A Few Shot Transfer Learning Approach Identifying Private Images With Fast User Personalization

Edoardo Serra, Sujeet Ayyapureddi, Qudrat E Alahy Ratul
Boise State University,
Boise, Idaho
{edoardoserra, sujeetayyapureddi,qudratratul}@boisestate.edu

Anna C Squicciarini
Pennsylvania State University
University Park, PA,
acs20@psu.edu

***Abstract*—As online image sharing has become commonplace, researchers have acknowledged the need to assist users in detecting sensitive (or private) images. However, image privacy classification tasks have shown to be nontrivial, as the designation of an image sensitivity requires considerations of the visual concepts in the image. In this paper, we propose an innovative framework that combines the power of knowledge transfer for efficient, personalized learning of individuals' privacy preferences toward images.**

Our approach defines a meta-model, which, given the query image and a small set of labeled images (used for the user-privacy customization), identifies if the query image is private for a target user. A generic user can efficiently customize this model by providing a small labeled training set. Moreover, our proposed framework includes transfer learning techniques to import basic patterns for image processing learned from other domains. Transfer learning enables fast and accurate processing of images, and allows few shot learning to focus on customization. This helps speed up the training process and avoid risk of overfitting. Our proposed framework significantly outperforms several baselines, including advanced object-oriented approaches and other CNN-based methods.

I. Introduction

The steep increase in the number of images that are shared online through social media apps and portals has highlighted the need for effective and efficient methods to assist users in identifying private images [3], [15]. Accordingly, a plethora of learning models attempting to learn image privacy designations has been recently proposed [20], [24], [25], [28]. Yet, automated classification of an image as private or public has shown to be challenging due to the need of capturing images sensitivity, when the definition of sensitive or private content may be subjective, and similar labeled images may be scarce. In general, a user's privacy expectation for a given image is related to specific contents therein, the users' privacy preference, and other contextual factors (expected audience, etc.) [32]. Contextual data surrounding an online image is however often difficult to collect, either unavailable, inconsistent, or hard to obtain without violating others' privacy or platforms' terms of use.

A customized algorithm for privacy designation for a given image would require that each user make available a representative data set for an algorithm to learn from and provide predictions. Ideally, the dataset should be sufficiently large to offer high accuracy guarantees. Yet, assuming the availability of such an (individualized) large representative dataset is unrealistic. The average user may not have a large number of images, nor be motivated to go through the manual labor of labeling their data, or even be willing to disclose their data to sophisticated software for privacy analysis.

We argue that an effective approach to help users identify private or sensitive images needs to learn from *small data*, and yet be specific to the visual content therein. Small data should not only enable personalized models, but also limit expensive and controversial profiling approaches. As such, typical machine learning approaches that learn from big data cannot be immediately used.

In this paper, we present an innovative framework that *combines the power of knowledge transfer with efficient, personalized learning*. Our approach provides a trained meta-model that, combined with a small training set of labeled images (from 4 to 16 private and public images in our experiments), helps users identify images with private content according to their own small training set. The meta-model takes advantage of transfer learning during training, so as to import basic patterns for image processing that are learned from other domains. Once the meta-model is trained, a generic user can customize this model by providing a small labeled training set. Customization is based on a few shot learning method and is therefore immediate: it does not require expensive computational resources and can be done simply on the user's machine.

Our experimental results, carried out on a large dataset of social network images, show that our method significantly outperforms several baselines, including advanced object-oriented approaches and other CNN-based methods. Precisely, our framework achieves an overall accuracy of 85% F-1 score with only 16 private and public images used for customization, showing a significant performance gain against classic transfer learning method and (65%) object-oriented few-shot learning (33%).

Our model generalizes well even with new private categories (not used for meta-training) and achieves f1-scores in the $[0.82, 0.85]$ range, depending on the new category considered.

Our contributions can be summarized as follows:

- We provide a customized learning model for image privacy. The model requires a minimal amount of labeled data and computational resources.
- We provide a robust modification of the few-shot learning approach that better supports user's customization of the

model. This solves the problem of multiple inconsistent labels generated by different user profiles.
- We integrate few-shot learning with transfer learning to improve generalization during the training and customization phases of our model.
- We collect and test our approach on a new large image dataset with different potential private categories.

The remaining of the paper is organized as follows. In the next section, we review related work. Next, we present our methodology, including a brief discussion of the two main models underlying our framework. In Section IV-A2, we present our experimental results carried out on our own dataset. In Section V we conclude the paper with pointers for future directions.

II. Related Work

Several recent works have developed approaches for automated privacy settings of images [2], [10], [13], [20], [26], [28], [34], [35], [37]. For instance, Buschek et al. [4] presented an approach to assign privacy settings to shared images using metadata (location, time, shot details) and visual features (faces, colors, edges). Zerr et al. [35] proposed privacy-aware image classification, and learned classifiers on Flickr photos using content features. These approaches mostly rely on the assumption that users make privacy decisions consistent with socially accepted definitions of privacy [32]. Accordingly, the authors focus on finding "universal" features and a universal model for image privacy detection, using a combination of features drawn from text or meta-data analysis, object detection or other conventional image-specific features (e.g., Scale-invariant feature transform (SIFT), color histogram deep learning (DL)) [20], [24]. However, these methods fail to account for individual preferences and rely on the assumption that a large universal model exists and can be trained.

Consistent with the recent success of Convolutional Neural Networks (CNNs) on a large scale dataset used for object recognition, e.g. [11], Tonge and Caragea analyzed various CNN architectures and modalities and achieve more accurate binary privacy predictions than earlier models [24], [25], [27], reaching a F-1 score of 86.3%. Tonge and colleagues [26] also looked into the combination of user tags and object tags for an alternative to analysis based on visual features alone.

Our approach, discussed next, also employs neural networks for part of the framework but is significantly superior as it reaches comparable if not higher performance than prior approaches, and it provides superior adaptability and personalization, which are key concerns in privacy problems [32].

To address these limitations, few studies have explored personalized privacy models using DL features [19], [38]. Spyromitros' approach is based on a trained logistic regression model for every user, using a dataset with both personalized labels and a generic label sample. This method relied on many labeled examples from each user [19]. Zhong and colleagues [38] offered a statistical approach attempting at a a compromise between these strategies, with more flexibility than the single model approach, less personalization than a truly customized approach, but borrowing statistical strength from alike users to reduce labeled data requirements. With a baseline profile of 15 images per user's group or profile, authors achieve an accuracy of 79.31%.

Yu et al. [33] proposed a framework that identifies a large set of privacy-sensitive object classes and their privacy settings by using a large set of labeled public and private images. This framework is based on object segmentation and provides recommendations of regions needing to be blurred. The recommendations are based on the privacy settings of the objects detected in the images. This approach varies from our proposed solution, in that it does not cater to the cases of limited labelled image availability. It also requires a large amount of training data to achieve high accuracy.

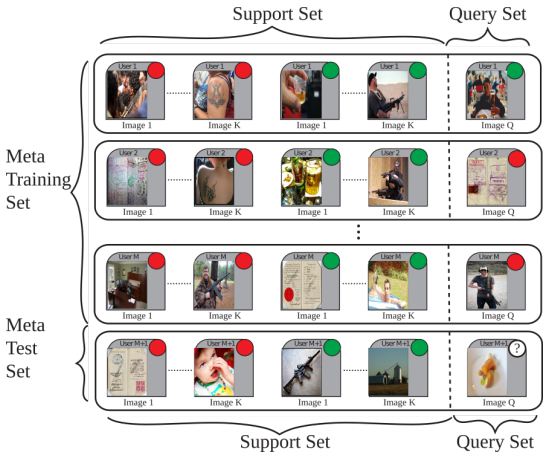

Fig. 1: Example of meta learning

III. Methodology

Our goal is to provide a powerful and *personalized* model that quickly identifies images deemed private by an individual. Our framework relies on the observation that users typically apply their own privacy definitions (or profiles, in what follows) when selecting images as private or public [1], [30]. These privacy profiles may result in the same image having different decisions (or labels) by different individuals, but can be learned given sufficient labeled images.

To capture individuals' models, our framework aims at learning users' privacy profiles based on few personal images, and integrates labeling preferences of similar users' for higher accuracy. Operationally, this is achieved by developing a learning framework that carefully combines learning approaches blending knowledge transfer with customized models.

Next, we provide some background information on two key learning models underlying our solution. We then discuss our proposed centroid-based framework, that integrates few-shot learning and transfer learning successfully. In addition, as a possible baseline for our approach, we propose a modification of the centroid-based approach that integrates object detection.

A. Background Methods

We rely on two key building blocks for our framework: 1) transfer learning to extract knowledge from small data samples, and 2) few-shot learning to allow training models to be customized based on few examples.

1) Transfer Learning: Transfer learning leverages knowledge from a related domain (called source domain) to improve learning performance or minimize the number of labeled examples required in a target domain. The closer the source and target domains are, the more effective is the transfer of knowledge.

We consider homogeneous learning [31], which addresses learning when source and target domains have the same feature space. In particular, we use Network-based deep transfer learning. Network-based deep transfer learning refers to the reuse of a partial network, along with its network structure and connection parameters that is pre-trained in the source domain. This structure is transferred to be a part of the deep neural network to be used in the target domain [23]. The reuse of such a network is done by fine-tuning the network for a few epochs. The learning rate is specific for each layer of the network with the initial layers having a smaller learning rate and the subsequent deep layers having an increasingly larger learning rate. The intuition is that the layers at the beginning of the network learn generic image patterns, such as edges. Deeper layers learn more specific patterns for the application, e.g., cat, mountain, wolf, etc. The fine-tuning affects the patterns specific for the application domain (in the deeper layers) and less the generic patterns (in the initial layer) useful to process generic images. As we present in Section IV-A2, we use Inception V3 [22] ResNet50 [8] and DenseNet201 [9] networks for transfer learning pre-trained on the Image Net dataset [5]. These networks offer a good compromise between the size of the network and the performance in terms of image recognition accuracy.

2) Few-Shot Learning: Few-shot learning attempts to discriminate between *N* classes with *K* examples of each (*N*-way-*K*-shot classification). Few-shot learning approaches the problem of a small data set by learning from similar problems (i.e., meta-learning). Accordingly, this learning model is effective in scenarios where training data is hard to find or where labelling data is expensive [6], [7]. These two issues are acute in privacy prediction tasks, making few-shot learning an excellent candidate for our image privacy problem. One popular way to deploy few-shot learning is through meta learning. Figure 1 illustrates am application of meta-learning for our binary image privacy classification problem. The goal of the meta-leaner is to learn from the given meta-training set and make a prediction on the query set of the meta test set. Each row pertaining to a user acts as a task which mimics a *N* way *K* shot classification task. Here, the goal is to classify the query set for $\mathrm{u}_{\mathrm{M}+1}$. The meta learner learns from the *N*-way-*K*-shot classification of the previous *M* users. The support set and query set in every task are used to mimic the support set and the query set of the meta test set. The support and query set of the meta training sets include labelled data. The support set of the meta test set is also labelled.

Model parameters are updated at each step of the meta training. The loss function, a cross-entropy function [16], varies with the performance of the model on the query set based on the knowledge of the support set.

One the most largely used networks for few-shot learning is the *matching network* proposed in [29]. Two networks, g and f, extract features from the images of the support set and the query image, respectively. Then, similarity a score, based on cosine similarity, is computed among the features of the query image and the features of each image in the support set. Given a query image Q, a set of public images $\{IMG_1, \ldots, IMG_k\}$ and a set of private images $\{IMG_{k+1}, \ldots, IMG_{2 \cdot k}\}$, the above mentioned similarities are used in a nearest-neighbours classification described by the equation below:

$$\hat{y} = \frac{\sum_{i=1}^{k} a(g(IMG_i), f(Q)) - \sum_{i=k+1}^{2 \cdot k} a(g(IMG_i), f(Q))}{\sum_{i=1}^{2*k} a(g(IMG_i), f(Q))}$$

where $a(x, y) = \exp x \cdot y$ and $\hat{y} \in [-1, 1]$ is the predicted class (1 public and -1 private).

A weakness of this approach is that the use of the nearest-neighbor classification function limits the expressive power of this classifier.

These limitations can lead to an under-fitting behavior. We confirm this under-fitting issue through our empirical evaluation in Section IV-B1. Our proposed approach (discussed next) extends and customizes the matching network [29]. We add a decision function (see Sect. 3.2.) represented by a fully connected neural network that increases the expressive power of the model and better deals with the inconsistent labels typical of our application domain.

B. A Centroid-based Few-Shot Transfer Learning Model

We propose a Centroid-based Few-Shot Transfer Learning (CFSTL) model for fast customized learning. The framework's architecture is shown Figure 2, and described next.

As shown, the model takes as input one query image IMG^Q that is to be classified as private or public, *K* sample images $IMG_1^{pu}, \ldots, IMG_K^{pu}$ labeled by user u as public and *K* sample images $IMG_1^{pr}, \ldots, IMG_K^{pr}$ labeled by the user as private. Both sets of private and public images represent the support set for few-shot learning. Each image is initially processed with a unique convolutional neural network (the network and its weights are the same for each image), extracting feature embeddings from the image. To extract the embeddings, we adopt Inception V3 convolutional neural networks (NNs) [22]. We denote the embedding vector extracted from image IMG with the Inception V3 as $V3(IMG)$, with $V3(IMG) \in \mathcal{R}^h$.

The embeddings of all the private images are averaged, and a private embedding centroid C_{pr} is calculated, i.e., $C_{pr} = \frac{\sum_{i=1}^{K} V3(IMG_i^{pr})}{K}$. Similarly, a public embedding centroid $C_{pu} = \frac{\sum_{i=1}^{K} V3(IMG_i^{pu})}{K}$ is calculated. The private and public centroids (C_{pr} and C_{pu}, respectively) are accurate representations of the user's u privacy preferences, as defined

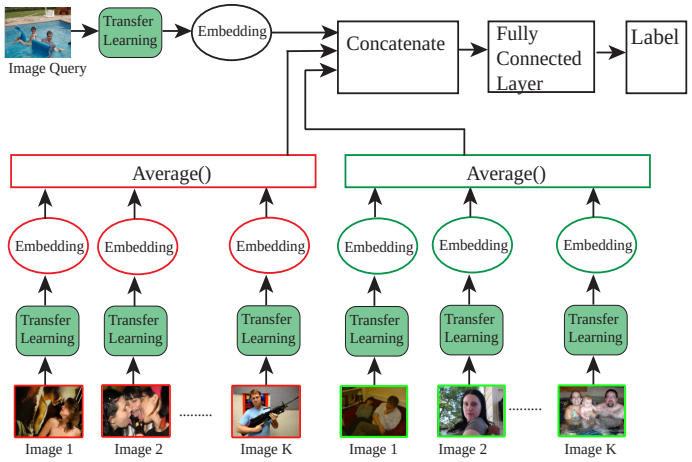

Fig. 2: CFSTL architecture

by the support set since $V3(\_)$ is fine-tuned during the meta training set.

Next, a concatenation of the embedding of the query image $V3(IMG^Q)$ and the two public and private centroids C_{pr} and C_{pu} is passed through a fully connected neural network with two layers. Such a fully connected neural network represents the decision function determining the final image label.

Model The CFSTL model is more formally defined as follows:

$$Q = V3(IMG^Q) \tag{1}$$

$$C_{pu} = \frac{\sum_{i=1}^{K} V3(IMG_i^{pu})}{K} \tag{2}$$

$$C_{pr} = \frac{\sum_{i=1}^{K} V3(IMG_i^{pr})}{K} \tag{3}$$

$$r = [Q \parallel C_{pu} \parallel C_{pr}] \tag{4}$$

$$hl = tanh(W_1 * r + B_1) \tag{5}$$

$$\hat{p} = \sigma(W_2 * hl + b_2) \tag{6}$$

Equations 5 and 6 represent the fully connected network where $W_1 \in \mathcal{R}^{h \times h}$, $B_1 \in \mathcal{R}^h$, $W_2 \in \mathcal{R}^{1 \times h}$, $b_2 \in \mathcal{R}$. $tanh(\_)$ denotes the tangential function, and σ is the sigmoid function. $\hat{p}$ in Equation 6 denotes the probability that image IMG^Q is private, and $1 - \hat{p}$ is the probability that the image is public.

Meta-training The meta training of the CFSTL model is performed by minimizing the cross-entropy on the output of the fully connected network. Let ex_i be a generic example in the training set TR of the form $ex_i = (IMG^Q, [IMG_1^{pu}, \ldots, IMG_K^{pu}], [IMG_1^{pr}, \ldots, IMG_K^{r}])$, the cross entropy function is defined as follows:

$$loss = \sum_{ex_i \in TR} p_i \cdot \log \hat{p}_i + p_i \cdot \log (1 - \hat{p}_i) \tag{7}$$

where p_i is the ground truth label of IMG^Q in ex_i and $\hat{p}_i$ is the estimated probability that the IMG^Q is private ($\hat{p}$ in Equation 6) returned by CFSTL model for the example ex_i. Note that each example i, as per Figure 1, is formed by a support set (i.e., private and public lists of images) and a query image.

C. An Object-Centric Framework for CFSTL

We modify the CFSTL model to develop an object-centric framework, referred to as OBJ-CFSTL. This is a *strong* baseline of our proposed CFSTL model. The reasoning for this object-centric model is intuitive. As the privacy of an image can revolve around identifying some key visual elements [33], [37], object segmentation offers the possibility of identifying and labeling different objects contained in an image.

We extract an object detection binary vector from an input image, using the pyramid scene parsing network [36]. The object vector denotes the presence (or absence) of objects contained in the image.

Specifically, given an image IMG, $ObS(IMG)$ denotes the binary vector output of the object segmentation network. The size of the vector $ObS(IMG)$ is the possible kinds of objects that the network can recognize [1]. In our experiments, we use an object recognition model that has $l = 1000$ different kind of objects. Rather than the location or the number of different objects present in the image, it is important to perform image attribution, i.e. learn what content is in fact in the image. $ObS(IMG)$ is passed through a neural network feature transformer, i.e. a fully connected layer with a number of inputs equal to the number of outputs. The transformer produces the final h-dimensional embedding of the image. We denote the feature transformer network as $FT : \mathcal{R}^l \rightarrow \mathcal{R}^h$.

Accordingly, the object detection module represents a pre-processing step, and it does not change the overall learning framework past the meta training step: the feature transformer function is a part of the entire network and will be trained by the few-shot learning meta-training. We modify the CFSTL formulation (eq. (2), (3), and (4)) as follows:

[1]Clearly, an image may include multiple objects of the same kind

$$Q = FT(ObS(IMG^{Q})) \tag{8}$$

$$C_{pu} = \frac{\sum_{i=1}^{K} FT(ObS(IMG_i^{pu}))}{K} \tag{9}$$

$$C_{pr} = \frac{\sum_{i=1}^{K} FT(ObS(IMG_i^{pr}))}{K} \tag{10}$$

Equations (5) to (7) are the same as the original architecture. Meta-training is also performed with the same cross-entropy function of the CFSTL model (Eq 7).

IV. Experiments

A. Dataset, Experiment Design, and Metrics

1) Dataset: In order to generate a representative dataset, we took a two-pronged approach. First, we characterize a set of visual concepts $\{c_1, ..c_k\}$, with each c_i being a potentially sensitive concept (e.g. nudity, violence or gore etc). Images may include one or more distinct visual concepts (e.g., a tattooed person drinking alcohol).

Accordingly, we selected eight visual concepts. The selected visual concepts are based on known sensitive concepts according to the existing literature on privacy and images [1], [14], [19], [30], [38]. Next, we collected images with an open commons license from the Flickr platform, using search functions of each visual concept and manually curated for accuracy. We considered images with at least two sensitive visual concepts to be private.

TABLE I: Image categories and count

Category ID	Description	Size
1	Public images	21,869
2	Kids	200
3	Weapons	250
4	Documents	192
5	Alcohol	247
6	Nudity in closed space	200
7	Closed space	220
8	Tattoo	211
9	Violent scenes	207
	Total	23596

Public images were instead taken from the Picalert repository [34]. The original study released a collection of images downloaded by Flickr. Images were annotated manually as private or public. We selected images that were not labeled by any original human labeler as private. Note that we could not use Picalert private images as the majority of the original private images are no longer available or unsuitable (e.g. too small or not licensed for public use) for research use.

Per Table I, the final dataset included 23,596 images, with 92% of them being of public nature. This ratio is consistent with the actual public vs. private image ratio in online platforms [20], [26].

For testing purposes, we introduce the notion of user *privacy profiles*, which may be defined as the combination of visual concepts deemed either private or public by an individual. A privacy profile is denoted as the subset S of the visual concepts deemed private by a given individual. In other words, given a user u, the list $S_u \subseteq \{c_1, \cdots, c_k\}$, denotes the types of images considered private by user u. In our dataset, 2^8-1 user privacy profiles can be identified, i.e., all possible combinations in the set of categories from 2 to 9 after removing the empty set.

We acknowledge that this dataset has some limitations, in that it does not actually come from users' private image repositories. However, our dataset is created consistently with a vast state of the art, that has shown how users rely on privacy mental models for decision making, which support use of generic categories for private concepts [3], [12], [26]. Further, as acknowledged by recent studies, the careful design of a realistic dataset against prototypical users' archetypes allows us to experiment with sensitive and private content *without raising ethical concerns*, given that personal images are naturally challenging to obtain, and platforms' terms of use prohibit crawling of protected content [37].

2) Training and Model Settings: Recall that CFSTL combines the power of two approaches, transfer learning, and few-shot learning. In regards to pure transfer learning, we perform 5-fold cross-validation according to the public and private label specific to a given user profile upp. For each fold, a training set is provided to fine-tune the pre-trained convolutional neural network from the transfer learning approach, and the obtained model is tested on the test set. Performance results are averaged for each privacy profile and each fold.

In regards to few-shot learning and its combination with transfer learning, we use the following settings. Given a user privacy profile, we randomly select a subset of users' profiles, denoted as UPP_{tr}, to use in the meta training. The remaining profiles, denoted as UPP_{te}, are for meta testing.

Once user profiles are divided into meta training and meta testing, we perform, within the meta training and meta testing sets, 5-fold cross-validation stratified according to the categories of the image. For each fold, the set of images for the meta training IMG_{tr} and the set of images for the meta test IMG_{te} are disjointed, i.e., $IMG_{tr} \cap IMG_{te} = \emptyset$. Specifically, to create the meta training for each of the five folds, given UPP_{tr}, we created 10 versions of every profile by randomly selecting K images from the image training set IMG_{tr} that were classified as public by upp, K images classified as private by upp, and one image in IMG_{tr} used as the query image. We perform the same steps for meta testing, with IMG_{tr} replaced by IMG_{te} and UPP_{tr} replaced by UPP_{te}.

The user privacy profiles amount used for training is equal to $|UPP_{tr}| = 128$ and $|UPP_{te}| = 127$ (the total amount is 127 that is the number of all the subsets of the categories included in Table I except the public image category and the excluding the empty set), and the number of images K for each class (private and public) in the support set is 16.

Note that we purposely set a large percentage of user profiles in the test set (around 50%) as we need to check the ability of the few-shot learning framework infer profiles that are different from the one in the training set, and make inference non-trivial, so as to verify the quality of our perfor-

mance.

During neural network training, the meta training is generated again at each epoch to guarantee a significant diversification of the examples provided to the model. In the meta test, the number of examples, for each privacy user profile in UPP_{te}, is pushed to 100 instead of 10. The training process is performed over Nvidea 2080Ti GPU card with batch size 32 (please note that one instance of the bach contains K private images, k public images and the query images) over 100 epochs. We use macro precision, recall and f1-score for performance metrics.

B. Experimental Results

We carried out two sets of experiments. First, we compared our proposed CFSTL approach with several powerful baselines, and second, we analyzed the performance of CFSTL under various conditions. CFSTL is deployed using Inception V3 for feature extraction, with the embeddings size h set to 1000. The resolution of the image in input is set by the as $224 \times 224 \times 3$ where the third dimension indicates three RGB channels. We also present in Section IV-C our qualitative attribution analysis, related to the ability of our model to detect sensitive portions of images that lead to a private label.

1) Baselines and Transfer Learning Networks.: We compare our Centroid-based Few Shot Learning (CFSTL) method (using Inception V3) with the following three baselines:

- **Inception V3 Transfer Learning (IV3-TL)**: A pretrained network (with Image Net over 1000 classes) provided by the Keras library that we use as a transfer learning approach. It is fine-tuned (by discarding the last classification layer) for each user privacy profile with the 90% of the dataset, as described in Section III-A1. The parameters of this network since it already trained are fixed and a details can be found in Keras library. This approach is not directly comparable with the few shot learning approaches, since alone it does not customize with few example the privacy profile of the user. Then it is not applicable but we report its results only for comparison.
- **Matching Network Few Shot Learning (MN-FSL)**: A highly popular few-shot learning approach described in Section III-A2 that uses Inception V3. The parameter of this network depends by retrained Inception V3 (from the Keras library) which provides a representation of size 1000. The construction of the marching network is done on the representation size of Inception V3 and the remaining parameters are the same of our CFSTL approach.
- **Object Segmentation Few Shot Learning (OS-FSL)**: A modification of our centroid based few-shot learning approach based on the Object Segmentation tasks (see Section III-C). In term of parameters, this model present the same parameters of our CFSTL approach. The unique difference is the substitution of the Inception V3 automatic feature extraction with precomputed features extracted by the object segmentation task and describing the type of objects found in the segmentation.

TABLE II: Baseline Comparison

Techniques	Precision (%)	Recall (%)	F1-score (%)
IV3-TL	84	61	66
MN-FSL	51	51	51
OS-FSL	25	50	33
CFSTL	**85**	**85**	**85**

TABLE III: Different transfer learning networks in CFSTL

Techniques	Precision (%)	Recall (%)	F1-score (%)
CFSTL Inception V3	**85**	**85**	**85**
CFSTL ResNet50	83	82	83
CFSTL DenseNet201	85	83	84

In Table II we report our results in terms of average macro precision, recall, and f1-score. The results show that CFTSL approach has the highest values across all performance measures. The matching network (MN-FSL) performs poorly. This can be explained by the lack of personalization offered by MN-FSL, and as a consequence ignores the fact that different profiles can assign a different label to the same image. As such, in the training set, the model observes many apparent inconsistencies justified only by the support set. MN-FSL uses a nearest-neighbors classification function based on cosine similarity, and such function has a small movement margin to justify such inconsistencies. In fact, we observed during the training phase, a strong under-fitting behavior.

In the OS-FSL case, the lower performance is likely due to the predefined number of object categories used for object recognition that is not sufficiently fine-grained for our multi privacy profile characterization. Another noticeable result is that even if the transfer learning approach (IV3-TL) provides a large training set for each user privacy profile, the results in terms of recall and then f1-score are significantly below CFSTL.

In Table III, we report the experimental results of using CFSTL with different transfer learning networks (Inception V3, ResNet50, and DenseNet201). All the results are similar to one another, with Inception V3 achieving highest performance, followed by DenseNet201.

2) Impact of Images' Support Set and of the User Privacy Profiles: We test the ability of our model to work with a varying number of images during the training phase. This is important, as it allows us to verify whether CFSTL can adapt and learn even in non-ideal training conditions.

We carry out two sets of experiments. First, we evaluate the performance of CFSTL varying the number of images in the support set and the amount of the user profiles used in the meta training phase (k= 4, 8, 12, 16, respectively). In this case, the percentage of user privacy profiles considered in training is 50% of the possible profiles (255 user privacy profiles). As shown in Figure 4, we observe that with only 12 or 16 private and public images, our approach is able to provide around

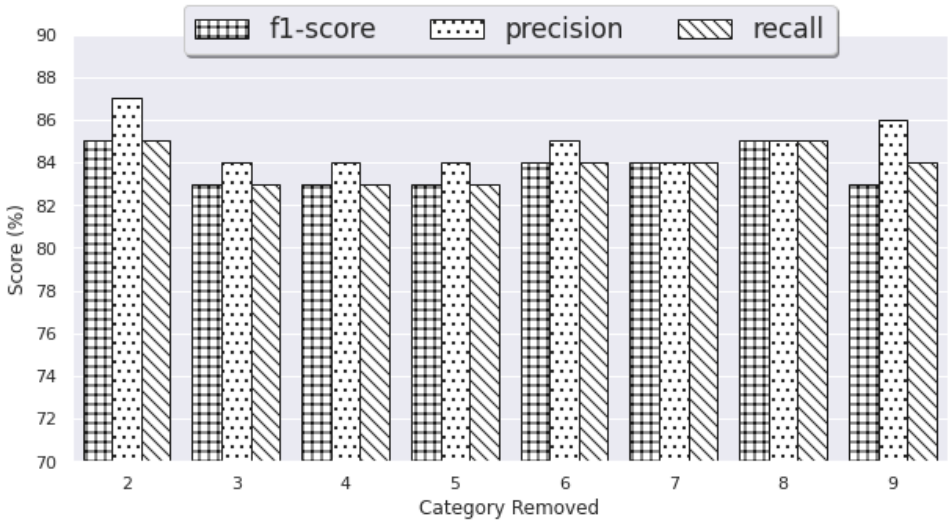

Fig. 3: CFSTL performance by removing one image category from the meta training phase

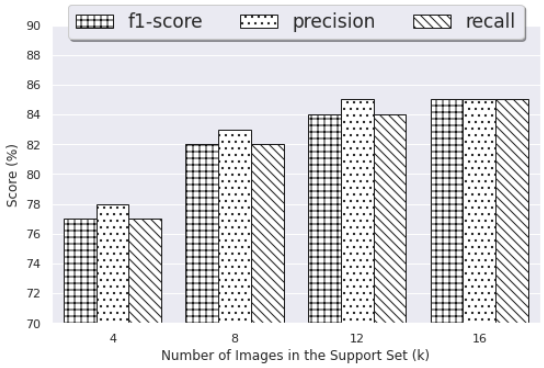

Fig. 4: Performance with varying number of images in the support set

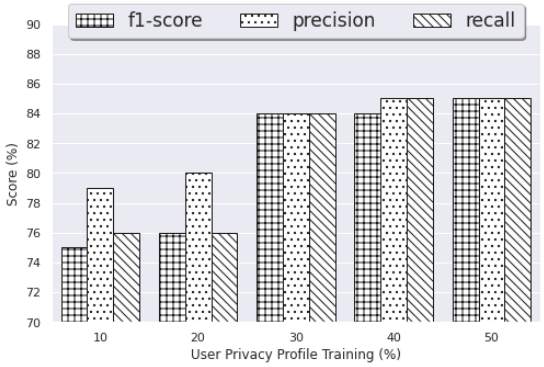

Fig. 5: Performance with varying user privacy profiles in meta-training

85% of f1-score, precision, and recall. However, reducing the number of images in training ($k = 4$) significantly impacts all three metrics.

In our second experiment, we vary the number of user privacy profiles in the training set. The number of images K for each of the two classes in the support set is fixed at 16. Figure 5 shows that as the number of user privacy profiles increases in the meta training phase, performance increases across all metrics. Notably, with only 30% of the available user privacy profiles for training, our approach reaches 84% F-1 score, and other metrics are equivalently strong.

3) Unknown Category in the Meta Test: We test the ability of our model to work with categories of images that were not used during meta training. Accordingly, we modify the meta training steps by removing all the images belonging to a given category, and we evaluate precision, recall, and f1-score on the meta test set, which also includes the previously removed category.

In Figure 3 we report three performance measurements (precision, recall, and f1-score) for each removed category (x-axis). The number of images K for each of the two classes in the support set is fixed at 16. As shown, removing a class from the meta-training does not change the overall performance of our approach. This result is confirmed regardless of the exact category removed, showing our approach's potential to generalize to unknown categories.

C. Qualitative Attribution Analysis

We use attribution techniques to interpret the classification results of our model according to the specific private category of images used in the support set. More specifically, an attribution procedure highlights the region of the image that is most relevant to the classification result. Specifically, for attribution we employ: gradient [17], saliency map [18], and integrated gradient [21].

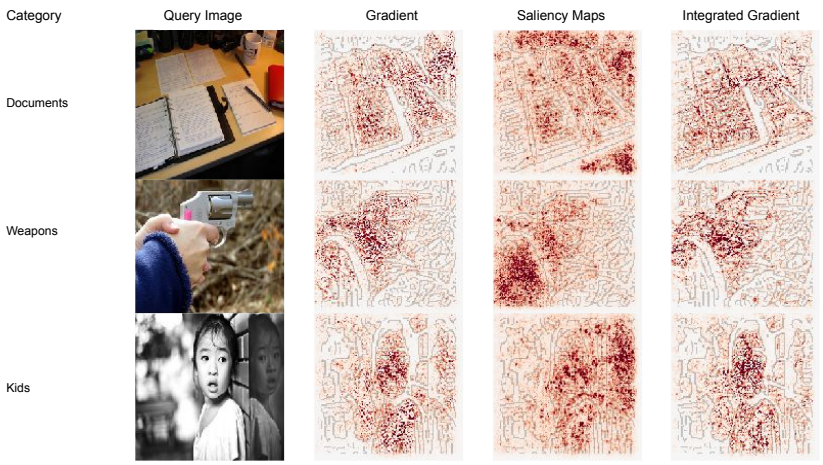

Fig. 6: CFSTL sample attribution analysis on different private categories with gradient, saliency maps, and integrated gradient.

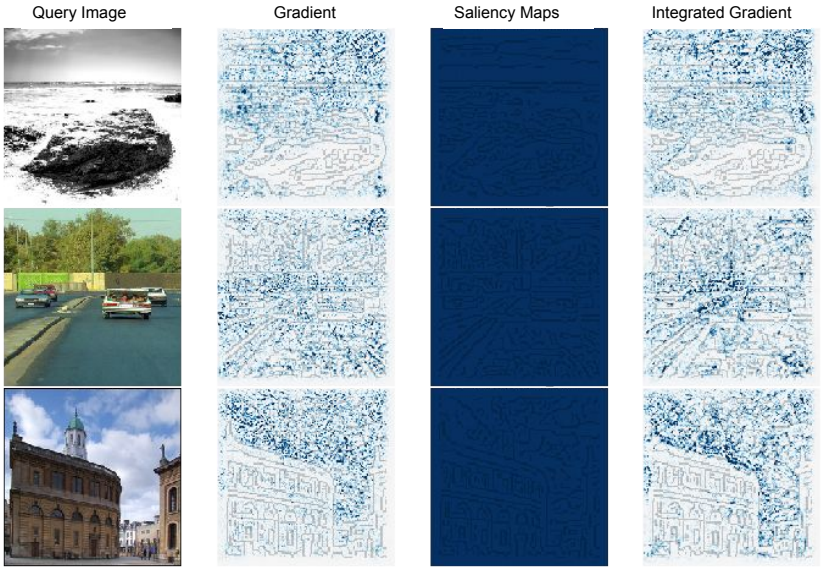

Fig. 7: CFSTL sample attribution analysis for public images with gradient, saliency maps, and integrated gradient.

In Figure 6, we report three different query images, their categories, and their attribution results. We randomly sample a support set where the private images only belong to the specific category of the query image. We can observe that gradient and integrated gradients provide the clearest results. Specifically, in the case of the document image, the integrated gradient highlights mostly the documents placed on the desk. In the weapon image, the highlights focus on the gun and the hands that are holding it. Finally, in the bottom image, the entire figure of the child is highlighted. Figure 7 shows attribution results for public images. The evident result is that consistently, all of the tested methods highlight the entire area of the image. This preliminary result shows that our model reliably understands the privacy concept provided in the support set. Privacy concepts may be used to provide an explanation of the labels applied.

V. Conclusion

In this paper, we presented an effective learning framework to address image privacy binary classification. Our proposed approach combines the power of unlabeled images with personalization models to achieve high accuracy on a variety of types of images. Our experiments are carried out on a large dataset of social network images and significantly outperform several baselines, including advanced object-oriented approaches and other CNN-based methods.

We plan to extend our framework in several ways. First, we will extend our explanatory analysis, so as to provide systematic spatial attribution and offer end-users some justifications of the classification results. Further, it would be helpful to provide mechanisms to help users with a selection of images that can help improve the support set for effective learning. The support set is crucial for the customization steps of our framework, and a carefully selected support set can improve

our classification performance further. We will explore how to integrate more user-specific labels, as well as extend our binary classification into a multi-label problem in order to support more fine-grained labels beyond the two public/private extremes currently used.

ACKNOWLEDGEMENTS

Portion of the work from Dr. Squicciarini was supported by a National Science Foundation Grant n. 1453080. This research was made possible by the National Science Foundation award #1820685 and Idaho Global Entrepreneurial Mission/Higher Education Research Council #IGEM22-001.

REFERENCES

[1] Ahern, S., Eckles, D., Good, N.S., King, S., Naaman, M., Nair, R.: Over-exposed?: Privacy patterns and considerations in online and mobile photo sharing. In: Proceedings of the SIGCHI Conference. pp. 357–366. CHI '07, ACM (2007)

[2] Backes, M., Gerling, S., Lorenz, S., Lukas, S.: X-pire 2.0: A user-controlled expiration date and copy protection mechanism. In: Proceedings of the 29th Annual ACM Symposium on Applied Computing. pp. 1633–1640. ACM (2014)

[3] Bonneau, J., Preibusch, S.: The privacy jungle: On the market for data protection in social networks. In: Economics of information security and privacy, pp. 121–167. Springer (2010)

[4] Buschek, D., Bader, M., von Zezschwitz, E., De Luca, A.: Automatic privacy classification of personal photos. In: Abascal, J., Barbosa, S., Fetter, M., Gross, T., Palanque, P., Winckler, M. (eds.) Human-Computer Interaction – INTERACT 2015, vol. 9297, pp. 428–435 (2015)

[5] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 248–255. IEEE (2009)

[6] Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE transactions on pattern analysis and machine intelligence **28**(4), 594–611 (2006)

[7] Fink, M.: Object classification from a single example utilizing class relevance metrics. In: Advances in neural information processing systems. pp. 449–456 (2005)

[8] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)

[9] Huang, G., Liu, Z., Weinberger, K.Q.: Densely connected convolutional networks. corr abs/1608.06993 (2016). arXiv preprint arXiv:1608.06993 (2016)

[10] Klemperer, P., Liang, Y., Mazurek, M., Sleeper, M., Ur, B., Bauer, L., Cranor, L.F., Gupta, N., Reiter, M.: Tag, you can see it!: Using tags for access control in photo sharing. In: SIGCHI Conference on Human Factors in Computing Systems. pp. 377–386. ACM (2012)

[11] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C., Bottou, L., Weinberger, K. (eds.) Advances in Neural Information Processing Systems 25, pp. 1097–1105. Curran Associates, Inc. (2012)

[12] Parra-Arnau, J., Rebollo-Monedero, D., Forné, J.: Measuring the privacy of user profiles in personalized information systems. Future Generation Computer Systems **33**, 53–63 (2014)

[13] Ra, M.R., Govindan, R., Ortega, A.: P3: Toward privacy-preserving photo sharing. In: Presented as part of the 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13). pp. 515–528 (2013)

[14] Ravichandran, R., Benisch, M., Kelley, P.G., Sadeh, N.M.: Capturing Social Networking Privacy Preferences:, pp. 1–18. Springer Berlin Heidelberg (2009)

[15] Sawyer, S., Griffiths, M., Light, B., Lincoln, S., Bateman, P.J., Pike, J.C., Butler, B.S.: To disclose or not: Publicness in social networking sites. Information Technology & People (2011)

[16] Shore, J., Johnson, R.: Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy. IEEE Transactions on information theory **26**(1), 26–37 (1980)

[17] Shrikumar, A., Greenside, P., Shcherbina, A., Kundaje, A.: Not just a black box: Learning important features through propagating activation differences. arXiv preprint arXiv:1605.01713 (2016)

[18] Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: Visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034 (2013)

[19] Spyromitros-Xioufis, E., Petkos, G., Papadopoulos, S., Heyman, R., Kompatsiaris, Y.: Perceived versus actual predictability of personal information in social networks. In: International Conference on Internet Science. pp. 133–147. Springer (2016)

[20] Squicciarini, A.C., Caragea, C., Balakavi, R.: Analyzing images' privacy for the modern web. In: Proceedings of the 25th ACM conference on Hypertext and social media. pp. 136–147. ACM (2014)

[21] Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 3319–3328. JMLR. org (2017)

[22] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016)

[23] Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., Liu, C.: A survey on deep transfer learning. In: International conference on artificial neural networks. pp. 270–279. Springer (2018)

[24] Tonge, A., Caragea, C.: On the use of" deep" features for online image sharing. In: Companion Proceedings of the The Web Conference 2018. pp. 1317–1321 (2018)

[25] Tonge, A., Caragea, C.: Dynamic deep multi-modal fusion for image privacy prediction. In: The World Wide Web Conference. pp. 1829–1840 (2019)

[26] Tonge, A., Caragea, C., Squicciarini, A.: Privacy-aware tag recommendation for image sharing. In: Proceedings of the 29th on Hypertext and Social Media, pp. 52–56 (2018)

[27] Tonge, A.K., Caragea, C.: Image privacy prediction using deep features. In: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA. pp. 4266–4267 (2016)

[28] Tran, L., Kong, D., Jin, H., Liu, J.: Privacy-cnh: A framework to detect photo privacy with convolutional neural network using hierarchical features. AAAI 2016 (2016)

[29] Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al.: Matching networks for one shot learning. In: Advances in neural information processing systems. pp. 3630–3638 (2016)

[30] Wang, Y., Norice, G., Cranor, L.F.: Who is concerned about what? a study of american, chinese and indian users' privacy concerns on social network sites. In: International Conference on Trust and Trustworthy Computing. pp. 146–153. Springer (2011)

[31] Weiss, K., Khoshgoftaar, T.M., Wang, D.: A survey of transfer learning. Journal of Big data **3**(1), 9 (2016)

[32] Xu, H., Teo, H.H., Tan, B.C., Agarwal, R.: Research note—effects of individual self-protection, industry self-regulation, and government regulation on privacy concerns: A study of location-based services. Information Systems Research **23**(4), 1342–1363 (2012)

[33] Yu, J., Zhang, B., Kuang, Z., Lin, D., Fan, J.: iprivacy: image privacy protection by identifying sensitive objects via deep multi-task learning. IEEE Transactions on Information Forensics and Security **12**(5), 1005–1016 (2016)

[34] Zerr, S., Siersdorfer, S., Hare, J., Demidova, E.: Privacy-aware image classification and search. In: Proc. of the 35th international ACM SIGIR conference on Research and development in information retrieval. pp. 35–44. SIGIR '12 (2012), http://doi.acm.org/10.1145/2348283.2348292

[35] Zerr, S., Siersdorfer, S., Hare, J., Demidova, E.: Privacy-aware image classification and search. In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. ACM, NY, USA (2012)

[36] Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2881–2890 (2017)

[37] Zhong, H., Li, H., Squicciarini, A., Rajtmajer, S., Miller, D.: Toward image privacy classification and spatial attribution of private content. In: 2019 IEEE International Conference on Big Data (Big Data). pp. 1351–1360. IEEE (2019)

[38] Zhong, H., Squicciarini, A.C., Miller, D.J., Caragea, C.: A group-based

personalized model for image privacy classification and labeling. In: IJCAI. vol. 17, pp. 3952–3958 (2017)