iPrivacy: Image Privacy Protection by Identifying Sensitive Objects via Deep Multi-Task Learning

Jun Yu, *Member, IEEE*, Baopeng Zhang, Zhengzhong Kuang, Dan Lin, and Jianping Fan

***Abstract*— To achieve automatic recommendation of privacy settings for image sharing, a new tool called *iPrivacy* (image privacy) is developed for releasing the burden from users on setting the privacy preferences when they share their images for special moments. Specifically, this paper consists of the following contributions: 1) massive social images and their privacy settings are leveraged to learn the object-privacy relatedness effectively and identify a set of privacy-sensitive object classes automatically; 2) a deep multi-task learning algorithm is developed to jointly learn more representative deep convolutional neural networks and more discriminative tree classifier, so that we can achieve fast and accurate detection of large numbers of privacy-sensitive object classes; 3) automatic recommendation of privacy settings for image sharing can be achieved by detecting the underlying privacy-sensitive objects from the images being shared, recognizing their classes, and identifying their privacy settings according to the object-privacy relatedness; and 4) one simple solution for image privacy protection is provided by blurring the privacy-sensitive objects automatically. We have conducted extensive experimental studies on real-world images and the results have demonstrated both the efficiency and effectiveness of our proposed approach.**

***Index Terms*— Image sharing, privacy setting recommendation, object-privacy alignment, image privacy protection, privacy-sensitive object classes, deep multi-task learning, tree classifier for hierarchical object detection.**

I. Introduction

With the growing popularity of smart-phones and other mobile devices, high-quality cameras are increasingly pervasive. As a result, capturing images and sharing them on social platforms like Facebook, Instagram and Foursquare has become a common part of our daily life. However, without the proper privacy protection [1]–[4], the shared images can reveal much of users' personal and social environments and their private lives since images can intuitively tell when and where a special moment took place, who participated and what were their relationships. Unfortunately, many people especially young users of social networks often share private images about themselves, their friends and classmates without being aware of the potential impact on their future lives caused by unwanted disclosure and privacy violations.

With the increasing concerns on image privacy [5]–[16], major social websites start offering privacy tools that allow users to manually specify coarse-grained privacy settings (preferences), such as whether an image is public, private or visible to their family members or friends. However, due to the lack of privacy knowledge, it would not be easy for common users to correctly configure privacy settings to achieve their desired levels of privacy protection; also, given the large number of images being shared and the tedious steps needed for fine-grained privacy settings, some users may not be willing to spend extra time on providing such fine-grained privacy settings [1]–[4].

To release the privacy setting burden from users, a new tool called ***iPrivacy*** (image Privacy) is developed to automate the privacy configuration process during social image sharing. Unlike many previous works [5]–[13] which typically recommend privacy settings based on similarity of users' profiles or image tags, we study the problem in a different angle by looking into the shared images themselves. Our idea is to automatically detect the privacy-sensitive objects from the images being shared, recognize their classes, and identify their privacy settings. Based on the detection results, our system would be able to warn the image owners what objects in the images need to be protected before sharing and also provide recommended privacy settings. For example, Fig. 1 shows two categories of privacy-sensitive object classes: (a) *user-independent classes* such as humans, locations and discrimination texts in images; and (b) *user-dependent classes* such as home shrines and visual attributes for personal hobbies. Considering 1.82 billions active users of social networks and trillions of shared images, there may exist a large set of privacy-sensitive object classes. Thus, the critical challenge to be conquered here is to identify all the privacy-sensitive object classes *efficiently* and learn the object-privacy relatedness *precisely* from the large number of social images so as to provide real-time privacy recommendation. To achieve this, we cannot directly adopt

Manuscript received July 24, 2016; revised September 30, 2016 and October 31, 2016; accepted November 6, 2016. Date of publication December 6, 2016; date of current version February 22, 2017. This work was supported by the National Science Foundation under Grant 1651166-CNS and Grant 1651455-CNS. The work of J. Yu was supported in part by the National Natural Science Foundation of China under Grant 61622205 and in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LR15F020002. The Associate Editor coordinating the review of this manuscript and approving it for publication was Prof. Stefano Tubaro. *(Corresponding author: Jianping Fan.)*

J. Yu was with UNC Charlotte, Charlotte, NC 28223 USA. He is now with the School of Computer Science, Hangzhou Dianzi University, Hangzhou 310018, China (e-mail: yujun@hdu.edu.cn).

B. Zhang was with UNC Charlotte, Charlotte, NC 28223 USA. He is now with the School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China (e-mail: bpzhang@bjtu.edu.cn).

Z. Kuang is with the College of Computer and Communication Engineering, China University of Petroleum, Qingdao 266580, China, and also with UNC Charlotte, Charlotte, NC 28223 USA (e-mail: zkung@uncc.edu).

D. Lin is with the Department of Computer Science, Missouri University of Science and Technology, Rolla, MO 65409 USA (e-mail: lindan@mst.edu).

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

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIFS.2016.2636090

Fig. 1. Illustration of some privacy-sensitive object classes: (a) user-independent classes: (a1) sensitive people, (a2) sensitive locations, (a3) public toilet, (a4) discrimination texts; (b) user-dependent classes: (b1) home shrines which may indicate personal religions, (b2) visual attributes which can indicate smoking in public, (b3) conference name tags, (b4) personal information tags.

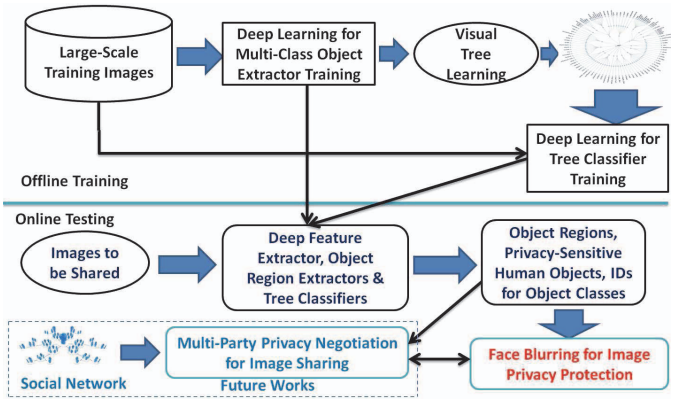

Fig. 2. An Overview of the key components of our iPrivacy system.

existing machine learning tools. Specifically, if a flat approach is employed, the computational cost will grow linearly with the total number of privacy-sensitive object classes (to be detected and recognized) and hence it is not scalable; if a hierarchical approach [19], [20] is adopted, the object detection process could be speed up dramatically but it would seriously suffer from the so-called *inter-level error propagation* problem, i.e., the mistakes made at the parent nodes will propagate to their child nodes and such mistakes cannot be recovered.

To address the aforementioned challenges, our iPrivacy system takes four main steps (as illustrated in Fig. 2): (1) Deep CNNs are learned to achieve semantic image segmentation and identify large numbers of object classes from massive social images, and an automatic object-privacy alignment algorithm is developed to learn the object-privacy relatedness and identify a set of privacy-sensitive object classes; (2) A visual tree is learned to organize large numbers of privacy-sensitive object classes hierarchically in a coarse-to-fine fashion, which can provide a good environment to determine the inter-related learning tasks automatically; (3) A deep multi-task learning algorithm is developed to learn more representative deep CNNs and more discriminative tree classifier jointly, so that we can achieve fast and accurate detection of large numbers of privacy-sensitive object classes; (4) Automatic recommendation of privacy settings for image sharing is achieved by detecting the underlying privacy-sensitive objects from the images being shared, recognizing their classes, and identifying their privacy settings according to the object-privacy relatedness. In addition, a simple solution for image privacy protection is further provided by blurring the privacy-sensitive objects automatically. Finally, to evaluate the performance of our proposed iPrivacy system, we have conducted extensive experimental studies on real-world images. The experimental results have demonstrated both efficiency and effectiveness of our proposed approach.

The remaining of the paper is organized as follows. Section 2 reviews the related work briefly; Section 3 presents our algorithm on leveraging deep learning for semantic image segmentation and our automatic object-privacy alignment algorithm for assigning the privacy settings given at the image level into the most relevant object classes; Section 4 introduces our algorithm for visual tree construction; Section 5 presents our algorithm for joint learning of the deep CNNs and the tree classifier over the visual tree; Section 6 reports the experimental results for algorithm and system evaluation; Section 7 concludes the paper and outlines the future work.

II. Related Work

In this section, we briefly review the most relevant research on: (1) privacy protection for social image sharing [5]–[18], [49], [50]; and (2) deep learning [23]–[30], [51] and multi-task learning for object detection [21], [22], [37], [38].

A. Privacy Protection for Social Image Sharing

Several recent works have studied how to automate the privacy setting process for image sharing [5]–[18]. Bonneau et al. [14] proposed the concept of privacy suites which recommend users a suite of privacy settings that "expert" users or other trusted friends have already set, so that normal users can either directly choose a setting or only need to do minor modification. Ravichandran et al. [13] studied how to predict a user's privacy preferences for location-based data (i.e., share her location or not) based on location and time of day. Fang et al. [16] proposed a privacy wizard to help users grant privileges to their friends. The wizard asks users to first assign privacy labels to the selected friends, and then uses this as the input to construct a classifier to classify friends based on their profiles and automatically assign privacy labels to the unlabeled friends. More recently, Klemperer et al. [12] studied whether the keywords and captions (which are provided by the users when they tag their photos) can be used to help users create and maintain access-control policies more intuitively, where the social tags created for organizational purposes can be re-purposed to help create reasonable access-control rules.

The aforementioned approaches focus on deriving policy settings for only traits, so they mainly consider social context such as one's friend list. While interesting, they may not be sufficient to address challenges brought by images for which privacy may vary substantially not just because of social contexts but also due to the actual image content as considered in our work. Zerr's work [17] and Squicciarini et al. [18] have explored privacy-aware image classification by using a mixed set of features, both content and meta-data. More recently, Tonge and Caragea [49] integrated the deep features for image privacy prediction and

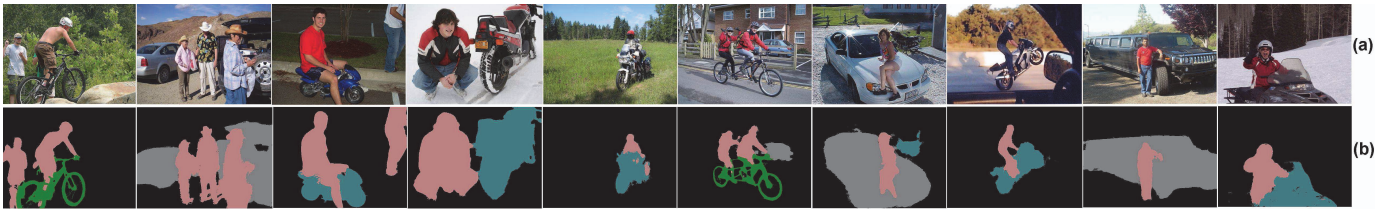

Fig. 3. Our results on semantic image segmentation: (a) original social images; and (b) semantic segmentation of object regions for human beings and cars, bikes, motorcycles.

Fig. 4. Our results on semantic image segmentation: (a) original social images; and (b) semantic segmentation of object regions for human beings and animals (horses, cats and dogs).

Spyromitros-Xioufis et al. [50] leveraged user-dependent images and privacy settings to support personalized privacy-aware image classification. Both teams have found that the deep features can yield remarkable improvements on the performance as compared with other handcrafted visual features such as SIFT, GIST and color histograms. Compared to them, our approach provides a finer level of image classification and is much more efficient.

B. Deep Multi-Task Learning for Object Detection

Our proposed iPrivacy is built upon deep learning techniques, and hence we briefly discuss the new contributions in our work compared to existing works on deep learning. Deep learning [23]–[25] has demonstrated its outstanding abilities on learning high-level features and significantly boosting the accuracy rates for large-scale object detection (i.e., detecting and recognizing large numbers of object classes), but they still have room to improve. For example, softmax is used to flatly map the high-level features into large numbers of object classes, where the inter-task correlations (inter-class visual similarities) are completely ignored. As a result, the process for learning the deep CNNs may be pushed away from the global optimum because the gradients of the objective function are not uniform for all the object classes and such learning process may distract on discerning the object classes that are hard to be discriminated [26]–[30]. In our work, a tree structure is seamlessly integrated with deep network to identify the inter-related learning tasks (the the gradients of the objective function for such inter-related learning tasks are more uniform) and avoid such distraction effectively.

Multi-task learning [21], [22] has demonstrated its ability on learning more discriminative classifiers by considering multiple inter-related learning tasks jointly. However, they cannot be directly applied to our case due to the following two reasons. First, traditional multi-task learning algorithms usually assume that all the tasks are equally related. However, such assumption does not hold in image sharing because it is unnecessary for each privacy-sensitive object class to be related with all the others. Second, although there have been some recent efforts in improving the efficiency [26]–[30], [37], [51], multi-task learning algorithms still yield high computational cost which would not be able to provide real-time recommendations for users who would like to share images instantly.

III. Semantic Image Segmentation and Automatic Object-Privacy Alignment

In social websites, privacy settings are given to entire images. Such image-level privacy cannot tell which object in the image is indeed the sensitive one that needs protection. Therefore, the first phase in our iPrivacy system is to obtain a finer-level privacy setting, i.e., object-level privacy settings. This phase consists of the following major steps: (1) obtain semantic segmentations of object regions; (2) identify a set of privacy-sensitive object classes from large numbers of social images; (3) convert image-level privacy setting to object-level privacy setting by assigning the image-level privacy settings to the most relevant object regions (object classes) in the images, and we can take lessons from some pioneering researches on automatic object-tag alignment [39]–[43]; and (4) learn object-privacy relatedness, i.e., the correspondences between the object classes and the privacy settings. In what follows, we elaborate the details of each step. Without loss of generality, we consider three groups of privacy settings in this work: (a) public; (b) private; and (c) shared with friends or family.

First, we segment each social image into a set of semantic object regions by integrating deep CNNs [45]–[48] with CRF (conditional random fields) models [44]. Specifically, we train a fully CNNs in an end-to-end way to enable pixel-level prediction and classification. Then, a CRF model is further learned to integrate the neighboring pixels for the same object classes to generate semantic object regions. As shown in Fig. 3 and Fig. 4, our integrated approach can identify semantic object regions and their categories (object classes) effectively.

After segmentation, we proceed to introduce our object-privacy alignment algorithm that precisely assigns image-level privacy settings to the object classes in the images. Note that the semantics of each social image can be described effectively by all its object classes. For a given social image, by projecting

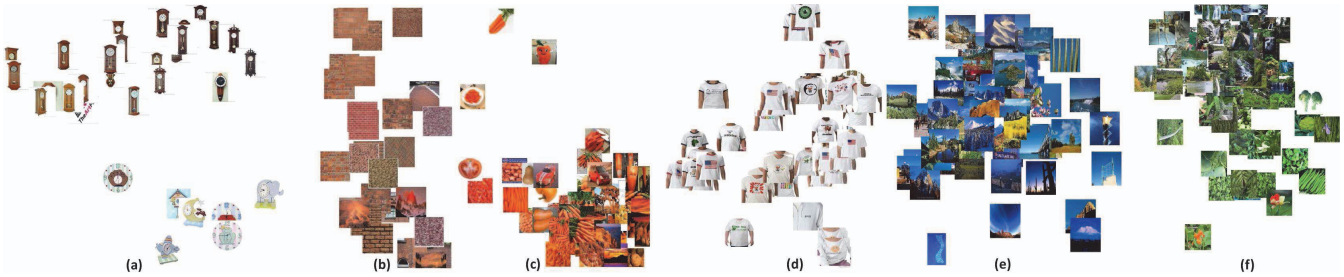

Fig. 5. Image clustering results, where the semantically-similar social images are illustrated according to their semantic similarities.

all its object classes over the full set of 1000 object classes, we can obtain a 1000-dimensional sparse representation for the given social image, e.g., bag of object classes. Based on this, we define the semantic similarity between two images as follows:

$$\kappa_I(X_i, X_j) = \sum_{l=1}^{1000} \delta(X_i^l, X_j^l) \tag{1}$$

where X_i and X_j are the bags of object classes in these two social images, $\delta(X_i^l, X_j^l)$ is defined as:

$$\delta(X_i^l, X_j^l) = \begin{cases} 1, & if\ X_i^l = X_j^l = 1; \\ 0, & otherwise \end{cases} \tag{2}$$

where $X_i^l = X_j^l = 1$ indicates that two images X_i and X_j contain the same object class, i.e., the lth object class. Then, we cluster all the social images based on their semantic similarities $\kappa_I(\cdot, \cdot)$. Some experimental results on image clustering are shown in Fig. 5, from which we can see that images in the same cluster contains similar object classes.

Next, we merge the privacy settings of the images in the same cluster to generate a short list of common privacy settings. Such common privacy settings are re-ranked according to their occurrence frequencies. Then, those with higher occurrence frequencies are selected to represent the privacy settings of the whole cluster. Specifically, given a cluster, let P denote the integrated set of privacy settings for all the images in this cluster, and let t be one particular privacy setting in P, we define the relevance score between the privacy setting t and an object class C_i as follows:

$$\gamma(C_i, t) = \frac{\parallel \Psi(C_i, t) \parallel}{\parallel \Psi(C, P) \parallel} \tag{3}$$

where $\Psi(C, P)$ denotes the full set of images in the cluster which contains a set of object classes $C = \{C_1, \cdots, C_i, \cdots, C_m\}$ and has the integrated set of privacy settings P, and $\Psi(C_i, t)$ denotes a subset of images in the cluster that contains object class C_i and has the privacy setting t. In summary, $\Psi(C_i, t) \subseteq \Psi(C, P)$, $C_i \in C$ and $t \in P$. $\parallel \Psi(C_i, t) \parallel$ is the number of images in $\Psi(C_i, t)$ and $\parallel \Psi(C, P) \parallel$ is the number of images in $\Psi(C, P)$.

It may seem more straightforward to calculate the object-privacy relevance score $\gamma(C_i, t)$ by only using the co-occurrences $\parallel \Psi(C_i, t) \parallel$ between the given privacy setting t and the object class C_i because their co-occurrences $\parallel \Psi(C_i, t) \parallel$ can indicate the probability for the privacy setting t to be assigned to the object class C_i. However, we normalize the co-occurrences $\parallel \Psi(C_i, t) \parallel$ because of two reasons: (a) each cluster may contain a large number of semantically-similar social images and hence a large number of object classes; (b) our focus is to find the common privacy settings for all the object classes.

Considering the large amount of social images, it is very likely to have some object classes co-occurring frequently in the same images. The frequently co-occurring object classes are strongly related, and hence, are more likely to share similar privacy settings. For examples, object classes, such as computer screens, offices and notebooks, may co-occur frequently in the same images with similar privacy settings. Therefore, we model such object co-occurrences as an object co-occurrence network in order to provide a good environment to refine the object-privacy relevance scores. Our object co-occurrence network consists of two key components: (a) object tags for interpreting the object classes; and (b) their co-occurrences in social images. In particular, given two object classes C_i and C_j, their co-occurrence $\phi(C_i, C_j)$ is defined as:

$$\phi(C_i, C_j) = \rho(C_i, C_j) \log \frac{\rho(C_i, C_j)}{\rho(C_i) + \rho(C_j)} \tag{4}$$

where $\rho(C_i, C_j)$ is the co-occurrence probability of two object classes C_i and C_j , $\rho(C_i)$ and $\rho(C_j)$ are their individual occurrence probabilities.

$$\rho(C_i, C_j) = \frac{N(C_i, C_j)}{N}, \quad \rho(C_i) = \frac{N(C_i)}{N}, \quad \rho(C_j) = \frac{N(C_j)}{N} \tag{5}$$

where $N(C_i, C_j)$ is the number of social images which contain the two object classes C_i and C_j simultaneously, $N(C_i)$ is the number of social images which contain the object class C_i, $N(C_j)$ is the number of social images which contain the object class C_j, and N is the total number of social images. The object classes, which have large values of co-occurrences $\phi(\cdot, \cdot)$, are connected to form an object co-occurrence network as shown in Fig. 6.

Over the obtained object co-occurrence network, we perform a random walk to refine the object-privacy relevance scores iteratively. Given an image cluster which contains m object classes, we use $\rho_k(C_i, t)$ to denote the object-privacy relevance score between the object class C_i and the privacy setting t at the kth iteration. For example, $\rho_0(C_i, t) = \gamma(C_i, t)$ is the initial object-privacy relevance score as defined in Eq.(3).

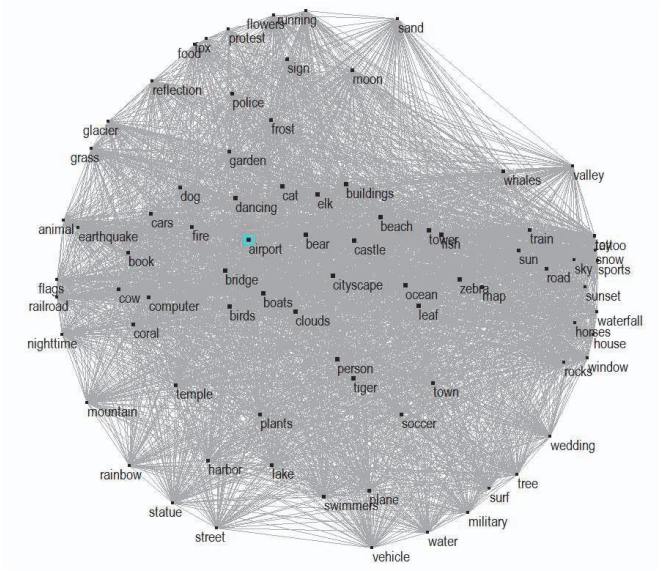

Fig. 6. A small part of our object co-occurrence network.

We further define Φ as an $m \times m$ transition matrix, whereby its element ψ_{ij} is used to define the transition probability from the object class C_i to its inter-related object class C_j on our object co-occurrence network. Such inter-object transition means that the frequently co-occurring object classes may share similar privacy settings. ψ_{ij} is defined as:

$$\psi_{ij} = \frac{\phi(C_i, C_j)}{\sum_{C_k \in \Omega_{C_i}} \phi(C_i, C_k)} \tag{6}$$

where Ω_{C_i} is the first-order neighbors of the object class C_i on our object co-occurrence network, and $\phi(C_i, C_j)$ is the inter-object correlation between C_i and C_j as defined in Eq. (4). Correspondingly, the random walk process is formulated as follows:

$$\rho_k(C_i, t) = \theta \sum_{C_j \in \Omega_{C_i}} \rho_{k-1}(C_i, t)\psi_{ij} + (1-\theta)\gamma(C_i, t) \tag{7}$$

where $\rho_{k-1}(C_i, t)$ is the object-privacy relevance score at the $(k-1)$th iteration and $\theta = 0.4$ is a weight parameter (e.g., we assume that the original relevance score is more important than the transition relevance score). For a given object class C_i, all the relevant privacy settings are re-ranked according to their object-privacy relevance scores with C_i and the privacy setting with the largest object-privacy relevance score is finally selected for the given object class C_i.

By assigning the privacy settings (given at the image level) into the most relevant object classes precisely, our object-privacy alignment algorithm can effectively: (a) identify a set of privacy-sensitive object classes (which their privacy settings are identified as private); and (b) determine the object-privacy correspondences precisely. Finally, recommending the privacy settings for the images being shared can be achieved by detecting their underlying privacy-sensitive objects and using such object-privacy correspondences to recommend the best-matching privacy settings for image sharing.

IV. Visual Tree Construction

The second phase in our iPrivacy system is to provide an effective organization of a set of privacy-sensitive object classes obtained from the previous step. For this, a visual tree $T = (\mathbb{V}, \mathbb{E})$ is learned which comprises a set of nodes $\mathbb{V}$ and a set of edges $\mathbb{E}$. Each non-leaf node $c \in \mathbb{V}$ is associated with a set of privacy-sensitive object classes $\mathbb{L}(c) \subseteq \{1, \ldots, M\}$ which are a subset of its parent node. Each leaf node is associated with one particular privacy-sensitive object class.

Given two privacy-sensitive object classes C_j and C_i, their inter-class visual similarity $S(C_i, C_j)$ is defined as:

$$S(C_i, C_j) = \frac{1}{R^2} \sum_{x_l \in C_i} \sum_{x_h \in C_j} \kappa_o(x_l, x_h) \tag{8}$$

where R is the number of training images, $\kappa_o(\cdot, \cdot)$ is the Gaussian kernel function for similarity characterization, x_l and x_h are the deep features for the lth social image from the object class C_i and the hth social image from the object class C_j, and R is the total number of social images from each object class. Given N privacy-sensitive object classes, their inter-class similarity matrix $\mathbf{S}$ is obtained automatically and its component is defined as $S_{ij} = S(C_i, C_j)$.

Such inter-class visual similarities are then used to determine their inter-class separability. The visually-similar privacy-sensitive object classes are difficult to be separated by the node classifiers. They should be assigned to the same coarse-grained group (superclass) of privacy-sensitive object classes to avoid early incorrect partitioning at a high-level node of the visual tree. In other words, partitioning mistakes at the high-level nodes are more critical because of inter-level error propagation. Therefore, we develop the following approach.

A top-down approach is applied to partition large numbers of privacy-sensitive object classes hierarchically, which starts from the root node (that contains all the privacy-sensitive object classes) and ends at the leaf nodes (each leaf node contains one particular privacy-sensitive object class). For each non-leaf node c, its M privacy-sensitive object classes are partitioned into B smaller groups (i.e., B children nodes at next level) by minimizing inter-group visual similarity and maximizing intra-group visual similarity:

$$min \left\{ \psi(c, B) = \sum_{l=1}^{B} \frac{\sum_{C_i \in G_l} \sum_{C_j \in G^c/G_l} S(C_i, C_j)}{\sum_{C_i \in G_l} \sum_{C_h \in G_l} S(C_i, C_h)} \right\} \tag{9}$$

where $S(C_i, C_j)$ is the inter-class visual similarity between two privacy-sensitive object classes C_i and C_j, $G^c = \{G_l | l = 1, \cdots, B\}$ is used to represent B groups (clusters) of M privacy-sensitive object classes for the current non-leaf node c, and G^c/G_l is used to represent other $B-1$ groups in G^c except G_l. Note that if the set $\mathbb{L}(c)$ for a non-leaf node c has less than B privacy-sensitive object classes, only $|\mathbb{L}(c)|$ child nodes are generated. This node partitioning process is performed repeatedly until a complete tree is created whereby each leaf node contains one single privacy-sensitive object class. At the end of partitioning, the privacy-sensitive object classes, which share significant common visual properties but may contain subtle visual differences, are finally assigned to the sibling leaf nodes under the same parent node.

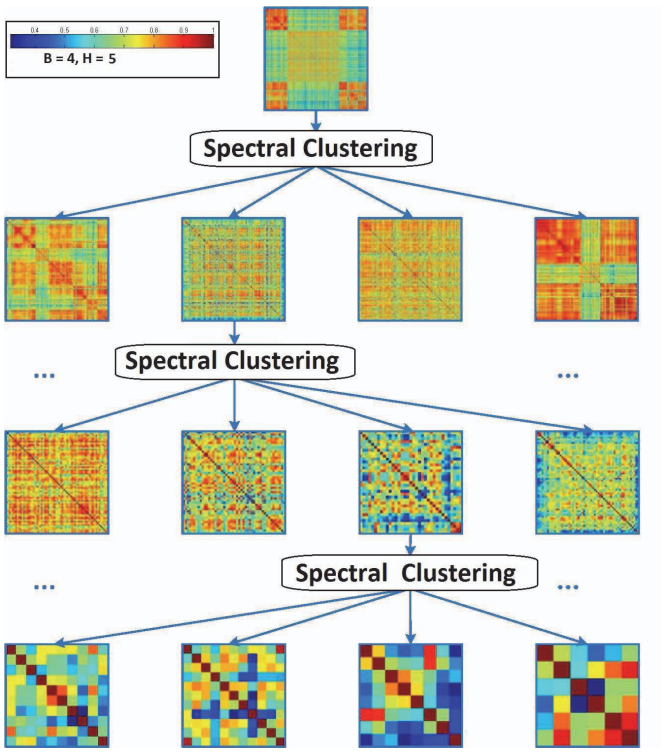

Fig. 7. Hierarchical partitioning of the inter-class similarity matrix $\mathbf{S}$ for visual tree construction: $B = 4$ and $H = 5$, where B is the branching factor to indicate the number of sibling child nodes under the same parent node and H is the maximum depth of visual tree (from root node to leaf node).

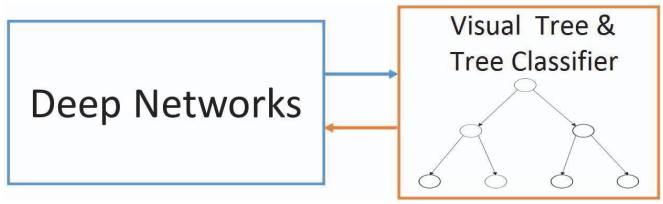

Fig. 8. The illustration of the key components of our deep multi-task learning algorithm for joint learning of deep CNNs and tree classifier.

Fig. 7 illustrates an example of this hierarchical clustering process when the branching factor is $B = 4$ and the maximum depth is $H = \log_B N_o$, where N_o is the totally number of privacy-sensitive object classes.

Our visual tree can organize large numbers of privacy-sensitive object classes hierarchically in a coarse-to-fine fashion, which allows our hierarchical deep multi-task learning algorithm (presented in the next section) to learn discriminative tree classifier effectively. Moreover, the use of the tree classifier can help reduce the computational cost significantly by ruling out unlikely groups of privacy-sensitive object classes (i.e., irrelevant high-level nodes) at an early stage

V. Joint Learning of Deep CNNs and Tree Classifier

The third phase in our iPrivacy system is to learn the tree classifier and the deep CNNs jointly over the visual tree in an end-to-end fashion, so that we can achieve fast and accurate detection of large numbers of privacy-sensitive object classes. As illustrated in Fig. 8, our deep network contains two parts: (a) deep CNNs for image representation; and (b) tree classifier for hierarchical object detection. The significant difference between our deep networks and traditional deep CNNs is that the tree classifier is used to replace the N-way flat softmax classifier [24], [32]. A *bottom-up* deep multi-task learning algorithm is further developed to achieve joint learning of the deep CNNs and the tree classifier.

A. Joint Learning for Sibling Leaf Nodes

For a given parent node C_h at the second level of the visual tree, the multi-task node classifiers for its visually-similar privacy-sensitive object classes (its sibling leaf nodes at the first level of the visual tree) are trained simultaneously by optimizing a joint objective function:

$$min\left\{\nu\sum_{l=1}^{R}\sum_{j=1}^{B}\xi_j^l+\delta_1 Tr\left(WW^T\right)+\frac{\delta_2}{2}Tr\left(WLW^T\right)\right\} \tag{10}$$

subject to:

$$\forall_{l=1}^{R}\forall_{j=1}^{B}:\quad y_j^l\left(W_j^T\cdot x_j^l+b\right)\geq 1-\xi_j^l,\ \ \xi_j^l\geq 0 \tag{11}$$

where R is the number of training images for each privacy-sensitive object class, W_j is the classifier parameter for the jth privacy-sensitive object class C_j, $Tr(\cdot)$ is used to represent the trace of matrix, ξ_j^l indicates the training error rate, δ_1 and δ_2 are the regularization parameters, ν is the penalty term, $W = (W_1, \cdots, W_j, \cdots, W_B)$, L is the Laplacian matrix of the relevant inter-class similarity matrix $\mathbf{S}$. The inter-class visual similarities $S(\cdot,\cdot)$ as defined in Eq. (8) are used to approximate the inter-task relationships, the manifold regularization term $Tr\left(WLW^T\right)$ is used to enforce that: if two visually-similar privacy-sensitive object classes C_i and C_j have larger inter-class visual similarity $S(\cdot,\cdot)$, their multi-task node classifiers will have stronger correlations.

The joint objective function as defined in Eqs.(10-11) is optimized by using the stochastic alternating direction method of multipliers (ADMM) algorithm [33], [34], which is able to handle non-smooth regularization term. The corresponding gradients for the joint objective function as defined in Eqs.(10-11) are calculated and they are back-propagated [31] through the deep CNNs to fine-tune the weights.

The dual problem for joint classifier training is defined as:

$$min\left\{\sum_{j=1}^{B}\sum_{l=1}^{R}\beta_l^j-\frac{1}{2\delta_1}\beta^T Y\Re\left(\Re+\frac{\delta_2}{\delta_1}\Re\left(L\bigotimes I\right)\Re\right)^{-1}\Re Y\beta\right\} \tag{12}$$

$$\textit{subject to:}\quad \forall_{l=1}^{R}\forall_{j=1}^{B}:\quad \sum_{l=1}^{R}\beta_l^j\cdot y_l^j=0,\ \ 0\leq\beta_l^j\leq 1 \tag{13}$$

where $Y = [y_j^l \mid l \in \{1, R\}, j \in \{1, B\}]^T \in \{0,1\}^{R\times B}$ and $y_j^l \in \{0,1\}^{B\times 1}$ is the class indicator vector for the training image x_j^l, I is the identical matrix, $L \bigotimes I$ is the Kronecker product between two matrix L and I, $\beta = (\beta_1, \cdots, \beta_j, \cdots, \beta_B)$ is the set of the dual variables, $\beta_j = (\beta_1^j, \cdots, \beta_l^j, \cdots, \beta_R^j)$, $\Re$ is a block diagonal similarity matrix and it is

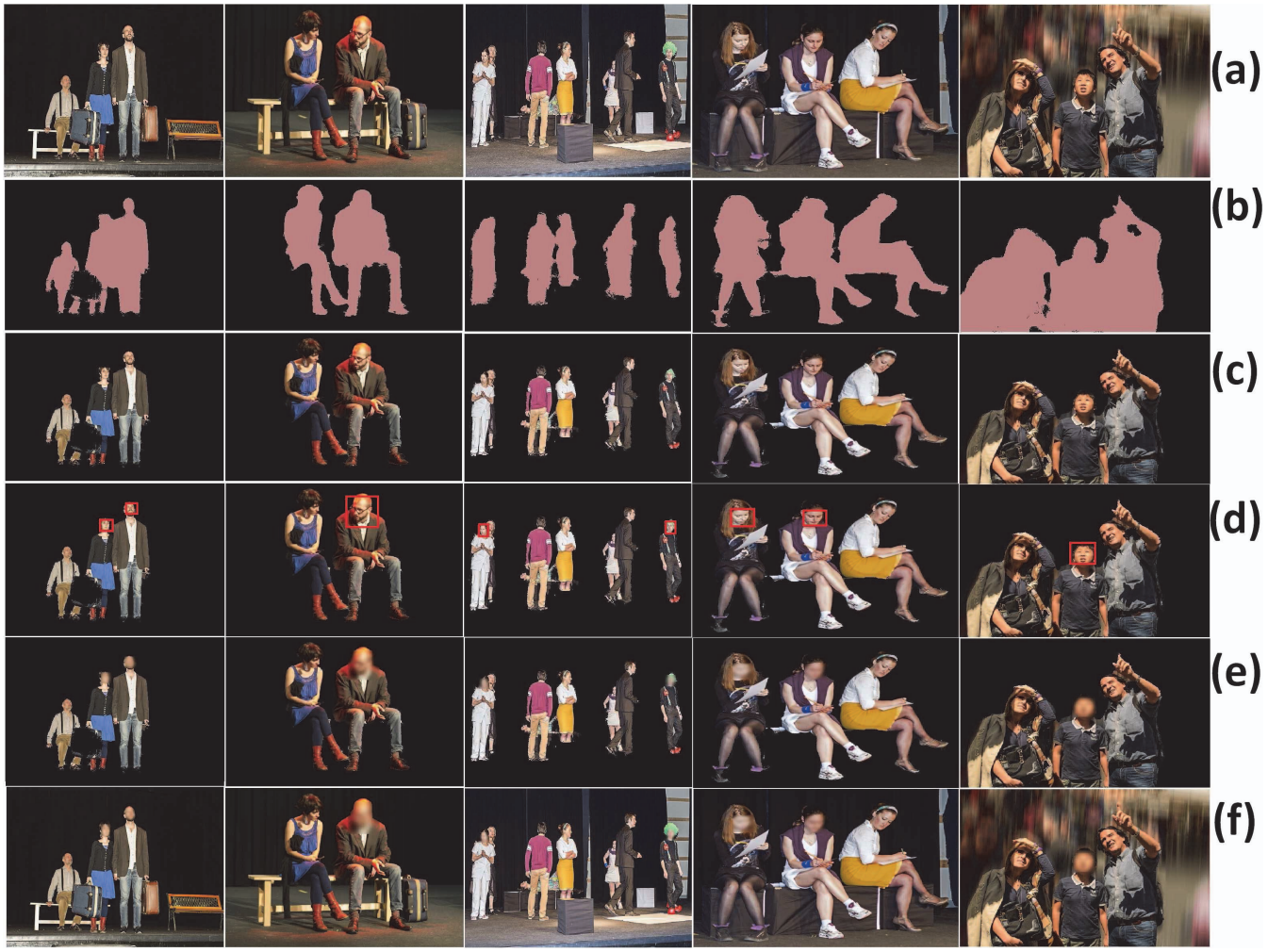

Fig. 9. Our experimental results on integrating the deep CNNs and the tree classifier to detect and recognize the most significant privacy-sensitive object class (i.e., human beings) and protecting image privacy via simply blurring privacy-sensitive objects: (a) original image; (b) human regions detected by deep learning; (c) detected human objects; (d) face identification for privacy-sensitive objects; (e) face blurring for privacy-sensitive objects; (f) shared images with privacy protection via simple face blurring.

defined as:

$$\Re = \begin{bmatrix} \Re_1 & & & \\ & \ddots & & \\ & & \Re_j & \\ & & & \ddots \\ & & & & \Re_B \end{bmatrix} \tag{14}$$

where $\Re_j \in \mathbb{R}^{R\times R}$ is the similarity matrix for R training images for the jth privacy-sensitive object class.

After the optimal β^* is obtained by solving Eqs. (12-13), we can compute the optimal $\alpha^* = (\alpha_1^*, \cdots, \alpha_j^*, \cdots, \alpha_B^*)$ as:

$$\alpha^* = \frac{1}{2\delta_1}\left(\Re + \frac{\delta_2}{\delta_1}\left(\Re\left(L \bigotimes I\right)\Re\right)^{-1}\Re Y\beta^*\right) \tag{15}$$

where $\alpha_j^* = (\alpha_j^{1*}, \cdots, \alpha_j^{l*}, \cdots, \alpha_j^{R*})$. Finally, the multi-task node classifiers for the sibling privacy-sensitive object classes under the same parent node (superclass) C_h are defined as:

$$\forall_{j=1}^{B}: \; f_{C_j}^1(x) = \sum_{l=1}^{R} \alpha_j^{l*}\kappa(x_j^l, x) + b_j^*, \quad C_j \in C_h \tag{16}$$

B. Joint Learning for Sibling Non-Leaf Nodes

For a given high-level non-leaf node C_k, training the multi-task node classifiers for its sibling child nodes is achieved by:

$$min\left\{\nu\sum_{m=1}^{R}\sum_{h=1}^{B}\xi_h^m + \gamma_1 Tr\left(WW^T\right) + \frac{\gamma_2}{2}Tr\left(WLW^T\right)\right\} \tag{17}$$

subject to:

$$\forall_{m=1}^{R}\forall_{h=1}^{B}: \; y_h^m\left(W_h^T \cdot x_h^m + b\right) \geq 1 - \xi_h^m, \quad \xi_h^m \geq 0, \quad C_h \in C_k \tag{18}$$

$$\forall_{h=1}^{B}: \; f_{C_h}^{l+1}(x) - f_{C_j}^{l}(x) \geq 0 \tag{19}$$

$$\forall_{h=1}^{B}: \; f_{C_h}^{l+1}(x) = \sum_{j=1}^{B}\eta_j f_{C_j}^{l}(x) \tag{20}$$

where $W = (W_1, \cdots, W_h, \cdots, W_B)$, the inter-level discrimination constraints as defined in Eqs. (19-20) are used to control inter-level error propagation [52]. The bundle method [35], [36] is used to solve the optimization problem as defined in Eqs. (17-20).

Fig. 10. Our experimental results on integrating the deep CNNs and the tree classifier to detect and recognize the most significant privacy-sensitive object class (i.e., human beings) and protecting image privacy via simply blurring privacy-sensitive objects: (a) original image; (b) human regions detected by deep learning; (c) detected human objects; (d) face identification for privacy-sensitive objects; (e) face blurring for privacy-sensitive objects; (f) shared images with privacy protection via simple face blurring.

By leveraging the visual tree to generate subtrees (each subtree contains one parent node and at most B sibling child nodes) iteratively and determine the inter-related learning tasks automatically, our deep multi-task learning algorithm can provide *an iterative solution* for large-scale machine learning, so that training large numbers of node classifiers over the visual tree (i.e., tree classifier) becomes computationally tractable.

The fourth phase in our iPrivacy system is to automatically detect the privacy-sensitive objects from the images being shared, recognize their classes, and identify their privacy settings for image sharing. After the tree classifier and the deep CNNs are available, they are used to predict the label (object class) for a given image being shared or an object proposal x in the image being shared, i.e., identifying its best-matching privacy-sensitive object class. After the privacy-sensitive objects are identified from the image being shared, a simple solution for image privacy protection is provided by blurring such privacy-sensitive objects automatically.

VI. Experimental Results for Algorithm and System Evaluation

All our experiments are conducted on a parallel set that comprises about 800,000 social images and their privacy settings. To assess the effectiveness of our proposed algorithms, we have evaluated multiple aspects: (a) whether our deep multi-task learning algorithm can obtain better results on detecting and recognizing large numbers of privacy-sensitive object classes; (b) whether our deep multi-task learning algorithm can control the inter-level error propagation more effectively as compared with other baseline methods and achieve higher accuracy rates on large-scale object detection; (c) whether blurring privacy-sensitive objects may significantly affect the utility of the shared images; and (d) whether detecting the privacy-sensitive object classes and learning their object-privacy correspondences can allow us to recommend better privacy settings for image sharing and image privacy protection.

As shown in Fig. 9 and Fig. 10, one can observe that our algorithm can effectively detect and recognize the most important privacy-sensitive object class (i.e., human beings) from the images being shared. Detecting and recognizing such privacy-sensitive objects from images is the first and most important step for automating image privacy protection. When such privacy-sensitive object classes are detected and recognized from the images being shared, we can incorporate an simple approach to blur such privacy-sensitive objects or backgrounds to protect the image privacy as illustrated in Fig. 9 (f) and Fig. 10 (f). The average time cost is around

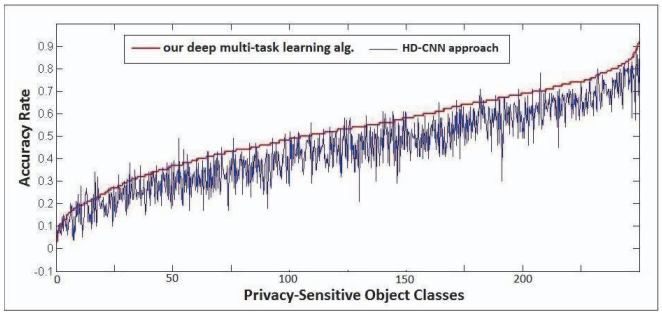

Fig. 11. The comparison on the accuracy rates on large-scale object detection and recognition when different approaches are used for tree classifier training: (a) our deep multi-task learning algorithm; (b) HD-CNN approach [29].

TABLE I
THE SHORT LIST OF PRIVACY-SENSITIVE OBJECT CLASSES IDENTIFIED BY THIS WORK

Categories	Privacy-Sensitive Object Classes
Human Beings	portrait, people in birthday party, human body, human hair, human face, human eye, human neck, people in award, mannequin modeling, customer, ···
Family	baby, children, relatives/family, friend, husband, wife, parents, brother, sister, cousin, kids at play, african american, couple, ···
Woman	girl, explicit women, female surfing, ···
Ethic	erotic, ···
House	home, bedroom, restroom, indoor, kitchen, ···
Clothes	suit, bikini, maillot, ···
Activity	drinking, wedding, swimming, bathing, working boys, sitting on boys, fishing, birthday parties, travel, vacations, fun summer vacation, ···
Work Lab	science lab, laptop, computer, personal , ···

TABLE II
THE SHORT LIST OF PUBLIC OBJECT CLASSES IDENTIFIED BY THIS WORK

Categories	Public Object Classes
Nature & Scenery	mountain, island, rock, sand, sea, coast, lake, river, sunset, sky, landscape, lakeside, sandbar, beach, cartoon, fire, ice, water, fashion, ···
Animal	pets, dog, cat, bird, wild animals, fish, ···
Plant	flower, tree, asian floral, ···
Season	winter, sprint, summer, autumn, ···
Transportation	road, traffic, boat, car, ···
Building	House outside, garden, bridge,shopping center,park, bank, ···
Planet	moon, sun, earth, ···
City Signs	New York, Washington, Beijing, ···

1.2 seconds for detecting and recognizing the privacy-sensitive objects from an image being shared. A short list of the privacy-sensitive object classes are illustrated in Table I, and a short list of public object classes are also illustrated in Table II.

To evaluate the effectiveness of our deep multi-task learning algorithm on controlling the inter-level error propagation, we have compared our algorithm with three baseline methods: (a) Traditional hierarchical multi-task learning approach [37]; (b) Label tree [38]; and (c) HD-CNN approach [29]. As shown in Fig. 11 and Fig. 12, the accuracy rates for our algorithm are higher than that for other three baseline methods.

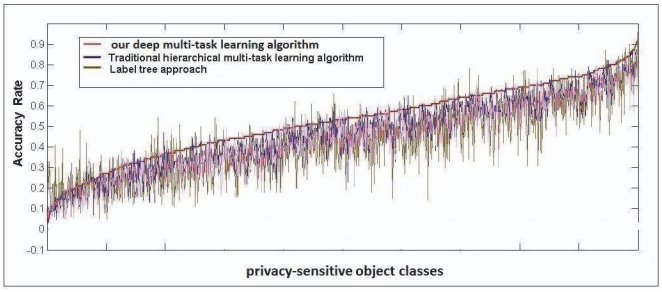

Fig. 12. The comparison on the accuracy rates on large-scale object detection and recognition when different approaches are used for classifier training: (a) our deep multi-task learning algorithm; (b) traditional hierarchical multi-task learning approach [37]; and (c) label tree [38].

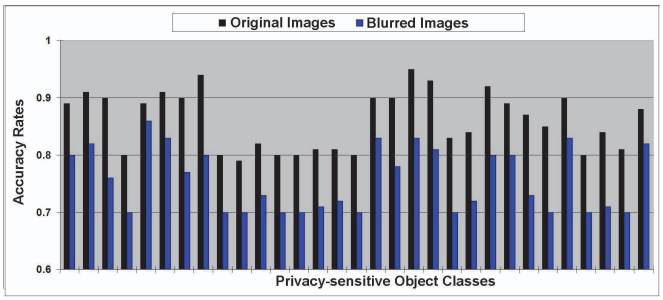

Fig. 13. The utility of blurred images could be low, i.e., sharing the blurred images for learning deep CNNs and tree classifier may dramatically decrease the accuracy rates for object detection.

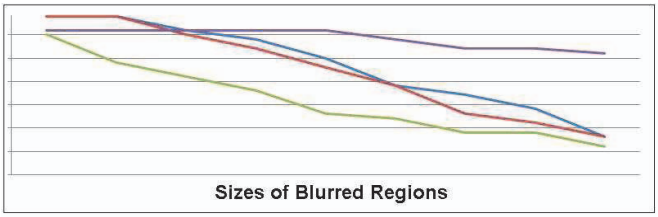

Fig. 14. The utility of blurred images (i.e., accuracy of tree classifier) may change quickly when the sizes of blurred regions are increased.

By explicitly leveraging the inter-class visual correlations for multi-task learning, our deep multi-task learning algorithm can achieve better performance on distinguishing the visually-similar privacy-sensitive object classes, which are usually hard to be distinguished.

Blurring the privacy-sensitive objects in the images being shared may allow us to protect the image privacy effectively, but it may seriously affect the utility of the shared images for classifier training. It is worth noting that the definition of image utility could be significantly different for various tasks. For the task of joint learning of the deep CNNs and the tree classifier, we have compared two approaches: (a) using original images; and (b) using blurred images. As shown in Fig. 13, one can observe that using the blurred images may seriously mislead the joint process for learning the deep CNNs and the tree classifier, which may dramatically decrease the accuracy rates for object detection (i.e., the utility of the shared (blurred) images). For the same object class, as shown in Fig. 14, increasing the sizes of the blurred regions may significantly decrease the detection accuracy rate because large

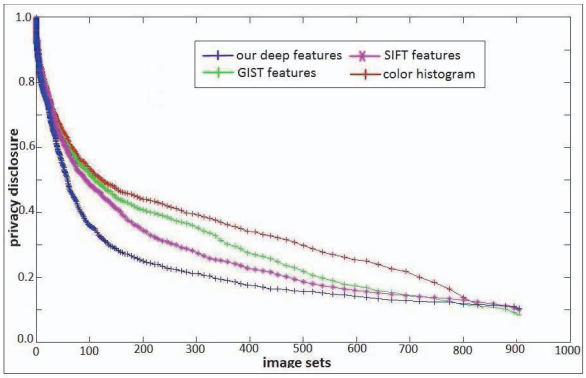

Fig. 15. The comparison on the effectiveness of our algorithm on privacy setting recommendation when different types of features are extracted for image representation and classifier training.

sizes of blurred regions may result in unrepresentative deep CNNs and generate irrelevant features for training the tree classifier with lower discrimination power.

Some pioneering researches have leveraged various features (both handcrafted and deep features) to train the classifiers for supporting privacy setting recommendation [17], [18], [49], [50]. Thus it is also very interesting to evaluate whether using different types of features may bring significant improvement on privacy setting recommendation, e.g., recommending better privacy settings can achieve better protection of image privacy or result in less privacy disclosure. For 90, 000 test images, we partition them into 900 smaller sets according to their underlying object-privacy relatedness and perform our algorithm evaluation over these 900 image sets (each set has 100 images with similar object-privacy relatedness) independently. For a given image set with T test images, its privacy disclosure is defined as:

$$PD = \frac{1}{T}\sum_{l=1}^{T}\delta(PS(I_l), OS(I_l)) \tag{21}$$

where T is the total number of test images (T = 100 in this experiment), $PS(I_l)$ indicates the predicted privacy setting for the lth given image I_l, $OS(I_l)$ is its original privacy setting given by users. $\delta(PS(I_l), OS(I_l))$ is defined as:

$$\delta(PS(I_l), OS(I_l)) = \begin{cases} 1, & if\ \ PS(I_l) \neq OS(I_l) \\ 0, & otherwise \end{cases} \tag{22}$$

The $\delta(\cdot,\cdot)$ function is used to emphasize that the privacy disclosure is counted only if the privacy setting for the given image is predicted incorrectly. We sort the image sets according to their values of privacy disclosures and illustrate them in orders.

As shown in Fig. 15, we have compared the performance of our algorithm for privacy setting recommendation when different types of features are used for image representation and tree classifier training. From these comparison experiments, we can observe multiple interesting facts: (1) For most image sets, the deep features can achieve the best performance as compared with other handcrafted features such as SIFT, GIST and color histograms; (2) For some difficult image sets (with huge uncertainty on the object-privacy correspondences), all these features (including deep features) may not be able to achieve acceptable performance, e.g., all of them have large privacy disclosures; (3) For some easy image sets (with very good object-privacy correspondences), all these features (including handcrafted features) can achieve good performance (with small privacy disclosures).

VII. Conclusions

In this paper, a novel approach called *iPrivacy* is developed to automate the process of privacy settings during image sharing. Specifically, by learning the object-privacy relatedness from massive social images, our object-privacy alignment algorithm can allow us to identify a large set of privacy-sensitive object classes and their privacy settings automatically. By learning the deep CNNs and the tree classifier jointly over the visual tree in an end-to-end way, our deep multi-task learning algorithm can achieve fast and accurate detection of privacy-sensitive object classes and recommend the best-matching privacy settings for newly uploaded images. A simple solution for image privacy protection is further provided by automatically blurring the privacy-sensitive objects in images. Our experimental results have demonstrated that our proposed algorithm has achieved very competitive results in terms of both the prediction accuracy and the computational efficiency.

References

[1] C. Wang, B. Zhang, K. Ren, and J. M. Roveda, "Privacy-assured outsourcing of image reconstruction service in cloud," *IEEE Trans. Emerg. Topics Comput.*, vol. 1, no. 1, pp. 166–177, Jun. 2013.

[2] O. Nov, M. Naaman, and C. Ye, "Motivational, structural and tenure factors that impact online community photo sharing," in *Proc. ICWSM*, 2009, pp. 138–145.

[3] A. C. Squicciarini, H. Xu, and X. Zhang, "CoPE: Enabling collaborative privacy management in online social networks," *J. Assoc. Inf. Sci. Technol.*, vol. 62, no. 3, pp. 521–534, 2011.

[4] N. Wang, H. Xu, and J. Grossklags, "Third-party apps on Facebook: Privacy and the illusion of control," in *Proc. ACM CHIMIT*, 2011, Art. no. 4.

[5] M. De Choudhury, H. Sundaram, Y.-R. Lin, A. John, and D. D. Seligmann, "Connecting content to community in social media via image content, user tags and user communication," in *Proc. IEEE ICME*, Jun./Jul. 2009, pp. 1238–1241.

[6] C. Yeung, C. Man, L. Kagal, N. Gibbins, and N. Shadbolt, "Providing access control to online photo albums based on tags and linked data," in *Proc. AAAI Symp.*, 2009, pp. 1–6.

[7] J. P. Pesce, D. L. Casas, G. Rauber, and V. Almeida, "Privacy attacks in social media using photo tagging networks: A case study with Facebook," in *Proc. ACM PSOSM*, 2012, Art. no. 4.

[8] D. Christin, P. S. López, A. Reinhardt, M. Hollick, and M. Kauer, "Share with strangers: Privacy bubbles as user-centered privacy control for mobile content sharing applications," *Inf. Secur. Tech. Rep.*, vol. 17, no. 3, pp. 105–116, 2013.

[9] M. Mannan and P. C. van Oorschot, "Privacy-enhanced sharing of personal content on the Web," in *Proc. ACM WWW*, 2008, pp. 487–496.

[10] N. Vyas, A. C. Squicciarini, C.-C. Chang, and D. Yao, "Towards automatic privacy management in Web 2.0 with semantic analysis on annotations," in *Proc. IEEE CollaborateCom*, Nov. 2009, pp. 1–10.

[11] A. C. Squicciarini, M. Shehab, and F. Paci, "Collective privacy management in social networks," in *Proc. ACM WWW*, 2009, pp. 521–530.

[12] P. Klemperer *et al.*, "Tag, you can see it!: Using tags for access control in photo sharing," in *Proc. CHI*, 2012, pp. 377–386.

[13] R. Ravichandran, M. Benisch, P. G. Kelley, and N. M. Sadeh, "Capturing social networking privacy preferences: Can default policies help alleviate tradeoffs between expressiveness and user burden?" in *Proc. SOUPS*, 2009, pp. 1–18.

[14] J. Bonneau, J. Anderson, and L. Church, "Privacy suites: Shared privacy for social networks," in *Proc. SOUPS*, 2009, pp. 1–2.
[15] J. Bonneau, J. Anderson, and G. Danezis, "Prying data out of a social network," in *Proc. ASONAM*, 2009, pp. 249–254.
[16] L. Fang and K. LeFevre, "Privacy wizards for social networking sites," in *Proc. ACM WWW*, 2010, pp. 351–360.
[17] S. Zerr, S. Siersdorfer, J. Hare, and E. Demidova, "Privacy-aware image classification and search," in *Proc. ACM SIGIR*, 2012, pp. 35–44.
[18] A. Squicciarini, D. Lin, S. Karumanchi, and N. DeSisto, "Automatic social group organization and privacy management," in *Proc. CollaborateCom*, Oct. 2012, pp. 89–96.
[19] O. Dekel, J. Keshet, and Y. Singer, "Large margin hierarchical classification," in *Proc. ICML*, 2004, p. 27.
[20] J. Wang, X. Shen, and W. Pan, "On large margin hierarchical classification with multiple paths," *J. Amer. Statist. Assoc.*, vol. 104, no. 487, pp. 1213–1223, 2009.
[21] T. Evgeniou, C. A. Micchelli, and M. Pontil, "Learning multiple tasks with kernel methods," *J. Mach. Learn. Res.*, vol. 6, pp. 615–637, Apr. 2005.
[22] H. Fei and J. Huan, "Structured feature selection and task relationship inference for multi-task learning," in *Proc. IEEE ICDM*, Dec. 2011, pp. 171–180.
[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.
[24] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM Multimedia*, 2014, pp. 675–678.
[25] J. Donahue *et al.*, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. ICML*, 2014, pp. 647–655.
[26] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in *Proc. ECCV*, 2014, pp. 94–108.
[27] P. Teterwak and L. Torresani, "Shared roots: Regularizing neural networks through multitask learning," Dept. Comput. Sci., Dartmouth College Hanover, NH, USA, Tech. Rep. TR2014-762, 2014.
[28] S. Li, Z.-Q. Liu, and A. B. Chan, "Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network," *Int. J. Comput. Vis.*, vol. 113, no. 1, pp. 19–36, 2015.
[29] Z. Yang *et al.*, "HD-CNN: Hierarchical deep convolutional neural network for image classification," in *Proc. IEEE CVPR*, Jun. 2015, pp. 2740–2748.
[30] N. Srivastava and R. R. Salakhutdinov, "Discriminative transfer learning with tree-based priors," in *Proc. NIPS*, 2013, pp. 2094–2102.
[31] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[32] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[33] S. Azadi and S. Sra, "Towards an optimal stochastic alternating direction method of multipliers," in *Proc. ICML*, 2014, pp. 1–9.
[34] H. Ouyang, N. He, L. Q. Tran, and A. Gray, "Stochastic alternating direction method of multipliers," in *Proc. ICML*, 2013, pp. 1–9.
[35] M. Zinkevich, A. J. Smola, M. Weimer, and L. Li, "Parallelized stochastic gradient descent," in *Proc. NIPS*, 2010, pp. 2595–2603.
[36] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. 19th Int. Conf. Comput. Statist. (COMPSTAT)*, 2010, pp. 177–186.
[37] S. Gopal, Y. Yang, and A. Niculescu-Mizil, "Regularization framework for large scale hierarchical classification," in *Proc. ECML*, 2012, pp. 1–13.
[38] S. Bengio, J. Weston, and D. Grangier, "Label embedding trees for large multi-class tasks," in *Proc. NIPS*, 2010, pp. 163–171.
[39] M. Ames and M. Naaman, "Why we tag: Motivations for annotation in mobile and online media," in *Proc. ACM CHI*, 2007, pp. 971–980.
[40] D. Liu, X.-S. Hua, L. Yang, M. Wang, and H.-J. Zhang, "Tag ranking," in *Proc. ACM Multimedia*, 2009, pp. 351–360.
[41] K. Q. Weinberger, M. Slaney, and R. Van Zwol, "Resolving tag ambiguity," in *Proc. ACM Multimedia*, 2008, pp. 111–120.
[42] D. Liu, S. Yan, Y. Rui, and H.-J. Zhang, "Unified tag analysis with multi-edge graph," in *Proc. ACM Multimedia*, 2010, pp. 25–34.
[43] Y. Shen and J. Fan, "Leveraging loosely-tagged images and inter-object correlations for tag recommendation," in *Proc. ACM Multimedia*, 2010, pp. 5–14.
[44] L.-C. Chen, G. Papandreou, I. Kokkions, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional neural networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, May 2015, pp. 1–14.
[45] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proc. IEEE CVPR*, Jun. 2015, pp. 447–456.
[46] P. O. Pinheiro and R. Collobert, "From image-level to pixel-level labeling with convolutional networks," in *Proc. IEEE CVPR*, Jun. 2015, pp. 1713–1721.
[47] S. Zheng *et al.*, "Conditional random fields as recurrent neural networks," in *Proc. IEEE ICCV*, Dec. 2015, pp. 1529–1537.
[48] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE CVPR*, Jun. 2015, pp. 3431–3440.
[49] A. Tonge and C. Caragea, "Privacy prediction of images shared on social media sites using deep features," in *Proc. AAAI Symp.*, 2015, pp. 4266–4267.
[50] E. Spyromitros-Xioufis, S. Papadopoulos, A. Popescu, and Y. Kompatsiaris, "Personalized privacy-aware image classification," in *Proc. ACM ICMR*, 2016, pp. 71–78.
[51] P. Kontschieder, M. Fiterau, A. Criminisi, and S. R. Bulò, "Deep neural decision forests," in *Proc. IEEE ICCV*, Dec. 2015, pp. 1467–1475.
[52] J. Fan, N. Zhou, J. Peng, and L. 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.

Jun Yu (M'13) received the B.Eng. and Ph.D. degrees from Zhejiang University, Zhejiang, China. He was an Associate Professor with the School of Information Science and Technology, Xiamen University. From 2009 to 2011, he was with Nanyang Technological University, Singapore. From 2012 to 2013, he was a Visiting Researcher with Microsoft Research Asia. He was a short-term Visiting Scholar with UNC Charlotte. He is currently a Professor with the School of Computer Science and Technology, Hangzhou Dianzi University. He has authored and co-authored over 50 scientific articles. His research interests include multimedia analysis, machine learning, image processing, and image privacy protection. He has (co-)chaired for several special sessions, invited sessions, and workshops. He served as a Program Committee Member or reviewer top conferences and prestigious journals. He is a Professional Member of the ACM and CCF.

Baopeng Zhang received the Ph.D. degree in computer science from Tsinghua University in 2008. He was a Visiting Scholar with UNC Charlotte from 2015 to 2016. He is currently an Assistant Professor with the School of Computer and Information Technology, Beijing Jiaotong University, China. His research interests include semantic image/video classification and retrieval, statistical machine learning, large-scale semantic data management and analysis, and image privacy protection.

Zhengzhong Kuang received the B.S. degree in computer science from the China University of Petroleum, Tsingdao, China, in 2014, where he is currently pursuing the Ph.D. degree. He is also a Visiting Student with UNC Charlotte. His research interests include computer vision, machine learning, and image privacy protection.

Dan Lin received the Ph.D. degree in computer science from the National University, Singapore, in 2007. She was a Post-Doctoral Research Associate with Purdue University for two years. She is currently an Associate Professor and the Director of Cybersecurity Laboratory with the Missouri University of Science and Technology. Her main research interests cover many areas in the fields of database systems and information security.

Jianping Fan received the M.S. 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 Post-Doctoral Researcher with Fudan University, Shanghai, during 1997–1998. From 1998 to 1999, he was a Researcher with the Japan Society of Promotion of Science, Department of Information System Engineering, Osaka University, Osaka, Japan. From 1999 to 2001, he was a Post-Doctoral Researcher with the Department of Computer Science, Purdue University, West Lafayette, IN, USA. He is currently a Professor with University of North Carolina at Charlotte. His research interests include image/video privacy protection, automatic image/video understanding, and large-scale deep learning.