average AUC for COREL 5000 dataset by EM-DD, DD-SVM, ASVM, ConMIL

widely used COREL 5000 benchmark dataset. Second, we will apply ConMIL in one of the important branches of image categorization: object class recognition, by using the standard benchmark dataset from Caltech. Third, we will analyze the influence of the concurrent tensor's order on the classification accuracy and computational cost.

4.1. Evaluation on COREL 5000

The COREL 5000 dataset has 50 semantically diverse categories, with each category containing 100 images. For the experiments, the images are first segmented using JSEG (see Fig. 2 for example segmentation results), and only regions larger than $1/25$ of original image are kept. As a result, each image contains typically less than 10 regions. A set of low-level features is extracted from each region to represent an instance, including color correlogram, color moment, region size, wavelet texture and shape (normalized inertia of order 1, 2, 3)[4].

During the experiments, images within each category are randomly partitioned into two halves to form the training and the testing sets. To determine the parameters σ^2 (Gaussian Kernel radius) and λ in Eq.(17), we conduct a twofold cross-validation on the training set. We choose σ^2 from 1 to 20 with step size 2, and λ from 0.1, 1, 10, 100. The pair of parameters that achieves the best performance

Algorithms	Airplanes	Cars	Faces	Motorbikes
ConMIL	**0.992**	**0.984**	0.976	**0.98**7
MILES [3]	0.980	0.945	**0.995**	0.967
Fergus et al. [6]	0.902	0.903	0.964	0.925
Opelt et al. [13]	0.889	0.901	0.935	0.922
Bar-Hillel et al.[2]	0.897	0.977	0.917	0.931

Table 2. Comparison of object recognition performance using ConMIL($n = 4$) and other four algorithms. The number in the table is the true positive rates at the EER point on the ROC Curve.

on the validation set is selected. To ensure a fair comparison with key existing MIL algorithms, their parameters are determined using the same manner. Each experiment is repeated for 10 random splits, and the results reported are average over these runs.

There are various measurements for evaluating performance, including ROC curve, precision-recall curve, etc. The most widely measurement in recent years is AUC (area under the ROC curve) [8]. We will use AUC in this paper. The ROC curve plots the true positive rate (i.e. the recall) as a function of the false positive rate, and AUC measures the probability that a randomly chosen positive image will be ranked higher than a randomly chosen negative image.

We next compare the performance of ConMIL with that of both bag-based MIL, DD-SVM, and instance-based MIL, EM-DD and ASVM-MIL, and results are summarized in Table 1 and in Figure 4. The following observations can be made:

- Overall, ConMIL achieves the best results at AUC = 0.916.
- ConMIL performs well on categories with complex objects (e.g. see "building" in Figure 2) or with complex scene (e.g. see "beach" in Figure 2). This is because ConMIL considers the semantic linkage of these concurrent regions and encodes this information into the inference of the instance labels.
- The proposed ConMIL can not only categorize an image (bag) but also directly label the regions (instances). That is, it can localize the target object in an image. At the bottom row of Figure 2, we show the probability (salience) map of the localization results. The pixel value in each map indicates the computed probability score of region being positive. As illustrated, the algorithm has successfully localized the target regions for each category. We also calculate the AUC on the instance level to validate such localization ability, and the ground truth on these instances is manually labeled. The experiments prove our algorithm has a competitive performance on instance level as well (84.7%) compared with EM-DD (68.4%) and ASVM-MIL (74.5%).

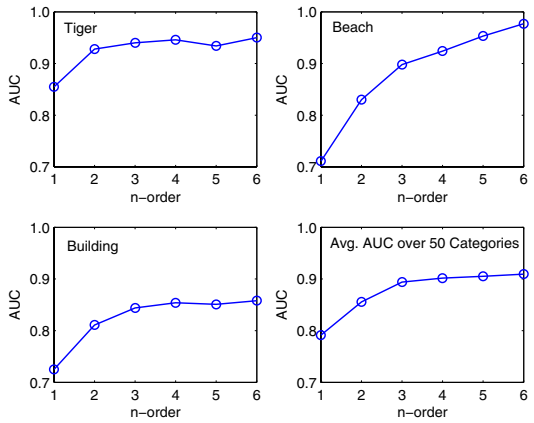

Figure 3. Learning curves (AUC v.s. current tensor order n) for tiger, beach, building and average over all 50 categories.

4.2. Evaluation on Caltech Dataset

While COREL 5000 contains both scenes, e.g., beach, and objects, e.g., tiger, the Caltech dataset has mostly objects. The significant variation in color, pose and lighting make this data set quite challenging. This data set was used extensively by several different research teams, both MIL-based [3] and non-MIL-based [6][2][13]. We will compare ConMIL with all of them.

Caltech contains 4 sets of object classes: Faces (450 images), Motorbikes (800 images), Airplane (800 images), Cars (800 images) and Background (900 general background and 1,370 road background). We follow the methodology in [6] for feature extraction. Salient regions are extracted using Kadir's salient region detector [9], and a feature vector of 18 dimensions is obtained to represent each salient region. The first 15 feature components are the first principle components output by PCA and the last 3 components are the scale and location of the extracted salient region. Similar to the COREL 5000 case, Caltech images in each category are also partitioned in half to form a training set and a testing set of equal size, and the two parameters σ^2 and λ are determined by a twofold cross-validation on the training set.

Table 2 reports the results compared to other 4 algorithms. To ensure fair comparison and to be consistent with the experiment setting in [3], the images in the object category Airplane, Faces and Motorbikes are tested against the general background while the Cars images are tested against the road background. Just as in [3], the performance is measured by the equal-error-rates (EER) point on the ROC curve. Compared with the four existing algorithms, ConMIL again gives the most competitive overall result, validating its robustness to different data sets.

Order n	1	2	3	4	5	6
ConMIL	0.67	2.5	6	28	100	200

EM-DD [20]	65
DD-SVM [4]	78
ASVM-MIL [19]	110

Table 3. Running time of ConMIL with different tensor order (top table) and other 3 algorithms(bottom table) (min)

4.3. ConMIL Analysis

Figure 3 shows how performance changes when the tensor order n changes. We plot the AUC results of three COREL 5000 categories (tiger, beach, and building) and the average AUC over all COREL 5000 categories versus the tensor order from 1 to 6. We also give the running time spent for different tensor orders from 1 to 6 in Table 3. The algorithms are run on a Pentium IV 3GHz PC. Combining Figure 3 and Table 3, the following observations can be made:

- When the order is 1, the concurrent tensor degrades to a vector, and ConMIL degrades into a regular MIL. Because the regular MIL does not model the inter-dependency inside a concept, n=1 gives the worst results.
- Higher-order concurrent tensor gives better results, but the performance saturates around n=4. This is because there rarely exists order-4 relationship in a well-defined concept/category. n=4 also give a good trade-off on the computation cost. n=4 is therefore our default setting in the experiments.
- ConMIL not only achieves good accuracy, but also is computationally efficient. As shown in Table 3, at n=4, it uses less time than all the other approaches.

5. Conclusion

To model the inter-dependency among regions in an image concept/category, in this paper we have proposed a new concurrent tensor-based MIL algorithm, ConMIL. It not only models the semantic linkage between the instances, but also avoids overfitting by formulating the label inferring processes into a regularization framework. Furthermore, it uses RKHS to extend predicted labels to any instances in the whole feature space to facilitate the testing process. Using two widely studied datasets, we have demonstrated that ConMIL achieves high classification accuracy on both bags and instances, is robust to different datasets, and is computationally efficient.

References

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In

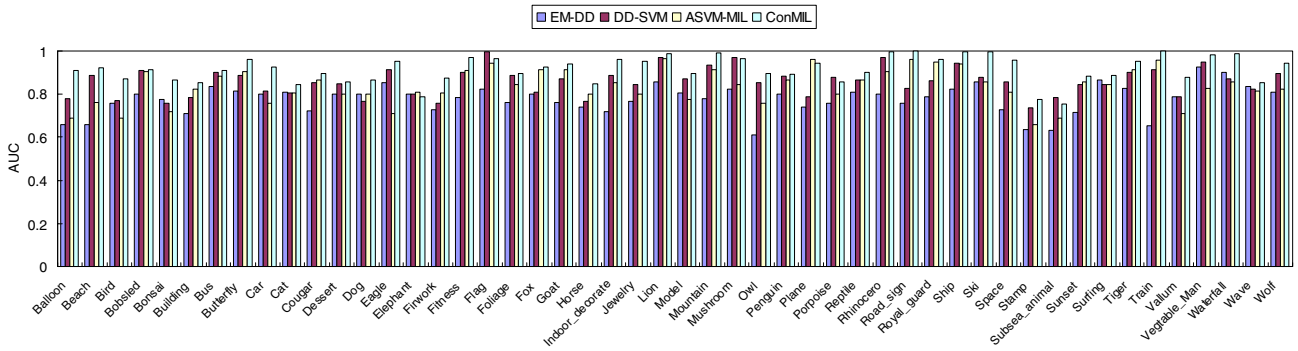

Figure 4. AUC value of 50 categories by EM-DD, DD-SVM and CTI-MIL (n=4).

Proc. of Advances in Neural Information Processing System, 2002. 2

[2] A. Bar-Hillel, T. Hertz, and D. Weinshall. Object class recognition by boosting a part-based model. In *Proc. of IEEE International Conference on CVPR*, 2005. 6, 7

[3] Y. Chen, J. Bi, and Z. Wang. Miles: Multiple-instance learning via embedded instance selection. *IEEE Transaction on Pattern Analysis and Machine Learning*, 28(12):1931–1947, 2006. 1, 2, 6, 7

[4] Y. Chen and J. Z. Wang. Image categorization by learning and reasoning with regions. *Journal of Machine Learning Research*, 5:913–939, 2004. 2, 6, 7

[5] T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Advances in Computational Mathematics*, 13(1):1–50, 2000. 2

[6] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pages 264–271, 2003. 6, 7

[7] T. Gartner, P. A. Flach, A. Kowalczyk, and A. J. Smola. Multi-instance kernels. pages 179–186, 2002. 2

[8] J. A. Hanley and B. J. McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143:29–36, 1982. 6

[9] T. Kadir and M. Brady. Scale, saliency and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001. 7

[10] E. Kofidis and P. A. Regalia. On the best rank-1 approximation of higher-order super-symmetric tensors. *SIAM Journal on Matrix Analysis and Applications*, 23(3):863–884, 2002. 1, 3

[11] D. C. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming B*, 45(1-3):503–528, 1989. 5

[12] O. Maron and A. L. Ratan. Multiple-instance learning for natural scene classification. In *Proc. International Conference on Machine Learning*, pages 341–349, 1998. 1, 2, 3

[13] A. Opelt, M. Fussenegger, and A. Pinz. Weak hypotheses and boosting for generic object detection and recognition. In *Proc. of ECCV*, 2004. 6, 7

[14] S. Ray and M. Craven. Supervised versus multiple instance learning: An empirical comparison. In *Proc. International Conference on Machine Learning*, pages 697–704, 2005. 2

[15] B. Schölkopf, R. Herbrich, and A. J. Smola. A generalized representer theorem. In *Proc. of the Annual Conference on Computational Learning Theory*, pages 416–426, 2001. 2, 5

[16] A. Shashua, R. Zass, and T. Hazan. Multi-way clustering using super-symmetric non-negative tensor factorization. In *Proc. of European Conference on Computer Vision*, pages 595–608, 2006. 4

[17] P. Viola, J. C. Platt, and C. Zhang. Multiple instance boosting for object detection. In *Proc. of Advances in Neural Information Processing System*, 2005. 1, 2

[18] R. Yager. On a general class of fuzzy connectives. *Fuzzy Sets and Systems*, 4:235–242, 1980. 3, 4

[19] C. Yang, M. Dong, and J. Hua. Region-based image annotation using asymmetrical support vector machine-based multi-instance learning. In *Proc. of IEEE International Conference on CVPR*, 2006. 1, 2, 6, 7

[20] Q. Zhang and S. A. Goldman. Em-dd: An improved multiple-instance learning technique. In *Proc. of Advances in Neural Information Processing System*, 2001. 2, 6, 7