AUTOMATIC IMAGE ANNOTATION BY INPUT-OUTPUT STRUCTURAL GROUPING SPARSITY

P. Jemimah Jeba Priya, T. Nalini

Abstract—Automatic image annotation (AIA) is used in image retrieval systems and important for image understanding. In this paper we introduce an input and output structural grouping sparsity into a regularized regression model for image annotation. In the proposed regression model, we relax the solving process as a framework of the bilayer regression model for multilabel boosting by the selection of heterogeneous features with structural grouping sparsity (Bi-MtBGS). In the first-layer of regression we select the discriminative features for each label. For input high-dimensional heterogeneous features such as color; texture, and shape, different kinds (groups) of features have different intrinsic discriminative power. The proposed structural grouping sparsity can be used not only to select group-of-features but also to conduct within-group selection. The second-layer regression is to refine the feature selection model learned from the first layer, which can be taken as a multilabel boosting process. Hierarchical correlations among output labels are well represented by a tree structure. Hence the proposed tree-structured grouping sparsity can be used to boost the performance of multilabel image annotation. Experiment on image data sets demonstrate that the proposed approach has better performance of multilabel image annotation.

Keywords—Automatic image annotation (AIA), structural grouping sparsity, structured feature selection, tree-structured grouping sparsity, machine learning.

I. INTRODUCTION

Recent years have seen a rapid increase in the size of digital image collections. Every day, both military and civilian equipment generates giga-bytes of images. A huge amount of Information is out there. However, we cannot access or make use of the information unless it is organized so as to allow efficient browsing, searching, and retrieval. Image retrieval has been a very active research area since the 1970s, with the thrust from two major research communities, database management and computer vision. These two research communities study image retrieval from different angles, one being text-based and the other visual-based.

In the social media websites such as Facebook and Flickr, there is an explosive growth of image data, which brings along great challenges in the research of image retrieval. During the early stage, a very popular framework was to manually annotate the images and then use text-based retrieval technologies to perform image retrieval [1]. But due to the vast amount of labor in manual image annotation and perception subjectivity, imprecise annotation may cause unrecoverable mismatches in retrieval processes [1]. Another approach called Content-based image retrieval targets to index images by their low-level visual features for similar image retrieval [2], [3]. The performance of image retrieval by visual features is bounded due to the semantic gap between low-level visual features and high-level semantics [1], [2].

One of the key techniques used in reducing the semantic gap is the machine learning methods to associate low-level visual features with high-level semantic concepts [4], or the named automatic image annotation (AIA) techniques [5]. The main concept of AIA techniques is to automatically learn semantic concept models from a large number of image samples and use the concept models to label new images [5]. Based on the machine learning point of view, the approaches of AIA can be roughly classified into the generative and discriminative models. The generative model is used to learn a joint distribution over image features and concepts (or annotation tags). For annotating a new image, the learned generative model computes the conditional probability over tags given the visual features [6], [7]. The discriminative model trains a separate classifier from visual features for each tag. These classifiers predict particular tags for test images [8], [9]. Similarly, we can also train a regression model (regression coefficients) for predicting tags for test images, taking features as predictors (input variables) and tags as responses (output labels).

When using the discriminative model to perform AIA, two key issues should be addressed. First, the “input structure” of visual features should be taken into consideration. Taking each type of visual features as a group, we could obtain input structures of feature groups and within-group features. Furthermore, different subsets of heterogeneous features have different intrinsic discriminative power to characterize the semantics of images. Considering the input structure of visual features, to better annotate images with suitable tags, a process of structured visual feature selection should be integrated into the learned discriminative model.

Second, due to the visually polysemous nature of images [12], one image may be labeled by multiple tags. Correlations or structures among multiple labeled tags usually indicate
certain correlations of visual content. Some works even defined hierarchical structures among multiple labeled tags and exploited these structures for better AIA [18]–[20]. Therefore, the “output structure” among the correlated multiple tags should be also considered during multitag image annotation.

The rest of this paper is organized as follows. In Section II we present the background of our work. Section III describes the Tree-Structured Multilabel Boosting by Structural Grouping Sparsity, Visual Feature Selection by Structural Grouping Sparsity, Multilabel Boosting by Tree-Structured Grouping Sparsity. Section IV describes the experiments and results. We conclude in Section V.

II. RELATED WORK

Image annotation or automatically annotate images with keywords is a solution to this problem. It is based on some machine learning techniques, which learn the correspondence between visual features and semantics of images. That is, image annotation systems can recognize or classify visual features into some pre-defined classes [9]. Figure 1 shows a general architecture of image annotation systems.

![Fig. 1 Block diagram of an image annotation system](image)

The segmentation component partitions images into local contents via either some block or region based method (c.f. Section 2). Then, the feature extraction component extracts low-level features from the segmented images (c.f. Section 3). That is, each segmented block or region is represented by feature vectors. Next, the annotation component assigns the (low-level) feature vectors to some pre-defined categories. This performs like the pattern classification task. Finally, the post-processing component (dependent on the application) uses the output of the annotation component to decide on some recommended action for the final decision.

In general, visual content of an image can be represented by either global or local features. Global features take all the pixels of an image into account. Color histograms [10], for examples, can be extracted to represent or describe the global color content of images. In this case, an image can be described that it contains 40% of blue, 37% of yellow and so on. However, as global features consider the visual features of the whole image, they cannot completely describe different parts of an image. On the other hand, image segmentation into local contents (i.e. different regions or areas) is able to provide more detailed information of images.

In general, there are two strategies for extracting local features [11–18]. The first one is to partition a set of fixed sized blocks or tiles (see Fig.2 for some examples) and the second for a number of variable shaped regions of interest (see Fig.3). After performing block and/or region based segmentation, low-level feature(s) can be extracted from the tiles or regions for local feature representation [19].

A. IMAGE LOW-LEVEL FEATURES

In general, low-level features such as color, texture, shape, and spatial relationship are extracted to represent image features.

1) Color

Color is the most used visual feature for image retrieval due to the computational efficiency of its extraction. All colors can be represented variable combinations of the three so-called primary colors: red (R), green (G), and blue (B). There are some other color spaces for representing the color feature, such as HSV, L*u*v*, YIQ, etc. [13]. In particular, color histogram [10] is one common method used to represent color contents for indexing and retrieval. It shows the proportion of pixels of each color within the image, which is represented by the distribution of the number of pixels for each quantized bin.

2) Texture

Texture is an important element of images for surface, object identification, and region distinctions. In addition to colors, it has been extracted to classify and recognize objects and scenes. Texture can be regular or random. Most natural textures are random. Regular textures are composed of textures that have a regular or almost regular arrangement of identical, or at least similar, components. Irregular textures are composed of irregular and random arrangements of components related some statistical properties [20].

![Fig. 2 Examples of block-based segmentation](image)

![Fig. 3 Region-based segmentation](image)
3) **Shape**

Shape is one of the most important features for describing the content or object(s) of an image. Compared with colour and texture features, shape features are usually described after images have been segmented into regions or objects. The shape representations can be divided into two categories, boundary-based (or edge detection) and region based. The former uses only the outer boundary of the shape, such as the chain code method, while the latter uses the entire shape region [1]. However, to effectively extract shape features depends on segmentation methods.

4) **Spatial Relationship**

Objects and the spatial relationships (such as left of, inside, and above) among objects in an image are used to represent the image content [21]. That is, an image can be divided into a number of sub-blocks and colour, texture, and/or shape features are extracted from each of the sub-blocks. Then, we can project them along the x and y axes, such as ‘left/right’, ‘below/above’ relationships between them. Ko et al. [14] consider spatial colour histograms which show better performances than the traditional one, i.e. global colour histogram.

III. **PROPOSED SYSTEM**

In this paper, we propose to combine structures of input features and output multiple tags into one regression framework for better performance of multilabel image annotation. The training process of the proposed framework is illustrated in Fig. 4. For each image, we can extract heterogeneous visual features, e.g., color, texture, and shape, in order to form matrix. Each coefficient vector, e.g., column vector in matrix, is used to fit the values of visual features to the values of corresponding label indicators in matrix. Since there are multiple labeled tags, the proposed framework is a process of multiresponse regression. As shown in Fig. 4, we propose to impose a well-defined structural grouping sparsity penalty on the input direction of matrix, i.e., on each column vector of, to perform heterogeneous visual feature selection. In addition, simultaneously, we propose to impose a tree-structured grouping sparsity penalty on the output direction of matrix, i.e., on each row vector of, to perform tree-structured multilabel boosting. The penalty of structural grouping sparsity is defined as follows. Taking each type of features as a group (see Fig. 4 for example), we get three groups of visual features, which correspond to color, texture, and shape. We first propose to impose the penalty of a sum of (not squared) - norm on each group of coefficients, which corresponds to the group of input features, to perform group selection. In Fig. 4, denotes that corresponding groups of features (color features) are selected and otherwise. In order to further discern the most discriminative features within groups, we propose to simultaneously impose on the whole coefficient vectors (e.g., column vector) an norm penalty to induce within-group sparsity. Therefore, the goal of the structural grouping sparsity is to select finite groups of heterogeneous features and identify subgroup within homogeneous features for predicting particular tags of images.

In this section, we will describe the framework of bilayer regression for multilabel boosting by selecting heterogeneous features with structural grouping sparsity (Bi-MtBGS). In order to solve this framework efficiently, we relax the solving process as a bilayer regression model regularized by input–output structural grouping sparsity.

A. **Visual Feature Selection by Structural Grouping Sparsity**

For the jth label, first-layer regression is formulated to solve the following regularized regression model:

\[
\min_{\beta_j} \|y - X_{\cdot j}\|_2^2 + \lambda_1 \sum_{l=1}^L \|\beta_j^{(l)}\|_1 + \lambda_2 \sum_{l=1}^L \|\beta_j^{(l)}\|_2^2
\]

where \(\lambda_1 > 0\) and \(\lambda_2 > 0\) are regularization parameters, \(\lambda_1 \|\beta_j^{(l)}\|_1\) is the instantiation of the input sparsity penalty \(P^{(l)}(\beta_j)\) is called the penalty of structural grouping sparsity. When the extracted heterogeneous features are divided into L disjoint groups of homogeneous features, the penalty in above equation can induce sparsity at the group level in order to conduct group selection (see Fig. 4). Since the \(l_1\)-norm penalty can induce sparsity on the whole coefficient vector \(\beta_j\), the structural grouping sparsity \(P^{(l)}(\beta_j)\) can also conduct within-group feature selection. Therefore, the structural grouping sparsity in the equation (1) can be used to perform structured visual feature selection in the input direction.

![Fig. 4 Training process of the proposed framework. In matrix X, different colors of rectangles indicate different types of heterogeneous visual features. Rectangles in dashed border indicate features’ corresponding coefficients. Blank rectangles indicate zero values in coefficient matrix B and indicator matrix Y. Dark rectangles in Y indicate that images are labeled by corresponding tags.](image1)

![Fig. 5 Correlations among the labels are exploited by agglomerative hierarchical clustering on the label indicator matrix B in Fig. 1 to form a tree structure T in Fig. 5(a). (a) Tree structure of annotated labels in MIML data set, (b) Tree-structured groups of regression coefficients.](image2)
B. Multilabel Boosting by Tree-Structured Grouping Sparsity

Let $\mathbf{B}$ be the solution of equation, second-layer regression is formulated to solve the following regularized regression model:

$$
\hat{\mathbf{B}} = \arg \min_{\mathbf{B}} \| \mathbf{Y} - \mathbf{X} \mathbf{B} \|_2^2 + \gamma \sum_{d=1}^{p} \sum_{v \in T} w_v \| \mathbf{B}_v \|_2
$$

(2)

The instantiation of the output-sparsity penalty in (2) and is called the penalty of tree-structured grouping sparsity. Suppose the correlations among the labels (outputs) can be represented as a tree with the set of vertices of size. Each of the leaf nodes corresponds to an output label (tag), and the internal nodes of the tree represent groups of output labels located at the leaves of the subtree rooted at the given internal node (see the tree structure in Fig. 5 for example). Taking the leaf node as a singleton group, internal nodes as corresponding groups of tags located at the leaves of the subtree rooted at the given internal node, we get a tree-structured set of groups, where represents a group of tags corresponding to node. For example, in Fig. 5(a), we have sea and sea sunset desert mountain tree. The weight in (2) is defined as:

$$
w_v = \begin{cases}
g_v \prod_{m \in \text{Ancestor}(v)} s_m, & \text{v is an internal node,} \\
\prod_{m \in \text{Ancestor}(v)} s_m, & \text{v is a leaf node}
\end{cases}
$$

(3)

Where the two quantities and are associated with the internal node of the tree, which satisfy condition with each node in Fig. 5(a) for each node is calculated by formula (3).

IV. EXPERIMENTS AND RESULTS

In this section, we will introduce the solutions and algorithms of solving (1) and (2), respectively, which presents the final solutions of the proposed framework Bi-MtBGS.

A. Regularized Regression with Structural Grouping Sparsity

The objective function in (1) is convex and separable (considering the zero point in norm penalty) so that the blockwise coordinate descent at the group level and piecewise coordinate descent within group for individual features can be used for optimization.

1) Group Selection: The subgradient equation of the first two terms in (4) with respect to $\beta_{jl}$ is

$$
-X_j^T \left( Y_{(-j)} - \sum_l X_{jl} \beta_{jl} \right) + \lambda_1 s_{jl} = 0
$$

(4)

Algorithm 1 Regularized Regression with Structural Grouping Sparsity

Input: Training images $\mathbf{X} \in \mathbb{R}^{n \times p}$ containing $n$ samples with $p$ dimensionality, corresponding label indicator matrix $\mathbf{Y} \in \{0,1\}^{n \times k}$, initialization of coefficient matrix $\mathbf{B} \in \mathbb{R}^{p \times k}$. Suppose that $L$ types of heterogeneous visual features are extracted from each image sample, we have $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_L)$ with $\mathbf{X}_l \in \mathbb{R}^{n \times p_l}$ ($l = 1, \ldots, L$).

Output: The estimated $\mathbf{B}^\dagger$.

1: for $j = 1, \ldots, J$
2: iterate over groups $l = 1, 2, \ldots, L, 1, \ldots,$
3: if condition in (8) holds then
4: $\beta_{jl} = \{0 \}^p$
5: else
6: iterate over $m = 1, 2, \ldots, p_l, 1, \ldots$
7: initialize $I_m = \{1, \ldots, m_k\}$ and $I_{m_n} = \emptyset$
8: while optimality violator exists in $I_m$
9: find the maximum violator viol$_m$ ($m \in I_2$) and move to $I_{m_n}$
10: for each element in $I_{m_n}$
11: while viol$_m$ > $\tau$
12: update $\theta_m$ by (13)
13: if $\theta_{m}^{new} \leq L$ or $\theta_{m}^{new} \geq H$ then $\theta_{m}^{new} = (L + H) / 2$
14: compute $g_m$ using (10)
15: if $g_m > 0$ then $H = \theta_m$
16: else if $g_m < -\tau$ then $L = \theta_m$
17: end while
18: end for
19: end while
20: until convergence
21: output $\beta_{jl}$
22: end if
23: until convergence
24: output $\hat{\mathbf{B}}_j = (\hat{\beta}_{j1}, \ldots, \hat{\beta}_{jL})^T$
25: end for
26: output $\mathbf{B} = (\hat{\beta}_1, \ldots, \hat{\beta}_J)$.

B. Tree-Structured Multi-Label Boosting Methods

By introducing additional variables, an alternative optimization algorithm for solving (2) was developed to estimate $\mathbf{B}^\dagger$. However, the discussion of the initialization of was ignored. In this paper, taking $\mathbf{B}^\dagger$ estimated from solving (1) as the initialization input for (2), we form a bilayer regression model.

Algorithm 2 Multilabel Boosting with Tree-structured Grouping Sparsity

Input: Training data $(\mathbf{X}, \mathbf{Y})$, the initialization of coefficient matrix $\mathbf{B}$ is the output from Algorithm 1.

Output: The estimated $\mathbf{B}^\dagger$.

1: iterate
2: update $b_{dv}$ using (14), $\forall d, v$
3: for $j = 1, \ldots, J$
4: update $\beta_{j}$ using (16)
5: end for
6: until convergence
7: output $\hat{\mathbf{B}} = (\hat{\beta}_1, \ldots, \hat{\beta}_J)$.
We will evaluate the performance of our proposed Bi-MtBGS framework in automatic multilabel image annotation. The effectiveness of the structured feature selection is demonstrated using two open benchmark image data sets, and then we evaluate and compare the performance of Bi-MtBGS with state-of-the-art methods.

<table>
<thead>
<tr>
<th>Criteria</th>
<th>CCA-edge</th>
<th>SVM</th>
<th>MTLS</th>
<th>MLMC</th>
<th>CAR</th>
<th>Bi-MtBGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>micro-AUC</td>
<td>0.865±0.087</td>
<td>0.837±0.119</td>
<td>0.821±0.101</td>
<td>0.820±0.102</td>
<td>0.791±0.090</td>
<td>0.867±0.088</td>
</tr>
<tr>
<td>macro-AUC</td>
<td>0.780±0.101</td>
<td>0.738±0.125</td>
<td>0.750±0.101</td>
<td>0.745±0.102</td>
<td>0.708±0.090</td>
<td>0.784±0.103</td>
</tr>
<tr>
<td>micro-F1</td>
<td>0.341±0.012</td>
<td>0.189±0.015</td>
<td>0.290±0.018</td>
<td>0.286±0.019</td>
<td>0.260±0.018</td>
<td>0.348±0.011</td>
</tr>
<tr>
<td>macro-F1</td>
<td>0.310±0.012</td>
<td>0.093±0.015</td>
<td>0.222±0.018</td>
<td>0.224±0.019</td>
<td>0.206±0.018</td>
<td>0.356±0.011</td>
</tr>
</tbody>
</table>

V. CONCLUSION

This paper has proposed a framework of multilabel learning for image annotation called the Bi-MtBGS. The Bi-MtBGS method is attractive due to its subgroup feature identification by structural grouping penalty in heterogeneous feature settings along with its tree-structured multilabel boosting capability. Bi-MtBGS achieves better performance of multilabel image annotation compared with state-of-the-art methods. Bridging the semantic gap for image retrieval is a very hard problem to solve. In the context of automatic image annotation, the major difficulty is to make computers understand image content in terms of high-level concepts or semantics, which is closely related to the problem of computer vision and object recognition. To bridge the semantic gap between low-level features and high-level concepts could be approached by image classification. A learning machine or classifier is trained by learning low-level features for classifying images into some conceptual categories. The classification process can be thought of as image annotation.

REFERENCES