You Can't See Me: Providing Privacy in Vision Pipelines via Wi-Fi Localization

Shazal Irshad*, Ria Thakkar+, Eric Rozner*, Eric Wustrow*
*University of Colorado Boulder, +Google

***Abstract*—Today, video cameras are ubiquitously deployed. These cameras collect, stream, store, and analyze video footage for a variety of use cases, ranging from surveillance, retail analytics, architectural engineering, and more. At the same time, many citizens are becoming weary of the amount of personal data captured, along with the algorithms and datasets used to process video pipelines. This work investigates how users can opt-out of such pipelines by explicitly providing consent to be recorded. An ideal system should obfuscate or otherwise cleanse non-consenting user data, ideally before a user even enters the video processing pipeline itself. We present a system, called ConsentBox, that enables obfuscation of users without using complex or personally-identifying vision techniques. Instead, a user's location on a video frame is estimated via Wi-Fi localization of a user's mobile device. This estimation allows us to remove individuals from frames before those frames enter complex vision pipelines.**

I. INTRODUCTION

Camera surveillance has recently become commonplace, with the average person being surveyed at a higher rate than they think [1]. Coupled with deep learning advances such as facial recognition and the prevalence of HD cameras, modern vision systems can track a surprising amount of information about users with high precision. The omnipresence of these systems is concerning to the general public and politicians alike [2]. Given cameras are ubiquitously deployed for a variety of use-cases, it's intractable to obtain the consent of every single person observed by the world's network of cameras. This paper studies how users can regain control of their privacy in a tractable and automated fashion: users not consenting to be analyzed by video pipelines should not appear within any vision-based analysis.

Users may wish to opt out of computer vision pipelines for a variety of reasons. One issue is mistrust in new and developing AI technology. For example, inaccurate facial recognition matches have lead to false arrests [3]. Some works have shown racial and gender bias in many vision pipelines and datasets [4]. And while many vision tasks center on personal identification or facial recognition, deep learning models are being analyzed for a variety of *fine-grained* tasks, such as activity recognition [5], lip-reading [6], action anticipation [7], visual keystroke inference [8], emotional recognition [9], social relationship inference [10], and more. Users may be uncomfortable with the use-cases or privacy invasion of these techniques. As data leaks and ransom attacks become more popular, users become more skeptical of data about themselves (such as video recordings) being transmitted, stored, analyzed, or mis-used by third parties. In short, a pressing need for users is the ability to easily and seamlessly provide consent to vision pipelines regarding the inclusion of personal data.

Some potential solutions to providing user consent include sharing the location of all cameras in a given area, whether through an app or as a dataset [11]. Large data aggregates are not only hard to read and understand, but also difficult to gather and maintain. Such datasets require substantial data canvassing, coordination, and upkeep. Even if such a system could be deployed, the onus falls onto the user, and not the vision infrastructure, to maintain privacy because users would have to avoid video-surveilled areas. An alternative solution could remove users from video recordings via techniques like facial recognition, stripping the user from the vision pipeline or its derived dataset after identification. But some users may be uncomfortable with the level of identifying information required to train facial identification and may worry about other issues such as user misidentification, protection of personal data, and being in the vision pipeline at all.

The problem is there is no way for users monitored by a camera-based monitoring system to simply opt out and not have their physical identifying information stored. This paper proposes a system called ConsentBox that enables users to denote their consent, with cooperating vision frameworks automatically obfuscating non-consenting users early in the vision pipeline. The technique can ensure personal data does not enter vision execution frameworks (such as CNNs) and is not transferred, stored, or analyzed via the edge or cloud.

We advocate for co-opting Wi-Fi localization to aid in automatically obfuscating non-consenting users. Wi-Fi localization has been studied for decades, with state-of-the-art accuracies at decimeter-level [12]. Today, Wi-Fi is ubiquitously deployed, making it a good candidate to aid in automated consent adherence. If a user's location is known, along with the coordinates and characteristics of a camera (tilt, rotation, focal lengths, etc), then where a user resides on the image a camera captures can be inferred. Once a user's location on the camera's image is obtained, a variety of obfuscation techniques can be used early in the video pipeline: from cropping a user out of the frame, to drawing opaque bounding boxes around users, to blurring the image. Thus, privacy can be preserved.

Using Wi-Fi in this way has numerous benefits. First, mobile devices are often colocated with a user, serving as an accurate and easy proxy to a user's location. Wi-Fi analysis and localization are extremely light-weight, meaning ConsentBox

979-8-3503-4693-0/23/$31.00 ©2023 IEEE

can be deployed on a variety of sensor and IoT devices. Our techniques could fit into small, trusted codebases run early in the video pipelines, either in software or hardware. As such, ConsentBox is compatible with simple and advanced vision analysis (such as deep learning networks). Although Wi-Fi is increasingly accurate and new technologies like mmWave further increase accuracy, localization errors still exist. Therefore, ConsentBox provides a trade-off: larger regions of image obfuscation can mitigate Wi-Fi error. In these cases, portions of the input frame that do not contain ConsentBox users may be obscured in the name of privacy. In the most conservative case, the whole input frame could be obscured if a ConsentBox user is detected nearby.

By obfuscating users simply and cheaply, ConsentBox can ensure user privacy in two important scenarios: data collection needed for training purposes and real-time analysis used for inference and analytics. In the training data scenario, previous work has shown Membership Inference Attacks can leak information about users included in the training data of neural networks [13], [14]. Therefore, users may not want to be included in images collected for training purposes. In real-time analytics, we envision ConsentBox providing different levels of privacy. In a perfect scenario, ConsentBox totally obfuscates non-consenting users 100% of the time. In reality, ConsentBox could still miss obfuscating a user perfectly (either a portion of a user enters a video pipeline or some frames miss a user). In these cases, however, ConsentBox still provides privacy benefits by thwarting fine-grained vision analysis, such as activity recognition or lip-reading, by omitting important spatial and temporal information gleaned from video input.

The contribution of this paper is using Wi-Fi to automatically obtain user consent in video pipelines. Our work introduces a variety of questions to the research community, such as: What threat models should be considered? What trade-offs exist in providing privacy, while maintaining accuracy? How can user devices integrate with a larger camera ecosystem to automatically infer consent? How accurate are preliminary Wi-Fi-based techniques in indoor environments? And finally, what set of challenges remain to tackle the problem?

II. Background

This section overviews pertinent works required to build ConsentBox and others that have tackled the problem.

Computer vision Today, deep neural networks (DNNs) show great promise on a variety of tasks, from object detection [15], facial recognition, image segmentation, scene explanation, and more. Traditionally, these techniques are computationally expensive and require a significant amount of training data to perform well. Due to the immense resources required, it is common to leverage third-party cloud providers to perform these tasks [16].

Wi-Fi localization Wi-Fi Localization aims to obtain a user's location via angle of arrival, signal strength, time-of-flight, and multipath of Wi-Fi signals. Recent papers claim decimeter-level accuracy even in complicated indoor environments [17], [12]. Readings from multiple APs can be combined to infer location or even a single AP can be used [17]. In addition, a recent addition to the Wi-Fi standard, called Fine Time Measurement (FTM) uses time-of-flight and trilateration from multiple APs to obtain location estimates. The output of Wi-Fi localization is a user's location, or (X,Y,Z) coordinates in a physical 3D space.

Privacy preserving surveillance Previous works have defined systems to enable user privacy in surveillance systems. PrivacyCam [18] produces video streams with different levels of detail, encrypting each stream with separate keys to ensure proper access control. Other approaches utilize movement data and RFID badges to assist in user masking (i.e., privacy) [19], [20]. For example, in [20] users wear RFID badges and motion sensors placed at room entry and exit points initiate an RFID scan when motion is detected. Users are identified by their RFID badge, and users not allowed in a given authorized area prompt retrieval of unaltered video and users who are allowed to be in a certain area are masked. Our work shares the general idea of granting privacy in vision-based systems, but differs in that it uses existing infrastructure (Wi-Fi) to pinpoint user locations.

User obfuscation Several techniques have been proposed to obfuscate users, either within or outside vision pipelines. Within the pipeline, techniques can identify users based on physical characteristics and then remove users from analyzed video. Such techniques often require training data or information about a user– both of which some users may not want to share. Outside of pipelines, wearable devices such as masks or specialized hats have been developed to confuse video processing systems. FacePet [21], a smart-glasses based wearable, extrudes visible light to thwart facial recognition systems. The problem with wearables is they may not be robust against all identification techniques, such as gait analysis. Furthermore, such techniques can place undue burden on users. We investigate simple and universal techniques to opt-out of video processing.

III. Design

This section outlines the design space to enable ConsentBox, breaking down different piece-parts of the system.

Threat model We assume participants (*e.g.,* Alice) wish to remain private from video surveillance comprised of two parts: a trusted *front-end* camera that captures images and is capable of only basic processing, and an untrusted, logical *back-end* that stores images and performs analysis using machine learning models. Alice trusts the front-end will be able to recognize her and not send her images to the back-end, run by an untrusted third-party who may be subject to subpoenas or compromise by attackers. In other words, Alice and the front-end cooperate allowing Alice to opt-out of video surveillance and analysis that may occur off-site.

While the front-end camera could also be compromised or malicious, we assume Alice trusts the owner, as it may be a local business or municipality that she directly interacts with.

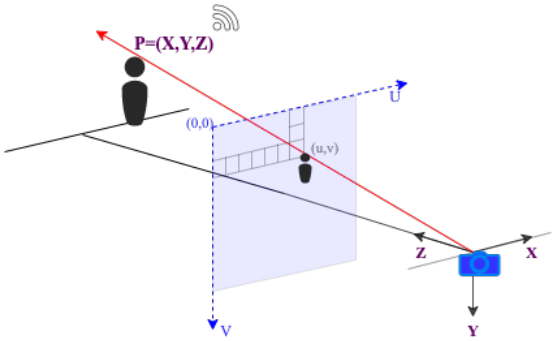

Fig. 1: Forward Projection

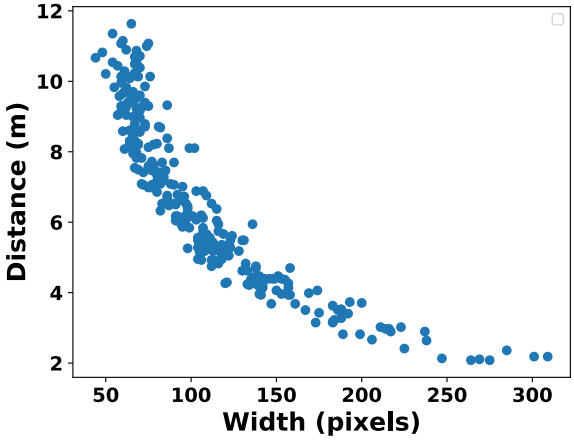

Fig. 2: Distance mapping to bounding box width

We discuss extensions that would allow Alice to remove trust from even the front-end components in Section V.

How can Wi-Fi be used to obfuscate a user? Obfuscation occurs by translating a point in 3D physical space $P(X, Y, Z)$, obtained from Wi-Fi localization estimates, into a point in 2D pixel coordinate space $<u, v>$, captured via a camera. Points are mapped from 3D space to 2D pixel space using a well-known image processing technique called *forward projection* [22], as shown in Figure 1. The equations below show how to derive $<u, v>$ from P using the camera's intrinsic parameters which are known a priori. The easily-obtained intrinsic parameters are focal length, defined in pixels for width (f_x) and height (f_y), and principal point, the center of projection in pixel coordinates (c_x, c_y). The depth τ (in meters) of P from the camera is also required, and can be calculated from P's distance to the camera's known physical position. The following set of equations define forward projection:

$$u = (X \cdot f_x)/\tau + c_x \quad (1)$$

$$v = (Y \cdot f_y)/\tau + c_y \quad (2)$$

Hiding users We call the projected $<u, v>$ point the *Wi-Fi Projected Point*. Once the Wi-Fi Projected Point of a user has been identified, their image can be obfuscated. A user can be blurred, her face can be blocked, or an opaque bounding box can be drawn around the user. We use the bounding box approach, with the box centered on a user's Wi-Fi Projected Point. Bounding boxes can remove features blurring or facial scrubbing may reveal like skin tone, clothing, or body shape.

Although drawing a bounding box seems straightforward, determining the width and height of the bounding box is non-trivial. Too small bounding boxes will not cover the user fully, leaking privacy. Too large bounding boxes may unnecessarily cover other users or important information in the frame. Bounding box sizes are estimated via a data-driven approach: images of individuals are collected in different rooms and varying depths, YOLOv3 [15] is run over all the images to detect users, and the width of the generated bounding box is stored for the corresponding depth. A polynomial regression line is fit over the collected data, which allows us to estimate the width of bounding box from the depth obtained using Wi-Fi localization. Figure 2 shows the box width as depth increases.

Is Wi-Fi localization accuracy sufficient? Recent Wi-Fi localization techniques can localize users with decimeter-level accuracy [17], but such works suffer from long-tail errors due to multi-path, loss, and dynamic conditions. Figure 3 demonstrates these errors by measuring the distance, in pixels, from the Wi-Fi Projected Point to its corresponding ground truth point. We calculate ground truth as the center of a user obtained via YOLOv3. Methodology is discussed in Section IV. The CDF shows reasonable 80^{th}-percentile errors under 200 pixels, but higher tail errors up to 1400 pixels (camera resolution is 1280x720). We allow $<u, v>$ to become negative when Wi-Fi estimates place a user outside the frame, and the largest errors come when a user is close to the camera. In general, Wi-Fi works well but improvements can be made. As Wi-Fi localization increases in accuracy, such gains can easily benefit ConsentBox. However, the large tail errors motivate additional techniques to improve accuracy.

Can accuracy be improved? We ask can camera data itself be used in a way that improves accuracy but also preserves privacy? Camera data must be used carefully, and processing must be extremely light-weight so ConsentBox can be run on a variety of under-powered IoT and sensor devices. We believe there exists a privacy-accuracy trade-off in using image data. On one end, Wi-Fi can solely be used to ensure *no* image data is processed, at the cost of lower accuracy. On the other end, DNNs could accurately infer user information, but at a cost of increased complexity and reduced privacy (*e.g.,* if processing in the cloud). We aim to find a sweet-spot in this trade-off, and use a simple, fast, and lightweight image processing technique that can be easily run on the front-end processor.

Therefore, we utilize *background subtraction* to assist in finding the potential location of a user. Background subtraction is simple: pixels differing from the background can easily be identified. It is also privacy preserving because the algorithms and models involved cannot be easily used to identify specific users. Bounding boxes are derived from an algorithm that

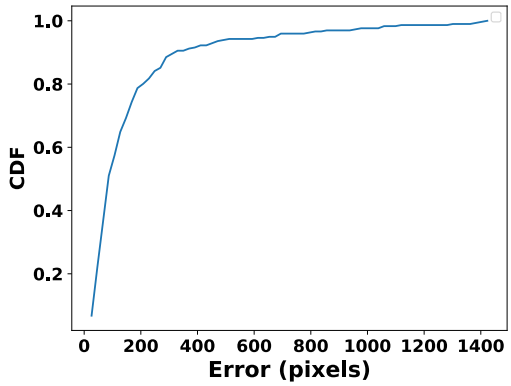

Fig. 3: Error of Wi-Fi's point on an image to the ground truth

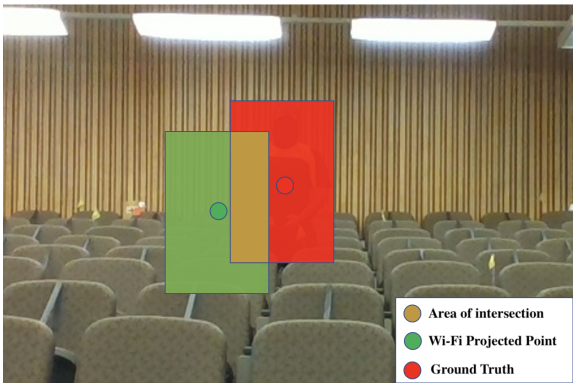

Fig. 4: Intersection over Union example

draws a bounding box around the contours formed from groupings of pixel deltas [23]. Ideally this should give us the only bounding box in which the user is present. However, we find that this background subtraction approach can output many bounding boxes, due to multiple objects in a frame, varying lighting conditions, and changing camera exposure. To solve this, we filter bounding boxes which are smaller than a particular size, which is determined by the room dimensions. From the remaining boxes, the final obfuscation bounding box is selected that minimizes the euclidean distance between the center of the bounding box and the Wi-Fi Projected Point.

How to deal with multiple users in the frame? When multiple users are in frame, it becomes more difficult to tell which Wi-Fi Projected Points should map to which users' bounding box. An extremely conservative approach could obscure all users if a ConsentBox user is detected nearby. A more practical approach tries to obfuscate ConsentBox users only. Therefore, it is important to accurately *map* the identity of a user in the pixel space to the correct identity of user in the physical space.

For mapping, background subtraction is modified to map multiple Wi-Fi Project Points to multiple bounding boxes. A Wi-Fi point is mapped to an associated bounding box using the Hungarian Algorithm [24]. In some instances, multiple users may share a bounding box, especially if the bounding box is large. We are investigating how to improve the current scheme, perhaps by considering user trajectories, dealing with cases in which two users are next to one another, and more precisely matching bounding boxes to estimated user sizes.

Privacy model: how do users integrate with ConsentBox? We need a way for ConsentBox users to declare their consent. This will be achieved by two components: a user application on the mobile device and a secure environment running on the front-end camera processor. Users can select whether they want to obfuscate themselves and this will be communicated to the front-end. A mapping from user to MAC address could allow the system to identify a ConsentBox user. However, users may not want their MAC address analyzed, as the system would then be able to track the user at a coarse-grained level. To overcome this, we can assign a large pool of ConsentBox MAC addresses that phones can randomly utilize. ConsentBox will then hide any user with a MAC address from this pool, without knowing *which* user is being hidden.

IV. Evaluation

This section details the methodology and evaluation results.

Methodology For dataset collection we use a monocular camera and for Wi-Fi localization we use 802.11mc FTM. Our setup consists of three Google APs and a Google Pixel 3 phone. One AP is colocated with the camera and other two APs are typically positioned 10-20 meters away from the first AP. For FTM data we run WiFiRttScan Android application on the phone, 50 FTM readings are collected for each AP, and a minimum of these readings is used for trilateration. We collected data across five different classrooms, with 296 points in total.

To test the method to pair multiple users in a room, multiple images from our dataset are combined. A coordinate pair generator generated random combination pairings and we selected the first 20 pairs.

In order to calculate the accuracy of our system we use Intersection over Union (IoU) metric. IoU is generally used to measure the accuracy of object detection, with ranges [0,1] where 0 being no intersection to 1 being perfect intersection. IoU divides the area of intersection by the total area of union of the bounding boxes. The basic idea can be seen in Figure 4.

Wi-Fi only obfuscation We first present results of Wi-Fi only obfuscation (without background subtraction), shown in Figure 5. We compare the IoU between our computed bounding box (based on Wi-Fi location) and a ground-truth one centered on a manually measured point. We also measure the impact of varying box sizes, with a constant multiplier applied to box dimensions determined by the distance-based model described in Section III.

IoU performs better with increased bounding box size because Wi-Fi errors can be masked. Average IoU values increase from 0.16 for base bounding box size to 0.39 for 2× bounding box size, but it comes with a drawback that

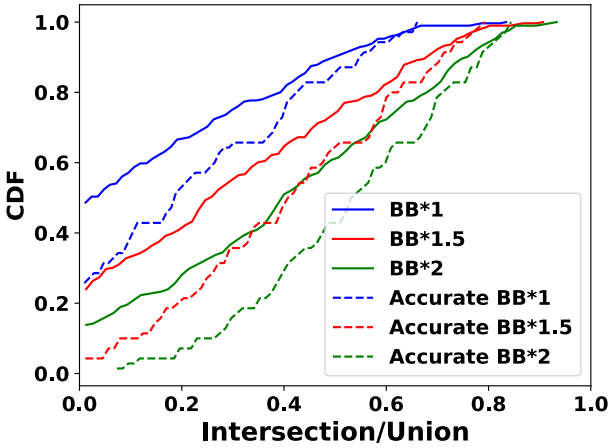

Fig. 5: Intersection over Union error over all Wi-Fi estimates and a subset of accurate Wi-Fi estimates

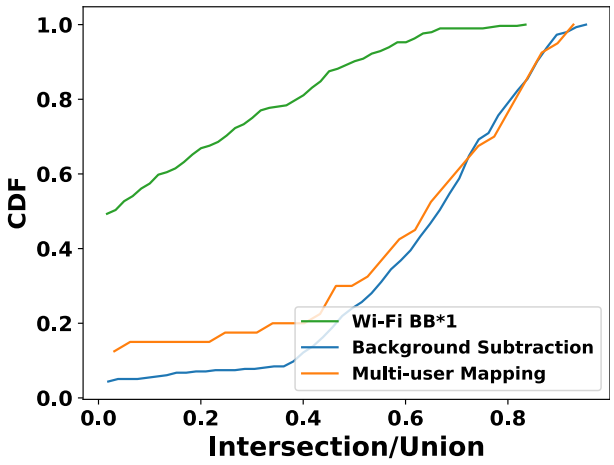

Fig. 6: Intersection-over-Union with Background Subtraction optimization

bigger bounding boxes cover more area in which a user is not present. We also measure results for a subset of data where Wi-Fi performs well (error of less than 0.6 meters), denoted *Accurate* in the graph. With good accuracy, Wi-Fi only obfuscation gives an average IoU value of 0.22 for the base bounding box size and 0.52 for 2× size. This shows improvements in Wi-Fi localization can be used to improve ConsentBox accuracy. Overall, these results show a trade-off for ConsentBox configuration: larger multipliers provide better privacy at the cost of obfuscating more area of an image.

Wi-Fi + background subtraction obfuscation Figure 6 shows the CDF for IoU when we apply the background subtraction method, with an average improvement of 3.81× compared to Wi-Fi only obfuscation, and 2.77× as compared to accurate Wi-Fi only obfuscation. Compared to the previous Wi-Fi results, the CDF is generally improved (shifted to the right), with IoU above 0.5 occurring 75.67% of the time, showing simple background subtraction can improve accuracy.

Multiple users The mapping algorithm's accuracy is 85% (17/20 cases mapped correctly) in the multi-user dataset. Figure 6 shows the CDF for IoU, again using Wi-Fi + background subtraction. While accuracy is lower than with an individual user, it is still better than Wi-Fi only. We plan to analyze larger and more complex multi-user datasets in future work.

V. Discussion and Challenges

This section provides discussion and future challenges.

ConsentBox use-cases ConsentBox *is not* to be used in instances for public safety and security. Instead, we envision ConsentBox's usage in scenarios were users do not need to be individually identified. Examples include retail analytics, traffic monitoring, urban planning, parking occupancy, architectural engineering, and more. In the future, local ordinances could specify where ConsentBox should be deployed and vision pipelines could even be ConsentBox-certified.

Video-based analysis and accuracy Our current analysis pipeline is light-weight and easily handles current frame-rates needed for video analysis. Currently, ConsentBox may fail, or partially fail, to obscure a user for a single frame. There are numerous ways to address this challenge. First, ConsentBox can slightly delay video frames in order to smooth and expand bounding boxes over multiple frames. Modelling user trajectories could improve accuracy by utilizing the aforementioned smoothing and expanding techniques. Second, ConsentBox could intelligently sample which obfuscated video frames are exported to the vision pipeline, perhaps using confidence in Wi-Fi measurements to make decisions (*e.g.,* confidence could be derived by analyzing the variance of FTM readings or output directly [12]). Confidence in Wi-Fi readings could also be used to determine bounding box sizes. Last, fusing more information from the user (such as IMU data) can improve location estimates [25]. In short, even without 100% accuracy ConsentBox can still impair fine-grained vision tasks such as activity recognition or lip reading by obfuscating important temporal and spatial video data required by these fine-grained vision tasks.

Changing environment An important part of ConsentBox is background subtraction. While it gives good results in static environments, gaining high accuracy can be a challenging in dynamic environments. Constantly changing environments can be dealt with by keeping up-to-date daily background images. For highly dynamic environments, like malls, using information across multiple adjacent frames can help in more accurate background subtraction results [26].

Wi-Fi measurement rate ConsentBox's accuracy is dependent on Wi-Fi localization, and localization accuracy can be impacted by the measurement rate. High measurement rates typically provide better accuracy. Users can install a ConsentBox app on their phone which could explicitly send localization probes. Even without explicit probes, several Wi-Fi localization techniques work with single packets [12].

User buy-in and choice We've presented a system allowing users to opt-out. We also envision the opposite, where ConsentBox could block *all* users (based on background subtraction), except those who opt-in. We also plan to explore different levels of privacy that can be granted, with perhaps the system providing SLAs to block the user a percentage of the time, or only in certain regions or times of day. Users could also specify different preferences (*e.g.,* totally obfuscate user versus prevent fine-grained activity recognition), which could then help configure bounding box sizes. This future work could involve user studies to understand what levels of obfuscation are acceptable to people under different scenarios.

Secure enclave Our design assumes the front-end camera processor is trusted by participants. However, this trust could be replaced with remote attestation systems like Intel SGX. If ConsentBox were implemented in an SGX secure enclave running on the camera, users could receive a remote attestation that cryptographically proves it is running ConsentBox. While previous systems have proposed implementing sensitive DNN operations inside SGX enclaves entirely [27], these memory-intensive workloads can have high overheads which could limit their utility. In addition, users would have to allow a variety of DNN functions via remote attestation; in contrast, allowing just the ConsentBox implementation would protect their privacy while enabling a wide variety of back-end applications.

VI. Conclusion

Video surveillance systems are becoming widespread, but today's users have little or no ability to consent to being a part of a computer vision pipeline. As society's views of surveillance, vision-based AI, and equality in algorithm and datasets evolve, we believe the time has come to enable users to give consent and take back their privacy. As a result, we present a system called ConsentBox in which a user's mobile device can be used to obfuscate a user within an image frame by mapping real-world user coordinates, obtained via Wi-Fi localization, to pixel values. We show how forward projection can be used to mask users and highlight design points of the system. An evaluation shows the ConsentBox accuracy, and how simple image analysis can make obfuscation more accurate. Finally, discussion is provided on future work.

Acknowledgements This work is partially funded by NSF-1908910.

References

[1] A. Doyle, R. Lippert, and D. Lyon, *Eyes everywhere: The global growth of camera surveillance*. Routledge, 2013.

[2] M. Madden and L. Rainie, "Americans' attitudes about privacy, security and surveillance," Aug 2020. [Online]. Available: https://www.pewresearch.org/internet/2015/05/20/americans-attitudes-about-privacy-security-and-surveillance/

[4] J. Buolamwini and T. Gebru, "Gender shades: Intersectional accuracy disparities in commercial gender classification," in *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*. PMLR, 23–24 Feb 2018. [Online]. Available: https://proceedings.mlr.press/v81/buolamwini18a.html

[3] K. Hill, "Another arrest, and jail time, due to a bad facial recognition match," *The New York Times*, 2020. [Online]. Available: https://www.nytimes.com/2020/12/29/technology/facial-recognition-misidentify-jail.html

[5] F. Karim Et al., "Multivariate LSTM-FCNs for time series classification," *Neural Networks*, vol. 116, pp. 237–245, aug 2019. [Online]. Available: https://doi.org/10.10162Fj.neunet.2019.04.014

[6] A. Fernandez-Lopez and F. M. Sukno, "Survey on automatic lip-reading in the era of deep learning," *Image and Vision Computing*, vol. 78, pp. 53–72, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0262885618301276

[7] J. Liang, L. Jiang, J. C. Niebles, A. Hauptmann, and L. Fei-Fei, "Peeking into the future: Predicting future person activities and locations in videos," 2019.

[8] J. Lim, T. Price, F. Monrose, and J.-M. Frahm, "Revisiting the threat space for vision-based keystroke inference attacks," 2020.

[9] H.-Q. Khor, J. See, R. C. W. Phan, and W. Lin, "Enriched long-term recurrent convolutional network for facial micro-expression recognition," 2018.

[10] J. Li, Y. Wong, Q. Zhao, and M. S. Kankanhalli, "Dual-glance model for deciphering social relationships," 2017.

[11] T. Winkler and B. Rinner, "User-centric privacy awareness in video surveillance," *Multimedia Systems*, vol. 18, no. 2, pp. 99–121, 2012. [Online]. Available: https://doi.org/10.1007/s00530-011-0241-1

[12] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "Spotfi: Decimeter level localization using wifi," in *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, ser. SIGCOMM '15. New York, NY, USA: ACM, 2015, pp. 269–282. [Online]. Available: http://doi.acm.org/10.1145/2785956.2787487

[13] A. Salem, Y. Zhang, M. Humbert, M. Fritz, and M. Backes, "Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models," *CoRR*, vol. abs/1806.01246, 2018. [Online]. Available: http://arxiv.org/abs/1806.01246

[14] F. Mo et al., "Darknetz: Towards model privacy at the edge using trusted execution environments," *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, Jun 2020. [Online]. Available: http://dx.doi.org/10.1145/3386901.3388946

[15] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *CoRR*, vol. abs/1804.02767, 2018. [Online]. Available: http://arxiv.org/abs/1804.02767

[16] S. Naderiparizi Et al., "Glimpse: A programmable early-discard camera architecture for continuous mobile vision," in *MobiSys '17*. Association for Computing Machinery, 2017. [Online]. Available: https://doi.org/10.1145/3081333.3081347

[17] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single wifi access point," in *NSDI 2016)*. [Online]. Available: https://www.usenix.org/conference/nsdi16/technical-sessions/presentation/vasisht

[18] A. Senior Et al., "Enabling video privacy through computer vision," *IEEE Security Privacy*, vol. 3, no. 3, pp. 50–57, 2005.

[19] S.-c. Cheung Et al., *Protecting and Managing Privacy Information in Video Surveillance Systems*, 06 2009, pp. 11–33.

[20] J. Wickramasuriya Et al., "Privacy protecting data collection in media spaces." New York, NY, USA: Association for Computing Machinery, 2004. [Online]. Available: https://doi.org/10.1145/1027527.1027537

[21] A. J. Perez Et al., "Facepet: Enhancing bystanders' facial privacy with smart wearables/internet of things," *Electronics*, vol. 7, no. 12, 2018. [Online]. Available: https://www.mdpi.com/2079-9292/7/12/379

[22] D435i, "Forward projection in camera." [Online]. Available: https://dev.intelrealsense.com/docs/projection-in-intel-realsense-sdk-20

[23] "Opencv contours," https://docs.opencv.org/4.5.3/d4/d73/tutorial_py_contours_begin.html

[24] H. W. Kuhn and B. Yaw, "The hungarian method for the assignment problem," *Naval Res. Logist. Quart*, pp. 83–97, 1955.

[25] A. T. Mariakakis, S. Sen, J. Lee, and K.-H. Kim, "Sail: Single access point-based indoor localization," ser. MobiSys '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 315–328. [Online]. Available: https://doi.org/10.1145/2594368.2594393

[26] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, "Subsense: A universal change detection method with local adaptive sensitivity," *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 359–373, 2015.

[27] F. Tramèr and D. Boneh, "Slalom: Fast, verifiable and private execution of neural networks in trusted hardware," 2019.