Exploiting Related and Unrelated Tasks for Hierarchical Metric Learning and Image Classification

Yu Zheng, Jianping Fan, Ji Zhang, and Xinbo Gao, *Senior Member, IEEE*

***Abstract*— In multi-task learning, multiple interrelated tasks are jointly learned to achieve better performance. In many cases, if we can identify which tasks are related, we can also clearly identify which tasks are unrelated. In the past, most researchers emphasized exploiting correlations among interrelated tasks while completely ignoring the unrelated tasks that may provide valuable prior knowledge for multi-task learning. In this paper, a new approach is developed to hierarchically learn a tree of multi-task metrics by leveraging prior knowledge about both the related tasks and unrelated tasks. First, a visual tree is constructed to hierarchically organize large numbers of image categories in a coarse-to-fine fashion. Over the visual tree, a multi-task metric classifier is learned for each node by exploiting both the related and unrelated tasks, where the learning tasks for training the classifiers for the sibling child nodes under the same parent node are treated as the interrelated tasks, and the others are treated as the unrelated tasks. In addition, the node-specific metric for the parent node is propagated to its sibling child nodes to control inter-level error propagation. Our experimental results demonstrate that our hierarchical metric learning algorithm achieves better results than other state-of-the-art algorithms.**

***Index Terms*— Hierarchical metric learning, multi-task learning, related and unrelated tasks, visual tree.**

I. Introduction

LARGE-SCALE image classification, e.g., classifying millions of images into thousands or even tens of thousands of image categories (object classes and scenes), is still a challenging issue even though current state-of-the-arts methods have achieved impressive progress in terms of their accuracy rates [1]–[5]. To achieve good performance in large-scale image classification, we need to learn large numbers of classifiers to recognize thousands or even tens of thousands of image categories so that the corresponding systems can automatically interpret rich and diverse semantics of millions of images similar to knowledgeable human beings. Unfortunately, the existing methods still encounter many obstacles to accomplishing this ambitious task. Unlike small-scale image classification (i.e., classifying images into tens or hundreds of categories), it is still very difficult to classify millions of images into large numbers of categories by directly using traditional flat classifiers because the flat classifier, e.g., the OVR (one-versus-rest) approach or the OVO (one-versus-one) approach, has serious problems in addressing large-scale image classification [6]–[8]. The computational cost for the flat OVR binary approach (i.e., learning an OVR binary classifier for each category independently without considering the inter-category correlations) grows linearly with the number of image categories; in contrast, the computational cost for the flat OVO approach grows quadratically with the number of image categories. Thus, such flat approaches become computationally impractical for large-scale image classification applications (in which the number of image categories to be recognized is typically very large). One way to address this issue is to integrate a tree structure to hierarchically organize large numbers of image categories in a coarse-to-fine fashion [9]–[12], and such a hierarchical approach can achieve acceptable (sublinear) computational complexity at test time. In addition, the underlying tree structure may provide the correlations of image categories (i.e., automatically identify the interrelated and unrelated learning tasks).

Hierarchical learning has been successfully applied in many applications, such as fingerprint identification [13], medical diagnosis [14] and scene understanding [3], [15]. According to the organizations of large numbers of image categories, hierarchical learning can be roughly categorized into three types: (a) semantic tree, (b) label tree, and (c) visual tree. Particularly, the visual tree employs hierarchical clustering to build a tree structure to characterize the visual similarities among large numbers of image categories. As shown in Fig. 1, because feature space is the common space for classifier training and image classification, the visual tree can provide a good environment to train more discriminative tree classifiers for large-scale image classification applications. Recently, many

Manuscript received August 13, 2018; revised April 2, 2019, July 17, 2019, and August 6, 2019; accepted August 12, 2019. Date of publication September 5, 2019; date of current version October 9, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61906143, Grant 61432014, Grant 61772402, Grant U1605252, and Grant 61671339, in part by the National Key Research and Development Program of China under Grant 2016QY01W0200, in part by National High-Level Talents Special Support Program of China under Grant CS31117200001, in part by the China Postdoctoral Science Foundation under Grant 2018M643584, and in part by the Fundamental Research Funds for the Central Universities under Grant JB191505. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Gang Hua. *(Corresponding author: Xinbo Gao.)*

Y. Zheng is with the School of Cyber Engineering, Xidian University, Xi'an 710071, China (e-mail: yuzheng.xidian@gmail.com).

J. Fan is with the Department of Computer Science, University of North Carolina at Charlotte, Charlotte, NC 28223 USA (e-mail: jfan@uncc.edu).

J. Zhang is with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: zhang_ji@stu.xjtu.edu.cn).

X. Gao is with the School of Electronic Engineering, Xidian University, Xi'an 710071, China (e-mail: xbgao@mail.xidian.edu.cn).

Digital Object Identifier 10.1109/TIP.2019.2938321

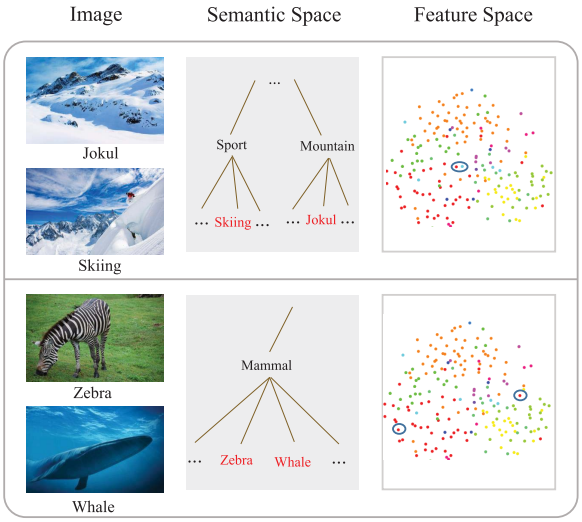


Fig. 1. Sample images are illustrated in semantic space and feature space. As shown in the upper part, some images are far apart in semantic space but very close in feature space (i.e., Jokul and Skiing are irrelevant in semantic space, but these two scenes are very similar, so the distance in feature space is small). However, as shown in the lower part, some images are far apart in feature space but very close in semantic space (i.e., zebras and whales are both mammals, but they differ in appearance, texture, and color.)

studies have focused on learning a discriminative classifier over the visual tree [16]–[19]. Specifically, multi-task learning is widely employed to learn the node classifiers in a tree classifier, and the reasons are as follows: (a) some image categories may have strong inter-category visual similarities, especially the fine-grained image categories, and (b) such visual trees can provide a good environment to automatically identify the interrelated learning tasks (i.e., the learning tasks for training the classifiers for multiple sibling child nodes under the same parent node are typically interrelated). Therefore, multi-task learning [20]–[22] could be a reasonable solution for training discriminative node classifiers over a visual tree. However, only the related tasks are considered in conventional multi-task learning approaches, while the unrelated tasks that may also provide valuable prior knowledge for enhancing multi-task learning are completely ignored [16], [23]. The related tasks may tend to share some of the features, while the unrelated tasks tend not to share any features. Furthermore, a visual tree may provide a good environment to identify which tasks are related and which tasks are not. Thus, it is very attractive to develop more effective algorithms to leverage the visual tree to make full use of both related and unrelated tasks for multi-task learning.

In addition, it is worth noting that each node classifier over the visual tree may utilize different features; thus, feature selection should be performed for node classifier training. Distance metric learning can be effective in improving the performance of the node classifiers by adopting a proper metric to achieve more accurate similarity characterization [24]–[26]. Some researchers have applied metric learning in hierarchical classifier training [17], [18]. However, these algorithms do not make full use of the internode visual correlations and the inter-level visual correlations over the tree structures; thus, they are not scalable for large-scale image classification applications.

Motivated by these observations, in this paper, a hierarchical metric learning algorithm is developed by exploiting the related and unrelated tasks to learn a tree of multi-task metrics for large-scale image classification. Fig. 2 illustrates the system flowchart for our hierarchical metric learning. First, an enhanced visual tree is constructed to hierarchically organize large numbers of image categories, and such an enhanced visual tree can provide a good environment to automatically identify the related tasks and unrelated tasks. After the visual tree is constructed (i.,e., similar categories in feature space have been clustered together, whereas dissimilar categories have been separated on the visual tree), the hierarchical classifier is trained over the visual tree in a top-to-bottom way. We consider the learning tasks of training the classifiers for the sibling child nodes under the same parent node to be strongly related, while the learning tasks of training the classifiers for the child nodes under different parent nodes are considered to be unrelated. Second, for the root node at the first level of the visual tree, a multi-task metric learning algorithm is used to learn its node classifier to separate its sibling child nodes at the next level. Third, for the non-root node of the visual tree, both the related tasks and unrelated tasks are utilized to learn a multi-task metric to separate its sibling child nodes at the next level. Fourth, for the non-root node of the visual tree, the inter-level visual correlations (between the parent node and its sibling child nodes at the next level of the visual tree) are utilized to enhance the learning of the multi-task metric. We make the node-specific individual metric for its parent node at the upper level of the visual tree and its commonly shared metric to be similar or close so that more discriminative metrics can be learned to effectively control inter-level error propagation.

The rest of the paper is organized as follows. In Section 2, we review some relevant work. In Section 3, a hierarchical learning algorithm is developed to leverage both the related tasks and unrelated tasks to train a more discriminative tree classifier over the visual tree. Section 4 demonstrates the experiments involving our proposed algorithm, followed by conclusions in Section 5.

II. Related Work

This paper introduces an algorithm to leverage both the related tasks and the unrelated tasks to learn hierarchical multi-task metrics over a visual tree for large-scale image classification applications. Therefore, our discussion of related work contains three parts: hierarchical learning, multi-task learning and metric learning.

A. Hierarchical Learning

When large numbers of image categories are considered, the taxonomies are typically used to hierarchically organize them in a coarse-to-fine fashion. The integration of the taxonomies for hierarchical classifier training makes the hierarchical approach very attractive for large-scale image classification applications because it can achieve sublinear

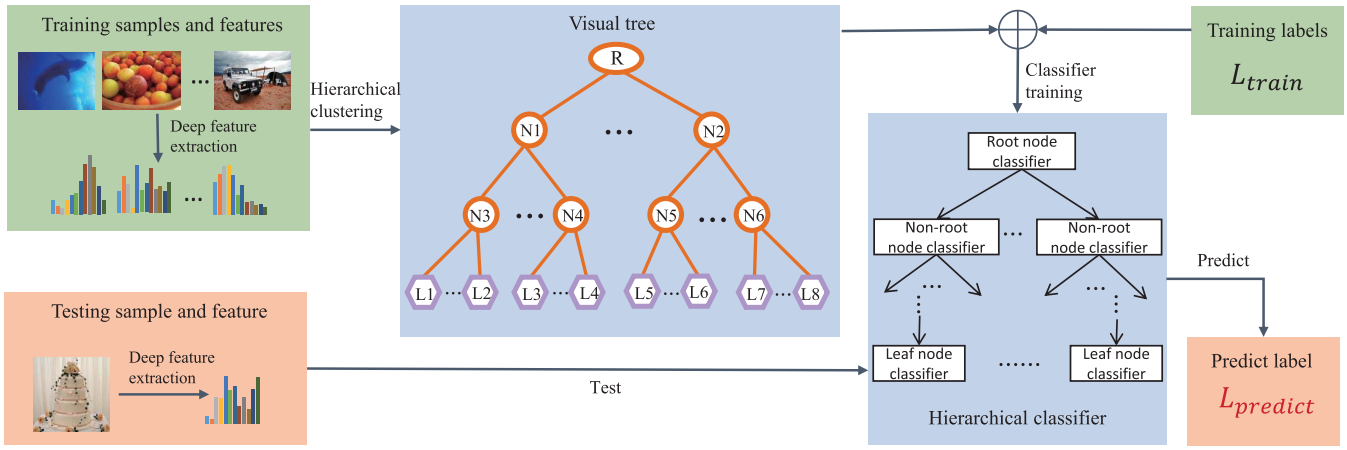


Fig. 2. The flowchart of our hierarchical metric learning algorithm, where the training samples are first used to build a visual tree by using unsupervised hierarchical clustering, and multi-task metric learning is then applied to train the hierarchical classifier over the visual tree in a top-to-bottom way. Both related tasks and unrelated tasks are utilized to train the multi-task metric classifier over the visual tree. In the test phase, the test samples pass through the hierarchical classifier from top-to-bottom until reaching the leaf node, and the class label of the leaf node is then the predicted label.

computational complexity at test time. Therefore, hierarchical learning has attracted sufficient attention in the communities of computer vision, machine learning and multimedia computing [10], [12], [16]–[18], [27]–[33]. Hierarchical learning can roughly be categorized into three types: (a) semantic tree [28], [30], [34], (b) label tree [31]–[33], [35], and (c) visual tree [12], [15], [29]. Some researchers prefer to employ semantic trees to support hierarchical classifier training because semantic trees can utilize the relationship among the object classes in the real world to automatically build the tree structure. Naphade et al. proposed a hierarchical approach based on a concept ontology for multimedia computing [34]. Marszalek et al. employed the affiliation between nouns of WordNet to build a semantic tree for visual recognition [30], [36]. The ImageNet data set was created by associating each category with each word on WordNet [37]. Zhang et al. employed the tree structure to create the quadruplet and then proposed a multi-task learning framework for fine-grained feature representation by utilizing both classification loss and similarity loss [38]. However, feature space is the common space employed for classifier training and image classification. Therefore, some researchers learn a label tree directly from large amounts of training images in feature space. The label tree is a kind of visual hierarchy that is able to characterize the inter-category visual correlations in feature space. However, it could be very expensive to learn a label tree for organizing large numbers of image categories [35]. To learn the label tree for N image categories, N OVR binary classifiers are first learned and further evaluated to obtain an N∗N confusion matrix; the label tree is then learned from this N∗N confusion matrix. Thus, some researchers develop visual trees because they are very attractive for learning the visual hierarchy with less computational cost. Nister et al. built a vocabulary tree to speed up dictionary search by employing hierarchical clustering [39]. Fan et al. employed hierarchical spectral clustering to build the visual tree [29]. Zhao et al. proposed a feature selection method by employing a hierarchical tree structure, wherein both the parent-children correlations and the sibling correlations are considered for hierarchical regularization [40]. Motivated by the above analysis, we choose the visual tree approach to hierarchically organize a large number of categories.

B. Multi-Task Learning

Multi-task learning intends to combine similar tasks to jointly learn them [20], [21], [23], [41], [42]. From the most recent work, it is obvious that when the inter-task relationships are effectively identified, the performance when simultaneously learning the classifiers for the related tasks is better than that when independently learning the classifiers for each task. In recent years, there have been many excellent works on multi-task learning. Argyriou et al. pioneered convex multi-task feature learning [20]. Liu et al. combined the idea of hierarchical clustering and multi-task learning for action recognition [43]. Zhang et al. employed deep multi-task learning to detect facial landmarks [41]. However, the critical issue of multi-task learning is how to automatically identify the related tasks. Some researchers employ the hierarchical structure to identify the related tasks [15], [29]. Fan et al. proposed hierarchical learning of a tree classifier by employing the visual tree to identify the related tasks. They considered the multiple sibling child nodes under the same parent node to be typically related. However, none of these methods have exploited the unrelated tasks to enhance multi-task learning, even though such unrelated tasks may provide valuable prior knowledge. Moreover, the related tasks may tend to share some of the features, while the unrelated tasks may tend to not share any features. The idea of exploiting unrelated tasks has been proposed in recent works [16], [23], [44]. Similar to the related tasks, determining how to identify the unrelated tasks is also a critical issue. Therefore, it is very attractive to develop hierarchical multi-task learning algorithms that can effectively identify both the related tasks and the unrelated tasks and leverage such knowledge to enhance multi-task learning.

C. Metric Learning

Metric learning concerns learning a reasonable metric over the input space, and it has attracted considerable attention recently [24], [45]–[50]. Xing et al. learned a good distance metric for similar point pairs by respecting these relationships [45]. Weinberger et al. presented a Mahalanobis distance function for the k-nearest neighbors (kNN) classifier by utilizing a triplet loss that forces exemplars from the same class to be clustered together, while exemplars from different classes are effectively separated [24]. Davis et al. proposed an information-theoretic Mahalanobis distance metric approach by minimizing the differential relative entropy between two distance functions [46]. In addition, some researchers extended distance metric learning to the multi-task setting [22], [51], [52]. Parameswaran et al. developed large-margin multi-task metric learning [22]. Ma et al. employed multi-task metric learning to attain person re-identification [51]. Bhattarai et al. proposed coupled projection multi-task metric learning (CP-mtML) that employs pairwise constraints for face retrieval [52]. Moreover, some interesting approaches have been developed to combine metric learning and hierarchical learning [17], [18]. Grauman et al. developed a tree of disjoint metrics by adding a trace-norm regularization term into the objective function [17]. Lei et al. developed a hierarchical approach by employing large-margin metric learning to learn the node classifiers [18].

III. Hierarchical Metric Learning

A. Construction of the Visual Tree

When large numbers of image categories need to be recognized, it is very attractive to construct a visual tree to hierarchically organize them in a coarse-to-fine fashion according to their inter-category visual similarities. Because feature space is the common space for classifier training and image classification, such a visual tree can provide a good environment to train more discriminative tree classifiers for large-scale image classification applications.

General visual trees are often constructed by employing hierarchical clustering. Since the intermediate nodes of the visual tree have no semantic meaning, it is difficult to prespecify the number of cluster centers in the visual tree construction. Therefore, we utilize a hierarchical affinity propagation (AP) clustering algorithm to learn a visual tree [53]. The advantage of AP clustering is that the number of cluster centers can be learned automatically. Motivated by the work [15], the Hausdorff distance [54] is used to characterize the inter-category visual similarities. In AP clustering, the similarity matrix between samples needs to be calculated first. Considering that there are multiple representations for each category in visual tree construction, we employ the Hausdorff distance to characterize the inter-category similarities S. In particular, AP clustering sets the diagonal of the similarity matrix $s(k,k)$ to a preference parameter. Samples larger than the preference parameter may be selected as cluster centers. The preference parameter is defined as:

$$p = \mu * \text{median(s)} \tag{1}$$

Particularly, μ is a coefficient of the preference parameter that can affect the number of cluster centers. The smaller the coefficient is, the greater the number of cluster centers. Then, the responsibility matrix R and the availability matrix A are calculated.

The responsibility is updated as:

$$r_{t+1}(i,k) = \begin{cases} s(i,k) - \max\limits_{j \neq k}\{a_t(i,j) + r_t(i,j)\}, & \text{i} \neq k \\ s(i,k) - \max\limits_{j \neq k}\{s(i,j)\}, & \text{i} = k \end{cases} \tag{2}$$

The availability is updated as:

$$a_{t+1}(i,k) = \begin{cases} \min\{0, r_{t+1}(k,k) + \sum\limits_{j \neq i,k} \max\{r_{t+1}(j,k), 0\}\}, & \\ & \text{i} \neq k \\ \sum\limits_{j \neq k} \max\{r_{t+1}(j,k), 0\}, & \text{i} = k \end{cases} \tag{3}$$

To prevent oscillation, the damping coefficient σ is introduced, where the value of σ defaults to 0.5. Then, the update of the responsibility and availability is defined as:

$$\begin{aligned} r_{t+1}(i,k) &= \sigma * r_t(i,k) + (1-\sigma) * r_{t+1}(i,k) \\ a_{t+1}(i,k) &= \sigma * a_t(i,k) + (1-\sigma) * a_{t+1}(i,k) \end{aligned} \tag{4}$$

The above update rules are repeated until the responsibility and availability are stable. For sample i, the sample k that maximizes $a(i,k) + r(i,k)$ is taken as the cluster center.

Our visual tree can be built by top-down hierarchical AP clustering. Fig. 3 illustrates the visual tree for the ILSVRC-2012 image set. In this visual tree, the root node contains all the image categories, the non-root node contains some of the image categories, and the leaf node contains only one single image category.

After the visual tree is available, a hierarchical multi-task metric is learned over the visual tree in a top-down fashion. For the root node, a multi-task metric is learned to build the root node classifier. For the non-root nodes, we employ multi-task learning by exploiting related and unrelated tasks to learn their multi-task metrics for building their node classifiers; meanwhile, both the inter-node visual correlations and the inter-level visual correlations are utilized to learn more discriminative metrics.

In addition, the structure of the visual tree can effectively prevent error propagation, which is one of the important issues in hierarchical learning. At the top level of the visual tree, all categories are clustered into several super-categories. The categories that can be separated at the beginning are the easily distinguishable categories. Then, each super-category continues to cluster the atomic categories into sub-super-categories. However, these atomic categories belong to the same parent node, which means they have visual similarities; thus, they are hard-to-distinguish categories. In this way, the high-level node classifiers always handle the more easily distinguishable categories in feature space, which can effectively prevent error propagation.

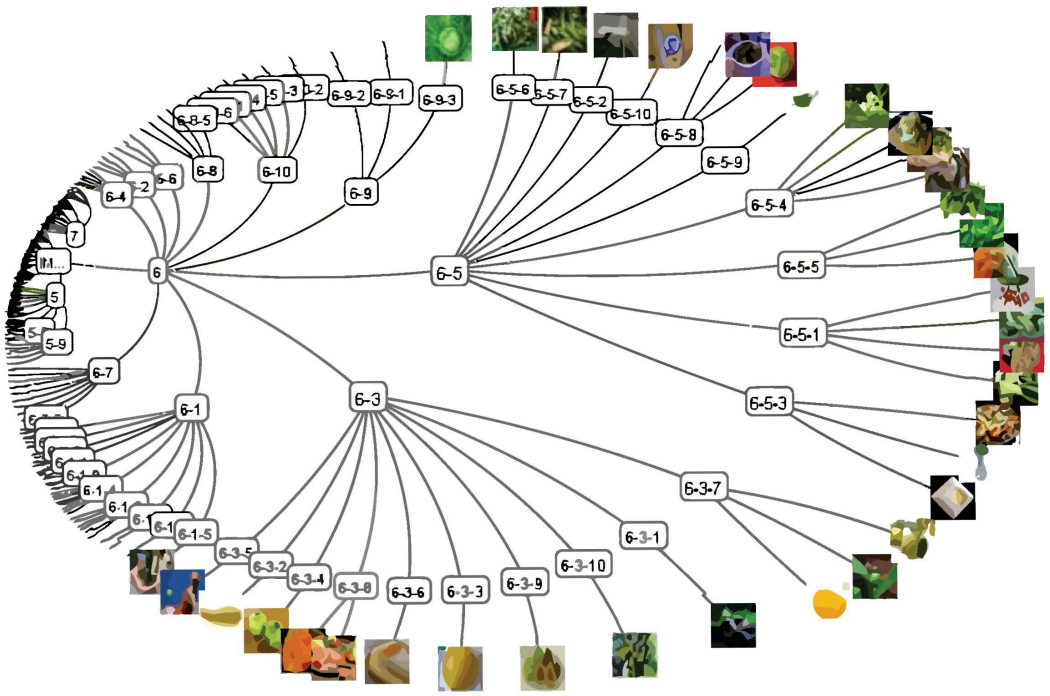


Fig. 3. The visual tree for the ILSVRC-2012 image set with 1000 categories. The visual tree is an imbalanced tree, and the depth of the visual tree is 4 (from the root node to the leaf nodes). One icon image of a leaf node is used to represent one particular image category.

B. Background on Distance Metric Learning

Here, we review the large-margin distance metric learning algorithm as introduced in [24]. Overall, the large-margin distance metric learning algorithm is a k-nearest neighbors (kNN)-based approach under the single task framework. Therefore, this algorithm is used for the local neighborhood exemplars of the target sample. The general kNN uses the Euclidean metric to measure the distance between the samples, and the distance metric learning uses the Mahalanobis metric to learn a linear projection matrix that maps samples from the original feature space to another correcting feature space. This algorithm focuses on calculating a symmetric positive-definite matrix that forces the exemplars from the same class to be clustered together, while exemplars from different classes are effectively separated. The Mahalanobis distance between two samples x_i, x_j is defined as:

$$d_W(x_i, x_j) = \sqrt{(x_i - x_j)^T W (x_i - x_j)} \tag{5}$$

where W is the Mahalanobis metric matrix.

Large-margin distance metric learning leverages a large margin to keep the exemplars from the same-category exemplars closer to the target samples than the exemplars from the different-category exemplars. Thus, the loss function is a triplet loss, defined as:

$$d^2_W(x_i, x_k) - d^2_W(x_i, x_j) \geq 1 \tag{6}$$

where x_i is the target sample, x_j is a same-category exemplar, and x_k is a different-category exemplar.

The cost functions are proved to be convex, and the unknown variables form a symmetric positive-definite matrix [24], so the optimization problem is transformed into a semi-definite programming (SDP) problem. Therefore, it can be solved by any standard SDP toolbox.

C. Multi-Task Metric Learning for the Root Node

In standard distance metric learning, the Mahalanobis metric matrix is learned directly from all the training samples. Parameswaran et al. extended distance metric learning to the multi-task setting [22]. The core idea of this algorithm is to divide the Mahalanobis metric matrix into two parts: a common metric and an individual metric. The common metric is shared by all related tasks and represents the common properties of all tasks; in contrast, the individual metric is for each particular task and represents the individual properties of this task. The Mahalanobis distance of the multi-task metric is defined as:

$$d_t(x_i, x_j) = \sqrt{(x_i - x_j)^T (W_0 + W_t)(x_i - x_j)} \tag{7}$$

where W_0 is the common metric $W_0 \succeq 0$ shared by all related tasks, and W_t is the individual metric $W_t \succeq 0$ for each particular task t, $t = 1, \ldots, T$.

For multi-task learning, determining how to identify the related tasks has always been an open problem. Fortunately, our visual tree can provide a good environment for automatically identifying such related tasks, e.g., the image categories that are clustered into one super-category may have high visual similarity in feature space. Therefore, a multi-task metric learning algorithm is leveraged to jointly learn their inter-related classifiers to separate the sibling child nodes under the root node, e.g., Fig. 4(a) shows the root node

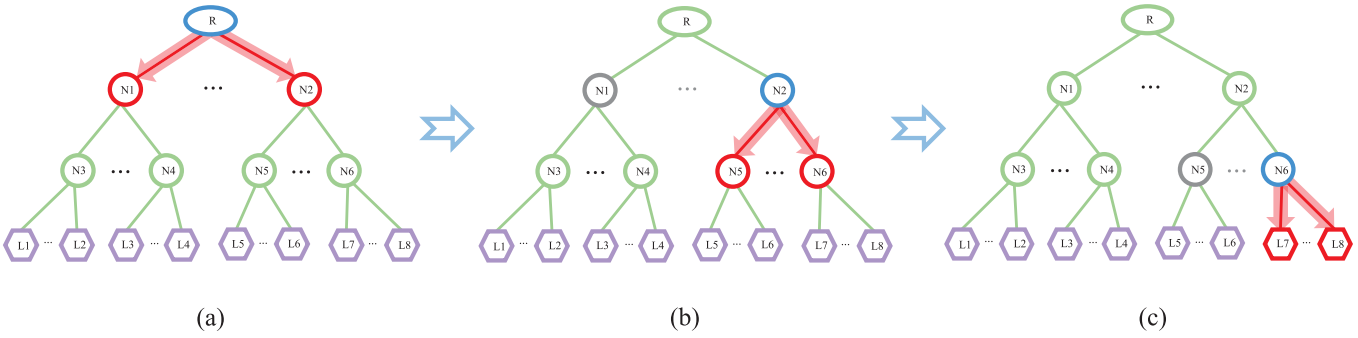


Fig. 4. The process of training a hierarchical classifier over a visual tree in a top-to-bottom way. The elliptical nodes represent the root nodes; the circular nodes represent the non-root nodes; and the hexagonal nodes represent the leaf nodes. The blue node represents the node classifier; the red nodes represent the child nodes; and the gray nodes represent the unrelated nodes. Subfigure (a) represents the root node classifier; subfigure (b) represents the non-root node classifier; and subfigure (c) represents the leaf node classifier.

classifier. The objective function of multi-task metric learning is defined as:

$$\begin{aligned}\min_{W_0,\ldots,W_T} \quad & \sum_{t=1}^{T}\left[\gamma_t \left\|W_t\right\|_F^2 + \sum_{i,j} d_t^2(x_i,x_j) + \sum_{i,j,k}\xi_{i,j,k}\right] \\ & + \gamma_o \left\|W_0 - I\right\|_F^2 \\ s.t.\ & d_t^2(x_i,x_k) - d_t^2(x_i,x_j) \geq 1 - \xi_{ijk} \\ & \xi_{ijk} \geq 0 \\ & W_0, W_1, \ldots, W_T \succeq 0 \end{aligned} \tag{8}$$

where x_i is the target sample, x_j is an exemplar from the same category, x_k is an exemplar from a different category, and γ_0 and γ_t are trade-off parameters used to adjust the common metric W_0 and the individual metric W_t, $t = 1, \ldots, T$. In the extreme cases, if $\gamma_t = 0$, only the common metric W_0 is learned, and the multi-task metric learning will reduce to the standard distance metric learning; in contrast, if $\gamma_0 = 0$, then only individual metric W_t is learned, and the multi-task metric learning will reduce to multiple single-task distance metric learning.

For the root node, we treat each of its child nodes as one task, and a multi-task metric learning algorithm is employed to learn the corresponding node classifier so that such sibling child nodes under the root node can be separated more effectively. Through our multi-task metric learning algorithm, both the common visual similarity and the node-specific visual difference are characterized. Therefore, by leveraging the clustering algorithm to generate super-categories and automatically determine the related learning tasks, our multi-task metric learning algorithm can obtain higher discrimination power to effectively distinguish the sibling child nodes under the root node.

D. Hierarchical Metric Learning for Non-Root Nodes

The visual tree can provide a good environment to identify the related tasks. For a non-root node, its sibling child nodes share some common visual properties because they are assigned to the same parent node. Thus, as introduced in the previous section, a multi-task metric learning algorithm can be used to learn the node classifier for such a non-root node to effectively separate its sibling child nodes. It is worth noting that the visual tree can provide a good environment for effectively identifying both the related tasks and the unrelated tasks. For one non-root node, its sibling child nodes may share similar visual properties, and learning the metrics for its sibling child nodes can be treated as the related tasks such that a common metric is learned to characterize their common visual properties; on the other hand, the visual similarities between its sibling child nodes and the child nodes from other non-root nodes could be very weak, and we can consider them as unrelated tasks. We can add this prior knowledge into multi-task metric learning to learn more discriminative metrics and node classifiers.

We can observe that the related tasks share the common metric and that the unrelated tasks may rely on a different metric. To characterize such effects, an orthogonal regularization term is added into the objective function of multi-task metric learning. For a non-root node n, let $S(n)$ denote its sibling nodes. The orthogonal regularization term is defined as:

$$\Omega(W_0) = \left\|W_u^T W_0\right\|_F^2 \tag{9}$$

where $u \in S(n)$, W_u is the individual metric of sibling nodes, and it is a known term. W_0 is the common metric shared by all the sibling child nodes under the same non-root node n.

The regularization term forces the common metric for all the sibling child nodes of n to be orthogonal to the individual metrics for the nodes in $S(n)$ as much as possible. Thus, the objective function is:

$$\begin{aligned}\min_{W_0,\ldots,W_T} \quad & \sum_{t=1}^{T}\left[\gamma_t \left\|W_t\right\|_F^2 + \sum_{i,j} d_t^2(x_i,x_j) + \sum_{i,j,k}\xi_{i,j,k}\right] \\ & + \sum_{u\in S(n)} \alpha \left\|W_u^T W_0\right\|_F^2 + \gamma_o \left\|W_0 - I\right\|_F^2 \end{aligned} \tag{10}$$

Theorem 1: The optimization problem in Eq. 10 is convex.

We are interested in proving that the optimization problem in Eq. 10 is convex. Reference [22] has proved that the optimization problem in Eq. 8 is convex. According to the principle of operations that preserve convexity, a nonnegative weighted sum of convex functions is convex. Because Eq. 10

is equal to Eq. 8 plus the function Ω and α is a weighting parameter > 0, it is only necessary to prove that function Ω is convex. The result of the proof can be found in the Appendix.[1]

In our visual tree, there is another inter-level visual correlation we can utilize, e.g., for a non-root node, its common metric can be borrowed from the individual metric of its parent node on the visual tree. Thus, we may borrow the individual metric from the parent node as a regularization term for its node metric. This inter-level regularization term is defined as:

$$\left\| W_0 - W_p \right\|_F^2 \tag{11}$$

where W_p is the individual metric for parent node p at the upper level of the visual tree. By this inter-level regularization term, we force the individual metric for the parent node to be as close to the common metric for all its sibling child nodes as possible. Since the inter-level regularization term is similar to the last term of Eq. 10, we combine them into one. Therefore, for the sibling child nodes under the same parent node, the objective function of the hierarchical multi-task metric learning is defined as:

$$\begin{aligned} \min_{W_0,\ldots,W_T} & \sum_{t=1}^{T} \left[\gamma_t \left\| W_t \right\|_F^2 + \sum_{i,j} d_t^2(x_i, x_j) + \sum_{i,j,k} \xi_{i,j,k} \right] \\ & + \sum_{\mu \in S(n)} \alpha \left\| {W_u}^T W_0 \right\|_F^2 + \gamma_0 \left\| W_0 - W_p \right\|_F^2 \\ s.t.\ & d_t^2(x_i, x_k) - d_t^2(x_i, x_j) \geq 1 - \xi_{ijk} \\ & \xi_{ijk} \geq 0 \\ & W_0, W_1, \ldots, W_T \succeq 0 \end{aligned} \tag{12}$$

where γ_0 is used to control the inter-level regularization. When the multi-task metrics are available for the sibling non-root nodes at the current level, a level-by-level process is adopted to recursively learn the multi-task metrics at the next level of the visual tree until the leaf nodes are reached.

By employing the visual tree to identify the related tasks and unrelated tasks, our hierarchical multi-task metric learning algorithm can utilize both related and unrelated tasks to train the multi-task metrics for non-root nodes. This prior knowledge may help us learn more discriminative metrics over a visual tree. As shown in Fig. 4(b) and Fig. 4(c), we utilize inter-level visual correlations to propose a regularization term that forces the inter-unrelated tasks to use different features. Moreover, by forcing the individual metric for the parent node to be close to the common metric for its child nodes, we can obtain more accurate results at the early stage for hierarchical image classification, which may effectively control the inter-level error propagation.

IV. Experimental Results and Analysis

In this section, we evaluate the proposed approach on three data sets: one synthetic data set and two public real data sets. We first investigate the performance of the proposed multi-task metric learning that utilizes both related and unrelated tasks on the synthetic data set. Then, we evaluate the performance of our hierarchical multi-task metric learning on two public real data sets.

[1]An indirect proof can also be obtained by employing [16], [23]

A. Synthetic Data

One open issue for multi-task learning is determining how to identify the related tasks or unrelated tasks. Since it is difficult to find an appropriate data set that contains both related tasks and unrelated tasks, we create a synthetic data set for algorithm evaluation. The synthetic data set is created as follows. The data set contains 6 tasks and can be divided into two groups, the main group and auxiliary group, where each group contains 3 related tasks, and the tasks between the two groups are unrelated. The dimension of the samples is $d = 20$. For the 3 related tasks in the main group, the first 10 dimensions are useful for related tasks, and the rest are not important. Each sample in the main group is represented as $(x_1, \ldots, x_{10}, 0, \ldots, 0)$, and each component x_i is generated by $N(0, 1)$. The coefficient vectors of the 3 related tasks are:

$$\begin{aligned} \delta_1 &= [1\ 2\ 3\ 1\ 2\ 3\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0], \\ \delta_2 &= [1\ 2\ 3\ 1\ 2\ 3\ 1\ 0\ 2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0], \\ \delta_3 &= [1\ 2\ 3\ 1\ 2\ 3\ 0\ 1\ 1\ 2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]. \end{aligned}$$

For each sample, the label is generated by $y = sign(\delta_i x + n_{wg})$, where n_{wg} is white Gaussian noise with $\sigma_{wg} = 0.1$.

For the auxiliary group, the first 10 dimensions of each sample are not important, and the rest are useful for related tasks. Each sample in the auxiliary group is represented as $(0, \ldots, 0, x_{11}, \ldots, x_{20})$. The coefficient vectors of the 3 related tasks in the auxiliary group are:

$$\begin{aligned} \delta_4 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 2\ 3\ 1\ 2\ 3\ 0\ 0\ 1\ 1], \\ \delta_5 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 2\ 3\ 1\ 2\ 3\ 1\ 0\ 2\ 0], \\ \delta_6 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 2\ 3\ 1\ 2\ 3\ 0\ 1\ 1\ 2]. \end{aligned}$$

We generate a total of 2,400 samples, where each task contains 400, half of which are used for training and half for testing.

1) Experimental Settings: In our multi-task metric learning, as shown in Eq. 10, there are three parameters, α, γ_0 and γ_t, that affect the results of the algorithm. Therefore, we seek an appropriate choice of these three parameters through the experiments. Since γ_0 and γ_t are trade-off parameters, we set $\gamma_0 = 1$ and only change γ_t. The experimental results are shown in Fig. 5. The x-axis indicates the value of the parameter. Since the values have magnitude differences, we use a logarithmic coordinate system for the x-axis. The y-axis indicates the classification accuracy rates (%). According to the experimental results, we set $\alpha = 10$, $\gamma_0 = 1$ and $\gamma_t = 100$.

2) Experiments on Different Approaches: In this experiment, we compare our proposed approach with 3 different nearest neighbor-based approaches; each of them is briefly introduced as follows.

Euclid: This is the baseline method of the kNN classifier, which utilizes the Euclidean distance to measure the distance between the samples.

LMNN: Weinberger et al. presented a Mahalanobis distance function for the kNN classifier by utilizing a triplet loss

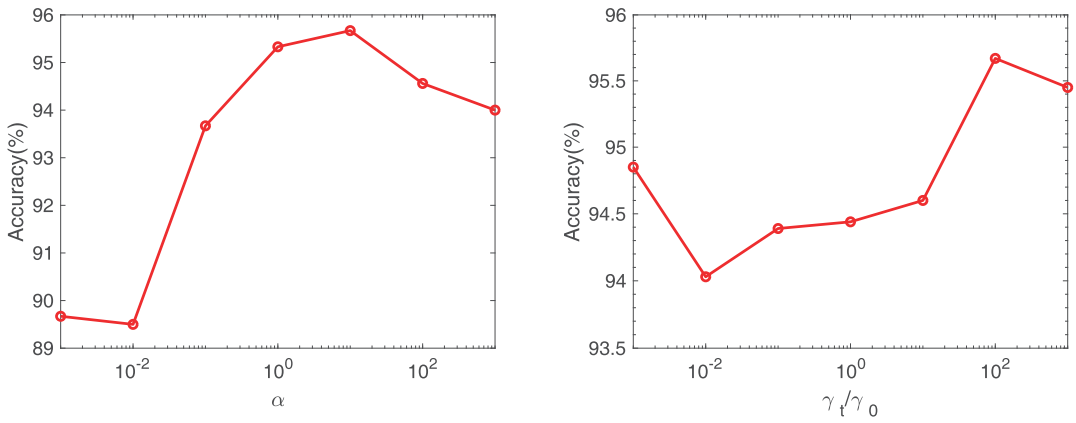


Fig. 5. The effect of parameters α and γ_t/γ_0 in the multi-task metric algorithm.

TABLE I
COMPARISON OF THE DIFFERENT NEAREST NEIGHBOR-BASED APPROACHES AND THE PROPOSED MTRU ON A SYNTHETIC DATA SET

Setting	Approaches	Accuracy (%)
Single-task	Euclid	80.50
	LMNN	85.67
Multi-task	MTML	91.17
	MTRU	95.67

that forces exemplars from the same class to cluster, while exemplars from different classes are separated [24].

MTML: Parameswaran et al. extended the LMNN to the multi-task setting and developed large-margin multi-task metric learning [22].

MTRU: This is the proposed approach, in which multi-task metric learning utilizes both related and unrelated tasks.

In our proposed approach, we first utilize a multi-task metric learning (MTML) algorithm to train the auxiliary group so that one common metric and three individual metrics can be obtained. We consider the tasks between the main group and the auxiliary group to be unrelated. Thus, we treat individual metrics of the auxiliary group as the unrelated tasks of the main group. When these individual metrics are available, we can train the proposed algorithm with the main group. The results of the comparison are shown in Table I. From the results, one can observe that the multi-task based approaches are significantly better than the single-task-based approaches because in our synthetic data set, the tasks in the main group are strongly related, and the multi-task learning can effectively utilize the common properties among tasks to improve the classification accuracy. In addition, one can observe that the proposed approach achieves the best results, especially much better than those of the MTML algorithm. The reason is that the proposed approach takes advantage of the prior knowledge of unrelated tasks, which can help us learn a more discriminative common metric for multi-task metric learning. Therefore, we can conclude that leveraging the advantages of both the related tasks and unrelated tasks can significantly enhance the performance of multi-task metric learning.

B. Real Data

We evaluate the proposed hierarchical classification on two real data sets: CIFAR-100 and ILSVRC-2012. The two data sets are briefly introduced as follows.

CIFAR-100 [55]: CIFAR-100 contains 100 categories, each with 600 samples, with 500 for training and 100 for testing. Each sample is a 32×32 color image, and it was transformed into a 3072-dimension vector.

ILSVRC-2012 [37]: The ILSVRC data set is a subset of ImageNet. ILSVRC-2012 contains 1,000 image categories, 1.2 million training images, 50 thousand validation images, and 150 thousand test images. Since the toolkit of the data set does not provide the ground truth of the test images, we therefore utilize validation images to evaluate all models.

In these two data sets, we utilize the visual tree to build the hierarchical structure and train the hierarchical classifier from top-to-bottom over the visual tree. Since deep learning has been proven to have strong feature learning capabilities [56], [57], we use the deep features [58], [59] to train all the classifiers on both data sets. The deep features are from the last fully connected layers of the CNN, which is designed for improving the performance of a variety of multiple task visual recognition applications [60]. We use the *mean accuracy* (%) to evaluate the performances of all approaches on both data sets.

1) Visual Tree Learning:

a) Experimental settings: For the visual tree construction, we employ AP clustering to measure the feature similarities of categories from top-to-bottom. The reason we chose the AP algorithm is that it does not specify the number of clustering centers. However, the preference parameter of AP clustering can control the number of cluster centers. In theory, the larger the preference parameter is, the smaller the number of cluster centers, whereas the smaller the parameter is, the greater the number of cluster centers. The choice of preference parameter will seriously affect the structure of the visual tree, and the structure of the visual tree will seriously affect the hierarchical classifier training, ultimately affecting the classification results. Therefore, we need to select the appropriate preference parameters for both the CIFAR-100 and ILSVRC-2012 data sets. The effect of the appropriate coefficient of the preference

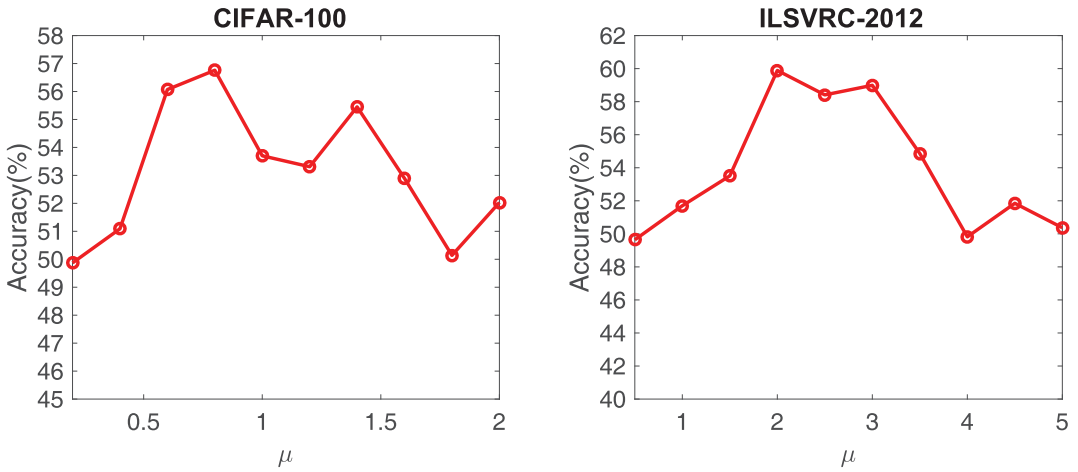
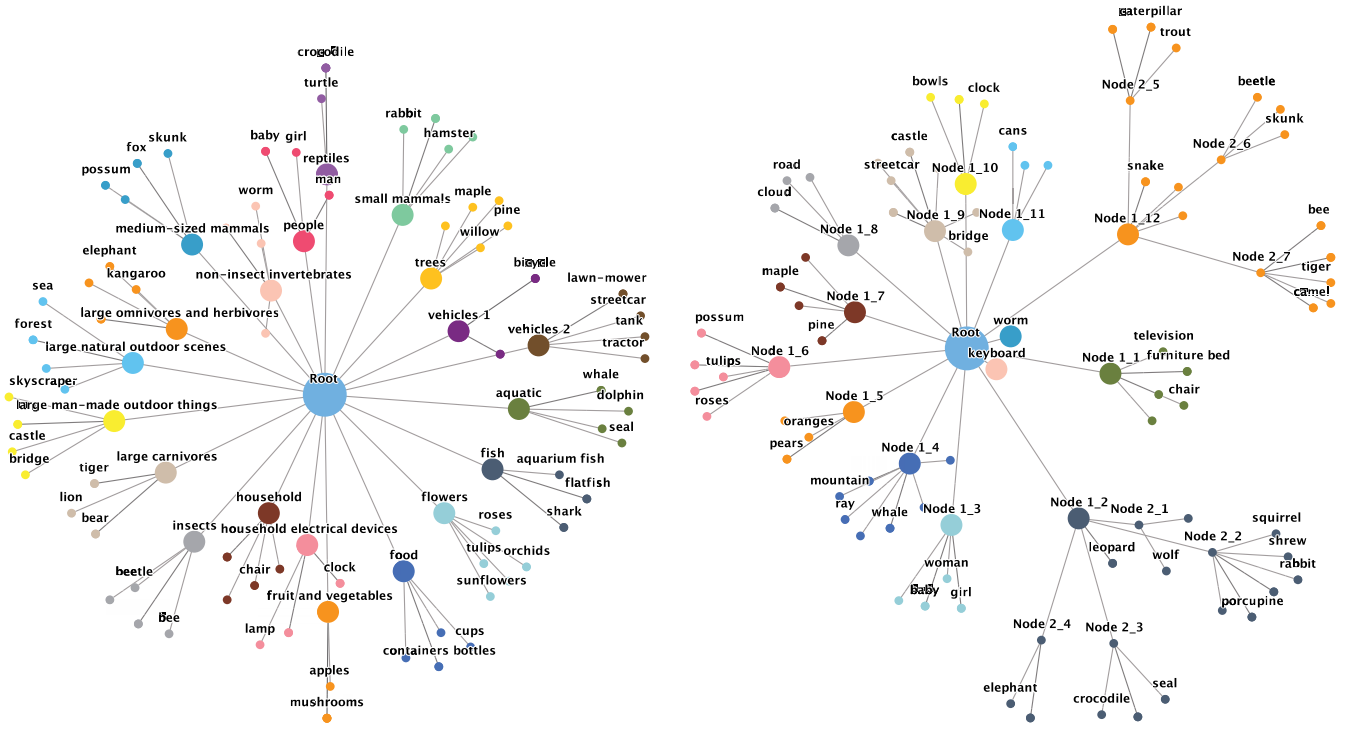


Fig. 6. The effect of the coefficient of the preference parameter μ on the hierarchical classifier.



(a) Semantic tree (b) Visual tree

Fig. 7. The semantic tree and visual tree of the CIFAR-100 data set.

parameter is shown in Fig. 6. We set the parameter μ as the coefficient of the preference parameter. The x-axis indicates the parameter μ. The y-axis indicates the classification accuracy rates (%) of our hierarchical classifier. One can observe that the curve changes irregularly. The reason is that only the tree structure reflects the true data distribution in feature space, and our hierarchical classifier can achieve better results. According to the findings in Fig. 6, we set the preference $\mu = 0.8$ for CIFAR-100 and $\mu = 2.0$ for ILSVRC-2012.

b) Experiments on the hierarchical structure: In this experiment, we compare the visual tree structure with the semantic tree structure. Fig. 7 illustrates the semantic tree and visual tree of the CIFAR-100 data set. From Fig. 7(a), we can observe that the semantic tree is a balanced tree structure. In contrast, the visual tree in Fig. 7(b) is an imbalanced structure. In the CIFAR-100 data set, 100 categories are forcibly grouped into 20 super-categories according to semantics. However, feature space is the common space employed in classifier training and image classification. In addition, similarity in semantic space does not mean that visual features are similar. As shown in Fig. 8, the category *forest* belongs to the super-category *large natural outdoor scenes*. However, it is difficult to say that *forest* has similarities in visual characteristics to *white clouds*, *plains*, *mountain* and *sea*, even

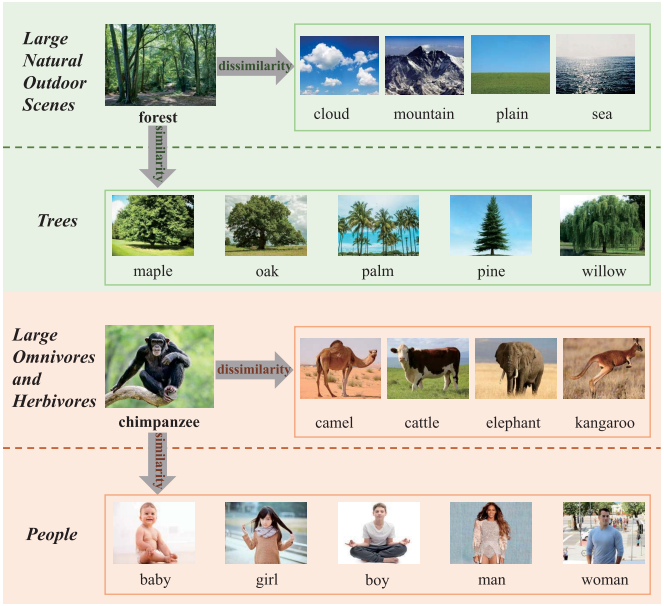


Fig. 8. Sample images of the CIFAR-100 data set in semantic space. (Due to the low image resolution of CIFAR-100, these sample images are from the Internet.)

TABLE II
COMPARISON OF DIFFERENT HIERARCHICAL STRUCTURES

Data set	CIFAR-100		ILSVRC-2012	
Approach	HSVM	HMRU	HSVM	HMRU
Semantic tree	46.62	48.97	44.47	53.23
Visual tree	48.68	56.76	49.81	59.86

if they belong to the same super-category. Correspondingly, since the forest is made up of trees, the category *forest* has visual similarity to all categories in the super-category of *trees*, so they are grouped into one super-category in the visual tree. Similarly, *chimpanzee* looks more similar to *people* than to other *large omnivores and herbivores*. In addition, since categories are usually imbalanced in feature space, it is reasonable to build an imbalanced visual tree structure.

The results of the comparison experiment between the semantic tree and visual tree are shown in Table II. The *CIFAR-100 data set* comes with own semantic tree structure. For the *ILSVRC-2012 data set*, we build the semantic tree structure by employing WordNet [36]. In this experiment, we employ two approaches to compare the visual tree and semantic tree. The baseline approach is a hierarchical SVM (HSVM). Another method is the proposed hierarchical multiple related and unrelated task learning (HMRU). We can observe that the visual tree structure can achieve better results than the semantic tree, regardless of the baseline approach or the proposed approach. Because the visual tree can construct a hierarchical structure according to the similarity in feature space, the hierarchical structure can more accurately characterize the distribution of the category features. Moreover, the semantic tree is constructed based on the semantic relationship of the category labels, and the semantic relationships of some categories are complex and confusing. For example, the super-category *large carnivores* in the CIFAR-100 data set has subcategories *leopard*, *bear*, *lion*, *tiger*, and *wolf* that may be far apart biologically, but because there are only a few carnivores in this data set, their semantic link is forcibly constructed to meet the rule of partitioning 100 categories evenly. This kind of imprecise hierarchical structure poses difficulties in helping the classifier achieve good results.

c) Experiments on correlations of tasks in the visual tree: To verify that the visual tree can effectively characterize the connections between image categories to help learn related and unrelated tasks, we use a matrix correlation coefficient to measure the degree of correlation between tasks; it is defined as:

$$corr = \frac{\sum_i \sum_j (W_{i,j} - \overline{W})(V_{i,j} - \overline{V})}{\sqrt{\left(\sum_i \sum_j (W_{i,j} - \overline{W})^2\right)\left(\sum_i \sum_j (V_{i,j} - \overline{V})^2\right)}} \quad (13)$$

where $\overline{W} = mean(W)$ and $\overline{V} = mean(V)$. The closer the coefficient is to 1, the higher the correlation.

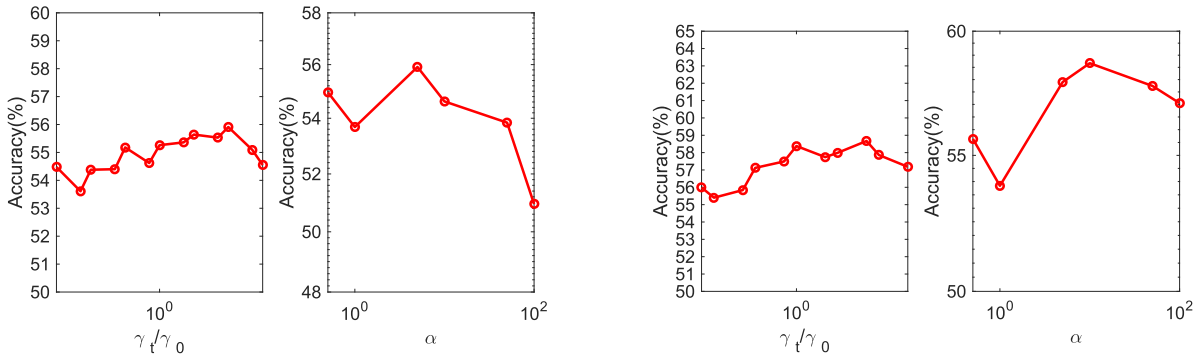
Table III illustrates the experimental results of correlations for different tasks on the CIFAR-100 data set. In this experiment, we select 2 intermediate nodes in the second layer of the visual tree of the CIFAR-100 data set, both of which contain 4 child nodes. According to the definition of the visual tree, all of the child nodes under a parent node are related tasks, and the child nodes under different parent nodes are unrelated tasks. We calculate the matrix correlation coefficient of the distance metric for all 8 tasks. According to the results in Table III, we can observe that the child nodes under the same parent node have strong correlations with the distance metric matrix, while the child nodes under different parent nodes have weak correlations. This result shows that the visual tree can effectively partition the related and unrelated tasks and, further, can help the training of hierarchical classifiers.

2) Hierarchical Classification:

a) Experimental settings: In this experiment, there are also three parameters, γ_0, γ_t and α, that can affect the experimental results, as shown in Eq. 12. Therefore, we select the appropriate parameters for both data sets through the experiments. Similarly, we set $\gamma_0 = 1$, and only change γ_t. In particular, we set all γ_t, $t = 1, \ldots, T$ to be equal because we believe that the contributions of all tasks are equal. The experimental results are shown in Fig. 9. The left side shows the results for CIFAR-100, and the right side shows the results for ILSVRC-2012. The first column shows the results of the effect of parameter γ_t/γ_0; the second column is the results for parameter α. The x-axis indicates the value of the parameters. We also use a logarithmic coordinate system for the x-axis. The y-axis indicates the classification accuracy rates (%). In this experiment, the parameter γ_t/γ_0 is chosen from [0.005 0.01 0.05 0.1 0.5 1 5 10 50 100 500], and parameter α is chosen from [0.5 1 5 10 50 100]. From the results, we can observe that parameter γ_t has a slight effect on the results, whereas parameter α can more obviously affect the results. The results show that our orthogonal regularization term has a great influence on the hierarchical classifier. According to the

TABLE III
THE CORRELATION RESULTS FOR DIFFERENT TASKS ON THE CIFAR-100 DATA SET

corr	Task-one-1	Task-one-2	Task-one-3	Task-one-4	Task-two-1	Task-two-2	Task-two-3	Task-two-4
Task-one-1	1.0000	0.9138	0.8202	0.7518	0.5852	0.5701	0.5774	0.5753
Task-one-2	0.9138	1.0000	0.8475	0.7567	0.5814	0.5962	0.5813	0.5694
Task-one-3	0.8202	0.8475	1.0000	0.7482	0.5774	0.5851	0.5692	0.5671
Task-one-4	0.7518	0.7567	0.7482	1.0000	0.6015	0.5988	0.5860	0.5910
Task-two-1	0.5852	0.5814	0.5774	0.6015	1.0000	0.9427	0.9169	0.9078
Task-two-2	0.5701	0.5962	0.5851	0.5988	0.9427	1.0000	0.8972	0.9024
Task-two-3	0.5774	0.5813	0.5692	0.5860	0.9169	0.8972	1.0000	0.9132
Task-two-4	0.5753	0.5694	0.5671	0.5910	0.9078	0.9024	0.9132	1.0000



(a) CIFAR-100 data set (b) ILSVRC-2012 data set

Fig. 9. The effect of parameters on the hierarchical classifier.

experimental results in Fig. 9, we set $\gamma_t/\gamma_0 = 100$ and $\alpha = 5$ for the CIFAR-100 data set and $\gamma_t/\gamma_0 = 50$ and $\alpha = 10$ for the ILSVRC-2012 data set.

b) Experiments on different approaches: We compare the proposed approach with several related approaches, and these approaches can be divided into two types: flat approaches and hierarchical approaches. Both of them are briefly introduced as follows.

(1) Flat approaches

Euclid: The baseline method of the kNN classifier, which utilizes the Euclidean distance to measure the distance between the samples.

LMNN: Weinberger et al. presented a Mahalanobis distance function for the kNN classifier by utilizing a triplet loss that forces exemplars from same class to be clustered, while exemplars from different classes are separated [24].

(2) Hierarchical approaches

HLMM: Lei et al. developed hierarchical metric learning that employs a large-margin nearest neighbors algorithm to learn node classifiers [18].

HMTL: A hierarchical multi-task metric algorithm that employs large-margin multi-task metric learning to learn node classifiers; this approach does not leverage the inter-level correlations [22].

ToM: A hierarchical metric algorithm that employs inter-level correlations to learn a tree of disjoint metrics [17].

HMML: A hierarchical multi-task metric algorithm that employs multi-task metric learning to learn node classifiers; in addition, this approach has a regularization term developed by leveraging the inter-level correlations [15].

CNN: The convolutional neural network with AlexNet; the reason we chose this model is that it has the same structure as the feature extraction algorithm in this experiment. We use the *Matconvnet* [61] deep learning toolbox[1] to implement this algorithm.

TABLE IV
COMPARISON ON THE CIFAR-100 DATA SET

Structure	Approaches	Mean Accuracy (%)
Flat	Euclid	46.03
	LMNN	48.84
Hierarchical	HLMM	50.16
	HMTL	52.78
	ToM	53.02
	CNN	54.20
	HMML	54.33
	HMRU	56.76

(A) CIFAR-100

The experimental results on the *CIFAR-100* data set are shown in Table IV. First, we can clearly observe that the hierarchical classifiers can achieve better results than the flat classifiers because the hierarchical classifiers can effectively separate the dissimilar categories as early as possible and can solve the data imbalance problem of flat classifiers. Second, the hierarchical approaches that utilize structural prior knowledge often achieve better results. HMTL, which leverages inter-category correlations in multi-task metric learning to learn a node classifier, can achieve better performance than HLMM, which does not employ the multi-task setting. Similarly, ToM, which utilizes inter-level correlations to train a tree of metrics, also achieves better performance than HLMM. In addition, HMML utilizes both the inter-category correlations and inter-level correlations and achieves better

[1] http://www.vlfeat.org/matconvnet/

TABLE V
COMPARISON ON THE ILSVRC-2012 DATA SET

Structure	Approaches	Mean Accuracy (%)
Flat	Euclid	44.09
	LMNN	45.72
Hierarchical	HLMM	51.67
	HMTL	52.94
	ToM	56.62
	CNN	56.69
	HMML	58.76
	HMRU	59.89
	CNN(GoogLeNet)	66.30
	HMRU(GoogLeNet)	66.82

performance. Third, the proposed HMRU approach achieves the best performance in this experiment because the proposed method leverages both the related and unrelated tasks to train the hierarchical classifier. Moreover, the above approaches always search for the inter-category visual similarity to determine the related tasks. However, the visual tree not only provides us with a good environment for identifying the related tasks but also provides a good environment for identifying the unrelated tasks. By employing the unrelated task as prior knowledge, the proposed approach can learn a more discriminative classifier. Additionally, the CNN does not outperform all the hierarchical approaches because although it can learn powerful feature representations, it ignores the inter-relations among categories at the training phase.

(B) ILSVRC-2012

The experimental results on the *ILSVRC-2012* data set are shown in Table V. From the results, conclusions similar to those based on the *CIFAR-100* data set can be drawn. One can observe that the proposed approach has advantages over the other approaches.

To further evaluate the proposed approach, we apply the proposed HMRU approach to GoogLeNet [62] framework. In this experiment, we employ GoogLeNet as the feature extractor and then integrate the proposed HMRU approach as the classifier. The original GoogLeNet is used as the baseline approach. We can observe that proposed approach can achieve a better result than the baseline approach. However, the improvement is not obvious, which means that the feature representation of the baseline approach is very powerful, and a good classification result can be achieved with a simple classifier.

V. CONCLUSION

In this paper, a hierarchical multi-task metric learning algorithm is developed for image classification, where both the related tasks and unrelated tasks are utilized to train hierarchical multi-task metrics over a visual tree. Both the inter-category visual correlations and the inter-level visual correlations are utilized to train hierarchical multi-task metrics. By utilizing this structural information, more discriminative hierarchical multi-task metrics can be learned for large-scale image classification. Our proposed algorithms are evaluated over three data sets, demonstrating that our proposed approach outperforms the other hierarchical approaches in terms of classification accuracy.

APPENDIX
PROOF OF THEOREM 1

In this appendix, we present the proof of Theorem 1. The key is to prove that the function Ω is convex. For writing purposes, we change W_0 in function Ω to V; then, function Ω is defined as:

$$\Omega(V) = \left\| W_u^T V \right\|_F^2$$

We will employ the definition of convexity to prove the theorem. W_u is a known quantity, and $W_u \succeq 0$. We will show that for any $t \in [0, 1]$, the following inequality holds:

$$t\Omega(V_1) + (1-t)\Omega(V_2) \geq \Omega(tV_1 + (1-t)V_2)$$

We have:

$$t\Omega(V_1) = t \left\| {W_u}^T V_1 \right\|_F^2$$
$$(1-t)\Omega(V_2) = (1-t) \left\| {W_u}^T V_2 \right\|_F^2$$
$$\Omega(tV_1 + (1-t)V_2) = \left\| {W_u}^T (tV_1 + (1-t)V_2) \right\|_F^2$$

and:

$$\begin{aligned} &t\Omega(V_1) + (1-t)\Omega(V_2) - \Omega(tV_1 + (1-t)V_2) \\ &= t \left\| {W_u}^T V_1 \right\|_F^2 + (1-t) \left\| {W_u}^T V_2 \right\|_F^2 \\ &\quad - \left\| t{W_u}^T V_1 + (1-t){W_u}^T V_2 \right\|_F^2 \end{aligned}$$

According to the definition of the matrix norm, we can obtain $\|cA\| = |c| \|A\|, c \in \mathbb{R}$ and $\|A + B\| \leq \|A\| + \|B\|$;

then, we have:

$$\begin{aligned} &t\Omega(V_1) + (1-t)\Omega(V_2) - \Omega(tV_1 + (1-t)V_2) \\ &\geq t \left\| {W_u}^T V_1 \right\|_F^2 + (1-t) \left\| {W_u}^T V_2 \right\|_F^2 \\ &\quad - (t \left\| {W_u}^T V_1 \right\|_F + (1-t) \left\| {W_u}^T V_2 \right\|_F)^2 \\ &= t \left\| {W_u}^T V_1 \right\|_F^2 + (1-t) \left\| {W_u}^T V_2 \right\|_F^2 - t^2 \left\| {W_u}^T V_1 \right\|_F^2 \\ &\quad - (1-t)^2 \left\| {W_u}^T V_2 \right\|_F^2 - 2t(1-t) \left\| {W_u}^T V_1 \right\|_F \left\| {W_u}^T V_2 \right\|_F \\ &= t(1-t)(\left\| {W_u}^T V_1 \right\|_F^2 - 2 \left\| {W_u}^T V_1 \right\|_F \left\| {W_u}^T V_2 \right\|_F \\ &\quad + \left\| {W_u}^T V_2 \right\|_F^2) \\ &= t(1-t)(\left\| {W_u}^T V_1 \right\|_F - \left\| {W_u}^T V_2 \right\|_F)^2 \\ &\geq 0 \end{aligned}$$

The function Ω is convex; hence, this concludes the proof.

REFERENCES

[1] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*. [Online]. Available: https://arxiv.org/abs/1603.04467

[2] R. Babbar, I. Partalas, E. Gaussier, M.-R. Amini, and C. Amblard, "Learning taxonomy adaptation in large-scale classification," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 3350–3386, 2016.

[3] Y. Qu *et al.*, "Joint hierarchical category structure learning and large-scale image classification," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4331–4346, Sep. 2017.

[4] M. Ristin, M. Guillaumin, J. Gall, and L. van Gool, "Incremental learning of random forests for large-scale image classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 3, pp. 490–503, Mar. 2016.

[5] Z. Wen, B. Hou, and L. Jiao, "Discriminative dictionary learning with two-level low rank and group sparse decomposition for image classification," *IEEE Trans. Cybern.*, vol. 47, no. 11, pp. 3758–3771, Nov. 2017.

[6] T. Berg and P. N. Belhumeur, "POOF: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, Jun. 2013, pp. 955–962.

[7] Y. Chai, E. Rahtu, V. Lempitsky, L. Van Gool, and A. Zisserman, "TriCoS: A tri-level class-discriminative co-segmentation method for image classification," in *Proc. Eur. Conf. Comput. Vis.*, Florence, Italy, 2012, pp. 794–807.

[8] T. Zhu, Y. Lin, and Y. Liu, "Synthetic minority oversampling technique for multiclass imbalance problems," *Pattern Recognit.*, vol. 72, pp. 327–340, Dec. 2017.

[9] S. Gopal, "Large-scale structured learning," Ph.D. dissertation, Carnegie Mellon Univ., Pittsburgh, PA, USA, 2014.

[10] Y. Li, Q. Huang, W. Xie, and X. Li, "A novel visual codebook model based on fuzzy geometry for large-scale image classification," *Pattern Recognit.*, vol. 48, no. 10, pp. 3125–3134, Oct. 2015.

[11] B. Zhao, F. Li, and E. P. Xing, "Large-scale category structure aware image categorization," in *Proc. Adv. Neural Inf. Process. Syst.*, Granada, Spain, 2011, pp. 1251–1259.

[12] J. Fan, J. Zhang, K. Mei, J. Peng, and L. Gao, "Cost-sensitive learning of hierarchical tree classifiers for large-scale image classification and novel category detection," *Pattern Recognit.*, vol. 48, no. 5, pp. 1673–1687, 2015.

[13] D. Peralta, I. Triguero, S. García, Y. Saeys, J. M. Benitez, and F. Herrera, "Distributed incremental fingerprint identification with reduced database penetration rate using a hierarchical classification based on feature fusion and selection," *Knowl.-Based Syst.*, vol. 126, pp. 91–103, Jun. 2017.

[14] J. L. Bruse *et al.*, "Detecting clinically meaningful shape clusters in medical image data: Metrics analysis for hierarchical clustering applied to healthy and pathological aortic arches," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 10, pp. 2373–2383, Oct. 2017.

[15] Y. Zheng, J. Fan, J. Zhang, and X. Gao, "Hierarchical learning of multi-task sparse metrics for large-scale image classification," *Pattern Recognit.*, vol. 67, pp. 97–109, Jul. 2017.

[16] L. Xiao, D. Zhou, and M. Wu, "Hierarchical classification via orthogonal transfer," in *Proc. Int. Conf. Mach. Learn.*, Bellevue, WA, USA, 2011, pp. 801–808.

[17] K. Grauman, F. Sha, and S. J. Hwang, "Learning a tree of metrics with disjoint visual features," in *Proc. Adv. Neural Inf. Process. Syst.*, Granada, Spain, 2011, pp. 621–629.

[18] H. Lei, K. Mei, J. Xin, P. Dong, and J. Fan, "Hierarchical learning of large-margin metrics for large-scale image classification," *Neurocomputing*, vol. 208, pp. 46–58, Oct. 2016.

[19] S. Gopal and Y. Yang, "Recursive regularization for large-scale classification with hierarchical and graphical dependencies," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Chicago, IL, USA, 2013, pp. 257–265.

[20] A. Argyriou, T. Evgeniou, and M. Pontil, "Convex multi-task feature learning," *Mach. Learn.*, vol. 73, no. 3, pp. 243–272, 2008.

[21] F. Cai and V. Cherkassky, "Generalized SMO algorithm for SVM-based multitask learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 6, pp. 997–1003, Jun. 2012.

[22] S. Parameswaran and K. Q. Weinberger, "Large margin multi-task metric learning," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2010, pp. 1867–1875.

[23] B. Romera-Paredes, A. Argyriou, N. Berthouze, and M. Pontil, "Exploiting unrelated tasks in multi-task learning," in *Proc. Int. Conf. Artif. Intell. Statist.*, La Palma, Canary Islands, 2012, pp. 951–959.

[24] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *J. Mach. Learn. Res.*, vol. 10, pp. 207–244, Feb. 2009.

[25] Q. Qian, R. Jin, S. Zhu, and Y. Lin, "Fine-grained visual categorization via multi-stage metric learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 3716–3724.

[26] B. Babenko, S. Branson, and S. Belongie, "Similarity metrics for categorization: From monolithic to category specific," in *Proc. IEEE Int. Conf. Comput. Vis.*, Kyoto, Japan, Sep./Oct. 2009, pp. 293–300.

[27] L. Zhang, S. K. Shah, and I. A. Kakadiaris, "Hierarchical multi-label classification using fully associative ensemble learning," *Pattern Recognit.*, vol. 70, pp. 89–103, Oct. 2017.

[28] L.-J. Li, C. Wang, Y. Lim, D. M. Blei, and L. Fei-Fei, "Building and using a semantivisual image hierarchy," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, San Francisco, CA, USA, Jun. 2010, pp. 3336–3343.

[29] J. Fan, N. Zhou, J. Peng, and Y. Gao, "Hierarchical learning of tree classifiers for large-scale plant species identification," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4172–4184, Nov. 2015.

[30] M. Marszalek and C. Schmid, "Constructing category hierarchies for visual recognition," in *Proc. Eur. Conf. Comput. Vis.*, Marseille, France, 2008, pp. 479–491.

[31] S. Bengio, J. Weston, and D. Grangier, "Label embedding trees for large multi-class tasks," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2010, pp. 163–171.

[32] J. Deng, S. Satheesh, A. C. Berg, and F. Li, "Fast and balanced: Efficient label tree learning for large scale object recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, Granada, Spain, 2011, pp. 567–575.

[33] B. Liu, F. Sadeghi, M. Tappen, O. Shamir, and C. Liu, "Probabilistic label trees for efficient large scale image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, Jun. 2013, pp. 843–850.

[34] M. Naphade *et al.*, "Large-scale concept ontology for multimedia," *IEEE MultiMedia*, vol. 13, no. 3, pp. 86–91, Jul./Sep. 2006.

[35] G. Griffin and P. Perona, "Learning and using taxonomies for fast visual categorization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Anchorage, AK, USA, Jun. 2008, pp. 1–8.

[36] G. A. Miller, "WordNet: A lexical database for English," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[37] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Jun. 2009, pp. 248–255.

[38] X. Zhang, F. Zhou, Y. Lin, and S. Zhang, "Embedding label structures for fine-grained feature representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 1114–1123.

[39] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, New York, NY, USA, Jun. 2006, pp. 2161–2168.

[40] H. Zhao, P. Zhu, P. Wang, and Q. Hu, "Hierarchical feature selection with recursive regularization," in *Proc. Int. Joint Conf. Artif. Intell.*, Melbourne, NSW, Australia, 2017, pp. 3483–3489.

[41] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in *Proc. Eur. Conf. Comput. Vis.*, Zürich, Switzerland, 2014, pp. 94–108.

[42] Y. Luo, Y. Wen, D. Tao, J. Gui, and C. Xu, "Large margin multi-modal multi-task feature extraction for image classification," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 414–427, Jan. 2016.

[43] A.-A. Liu, Y.-T. Su, W.-Z. Nie, and M. Kankanhalli, "Hierarchical clustering multi-task learning for joint human action grouping and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 1, pp. 102–114, Jan. 2017.

[44] H. Wang and N. Ahuja, "Facial expression decomposition," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, Nice, France, Oct. 2003, pp. 958–965.

[45] E. P. Xing, M. I. Jordan, S. J. Russell, and A. Y. Ng, "Distance metric learning with application to clustering with side-information," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2003, pp. 521–528.

[46] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *Proc. Int. Conf. Mach. Learn.*, Corvalis, OR, USA, 2007, pp. 209–216.

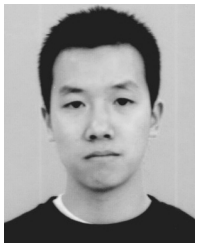
[47] M. P. Kumar, P. H. S. Torr, and A. Zisserman, "An invariant large margin nearest neighbour classifier," in *Proc. IEEE Int. Conf. Comput. Vis.*, Rio de Janeiro, Brazil, Oct. 2007, pp. 1–8.

[48] N. Verma, D. Mahajan, S. Sellamanickam, and V. Nair, "Learning hierarchical similarity metrics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 2280–2287.

[49] W. Zuo *et al.*, "Distance metric learning via iterated support vector machines," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4937–4950, Oct. 2017.

[50] G. Cheng, P. Zhou, and J. Han, "Duplex metric learning for image set classification," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 281–292, Jan. 2018.

[51] L. Ma, X. Yang, and D. Tao, "Person re-identification over camera networks using multi-task distance metric learning," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3656–3670, Aug. 2014.
[52] B. Bhattarai, G. Sharma, and F. Jurie, "CP-mtML: Coupled projection multi-task metric learning for large scale face retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 4226–4235.
[53] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, Feb. 2007.
[54] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, vol. 317. Springer, 2009.
[55] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf
[56] F. Wang *et al.*, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, Jul. 2017, pp. 3156–3164.
[57] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.
[58] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," 2014, *arXiv:1408.5093*. [Online]. Available: https://arxiv.org/abs/1408.5093
[59] J. Donahue *et al.*, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn.*, Beijing, China, 2014, pp. 647–655.
[60] B. Du, W. Xiong, J. Wu, L. Zhang, L. Zhang, and D. Tao, "Stacked convolutional denoising auto-encoders for feature representation," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 1017–1027, Apr. 2016.
[61] A. Vedaldi and K. Lenc, "MatConvNet: Convolutional neural networks for MATLAB," in *Proc. 23rd ACM Int. Conf. Multimedia*, Brisbane, QLD, Australia, 2015, pp. 689–692.
[62] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 1–9.



Yu Zheng received the B.Eng. degree in electronic information engineering from Xidian University, Xi'an, China, in 2012, and the Ph.D. degree in intelligent information processing from the VIP Laboratory, School of Electronic Engineering, Xidian University in 2017. In 2018, he joined the School of Cyber Engineering, Xidian University. His current research interests include machine learning and computer vision.

Jianping Fan received the M.Sc. degree in theory physics from Northwestern University, Xi'an, China, in 1994, and the Ph.D. degree in optical storage and computer science from the Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Shanghai, China, in 1997. He was a Postdoctoral Researcher with Fudan University, Shanghai, from 1997 to 1998. From 1998 to 1999, he was a Researcher with the Japan Society of Promotion of Science (JSPS), Department of Information System Engineering, Osaka University, Osaka, Japan. From 1999 to 2001, he was a Postdoctoral Researcher with the Department of Computer Science, Purdue University, West Lafayette, IN, USA. He is currently a Professor with the University of North Carolina at Charlotte. His research interests include image/video privacy protection, automatic image/video understanding, and large-scale deep learning.

Ji Zhang received the B.S. degree in computer science and technology from Xi'an Jiaotong University, Xi'an, China, in 2012, where he is currently pursuing the Ph.D. degree with the Institute of Artificial Intelligence and Robotics. He was a Visiting Student with the University of North Carolina at Charlotte from 2015 to 2016. His research interests include computer vision and large-scale machine learning.

Xinbo Gao (M'02–SM'07) received the B.Eng., M.Sc., and Ph.D. degrees in signal and information processing from Xidian University, Xi'an, China, in 1994, 1997, and 1999, respectively. From 1997 to 1998, he was a Research Fellow with the Department of Computer Science, Shizuoka University, Shizuoka, Japan. From 2000 to 2001, he was a Postdoctoral Research Fellow with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong. Since 2001, he has been with the School of Electronic Engineering, Xidian University. He is currently a Cheung Kong Professor with the Ministry of Education of China and a Professor of pattern recognition and intelligent system and the Dean of the Graduate School, Xidian University. He has published six books and around 300 technical articles in refereed journals and proceedings. His current research interests include image processing, computer vision, multimedia analysis, machine learning, and pattern recognition. He is also a fellow of the Institute of Engineering and Technology and the Chinese Institute of Electronics. He has served as the general chair/co-chair, the program committee chair/co-chair, or a PC member for around 30 major international conferences. He is on the editorial boards of several journals, including *Signal Processing* (Elsevier) and *Neurocomputing* (Elsevier).