PrivateMail: Supervised Manifold Learning of Deep Features With Privacy for Image Retrieval

Praneeth Vepakomma, Julia Balla, Ramesh Raskar

Massachusetts Institute of Technology
vepakom@mit.edu

Abstract

Differential Privacy offers strong guarantees such as immutable privacy under any post-processing. In this work, we propose a differentially private mechanism called PrivateMail for performing supervised manifold learning. We then apply it to the use case of private image retrieval to obtain nearest matches to a client's target image from a server's database. PrivateMail releases the target image as part of a differentially private manifold embedding. We give bounds on the global sensitivity of the manifold learning map in order to obfuscate and release embeddings with differential privacy inducing noise. We show that PrivateMail obtains a substantially better performance in terms of the privacy-utility trade off in comparison to several baselines on various datasets. We share code for applying PrivateMail at http://tiny.cc/PrivateMail.

1 Introduction

Privacy preserving computation enables distributed hosts with 'siloed' away data to query, analyse or model their sensitive data and share findings in a privacy preserving manner. As a motivating problem, in this paper we focus on the task of privately retrieving nearest matches to a client's target image with respect to a server's database of images. Consider the setting where a client would like to obtain the k-nearest matches to its target from an external distributed database. State of the art image retrieval machine learning models such as (Matsui, Yamaguchi, and Wang 2020; Chen et al. 2021; Zhou, Li, and Tian 2017; Dubey 2020) exist for feature extraction pior to obtaining the neighbors to a given match in the learnt space of deep feature representations. Unfortunately, this approach is not private. The goal of our approach is to be able to use these useful features for the purpose of image retrieval in a manner, that is formally differentially private. The seminal idea for a mathematical notion of privacy, called differential privacy, along with its foundations is introduced quite well in (Dwork, Roth et al. 2014). In our approach, we geometrically embed the image features via a supervised manifold learning query that we propose. Our query falls within the framework of supervised manifold learning as formalized in (Vural and Guillemot 2017). We then propose a differentially private mechanism to release the outputs of this query. The privatized outputs of this query are used to perform the matching and retrieval of the nearest neighbors in this privatized feature space. Differential privacy aims to prevent membership inference attacks (Shokri et al. 2017; Truex et al. 2018; Li and Zhang 2020; Song, Shokri, and Mittal 2019; Shi, Davaslioglu, and Sagduyu 2020). It has been shown that differential privacy mechanisms can also prevent reconstruction attacks under a constraint on the level of utility that can be achieved as shown in (Dwork et al. 2017; Garfinkel, Abowd, and Martindale 2018). Currently cryptographic methods for the problem of information retrieval were studied in works like (Xia et al. 2015). These methods ensure to protect the client's data via homomorphic encryption and oblivious transfer. However, they also come with an impractical trade-off of computational scalability, especially when the size of the server's database is large and the feature size is high-dimensional as is always the case in practice (Elmehdwi, Samanthula, and Jiang 2014; Lei et al. 2019; Yao, Li, and Xiao 2013).

Motivation

1. Currently available differential privacy solutions for biometric applications where content based matching of records is performed (Steil et al. 2019; Chamikara et al. 2020) is based on a small number of hand-crafted features. We instead consider state of the art feature extraction used by recent deep learning architectures specialized for image retrieval such as (Jun et al. 2019). We privatize these features and share them in the form of differentially private embeddings that are in turn used for the image retrieval task.
2. Cryptographic methods with strong security guarantees are currently not scalable computationally, for secure k-nn queries (Elmehdwi, Samanthula, and Jiang 2014; Lei et al. 2019; Yao, Li, and Xiao 2013) especially when the server-side database is large as is typically the case in real-life scenarios.

2 Contributions

1. The main contibution of our paper is a differentially private method called *PrivateMail* for private release of outputs from a supervised manifold learning query that embeds data into a lower dimension. We test our scheme for differentially private 'content based image retrieval', where the matches to a target image requested by a client are retrieved from a server's database while maintaining differential privacy.
2. We show a substantial improvement in the utility-privacy

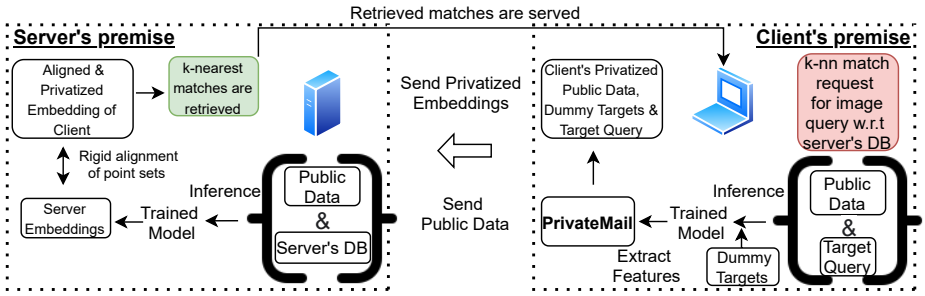

Figure 1: This illustration shows the lifecycle of interactions between client and server side entities for private image retrieval. The interaction starts from the red bubble on the right. At first, the client and server train a good on-premise machine learning model that is tailored for image retrieval. The client extracts features from this model on the target query image, dummy targets that are only known to the client as well as a public dataset known to the client and server. The extracted features go through the proposed Private-Mail for embedding them via locally differentially private supervised manifold learning. These private embeddings are aligned at server prior to performing the nearest-neighbor retrieval of matches that are served back to the client. The privatized representation of public dataset is used as an anchor in order to align the feature embeddings between the client and server.

trade-off of our embeddings over 5 existing baselines.

3. The supervised manifold learning query that we propose to geometrically embed features extracted from deep networks is novel in itself. That said, we would only consider this as a secondary contribution to this paper.

3 Related work

Non-private image search and retrieval: Current state of the art pipelines for content based image retrieval under the non-private setting are fairly matured and based on nearest neighbor queries performed over specialized deep feature representations of these images. The query image and the database of images are compared in this learnt representation space. A detailed set of tutorials and surveys on this problem in the non-private setting is provided in (Matsui, Yamaguchi, and Wang 2020; Chen et al. 2021; Zhou, Li, and Tian 2017; Dubey 2020).

Private manifold learning: There have been recent developments in learning private geometric embeddings with differentially private unsupervised manifold learning. Notable examples include distributed and differentially private version of t-SNE (Van der Maaten and Hinton 2008) called DP-dSNE (Saha et al. 2020, 2021) and (Arora and Upadhyay 2019) for differentially private Laplacian Eigenmaps (Belkin and Niyogi 2003, 2007). Furthermore, the work in (Choromanska et al. 2016) provides a method for differentially private random projection trees to perform unsupervised private manifold learning. The work in (Upadhyay 2014) also studies Riemannian manifold learning with differential privacy for manifolds with a bounded condition number and geodesic covering regularity. However, none of these works consider differentially private manifold learning in the supervised setting that we explore in this paper. We show a substantial improvement in privacy-utility trade-offs of the supervised manifold embedding approach over existing baselines that include private and non-private methods in the supervised and unsupervised paradigms.

4 Approach

Motivated by the supervised manifold learning framework in (Vural and Guillemot 2017) that is based on a difference of two unsupervised manifold learning objectives, we present an iterative update to efficiently optimize it. We refer to this iterative optimization as the *supervised manifold learning query (SMLQ)*. We then provide a privacy mechanism called *PrivateMail* to perform this supervised manifold learning query with a guarantee of differential privacy. To do that, we derive the sensitivity of our query that is required to calibrate the amount of noise needed to attain differential privacy. As part of experimental results, we apply our approach to a novel task of differentially private image retrieval, that has not been well-studied in current literature as opposed to the non-private image retrieval task which is a widely studied problem.

Notation	Description
n	Sample size
d	Data dimension
k	Embedded dimension
$\mathbf{X}_{n\times d}$	Data matrix
$\mathbf{Y}_{n\times 1}$	Labels
f	Manifold learning map
σ	Gaussian kernel bandwidth
σ_q	std. dev. of entries in $\mathbf{Q}$
α	regularization in $\mathbf{L_X} - \alpha\mathbf{L_Y}$
$\mathbf{Q}$	$Q_{i,j} \sim N(0, \sigma_q^2)$

Table 1: Notations

5 Moving from unsupervised to supervised manifold learning

We first briefly introduce some preliminaries for unsupervised manifold learning in order to build upon it to introduce supervised manifold learning.

Preliminaries for unsupervised manifold learning

This problem is a discrete analogue of the continuous problem of learning a map $f : \mathcal{M} \mapsto \mathbb{R}^k$ from a smooth, com-

pact high dimensional Riemannian manifold such that for any two points x_1, x_2 on $\mathcal{M}$, the geodesic distance on the manifold $d_{\mathcal{M}}(x_1, x_2)$ is approximated by the Euclidean distance $\|f(x_1) - f(x_2)\|$ in $\mathbb{R}^k$. Different manifold learning techniques vary in their tightness of this approximation on varying datasets. Manifold learning techniques like Laplacian Eigenmaps (Belkin and Niyogi 2005), Diffusion Maps (Coifman and Lafon 2006) and Hessian Eigenmaps (Donoho and Grimes 2003) aim to find a tighter approximation by trying to minimize a relevant bounding quantity $\mathbf{B}$ such that $\|f(x_1) - f(x_2)\| \leq \mathbf{B} \cdot d_{\mathcal{M}}(x_1, x_2) + o(d_{\mathcal{M}}(x_1, x_2))$. Different techniques propose different possiblilities for such a $\mathbf{B}$. For example, Laplacian Eigenmaps uses $\mathbf{B} = ||\nabla f(x_1)||$ for which it is shown that this relation holds as

$$||f(x_1) - f(x_2)|| \leq ||\nabla f(x_1)|| \cdot ||x_1 - x_2|| + o(||x_1 - x_2||)$$

Hence, controlling $||\nabla f||_{L^2(\mathcal{M})}$ preserves geodesic relations on the manifold in the Euclidean space after the embedding.

From continuous to discrete

This quantity of $||\nabla f||_{L^2(\mathcal{M})}$ in the continuous domain can be optimized via chosing the eigenfunctions of the Laplace-Beltrami operator in order to get the optimal embedding. This is explained in a series of papers by (Giné, Koltchinskii et al. 2006; Belkin and Niyogi 2007; Jones, Maggioni, and Schul 2008). From a computational standpoint we note that, for a specific graph defined on all pairs of data points with an adjacency matrix $\mathbf{W_X}$ and corresponding graph Laplacian $\mathbf{L_X}$, the following quantity

$$\Sigma_{i,j}(||\mathbf{f(X_i)} - \mathbf{f(X_j)}||^2 \cdot [\mathbf{W_X}]_{ij}) = \mathrm{Tr}(\mathbf{f(X)}^T \mathbf{L_X f(X)}) \tag{1}$$

is the discrete version of $||\nabla f||^2_{L^2(\mathcal{M})}$ under the assumption that the dataset $\mathbf{X}$ is a sample lying on the manifold $\mathcal{M}$. Here, $\mathbf{f(X_i)}$ and $\mathbf{f(X_j)}$ refer to the k dimensional real-valued output of the manifold learning map f at two single points represented by i and j rows in the data matrix $\mathbf{X}_{n \times d}$. Similarly, $\mathbf{f(X)}$ refers to mapping the points indexed by each row in $\mathbf{X}$ to $\mathbb{R}^k$. That is, the output of $\mathbf{f(X)}$ is a real-valued matrix of dimension $n \times k$. Therefore, the equivalent solution to map $\{\mathbf{X}_1, ...\mathbf{X}_n\} \subset \mathbb{R}^d$ while preserving local neighborhood into $\{f(\mathbf{X}_1), ...f(\mathbf{X}_n)\} \subset \mathbb{R}^k$ is to minimize this objective function in (1) for a specific graph Laplacian $\mathbf{L_X}$ that we describe below. This popular graph Laplacian, under which the above results were studied is that of graphs whose adjacency matrices are represented by the Gaussian kernel given by

$$\mathbf{L(X}, \sigma)_{ik} = \begin{Bmatrix} \sum_{k \neq i} e^{(-\frac{\|\mathbf{x_i} - \mathbf{x_k}\|^2}{\sigma})} & \text{if } i = k \\ -e^{(-\frac{\|\mathbf{x_i} - \mathbf{x_k}\|^2}{\sigma})} & \text{if } i \neq k \end{Bmatrix} \tag{2}$$

where the scalar σ in here is also referred to as kernel bandwidth. The seminal work in (Giné, Koltchinskii et al. 2006; Belkin and Niyogi 2005, 2007) showed that this discrete Graph Laplacian converges to the Laplace-Beltrami operator. Minimizing this objective of Equation 1 under the constraint $\mathrm{Tr}(\mathbf{f(X)}^\mathbf{T} \mathbf{D f(X)}) = \mathbf{I}$ where $\mathbf{I}$ is identity matrix, to avoid a trivial solution of $\mathrm{Tr}(\mathbf{f(X)}^\mathbf{T} \mathbf{L_X f(X)}) = 0$ is equivalent to setting the solution for the embedding $\mathbf{f(X)}$ to be the d smallest eigenvectors of $\mathbf{L_X}$.

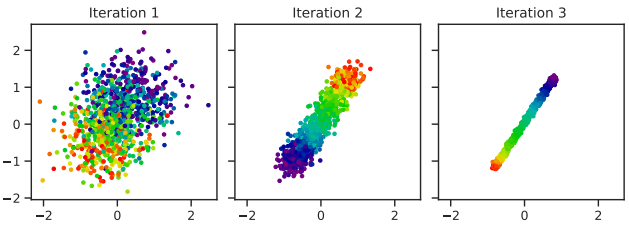

Figure 2: Embeddings of our supervised manifold learning query on CUB-200-2011 for 3 iterations with input features extracted from state-of-the-art CGD (Jun et al. 2019) deep image retrieval architecture with ResNet 50 backbone and G type global descriptors. The colors indicate different class labels. We show that these embeddings preserve information about the class separation and the locality structure required for classification.

Supervised manifold learning queries (SMLQ)

It has been shown in (Vural and Guillemot 2017) that this formulation for unsupervised manifold learning of minimizing equation (2) can be extended to the case of supervised manifold learning by posing the objective function as a difference of the terms in (1) as shown below.

$$v(\mathbf{f(X)}) = \mathrm{Tr}(\mathbf{f(X)}^\mathbf{T} \mathbf{L_X f(X)}) - \alpha\, \mathrm{Tr}(\mathbf{f(X)}^\mathbf{T} \mathbf{L_Y f(X)}) \tag{3}$$

Note that the formula for computing $\mathbf{L_Y}$ over $\mathbf{Y}$, is the same as the one used in (2) to compute $\mathbf{L_X}$ from $\mathbf{X}$. They provide results explaining the effect of optimizing such a loss for the purposes of learning an embedding $\mathbf{f(X)}$ for supervised learning. Their results are agnostic to the choice of neighborhood graphs defined on $\mathbf{X}, \mathbf{Y}$ to obtain the corresponding Laplacians used in this objective. An example for such an embedding when applied to features extracted from state-of-the-art CGD (Jun et al. 2019) deep image retrieval architecture with ResNet 50 backbone is shown in Figure 2.

Separation-regularity trade-off The intuition is that since equation (3) is a discrete version of a difference of terms of the kind in (1), therefore this formulation looks for a function that has a slow variation on the manifold $\mathcal{M}_X$ in order to smoothly preserve neighborhood relations between the input features. It does this while ensuring the function has a fast variation on a manifold $\mathcal{M}_Y$ with regards to $\mathbf{Y}$, therefore encouraging larger separation with regards to the label manifold. Therefore, this second term acts as a regularizer to make sure similar features are not embedded way closer than needed. This is mathematically substantiated by Theorem 9 in (Vural and Guillemot 2017) (restated in the Appendix D) as it shows that this regularization is required in order to minimize the generalization error of a classifier applied on the output of supervised manifold learning obtained via minimization of equation (3) for any choice of

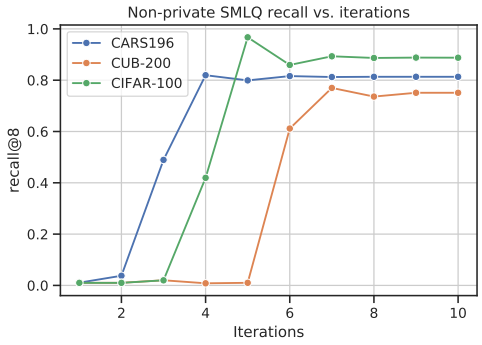

Figure 3: The convergence of our SMLQ across three datasets is shown with respect to image recall based on feature embeddings over the iterations. All three datasets reasonably converge in as quick as 7 iterations. The image recall metric is discussed in the Experiments section.

positive semidefinite $\mathbf{L_X}, \mathbf{L_Y}$.

Theorem 1. *For a fixed α, the iterate*

$$\mathbf{X_t} = \frac{\mathbf{Diag(L_X)^{-1}}}{\mathbf{2}}[\alpha\mathbf{L_Y} - \mathbf{L_X}]\mathbf{X_{t-1}} + \mathbf{X_{t-1}} \quad (4)$$

monotonically minimizes the objective

$$v(\mathbf{X}_t) = \mathrm{Tr}(\mathbf{X_t^T L_X X_t}) - \alpha\,\mathrm{Tr}(\mathbf{X_t^T L_Y X_t})$$

Proof Sketch. The full proof along with the required background is in, appendix **??**. The proof strategy involves using the majorization-minimization (Hunter and Lange 2004; Lange 2016; Zhou et al. 2019) procedure in order to obtain this iterative update. We first derive a majorization function, which always upper bounds the objective everywhere except at the current iterate, where it touches it. We then note that this majorization function is a sum of convex and concave functions. This makes the minimization of the majorization function to be equivalent to using the concave-convex procedure (Yuille and Rangarajan 2002). As the update is based on majorization-minimization (MM) and CCCP which itself is a special case of MM, it thereby guarantees monotonic convergence (Hunter and Lange 2004). We refer to this iterate as the *Supervised Manifold Learning Query (SMLQ)* and the rest of the paper focuses on releasing the outputs of SMLQ with differential privacy. □

As shown in Figure 3, our iterative update converges in just 5 to 7 iterations to embed deep feature representations needed for an image retrieval task tested on 3 datasets as further detailed in the experimental section.

Complexity analysis The graph Laplacian based on the Gaussian kernel in our method is sparse and computing the sparse matrix-vector product for this specific graph Laplacian has been studied to take $\mathcal{O}(n)$ time (Alfke et al. 2018). Since in the term $\mathbf{L_Y X}_{t-1}$, the number of columns in $\mathbf{X}_{t-1}$ is k, we have an overall time complexity of $\mathcal{O}(nk)$ as the addition of $n \times k$ matrices also takes $\mathcal{O}(nk)$. That said, this does not include the complexity required to construct the Laplacian. This has been studied in (Sanjeev and Kannan 2001).

6 Privatization of the Supervised Manifold Learning Query

Preliminaries

We first share some required preliminaries on differential privacy (DP). Differential privacy guarantees that the presence of a particular record in a dataset does not significantly affect the output of a query on the dataset.

Definition 1 ((ϵ, δ)-Differential Privacy (2014))**.** *A randomized algorithm $\mathcal{A} : \mathcal{X} \to \mathcal{Y}$ is (ϵ, δ)-differentially private if, for all neighboring datasets $\mathbf{X}, \mathbf{X}' \in \mathcal{X}$ and for all $S \in \mathcal{Y}$,*

$$\Pr[\mathcal{A}(\mathbf{X}) \in S] \leq e^{\epsilon} \Pr[\mathcal{A}(\mathbf{X}') \in S] + \delta$$

Post-Processing Invariance Differential privacy is immune to post-processing, meaning that an adversary without any additional knowledge about the dataset $\mathbf{X}$ cannot compute a function on the output $\mathcal{A}(\mathbf{X})$ to violate the stated privacy guarantees.

Gaussian noise mechanism A query on a dataset can be privatized by adding controlled noise from a predetermined distribution. One popular private mechanism is the Gaussian mechanism (Dwork et al. 2006), which adds Gaussian noise depending on the query's *sensitivity*.

Definition 2 (l_2-sensitivity)**.** *Let $f : \mathcal{X} \to \mathbb{R}^k$. The l_2-sensitivity of f is*

$$\Delta_2^{(f)} = \max_{\mathbf{X}, \mathbf{X}' \in \mathcal{X}} \|f(\mathbf{X}) - f(\mathbf{X}')\|_2$$

where $\mathbf{X}, \mathbf{X}'$ are neighboring databases.

Definition 3 (Gaussian Mechanism (2014))**.** *Let $f : \mathcal{X} \to \mathbb{R}^k$. The Gaussian mechanism is defined as $\mathcal{M}_G(\mathbf{X}) = f(\mathbf{X}) + \mathbf{Y}$, where $\mathbf{Y} \sim \mathcal{N}^k(0, \sigma^2)$ with $\sigma \geq \frac{\sqrt{2\ln(1.25/\delta)}\Delta_2^{(f)}}{\epsilon}$. The Gaussian mechanism is (ϵ, δ)-differentially private.*

We use the above mechanism to privatize the SMLQ, for which we derive the sensitivity. Note that the query's utility could be improved even further via the more recent analytical Gaussian mechanism in (Balle and Wang 2018).

Derivation of SMLQ sensitivity

We derive a bound on the sensitivity for the first iteration of the SMLQ, $f(\mathbf{X}) = \frac{1}{2}\mathrm{Diag}(\mathbf{L_X})^{\dagger}\,[\alpha\mathbf{L_Y} - \mathbf{L_X}]\,\mathbf{Q} + \mathbf{Q}$, where we initialize $\mathbf{X_0}$ to a matrix $\mathbf{Q}$ such that each entry is distributed as $\mathbf{Q}_{ij} \sim \mathcal{N}(0, \sigma_q^2)$, for which σ_q is a hyperparameter chosen by the user. It is typical to use random initialization for iterative optimization. We also assume that $\mathbf{X} \in \mathbb{R}^{n \times k}$ is normalized to have unit norm rows. Under all possible cases of adding one additional unit norm record to $\mathbf{X}$ to produce a neighboring dataset $\tilde{\mathbf{X}} \in \mathbb{R}^{(n+1)\times k}$ (denoted by the constraint $d(\mathbf{X}, \tilde{\mathbf{X}}) = 1$), the sensitivity of our query is defined as $\Delta_2^{(f)} = \max_{\mathbf{X}, \tilde{\mathbf{X}}: d(\mathbf{X}, \tilde{\mathbf{X}})=1} \|f(\mathbf{X}) - f(\tilde{\mathbf{X}})\|_F$. Note that we append an extra row of zeroes to $\mathbf{X}$ and $\mathbf{Y}$ such that the matrix dimensions agree with $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ when evaluating $f(\mathbf{X}) - f(\tilde{\mathbf{X}})$. To simplify further calculations, we let $\mathbf{M}$ denote the matrix defined by

$$\mathbf{M}(\mathbf{X}, \tilde{\mathbf{X}}) = \mathrm{Diag}(\mathbf{L_X})^{\dagger}\,[\alpha\mathbf{L_Y} - \mathbf{L_X}] - \mathrm{Diag}(\mathbf{L_{\tilde{X}}})^{\dagger}\,[\alpha\mathbf{L_{\tilde{Y}}} - \mathbf{L_{\tilde{X}}}] \quad (5)$$

PrivateMail

1. **Client's input:** Raw data (or activations) $\mathbf{X}$ normalized to have unit norm rows and integer labels $\mathbf{Y}$, Gaussian kernel bandwidth σ, regularizing parameter α, variance σ_q^2 for random embedding initialization.
2. **Client computes embedding:** $\mathbf{X_t} = \frac{\mathbf{1}}{\mathbf{2}} \operatorname{Diag}(\mathbf{L_X})^{\dagger}[\alpha \mathbf{L_Y} - \mathbf{L_X}]\mathbf{X_{t-1}} + \mathbf{X_{t-1}}$ with initialization $\mathbf{X_0} = \mathbf{Q}$ such that $\mathbf{Q}_{ij} \sim \mathcal{N}(0, \sigma_q^2)$, $\mathbf{L_X}$ and $\mathbf{L_Y}$ are graph Laplacians formed over adjacency matrices upon applying Gaussian kernels to $\mathbf{X}, \mathbf{Y}$ with bandwidth σ .
3. **Client side privatization:** The client takes the following actions:
   (a) **Initialization:** Compute constant M that depends on chosen α, σ and data size n as defined in appendix **??**.
   (b) **Computation of global-sensitivity:** Compute upper bound on global sensitivity as $\Delta = \frac{M\sqrt{n+1}}{2}\|\mathbf{Q}\|_F$
   (c) **Add differentially private noise** Release $\mathbf{X_t}$ with the global sensitivity upper bound in step $3(b)$ via the (ϵ, δ)-differentially private multi-dimensional Gaussian mechanism: $\mathbf{X_t} + \mathcal{N}^{n\times k}\left(\mu = 0, \sigma^2 = \frac{2\ln(1.25/\delta)\cdot\Delta^2}{\epsilon^2}\right)$

Figure 4: Protocol for the proposed PrivateMail mechanism

and let $\mathbf{M_i}$ denote the ith row of $\mathbf{M}$.

Theorem 2. ***SMLQ sensitivity bound*** *We have that,* $\Delta_2^{(f)} \leq \frac{M\sqrt{n+1}}{2}\|\mathbf{Q}\|_F$. *where* M *is a constant defined in appendix* **??** *such that* $M \geq \|\mathbf{M_i}\|$ *for all* $\mathbf{X}$ *and* $\tilde{\mathbf{X}}$.

Proof. Note that $f(\mathbf{X}) - f(\tilde{\mathbf{X}})$ may be expressed as the product $\frac{1}{2}\mathbf{MQ}$. Thus, by sub-multiplicativity of the Frobenius norm, the global sensitivity is bounded by

$$\begin{aligned}\Delta_2^{(f)} &= \max_{\mathbf{X},\tilde{\mathbf{X}}:d(\mathbf{X},\tilde{\mathbf{X}})=1} \left\|\frac{1}{2}\mathbf{MQ}\right\|_F \\ &\leq \frac{1}{2}\|\mathbf{Q}\|_F \cdot \max_{\mathbf{X},\tilde{\mathbf{X}}:d(\mathbf{X},\tilde{\mathbf{X}})=1} \|\mathbf{M}\|_F \qquad (6)\end{aligned}$$

Since $\|\mathbf{M}\|_F = \sqrt{\sum_{i=1}^{n+1}\|\mathbf{M_i}\|^2}$, then if M is a constant as defined in the theorem, we have $\|\mathbf{M}\|_F \leq \sqrt{\sum_{i=1}^{n+1} M^2} = M\sqrt{n+1}$. Substituting this expression into the above inequality, we obtain the bound in the theorem. The derivation of a constant M relies on expanding the definition of the Laplacian matrices in (4) and applying law of cosines for the difference of vectors. For the full derivation, see appendix **??**. □

The above bound on $\Delta_2^{(f)}$ is computed for the sensitivity parameter when adding differentially private noise to the data embedding. Figure 4 summarizes the procedure for privatization, which we call *PrivateMail*.

Private iteration-distribute-recursion framework

We show that the proposed SMLQ, fortunately can be applied under a specific framework that we propose so that it can be used in conjunction with the post-processing property of differential privacy to its advantage in obtaining a much better trade-off of utility and privacy. In addition, it allows for distributing the work required for completing the iterative embedding across multiple distributed entities while still preserving the privacy. This helps further reduce the computational requirements of the client device, prior to distributing the work. The framework still holds in improving the utility-privacy trade-off even if used without distributing the computation. We notice that the only term that requires accessing the sensitive raw dataset is $\mathbf{L_X}$, but the good thing is that this term does not change over iterations, and hence is not sub-scripted by iteration t as we show in equation 4. Therefore, we first apply our proposed differentially private release of PrivateMail, to just the first iteration. The privately obtained embedding is instead used this time to rebuild the graph Laplacian $\mathbf{L_X}$. From the next iteration onwards this modified Laplacian is used instead and the post-processing property of differential privacy now holds as no iteration from now onwards needs access to the raw dataset. For this reason these iterations can as well be continued over the server or another device as opposed to the original client device that runs the first PrivateMail iteration.

PrivateMail for Image Retrieval

We apply the proposed PrivateMail mechanism to the task of private content-based image retrieval, where a client seeks to retrieve the k-nearest neighbors of their target image $\mathbf{r}$ from a server's database $\mathcal{S}$ based on the feature embedding of their target which is sent to the server. The objective is to preserve the privacy of the client's target image. We assume the setting in which the client and server have access to a relevant public database $\mathcal{P}$ of images. We propose a differentially private image retrieval algorithm where we first generate feature vectors for $\mathbf{r}$, $\mathcal{P}$, and $\mathcal{S}$ using any feature extraction model of choice. We then generate low-dimensional embeddings for these features using the SMLQ in (4). Since the query relies on the graph Laplacian of a dataset, a single target image feature is insufficient to generate its embedding. Therefore, the client concatenates $\mathbf{r}$ with the public dataset $\mathcal{P}$. The client runs one iteration of PrivateMail where noise is added via the Gaussian mechanism before recomputing the Laplacian over the private embedding. This makes the next iterations that we run to be differentially private due to the post-processing invariance property as the iteration is now functionally independent of the raw features. We then run post-processing embeddings for a varying number of iterations depending on the dataset. Furthermore, since the client and server have access to different data, the embedding of

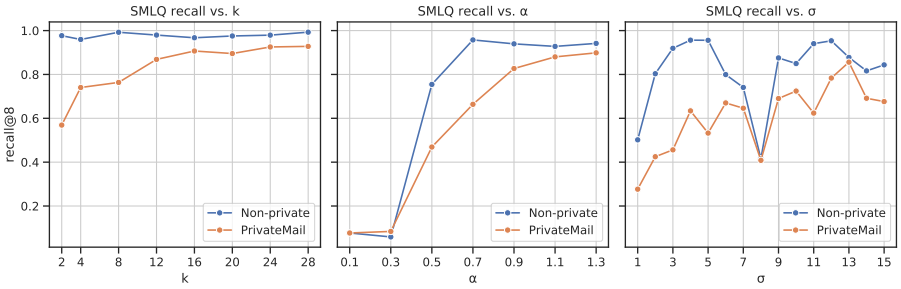

Figure 5: The effect of k, α, and σ on retrieval performance with the non-private SMLQ and the private version of PrivateMail.

$\mathbf{r} \cup \mathcal{P}$ on the client is not guaranteed to align with that of $\mathcal{S}$ on the server. We thus also concatenate $\mathcal{S}$ with $\mathcal{P}$ so the public data serves as a common "anchor" for the embeddings, which is used to align the the embeddings of $\mathbf{r}$ and $\mathcal{S}$ via the Kabsch-Umeyama rigid-transformation algorithm (Umeyama 1991). Once the server retrieves the k-nearest neighbors of the client's privatized embedding of $\mathbf{r}$ with respect to the server's non-private embedding of $\mathcal{S}$, the server gains additional information about $\mathbf{r}$ based on its neighbors. To obfuscate $\mathbf{r}$, we append a dataset $\mathcal{P}_\mathbf{r}$ of dummy queries to $\mathbf{r} \cup \mathcal{P}$ on the client-side. $\mathcal{P}_\mathbf{r}$ is generated by uniformly sampling images from the public dataset such that $\mathcal{P}_\mathbf{r}$ contains one image of every class besides the class of $\mathbf{r}$. The client's target image class is equally likely to be any of the possible classes in the dataset, so the server cannot directly infer the target class. The client is then able to filter out the retrieved images for the dummy targets. This process is visualized in Figure 1 and described in greater detail in Algorithm 1.

Algorithm 1: Differentially Private Image Retrieval

Input: Query $\mathbf{r}$, requested number of retrieved images k, number of post-processing iterations T.
Output: Server returns k nearest matches w.r.t $\mathcal{S}$.
Feature extraction: Client extracts image-retrieval features $\mathbf{X}_\mathbf{r}$, $\mathbf{X}_\mathcal{P}$, $\mathbf{X}_{\mathcal{P}_\mathbf{r}}$ for $\mathbf{r}$, $\mathcal{P}$, $\mathcal{P}_\mathbf{r}$ from trained ML model. Server extracts features $\mathbf{X}_\mathcal{P}$, $\mathbf{X}_\mathcal{S}$ for $\mathcal{P}$,$\mathcal{S}$.
Obfuscation: Client concatenates $\mathbf{X}_\mathbf{r} \cup \mathbf{X}_{\mathcal{P}_\mathbf{r}}$ and labels.
Anchoring with public data: Client concatenates $\mathbf{X}_{\text{client}} = \{\mathbf{X}_\mathbf{r} \cup \mathbf{X}_{\mathcal{P}_\mathbf{r}}\} \cup \mathbf{X}_\mathcal{P}$ and corresponding labels. Server concatenates $\mathbf{X}_{\text{server}} = \mathbf{X}_\mathcal{S} \cup \mathbf{X}_\mathcal{P}$ and corresponding labels.
Privatization: Client runs **PrivateMail** mechanism on $\mathbf{X}_{\text{client}}$ and $\mathbf{Y}_{\text{client}}$ for 1 iter to obtain embedding $\mathbf{X}'_{\text{client}}$.
for $t = 1$ **to** T **do**
  Client only runs step 2 of **PrivateMail** on $\mathbf{X}'_{\text{client}}$ (using $L_{\mathbf{X}'_{\text{client}}}$) to update the embeddings.
  Server only runs step 2 of **PrivateMail** on $\mathbf{X}_{\text{server}}$ to obtain embedding $\mathbf{X}'_{\text{server}}$.
end
Align: Non-private server embeddings and privatized client embeddings are aligned at server using Kabsch-Umeyama algorithm (Umeyama 1991)
Retrieve: Server retrieves k nearest matches for each embedding of $\mathbf{r} \cup \mathcal{P}_\mathbf{r}$ in aligned dataset and serves to the client.
Result parsing: Client locates retrieved images for $\mathbf{r}$.

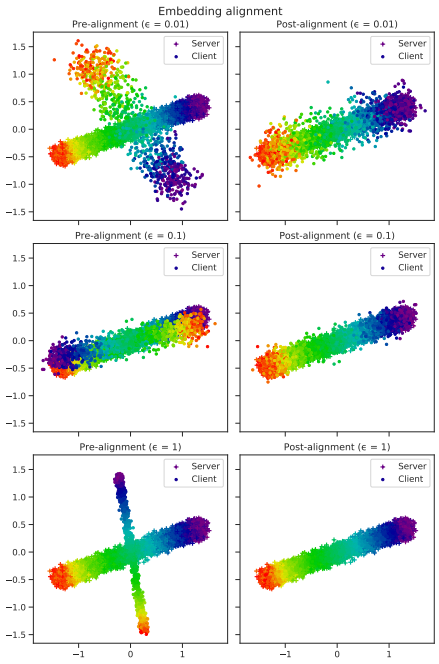

Figure 6: Embeddings for CARS196 data (with $\alpha = 0.5$ and parameters in appendix A) at varying privacy levels ϵ. We show that alignment improves as less noise is added. The privacy induced noise can be seen at various levels of ϵ.

7 Experiments

Datasets In this section we present experimental results on three important image retrieval benchmark datasets of i) Caltech-UCSD Birds-200-2011 (CUB-200-2011) (Welinder et al. 2010), ii) Cars196 (Krause et al. 2013), and iii) CIFAR-100 (Krizhevsky, Hinton et al. 2009).

Methodology We use the state-of-the-art image retrieval method of *'combination of multiple global descriptors'* (CGD) (Jun et al. 2019) with ResNet-50 (He et al. 2015) backbone to generate features for the Cars196 and CUB-200-2011 datasets. CIFAR-100 features are extracted directly from ResNet-50 pre-trained on ImageNet (Deng et al. 2009). We run Algorithm 1 on each dataset with the parameters outlined in appendix A.

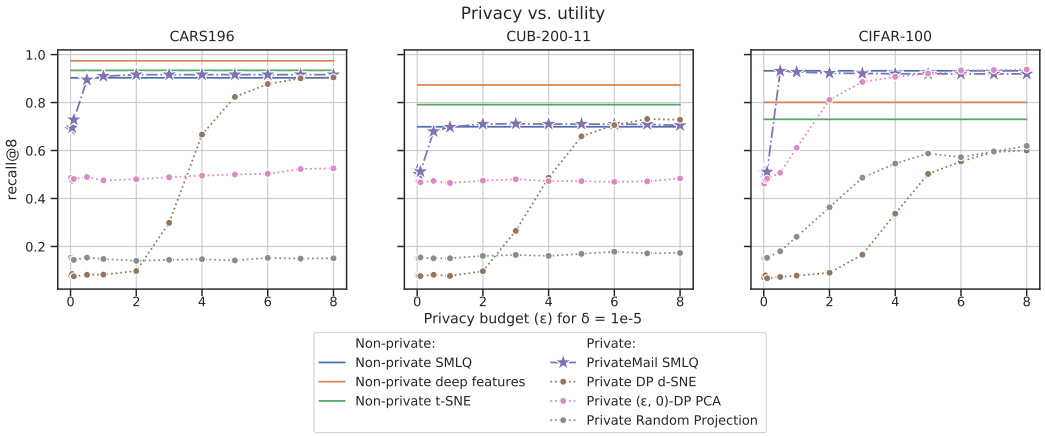

Figure 7: We compare the privacy-utility trade-off of PrivateMail with Recall@k experiments with $k = 8$ for three datasets on 6 baselines that include private and non-private methods. The lower values of ϵ refer to higher levels of privacy.

Quantitative metrics We measure retrieval performance using the Recall@k metric as used in this popular non-private image retrieval paper (Jun et al. 2019). As our proposed work is a differentially private algorithm, we study the *utility-privacy trade-off* by looking at the recalls obtained at varying levels of ϵ. Note that lower ϵ refers to higher privacy.

Baselines

We compare utility of our proposed PrivateMail mechanism against several important baselines as below.

Non-private state of the art for image retrieval We compare against the non-private method of CGD that unfortunately does not preserve privacy, and see how close we get to its performance while also preserving privacy. Note that there exists a trade-off of privacy vs utility and the main goal is to preserve privacy, while attempting to maximize utility.

Differentially private unsupervised manifold embedding A comparison with differentially private unsupervised manifold embedding method of DP-dSNE (Saha et al. 2017, 2020, 2021) is done as this is one of the most recent manifold embedding methods with differential privacy.

Non-private supervised manifold embedding We compare against non-private supervised manifold embedding to show how close our differentially private version fares in terms of achievable utility when the privacy is not at all preserved.

Non-private unsupervised manifold embedding We compare against non-private unsupervised manifold embedding method of t-SNE (Van der Maaten and Hinton 2008) to show the benefit of a supervised manifold embedding over an unsupervised embedding in terms of the utility.

Differentially private classical projections We compare against differentially private versions of more classical methods such as private PCA (Chaudhuri, Sarwate, and Sinha 2013) and private random projections (Kenthapadi et al. 2012).

Evaluation

As shown in Figure 7, PrivateMail SMLQ obtains a substantially better privacy-utility trade-off over all the considered private baselines on all the datasets. It also reaches closer to the methods that do not preserve privacy on CARS196. It even meets the non-private performance on CIFAR-100 at much higher levels of privacy (lower ϵ's). DP-dSNE reaches the performance of PrivateMail only at low levels of privacy on 2 out of the 3 datasets, while PrivateMail does substantially better at high-levels of privacy preservation. A similar phenomenon happens again with respect to private PCA on CIFAR-100.

Effect of k, α, σ In Figure 5, we study the sensitivity of our method's performance with respect to various parameters such as choice of embedding dimension k, the weighting parameter α which acts as a regularizer for the embedding by weighting the graph Laplacians in the term $\mathbf{L_X} - \alpha\mathbf{L_Y}$ in our embedding update, and the σ parameter used in defining the Gaussian kernels used to build $\mathbf{L_X}, \mathbf{L_Y}$. As shown, tuning of k, α is stable while tuning of σ requires a bit of a grid search. However, since we are in the supervised setting, standard methods for tuning could be used for practical purposes.

Qualitative visualizations Example of PrivateMail embeddings are given in Figure 6 for different values of privacy parameter ϵ pre- and post- server-client alignment.

8 Conclusion

We proposed a differentially private supervised manifold learning method and applied it to the private image retrieval problem. That said, there are a broad range of applications for manifold learning beyond that of image retrieval. Therefore, it would be interesting to investigate the potential benefits of doing these other tasks in a privacy preserving manner. We would like to extend the derived global sensitivity results to smooth sensitivities (Nissim, Raskhodnikova, and Smith 2007) in order to potentially further improve the privacy-utility trade-off.

References

Alfke, D.; Potts, D.; Stoll, M.; and Volkmer, T. 2018. NFFT meets Krylov methods: Fast matrix-vector products for the graph Laplacian of fully connected networks. *Frontiers in Applied Mathematics and Statistics*, 4: 61.

Arora, R.; and Upadhyay, J. 2019. Differentially Private Graph Sparsification and Applications. *Advances in neural information processing systems*.

Balle, B.; and Wang, Y.-X. 2018. Improving the Gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *International Conference on Machine Learning*, 394–403. PMLR.

Belkin, M.; and Niyogi, P. 2003. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6): 1373–1396.

Belkin, M.; and Niyogi, P. 2005. Towards a theoretical foundation for Laplacian-based manifold methods. In *International Conference on Computational Learning Theory*, 486–500. Springer.

Belkin, M.; and Niyogi, P. 2007. Convergence of Laplacian eigenmaps. *Advances in Neural Information Processing Systems*, 19: 129.

Chamikara, M. A. P.; Bertok, P.; Khalil, I.; Liu, D.; and Camtepe, S. 2020. Privacy preserving face recognition utilizing differential privacy. *Computers & Security*, 97: 101951.

Chaudhuri, K.; Sarwate, A. D.; and Sinha, K. 2013. A Near-Optimal Algorithm for Differentially-Private Principal Components. *Journal of Machine Learning Research*, 14.

Chen, W.; Liu, Y.; Wang, W.; Bakker, E.; Georgiou, T.; Fieguth, P.; Liu, L.; and Lew, M. S. 2021. Deep Image Retrieval: A Survey. *arXiv preprint arXiv:2101.11282*.

Choromanska, A.; Choromanski, K.; Jagannathan, G.; and Monteleoni, C. 2016. Differentially-private learning of low dimensional manifolds. *Theoretical Computer Science*, 620: 91–104.

Coifman, R. R.; and Lafon, S. 2006. Diffusion maps. *Applied and computational harmonic analysis*, 21(1): 5–30.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Donoho, D. L.; and Grimes, C. 2003. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences*, 100(10): 5591–5596.

Dubey, S. R. 2020. A Decade Survey of Content Based Image Retrieval using Deep Learning. *arXiv preprint arXiv:2012.00641*.

Dwork, C.; Kenthapadi, K.; McSherry, F.; Mironov, I.; and Naor, M. 2006. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*, 486–503. Springer Berlin Heidelberg.

Dwork, C.; Roth, A.; et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4): 211–407.

Dwork, C.; Smith, A.; Steinke, T.; and Ullman, J. 2017. Exposed! a survey of attacks on private data. *Annual Review of Statistics and Its Application*, 4: 61–84.

Elmehdwi, Y.; Samanthula, B. K.; and Jiang, W. 2014. Secure k-nearest neighbor query over encrypted data in outsourced environments. In *2014 IEEE 30th International Conference on Data Engineering*, 664–675. IEEE.

Garfinkel, S.; Abowd, J. M.; and Martindale, C. 2018. Understanding Database Reconstruction Attacks on Public Data: These attacks on statistical databases are no longer a theoretical danger. *Queue*, 16(5): 28–53.

Giné, E.; Koltchinskii, V.; et al. 2006. Empirical graph Laplacian approximation of Laplace–Beltrami operators: Large sample results. In *High dimensional probability*, 238–259. Institute of Mathematical Statistics.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep Residual Learning for Image Recognition. *arXiv preprint arXiv:1512.03385*.

Hunter, D. R.; and Lange, K. 2004. A tutorial on MM algorithms. *The American Statistician*, 58(1): 30–37.

Jones, P. W.; Maggioni, M.; and Schul, R. 2008. Manifold parametrizations by eigenfunctions of the Laplacian and heat kernels. *Proceedings of the National Academy of Sciences*, 105(6): 1803–1808.

Jun, H.; Ko, B.; Kim, Y.; Kim, I.; and Kim, J. 2019. Combination of multiple global descriptors for image retrieval. *arXiv preprint arXiv:1903.10663*.

Kenthapadi, K.; Korolova, A.; Mironov, I.; and Mishra, N. 2012. Privacy via the johnson-lindenstrauss transform. *arXiv preprint arXiv:1204.2606*.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3D Object Representations for Fine-Grained Categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*. Sydney, Australia.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Lange, K. 2016. *MM optimization algorithms*. SIAM.

Lei, X.; Liu, A. X.; Li, R.; and Tu, G.-H. 2019. Seceqp: A secure and efficient scheme for sknn query problem over encrypted geodata on cloud. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, 662–673. IEEE.

Li, Z.; and Zhang, Y. 2020. Label-Leaks: Membership Inference Attack with Label. *arXiv preprint arXiv:2007.15528*.

Matsui, Y.; Yamaguchi, T.; and Wang, Z. 2020. CVPR2020 Tutorial on Image Retrieval in the Wild. https://matsui528.github.io/cvpr2020_tutorial_retrieval/.

Nissim, K.; Raskhodnikova, S.; and Smith, A. 2007. Smooth sensitivity and sampling in private data analysis. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, 75–84.

Saha, D. K.; Calhoun, V. D.; Du, Y.; Fu, Z.; Panta, S. R.; Kwon, S.; Sarwate, A.; and Plis, S. M. 2021. Privacy-preserving quality control of neuroimaging datasets in federated environment. *bioRxiv*, 826974.

Saha, D. K.; Calhoun, V. D.; Panta, S. R.; and Plis, S. M. 2017. See without looking: joint visualization of sensitive multi-site datasets. In *IJCAI*, 2672–2678.

Saha, D. K.; Calhoun, V. D.; Yuhui, D.; Zening, F.; Panta, S. R.; and Plis, S. M. 2020. dSNE: a visualization approach for use with decentralized data. *BioRxiv*, 826974.

Sanjeev, A.; and Kannan, R. 2001. Learning mixtures of arbitrary gaussians. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, 247–257.

Shi, Y.; Davaslioglu, K.; and Sagduyu, Y. E. 2020. Over-the-air membership inference attacks as privacy threats for deep learning-based wireless signal classifiers. In *Proceedings of the 2nd ACM Workshop on Wireless Security and Machine Learning*, 61–66.

Shokri, R.; Stronati, M.; Song, C.; and Shmatikov, V. 2017. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, 3–18. IEEE.

Song, L.; Shokri, R.; and Mittal, P. 2019. Membership inference attacks against adversarially robust deep learning models. In *2019 IEEE Security and Privacy Workshops (SPW)*, 50–56. IEEE.

Steil, J.; Hagestedt, I.; Huang, M. X.; and Bulling, A. 2019. Privacy-aware eye tracking using differential privacy. In *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*, 1–9.

Truex, S.; Liu, L.; Gursoy, M. E.; Yu, L.; and Wei, W. 2018. Towards demystifying membership inference attacks. *arXiv preprint arXiv:1807.09173*.

Umeyama, S. 1991. Least-squares estimation of transformation parameters between two point patterns. *IEEE Computer Architecture Letters*, 13(04): 376–380.

Upadhyay, J. 2014. Randomness efficient fast-johnson-lindenstrauss transform with applications in differential privacy and compressed sensing. *arXiv preprint arXiv:1410.2470*.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Vural, E.; and Guillemot, C. 2017. A Study of the Classification of Low-Dimensional Data with Supervised Manifold Learning. *J. Mach. Learn. Res.*, 18(1): 5741–5795.

Welinder, P.; Branson, S.; Mita, T.; Wah, C.; Schroff, F.; Belongie, S.; and Perona, P. 2010. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology.

Xia, Z.; Zhu, Y.; Sun, X.; Qin, Z.; and Ren, K. 2015. Towards privacy-preserving content-based image retrieval in cloud computing. *IEEE Transactions on Cloud Computing*, 6(1): 276–286.

Yao, B.; Li, F.; and Xiao, X. 2013. Secure nearest neighbor revisited. In *2013 IEEE 29th international conference on data engineering (ICDE)*, 733–744. IEEE.

Yuille, A. L.; and Rangarajan, A. 2002. The concave-convex procedure (CCCP). In *Advances in neural information processing systems*, 1033–1040.

Zhou, H.; Hu, L.; Zhou, J.; and Lange, K. 2019. MM algorithms for variance components models. *Journal of Computational and Graphical Statistics*, 28(2): 350–361.

Zhou, W.; Li, H.; and Tian, Q. 2017. Recent advance in content-based image retrieval: A literature survey. *arXiv preprint arXiv:1706.06064*.