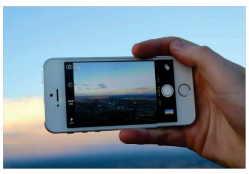
プライバシー保護一人称ビジョン

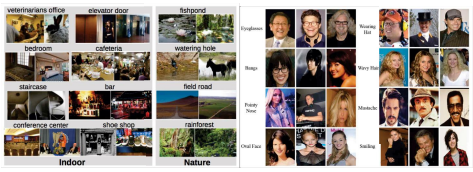
～安全な画像認識の基盤技術～

米谷　竜
東京大学生産技術研究所

コンピュータビジョン＝「見え方の科学」

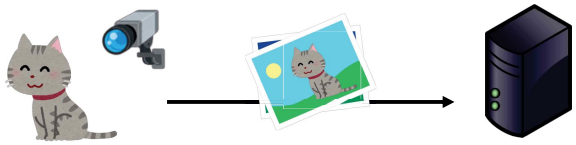
- 計算機に外界を視覚的に理解させたい
- 画像認識: 画像に映る外界の内容を判断できるか？





統計的機械学習に基づく画像認識

- 判断のためのモデルを**大量の事例（画像＋判断）**から学習



データを撮影　計算機にアップロード　学習

高性能な計算機

とはいえ撮影・提供しにくいデータも存在する

- 例: ウェアラブルカメラで撮影されたライフログ映像
  -> 人々のプライベートな瞬間が含まれるため

どこにいたか
誰と会ったか
Etc...

解決したい問題

画像認識技術が社会的に受容される将来を切り拓くため，ウェアラブルカメラ映像等センシティブなデータを，安心して扱えるコンピュータビジョンの基盤技術を開発したい．

安全な画像認識を目指して: その1

- 画像から抽出された**特徴量**のみを計算機に提供する？

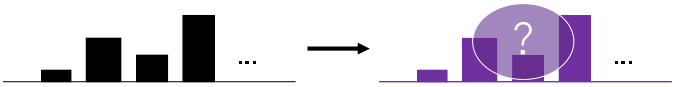
特徴量:
その画像を表す数値列

→不十分！特徴量から元画像をある程度復元可能

左: 元画像
右: 特徴量からの復元結果

安全な画像認識を目指して: その2

- 特徴量を**暗号化**して提供
- 暗号化したまま計算が可能な**準同型暗号**の利用



今回の成果

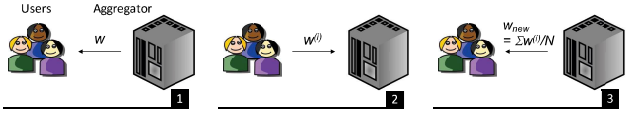
ある条件下において，データ提供者の持つプライベートな画像データの内容が分からないようにしつつ，計算機がその画像データを用いて統計的機械学習を行える枠組みを提案し，いくつかの実験でその有効性を確認しました．

Privacy-Preserving Visual Learning with Doubly-Permuted Homomorphic Encryption

Ryo Yonetani, Vishnu Naresh Boddeti, Kris M. Kitani, and Yoichi Sato, International Conference on Computer Vision (ICCV), 2017

Introduction: Distributed Supervised Learning

1. Aggregator distributes classifier weight w
2. Users update w using own data and send $w^{(i)}$ back to the aggregator
3. Aggregator computes $w_{new} = \frac{1}{N}\sum_i w^{(i)}$ and distributes it again



Homomorphic Encryption for Distributed Learning

Using Paillier cryptosystem [Paillier, '99] for secure aggregation of classifiers

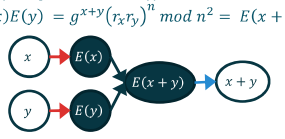
Encryption (pub-key $n = pq, g = n + 1$)
r is a random variable generated for each encryption

$E(x) = g^x r^n \bmod n^2$

Decryption (priv-key p, q)
E(x) cannot be inverted to x without both p and q

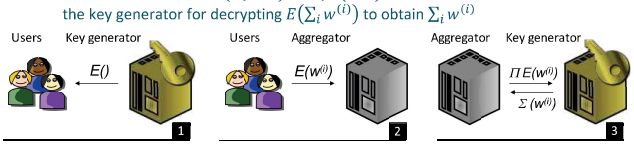
Computing sum over encrypted data:

$E(x)E(y) = g^{x+y}(r_x r_y)^n \bmod n^2 = E(x+y)$



Homomorphically-encrypted distributed learning protocol

1. Key generator issues and distributes a pub key to all parties
2. Users encrypt $w^{(i)}$ and send $E(w^{(i)})$ to the aggregator
3. Aggregator computes $E(\sum_i w^{(i)}) = \prod_i E(w^{(i)})$ and sends it to the key generator for decrypting $E(\sum_i w^{(i)})$ to obtain $\sum_i w^{(i)}$



Pros: $w_{new} = \frac{1}{N}\sum_i w^{(i)}$ can be computed without knowing each plain $w^{(i)}$
Cons: Encryption time (e.g., 10 minutes* to encrypt 200k weights)

* Using Python+GMP. More efficient implementations are available at https://medium.com/snips-ai/benchmarking-paillier-encryption-15631a0b5ad8

Proposed: Doubly-Permuted Homomorphic Encryption (DPHE)

A new homomorphic encryption scheme for sparse high-dimensional data

Input: sparse high-dim data
Updating classifiers with sparse constraints

$w^{(i)}$ [1 0 2 0 0]

90% sparse classifier
= 10x faster encryption!

Encrypted non-zero weights $E(v^{(i)})$
Using Paillier cryptosystem

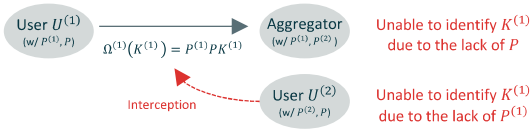
Shuffled non-zero indices $\Omega^{(i)}(K^{(i)})$
Using double permutation

$\Omega^{(i)}(K^{(i)}) = P^{(i)}PK^{(i)}$

$\Omega^{(i)}(E(w^{(i)}))$ can be reconstructed at the aggregator side

Two permutation matrices for encrypting non-zero indices

$P^{(i)}$: shared between **user $U^{(i)}$ and the aggregator** | P: shared between **all users but the aggregator**



Unable to identify $K^{(1)}$ due to the lack of P

Unable to identify $K^{(1)}$ due to the lack of $P^{(1)}$

Computing encrypted sum:
(see our paper for the detail)

$PE(w^{(i)}) = (P^{(i)})^{-1}\Omega^{(i)}(E(w^{(i)}))$

$PE\left(\sum_i w^{(i)}\right) = \prod_i PE(w^{(i)})$

Experiments

Implementation details

- Classifier was updated via SGD with the elastic net regularization
- Feature were extracted from pre-trained deep nets (e.g., ResNet trained on ImageNet)

Performance evaluation

- DHPE performed comparably to SoTA methods (HZRS14, ZF13, LLWT15) while preserving privacy and outperformed existing privacy-preserving methods (PRR10, RA12)

Classification accuracy (# users = 5)

Methods	Caltech101	Caltech256	Privacy
HZRS14	93.4 ± 0.5	N/A	NO
ZF13	85.4 ± 0.4	72.6 ± 0.1	NO
PRR10	41.6 ± 1.2	55.9 ± 0.5	YES
RA12	83.8 ± 1.1	68.0 ± 0.3	YES
DPHE (Ours)	89.3 ± 0.8	74.7 ± 0.4	YES

Encryption time

Sparsity	Accuracy	Time (sec)
0.01	89.7	620
95.6	88.2	62

Number of users

# Users	Accuracy
10	88.9
100	85.6