I-Pic: A Platform for Privacy-Compliant Image Capture

Conference Paper · June 2016

DOI: 10.1145/2906388.2906412

CITATIONS
47

READS
111

9 authors, including:

Rijurekha Sen
Indian Institute of Technology Delhi
23 PUBLICATIONS **897** CITATIONS
SEE PROFILE

Seong Joon Oh
Naver
34 PUBLICATIONS **594** CITATIONS
SEE PROFILE

Rodrigo Benenson
Max Planck Institute for Informatics
57 PUBLICATIONS **9,249** CITATIONS
SEE PROFILE

I-Pic: A Platform for Privacy-Compliant Image Capture

Paarijaat Aditya, Rijurekha Sen and Peter Druschel
Max Planck Institute for Software Systems (MPI-SWS)

Seong Joon Oh, Rodrigo Benenson, Mario Fritz and Bernt Schiele
Max Planck Institute for Informatics

Bobby Bhattacharjee
University of Maryland

Tong Tong Wu
University of Rochester

Abstract

The ubiquity of portable mobile devices equipped with built-in cameras have led to a transformation in how and when digital images are captured, shared, and archived. Photographs and videos from social gatherings, public events, and even crime scenes are commonplace online. While the spontaneity afforded by these devices have led to new personal and creative outlets, privacy concerns of bystanders (and indeed, in some cases, unwilling subjects) have remained largely unaddressed.

We present I-Pic, a trusted software platform that integrates digital capture with user-defined privacy. In I-Pic, users choose a level of privacy (e.g., image capture allowed or not) based upon social context (e.g., out in public vs. with friends vs. at workplace). Privacy choices of nearby users are advertised via short-range radio, and I-Pic-compliant capture platforms generate edited media to conform to privacy choices of image subjects.

I-Pic uses secure multiparty computation to ensure that users' visual features and privacy choices are not revealed publicly, regardless of whether they are the subjects of an image capture. Just as importantly, I-Pic preserves the ease-of-use and spontaneous nature of capture and sharing between trusted users. Our evaluation of I-Pic shows that a practical, energy-efficient system that conforms to the privacy choices of many users within a scene can be built and deployed using current hardware.

1. INTRODUCTION

The spontaneity afforded by mobile devices with cameras have led to new creative outlets that continue to have broad and lasting social impact. As every facet of event reporting, ranging from personal journals to war correspondence, is transformed, however, there is a growing unease about the dilution of privacy that inevitably accompanies digital capture in public, and in some cases, private fora. This paper describes I-Pic, a platform for *policy-compliant* image capture, whereby captured images are automatically edited according to the privacy choices of individuals photographed. I-Pic's design was motivated by a user-study, described in Section 2, which found that:

MobiSys'16 June 25-30, 2016, Singapore, Singapore
ACM 978-1-4503-4269-8/16/06.
http://dx.doi.org/10.1145/2906388.2906412.

Capture policies should be individualized: Privacy concerns vary between individuals. Even in the same situation, different subjects have different preferences. This finding motivated I-Pic to preclude options that impose blanket or venue specific policies [1, 2, 3].

Policies should be situational: Study subjects stated consent to be photographed at certain times, places, events, or by certain photographers, but would make different choices in other circumstances. This motivated I-Pic to not impose a static policy per individual [4], and to avoid solutions that require prior arrangements between specific subjects and photographers (whitelisting or blacklisting).

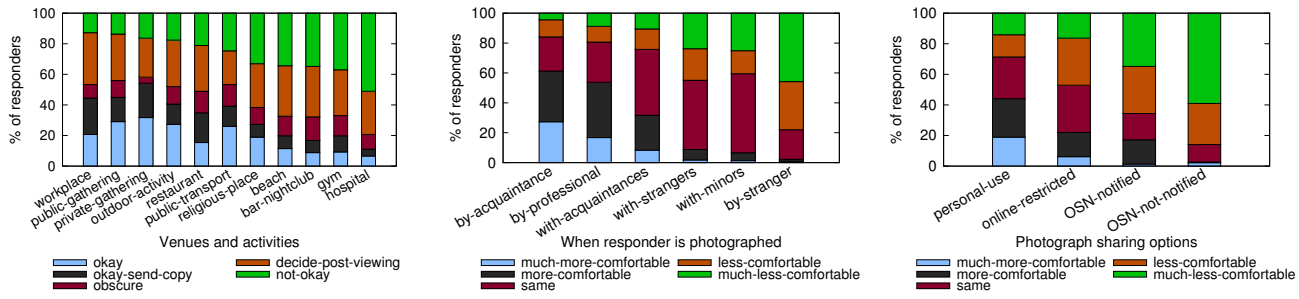
Compliance by courtesy is sufficient: An overwhelming majority of our subjects stated that they would choose to comply with the privacy preferences of friends and strangers, especially if doing so didn't interfere with the spontaneity of image capture. I-Pic provides such a platform but is not meant to stop determined users from taking pictures against the wishes of others; indeed, these users could simply use a non-I-Pic compliant device.

Consider a strawman system where mobile devices broadcast their owner's privacy preferences via Bluetooth. Without additional information, a camera would have to edit the image according to the most restrictive policy received, even if the corresponding person does not appear in the image at all! To be practical, polices must be accompanied by a visual signature so that a camera can associate a person captured in an image with a policy.

However, Bluetooth transmissions can cross walls, which would create a serious privacy problem if visual signatures were broadcast in the clear: Next-door neighbors could identify persons whom they have never seen or photographed! To avoid this problem, I-Pic relies on secure multiparty computation (MPC) to ensure that a capture device learns only a person's privacy choice, and only if that person was captured; otherwise, neither side learns anything.

User studies and privacy requirements inform the architectural components of I-Pic: Users advertise their presence over BLE (Bluetooth Low Energy): these broadcasts are received by I-Pic-compliant capture platforms. When an image is taken, the platform determines if any of the captured people match the visual signatures of nearby users using MPC. If there is a match, the platform learns the policy and edits the image accordingly, e.g., by occluding the person's face. To maintain the responsiveness of image capture, unedited images are shown to the photographer immediately, but cannot be shared until the image is processed in the background.

After presenting the results of our online survey in Section 2, we describe the main technical design of I-Pic in Section 4, along with prior work in face recognition and cryptography we build on. Next,



(a) Choices based on physical situations (b) Choices based on social situations (c) Choices based on use of images

Figure 1. Variety in privacy preferences under similar physical, social and image usage scenarios

we presents results of an experimental evaluation in Section 5. We discuss related work in Section 6 and conclude in Section 7.

2. ONLINE SURVEY

I-Pic's design was informed by an online survey designed to provide a broader perspective on personal expectations and desires for privacy. The survey, and experiments with I-Pic, were conducted with user consent under an IRB approval from the University of Maryland. The survey included an optional section on user demographic, including gender, age, and ethnicity.

We publicized the survey on mailing lists and online social networks on November 10th, 2015. The survey is available online at `http://goo.gl/forms/6tGG0YmFFG`, and the results here present a snapshot of all responses collected on December 4th, 2015. As of this date, there were 227 responses, with 208 responders also answering the demographic questions. Respondents represented 32 countries. The age distribution is shown in Table 1.

Age group	Fraction of participants
less than 20 years	9.2%
20 - 30 years	56.6%
30 - 40 years	25.1%
40 - 50	4.8%
more than 50 years	3.9%
Unspecified	0.4%

Table 1. Age groups of survey participants

Questions in the survey envisioned different venues and activities and presented participants with different privacy options: (a) agree to be captured in any photograph, (b) agree, but would like a copy of the image, (c) please obscure my appearance in any image, (d) can decide my preference only after viewing the photo, or (e) do not wish to be captured in any photograph. Participants were asked to choose the privacy action they considered most appropriate for each scenario (Figure 1(a)). To help visualize a common scenario and to provide perspective for others, participants were shown an image of people on a platform waiting to board a train, some with faces clearly visible. The survey also gauged individual's level of comfort depending on their relationship to the photographer or the other subjects in the photograph (Figure 1(b)). Finally, we asked how potential uses of an image influence responders' level of comfort with being captured (Figure 1(c)).

In Figure 1(a), the x-axis is sorted by the percentage of responders who chose the most private action of "do not wish to be captured", increasing from left to right. Our results show a mix of privacy concerns for different scenarios. In Figures 1(b) and 1(c), the x-axis is sorted by the percentage of responders who were much less comfortable with photography, increasing from left to right. Once again, for these social situations or image usage scenarios, the privacy concerns of responders is not uniform. *These results demonstrate the necessity of diversity in privacy policy, and argue against venue based policies that cannot be customized for individuals [2, 5].*

Unsurprisingly, privacy preferences are not unanimous for any scenario; there are, however, trends. Responders tend to be more restrictive in venues such as beaches, gyms and hospitals (in Figure 1(a)); with strangers in a social situation (in Figure 1(b)); and when images can potentially be shared online (in Figure 1(c)). These trends can be useful as they suggest default policies appropriate for different situations.

Number of privacy preferences	Fraction of participants
1	12.7%
2	27.8%
3	32.2%
4	19.4%
5	7.9%

Table 2. Variety in privacy preferences for same person

Table 2 shows the percentage of responders versus the number of different privacy choices for each responder. The table shows that individuals prefer different privacy choices depending on the given situation. *This finding illustrates the utility of context-specific policies, and demonstrates the shortcomings of individualized hard-coded policies, e.g., bar-codes on clothing [4].*

The survey asked whether responders cared about *by-stander* privacy when respondents themselves capture images. An overwhelming majority (96.47%) answered in the affirmative, motivating a system such as I-Pic. About a quarter (28%) agreed if the overhead of the solution was low; another quarter (26%) agreed if the aesthetics of images remain good.

Respondent Selection Bias The survey was voluntary and anonymous. The URL for the survey was advertised on mailing lists and social networks used by the authors and their friends, leading to a bias in how respondents learned about the survey. However, we believe that the results presented here still have merit as they represent views across different age groups and ethnicities. The results overwhelmingly support the thesis that users often desire privacy from digital capture in social situations, and further that "one-size-fits-all" solutions to image privacy are not effective. Moreover, as photographers, the responders overwhelmingly consider bystander privacy to be important. These observations inform I-Pic's design, described next.

3. I-PIC ARCHITECTURE

3.1 I-Pic overview

Figure 2 shows I-Pic's major components and their interaction. The two types of principals in the system are *bystanders* or users

who may be photographed, and *photographers* who capture images. Both are assumed to operate an I-Pic-compliant *platform*. Associated with each principal is a cloud-based *agent* to which the principals offload compute-intensive tasks. The photographer is associated with a *Capture Agent*; each bystander is associated with a *Bystander Agent*. We note that agents are logical constructs; functions provided by the agent can be implemented within mobile devices should I-Pic be used without wide-area connectivity.

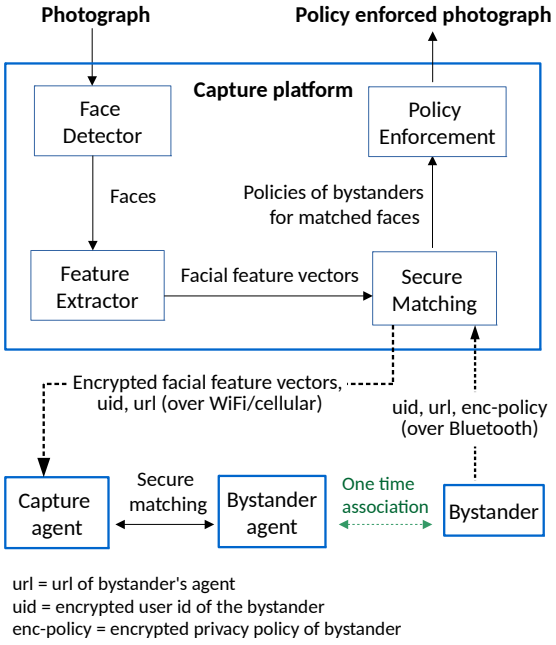


Figure 2. I-Pic major components

I-Pic requires a one-time *Association* protocol between users and their agent. Users *periodically broadcast* their presence using BLE. Once an image is captured, the *Face Detection*, *Feature Extraction*, and *Secure Matching* protocols are executed. If a user is identified, the capture platform uses the *Policy Enforcement* protocol to modify the photograph as requested. We describe these sub-protocols next.

Association: Users select an agent as a proxy and provide it with photographs, which are used to train an SVM classifier for face recognition. A user trusts her agent not to leak her visual signature. The association protocol also exchanges a master key between agent and user's device, which is used to generate session keys in the future.

Next, users initialize their privacy profile, which is locally stored on their device, by choosing relevant contexts based on location (e.g. office, home, gym, bar/restaurant, public spaces) and time (work hours, off-work hours), and by choosing an appropriate action for each context (agree to appear with face, blur face).

Periodic Broadcast: Users periodically broadcast a encrypted policy that specifies how to treat the user's picture if she appears in a photograph. This broadcast also includes sufficient information to identify the user's agent. The policy is encrypted with a session key generated using the current time (divided into 15-minutes epochs) and the master key exchanged with the user's agent.

Capture platforms receive and cache policies. Once a photograph is captured, if a user is identified, then the associated policy can be decrypted.

Secure Matching: Upon image capture, the platform detects and tries to recognize faces. These components leverage our prior work in face detection [6] and facial feature extraction [7], as detailed in Section 4.1.

The capture platform encrypts the extracted features and uploads them to its agent, along with the network identifiers of all bystander agents that it has received as broadcast recently. The *Capture Agent* and the *Bystander Agent* compare extracted features and a bystander's classifier weight vector by implementing a secure dot-product protocol [8] followed by a secure threshold comparison protocol based on garbled circuits [9]. If the threshold passes, then the session key used to encrypt user's policy is revealed to the capture platform.

Policy Enforcement: When granted a session key for a user, the capture platform decrypts the corresponding user's privacy policy and performs the action requested. Our current implementation only supports face obfuscation, which we implement using the OpenCV library. More sophisticated techniques exist. For instance, it is possible to morph a face into another face [10] instead of blurring it. Furthermore, it is also possible to remove an entire body from an image and extrapolate the background so that the removal is not obvious [11]. While such advanced image processing techniques are not the subject of this paper, I-Pic can take advantage of them.

If a captured face cannot be matched against any bystander, but all advertised policies have been evaluated, I-Pic defaults to blurring the face. This protects the privacy of bystanders who either do not own a smart device or are not I-Pic users.

Similarly, all unmatched faces are blurred if the identification protocol does not complete for some policies, likely due to lack of network connectivity. The platform maintains an encrypted copy of the original image, which can be used to release an unblurred face in the original image as the protocol completes in the future.

3.2 Threat model

I-Pic's cryptographic protocols ensure that a non-compliant capture device cannot learn the feature vectors of a bystander who does not appear in a captured image. For privacy policies of bystanders to be correctly applied, the capture platform on users' devices is assumed to implement the I-Pic protocol correctly. Third-party applications installed on users' devices are untrusted.

Users of capture devices may be able to bypass I-Pic by "rooting" their device; a different implementation could integrate I-Pic into the device firmware or implement the protocol on a trusted hardware platform, thus raising the bar for bypassing I-Pic's privacy protection. We dismissed this approach, because uncooperative photographers could in any case use a non-I-Pic compliant camera. Our goal instead is to enable cooperative photographers to respect bystander's privacy wishes in an unobtrusive manner, without introducing new attack vectors. We believe that most users welcome the ability to automatically comply with bystander's wishes, as it enables them to take pictures freely, without worrying whether they might offend others. This was also observed in our online survey (Section 2), where 96% of the participants indicated that they cared about bystanders' privacy.

The *Bystander Agent* must be trusted by the bystander not to leak her visual signature. The *Capture Agent*, on the other hand, does not have access to either the users' visual signature stored on the *Bystander Agent* or the features vectors extracted by the capture device. However, *Bystander Agent* and *Capture Agent* are assumed not to collude, else they could joinly extract the feature vectors of people captured in an image. *Capture Agent* is additionally expected to construct the garbled circuit used for secure threshold comparison (described in Section 4.2) accurately.

Cloud agents learn when an I-Pic compliant device captures an image, and the *Capture Agent* learns the IP address of that cam-

era device (Technically, both could be spoofed since the request may use an identifier without capturing an image, and the source IP address in a request could be that of a forwarding relay). I-Pic protocols are designed to ensure that the cloud agents do not learn if a user appears in an image, or the user's current context or policy. The following Section 4 describes the I-Pic protocols in detail.

4. I-PIC DESIGN

Next, we describe the design of I-Pic in more detail. Figure 3 shows the I-Pic workflow in normal operation.

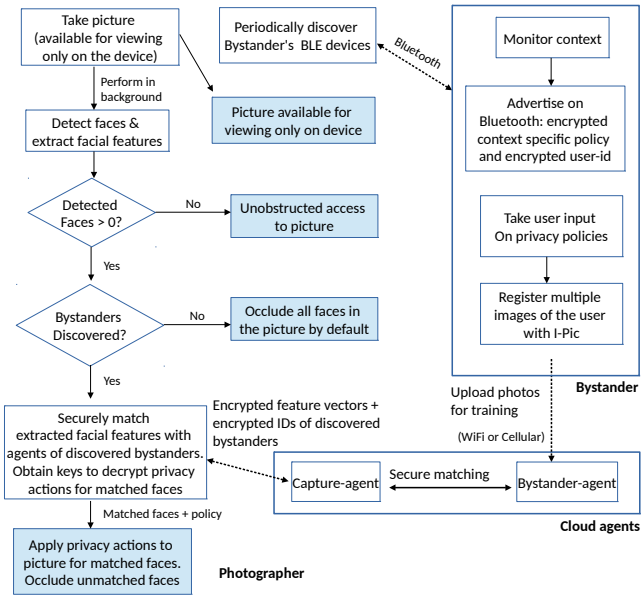


Figure 3. I-Pic workflow

I-Pic compliant devices broadcast their encrypted $(userid, policy)$ pairs periodically. I-Pic compliant capture devices additionally discover other Bluetooth devices periodically and add any received pairs to a local cache of nearby users. The entries are flushed from the cache when a device's broadcast has not been received for 10 minutes.

When an image is captured, I-Pic intercepts the raw image data. The captured image is available for viewing immediately but cannot be shared until the image is processed. A background task runs the vision pipeline described below in Section 4.1 to detect faces and extract feature vectors for each. Next, for each feature vector extracted from the image, the background task performs the secure matching protocol described below in Section 4.3 to determine if it matches with the registered classifiers of any of the bystanders in the cache, and decrypts the policies of any matching bystanders.

Finally, the I-Pic background task edits the image according to the policies of the users captured in the image. By default, any face detected in the image that did not match the signature of a bystander is occluded. This conservative choice errs on the side of privacy in case of a bystanders who does not carry a mobile device or does not use I-Pic, whose BLE broadcast was not received, or whose visual signature did not match due to a false negative of the face recognition.

4.1 Image processing

The goal of I-Pic's image processing is to identify people captured in the image, extract visual signatures for each person, and match these signatures with those advertised by nearby bystanders.

Detecting and recognizing people in images is an active area of research in computer vision. The current I-Pic prototype relies on face recognition as a well-understood and natural technique for detecting and recognizing people. More general techniques for people detection and recognition based on full-body visual signatures can be integrated into I-Pic in the future.

In the following, we briefly describe I-Pic's face detection, feature extraction, and face recognition pipeline.

Face detection: I-Pic must detect faces with high recall, ensuring that bystanders' faces are detected with high probability regardless of size, focus, pose, angle, lighting, or partial occlusion. Unlike the primary subjects of an image, bystanders are not posing for the camera, may be in the background, poorly lit, or out of focus, which makes their detection challenging.

We use the open source HeadHunter [6] prototype developed as part of our prior work on face detection. HeadHunter achieves face detection recall of $\sim$95% on standard image datasets like the Annotated Faces in the Wild (AFW) [12]. For I-Pic, we ported HeadHunter to a mobile tablet with a GPU, as described in Section 5. As we will show in Section 5.4.3, HeadHunter is superior to other face detectors available for mobile platforms.

Feature extraction: We use the state of the art person recognition method from our prior work [7]. Unlike typical face recognition systems that can recognize only the frontal faces, our person recognition system has been trained to generalize across head pose by utilizing hairstyle and context information. Due to this generalization, it outperforms other cutting-edge face recognition systems in a social media photo setting [13], where individuals often do not pose for the camera. Since I-Pic aims at identifying bystanders, this person recognition system is highly relevant.

Our person recognition system is based on a convolutional neural network (AlexNet [14]) pretrained on the ImageNet [15] classification task, and fine-tuned for the person identification task on People In Photo Albums (PIPA [13]), a large database of people in social media photos. While our prior work [7] uses five different body regions (face, head, upper/full body, and scene) to maximize the performance, we only extract features from the face region, and denote this cue as FNet.

Given a face, the original FNet extracts a 4096-dimensional feature vector. To ensure the efficiency of the secure matching algorithm, which is inversely proportional to the number of dimensions, we reduce this feature vector to 128 dimensions. We found that using the neural network itself for dimensionality reduction results in a smaller drop in overall recognition accuracy than using Principal Component Analysis. Specifically, we insert a 128-dimensional fully connected layer before the last layer in the AlexNet, randomly initialize the weights, and tune it using Stochastic Gradient Descent. Our FNet features are extracted from this 128-dimensional layer after forward passing Headhunter face detections through the network. All the training and feature extraction in neural networks are done using the open source deep learning framework Caffe [16].

Face recognition: When a user registers, I-Pic extracts FNet features from the set of portraits he or she provides. Per-user SVM classifiers are then trained on the FNet features, where positive examples consist of the portraits provided by the corresponding user, and negative examples from the other users and $\sim$12K celebrity faces in the Labeled Faces in the Wild dataset (LFW) [17]. On average, there are $\sim$15 positive examples per user, captured with different viewpoints and facial expressions. Users may subsequently provide additional images for training, for instance, if they start to wear glasses or grow a beard. The liblinear [18] package has been used to train the SVMs.

In normal operation, HeadHunter detects faces in captured images, and the corresponding FNet features are extracted. I-Pic compares the feature vector of each detected face against the trained SVM classifiers of each bystander using a dot product computation. If the dot product is above a certain threshold, the classifier indicates a match. To ensure privacy, I-Pic computes the dot product and threshold comparison as part of a secure multiparty computation between the photographer's capture agent and each bystander's agent.

Before we describe the secure matching protocol, we briefly review the underlying crypto protocols.

4.2 Cryptographic Protocols

I-Pic composes two standard protocols to achieve secure matching: secure dot product and garbled circuits.

Secure dot product: The secure dot product protocol allows two parties, each with a private vector, to compute the vector dot product without divulging the vectors. We use the protocol described in [8], which is based on the Paillier homomorphic encryption scheme [19]. We use the notation $[\![a]\!]_{pk}$ to represent the encryption of a number a using a public key pk. The Paillier encryption scheme is additively homomorphic, i.e., given $[\![a]\!]_{pk}$ and $[\![b]\!]_{pk}$, it is possible to compute $[\![a+b]\!]_{pk} = [\![a]\!]_{pk}[\![b]\!]_{pk}$. It follows that given $[\![a]\!]_{pk}$ and an integer c, one can compute $[\![ca]\!]_{pk} = ([\![a]\!]_{pk})^c$. These two primitives can be combined to compute the dot product securely. More detail can be found in [20, 8].

A straightforward application of this protocol in I-Pic, however, faces two problems: First, the capture device learns the dot products, which would enable a 'rogue' capture device to learn the classifier weight vector of each bystander. By computing dot products using a series of standard basis vectors (vectors that have a value of one in one dimension and zero in all others), the dot product values reveal the dimensions of a bystander's weight vector. To prevent this attack, we use garbled circuits [9], described below, to compute whether the dot product exceeds a threshold $\mathcal{E}$ without revealing the dot product itself.

Second, a capture device typically needs to compare several feature vectors, corresponding to multiple faces that appear in a photo, to the classifier weight vector of a bystander. For n feature vectors with m dimensions, the secure dot product computations require nm encryptions (and n decryptions). We can optimize this computation as follows.

Optimized n x 1 secure dot product: I-Pic reduces the number of encryptions from nm to m using ideas from [21]. Consider a matrix V of n vectors with m dimensions each, corresponding to n faces in a photograph, where $V_{i,j}$ is the jth element in the ith vector. Let $c_j = [V_{1,j}, V_{2,j}, ..., V_{n,j}]$ be the jth column of V. The photographer computes an encryption of c_j as $[\![c_j]\!]_{pk} = [\![(V_{1,j}) \parallel (V_{2,j}) \parallel ... \parallel (V_{n,j})]\!]_{pk}$, where $\parallel$ denotes concatenation. This involves only one encryption to produce the ciphertext for n values. The photographer sends $[\![\mathbf{c}_1]\!]_{pk}, ..., [\![\mathbf{c}_m]\!]_{pk}$, the encrypted user ids (*uid*) of the discovered bystanders, and pk to the *Bystander Agent*. For each bystander, the *Bystander Agent* computes $[\![v_{b_j}\mathbf{c}_j]\!]_{pk} = ([\![\mathbf{c}_j]\!]_{pk})^{v_{b_j}}$ for $1 \le j \le m$, where v_b is the classifier weight vector of a bystander. Multiplying these encrypted values, the *Bystander Agent* obtains a packed encryption of the dot products, $[\![P_1 \parallel ... \parallel P_n]\!]_{pk} = [\![V_1 \cdot v_b \parallel V_2 \cdot v_b \parallel ... \parallel V_n \cdot v_b]\!]_{pk} = [\![v_{b_1}\mathbf{c}_1]\!]_{pk}[\![v_{b_2}\mathbf{c}_2]\!]_{pk}...[\![v_{b_m}\mathbf{c}_m]\!]_{pk}$ and sends it back to the photographer, who decrypts (using sk) and unpacks the values to recover the individual dot products.

Garbled circuits for secure threshold computation: Garbled circuits allow two parties holding inputs x and y, respectively, to evaluate an arbitrary function *f(x,y)* without disclosing their inputs. The basic idea is that one party (the garbled circuit generator—the *Capture Agent* in our setting), prepares an "encrypted" version of a boolean circuit computing *f*; the second party (the circuit evaluator—the *Bystander Agent* in our case) then obliviously computes the output of the circuit. The combination of *secure dot product* and *garbled circuits* can provide the property that the bystander's session key is revealed to the capture device if, and only if, there is a match between an extracted feature vector and the classifier weight vector of a bystander. The capture device can then decrypt the bystander's policy.

4.3 Secure matching protocol

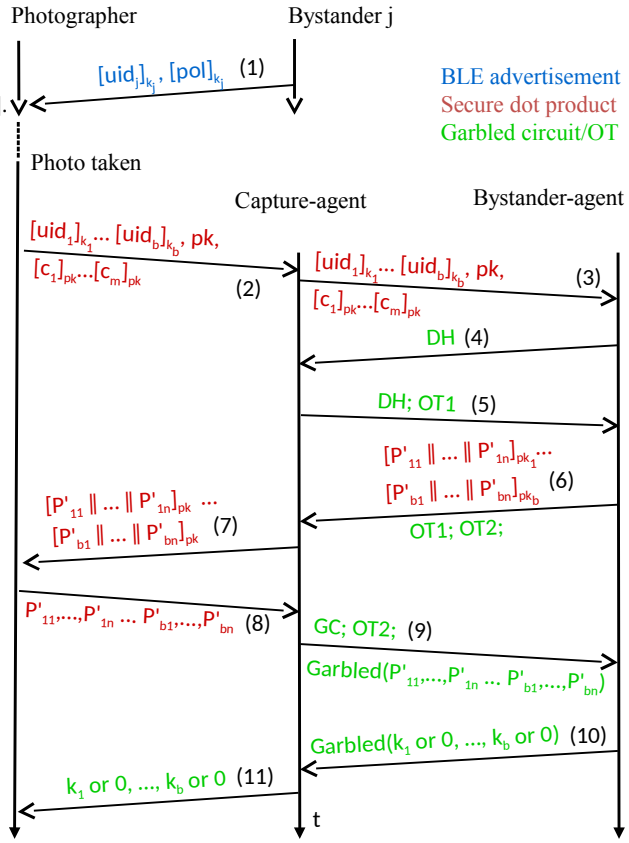


Figure 4. I-Pic secure matching protocol for one image with n faces (each facial feature vector has m dimensions). The photographer receives an advertisement from one of b bystanders (blue). The secure dot product computation requires one round trip (red). The garbled circuit (GC) requires a DH key exchange and two rounds of oblivious transfers (OT) (green).

An example message exchange of the secure matching protocol for one image with n detected faces and b bystanders is shown in Figure 4. The photographer's device computes the m encrypted column vectors according to the "optimized n x 1" secure dot product protocol, which requires m encryptions. The device sends these vectors to the *Bystander Agent* (via the *Capture Agent*) along with the encrypted user ids of the b bystanders (Message 2 and 3 in Figure 4).

The I-Pic *Bystander Agent*[1] now looks up the classifier weight vectors of the b bystanders. For each bystander, it computes the encrypted packed dot products, $[\![P_{i,1} \parallel P_{i,2} \parallel ... \parallel P_{i,n}]\!]_{pk}, 1 \leq i \leq b$, of the bystander feature vector and the n image feature vectors.

Secure thresholding The *Bystander Agent* computes *obscured* encrypted packed dot products, $[\![P'_{i,1} \parallel P'_{i,2} \parallel ... \parallel P'_{i,n}]\!], 1 \leq i \leq b$, by adding a different random value $R_{i,j}$ to each dot product $P_{i,j}$, for $1 \leq i \leq b, 1 \leq j \leq n$. This is performed by multiplying each of the b packed encrypted values containing n dot products each, $[\![P_{i,1} \parallel P_{i,2} \parallel ... \parallel P_{i,n}]\!]_{pk}$, with $[\![R_{i,1} \parallel R_{i,2} \parallel ... \parallel R_{i,n}]\!]_{pk}$ for $1 \leq i \leq b$. These *obscured* encrypted packed dot products are sent to the photographer's device via the *Capture Agent* (Message 6 and 7).

The photographer's device decrypts the b packed encrypted values containing n *obscured* dot products each, which requires b decryption operations. The device forwards these obscured dot products to the *Capture Agent* (Message 8), which then constructs a garbled circuit that takes as input n obscured dot products $P'_{i,j} = P_{i,j} + R_{i,j}$, n random values $R_{i,j}$, a session key K_i, and the threshold $\mathcal{E}$ (all provided by the *Bystander Agent*), for $1 \leq i \leq b, 1 \leq j \leq n$. The circuit computes

$$f(P'_{i,j}, \mathcal{E}, R_{i,j}, K_i) = \begin{cases} K_i & \text{if } P'_{i,j} > \mathcal{E} + R_{i,j} \\ 0 & \text{Otherwise} \end{cases}$$

that is, the circuit reveals a bystander's session key iff the dot product of the bystander's classifier weight vector and an image feature vector exceed the threshold.

Delivering the *Bystander Agent*'s inputs to the garbled circuit requires a Diffie-Hellman key exchange (DH) and two rounds of oblivious transfers (NPOT [22] and OTEXT [23]), which are partly piggy-backed on the secure dot product protocol messages, and shown in Figure 4 (Messages 4, 5, 6 and 9). The *Capture Agent* now sends the circuit to the *Bystander Agent*, along with the garbled values of the obfuscated inputs $P'_{i,j}$, and the garbled values of *Bystander Agent*'s inputs as part of the OTEXT oblivious transfer (Message 9). The *Bystander Agent* executes the circuit b times with the appropriate inputs, and returns the garbled results to the *Capture Agent* (Message 10). After ungarbling the results, the *Capture Agent* returns the session keys for the matched bystanders to the photographer's device (Message 11).

As composed, the matching protocol has the desired property that a photographer learns a bystander's current session key if and only if a feature vector in the image matches that bystander's classifier weight vector. Garbled circuits also ensure that the *Bystander Agent* does not learn whether there was a match between the encrypted facial feature vectors and a bystander. Additionally, no principal learns the vectors held by the other principals nor the magnitude of the dot products.

Note that the *Capture Agent* is trusted to construct the garbled circuit correctly. This requirement could be relaxed if one is willing to run additional checks [24] at some additional computational and runtime overhead.

5. EVALUATION

We have prototyped I-Pic on Android version 4.4.2. In our deployment, we used a Google Project Tango Tablet [25] as the photographer's capture device and Galaxy Nexus[2] phones as bystander devices. The Nexus phones advertised their presence once every 640ms over BLE.

We ported HeadHunter [6] to Android for face detection. HeadHunter is optimized for execution on CUDA-enabled GPUs [26]; the Tango Tablet allows us to access CUDA cores. The camera output on the tablet (available as a JPEG file) is first histogram equalized [27] and then resized to 640x360 before being input to HeadHunter. HeadHunter outputs bounding boxes corresponding to detected faces.

To extract feature vectors from facial images, we used an Android port of the Caffe framework [28] and ran it with our FNet neural network. The extracted vectors were normalised such that each feature value was in the range $[0, 1]$. We ported existing Java secure dot product and garbled circuit implementations [29] to C++ on Android to optimize for runtime and energy consumption. The various agents were implemented as HTTP servers.

We begin with a description of I-Pic deployments in various settings; these deployments were also approved by the University of Maryland IRB. While we gained intuition about our vision pipeline using standard face recognition datasets (and the pipeline's performance compares well with the state-of-the-art on them), all results presented here evaluate I-Pic on images captured "in the wild", reflecting spontaneous image capture in different social situations with a range of lighting conditions, camera angles, distances, and poses.

5.1 Deployments

To evaluate I-Pic, we registered fifteen volunteers from our institutions using the registration procedure detailed in Section 4.1. Each volunteer received a Galaxy Nexus device for BLE advertisement, which they carried on their person. Registered users could choose to either *show* or *blur* their face when photographed; this setting could be changed at their discretion.

The photographs in our results were captured over three days (see Table 3), and were taken using the Tango tablet and a DSLR camera. We used the DSLR setup (Sony A7, 35mm f/2.8 lens, 1/80 fixed exposure time with Sony HVL-F32M flash) to simulate better tablet cameras with higher resolution and faster apertures expected in future tablets. The photographs captured by the DSLR were manually fed into the I-Pic processing pipeline.

We annotated all photographs manually with ground truth face rectangles using the open source annotation tool Sloth [30]. For each face, we manually added other information, such as the identity of registered users, pose, and lighting condition.

Date	Capture device	Number of photographs	Number of ground-truth faces
Nov 20	Tango tablet	81	277
Nov 27	Tango tablet	176	553
Dec 02	DSLR	130	843
	All	387	1673

Table 3. Experimental dataset

5.2 I-Pic decision tree

In I-Pic, faces in photographs end up being edited (e.g., blurred) or remain unchanged, correctly or incorrectly, depending on decisions made by different subsystems. Figure 5 shows the possible paths through I-Pic, culminating in leaf nodes colored green if I-Pic preserves user privacy and red if it does not. Note that it is possible

[1]To simplify exposition, the description here assumes a single *Bystander Agent* service. The capture device would have to execute the protocol for each *Bystander Agent* in case more than one is discovered.

[2]Galaxy Nexus has Bluetooth hardware capable of BLE advertising, but the functionality is not available via standard API calls. We patched the kernel to enable BLE advertising.

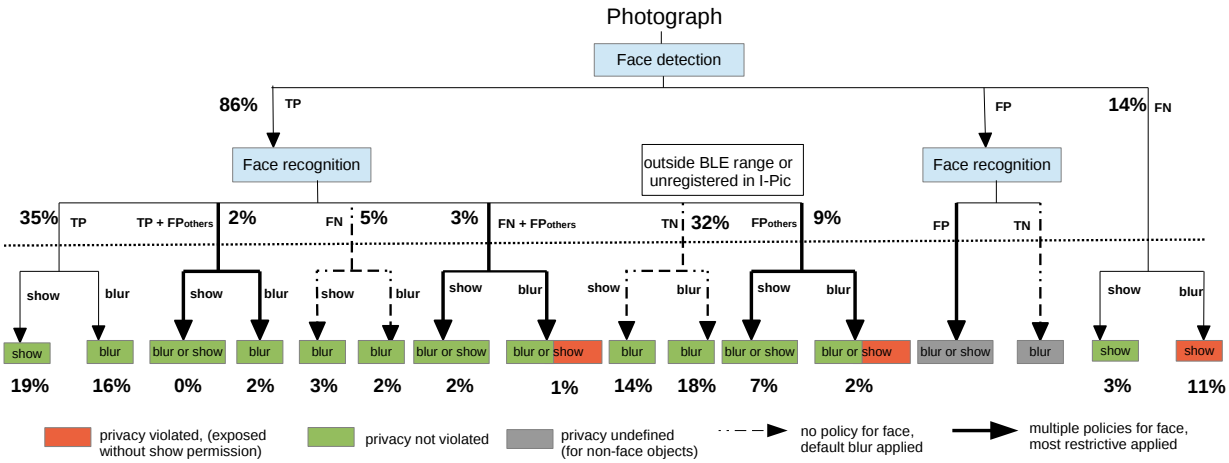


Figure 5. I-Pic decision tree

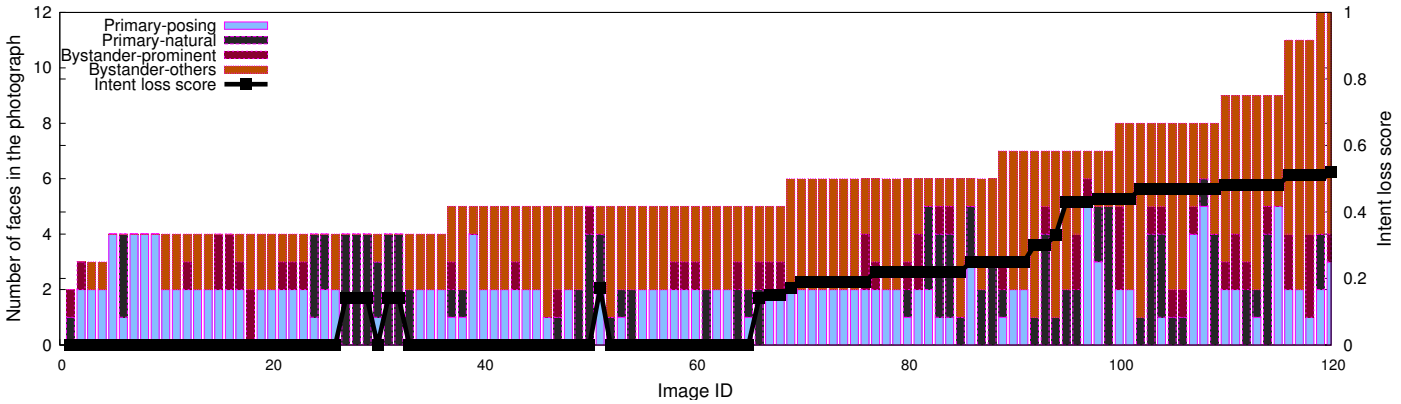


Figure 6. I-Pic Intent score of images

for I-Pic to make a mistake, e.g., not recognize a face, and for the corresponding path to still lead to a green leaf node, e.g., because the user policy stated not to obscure their face. Finally, some leaf nodes are grey, corresponding to privacy irrelevant mistakes where non-faces were detected as faces and possibly blurred.

Understanding this decision tree, and in particular, analyzing where privacy-relevant errors can accrue, will enable us to parameterize and evaluate our vision pipeline in the context of I-Pic's overall goal.

The decision tree has three stages: (1) face detection, (2) face recognition and (3) policy application. Stages 1 and 2 are computational and depend solely on the accuracy of the vision pipeline. The diagram separates these from Stage 3, which is contingent on user choices. For instance, if users choose more permissive policies, then errors from previous stages will less likely result in privacy violations, and vice-versa.

Face Detection: Stage 1 may result in three outcomes: True Positive (*TP*), where I-Pic detects a face marked in ground truth; False Positive (*FP*), where I-Pic detects a non-face object as a face; or False Negative (*FN*), where I-Pic does not detect a face marked by ground truth. All *TP* and *FP* detections are passed to the face recognition engine in the next stage.

The *FN* faces bypass the I-Pic pipeline and remain unchanged, and can potentially lead to a privacy violation (red leaf node). To minimize these cases, we bias the face detection engine towards higher recall (lower *FN*) at the expense of lower precision (higher *FP*). This means that a non-face object occasionally gets blurred in an image, in exchange for increased privacy.

Face Recognition: For a *TP* face detection output, there are six possible choices for recognition in the I-Pic pipeline: (1) True Positive (*TP*), where the detected face is matched only with the individual identified in ground truth; (2) True Positive along with False Positives (TP^*), where the face is matched with the ground truth individual, but also with others[3]; (3) False Negative (*FN*), where the face is not matched with the ground truth person; (4) False Negative along with False Positives (FN^*): I-Pic does not match with the ground truth, but instead matches with one or more other registered individuals; (5) True Negative (*TN*), where I-Pic correctly does not match the face to any registered individual; and (6) False Positive(s) (FP^*), where I-Pic incorrectly matches the face to one or more registered users.

Two leaf nodes have privacy violations for face recognition. *FP* is responsible for both paths, while one of them also requires a *FN*. Thus lower *FP* or high precision has higher priority for recognition, and adequate balance with low *FN* or high recall is also necessary. These requirements guide the parameterization of the I-Pic face recognition engine.

[3]We allow multiple matches; any registered face that exceeds a similarity threshold is considered a match.

Misdetected faces (*FP* in detection) are also fed into the recognition protocol, and may lead to (1) True Negatives (*TN*) whereby I-Pic does not recognize the "face" as a registered user, or (2) False Positives (FP^*) where I-Pic mistakenly matches the "face" to one or more registered users.

Policy: Each detected face leads to an action, as shown by the leaves of the tree. If the recognition engine outputs a single user, then the action corresponding to that users' policy is undertaken. However, in cases of multiple matches, e.g., due to TP^*, FN^* or FP^*, the most restrictive policy chosen by any "recognized" user is applied. For all unrecognized users, I-Pic blurs faces by default.

We will detail an experiment with 687 faces in 120 images to examine I-Pic's privacy violations in Section 5.3. The percentages below the leaves in Figure 5 show the fraction of faces that mapped to each path in the decision tree, in this experiment. As can be seen from the percentage values, the privacy preferences of 14% of 687 captured faces were violated, primarily due to errors early in the vision pipeline (face detection). In the next sections, we will present detailed evaluations of the vision pipeline, whose accuracy primarily determines I-Pic's performance.

5.3 I-Pic overall performance

We begin with an evaluation of I-Pic's overall performance in terms of its primary goals, which are to (i) respect bystanders' privacy, and to (ii) preserve the photographer's intent to the extent allowed by subjects' privacy choices.

Toward this end, we took a sample of 120 images with 687 faces marked in the ground-truth. We additionally marked each face according to its role in the image, as shown in Table 4, along with the frequency of faces with a given role.

Name	Role in photograph	Number of occurrences
PP	primary subject posing	185
PN	primary subject natural	115
BP	prominent bystander	56
BO	other bystanders	331

Table 4. Roles of faces captured in images

Many of the captured faces correspond to unregistered individuals. Since we don't know the privacy preferences of these individuals, we assigned them policies manually, so that we can process each image as if each captured person were registered with a policy. We assigned the *show-face* policy to the 185 PP faces, since it would be inconsistent for a person who poses for a photograph to refuse to have their face shown. For the remaining 502 faces, we randomly choose one of *show-face* or *blur-face* policies.

The percentage values given at the leaves in Figure 5 show what fraction of these 687 faces had what outcome when run through the I-Pic system. As we can see, privacy was violated in 14% of the cases, while the remaining 86% had no privacy violation.

We also assign a privacy loss score in each case of violation. These scores provide a subjective measure of the severity of the privacy violation depending on the role of the face in the image, with higher scores indicating a more severe violation. The privacy loss scores are given in Table 5, with the last column indicating how many of each type of violation occurred in the 687 faces.

About 2% of cases had the most severe privacy violation, which is to show a primary subject not posing for the camera against their wishes. Also about 2% of cases had a clearly visible bystander shown against their wishes, and around 10% were less severe cases, where a not prominently depicted bystander was not blurred. We conclude that, overall, I-Pic observes subjects' policies in most cases (86%). Moreover, violations that did occur were mostly in the moderate or mild category.

Privacy loss score	penalization scenario	occurrences
3	PN privacy violated	15 (2.18%)
2	BP privacy violated	12 (1.75%)
1	BO privacy violated	70 (10.19%)
0	no privacy violated	590 (85.88%)

Table 5. Privacy loss scores

The second aspect of I-Pic's overall performance is its ability to preserve the photographer's intent, to the extent allowed by the subject's policies. Similar to the privacy loss score, we can define a subjective intent loss score, which penalizes blurring a posing primary subject (score 3), blurring a non-posing primary subject with a *show-face* policy (score 2), and bystanders with *show-face* policies (score 1) in decreasing order of severity. The ordering is based on a subjective judgment of intent loss severity when a face is unnecessarily blurred, based on the face's role in the image. We note that our assignment of an intent penalty for the bystander case is conservative, as it is unclear whether a photographer should have expectations about capturing bystanders.

Figure 6 shows the intent loss scores for the 120 images, normalized by the maximum intent loss that could occur in a given image. The images are sorted by increasing number of faces from left to right. The bars represent the image composition in terms of roles of the faces depicted in it. I-Pic preserves the photographer's intent, as measured by our score, perfectly in 55 (45.8%) of the images, with the intent loss increasing for pictures with more faces. The vast majority of intent loss cases are caused by a failure to recognize the face of a bystander with a permissive policy, combined with I-Pic's default policy to blur.

Being focused on privacy, I-Pic biases its choices towards privacy, including the default policy and the rule to apply the most restrictive policy in case of multiple matches. As a result, losses in the vision pipeline come at the expense of intent rather than privacy. In the following subsections, we investigate circumstances that lead to imperfections in the vision pipeline, which are causal for the losses in privacy and intent reported here.

5.4 Vision pipeline analysis

The I-Pic decision tree demonstrates how (and how many) privacy violations occur as a result of errors in the vision pipeline. An obvious case is when a face is not detected, and thus not blurred in post-process. We have identified and manually labeled images with factors that affect detection and analysis, as we explain next. This analysis is done with our full image dataset of 387 images, where 1673 faces have been manually marked with ground truth (Table 3).

5.4.1 Factors affecting detection and recognition

The factors labeled in the ground truth (lighting, pose, and size) greatly affect whether a face is detected or not. We determine size based on the number of pixels in the image the face occupies; "small" faces (**s-Sm**) have a bounding box with at least one dimension less than 100 pixels[4]; all other faces are "large" (**s-Lg**). Pose is one of "frontal, profile, tilted head" (**p-Std**); "facing up, down" (**p-Avert**); "back turned, obstructed view" (**p-Occ**). Lighting is one of "Bright, even lighting" (**l-Good**); "Low even lighting" (**l-Low**); "Backlit, Shadow, Strong directional" (**l-Poor**).

Figure 7 decomposes face detection recall along these factors, for our image dataset (Table 3). The figure includes example images corresponding to different conditions for visual reference. The recall values for detection can be as high as 95% to as low as 32%,

[4]The Tango camera produces 2688 x 1520 pixels images and Sony A7 produces 4240 x 2832 pixels images

based on lighting, face size, and how occluded a face is in a photograph.

The leftmost bar with recall around 32% represents all combinations of factors combined with a partly occluded pose (**p-Occ**). 20% of the faces in our dataset are in this category. Together with the faces that suffer from low or poor illumination and an averted pose (four leftmost bars), they have recall below 50%. Faces in this category are probably not clearly recognizable even for humans without contextual information.

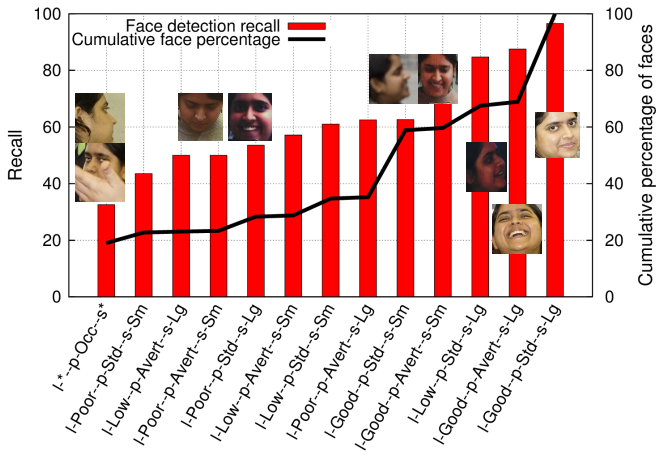


Figure 7. Face detection accuracy vs. illumination conditions, face poses and face sizes

Face characteristic	Recognition recall
l-Good–p-Std–s-Lg	85.22%
l-Good–p-Avert–s-Lg	82.79%
l-Low–p-Std–s-Lg	78.62%
l-Good–p-Avert–s-Sm	67.38%
l-Good–p-Std–s-Sm	66.29%
p-Occ or **l-Poor**	20.49%

Table 6. Face recognition recall vs. different illumination conditions, face poses and face sizes

Table 6 shows the face recognition recall for a subset of illumination, pose and size characteristics. Recognition recall is only meaningful for individuals who are registered in the I-Pic system. Our 15 registered individuals occured with the subset of conditions given in Table 6, while only unregistered individuals occured in other conditions.

p-Occ and **l-Poor** lead to poor recognition recall. This effect is intuitive, as occlusion or directional lighting distorts the facial features, making it harder to match with registered face models. Additionally, **s-Sm** performs worse than **s-Lg**. Our FNet neural network scales the input image to 227 x 227 pixels before feature extraction. Since **s-Sm** faces are less than 100 pixels in either width or height, this upscaling potentially affects the face recognition accuracy for small faces.

Precision for face detection or recognition do not show any marked correlation under different illumination, pose or size. In summary, good detection recall (>60%) and excellent recognition recall of nearly 80% occurs when pose is frontal or averted, illumination is good or low, or the size is large. This category includes about 65% of the faces in our images, and represents cases where subjects are clearly recognizable and privacy is most important.

5.4.2 *Mapping back to events*

The previous section identified different factors affecting I-Pic's face detection and recognition. But in what scenarios can one expect favorable conditions? In this section, we describe the scenarios in which we have evaluated I-Pic, and catalog photographs and faces from each scenario according to our factors. We note that photographers were *not* aware of these factors when the photographs were taken.

Table 7 lists four events where we obtained about 64% of our captured images. These images contain 970 manually annotated faces; the table lists the number of faces for each context. Figure 8 shows representative images from each event; Figures 9(a)-(c), show the illumination, poses and size distribution for these 970 faces.

Context name	characteristics	illumination
Campus (180 faces)	Individuals posing outdoors, with some bystanders present	Natural light
Social (237 faces)	Afternoon tea session with 40 people in an indoor atrium	Combination natural and fluorescent light
Office (129 faces)	Daily exchanges in offices and corridors	Fluorescent light
Party (424 faces)	Crowded party in small indoor venue	Back and directional lighting from lamps

Table 7. Four different social contexts

Figure 10 plots the recall ($\frac{TP}{TP+FN}$) and precision ($\frac{TP}{TP+FP}$) for both detection and recognition for the four events. The plot also includes data for *All*, corresponding to all 1673 faces in our evaluation, including those taken outside the four events.

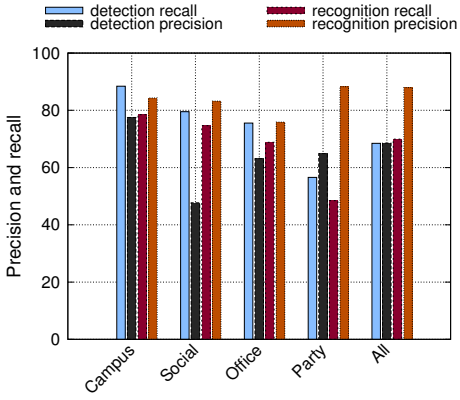


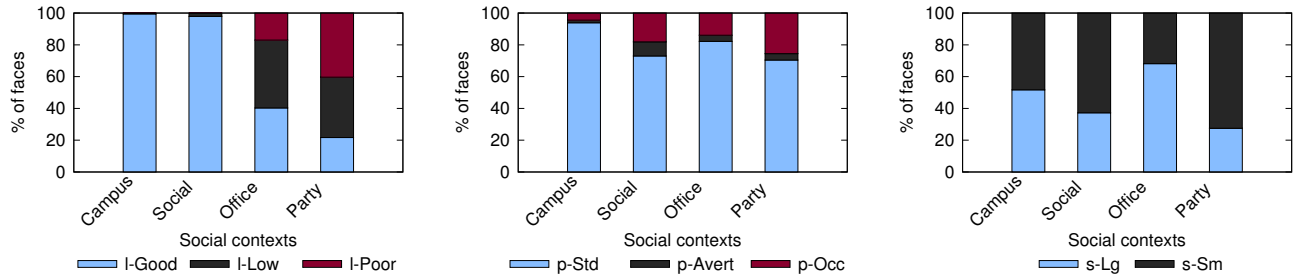
Figure 10. I-Pic vision pipeline performance

Both detection and recognition recalls depend on contexts: *Campus* photographs taken outdoors with favorable poses have high recall for both detection and recognition. In contrast, challenging lighting and occluded faces in the indoor *Party* context lead to low recall.

Face recognition precision is high, independent of the social context. However, face detection precision varies with context. Manual inspection of the images revealed that busy scenes with many people have more false positives in face detection. Here, body parts like ears or hands, or striped clothing, accidentally match the face detection template of HeadHunter. This shows up as lower precision in the *Social* context, which has crowded scenes.

As discussed in Section 5.2, I-Pic is biased towards higher recall for face detection and higher precision for face recognition, to

Figure 8. Examples images from four events used in evaluating I-Pic.



(a) Illumination (b) Face poses (c) Face sizes

Figure 9. Variety in illumination conditions, faces poses and face sizes in different contexts and resulting performance

maximize the privacy scores of the system. Figure 10 shows the effects of these choices on the vision pipeline performance.

In summary, the *Campus*, *Social*, and *Office* contexts have recall in the 70-80% range for both face detection and recognition. The challenging scenarios like *Party* provide an opportunity for future vision research. Our image dataset, captured with mobile cameras, will be very useful to design new vision algorithms, which I-Pic can incorporate in the future.

5.4.3 *Comparison to existing face detectors*

We have used our own research prototype, HeadHunter, for face detection. A natural question to consider is how well existing, widely used face detectors, such as those bundled with Android or OpenCV, compare. Table 8 shows the precision and recall for different face detection libraries on our dataset. HeadHunter vastly outperforms the competition, justifying its use within I-Pic. Note that low detection recall, in particular, leads to false negatives in I-Pic, which can lead to privacy violations.

Library	Precision	Recall
Android	38.65	5.49
Snapdragon	94.28	5.91
OpenCV	31.27	49.91
HeadHunter	68.47	68.55

Table 8. Comparison of face detection libraries

5.5 Secure Feature Comparison

Next, we present microbenchmarks evaluating the processing and bandwidth requirements of the secure vector matching protocol with varying numbers of faces and bystanders. During these experiments, both the cloud agents are running on the same machine and are on the same 802.11 WiFi network as the I-Pic devices. In each run of the experiment, we generated feature vectors randomly.

Consider Figures 11 and 12, which show the protocol's total runtime latency and its breakdown. Latency includes computations on the device, on the cloud agents, and the network transit time between the device and the *Capture Agent*.

The number of input vectors that have to be encrypted and transmitted increases with the number of faces in the photograph, resulting in an expected linear increase in runtime in Figure 11. Figure 12 shows that a major contribution to this runtime is the client side encryption of feature vectors for the secure dot product part of the protocol (Step 2 in Figure 4). Due to the "n x 1 dot product" optimization, described in Section 4.2, the client side runtime does not increase significantly with the number of faces.

From separate measurements (not shown in Figure 12) we know that these client side encryption operations show a 2x reduction in runtime on mobile platforms supporting a 64bit ARMv8-A instruction set. [5]

Increasing the number of bystanders for a fixed number of faces increases the runtime linearly, but importantly, it does not significantly increase the client-side runtime (Figure 12). This is a desirable property as the photographer's overhead does not signficantly depend on the number of bystanders in the vicinity.

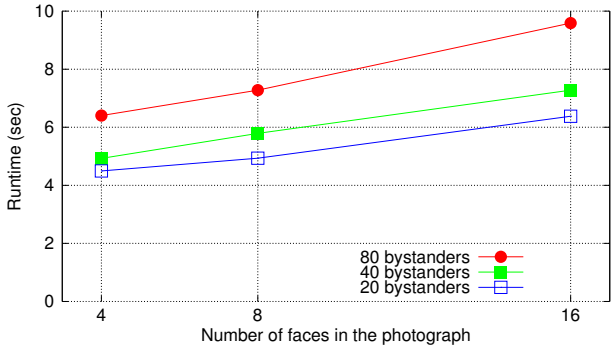


Figure 11. Total Runtime of the secure matching protocol

Figure 13 shows the data transmitted between the device and *Capture Agent*, and between the cloud agents. We observe that data transmitted between the device and *Capture Agent* is less than 100KB and it does not increase significantly with the number of

[5]Measurements are not shown here because the Tango tablet does not support the ARMVv8-A instruction set.

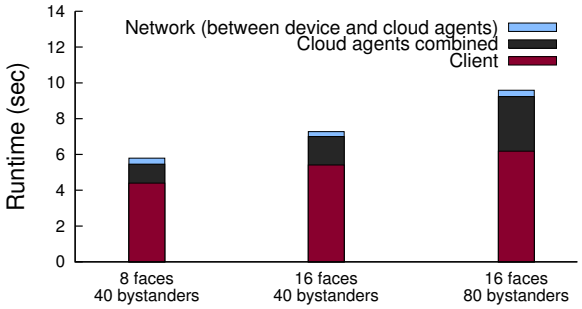


Figure 12. Runtime breakdown of secure matching protocol

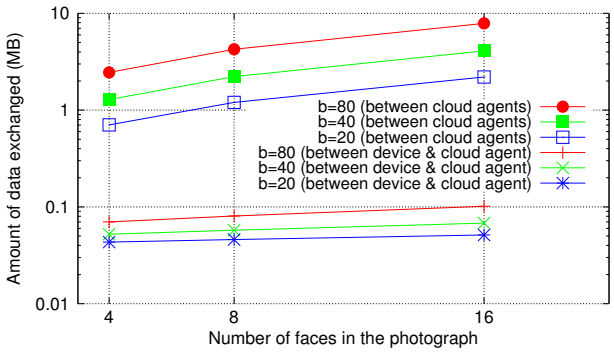


Figure 13. Total data exchanged for the secure matching protocol for different number of bystanders

faces or bystanders. This figure also shows the effects of adding the garbled circuit. The garbled circuit affects the data exchanged (and the latency) between the cloud agents, which increases both with the number of bystanders and the number of faces. Garbled circuits are evaluated by the *Bystander Agent* for each bystander and the number of inputs to each garbled circuit depends on the number of faces.

Overall the results show that the secure matching protocol can be efficiently executed. Moreover, computation can be offloaded to a significant extent from the client devices to the cloud agents.

5.6 Runtime and Energy Consumption

Figure 14 plots the overall time taken for I-Pic to process different photographs, along with times spent in different vision and secure matching tasks. In each case, the capture platform received and processed between 3 and 10 BLE advertisements, with varying number of faces in the photograph as plotted along the x-axis. The times for secure matching includes network communication and all cryptographic functions. Face detection dominates, often requiring 25 seconds per photograph. Recall that the processing takes place asynchronously in the background, and does not interfere with the users' experience while capturing and reviewing images.

While the face detection cost in particular is high in our prototype (70–80% of total processing time), we believe it is encouraging that best-of-breed face detection is feasible on mobile devices available today. Advances in mobile hardware capabilities, driven in part by emerging virtual reality applications, will benefit HeadHunter and other stages of the I-Pic pipeline in the near future. Moreover, face detection is already being offered as a standard feature on mobile platforms, and future implementations (possibly hardware supported) with better accuracy could directly benefit I-Pic.

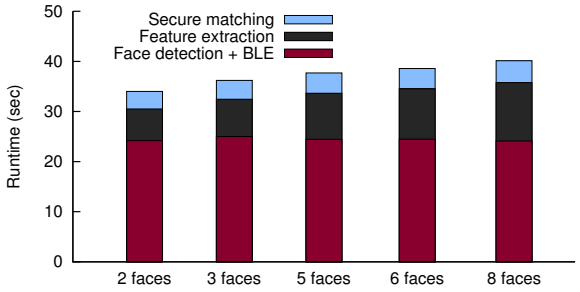


Figure 14. Overall and task level runtimes of I-Pic prototype. 10 bystanders were discovered in each case.

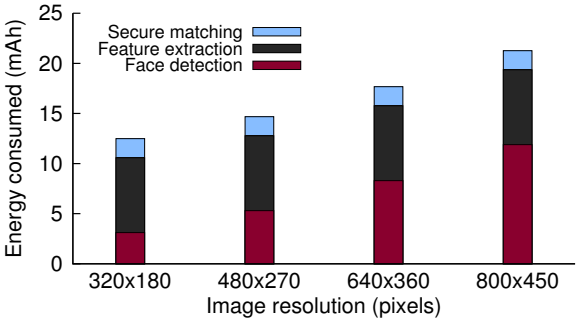


Figure 15. Energy consumption of I-Pic prototype for different image resolutions, 30 faces.

We measured the energy consumption of the various subcomponents of I-Pic using the Monsoon Power Monitor [31]. We attached the power monitor to a Nvidia Shield Tablet K1 [32][6] and processed an image with 30 faces in it. Figure 15 shows the energy consumption for different resolutions of the input image. The face detector uses the GPU, whereas the feature extraction is CPU bound. Energy consumption of face detection is independent of the number of faces in an image, whereas it is linear in the number of faces for feature extraction. The secure matching algorithm was run with the 30 faces extracted from the image along with 40 simulated bystanders[7].

Image resolution (pixels)	Number of images processed (containing 30 faces each)
320x180	408
480x270	347
640x360	288
800x450	239

Table 9. I-Pic's projected capacity on a 5100 mAh battery

Using these measurements, Table 9 shows I-Pic's projected capacity on the Nvidia Shield tablet, which has a 5100 mAh battery. More than 288 images and 8640 faces can be processed on a single charge. Figure 16 compares the face detection accuracy versus the resolution of input images, and serves to highlight the trade-off between accuracy and energy consumption of the prototype. Reducing the resolution to 480x270 pixels enables the prototype

[6]We used the Shield tablet for the power measurements because the Monsoon power monitor is unable to power the Tango tablet. The latter requires a 7.5 volts power supply whereas the Monsoon power monitor can only supply a maximum of 4.5 volts.

[7]BLE scanning for 5 seconds consumes 0.12 mAh of energy, which is accounted for in Figure 15 but not shown separately.

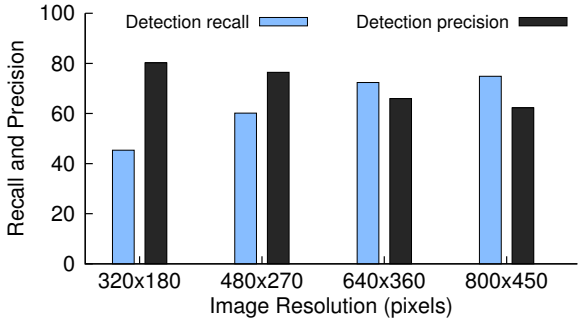


Figure 16. Face detection accuracy of I-Pic prototype for different image resolutions

to process 20% more images, but comes at a high (12%) drop in face detection recall. On the other hand increasing the resolution to 800x450 only gives diminishing returns for face detection recall when compared to the increased energy consumption that accompanies it.

6. RELATED WORK

Privacy in the presence of recording devices: Hoyle et al. [33] seek to understand users' concerns about continuous recording using wearable cameras, by studying a large user population of avid life-loggers. Denning et al. [34] conduct a large scale user survey to understand bystanders' privacy concerns in public places like coffee shops and possible ways to mitigate them. Our online survey additionally shows that privacy concerns are very personal and dependent on the situation.

Roesner et al. [2] present a system that shares a venue's privacy preferences with wearable devices in an unobtrusive way. The idea is to convey privacy expectations associated with places like gyms and washrooms with broadcast messages or visual signs. The wearable devices in the venue pick up these messages or visual cues and obey the specified privacy protocol. Unlike I-Pic, this system has no way to associate a privacy policy with an object or person that appears in an audiovisual recording.

Visual markers to convey privacy policies to nearby wearable recording devices are also used in [3]. [35] explores the expression of bystanders' privacy intent using gestures. Unlike I-Pic, these approaches require either physical tagging of objects and locations, or explicit user actions (i.e., gestures) to convey privacy choices. Moreover, I-Pic enables user-defined, personalized, context-dependent privacy choices.

In the work by Bo et al. [4], individuals wear clothes with a printed barcode, which encodes the wearer's public key. When an image of an individual showing face and barcode is uploaded to an image server, the server garbles the face pixels, using the public key encoded in the barcode. Only the individual who owns the associated private key can later extract the actual face image. I-Pic, on the other hand, does not require its users to wear any visual markers, it does not require users to trust an image server with their private images, and can support context-dependent privacy policies.

In [36, 37, 38], the authors address privacy concerns in untrusted perceptual and augmented reality applications, by partially processing media stream within the trusted platform, thus denying apps access to the raw media streams. An augmented reality app, for instance, might be provided only with the position of relevant objects within a video stream sufficient for the app to overlay its own information, but not the full video. I-Pic also relies on the trusted platform, but focuses on enforcing individual's privacy policies regarding image capture by nearby devices.

Zero-Effort Payments [39], similar to I-Pic, uses face recognition and proximate device detection using BLE to identify a user in an image, but their goal instead is to create a mobile payment system. Unlike I-Pic, which is tuned to identify even small faces in diverse range of photographic contexts, their system is meant to visually identify a user, with human assistance, when she is in close proximity to the cashier. Furthermore, they acknowledge concerns of user privacy in such a monitored environment and propose the use of signage indicating that a face recognition system is deployed in the area. Such a privacy solution is only viable in select scenarios, and lacks the flexibility provided by I-Pic.

Visual fingerprints: Performance on human identification and re-identification tasks has greatly improved over the last decade. Most notably, face recognition on large databases in realistic settings is even approaching human performance [40]. Besides the identity, a person can also be described and identified by a set of attributes [41, 42]. I-Pic uses a state of the art face recognition algorithm based on neural networks, but can benefit from using semantic attributes describing a face, including features from other body parts in addition to the face.

Cryptograhic primitives: There is complementary work to protect the privacy of biometric data [43, 44] by projecting or encrypting representations. It is possible that these approaches could be used in I-Pic to further reduce trust in the Cloud service by obscuring users' visual signatures.

InnerCircle [45] describes a secure multi-party protocol for location privacy, which computes in a single round whether the distance between two encrypted coordinates is within some radius r. This computation is similar to I-Pic's secure dot product and theshholding computation. However, the protocol's efficiency degrades exponentially with the number of bits of precision of the distance. Since our threshold comparison involves dot products of large feature vectors, we use garbled circuits for the threshold comparison instead.

7. CONCLUSIONS AND FUTURE WORK

I-Pic allows users to respect each others' individual and situational privacy preferences, without giving up the spontaneity, ubiquity, and flexibility of digital capture. The I-Pic design and prototype demonstrate that the technical impediments for privacy-compliant imaging can be reasonably overcome using current hardware platforms. I-Pic leverages cutting-edge face detection and recognition technology, which is often perceived as a threat to privacy, to instead increase user's privacy regarding digital capture. Future advances in mobile platform hardware and computer vision will directly benefit I-Pic and further improve the efficiency and accuracy of its I-Pic privacy enforcement.

8. ACKNOWLEDGMENTS

We would like to thank the participants in our online survey and the live photography sessions. Our special thanks to Aniket Kate and Michelle Mazurek for their advise on the crypto pipeline and online survey, respectively. We are grateful to Christian Klein and Aaron Schulman for their help with energy measurements. Finally, we would like to thank the anonymous reviewers and our shepherd Jeremy Gummeson for their helpful feedback. This research was supported by the European Research Council (ERC Synergy imPACT 610150) and the German Research Foundation (DFG CRC 1223). Rijurekha Sen was support by a Humboldt post-doctoral research fellowship.

9. REFERENCES

[1] Lost lake cafe, seattle restaurant, kicks out patron for wearing google glass. http://www.huffingtonpost.com/2013/11/27/lost-lake-cafe-google-glass_n_4350039.html.

[2] Franziska Roesner, David Molnar, Alexander Moshchuk, Tadayoshi Kohno, and Helen J. Wang. World-driven access control for continuous sensing. In *ACM Conference on Computer and Communications Security (CCS)*, 2014.

[3] Nisarg Raval, Animesh Srivastava, Ali Razeen, Kiron Lebeck, Ashwin Machanavajjhala, and Landon P. Cox. What you mark is what apps see. In *ACM International Conference on Mobile Systems, Applications, and Services (Mobisys)*, 2016.

[4] Cheng Bo, Guobin Shen, Jie Liu, Xiang-Yang Li, Yongguang Zhang, and Feng Zhao. Privacy.tag: Privacy concern expressed and respected. In *ACM Conference on Embedded Networked Sensor Systems (Sensys)*, 2014.

[5] Nisarg Raval, Animesh Srivastava, Kiron Lebeck, Landon P. Cox, and Ashwin Machanavajjhala. Markit: Privacy markers for protecting visual secrets. In *UPSIDE, Workshop at ACM International Joint Conference on Pervasive and Ubiquitous Computing (Ubicomp)*, 2014.

[6] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool. Face detection without bells and whistles. In *European Conference on Computer Vision (ECCV)*, 2014.

[7] S. Joon Oh, R. Benenson, M. Fritz, and B. Schiele. Person recognition in personal photo collections. In *International Conference on Computer Vision (ICCV)*, 2015.

[8] Bart Goethals, Sven Laur, Helger Lipmaa, and Taneli Mielikainen. On private scalar product computation for privacy-preserving data mining. In *7th Annual International Conference in Information Security and Cryptology (ICISC)*, 2004.

[9] Andrew Chi-Chih Yao. How to generate and exchange secrets. In *27th Annual Symposium on Foundations of Computer Science (FOCS)*, 1986.

[10] Terence Sim and Li Zhang. Controllable face privacy. In *The 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, 2015.

[11] Antonio Criminisi, Patrick Perez, , and Kentaro Toyama. Region filling and object removal by exemplar-based image inpainting. In *IEEE Transactions on image processing, vol. 13, no. 9, September*, 2004.

[12] X. Zhu and D. Ramanan. Face detection, pose estimation and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[13] Ning Zhang, Manohar Paluri, Yaniv Taigman, Rob Fergus, and Lubomir Bourdev. Beyond frontal faces: Improving person recognition using multiple cues. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Conference on Neural Information Processing Systems (NIPS)*. 2012.

[15] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. *Computer Vision and Pattern Recognition (CVPR)*, 2009.

[16] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[17] Gary B. Huang Erik Learned-Miller. Labeled faces in the wild: Updates and new reporting procedures. Technical Report UM-CS-2014-003, University of Massachusetts, Amherst, May 2014.

[18] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 2008.

[19] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology (EUROCRYPT)*, 1999.

[20] Paarijaat Aditya et al. Technical Report: I-Pic: A Platform for Privacy-Compliant Image Capture. http://www.mpi-sws.org/~paditya/papers/ipic-tr.pdf.

[21] Yan Huang, Lior Malka, David Evans, and Jonathan Katz. Efficient privacy-preserving biometric identification. In *18th Network and Distributed System Security Conference (NDSS)*, 2011.

[22] Moni Naor and Benny Pinkas. Computationally secure oblivious transfer. In *Journal of Cryptology*, 2005.

[23] Yuval Ishai, Joe Kilian, Kobbi Nissim, and Erez Petrank. Extending oblivious transfers efficiently. In *Advances in Cryptology (CRYPTO)*, 2003.

[24] Yehuda Lindell. Fast cut-and-choose based protocols for malicious and covert adversaries. In *Advances in Cryptology (CRYPTO)*, 2013.

[25] Project Tango Tablet Development Kit. https://store.google.com/product/project_tango_tablet_development_kit.

[26] CUDA. http://www.nvidia.com/object/cuda_home_new.html.

[27] Jose-Luis Lisani, Ana-Belen Petro, and Catalina Sbert. Color and Contrast Enhancement by Controlled Piecewise Affine Histogram Equalization. *Image Processing On Line*, 2:243–265, 2012. http://dx.doi.org/10.5201/ipol.2012.lps-pae.

[28] Caffe-Android-Lib. https://github.com/sh1r0/caffe-android-lib.

[29] Might Be Evil. http://mightbeevil.org/.

[30] A universal labeling tool: Sloth. https://cvhci.anthropomatik.kit.edu/~baeuml/projects/a-universal-labeling-tool-for-computer-vision-sloth/.

[31] Monsoon Power Monitor. https://www.msoon.com/LabEquipment/PowerMonitor.

[32] Nvidia. Nvidia Shield Tablet K1. https://shield.nvidia.com/tablet/k1.

[33] Roberto Hoyle, Robert Templeman, Steven Armes, Denise Anthony, David Crandall, and Apu Kapadia. Privacy behaviors of lifeloggers using wearable cameras. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (Ubicomp)*, 2014.

[34] Tamara Denning, Zakariya Dehlawi, and Tadayoshi Kohno. In situ with bystanders of augmented reality glasses: Perspectives on recording and privacy-mediating technologies. In *ACM Conference on Human Factors in Computing Systems (CHI)*, 2014.

[35] Jaeyeon Jung and Matthai Philipose. Courteous glass. In *UPSIDE, Workshop at ACM International Joint Conference on Pervasive and Ubiquitous Computing (Ubicomp)*, 2014.

[36] Loris D'Antoni, Alan Dunn, Suman Jana, Tadayoshi Kohno, Benjamin Livshits, David Molnar, Alexander Moshchuk, Eyal Ofek, Franziska Roesner, Scott Saponas, Margus Veanes, and Helen J. Wang. Operating system support for augmented reality applications. In *Workshop on Hot Topics in Operating Systems (HotOS)*, 2013.

[37] Suman Jana, Arvind Narayanan, and Vitaly Shmatikov. A scanner darkly: Protecting user privacy from perceptual applications. In *IEEE Symposium on Security and Privacy*, 2013.

[38] Suman Jana, David Molnar, Alexander Moshchuk, Alan Dunn, Benjamin Livshits, Helen J. Wang, and Eyal Ofek. Enabling fine-grained permissions for augmented reality applications with recognizers. In *Usenix Security Symposium (Usenix Security)*, 2013.

[39] Christopher Smowton, Jacob R. Lorch, David Molnar, Stefan Saroiu, and Alec Wolman. Zero-effort payments: Design, deployment, and lessons. In *Proceedings of the ACM International Joint Conference on Pervasive and UbiquitousComputing (UbiComp)*, 2014.

[40] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[41] Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Describing people: Poselet-based attribute classification. In *International Conference on Computer Vision (ICCV)*, 2011.

[42] Ning Zhang, Manohar Paluri, Marc'Aurelio Ranzato, Trevor Darrell, and Lubomir D. Bourdev. PANDA: pose aligned networks for deep attribute modeling. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[43] Zhou Lingli and Lai Jianghuang. Security algorithm of face recognition based on local binary pattern and random projection. In *International Conference on Computational Intelligence (ICCI)*, 2010.

[44] Yongjin Wang and Konstantinos N. Plataniotis. An analysis of random projection for changeable and privacy-preserving biometric verification. *IEEE Transactions on Systems, Man, and, Cybernetics: part B: CYBERNETICS, Vol. 40, No. 5*, 2010.

[45] Per Hallgren, Martin Ochoa, and Andrei Sabelfeld. Innercircle: A parallelizable decentralized privacy-preserving location proximity protocol. In *Proceedings of the 13th Annual Conference on Privacy, Security and Trust (PST)*, 2015.