# Consensual and Privacy-Preserving Sharing of Multi-Subject and Interdependent Data

Alexandra-Mihaela Olteanu, Kévin Huguenin, Italo Dacosta, Jean-Pierre Hubaux

# Consensual and Privacy-Preserving Sharing of Multi-Subject and Interdependent Data

Alexandra-Mihaela Olteanu
EPFL, UNIL–HEC Lausanne
alexandramihaela.olteanu@epfl.ch

Kévin Huguenin
UNIL–HEC Lausanne
kevin.huguenin@unil.ch

Italo Dacosta
EPFL
italo.dacosta@epfl.ch

Jean-Pierre Hubaux
EPFL
jean-pierre.hubaux@epfl.ch

***Abstract*—Individuals share increasing amounts of personal data online. This data often involves–or at least has privacy implications for–data subjects other than the individuals who shares it (e.g., photos, genomic data) and the data is shared without their consent. A sadly popular example, with dramatic consequences, is revenge pornography. In this paper, we propose ConsenShare, a system for sharing, in a consensual (wrt the data subjects) and privacy-preserving (wrt both service providers and other individuals) way, data involving subjects other than the uploader. We describe a complete design and implementation of ConsenShare for photos, which relies on image processing and cryptographic techniques, as well as on a two-tier architecture (one entity for detecting the data subjects and contacting them; one entity for hosting the data and for collecting consent). We benchmark the performance (CPU and bandwidth) of ConsenShare by using a dataset of 20k photos from Flickr. We also conduct a survey targeted at Facebook users ($\mathbf{N = 321}$). Our results are quite encouraging: The experimental results demonstrate the feasibility of our approach (*i.e.,* acceptable overheads) and the survey results demonstrate a potential interest from the users.**

## I. Introduction

Individuals share increasing amounts of personal data online. Powered by the emergence of specialized platforms, such as OSNs, the variety of the personal data shared online has also substantially increased over the last decade, including content as diverse as contact data (address books), multimedia data (photo, audio, videos), location data and genomic data.

Recent studies highlighted the fact that such data often involves (and has privacy implications for) data subjects other than the individual who shares them online [5]. This concept, referred to as *multiple-subject personal data* (MSPD; the term was coined by Gnesi et al. [26]) or as *co-owned/multi-party data* (by Such et al. [70]), applies to numerous types of data, one of the most widespread examples being group photos and videos. A sadly popular example [59], [60], with dramatic consequences, is revenge pornography (*i.e.,* the disclosure of photos or videos portraying sexually explicit activity, typically after the end of the relationship between the partners), which can also occur on regular platforms such as Facebook [61].

Beyond MSPD data, recent studies showed that seemingly strictly personal data reveals information about other individuals [19], [38], [47], [55]. This concept is referred to as *interdependent personal data* (IPD; the term *interdependent privacy* was coined by Biczók and Chia [13]). The root cause of interdependent privacy is the fact that the personal data of somehow related people (e.g., friends, colleagues, relatives) are correlated. A typical example, introduced by Humbert et al. [38], is genomic data: The genomes of individuals, shared on specific platforms (*e.g.,* 23andme), reveal information about the genomes of their relatives.

Most of the time, the sharing and the disclosure of multiple-subject or interdependent personal data occurs without the consent of the involved individuals, possibly creating so-called multi-party privacy conflicts [69], [70], which are known to be difficult to resolve. Although the notion of consent is known to be fundamental and at the core of most of data-protection and privacy laws, as well as terms of use of online sharing platforms, very few technical solutions exist, to the best of our knowledge, for detecting and sharing such data, in a consensual and privacy-preserving way. Several protocols have been studied [12], [16], [41], [44], [80], [82], but there are no associated tools to aid users implement these and, more importantly, they are all based on the assumption that users are aware when data regarding them is shared, which is not always the case. Existing technical solutions are limited in terms of the considered adversary (*i.e.,* they typically disregard the case where the data is disclosed to the service provider), of the detection of the data-subjects and of the privacy guarantees. For instance, Facebook enables its users to review the tags that identify them in photos before they are made visible to other users, and possibly remove them; yet, even though such a tag could eventually be removed by a user, Facebook does have access to the corresponding information, *i.e.,* the fact that the tagged user most probably appears in the photo.

In this paper, we tackle the problem of designing and building a system for sharing, in a consensual and privacy-preserving way, multiple-subject or interdependent personal data (MSPD/IPD). Specifically, in accordance with Nissenbaum's definition of privacy as contextual integrity [52], we seek to give individuals control on the dissemination of data that involves them. This problem is difficult for several reasons. Identifying the data subjects of some data, or more generally the individuals whose privacy can be affected by the disclosure of the data, is far from trivial and highly data-

Network and Distributed Systems Security (NDSS) Symposium 2018
18-21 February 2018, San Diego, CA, USA
ISBN 1-1891562-49-5
http://dx.doi.org/10.14722/ndss.2018.23002
www.ndss-symposium.org

dependent. In addition, the fact that this identification, as well as the collection of the consent and the preprocessing/sharing of the data (in compliance with the obtained consent) must be done in a privacy-preserving way, with respect to the involved service providers and individuals, makes the design of such a system even more difficult. Our survey results (Section VIII) suggest that users are both concerned about this threat and potentially interested in our proposed solution.

We propose a generic solution able to handle various types of such data, and we identify the different core building blocks of a system for sharing data online, as well as the design choices to adapt to the specifics of the different data types. We focus on the case of photos, and we design and implement a working solution named ConsenShare. ConsenShare relies on two different entities: an identity management service (IMS) and a content management service (CMS). The first is in charge of identifying[1] and contacting the individuals involved in the data about to be shared on the platform that is operated by the second. The second is in charge of collecting the data and the consent, and of preprocessing and sharing the data. At the core of ConsenShare lies a distributed protocol based on standard cryptographic primitives and image processing operations, which ensures that the information learned by the IMS, the CMS and the involved individuals is minimal, especially in the case where some of the involved individuals do not give their consent. An example of a typical setting for ConsenShare would be, for the case of photos, Facebook acting as the IMS and Flickr as the CMS. ConsenShare is, to the best of our knowledge, the first such system; it addresses an important and timely problem. In fact, using such a system before sharing MSPD/IPD data online might become mandatory by law in a few years. Service providers and law makers are already making efforts in this direction, in particular for revenge pornography; these are not perfect–from a privacy perspective–for the users.[2,3,4,5] Such a solution would aid with law suits avoidance, as a CMS might be held liable for allowing the sharing of MSPD/IPD data (as was the case with fake news on OSNs). Furthermore, as our solution would represent a user-desired feature in an CMS, adoption might also lead to increasing the user base (and thus the revenue).

We perform a security and privacy analysis of ConsenShare. By using an unbiassed random sample of 17k+ photos from Yahoo's YFCC100m dataset (Flickr [72]), we also evaluate its performance in terms of CPU and bandwidth consumption, in the (worst case) scenario where all the individuals who appear on a photo are asked for consent. Our experimental results show that the CPU time is negligible for the users and for the CMS. As for the bandwidth overhead (w.r.t. to the baseline case where users directly upload their photos to the CMS), this is approximately equal to the photo size for the user who uploads the photo (as the photo must be sent to the IMS, in addition to the CMS) and 34.78% for the CMS; for the IMS, the bandwidth usage is roughly equal to the size of the uploaded photos. We complement our evaluation with an online survey on multiple-subject and interdependent personal data, targeted at Facebook users and conducted via the Amazon Mechanical Turk platform (N=321). The survey results indicate that a system like ConsenShare could be desirable. For instance, 69.5% of the participants are concerned by the sharing of multimedia data that involves them, 27.4% are potential victims of revenge pornography (*i.e.,* they have shared intimate photos or videos), and 53.6% would certainly use a system like ConsenShare. We also study the potential adoption of such a system by analyzing the incentives (*e.g.,* business opportunities and models) of the different stakeholders, namely the end-users, the IMS and the CMS. In summary, our contributions are the following: (1) We identify and frame the timely and critical problem of consensual and privacy-preserving sharing of MSPD/IPD data (2) We design, implement and evaluate the first system to address this problem for photos; we also propose a generic system for other types of data and identify the different challenges inherent to its design, as well as incentives for adoption for all the parties involved. Our results are quite encouraging: The experimental results demonstrate the feasibility of our approach and the survey results demonstrate potential interest from the users.

**Roadmap:** The remainder of the paper is organized as follows. We survey the related work, with an emphasis on legal, social and technical aspects of the problem, in Section II. We describe the system model and list our design goals in Section III. We give a high-level description of a generic solution, namely ConsenShare, in Section IV; we propose and give a detailed description of a solution specific to photos in Section V. We provide a security and privacy analysis of ConsenShare in Section VI. We report on our data-driven experimental evaluation of ConsenShare in Section VII. We discuss the adoption of ConsenShare, based on (among other things) the results of our user survey, as well as its limitations and its extension to data other than photos in Sections VIII and IX respectively. We conclude the paper in Section X.

## II. Related Work

Consensual and privacy-preserving sharing of multi-subject and interdependent data online is a multi-faceted problem, as advocated by Good [28]: it includes legal, social and technological dimensions. In this section, we survey the related work in these dimensions, beginning with the legal aspects.

The notion of individual control over information about oneself and more specifically that of consent for information disclosure is currently the basis of most definitions of privacy, including Nissembaum's contextual integrity [52], terms of use, and data protection and privacy laws in most countries [17]. This is the case for the Consumer Privacy Bill of Rights [3], adopted by the US White House, and the General Data Protection Regulation (GDPR) (Regulation EU 2016/679), recently adopted by the EU Parliament.

Although the general case of MSPD/IPD is not explicitly addressed by current laws, because of its complexity, it is mentioned in multiple places. For instance, in case law [2], the court found that "an individual's personal autonomy makes him master of all those facts about his own identity, such as his

[1]While not privacy-mindful, an application for identifying people from a picture taken in public, FindFace [22], is becoming popular in Russia.

[2]https://reddit.zendesk.com/hc/en-us/articles/205704725 and https://www.reddit.com/help/contentpolicy/

[3]https://www.nytimes.com/2017/04/05/us/facebook-revenge-porn.html

[4]https://www.theguardian.com/technology/2017/nov/07/facebook-revenge-porn-nude-photos

[5]https://support.twitter.com/articles/18311

name, health, sexuality, ethnicity, his own image [...] and *also of the 'zone of interaction' [...] between himself and others*". In addition, Opinion 5/2009 on online social networking produced by the Working Party on Data Protection (*i.e.,* an advisory board set up by the EU for the reform of the data protection laws) mentions the case of online social networks (OSN) users uploading data about other individuals, possibly not members of the OSN.

In the context of MSPD/IPD, specific data types received particular attention: photos, in light of the right to one's own image, genomic data [1], and more recently, photos and videos containing sexually-explicit content, namely revenge pornography, against which laws have been passed in Canada, France, Israel, Japan, the United Kingdom and in several states in the US (to name a few). In addition, online service providers, including Reddit[2], Facebook[3], and Twitter[5] have also reacted to this new trend and updated their terms of use accordingly. Yet, neither the laws nor terms of use are self-enforcing, and technical solutions are therefore needed.

Online services recently began including features to cope with content uploaded without the consent of some of the individuals whose privacy is affected by it, typically for photos. Facebook, for instance, enables its users to report such content and to remove references to their identities attached to shared content. However, such features still suffer from the following problems: (1) Individuals cannot automatically detect that content having privacy implications for them has been shared, unless an explicit reference to their identities is attached to it. (2) Even though the content is eventually removed, the damage, in terms of privacy, is done as the service provider and possibly some users have seen the content.

The need for and the design of collaborative privacy schemes for MSPD/IPD is an active topic in the literature. Gnesi et al. [27] introduce the notion of MSPD as data that contains identifiers that refer to more than one person, as is the case of pictures, phone records, co-locations or medical reports. They also discuss a technical solution for protecting MSPD based on user-defined privacy policies, however, it does not guarantee any protection against the service provider.

In the context of OSNs, Such et al. [69], [70] study the so-called multi-party privacy conflicts (MPC) in the case of pictures. They identify the sources of such conflicts and the different non-technical strategies used by users to cope with them, including avoidance or individual/collaborative resolution. Collaborative privacy policy enforcement solutions were also proposed by Beato et al. [11] (based on secret sharing), by Squicciarini et al. [67], [68] (based on the Clarke-Tax mechanism from game theory), by Ratikan et al. [58] (based on majority voting), as well as by Hu et al. [32]–[35] (based on access control). Again, in contrast to our work, these solutions assume a trusted model for the service provider (OSN). Ilia et al. [39] proposes a collaborative multi-party access control model for OSNs where the service provider is considered honest-but-curious. However, it assumes that the data uploader is honest, yet privacy careless. Our work, on the contrary, assumes that the data uploader and other users could be malicious. Finally, collaborative privacy policies mechanisms have been proposed in other contexts such as sharing photos through instant messaging platforms [45] and personal data stored on others' devices (e.g., phone numbers) [29].

Researchers also proposed mechanisms to defend against untrusted providers in OSNs. De Cristofaro et al. [18] propose a privacy-enhanced alternative to microblogging OSNs such as Twitter. Their solution protects posts' contents, hashtags and follower interests from the service provider. Ion et al. [10] describe a privacy-enhancing mechanism that enables users to share data over any web-based OSN and provides confidentiality against unauthorized parties, including the service provider. Feldman et al. [23] propose a framework for OSNs that provides not only confidentiality guarantees, but also integrity protection (*e.g.,* against equivocation attacks) against an untrusted service provider. Secure JPEG techniques [83] can also be used to hide part of the photo content from the service provider. Although these works offer different levels of protection against an untrusted service provider, they do not offer mechanisms for detecting/identifying the individuals involved in the content and for implementing collaborative privacy policies for MSPD/IPD.

Identifying individuals involved in content shared online is a difficult problem. In several cases, including the case of pictures, such identification comes down to a classification problem. For instance, machine learning techniques can be use for detecting faces on encrypted images [15], [81]. Moreover, Ziad et al. [86] describe the use of homomorphic encryption for performing general image-processing operations (*e.g.,* spatial filtering, anti-aliasing) on remote (untrusted) servers in a privacy-preserving way; to use such an approach in our framework, certification of results would also be needed (*e.g.,* through blind signatures). Closer to our work, He et al. [30] describe a system for partial image sharing in OSNs that enables data owners to define private regions in an image, support for popular image transformations and set different privacy policies for each user associated with an image. In the same area, Ilia et al. [40] propose a fine-grained access control mechanism that enables users associated with an image to restrict the exposure of their own faces; this approach handles multi-party privacy policies conflicts and is compatible with existing access control mechanisms. These works, however, focus only on the problem of sharing images in OSNs. Our work, in contrast, focuses on different MSPD/IPD types, not only images, and deals with the problem of detecting involved individuals in a privacy-preserving way.

## III. System Model & Design Goals

We describe next our system model, the adversaries and the threat model we consider, our assumptions and design goals.

*a) System Model:* In our model, we consider the following major entities: Users and a Content Management Service (CMS) – *e.g.,* Flickr for photos, YouTube for videos, or OpenSNP for genomic data. Users[6] can upload content to the CMS (with a certain target audience for visibility consisting of a set of users and/or the general public); any part of the content that concerns another user is sent to her for approval, along with any relevant contextual data (*e.g.,* the identity of the uploader, description, upload time, target audience, *etc.*); this content is only visible to these parties (and to the CMS) *only if* the concerned user grants their consent.

[6]Note that we refer to "regular" users; we do not consider professionals such as journalists, who follow specific accountability rules regarding the publication of content, are liable for it and have a reputation to uphold.

*b) Threat model:* In our model, we assume that the adversaries are the users (individuals), the online services (*e.g.,* the CMS–other services can be included in the protocol as we shall see, *e.g.,* the IMS) and third parties (*e.g.,* external observers). Individuals can be active adversaries. For instance, a malicious user could try to bypass the system to fully publish sensitive content (e.g., compromising photos of other users) without obtaining consent from the affected users (possibly by colluding with other malicious users or by creating fake profiles). A malicious user might also try to monitor and tamper with the communications among the different parties in our system to infer private information about other users, e.g., their real names. The CMS and the IMS are assumed to be honest-but-curious, *i.e.,* they will follow the protocol, but they could try to infer sensitive information from the data observed. For instance, the CMS might try to learn the sensitive content specific to particular users before they give consent or infer the social networks or strength of social ties of some users based on the consent requests that are sent out and their responses (*e.g.,* if Bob often accepts that Alice share content regarding him, they are likely good friends).

*c) System Assumptions:* We assume that secure two-way communication channels have been established between all parties in our system, typically over HTTPS; we assume that the CMS and the IMS are independent parties and that they do not collude (we discuss the case of collusion in Section IX). We further assume that data from the network layers (*e.g.,* IP address) cannot be used to leak users' identities: This is a reasonable assumption as many mobile users only access the Internet through a NAT gateway offered by their Internet provider, but could be relaxed if, for instance, users make use of VPN service or anonymous networks (e.g., Tor) to access the Internet. We do not consider fingerprinting attacks in our model. Finally, we assume that software that is run locally is trusted (trusted execution environment can be used).

*d) Design Goals:* Our main goal is to design a mechanism that, in a *private* way, (1) informs users every time a piece of content regarding them is submitted and (2) enables them to grant their consent *before* such content is available to any other party (except from the uploader, of course). To this end, the design goals of our system are as follows.

- Effectiveness: the registration process should be secure, registered users should be detected in uploaded content and the sensitive content involving them should not be revealed to any component of our system until *after* they consent.
- Privacy as anonymity for the users: data submission and consent operations should not leak information about the identity of the users involved (other than the uploader).
- Unlinkability: An adversary should not be able to aggregate or link consent operations regarding different individuals associated with a particular content.
- The system should detect any malicious user behavior (*e.g.,* attempts to bypass the protocol) and not allow the sharing of sensitive parts of the content, in such cases.
- Usability and transparency to the users. This includes the fact that consent operations should provide the involved users with enough contextual information for them to make an informed decision.

## IV. Highlevel Solution

In this section we describe our proposed framework, its core components and the main technical challenges.

*a) Framework Overview:* We envision a system to which a user registers with identity information. The system's role is (i) to detect, for any content that is uploaded, what are the users affected by this content (*e.g.,* for genomic data these are the close relatives; for photos and videos these are the people who appear; etc), (ii) to contact them and ask them for consent, providing them the option to express a decision either manually, through policies, or automatic (machine learning based) and (iii) to publish the content with the proper restrictions and obfuscations (depending on the users' consent decisions). Such a system consists of several components, which we describe next.

- *Content Management Service (CMS).* These are User Generated Content sites, such as Flickr, YouTube or OpenSNP.
- *Identity Management Service (IMS).* This handles users' identities and offers services to identify the users associated with a particular content. The IMS is also in charge of users' relationships (social and family). Typically, in practice, the role of the IMS could be played by agencies or by popular OSNs, such as Facebook or, it could be split across several entities or distributed.
- *User applications (CMS and IMS).* This is the component that users interact with to publish content form one of their devices to the CMS, to review consent requests either manually, through the use of policies, or machine learning automation (learning a user's decisions patterns from a few initial manual decisions).

*b) Challenges:* Key and challenging components of our framework include: claiming identity; determining the users involved by some particular content; contacting them privately and providing them with enough context to make informed consent decisions (while hiding information for which other involved users should grant consent); the variety of types of the consent decisions (removing or obfuscating all or some parts from the content, reducing the visibility audience, *etc.*); reducing the number of consent decisions (through policies or machine learning automation); and enforcing multiple and possibly conflicting individual consent decisions–all of these in a private way. Most of these components are very data-dependent. Therefore, for simplicity's sake, in what follows we will focus on photos. In Section IX, we discuss the extension to other types of MSPD/IPD.

## V. Specific Solution: The Case of Photos

In this section, we present a working solution, for the case of photos. The main entities in this case remain the photo uploader (we refer to her as Alice), the CMS, the IMS and (potentially) the other users that appear in a particular photo (consenters). We refer to any consenter as Bob.

### A. Overview of ConsenShare

ConsenShare enables any user, Alice, to upload photos to the CMS (see Figure 3. If such a photo, $P$, contains faces of other people, Alice can choose to remove these (by blurring


(a) Original photo (b) Background image (c) Face images (d) Final photo

Fig. 1. Example application of ConsenShare on a real photo: (a) Original photo, taken by Alice; (b) Background image produced by Alice's application and sent to the IMS and CMS. Note that faces that Alice wants to appear in the photo will be asked for consent, whereas others (*e.g.,* people in the background) will be simply blurred as in Google Street View; (c) Face images, extracted from the original image, by Alice's application; (d) Final photo, hosted by the CMS, assuming one of the people accepts and the other denies (note that Alice's consent is automatically granted, as she is the uploader). In order to enable the people who appear in the photo to make an informed consent decision with the general context in mind, they receive the background image with only their own faces shown (for privacy reasons). Note that the sizes of the blurred/cropped out regions can easily be customized and increased even to include the full body (*e.g.,* [76], [77]) to avoid people being recognized through features other than their face, such as clothes. Image source: Pixabay, (https://pixabay.com/p-1517163/?no_redirect).

them, similarly to blurring on Google Street View [25]) or– if she wants these to be visible in the photo–she must first remove the faces from the photo, upload them (encrypted) separately such that the corresponding people are asked for consent (Figure 1 illustrates on a concrete example how some of the photos look). In this latter case, only the background corresponding to photo $P$, namely $P_B$, is uploaded to the CMS (after some validation from the IMS to certify that no (known) faces appear in it). This version of the photo ($P_V$) is made visible to the target audience desired by Alice as soon as the upload completes. Faces for which consent must be asked are cropped out from $P_B$ and a protocol to identify the owner of the face, contact him, provide him the photo for review and collect his consent decision is executed; this involves the IMS at different stages. We emphasize that the parts of the photo for which other users must consent are protected, as one consenter, Bob, will only be provided with the photo consisting of the background and his own face. Before Bob grants consent, only Alice (who already has access to the full photo) and Bob are able to see the part of the photo containing Bob's face, as Bob's face image is encrypted using a key created by him. In addition to this, Bob is also provided with some contextual information about the photo (such as the identity of the uploader, upload time, description and the target audience for photo visibility) to help him decide. If Bob denies consent, his face will remain cropped out in the published photo, $P_V$. If Bob grants consent for his face to appear in the photo, he provides the CMS with the needed information to decrypt his face. Before adding Bob's face to $P_V$, the CMS first performs several validation steps to ensure that Alice or another party has not tampered with the original face appearing in $P$ and that consent has been granted by the correct user.

### B. Technical Challenges and Choices

Our solution comes with several challenges, discussed here.

*1) Identity claim:* In order to use ConsenShare, users (or their legal guardians for minors) must register by providing information for face recognition, which would be used to detect them in future photos shared by others. A typical way to do this, which became mainstream, is to provide the system with an ID and/or photos (which can be verified by humans), as exemplified, for instance, by Uber [74], [75] and Airbnb [6]. To reinforce the proof of identity, webcams can be used, similarly to Microsoft's Windows Hello [79], other solutions for biometric authentication proposed in the literature [31], [46], [51] or the upcoming Apple's biometric facial recognition for unlocking iPhones [8].

*2) Privacy-preserving face/body recognition:* A major challenge in the design of ConsenShare is the privacy-preserving face detection and recognition. Although there is work in the literature detailing how classification could be performed on encrypted data (*e.g.,* [15], [81]), it is not clear how applicable these would be to the rapidly evolving face recognition algorithms, or how efficient these would be. Furthermore, this option would raise the problem of the authenticity of the classification results. Thus, in our design, we focus on face recognition on regular images; We consider feature vector based face recognition, as described in Google's popular and efficient FaceNet framework [65]. We provide the desired privacy guarantees by: performing the face detection operations locally on the uploading client's device (thus the photo is not shared with any other parties); performing the face recognition on the IMS server based on the much less sensitive information, *i.e.,* the feature vectors; and validating these operations by the CMS using the original photo once consent is granted. Note that all local operations can be performed either in a mobile application, or a web app (Javascript).

To better understand these design decisions, we give here some background information about face recognition, which the familiar reader can skip. A face representation (or feature vector) is a multi-dimensional numerical vector that encodes the features of a face (*e.g.,* the eye distance). Its main properties are: (i) A face representation is unique to a face image, hence different face photos (even belonging to the same person) result in different, yet close in terms of distance, face representations. (ii) Typically, the Euclidean distance between face representation extracted from photos depicting the face of the same person is smaller than the distance between representations extracted from photos depicting faces of different people. Thus, distances can be translated into a measure of face (dis)similarity and the problem of face recognition for some input feature vector reduces to identifying the closest feature vector to it – in the distance space – from a set of available feature vectors of the registered users.

Note that detecting faces might not be enough, as recent work shows that identifying people is possible from features other than their face, such as their clothes [14]. Our framework can be extended to include body detection and obfuscation techniques (*e.g.,* [76], [77]), as for faces; such solutions would provide more privacy, but would also involve a utility loss due to the increased obfuscation.

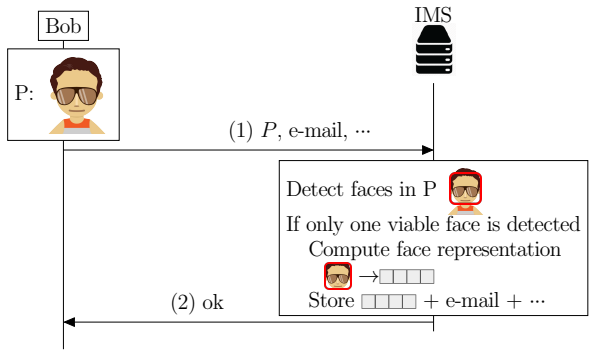



Fig. 2. ConsenShare: Register user protocol

### C. ConsenShare Main Operations

We describe the following operations: (i) register (Figure 2) and (ii) upload photo and grant/deny consent (Figure 3).

*1) Register:* The operations performed are the following.

1) Bob, a new user, uses the ConsenShare IMS application on his device to take a set of photos/videos, $P$, with her webcam. The application sends $P$, along with Bob's login information to the IMS. The idea behind this is to prevent people claiming an identity different than theirs.
2) The IMS detects the faces in $P$. If $P$ contains only one valid face, it computes the corresponding face representation, stores this along with the provided login information and sends an 'ok' response to Bob. Otherwise, an error message is returned to Bob and nothing is stored.

*2) Upload photo and grant/deny consent:* The operations performed are the following.

1) Alice selects the photo she wants to share: The CMS ConsenShare application splits it into background image, $P_B$, (which contains no faces) and the detected faces; this is done locally by the CMS provided application or webpage. Alice marks the faces that she does *not* want to ask for consent and these are blurred; the other faces are cropped out. Alice sends this $P_B$ to the IMS. The IMS should provide a public API for this operation. Note that Alice does not need to have an IMS account.
2) The IMS performs face recognition on the received $P_B$. If this is indeed a valid background photo (*i.e.,* contains no faces, contains no known faces, *etc.* depending on the policy implemented by the IMS–it could also depend on the requirements imposed by the CMS), the IMS signs and sends a message containing its hash (and information indicating whether the photo contains no face, no known faces, *etc.*). Otherwise, it signs and sends a message specifying the reasons for which the photo is not valid. The signed message, $\sigma$, is returned to Alice.
3) Alice logs in to the CMS using her credentials and forwards the received $\sigma$ together with $P_B$ and some $context$ information (*e.g.,* the desired visibility and a description of $P$) to the CMS, which verifies that all of the following holds: (1) $\sigma$ is a valid signature by the IMS, (2) the $P_B$ it received from Alice has the same hash as the one in message returned by the IMS–to prevent Alice from uploading a different photo. If these checks pass, the CMS creates a visible photo with the specified audience in $context$, $P_V$, consisting of the background image $P_B$ and stores $hash(P_B)$ and $context$.

For each detected face that must be asked for consent:

4) A face representation is computed by the ConsenShare CMS app on Alice's device and encrypted with the IMS's public key, $\mathrm{Pk}_{\mathrm{IMS}}$, using a *deterministic encryption scheme*(we discuss why using a deterministic encryption scheme is acceptable in this case in Section IX). This is then sent, along with $hash(P_B)$, to the CMS. $hash(P_B)$ is used as an identifier for the sharing of the photo.
5) The CMS generates a random session id, $sid$, marks it as pending and stores and forwards the received encrypted face representation and $sid$ directly to the IMS. As several photo uploads likely happen at the same time, the CMS acts as a mixing network for the IMS, preventing the IMS to link users to the same photo (details in Section VI).
6) The IMS decrypts the message to obtain the face representation and finds, among all the registered users, the one with the closest feature vector (*i.e.,* smallest Euclidean distance) that also satisfies a maximum acceptable similarity threshold. It then sends a message to this user (Bob), containing $sid$ (*e.g.,* a link embedded in a notification shown in the app running on Bob's device, typically the app would be provided by the IMS–eg Facebook Messenger). Note that Bob does not need to have a CMS account. In the case of a missing identity–*e.g.,* Bob is not yet a registered user–the protocol stops here and his face would remain cropped out.
7) Upon notification, the app running on Bob's device generates a pair of public/private session keys, $Pk_{sid}, Sk_{sid}$ and sends $Pk_{sid}$ and $sid$ directly to the CMS.
8) The CMS forwards $Pk_{sid}$ to Alice, who uses it to encrypt the part of the original photo containing Bob's face, as well as its position coordinates in the photo. It sends this encrypted information to the CMS.
9) The CMS stores the received information and forwards to Bob the encrypted face and the coordinates, as well as the background image $P_B$ and the corresponding $context$.
10) Bob's app recreates an image consisting of the $P_B$ and the portion in which his face appears (which he decrypts using $Sk_{sid}$), shows it to Bob along with the $context$ and uploader identity, and Bob makes a decision whether to give consent for allowing his face to be visible in this photo. Note that this can also be automated through machine learning techniques (*e.g.,* [54], [78], [85]), or enforced through policies (*e.g.,* "accept all from friends", "accept if not nude", "accept if I am the uploader", *etc.*). Before presenting the photo to Bob for consent, spam filtering techniques can be performed (*e.g.,* using senders white/black lists or performing face detection on the face image to ensure this is really a face image belonging to Bob and not some unsolicited ads, for instance).

(11a) If the decision is to allow Bob's face to appear in $P_V$, $Sk_{sid}$ is returned to the CMS as a response to $sid$. At this point, the CMS can decrypt the stored face and the coordinates it has received from Alice and validate the validity of the coordinates (*e.g.,* by verifying that the corresponding area of $P_B$ is cropped out) and that the feature representation obtained from this face image is identical to the one Alice sent to the IMS, through the CMS, in step (4). This is possible because

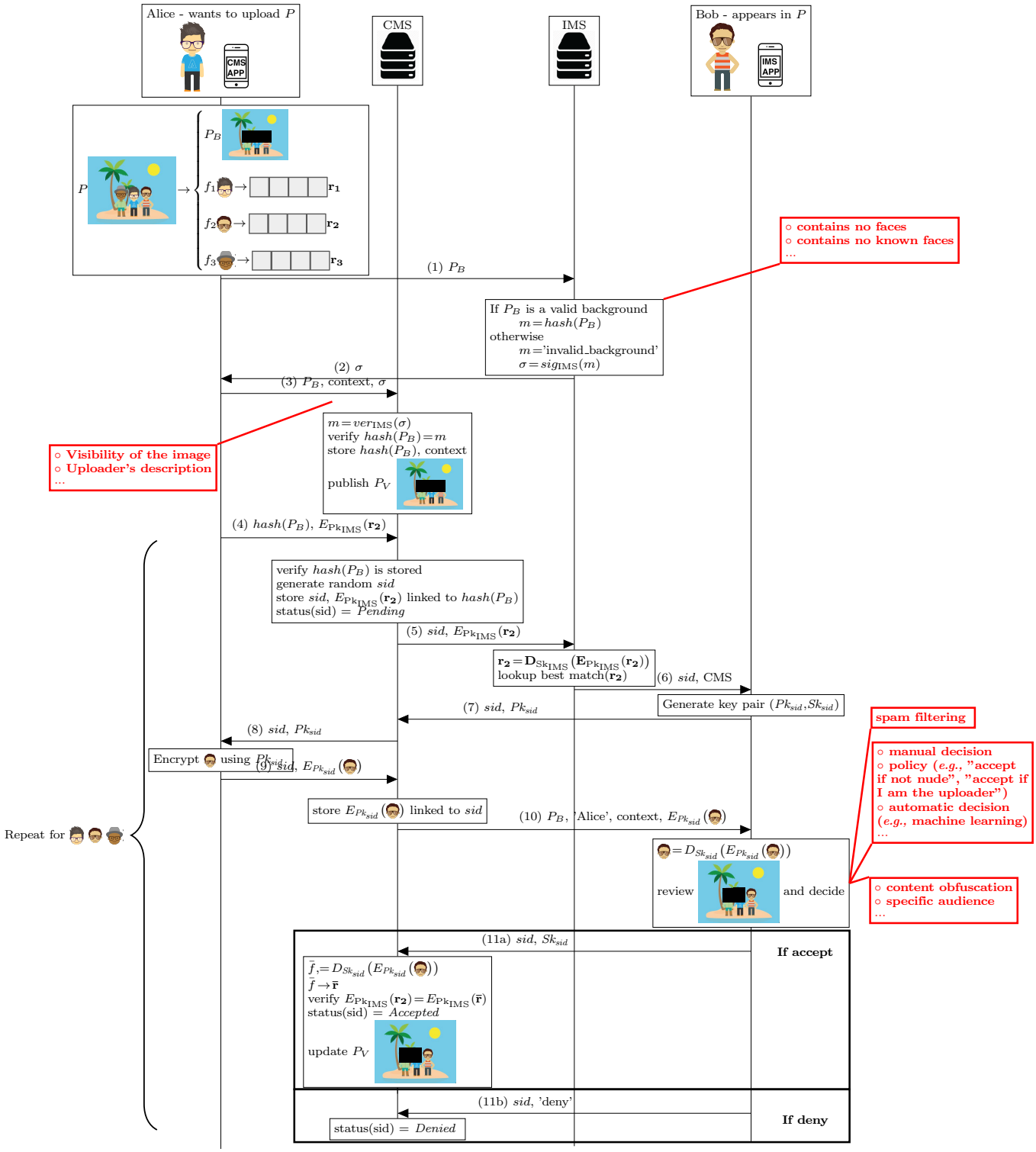



Fig. 3. ConsenShare: Upload photo protocol

the encryption scheme is deterministic. Note that, for the same person, the feature representation differs when the face image differs, therefore if two feature representations are identical, this guarantees that the original face images they were computed from are identical. If the validation is successful, the CMS adds Bob's face to the published photo $P_V$ and marks this $sid$ as accepted.

(11b) If the decision is to not allow Bob's face to appear in $P_V$, a 'deny' message is returned to the CMS as a response to $sid$, and the CMS then marks $sid$ as denied and the area corresponding to Bob's face remains cropped out.

Note that, in the case of no response from a user, his face simply remains obfuscated. This action can also be configured

by the CMS (e.g., default option after a timeout).

## VI. Security and Privacy Analysis

In this section, we demonstrate how ConsenShare satisfies the goals described in Section III. Note that the security of some parts of the system directly depend on that of the underlying technologies used (*e.g.,* face recognition is not perfect [50]). We discuss these in detail in Section IX.

### A. Effectiveness

First and foremost, the identity claim process that uses webcams at registration, prevents the creation of fake accounts. Face spoofing detection techniques (*e.g.,* [49]) can also be used to prevent malicious individuals from creating accounts on behalf of other users. Once registered, if a user's face appears in uploaded photos, he would either be asked for consent (with his face being encrypted in all communications and his identity hidden from the CMS) or his face would be blurred to begin with (depending on the uploader's choice). We discuss why malicious users cannot bypass this part of the protocol in detail in Section VI-C.

### B. Privacy as anonymity and unlinkability

Regarding the data that is visible or known to the different parties throughout the protocol, we emphasize that face detection is performed locally on the uploader's device, that is, faces are not transmitted to the IMS (only face vectors) and the CMS only receives encrypted versions of the faces to forward to the consenters. None of the consenters involved in a photo have access to each other's faces. The IMS only has access to the background image (which does not contain any faces) and to the face representations of the people appearing in all the photos (which do not disclose anything about the sensitive information–the actual face). As the CMS acts as a mixing network (it forwards a large number of messages to the IMS), the IMS cannot distinguish the lookup operations belonging to the same photo, thus it cannot link faces to faces or faces to a particular content. To make this property even stronger, the CMS can randomly mix and slightly delay messages sent to the IMS (using buffering and shuffling), as well as add dummy messages in step (5) (Figure 3). As for the CMS, it has access to the face-free background image. All the other data (face vectors and the faces) is sent encrypted to the CMS and only decrypted after the concerned users grant consent. The CMS is thus not able to identify the users that are involved in the same photo before they have given consent.

Furthermore, the CMS is also not able to link different faces (from different photos) of the same user, as face representations of a person always differ (even slightly) in each photo, making the encrypted version different. We consider the case where the CMS wants to identify users in photos submitted in the future, based on their face representations from previous granted consents (step 11a, Figure 3); the CMS would have to build a dictionary of possible face representations for a target user by adding noise at each position of the face representation array. This quickly becomes very expensive, *i.e.,* the time complexity is exponential in the size of the face representation array (*e.g.,* 128 positions in OpenFace [7]). Furthermore, we can easily protect against this attack with very little bandwidth and CPU time overhead by concatenating a random salt to the feature vector and to the face sent in step (4) and (9), respectively; In step (11), the CMS can retrieve the salt along with the face and perform the validation of the face representation. As for other similar timing, linking or side-channel attacks potentially performed by the IMS or by the CMS, these can also be deterred by traffic aggregation and randomization at the CMS and by adding dummy request at the client side (*e.g.,* the uploader's application can send more messages, to other users (in step (4)), which would be automatically disregarded by the consenters' application).

### C. Malicious user behavior

A malicious uploader cannot bypass the system by leaving faces visible in the background image, as this would immediately be detected by the IMS in step (2) (Figure 3). Malicious uploaders can also not bypass the system by sending an incorrect feature vector in step (4)–in order for someone else to provide consent in lieu of a target consenter–as this would be detected by the CMS in step (11a). Similarly, malicious users cannot bypass the system by providing a consenter with a face different than that sent to the CMS, as verifications of the face position in the photo and a comparison of the feature vector for that face and that sent to each consenter are performed in step (11a). Every message sent by an uploader, throughout the protocol, is validated by the CMS before any consenter's face is made visible to the target audience. Thus, malicious user behavior (even colluding users) results in sensitive parts of the photo not being shared with the target audience. Privacy of the sensitive content is also guaranteed, up to the point consent is granted by the concerned user. This is due to the security of the encryption schemes and the design of the system: The sensitive content is encrypted and not visible to the IMS, to the CMS, traffic snoopers or to any other users of the system.

### D. Usability and transparency

All consent requests contain contextual information for the consenter. However, in our solution, there is a chance of spamming attacks, *e.g.,* sending unwanted information to specific users (in the background of the photo, for instance). Although these can be annoying, we do not consider them extremely privacy invasive and well-known anti-spamming techniques (such as those used for e-mail) and the anti-spam mechanisms of the CMS can be used to handle this problem.

### E. Collusion cases

User-CMS and user-IMS collusions do not lead to additional information leakage. We discuss the case of CMS-IMS collusion (*e.g.,* the role of the CMS and that of the IMS are both played by Facebook). Typically, the CMS has information about the photos, but does not know anything about the identities of the faces appearing in them; the IMS can link every face with a user. In the case of collusion, the CMS and IMS entities would be able to link the users' identities to a particular photo, but the sensitive parts of the photos belonging to these users (their faces) would still not be visible unless these users grant their consent (as their decryption is only possible if consent is granted and both the CMS and IMS are honest-but-curious). Linkability could be reduced by adding dummy messages by the uploader application.

## VII. Implementation and Evaluation

To evaluate the performance of ConsenShare and demonstrate its practicality, we implemented a proof-of-concept prototype and evaluated its performance, in terms of CPU usage and bandwidth consumption, by relying on a real large photo dataset. We describe the implementation of our prototype, the dataset collection as well as our experimental setup, and we present the bandwidth and CPU consumption for the photo upload and grant consent operations.

### A. Prototype Implementation

We implemented the prototype and carried out our performance evaluation by using Python 3.6. The prototype code and documentation are available at https://www.dropbox.com/sh/g2296q7qv5kh1bd/AAAWgYAsEY6R8Q52lRQt1Ay6a?dl=0. Note that the prototype was not optimized yet; therefore, the CPU usage measurements should be considered as a loose upper bound. The prototype consists of the two server applications (the CMS and the IMS respectively), which we implemented with Flask, and a client application that supports the three main operations of the system: register (to the IMS), upload photo (to the CMS) and approve/deny consent (to the CMS). The servers use basic SQLite databases for local storage. All communication between these entities is achieved through JSON-based HTTPS requests and responses.

For face detection and feature-vector extraction, we use OpenFace [7], [56] (v.0.2.1), which is an open source Python implementation of Google's FaceNet framework [65]. We implemented the lookup operation (*i.e.,* retrieving the best matching record for an input feature vector) to return the database record that minimizes the Euclidean distance with the input feature vector: We implemented it in a naive way, that is by comparing it to all the records in the database. Note that there exist efficient techniques for finding the most similar face, based on a low-dimensional representation (embedding), in databases of up to hundreds of million faces (*e.g.,* [42]). Basic image manipulation operations, such as loading and saving image files as well as extracting faces and replacing them with black rectangles, were performed with the Python Imaging Library (Pilow, v.4.1.1). In order to avoid discrepancies in the image file sizes (and therefore in the bandwidth measurements), we configured Pilow to retain all the parameters of the JPEG/JFIF format and encoders (e.g., quality, color space, chrominance subsampling factor) from the original image processed, when saving (parts of) it.

For the basic cryptographic operations (hash, sign/verify, encrypt/decrypt and generate keys), we used the Python binding to the Networking and Cryptography library (PyNaCl [57], v.1.1.2). Specifically for sign/verify operations, we used the Ed25519 algorithm, with 128-bits security; for (cryptographic) hashing we used the SHA-256 algorithm with 128-bits security; for encryption, decryption and session keys generation we used the Curve25519 algorithm with 128-bits security (256-bits keys). We thus achieve a security of more than 112 bits, in compliance with the current NIST standards [53] for 2016-2030. For the simplicity of the implementation, we consider that the IMS generates the keys (Step 7) and that no context is sent along with the photo (Steps 3 and 10).

### B. Dataset

We relied on the Yahoo Flickr Creative Commons 100 Million (YFCC100m) dataset [72] that contains the metadata of 100 million photos from the Flickr photo hosting website. More specifically, we extracted an unbiased sample of 20k photos (we drew photo IDs uniformly at random, without replacement, by using Python built-in random number generator with a seed of 0; we skipped the files that were no longer available). For each selected photo, we downloaded its full-resolution version from Flickr, and we filtered out the photos for which the size of the photo after a load and save operation differed from the original size by more than 5% (see Figure 4a). Our final dataset contained 17,257 photos and is available at https://www.dropbox.com/sh/xjqpguxad354h4r/AABXaXw0gNZVvj_lPqOj776La?dl=0.

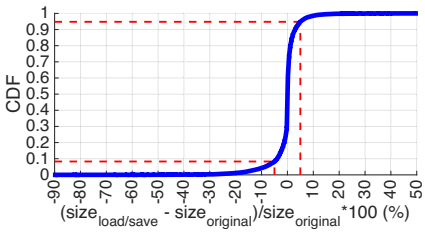
We computed statistics related to the sizes of the photos and to the faces that appear on them. In our final dataset, the average number of faces in a photo is $1.0$ (with a standard deviation of 3.7), and the maximum number of faces in a photo is 230. 61.2% of the photos do not contain any face, 20% of the photos contain exactly one face and 18.8% contain more than one face. Figure 4c illustrates the CDF of the number of faces per photo. Faces are generally small in size, compared to the actual photo: a face represents, on average, $1.1 \pm 3.3$ % of the photo size (with a maximum of 77.2 %), whereas the average size of all faces represents $3.0 \pm 5.6$ % of the photo size. As for the sizes of the photos, there is substantial variation (as can be observed in Figure 4b), the average photo size is $2.1 \pm 2.4$ MB and the maximum is 25.2 MB.

### C. Experimental Set-Up

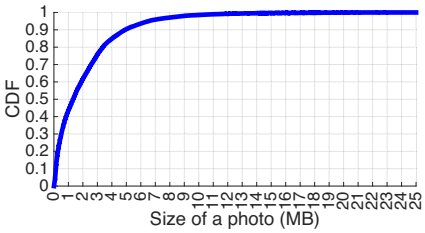
We evaluate the scenario where a user, Alice, wants to upload a photo in which potentially other people appear and we assume Alice wants all these people to appear in the photo–thus all faces are asked for consent. We do not consider the lookup operation in our evaluation (Step (6) (Figure 3)). Hence, we generically refer to a consenter by the name of Bob, for photos that contain at least one face. We perform the photo upload and grant consent (for all appearing faces) operations for all the photos in our dataset, sequentially. In Step (10) (Figure 3), we configured the CMS to provide Bob with a scaled version of the original background image with a maximum width of 1000 pixels, keeping the same image aspect ratio, quality and metadata as the original. We made use of one standard computer (Intel i7 CPU, 2.8 GHz, 8GB RAM) with Mac OS v.10.12.5. We did not use any optimization for Intel processors. The implementation of a ConsenShare prototype on Android is left to future work.

### D. Experimental Results

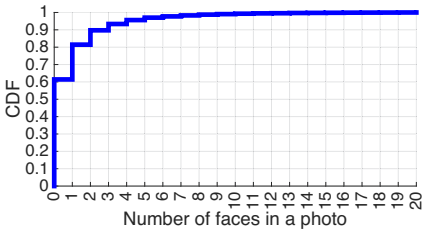
We present here the bandwidth and CPU requirements of our system for the upload photo and grant consent (from all parties) operations. All the results that we present are upper bounds of the real bandwidth and CPU consumption, as face detection is done much faster, some of the users might be blurred out (and thus not asked for consent), and more advanced image transformation techniques can be used (such as JPEG transmorphing [84] to reduce bandwidth consumption).




(a) CDF of the difference in photo sizes after the load/save operation. Only photos between the red dashed bars (within 5% of the original size) were kept, *i.e.,* 86.4% of the 20k downloaded photos.

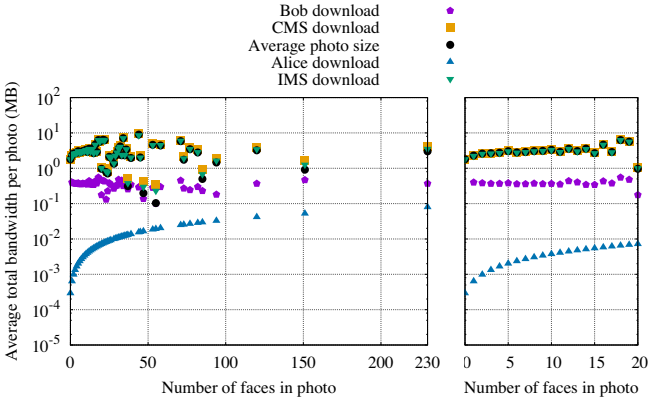



(b) CDF of the sizes of the photos after the load/save operation.

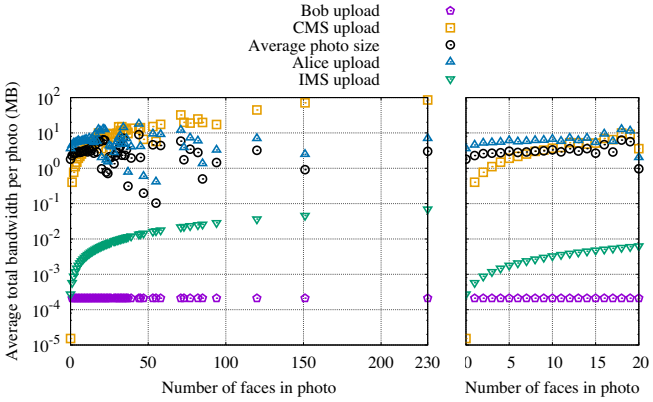



(c) CDF of the number of faces per picture, for pictures containing less or equal to 20 faces (99.7% of the photos.)

Fig. 4. Dataset statistics.




(a) Download bandwidth consumption (logscale)




(b) Upload bandwidth consumption (logscale)

Fig. 5. Average per-photo total bandwidth consumption for the uploader (Alice), the CMS, the IMS and the consenter (for the same photo, we consider the average bandwidth for one consenter, Bob). We illustrate these (y-axis) for different categories of photos, based on the number of faces they contain (x-axis). Note that for Bob, total upload and download bandwidth is 0 for photos that contain no face.

| Metric | Alice (uploader) | | CMS | | IMS | | Bob (consenter) | |
|---|---|---|---|---|---|---|---|---|
| | Upload | Download | Upload | Download | Upload | Download | Upload | Download |
| Total bandwidth (MB) | $4.2 \pm 4.8$ | $0.0007 \pm 0.0013$ | $0.4 \pm 1.4$ | $2.1 \pm 2.4$ | $0.0006 \pm 0.001$ | $2.1 \pm 2.4$ | $0.0001 \pm 0.0001$ | $0.2 \pm 0.3$ |
| Relative bandwidth overhead (%) | $101.0 \pm 4.7$ | $0.2 \pm 0.4$ | $33.1 \pm 135.0$ | $1.7 \pm 5.4$ | $0.1 \pm 0.4$ | $99.8 \pm 3.7$ | $0.02 \pm 0.07$ | $13.8 \pm 26.8$ |

TABLE I. AVERAGE PER-PHOTO TOTAL BANDWIDTH REQUIREMENTS AND RELATIVE BANDWIDTH OVERHEAD FOR THE UPLOADER (ALICE), THE CMS, THE IMS AND THE CONSENTER (FOR THE SAME PHOTO, WE CONSIDER THE AVERAGE BANDWIDTH FOR ONE CONSENTER, BOB). WE COMPUTE THE BANDWIDTH OVERHEAD RELATIVE TO THE BASELINE SCENARIO WHERE ALICE UPLOADS THE ORIGINAL PHOTO DIRECTLY TO THE CMS.

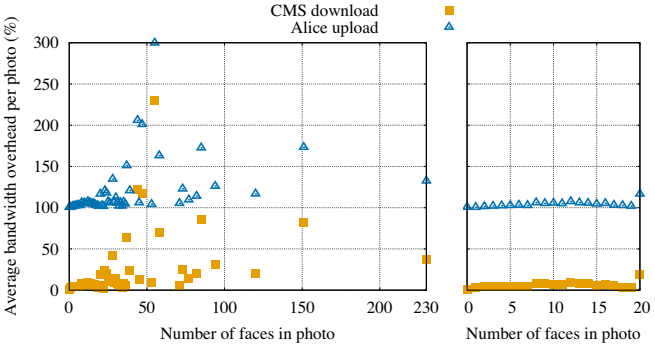
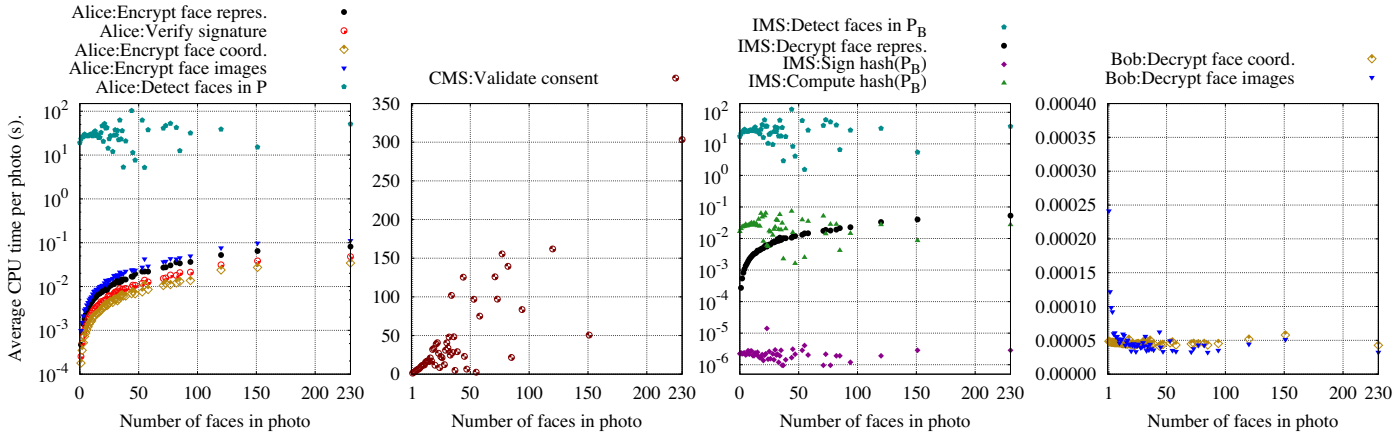



Fig. 6. Average per-photo relative bandwidth overhead of the uploader (Alice) and the CMS (download). These are computed with respect to the baseline scenario where Alice uploads the original photo directly to the CMS and expressed as a percent of the original photo size. We illustrate these (y-axis) for different categories of photos, based on the number of faces they contain (x-axis).

| | Operation | CPU time (s) $avg \pm stdev$ |
|---|---|---|
| Alice (uploader) | Detect faces in $P$ (1) | $21.5 \pm 21.3$ |
| | Encrypt face representations (4) | $4.7 \times 10^{-4} \pm 1.5 \times 10^{-3}$ |
| | Verify signature (9) | $2.6 \times 10^{-4} \pm 8.7 \times 10^{-4}$ |
| | Encrypt face coordinates (9) | $1.8 \times 10^{-4} \pm 6.2 \times 10^{-4}$ |
| | Encrypt face images (9) | $7.5 \times 10^{-4} \pm 2.6 \times 10^{-3}$ |
| CMS | Validate consent (11) | $1.3 \pm 5.4$ |
| IMS | Detect faces in $P_B$ (2) | $19.6 \pm 23.6$ |
| | Compute $hash(P_B)$ (2) | $0.02 \pm 0.02$ |
| | Sign $hash(P_B)$ (2) | $2.3 \times 10^{-6} \pm 3.4 \times 10^{-6}$ |
| | Decrypt face representations (5) | $2.8 \times 10^{-4} \pm 9.5 \times 10^{-4}$ |
| Bob (consenter) | Decrypt face coordinates (10) | $1.9 \times 10^{-5} \pm 2.5 \times 10^{-5}$ |
| | Decrypt face images (10) | $6.6 \times 10^{-5} \pm 5.0 \times 10^{-4}$ |

TABLE II. AVERAGE PER-PHOTO CPU TIMES IN SECONDS FOR THE UPLOADER, THE CMS, THE IMS AND A CONSENTER (FOR THE SAME PHOTO, WE CONSIDER THE AVERAGE TIME FOR ONE CONSENTER, BOB).

*1) Bandwidth:* We compute the average total bandwidth consumption in MB (on upload and on download) for one photo–for all the photos in our dataset, for each of the four entities: Alice, the CMS, the IMS, and one consenter, Bob. We refer to a baseline case where Alice directly uploads the photo to the CMS (providing *no privacy* for Bob). With respect to this baseline case, we compute the average *bandwidth overhead* (in MB) for one photo, which equals the total bandwidth from which the size of the original photo is subtracted (for Alice upload and CMS download) and, for the other cases, simply the total bandwidth. We also refer to the *relative bandwidth overhead*, expressed in percent, relative to the original photo




(a) CPU time for Alice (logscale) (b) CPU time for the CMS (c) CPU time for the IMS (logscale) (d) CPU time for Bob

Fig. 7. Average per-photo CPU times in seconds for different operations performed by the uploader, the CMS, the IMS and by a consenter (for the same photo, we consider the average time for one consenter, Bob). Note that the operations *Alice:Encrypt face repres.*, *Alice:Verify signature*, *Alice:Encrypt face coord.*, *Alice:Encrypt face images*, *CMS:Validate consent*, *IMS:Decrypt face repres.*, *Bob:Decrypt face coord.*, *Bob:Decrypt face images* take 0 CPU time when the photo contains no face.

size (dividing the bandwidth overhead by the original photo size). The average bandwidth requirements are presented in Table I. Notably, the average total bandwidth consumption for Alice on upload is $4.2 \pm 4.8$MB (roughly twice the original photo size – because she sends the background image twice[7]), the average total bandwidth consumption for the CMS and for the IMS on download is $2.1 \pm 2.4$MB and $2.1 \pm 2.4$MB, respectively (roughly the original photo size); the average total bandwidth consumption for the CMS on upload is $0.4 \pm 1.4$.[8] The other cases present negligible bandwidth consumption. Note that in a real system, the CMS upload cost could be substantially reduced by returning an even lower version of the background image to Bob. Figures 5 and 6 illustrate the total bandwidth consumptions (for all entities) and the relative bandwidth overheads (for Alice on upload and for the CMS on download), detailed for categories of photos containing a certain number of faces. Although there is some slight increase of bandwidth consumption w.r.t. to the number of faces in a photo (*e.g.,* for Alice on upload), this is negligible and most such increases can actually be due to an increase in photo size.

*2) CPU time:* We compute the average CPU time in seconds for one photo, for various operations, which we enumerate in Table II. Clearly the most expensive operation is the face detection performed by Alice (on the original photo) and by the IMS (on the slightly smaller sized background image for verification) with an average CPU time of $21.5 \pm 21.3$ s and $19.6 \pm 23.6$ s, respectively. We did not notice any pattern with the number of faces in the photo for these operations, but there is a noticeable increasing pattern with the photo size. Thus, a simple optimization of scaling down the photos when performing face-detection would drastically reduce this time, as shown in practice (*e.g.,* Amos et al [7] mention a runtime less than 0.1s and Taigman et al [71] mention a runtime of 1s per photo, for images from the Labeled faces in the wild dataset [36]). The validate consent operation (Step (11) (Figure 3)) – which includes face detection on all of the face photos in one photo for validation purposes and is performed by the CMS – takes, on average, $1.3 \pm 5.4$ s and is, as it can be seen in Figure 7, highly dependent on the number of faces appearing in the photo. However, even with 230 faces in a photo, this operation only takes 303s (remember that we did not use any CPU optimizations and, in practice, face recognition operations are already performed much faster by app/services like Facebook, even on phones). The CPU times for other operations are negligible.

We conclude that these results are acceptable and demonstrate the effectiveness of a system like ConsenShare.

## VIII. Incentives and Adoption

We present here the results of our survey and discuss the incentives for adoption by the different stakeholders.

### A. Survey

In order to gain insight into the individuals' perceptions of Multiple-Subject/Interdependent Personal Data (MSPD/IPD) (and of the associated privacy risks) and of a ConsenShare-like system, we conducted a survey targeted at Facebook users.

*1) Methodology:* We conducted our survey in mid-2017. We recruited participants through the Amazon Mechanical Turk (AMT) platform. To be eligible, they were required to have a minimum Human Intelligence Task (HIT) approval rate of 95% with at least 100 past approved HITs and an active Facebook account (AMT offers the possibility to specify this admission criterion). The survey took approximately 10 minutes to complete (median completion time of 8m48s) and each participant received a financial compensation of $3 in exchange for their participation. The survey was approved by our institution's ethics committee/institutional review board (application #006-2017/18.05.2017).

[7]Sending the background image to the CMS could be delegated to the IMS by providing the hash as opposed to the full $P_B$ in step (3) (Figure 3) and making the CMS request $P_B$ from the IMS directly, thus moving the load on the IMS.

[8]While this may at first glance seem high, note that recent statistics reported Flickr handles 1.68 million photo uploads per day, on average (and this is a lower bound, as it only includes photos uploaded with public visibility) [24]. At an average photo size of 2MB, this mens 6.4TB daily.

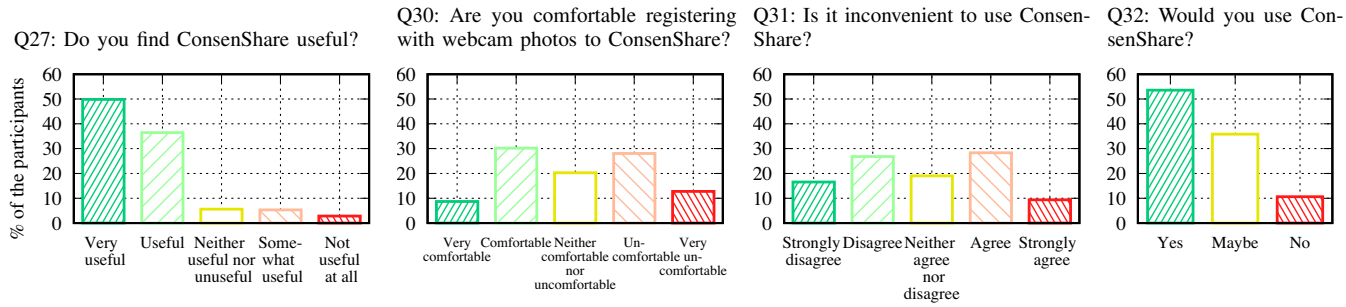



Fig. 8. Participants' responses to ConsenShare-related survey questions.

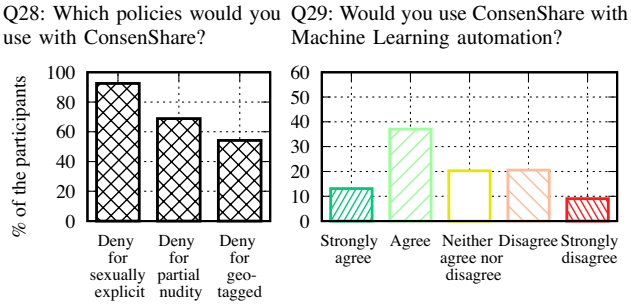



Fig. 9. Participants' responses to ConsenShare-related survey questions.

The survey was structured as follows. After the standard demographic questions (part I), we polled the participants about their perception of online data privacy for different data types and about their experience with discrimination causes by data available online (part II). We polled the participants about their sharing behaviors and those of their friends regarding multimedia content on OSNs; some of the questions were specific to sexually explicit content (part III). We polled the participants about their (un)tagging (face tags, "with tags", @ tags) behaviors and those of their friends on Facebook (part IV); the survey questions included screenshots from the Facebook website to illustrate the aforementioned tagging features. After a brief high-level description of ConsenShare, we polled the participants about their perceptions of a ConsenShare-like system. In particular, we polled them about their willingness to use such a system (part V), with a special emphasis on the consent decision process (manual, policy-based, machine learning-based). Finally, we asked the participants to confirm their agreement to save their responses and to use them in a scientific publication. The survey contained two duplicated questions in order to check the participants' attention; we used these to exclude the responses from inattentive participants from our dataset. The complete transcript of the survey and the anonymized and sanitized answers are available at https://www.dropbox.com/sh/0gr11r98qck1zmn/AAB_l-MUt09LAKAGrL7Ihi2ya?dl=0.

*2) Results:* We obtained a total of 536 complete responses. We ruled out duplicates (*i.e.,* when a participant completed the survey multiple times), the responses from inattentive participants (*i.e.,* when a participant's responses to the duplicate questions were inconsistent) and the participants that chose not to allow us to save their answers. This left us with 321 complete responses. The corresponding participant sample was balanced and diverse in terms of the participants' demographics: 53.0% of the participants were female, the participants had various areas of employments, and their ages ranged from 20 to 75 years old, with an average of 35.3 $\pm$10.4.

Our survey results indicate a potential user concern regarding the sharing of location data (86.3% of the participants), multimedia data (69.5% of the participants) and genomic data (60.8% of the participants). Furthermore, 10.3% of the participants claimed that they were victims of discrimination or prejudice based on online content about them. Of these, 33.3% reported that this happened more than once in the past. A staggering 66.7% reported that the cause was content shared by others, which highlights the gravity of MSPD/IPD privacy risks. As for the most common domains in which the discrimination or prejudice happened, 60.7% of the users referred to a job application (*e.g.,* Male, 28: *"They checked my profile online before the interview and he started making uncomfortable jokes about the game pages I like on it and a party photo."*, Male, 26: *"My employer watches my online activity and asked about that later"*), 30.3% to familial or social situations (*e.g.,* Male, 25: *"My marriage proposal got cancelled"*), 27.3% to professional situations, 15.2% to loan or mortgage and 9.1% to insurance premiums.

Regarding sharing multimedia content online, 60.1% of the participants reported that they share such content at least occasionally (a few times/month). 48.6% of the participants reported that this content contains faces of people other than themselves at least half of the times, whereas 41.4% declared that this happens sometimes, but less than half of the times. Only 10.0% said their multimedia content never features faces of others. Participants reported that their friends also share multimedia content about them at least occasionally (a few times/month) – for 45.5% of the participants.

Regarding revenge pornography, 4.1% of the participants declared that they were victims of revenge porn in the past. We also polled participants about whether other people have or had access to explicit photos of themselves. 27.4% of them declared that this is the case; of these, 48.9% declared that the person who has the photos took them with a device of their own, whereas 8.0% declared that a third party took and shared the photos. Asked whether they have or had explicit photos of someone else, 40.2% participants responded positively; of these, 41.9% reported that they are the ones that took these photos, whereas 16.3% said that a third party shared these photos with them. Many participants explained, in comments, that the photos were taken in the scenario of a (former) relationship. This illustrates that the number of potential victims of revenge pornography might be quite high.

We also polled participants about their Facebook behavior. Asked how they tag or mention their friends when posting photos or videos in which they appear, a staggering 41.1% declared that they do not tag or link their friend's profile in any way (in other words, the friend can be entirely unaware of the posted content and would thus not be able to remove/report the content). 7.2% of the participants said their friends ask them to remove photos that they have shared, at least occasionally and 11.2% said they noticed their friends contact Facebook about removing this content at least occasionally. 30.5% of the participants declared that they also ask their friends to remove content that they posted and 16.9% declared that they asked Facebook to remove such content.

Finally, we presented our framework to the participants. Aggregated participants' responses for this section of the survey are illustrated in Figures 8 and 9. Asked whether they find ConsenShare useful, 36.5% of the participants answered that this would be useful and 49.8% very useful (*e.g.,* Male, 37: "*I think this is a great way of giving control to individuals.*"; Female, 32: "*I think it's best for all parties involved [$\cdots$]*"; Female, 31: "*It would be nice to be asked if you wanted to be on the internet first, rather than letting any Tom, Dick, and Harry take your photo and post it[$\cdots$]*"). Interestingly, some participants even reported that they would be more comfortable with social platforms if such a solution was in place. Regarding the use of policies, only 3.4% of the participants declared that they would not use any policy, whereas 54.2% declared they would use a policy to deny consent for photos containing location information and 92.5% declared that they would use a policy to deny consent for photos containing explicit content. As for the use of automated decision making (via machine learning, for example), 20.2% of the participants were not sure that they would use this, whereas 50.2% declared that they would be in favor of using such a feature. Asked how comfortable they would be with a registration process similar to the one of ConsenShare (where a few photos would be required for registration), 38.9% of the participants reported being comfortable or very comfortable and 20.3% reported being undecided about this. Of the remaining participants (who reported being uncomfortable or very uncomfortable with this), some commented that this is because they do not trust webcams in general or because this system is not provided by a known company. Some also seemed confused about how the system would work, not understanding that these photos would not be stored or that face recognition is something that Facebook (the IMS here) already does on their photos. We thus recommend taking the results for this particular question with a grain of salt. In terms of the (in)convenience of first sharing the background photo and sanitized versions of the faces of the other people appearing in the photo, 43.3% of the participants found this convenient (*e.g.,* Female, 33: "*It would not be inconvenient at all! It sounds like a great solution! You still get your photo but your friends get to decide whether or not to have their faces visible – a win win!*"), whereas 19.0% were undecided. Finally, asked whether they would overall use the ConsenShare system, 53.6% of the participants answered positively (*e.g.,* Female, 32: "*It might seem inconvenient but I still favor this system and reasoning behind it.*", Male, 42: "*I think this could save a lot of head aches and keep people from being upset with one another over photo postings. This would allow anything that a person did not want online to be taken care of before it made it to a facebook page.*", Female, 34: "*I would definitely be interested in using this feature and really hope that you can develop it and get it out there. it would make me feel much more comfortable using social media and sharing images online.*") and 35.8% said that they would perhaps consider using the system.

While there are several limitations to our survey and future work we envision on this front–we discuss these in Section IX– the results indicate that there could be some need for a system like ConsenShare and a potential desire of the users to adopt it.

### B. Adoption

As we saw from our survey, a system such as ConsenShare would involve some tradeoff between the user experience as an uploader (waiting for friends to give consent before the full photo is visible), as well as the data that must be provided for registration[9], and his experience as a consenter (whose privacy would be much better protected). However, giving back control to the users' undeniably represents a substantial incentive for adoption on their part, which is further enhanced by the fact that the system is transparent and lightweight. Regarding the stakeholders, although adoption would come with some costs (*e.g.,* , increased bandwidth, deploying the infrastructure[10]), a major incentive for adoption by the CMS could be following new trends (*e.g.,* the fact that such consent-based mechanisms for MSPD/IPD data might become law-enforced) and avoiding lawsuits. Furthermore, as such features are evidently desired by their users, providing them would be good for reputation, providing a competitive advantage and likely increase the user base (and thus also the revenue from ads), as well as the shared data. This also creates new business opportunities for the CMS providers that implement the ConsenShare in-house solution and sell it to other CMS providers. As for the IMS, the obvious advantage would be a business-to-business arrangement with the various CMS providers (either transactional or subscription-based), an increased user base (thus revenue) and data (similar to that by the Facebook Connect feature). Furthermore, the IMS could monetize such features by providing them to the users in exchange for a premium fee. Note that existing services with large user databases including faces and relationships (typically OSNs) are perfect candidates to play this role, as they already have most of the data and technology needed. For instance, Facebook already performs face recognition in the background and could offer this service to different CMSs.

## IX. Discussion: Limitations and Extension

We discuss here the limitations and extensions of our work. It should be noted that this work represents a first step towards proposing a privacy-preserving generic framework for sharing MSPD/IPD data. First, it is worth mentioning that there is an

[9]Note that in the case where an existing system is used to play the role of the IMS (such as Facebook), the users' faces are already registered, making adoption quite straightforward for the users.

[10]For instance, the cost of adding the consent feature to a CMS such as Flickr would be minimal compared to the existing infrastructure: a simple interface with an IMS (*e.g.,* Facebook) and basic cryptography and image processing operations. In other words, the individual ConsenShare operations are not much different from what current CMS platforms are already doing.

inherent trade-off between the right to privacy and the right to freedom of speech. A possible middle-ground option, in the case of photos, would be to instantly publish critical content – such as photos of a mass civil action – on CMS platforms with blurred faces (similar to Google Street View and to many media outlets that already protect the identity of certain individuals, *e.g.,* minors, by blurring their faces in pictures and videos). As for the other side of the coin, the right to privacy is subject to debate for public figures/celebrities; such individuals could be detected using, for instance, Facebook's verified accounts feature, and their faces could be automatically posted, without the need for consent. Second, in the case of pictures, detection is not perfect. There is still a small margin of false positives (*i.e.,* detecting a face when none exists; if recognition matches such a "face" to a user, he can report this to the IMS, who can then improve its models upon checking the validity of this request) and false negatives (*i.e.,* not recognizing a face, which pose more problems from the privacy point of view, as these imply that a user appearing in a picture would not be recognized. Such a detected but unidentified face could be blurred out by default). These can be alleviated by asking the uploader for manual input in detecting users in the picture (similar to tagging on Facebook). Third, our current solution is centralized. In future work, we plan to design a decentralized P2P solution for the IMS and potentially the CMS. Fourth, we intend to extend our solution to incorporate interaction among users, providing them with the automatic tools to consider different options of sharing (*e.g.,* different obfuscation mechanisms, different target audiences, etc) and iteratively achieve a consensus – thus automating the social ad-hoc mechanisms user reportedly use today [70]. Fifth, the solution is CMS-specific, which means an individual user could upload the content on a different platform and just post a link to that content, by-passing the need for consent; however, this would have less impact and could even be blocked by the CMS, *e.g.,* blocking links to dubious websites. Sixth, as the issue of balance of control and the "ideal" privacy settings are culturally-dependent and not entirely law-enforced in the case of MSPD/IPD data (and regulations can differ depending on the country), our survey sample of participants is not necessarily representative of the global population; the vast majority of Mechanical Turk workers is reported to be US-based and they might have an IT-experience higher than the average; previous works have studied the profiles of Mechanical Turk workers [63]. Furthermore, it is possible that the participants' answers do not accurately indicate their true attitudes for adoption, as users' privacy attitudes and privacy decisions are not always rationally connected [4] and reported behaviors do not necessarily match the natural behavior [66]. For a more rigorous assessment of the usability and adoption potential of ConsenShare and the users' perception on the different design alternatives, in future work, we intend to run additional surveys using a fully-functional prototype, making use of the SeBIS intention scale [20], [21] to gain more insight into the participants' expertise and following specific guidelines for designing privacy/security surveys [43]. Finally, we discuss how the main building blocks in our framework can be adapted to other data, beyond photos. Note that for any type of data, there are several options to consider in the design, such as remove vs. obfuscate the data (and the available granularity); in what follows, we discuss some of the alternatives.

*a) The case of audio and video:* Considering audio and video data in our framework is a rather straightforward extension from our picture solution. Different solutions for identifying users in audio/video content have been proposed [48], [62], and various options can be used for separating the sensitive content (portions of the video in which a user appears) from the non-sensitive content: entirely cutting out the audio/video sections in which a user appears or altering the content of those sections to obfuscate that user.

*b) The case of genomic data:* Genomic privacy is a complex subject whose discussion involves many ethical and balance of control issues and closely ties to that of the privacy of others versus personal freedom. The topic of *who* are the affected parties and *how* their consent decisions should be expressed is still under debate both in the media (*e.g.,* [1]) and in the research community (*e.g.,* [9], [37], [38], [64]). There are several options that could be considered; we discuss how their implementation could be done in ConsenShare. Identity claim (at registration) in the case of genomic data would require formal identity proof (*e.g.,* an ID) and reporting of familial relationships. Detection of the involved individuals comes down to knowing these familial relationships (*e.g.,* through the IMS; Facebook already offers that option) and selecting the close relatives of the uploader. The degree of closeness for which individuals are considered as affected parties – and thus should grant consent – would be a configurable parameter of the system, which can be set according to the applicable regulation. In the most possible restrictive form of regulation, consent would be binary (yes/no) and a user would be allowed to share her whole genome only if all affected parties say yes.[11] In less restrictive regulatory options, consent can be refined to allow publication with a level of noise added to the full genome (*e.g.,* through differential privacy [73]) or after applying obfuscation techniques – typically at the SNP level – that would guarantee a certain level of privacy[12] to the affected relatives, while allowing the user who wants to share his genome some freedom. In this case, the individual desired level of privacy would be configurable for each user and the consent decision of an affected user would be the level of noise/obfuscation that the uploader must apply to his genome; the most restrictive of these – among all the relatives – would be applied to the uploader's genome before sharing.

A unique limitation in the case of genomic data is the fact that affected users (*i.e.,* unborn future relative, such as children) can appear after the user has already shared his genome with proper consent from his relatives at that time. In this case, we can offer the possibility of revoking consent, which would force the uploader to remove the data (a less than perfect solution, as the data might have already been duplicated).

*c) The case of co-location data:* Co-location data can be shared online by different means, for instance, by posting (and tagging) pictures or videos in which multiple people appear or directly tagging them in a post message. In the case of co-locations shared by using multimedia data, detection can be done as described above. In the case where co-located users

[11] Note that in this case, the CMS/IMS do not even need to see the data to determine the involved individuals.

[12] An individual's genomic privacy can be quantified using dedicated frameworks such as that proposed by Humbert et al. [38].

are directly tagged by the uploader, detection is, obviously, no longer needed. The context provided to a consenter in the case of co-location data could also include an estimation of the location privacy loss stemming from that reported co-location, as proposed by Olteanu et al. [55]. However, as co-locations introduce dependencies among the data of different users, once a co-location between Alice and Bob is shared, Alice's future location posts would also affect Bob's location privacy and vice versa. Hence the system should consider a window of influence of the correlation, by including an adjustable parameter for each user, specifying how much time after a shared co-location in which he is involved are other users required to ask for his consent to share location data.

## X. Conclusion

In this paper we propose ConsenShare, a generic framework for sharing MSPD/IPD data (*e.g.,* photos, videos, genomic data, *etc.*) with consent from all the involved individuals. ConsenShare is privacy-preserving by design not only with respect to other users of the system, but also with respect to the service providers. We implement and evaluate ConsenShare for photos and show that it is technically possible to provide users control in the sharing of photos in which their appear, while ensuring their privacy, and preserving the main features of existing CMS. In doing so, our work lays the foundation for the design of privacy-preserving sharing of MSPD/IPD data.

## Acknowledgments

The authors are grateful to Virgil Gligor, Ksenia Konyushkova and Róger Bermúdez Chacón for their insightful comments.

## References

[1] "Does your family have a right to your genetic code?" https://www.technologyreview.com/s/602946/do-your-family-members-have-a-right-to-your-genetic-code/, last visited: Aug. 2017.

[2] "Laws LJ Wood v Commissioner of Police for the Metropolis EWCA Civ 414," http://www.5rb.com/case/wood-v-commissioner-of-police-for-the-metropolis-ca/, 2009, last visited: Aug. 2017.

[3] "Consumer data privacy in a networked world: A framework for protecting intellectual property privacy and promoting innovation in the global digital economy. washington dc," https://www.whitehouse.gov/sites/default/files/privacy-final.pdf, 2013, last visited: Aug. 2017.

[4] A. Acquisti and J. Grossklags, "Privacy and rationality in individual decision making," *IEEE Security Privacy*, vol. 3, no. 1, 2005.

[5] S. Ahern, D. Eckles, N. S. Good, S. King, M. Naaman, and R. Nair, "Over-exposed?: privacy patterns and considerations in online and mobile photo sharing," in *Proc. of CHI*. ACM, 2007.

[6] "Introducing Airbnb verified ID," http://blog.atairbnb.com/introducing-airbnb-verified-id/, 2017, last visited: Aug. 2017.

[7] B. Amos, B. Ludwiczuk, and M. Satyanarayanan, "OpenFace: A general-purpose face recognition library with mobile applications," CMU-CS-16-118, CMU School of Computer Science, Tech. Rep., 2016.

[8] "Apple concept for biometric facial recognition could hint at iPhone 8," http://appleinsider.com/articles/17/03/16/apple-concept-for-biometric-facial-recognition-could-hint-at-iphone-8, 2017, last visited: Aug. 2017.

[9] E. Ayday and M. Humbert, "Inference attacks against kin genomic privacy," *IEEE Security & Privacy*, no. 5, 2017.

[10] F. Beato, I. Ion, S. Čapkun, B. Preneel, and M. Langheinrich, "For some eyes only: protecting online information sharing," in *Proc. of CODASPY*. ACM, 2013.

[11] F. Beato and R. Peeters, "Collaborative joint content sharing for online social networks," in *Proc. of PERCOM*. IEEE, 2014.

[12] A. Besmer and H. Richter Lipford, "Moving beyond untagging: photo privacy in a tagged world," in *Proc. of CHI*. ACM, 2010.

[13] G. BiczĂłk and P. H. Chia, "Interdependent Privacy: Let Me Share Your Data," in *Proc. of FC*, 2013.

[14] "Facebook can recognise you in photos even if you're not looking," https://www.newscientist.com/article/dn27761-facebook-can-recognise-you-in-photos-even-if-youre-not-looking/, last visited: Aug. 2017.

[15] R. Bost, R. A. Popa, S. Tu, and S. Goldwasser, "Machine learning classification over encrypted data." in *Proc. of NDSS*, 2015.

[16] H. Cho and A. Filippova, "Networked privacy management in facebook: A mixed-methods and multinational study," in *Proc. of CSCW*, 2016.

[17] "The OECD privacy framework," http://www.oecd.org/internet/ieconomy/privacy-guidelines.htm, 2013, last visited: Aug. 2017.

[18] E. De Cristofaro, C. Soriente, G. Tsudik, and A. Williams, "Hummingbird: Privacy at the time of Twitter," in *Proc. of S&P*. IEEE, 2012.

[19] R. Dey, C. Tang, K. Ross, and N. Saxena, "Estimating age privacy leakage in online social networks," in *Proc. of INFOCOM*, 2012.

[20] S. Egelman, M. Harbach, and E. Peer, "Behavior ever follows intention?: A validation of the security behavior intentions scale (sebis)," in *Proc. of CHI*, 2016.

[21] S. Egelman and E. Peer, "Scaling the security wall: Developing a security behavior intentions scale (sebis)," in *Proc. of CHI*, 2015.

[22] "Face recognition app taking russia by storm may bring end to public anonymity," https://www.theguardian.com/technology/2016/may/17/findface-face-recognition-app-end-public-anonymity-vkontakte, last visited: Aug. 2017.

[23] A. J. Feldman, A. Blankstein, M. J. Freedman, and E. W. Felten, "Social networking with frientegrity: privacy and integrity with an untrusted provider," in *Proc. of USENIX*, 2012.

[24] "How many public photos are uploaded to Flickr every day, month, year?" https://www.flickr.com/photos/franckmichel/6855169886, last visited: Aug. 2017.

[25] A. Frome, G. Cheung, A. Abdulkader, M. Zennaro, B. Wu, A. Bissacco, H. Adam, H. Neven, and L. Vincent, "Large-scale privacy protection in google street view," in *Proc. of ICCV*. IEEE, 2009.

[26] S. Gnesi, I. Matteucci, C. Moiso, P. Mori, M. Petrocchi, and M. Vescovi, "My Data, Your Data, Our Data: Managing Privacy Preferences in Multiple Subjects Personal Data," in *Proc. of APF*. Springer, 2014.

[27] ——, "My data, your data, our data: managing privacy preferences in multiple subjects personal data," in *Annual Privacy Forum*, 2014.

[28] N. S. Good, "Designing for informed consent: A multi-domain, interdisciplinary analysis of the technological means to provide informed consent, in order to manage users' privacy and security," Ph.D. dissertation, University of California at Berkeley, 2008.

[29] Y. Guo, L. Zhang, and X. Chen, "Collaborative privacy management: mobile privacy beyond your own devices," in *Proc. of MobiCom*. ACM, 2014.

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

[31] C. Holz, S. Buthpitiya, and M. Knaust, "Bodyprint: Biometric user identification on mobile devices using the capacitive touchscreen to scan body parts," in *Proc. of CHI*. ACM, 2015.

[32] H. Hu and G.-J. Ahn, "Multiparty authorization framework for data sharing in online social networks," in *Proc. of CODASPY*, 2011.

[33] H. Hu, G.-J. Ahn, and J. Jorgensen, "Detecting and resolving privacy conflicts for collaborative data sharing in online social networks," in *Proc. of ACSAC*. ACM, 2011.

[34] ——, "Enabling collaborative data sharing in google+," in *Proc. of GLOBECOM*. IEEE, 2012.

[35] ——, "Multiparty access control for online social networks: model and mechanisms," *IEEE TKDE*, 2013.

[36] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in un-

constrained environments," No 07-49, University of Massachusetts, Amherst, Tech. Rep., 2007.

[37] J.-P. Hubaux, S. Katzenbeisser, and B. Malin, "Genomic data privacy and security," *IEEE Security & Privacy*, no. 5, 2017.

[38] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, "Quantifying Interdependent Risks in Genomic Privacy," *ACM TOPS*, 2017.

[39] P. Ilia, B. Carminati, E. Ferrari, P. Fragopoulou, and S. Ioannidis, "Sampac: Socially-aware collaborative multi-party access control," in *Proc. of CODASPY*. ACM, 2017.

[40] P. Ilia, I. Polakis, E. Athanasopoulos, F. Maggi, and S. Ioannidis, "Face/Off: Preventing Privacy Leakage From Photos in Social Networks," in *Proc. of CCS*. ACM, 2015.

[41] H. Jia and H. Xu, "Autonomous and interdependent: Collaborative privacy management on social networking sites," in *Proc. of CHI*, 2016.

[42] H. Jégou, R. Tavenard, M. Douze, and L. Amsaleg, "Searching in one billion vectors: re-rank with source coding," in *Proc. of ICASSP*. IEEE, 2011.

[43] P. G. Kelley, "Conducting usable privacy & security studies with amazon's mechanical turk," in *Proc. of SOUPS*, 2010.

[44] A. Lampinen, V. Lehtinen, A. Lehmuskallio, and S. Tamminen, "We're in it together: interpersonal management of disclosure in social network services," in *Proc. of SIGCHI*. ACM, 2011.

[45] F. Li, J. Yu, L. Zhang, Z. Sun, and M. Lv, "A privacy-preserving method for photo sharing in instant message systems," in *Proc. of ICCSP*. ACM, 2017.

[46] C. Mallauran, J.-L. Dugelay, F. Perronnin, and C. Garcia, "Online face detection and user authentication," in *Proc. of MM*. ACM, 2005.

[47] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, "You are who you know: inferring user profiles in online social networks," in *Proc. of WSDM*. ACM, 2010.

[48] L. Muda, M. Begam, and I. Elamvazuthi, "Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques," *arXiv preprint arXiv:1003.4083*, 2010.

[49] J. Määttä, A. Hadid, and M. Pietikäinen, "Face spoofing detection from single images using texture and local shape analysis," *IET biometrics*, 2012.

[50] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proc. of CVPR*, 2015.

[51] K. Niinuma and A. K. Jain, "Continuous user authentication using temporal information," in *Proc. of SPIE*, 2010.

[52] H. Nissenbaum, *Privacy in context: Technology, policy, and the integrity of social life*. Stanford University Press, 2009.

[53] "Recommendation for Keymanagement, Part 1: General, sp 800-57 Part 1 Rev. 4." http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r4.pdf, January 2016, last visited: Aug. 2017.

[54] K. Olejnik, I. Dacosta, J. Soares Machado, K. Huguenin, M. E. Khan, and J.-P. Hubaux, "Smarper: Context-aware and automatic runtime-permissions for mobile devices," in *Proc. of S&P*. IEEE, 2017.

[55] A.-M. Olteanu, K. Huguenin, R. Shokri, M. Humbert, and J.-P. Hubaux, "Quantifying Interdependent Privacy Risks with Location Data," *IEEE TMC*, 2017.

[56] "OpenFace - Free and open source face recognition with deep neural networks," https://cmusatyalab.github.io/openface/.

[57] "PyNaCl - Python binding to the Networking and Cryptography library," https://github.com/pyca/pynacl, 2017, last visited: Aug. 2017.

[58] A. Ratikan and M. Shikida, "Privacy protection based privacy conflict detection and solution in online social networks," in *Proc. of HAS*. Springer, 2014.

[59] "Revenge porn - the ugly side of social media," Online, https://www.thetimes.co.uk/article/its-destroying-the-lives-of-17-year-olds-nbktc352n, 2017, last visited: Aug. 2017.

[60] "Revenge porn: Image-based abuse hits one in five Australians," Online, http://www.bbc.com/news/world-australia-39777192, 2017, last visited: Aug. 2017.

[61] "Military women ask Facebook to do more to stop revenge porn," http://www.refinery29.com/2017/04/149960/military-women-facebook-revenge-porn, 2017, last visited: Aug. 2017.

[62] D. A. Reynolds, "An overview of automatic speaker recognition technology," in *Proc. of ICASSP*, vol. 4. IEEE, 2002.

[63] J. Ross, L. Irani, M. Silberman, A. Zaldivar, and B. Tomlinson, "Who are the crowdworkers?: shifting demographics in mechanical turk," in *Proc. of CHI*, 2010.

[64] S. R. Savage, "Characterizing the risks and harms of linking genomic information to individuals," *IEEE Security & Privacy*, no. 5, 2017.

[65] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proc. of CVPR*, 2015.

[66] A. Sotirakopoulos, K. Hawkey, and K. Beznosov, "On the challenges in usable security lab studies: lessons learned from replicating a study on ssl warnings," in *Proc. of SOUPS*, 2011.

[67] A. C. Squicciarini, M. Shehab, and J. Wede, "Privacy policies for shared content in social network sites," *Proc. of VLDB*, 2010.

[68] A. C. Squicciarini, M. Shehab, and F. Paci, "Collective privacy management in social networks," in *Proc. of WWW*. ACM, 2009.

[69] J. M. Such and N. Criado, "Resolving multi-party privacy conflicts in social media," *IEEE KDE*, 2016.

[70] J. M. Such, J. Porter, S. Preibusch, and A. Joinson, "Photo privacy conflicts in social media: A large-scale empirical study," in *Proc. of CHI*, 2017.

[71] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Web-scale training for face identification," in *Proc. of CVPR*, 2015.

[72] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li, "YFCC100M: The new data in multimedia research," *Commun. ACM*, 2016.

[73] F. Tramèr, Z. Huang, J.-P. Hubaux, and E. Ayday, "Differential privacy with bounded priors: reconciling utility and privacy in genome-wide association studies," in *Proc. of CCS*, 2015.

[74] "Engineering Safety with Uber's real-time ID check," https://eng.uber.com/real-time-id-check/, 2017, last visited: Aug. 2017.

[75] "Selfies and Security," https://newsroom.uber.com/securityselfies/, 2017, last visited: Aug. 2017.

[76] H. Wang, X. Bao, R. R. Choudhury, and S. Nelakuditi, "Insight: Recognizing humans without face recognition," in *Proc. of HotMobile*. ACM, 2013.

[77] H. Wang, X. Bao, R. Roy Choudhury, and S. Nelakuditi, "Visually fingerprinting humans without face recognition," in *Proc. of MobiSys*. ACM, 2015.

[78] P. Wijesekera, A. Baokar, L. Tsai, J. Reardon, S. Egelman, D. Wagner, and K. Beznosov, "The feasibility of dynamically granted permissions: Aligning mobile privacy with user preferences," *arXiv preprint arXiv:1703.02090*, 2017.

[79] "Windows Hello: They can guess your password - not your face," https://www.microsoft.com/en-us/windows/windows-hello, 2017, last visited: Aug. 2017.

[80] P. Wisniewski, H. Lipford, and D. Wilson, "Fighting for my space: Coping mechanisms for sns boundary regulation," in *Proc. of SIGCHI*. ACM, 2012.

[81] D. J. Wu, T. Feng, M. Naehrig, and K. Lauter, "Privately evaluating decision trees and random forests," *PoPETs*, 2016.

[82] H. Xu, "Reframing privacy 2.0 in online social network," *U. Pa. J. Const. L.*, 2011.

[83] L. Yuan, "Privacy-friendly photo sharing and relevant applications beyond," Ph.D. dissertation, EPFL Switzerland, 2017.

[84] L. Yuan and T. Ebrahimi, "Image transmorphing with jpeg," in *Proc. of ICIP*. IEEE, 2015.

[85] L. Yuan, J. Theytaz, and T. Ebrahimi, "Context-dependent privacy-aware photo sharing based on machine learning," in *Proc. of IFIP SEC*. Springer, 2017.

[86] M. Ziad, A. Alanwar, M. Alzantot, and M. Srivastava, "Cryptoimg: Privacy preserving processing over encrypted images," *arXiv preprint arXiv:1609.00881*, 2016.