Privacy-CNH: A Framework to Detect Photo Privacy with Convolutional Neural Network Using Hierarchical Features

Lam Tran
University of Rochester
Department of Computer Science
Rochester, New York 14627
lam.tran@rochester.edu

Deguang Kong, Hongxia Jin
Samsung Research America
665 Clyde Ave
Mountain View, California 94043
doogkong@gmail.com hongxia@acm.org

Ji Liu
University of Rochester
Department of Computer Science
Rochester, New York 14627
jliu@cs.rochester.edu

Abstract

Photo privacy is a very important problem in the digital age where photos are commonly shared on social networking sites and mobile devices. The main challenge in photo privacy detection is how to generate discriminant features to accurately detect privacy at risk photos. Existing photo privacy detection works, which rely on low-level vision features, are non-informative to the users regarding what privacy information is leaked from their photos.

In this paper, we propose a new framework called ***Privacy-CNH*** that utilizes hierarchical features which include both object and convolutional features in a deep learning model to detect privacy at risk photos. The generation of object features enables our model to better inform the users about the reason why a photo has privacy risk. The combination of convolutional and object features provide a richer model to understand photo privacy from different aspects, thus improving photo privacy detection accuracy. Experimental results demonstrate that the proposed model outperforms the state-of-the-art work and the standard convolutional neural network (CNN) with convolutional features on photo privacy detection tasks.

Introduction

Mobile devices have revolutionized how people share photos with each other on social networks with a single click of a button. Many social networking websites allow the user to specify specific viewing privileges for specific groups of people. While social networking websites such as *Facebook* make an effort to protect their users' privacy, such websites mostly rely on the users themselves to report privacy violation before any action is taken. Moreover, the content of the photos is rarely analyzed by the websites before the photos are made available to view. After the photos are posted on the social network to the public to view, it is close to impossible to permanently delete the uploaded photos. As a result, the photos posted on these social networking websites may release users' home location, contact, bank account, family members, and other sensitive information to the world before the users notice them.

For example, the photo shown in Figure 1 was downloaded from *Flickr*. The photo contains an identification card

Figure 1: The photo of an identification card on *Flickr* that leaks the private information of the individual in the photo.

that reveals the private information such as name, age, address, date of birth and other information. This photo may be used by cybercriminals for fraudulent activities. The photo is privacy at risk because it may lead to identity theft.

Recently, photo privacy leakage is a major concern, and some of these incidents occurred without the awareness of the victim. The most recent event was the *iCloud* celebrity photo leaks (Villapaz 2004) where almost 500 private photos of several celebrities were leaked on the Internet. There are many other incidents that were not covered by the major media such as a man posted his ex-girlfriend's nude photos on *Facebook* as an act of revenge, and he was later convicted and sentenced to jail for 3 years in a California penitentiary (Rocha 2004).

Furthermore, a study by (Zerr et al. 2012) showed that more than 80% of photos shared by teenagers on *Facebook* were not approved by their parents and teachers. A recent research by (Besmer and Lipford 2008) had shown that *Facebook* profiles were used by employers and law enforcement for employment and crime investigation respectively. Even worse, a recent survey by (Wang et al. 2011) found that 23% of *Facebook* users regret posting on *Facebook*. Therefore, photo privacy detection is an important problem to solve in order to educate and protect users from the aforementioned

problems.

The question that naturally follows is *can we automatically detect the privacy risk of the images before being posted publicly?* Such information would be useful for the users to be aware about the potential risks of sharing their images with others. To achieve this goal, we need to extract useful features from the photos and build a model to detect privacy at risk photos. The major challenge in photo privacy detection is how to generate discriminant features to accurately detect privacy risk from the photo contents. Existing works on photo privacy detection, which rely on low-level vision features, are non-informative to the users regarding what privacy information is leaked from their photos.

In this work, we develop a framework, named **Privacy-CNH** (PCNH) that can automatically detect privacy at risk photos using the state-of-the-art deep learning approach. In particular, the PCNH trains a model that combines both convolutional and object features to better represent the privacy related features and improves detection performance. Our key contributions are summarized as follows:

- We formulate the photo privacy risk problem as a machine learning problem, which provides a generic framework to automatically and accurately understand the privacy risk of the photos by learning the hierarchical features extracted from the photos.
- Our framework adopts the state-of-the-art deep learning approach, which trains the CNN by combining both convolutional and object features, and leverages knowledge transferred from other datasets such as IMAGENET as well.
- We demonstrate that our proposed method is able to automatically detect the privacy at risk photos with the accuracy of 95% on the (Zerr et al. 2012) dataset and 90% on our designed dataset. The experimental results offer insight on understanding how privacy information is leaked from users' photos and the potential privacy risks.

The remainder of this paper is organized as follows: the Related Works section summarizes previous works in photo privacy and deep learning. The Problem Statement section states the problem that we addressed in this paper. The Data Set section describes the photo privacy data collection process. The Privacy-CNH Framework section shows the details of our deep learning architecture for photo privacy detection. The Experimentation and Discussion section reports the experimental results and discussion, and we conclude our paper and discuss future work in the Conclusion and Future Work section.

Related Works

Photo Privacy

Photo privacy is a very subjective topic; a person may have subjective opinion of whether a photo should be publicly or privately shared. However, most people generally agree that certain type of photos should not be shared with the general public such as the photos of driver license, legal document, pornographic photos, *etc*.

There are several methods of dealing with the problem of photo privacy detection and protection. Many existing works extract low-level image features such as color histogram and bag of visual words (BOVW) (Yang et al. 2007) features along with a face detector (Viola and Jones 2001) from the privacy dataset to develop a SVM (Burges 1998) classifier. For example, (Zerr et al. 2012) designed a game and recruited people to label the data from Flickr as public, private, or undecided for each photo and developed a SVM classifier with the low-level features. The work by (Squicciarini, Caragea, and Balakavi 2014) additionally conducted an extensive study with the (Zerr et al. 2012) dataset. The work (He et al. 2015) focused on privacy image sharing. Furthermore, the work by (Tan et al. 2014) focused on both detection and protection of photo privacy.

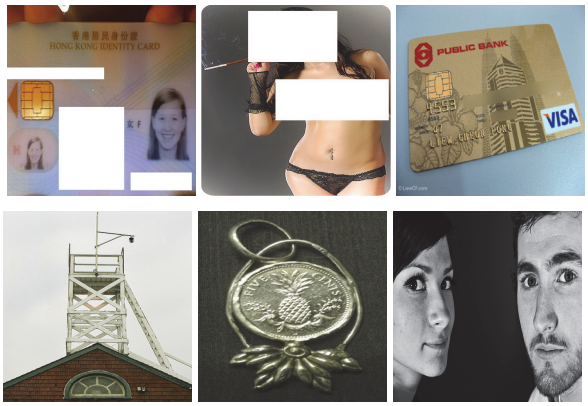

Figure 2: The figure shows photos in our dataset. **Top Row:** The photos are labeled by users as private. **Bottom Row:** The photos are labeled by users as public. We removed privacy sensitive data from the photos.

Similar to the work by (Zerr et al. 2012), the work by (Liu et al. 2011) also developed an user interface to use workers on Amazon Mechanical Turk (AMT) to label their dataset. Their user interface asked each worker to label each photo as share with "only me", "some friends", "all friends", "friends of friends", and "everyone". The labeled data from AMT is used to gauge the actual data on Facebook. They reported a big discrepancy between the actual data on Facebook and the data from the AMT workers. The discrepancy is mainly contributed to people on Facebook who use their default privacy setting during sign-up when they shared their photos on Facebook. As a result, the privacy setting of the users on Facebook is much lower than the desired settings from the data collected from the AMT workers.

Problem Statement

In this paper, we are interested in the problem of photo privacy detection from photos. We note that matrices are written as boldface uppercase, vectors are written as boldface lowercase, and scalars are denoted by lower-case letters. Let $y_n \in \{0, 1\}(1 \leq n \leq N)$ where 0 denotes as no public (no privacy risk) and 1 denotes as private (privacy risk) of a photo $\mathbf{x}_n$. We have a human-labeled dataset with N photos, $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ..., (\mathbf{x}_N, y_N)\}$ where $\mathbf{x}_n$ is an image instance and y_n is the privacy risk of image $\mathbf{x}_n$. Our

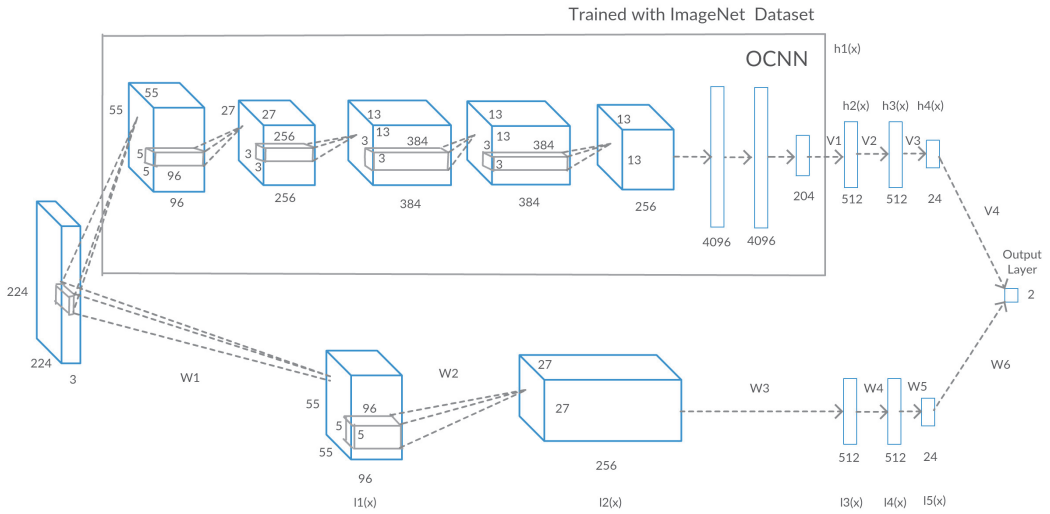

Figure 3: The proposed *"joint"* deep network architecture for privacy detection. It consists of two pipelines: (a) object features learning pipeline (b) convolutional features learning pipeline. Given the image features in the input layer, the object features learning pipeline processes the feature using $h_1(x), h_2(x), h_3(x), h_4(x)$ and the network structure is encoded as $\mathbf{V}^1, \mathbf{V}^2, \mathbf{V}^3, \mathbf{V}^4$, finally obtaining the photo privacy detection result in output layer; the convolutional features learning pipeline processes the feature using $\ell_1(x), \ell_2(x), \ell_3(x), \ell_4(x), \ell_5(x)$ and the network structure is encoded as $\mathbf{W}^1, \mathbf{W}^2, \mathbf{W}^3, \mathbf{W}^4, \mathbf{W}^5, \mathbf{W}^6$, finally obtaining the photo privacy detection result in output layer. The $h_i(x), \ell_j(x)(1 \leq i \leq 4, 1 \leq j \leq 5)$ are activation functions.

goal is to develop a classifier to accurately detect privacy risk of an unseen image sample $f : \mathbf{x} \rightarrow y$, where $\mathbf{x}$ denotes an unseen photo.

In order to apply machine learning to solve photo privacy risk detection problem, it is necessary to solve the following problems: **(1)** How do we extract and represent privacy related features from the photos? **(2)** How do we effectively learn the relationship between the privacy risk and the above privacy related features? **(3)** How do we build an automated classifier with high accuracy? Our work offers a framework that tackles these three problems in a *principled* way using deep learning framework, and we will offer the details of our solution in the following sections.

Data Set

Data collection for photo privacy is a challenging task. The main factor is due to the fact that photos shared on a public domain are usually shared with the general public, and private photos are limited. Secondly, photo privacy is subjective as discussed above, and it makes it challenging to automate the data collection process. To alleviate the latter problem, we use the dataset from (Zerr et al. 2012). This dataset was labeled by 81 people in a variety of professions between the age of 10 to 59. The participants were divided into six teams and the participants in each team were asked to label a set of photos as private, public, or undecided. There are a total of 37,535 photos in this dataset.

The dataset from (Zerr et al. 2012) is the only publicly available dataset for photo privacy research at this time. While many photo privacy researchers collect a small dataset for research; they are typically not able to share their data due to privacy concerns. In this work, we collected additional data to evaluate our algorithm. Our dataset consists of 3000 private photos of driver licenses/ID Cards, legal documents, pornographic photos, and group/family photos downloaded from Flickr. We use public photos in (Zerr et al. 2012) as public photos in our dataset. Figure 2 shows sample photos of our dataset and public photos from (Zerr et al. 2012) dataset.

Privacy-CNH Framework

In this paper, we design a PCNH deep learning framework to detect privacy at risk photos based on the CNN model.

Why Convolutional Neural Network?

The CNN pioneered by (LeCun et al. 1989) for optical character recognition is a workhorse for many computer vision classification and detection tasks. Furthermore, (Krizhevsky, Sutskever, and Hinton 2012) demonstrated on large scale object classification with CNN successfully on GPUs, and it has renewed interests in CNN from the computer vision community. In recent years, there has been an ample amount of deep learning papers in various areas such as face detection (Sun, Wang, and Tang 2013), object detection (Girshick et al. 2013), pedestrian detection (Sermanet et al. 2013), human posture detection (Toshev and Szegedy 2013) and many other areas. Moreover, CNN is able to learn filters without the need of hand-engineering features in traditional machine learning algorithms. Thus in this work, we propose to use a CNN model to solve the photo privacy detection problem.

In comparison to previous works, we aim to develop a photo privacy detection algorithm that leverages the usage

Algorithm 1 Privacy-CNH SVM Model

Input:
$\mathbf{X} \leftarrow \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ be N image of size 256×256.
$\mathbf{Y} \leftarrow \{y_1, y_2, \ldots, y_N\}$ be N corresponding labels
Randomly partition the training data set $(\mathbf{X}, \mathbf{Y})$ into 10 folds with equal size $(\mathbf{X}_i, \mathbf{Y}_i)$ $i = 1, \cdots, 10$.

Train OCNN
for $i \leftarrow 1$ to 10 **do**
    Train PCNN and PONN on ImageNet with OCNN fixed.
    Fine-tune model with Privacy Data on set $(\mathbf{X}', \mathbf{Y}') \leftarrow (\mathbf{X}, \mathbf{Y}) \setminus (\mathbf{X}_i, \mathbf{Y}_i)$.
    Remove the output layer of PCNH.
    Extract feature $u(\mathbf{X}') = [\hat{l}_4(\mathbf{X}'), \hat{h}_5(\mathbf{X}')]$
    Train SVM model with $u(\mathbf{X}')$
    Test on $(\mathbf{X}_i, \mathbf{Y}_i)$.
return: The averaged performance and all parameters $\mathbf{w}_s, \beta$.

Algorithm 2 Inference Privacy Risk Photo

Input: Input image $\mathbf{x}$

Compute $u(\mathbf{x}) = [\hat{l}_5(\mathbf{x}), \hat{h}_4(\mathbf{x})]$.
Predicts $\mathbf{x}$ with SVM model and save in y.
if y is privacy risk **then**
    Find object of $\mathbf{x}$ in OCNN.
    Alert User and Display Class o

of both convolutional and object features. We utilize deep learning and construct two convolutional CNNs, one to learn convolutional features and the other one to learn object features. By separating the CNNs, it allows us to parallelize the computation on two GPUs without significantly increasing computational time while improving performance. Furthermore, the benefit of using object features allows us to develop an algorithm that is more intuitive to inform users about privacy at risk photos before they post their photos on social networking sites. To the best of our knowledge, we are the *first* to apply deep learning to photo privacy with convolutional and object features.

Challenge

The main challenge in training a CNN with millions of parameters for photo privacy detection is that only a small number of training photos for privacy detection is available. However, CNN generally requires a large number of training data in order to achieve better performance. To overcome this limitation, we use dataset from a related domain (*a.k.a* transfer learning) to train the model. The network structure weights are fine tuned after a number of training steps are performed.

Towards this goal, we propose a deep network architecture composed of two disjoint CNNs and transfer learning from ImageNet (Russakovsky et al. 2014). The diagram of PCNH is shown in Figure 3. The privacy object neural network (PONN) corresponds to the object features (upper pipeline shown in Figure 3) and privacy convolutional neural network (PCNN) corresponds to convolutional features (lower line shown in Figure 3). The two CNNs are connected at the output layer for computational efficiency.

Detailed Design

Given the image features in the input layer, the object features learning pipeline processes the features using $h_i(x)(1 \leq i \leq 4)$ as the activation functions and the parameters of network structure is encoded as $\mathbf{V}^i(1 \leq i \leq 4)$, finally obtaining the photo privacy detection result in the output layer. The convolutional features learning pipeline processes the features using $\ell_j(x)(1 \leq j \leq 5)$ as the activation function and the parameters of network structure is encoded as $\mathbf{W}^s(1 \leq s \leq 6)$, finally obtaining the photo privacy detection result in the output layer. $h_i(x), \ell_j(x)(1 \leq i \leq 4, 1 \leq j \leq 5)$ are activation functions, which map from a vector to a scalar.

The PCNN in our network is used to extract the convolutional features. The first layer has 96 kernels with the size of 11×11 and stride of 4. The second layer has 256 kernels with the size of 5×5 and stride of 2. The final 3 layers are fully connected with 2 layers of 512 neurons, one layer of 24 neurons, and 2 neurons for the output layer. The last layer with 24 neurons in our CNN is inspired by (You et al. 2015) to capture the 24 human emotions described by (Plutchik 1980).

The first part of the PONN is referred to as the object CNN (OCNN) in our network. The OCNN is modeled in a similar fashion as (Krizhevsky, Sutskever, and Hinton 2012) since they showed the state-of-the-art result for large scale object classification. The OCNN is trained with 204 object classes from (Russakovsky et al. 2014) and 4 additional classes we collected to extract object-level features. The transfer learning is inspired by (Oquab et al. 2014) where they showed how to transfer middle-level representation features by replacing the output layer with another output layer of their problem domain. However, in photo privacy detection, it is important that we keep the OCNN as object classifier and not adapt to privacy detection domain. This leads us to pre-train the OCNN and fixes it in the PCNH learning phase. The output of the OCNN contains essential information about privacy risk to the user.

Finally, the output layer of the OCNN is collected to 3 fully connected layers. Similar to the PCNN, we use 24 neurons in the last layer. During training in PCNH, we only back propagate to the last 3 fully connected layers in the PONN. The parameters of the pre-trained OCNN remain fixed in the learning phase of PCNH. The detail of our algorithm is shown in Algorithm 1.

Note in this design, the PONN is added to replace the photo tag features since photo tag is most likely not available on the user's device. The user may be reluctant to annotate the image when using this app on the phone.

Model Specification and Inference

Let $y_n = 1$ denote as privacy risk and $y_n = 0$ denote as no privacy risk for image $\mathbf{x}_n$. Given the joint CNN structure

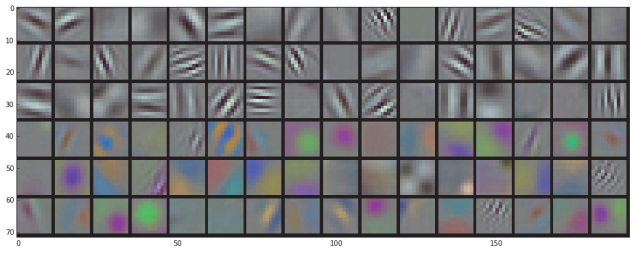

Figure 4: The filters learned from the PCNN with photo privacy dataset.

$\mathcal{V} = \{\mathbf{V}^1, \cdots, \mathbf{V}^4\}$, $\mathcal{W} = \{\mathbf{W}^1, \cdots, \mathbf{W}^5\}$, the posterior probability of privacy risk of image $\mathbf{x}_n$ is:

$$\mathrm{P}(y_n = 1|\mathbf{x}_n; \mathcal{V}, \mathcal{W}) = \frac{1}{1 + \exp(-z)} \tag{1}$$

where

$$z = \left(\mathbf{V}_k^4\right)^\intercal h_4(\mathbf{x}_n) + \left(\mathbf{W}_l^6\right)^\intercal \ell_5(\mathbf{x}_n) + \beta \tag{2}$$

where $\mathbf{V}^i$ and $\mathbf{W}^j$ are the parameters matrices with the superscript i, j indicating the layer in the CNN, k indexes the hidden unit in the layer i, l indexes the hidden unit in layer j, h_i and ℓ_j are the activation functions for the PONN and PCNN respectively, and β is the biased scalar term.

We first train our algorithm using ImageNet and fine-tune the PCNH with the privacy dataset. After the PCNH trained with gradient decent approach, we use PCNH as features extractor by removing the output layer of PCNH. The learned $h_4(x) \in \mathbb{R}^{24}$ and $l_5(x) \in \mathbb{R}^{24}$ are features in the privacy classification model. Let $u(\mathbf{x}_n)^\intercal = [\hat{h}_4(\mathbf{x}_n), \hat{l}_5(\mathbf{x}_n)]^\intercal \in [0, 1]^{48}$, where $\hat{h}_4(\mathbf{x}_n)$ and $\hat{l}_5(\mathbf{x}_n)$ are the normalized vector (softmax) of $h_4(\mathbf{x}_n)$ and $l_5(\mathbf{x}_n)$ respectively. We then fit a SVM model as follows:

$$\max_{\mathbf{w}_s,} \frac{\lambda}{2}\|\mathbf{w}_s\|^2 - \sum_{n=1}^{N} \alpha_n(y_n(\mathbf{w}_s^\intercal u(\mathbf{x}_n) + \beta) - 1) \tag{3}$$

where λ is a regularization parameter controlling model complexity, $\mathbf{w}_s$ and β are the parameters of the model, and $\alpha_1, \ldots, \alpha_n$ are the Lagrange multipliers. Algorithm 1 summarizes the learning algorithm.

We developed an algorithm to infer privacy at risk photos with the learned model from Algorithm 1. The algorithm takes a photo and alerts the user when privacy risk is detected and displays the object class with the highest probability in the OCNN which is a component of the PONN. The algorithm gives high level information to the user about what privacy risk information is leaked in the photo. The detail of this algorithm is shown in Algorithm 2.

Experimentation and Discussion

In the experimental setup, we used the photo privacy dataset from (Zerr et al. 2012) and the dataset we collected. Each dataset is partitioned into 10-folds with random permutation. The (Zerr et al. 2012) dataset contains 400 private and 400 public photos in each partition. In our dataset, we use

Figure 5: The photos above are predicted as private by PCNH. The photos in the top row are predicted correctly as private photos. The photos in the bottom row are predicted as privacy but the true labels are public. When a photo is predicted as private, the algorithm shows the top class of the OCNN to warn the user what kind of information is leaked.

300 private and 300 public photos in each partition. We train each algorithm on 9 sets of data and test on the remaining 1 set. We run a total of 10 trials and alternate the test set in each trail and report the averaged performance for each algorithm. All of the algorithms are run on a Linux X86 64 bits machine with 16G RAM and two GTX 780Ti GPUs.

Baseline Methods

We compare our PCNH with several baseline methods including SVM, PCNN, and PONN. We also run a second experiment with transfer learning on the same methods. For the CNN models, we implemented our algorithm with cuda-convnet2 by (Krizhevsky, Sutskever, and Hinton 2012). We prefer cuda-convnet2 over CAFFE (Jia et al. 2014) because cuda-convnet2 supports multiple GPUs implementation.

SVM with BOVW: The work of (Zerr et al. 2012) and (Squicciarini, Caragea, and Balakavi 2014) both explored SIFT descriptors by (Lowe 2004) to extract features from images and learned a visual word dictionary. The photos are encoded with a dictionary using BOVW approach and trained with SVM model for photo privacy detection.

Privacy CNN (PCNN): In this paper, we refer to the CNN by (Krizhevsky, Sutskever, and Hinton 2012) as PCNN. However, we modified the architecture to be more suited for privacy photo detection. PCNNs have been shown to achieve the state-of-the-art performance on many image classification tasks especially when the data is not linearly separable. PCNN typically starts with convolutional layers to extract low-level features such as edges and texture from an image. The extracted features are fed forward to more convolutional layers and finally to a set of fully connected layers. To prevent over-fitting in practice, a dropout rate with a probability of 0.5 is applied to each edge.

Privacy Object NN (PONN): PONN in this paper refers to the usage of object class features as the input to a deep network. PONN is useful for image classification tasks involving object information. Our motivation is to extract object class features to better inform user about privacy risk. The OCNN is pre-trained with 204 object classes. The out-

	(Zerr et al. 2012) Dataset				Our Dataset			
Method	Precision	Recall	F1	Accuracy	Precision	Recall	F1	Accuracy
SVM + BOVW	0.65	0.40	0.50	0.65	0.62	0.38	0.47	0.61
PONN	0.68	0.60	0.64	0.67	0.66	0.57	0.61	0.65
PCNN	0.72	0.58	0.64	0.74	0.70	0.55	0.61	0.72
PCNH	0.83	0.63	0.72	0.83	0.77	0.60	0.67	0.80
PONN + Transfer Learning	0.74	0.65	0.69	0.71	0.70	0.62	0.65	0.70
PCNN + Transfer Learning	0.89	0.80	0.84	0.89	0.85	0.77	0.80	0.84
PCNH + Transfer Learning	**0.94**	**0.85**	**0.89**	**0.95**	**0.90**	**0.80**	**0.84**	**0.90**

Table 1: The table above shows experiment results for the two privacy datasets of different algorithms using 10 fold cross validation.

	ID Card	Document	Nude	Group
Top 1	0.61	0.65	0.62	0.59
Top 5	0.74	0.79	0.77	0.75

Table 2: Top k ($k = 1, 5$) OCNN object classification accuracy for the 4 privacy classes.

put layer of the OCNN is connected to 3 fully connected layers. The detail of PONN is in the Privacy-CNH Framework section. We selected three fully connected layers to allow our model to learn more meaningful hierarchical features of the object classes.

Privacy-CNH (PCNH): The combination of PCNN and PONN is PCNH. This architecture leverages the usage of object and convolutional features to improve photo privacy detection accuracy. Our motivation in designing PCNH is due to the different type of human photos in the private and public photo sets. For example, nude photos should be private and the business portrait photos should be public. By combining PCNN and PONN, it allows the proposed framework to better differentiate between human photos that belong to private and public sets. This is accomplished by using PONN to detect object classes such as people, as well as using PCNN to detect the attribute of the photo with convolutional features.

Transfer Learning: In transfer learning, we first trained each model with ImageNet and fine tuned the model with the privacy data. In each of the model above, the deep networks were only used for features extraction. Then the features were extracted by removing the last output layer and were fitted into a SVM model.

Discussion

The performances of all algorithms are shown in Table 1 for both datasets and Figure 4 shows that the filter learned from the PCNN. The result indicates that SVM with BOVW achieved the lowest accuracy because the photo privacy detection problem is non-linearly separable. PONN achieved better performance than SVM + BOVW by more than 2%. We notice that more human photos are labeled as private. Since person is one of the 204 object classes, PONN mostly predicts human photos as private. PCNN achieved the second best performance and PCNH achieved the best performance in all four metrics compared to all of the other algorithms. Combining the two architectures (PCNN and PONN) into a new architecture gives us the benefit of both architectures. The experimental results suggest that PCNH are more suitable for photo privacy detection than baseline methods.

The results also showed that by first training on ImageNet dataset then fine tuning each model with the privacy dataset improved the accuracy dramatically. This is due to the insufficient amount of data to train each deep network model with only the privacy data. We observed that the accuracy improved between 4% to 15% with transfer learning. Figure 5 shows some correctly and incorrectly predicted photos using PCNH. Our model also displays the top class from the OCNN in each photo to warn the user about what kind of privacy data is leaked. Table 2 shows the accuracy of the class prediction by OCNN. The top k accuracy is the prediction of a class as one of the top k classes by OCNN.

Conclusion and Future work

Photo privacy detection is a challenging problem due to its inherent subjectivity. In this paper, we propose a new CNN model called PCNH that utilizes both convolutional and object features for photo privacy detection. The proposed model achieves higher photo privacy detection accuracy compared to previous works, including CNN with only convolutional features. Additionally, there are many advantages of using object features such as the object classes to improve the accuracy of photo privacy detection. The object classes can help inform the users of the nature of the privacy risk in their photos. It also helps to avoid the trouble of asking the user to annotate their images with text.

In future work, we plan to consider the localization to detect objects in the OCNN and train a model of multiple objects interaction to improve the privacy photo detection. Furthermore, we plan to conduct case studies related to privacy information leakage from photos. We envision that this work can help reduce privacy concerns by users of on-line social networking sites in the digital age.

Acknowledgements

This work was done during Lam Tran's internship mentored by Deguang Kong at Samsung Research America, Silicon Valley. Ji Liu is supported by the NSF grant CNS-1548078, the NEC fellowship, and the startup funding at University

of Rochester. Lam Tran is supported by NSF Graduate Research Fellowship DGE-1419118 ID 2011085425.

References

Besmer, A., and Lipford, H. R. 2008. Privacy perceptions of photo sharing in facebook. In *In Proc. SOUPS 2008, ACM Press*.

Burges, C. J. C. 1998. A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.* 2(2):121–167.

Girshick, R. B.; Donahue, J.; Darrell, T.; and Malik, J. 2013. Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR* abs/1311.2524.

He, J.; Liu, B.; Kong, D.; Bao, X.; Wang, N.; Jin, H.; and Kesidis, G. 2015. Puppies: Transformation-supported personalized privacy preserving partial image sharing. In *The Pennsylvania State University Technical Report*, CSE–2015–007.

Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; and Darrell, T. 2014. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In Pereira, F.; Burges, C. J. C.; Bottou, L.; and Weinberger, K. Q., eds., *Advances in Neural Information Processing Systems 25*. Curran Associates, Inc. 1097–1105.

LeCun, Y.; Boser, B.; Denker, J. S.; Henderson, D.; Howard, R. E.; Hubbard, W.; and Jackel, L. D. 1989. Backpropagation applied to handwritten zip code recognition. *Neural Comput.* 1(4):541–551.

Liu, Y.; Gummadi, K. P.; Krishnamurthy, B.; and Mislove, A. 2011. Analyzing facebook privacy settings: User expectations vs. reality. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, 61–70. New York, NY, USA: ACM.

Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* 60(2):91–110.

Oquab, M.; Bottou, L.; Laptev, I.; and Sivic, J. 2014. Learning and transferring mid-level image representations using convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, 1717–1724.

Plutchik, R. 1980. *A general psychoevolutionary theory of emotion*. New York: Academic press. 3–33.

Rocha, V. 2004. Revenge porn conviction is a first under california law. *Los Angeles Times* 91–110.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A. C.; and Fei-Fei, L. 2014. ImageNet Large Scale Visual Recognition Challenge.

Sermanet, P.; Kavukcuoglu, K.; Chintala, S.; and LeCun, Y. 2013. Pedestrian detection with unsupervised multi-stage feature learning. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR'13)*. IEEE.

Squicciarini, A. C.; Caragea, C.; and Balakavi, R. 2014. Analyzing images' privacy for the modern web. In *Proceedings of the 25th ACM Conference on Hypertext and Social Media*, HT '14, 136–147. New York, NY, USA: ACM.

Sun, Y.; Wang, X.; and Tang, X. 2013. Deep convolutional network cascade for facial point detection. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '13, 3476–3483. Washington, DC, USA: IEEE Computer Society.

Tan, J.; Drolia, U.; Martins, R.; Gandhi, R.; and Narasimhan, P. 2014. Short paper: Chips: Content-based heuristics for improving photo privacy for smartphones. In *Proceedings of the 2014 ACM Conference on Security and Privacy in Wireless & Mobile Networks*, WiSec '14, 213–218. New York, NY, USA: ACM.

Toshev, A., and Szegedy, C. 2013. Deeppose: Human pose estimation via deep neural networks. *CoRR* abs/1312.4659.

Villapaz, L. 2004. Apple investigating possible icloud involvement in nude celebrity photo leaks. *International Business Machines* 91–110.

Viola, P., and Jones, M. 2001. Robust real-time object detection. In *International Journal of Computer Vision*.

Wang, Y.; Norcie, G.; Komanduri, S.; Acquisti, A.; Leon, P. G.; and Cranor, L. F. 2011. "i regretted the minute i pressed share": A qualitative study of regrets on facebook. In *Proceedings of the Seventh Symposium on Usable Privacy and Security*, SOUPS '11, 10:1–10:16. New York, NY, USA: ACM.

Yang, J.; Jiang, Y.-G.; Hauptmann, A. G.; and Ngo, C.-W. 2007. Evaluating bag-of-visual-words representations in scene classification. In *Proceedings of the International Workshop on Workshop on Multimedia Information Retrieval*, MIR '07, 197–206. New York, NY, USA: ACM.

You, Q.; Luo, J.; Jin, H.; and Yang, J. 2015. Robust image sentiment analysis using progressively trained and domain transferred deep networks. In *Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI)*.

Zerr, S.; Siersdorfer, S.; Hare, J.; and Demidova, E. 2012. Privacy-aware image classification and search. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, 35–44. New York, NY, USA: ACM.