No Peeking through My Windows: Conserving Privacy in Personal Drones

Alem Fitwi[1], Yu Chen[1], Sencun Zhu[2]
[1]Dept. of Electrical and Computer Engineering, Binghamton University, SUNY, Binghamton, NY 13902, USA
[2]Department of Computer Science and Engineering, Penn State University, University Park, PA 16802, USA
Emails: {afitwi1, ychen}@binghamton.edu, sxz16@psu.edu

***Abstract*—The drone technology has been increasingly used by many tech-savvy consumers, a number of defense companies, hobbyists and enthusiasts during the last ten years. Drones often come in various sizes and are designed for a multitude of purposes. Nowadays many people have small-sized personal drones for entertainment, filming, or transporting items from one place to another. However, personal drones lack a privacy-preserving mechanism. While in mission, drones often trespass into the personal territories of other people and capture photos or videos through windows without their knowledge and consent. They may also capture video or pictures of people walking, sitting, or doing private things within the drones' reach in clear form without their go permission. This could potentially invade people's personal privacy. This paper, therefore, proposes a lightweight privacy-preserving-by-design method that prevents drones from peeking through windows of houses and capturing people doing private things at home. It is a fast window object detection and scrambling technology built based on image enhancing, morphological transformation, segmentation and contouring processes (MASP). Besides, a chaotic scrambling technique is incorporated into it for privacy purpose. Hence, this mechanism detects window objects in every image or frame of a real-time video and masks them chaotically to protect the privacy of people. The experimental results validated that the proposed MASP method is lightweight and suitable to be employed in drones, considered as edge devices.**

***Keywords*-Privacy, Light-weight Window-Object-Detection, Personal Drones, Chaotic Scrambling, Edge Computing.**

I. Introduction

The world has seen a rampant advancement of unmanned aerial vehicles (UAV) technology, also known as drones, over the last decade. They come in a variety of size, sophistication, and they take many roles. Today, they have a wider range of applications in transportation, search and rescue, military, surveillance, communication relays, filming, entertainment, and monitoring [8], [10], [26], [36], [39]. As a result, the public and private sectors, and individuals have shown growing interest in using the drone technologies in a way that serves their purposes. This is likely to cause proliferation of drones in the sky and these drones are capable of garnering a lot of private information about people and places [3]. While hovering in the sky, drones can be directed to collect personal information, monitor and spy people.

Furthermore, this information can be divulged into the wider cyberspace because drones are vulnerable to a range of attacks. The owners might not have full control of their drones. Drones are vulnerable to rudimentary interception and interruption attacks. A number of attacks on drones like video stealing, injection of malwares, and device hijacking have been reported since 2007 [2], [8], [13]. It was clearly demonstrated that some commercially available WiFi based drones and satellite based military-grade ones are vulnerable to basic security attacks [7], [8], [12]. This has the potential to cause the widespread of sensitive personal data garnered by the drones without the owners' permission into the cyberspace. For this reason, people have been growing more and more paranoid about their privacy in relation to the use of drones. To some people, it feels like drones and the invasion of privacy are synonyms.

In reality, personal or civilian drones have the capability to pick up virtually every information about what is happening in a certain specific scene. They can capture the images and record the footage of people in that scene without having their sanction to record them. The privacy of people is therefor at risk due to the fact that information, video or images unauthorizedly captured by drones' cameras and sensors could be abused by the owner of the drone and attackers who manage to penetrate into the drones [7], [17], [23], [34]. As a result, there have been a number of moves to enact laws and regulations to protect privacy and ensure safety in a bunch of developed countries where drones are widely used. For instance, UK, USA, and Canada have recently tried to legislate some laws and regulations in an attempt to address the privacy issues and to ensure safety in the aftermath of some incidents [15], [18], [28], [30], [33]. However, this is not enough to preserve privacy. In fact, they have failed to address the burning privacy concerns.

This paper introduces a privacy-conserving technique for personal drones to balance out their use and privacy. A privacy-preserving mechanism based on germane image processing technique is proposed where the personal drones are considered as edge devices with single board computers (SBC) like Raspberry PI 3 B^+. It detects window objects in images or real-time video frames and automatically scrambles the windows to prevent peeking through them in violation of the privacy rights of people inside the house. It focuses on enabling the construction of privacy-aware drone systems by design. The window-object-detection and scrambling method are proposed and designed based on a less resource-intensive and faster morphological and segmentation process (MASP) that exploits the very nature of windows. The scrambling method incorporated as part and parcel of the MASP is designed based on a random chaos. Hence, our proposed

978-1-7281-0846-9/19/$31.00 ©2019 IEEE

MASP system is capable of detecting and scrambling window objects in every image or real-time video frames captured by drones.

The remainder of this paper is then organized as follows. Section II presents background knowledge and previous work related to this paper. The window objects detection through morphological and segmentation techniques is introduced in Section III. Section IV reports the experimental results. The conclusions and future works are then presented in Section V.

II. Background Information And Related Works

Drones' popularity has increased beyond measures and they are employed in a number of areas including filming, relaying wireless connections, monitoring, search and rescue, military, transportation, and surveillance [8], [10], [12], [14], [39]. But most of them are not designed and manufactured with serious consideration of privacy and security. They are prone to basic attacks and the information they collect and carry can be compromised. Researchers have showed that personal drones are vulnerable to cache-poisoning and buffer overflow that have the potential of causing Denial of Service (DoS) attacks. The penetration tests that were conducted on personal drones like the Wireless Parrot Bebop UAVs revealed the aforementioned vulnerabilities [10], [12], [13]. Some reported cases like the interception of live feeds from US drones by Iraqi militants who managed to access and watch captured videos on their laptops using a $26 worth software shows that even military-grade drones are vulnerable [5], [23]. Hence, the information garnered and carried by cameras and sensors on personal drones operated by hobbyists and enthusiasts can be intercepted by attackers. That is, security problems of drones increases the degree of breaches of privacy of people whose personal details have been captured by such drones. The personal data might be disseminated into the vast cyberspace and become accessible to many wrong hands on top of the owners of the drones. Consequently, there have been ongoing efforts to bring about both legislative and technological solutions to the serious privacy issues in relation to the prolifically growing use of drones.

In parallel to the regulatory moves, endeavors have been underway to address the privacy issues of drones technologically by design. Efforts were exerted to develop security and privacy aware frameworks for drones [13], [10], [14], [39]. Several security challenges that endanger the privacy of drones were studied and analyzed, which led to the proposal of cybersecurity threat models. The models play very pronounced roles in aiding both users and designers of UAVs in understanding the possible threats with ease, vital for implementing some solutions. They, however, lack the ways and means for ensuring security and preserving privacy. They are only underlying frameworks on which one can build security and privacy solutions. In addition, there are attempts to leverage the advancement in machine learning technology, chaotic cryptography and image scrambling techniques to conserve privacy in drones and surveillance systems [1], [4], [31], [35], [38]. These works demonstrate how to build privacy-aware real-time video analytics in the surveillance systems; they are compute-intensive and impractical to be deployed on the edge, though. There are also many video and image scrambling methods [6], [9], [22], [40]. But they need to be redesigned for drones where resources are constrained. Besides, the Morphological techniques [29], [27] could be vital in processing images containing regularly shaped objects.

III. MASP: Windows Detection and Scrambling

By their very nature, windows comprise horizontal and vertical edges, and some of them contains arcs. They could also be multi-framed comprising multiple vertical and horizontal edges and lines. The edges are represented by changes from one class of pixels to another, and basically refers to the boundaries of windows. Whereas the lines refer to one class of pixels interlaced between two sets of the same class of pixels. They are the metal or wooden lines that partition a window into multiple panes. Besides, in most cases they are rectangular or closely rectangular shaped or combined rectangular and semi-circular shapes. Hence, in this paper we introduce a window detection and scrambling technique that exploits these natures of windows, which is a faster and computing-resources conscious method for resource-limited environments like drones. It is designed based on low-level, medium-level, and high-level image manipulations and processing techniques. The major steps include image enhancing, morphological processing, semantic segmentation, recognition, and scrambling.

A. Image/Video Frames Pre-processing

Following the acquisition of images as they are and videos split into frames, they are converted into gray-scale pictures in order to reduce the processing complexity. The assumption here is that images or video frames contain windows and walls. Hence, they are bi-modal in that they contain two classes of pixels, namely the window (foreground) pixels and the wall (background) pixels. A method of thresholding on the gray-scale image/frame [20] is adopted to compute the optimum threshold that separates the two classes. It exhaustively searches for the threshold that cuts down on the intra-class variance defined by Eq. (1) where variances of each class is multiplied by the class weight.

$$\delta^2_{overall}(t) = w_{win}(t)\delta^2_{win}(t) + w_{wall}(t)\delta^2_{wall}(t) \quad (1)$$

$$f(x,y)_{INV} = f(x,y)_{gray} - 255 \quad (2)$$

Then, inverting the gray image ($f(x,y)_{gray}$) using Eq. (2) after the binarization has been applied gives a white window and a black wall or background as portrayed in Fig. 1. The inversion is done by subtracting 255 from every pixel.

B. Morphological Transformations

Then, features important for detecting the window object are extracted from the inverted image using a morphological process performed based on the shape of target objects in the image. It works with a kernel that slides around to eliminate noises. That is, we employed rectangular morphology for detecting the horizontal and vertical lines and edges useful to detect the windows in the image because most windows have approximately rectangular shapes. The window object

Fig. 1. Image/Frame pre-processing, $f(x,y)_{input}$ vs $f(x,y)_{INV}$

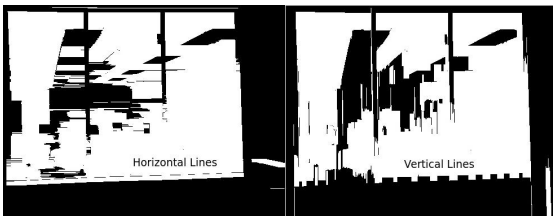

Fig. 2. Horizontal versus vertical lines.

dimensions are assumed to be smaller than that of the full image. In most energy efficient buildings' design, the maximum window-to-wall ratio (WWR) is 60%. Hence, basically we adopted 3x3 kernel whose horizontal and vertical lengths are (image_width-8) and (image_height-8), respectively. These selections were made following a string of tests and consultations to building standards. Figure 2 portrays the output of the morphological transformation which combines the images of detected horizontal and vertical lines based on two weights, each assigned a value of 0.5.

C. Semantic Segmentation

Pre-processed images are then divided into segments that contain meaningful objects, window objects in this case. That is, the segmentation process groups together the pixels that have similar attributes. For we are working on bi-modal images where there are two classes, windows and walls, a semantic segmentation is employed. The pixels that belong to the window object are represented by one color and those pixels that belong to the wall or background class are represented by another class of color.

Following the segmentation process, we used a contouring technique based on the Suzuki algorithm [32] to extract the location coordinates of all window objects on every input image/video frame. Figure 3 gives two input images: the one on top containing only a window and the other (on the bottom) containing three window objects. Figure 4 illustrates the correctly extracted windows from the input images.

Fig. 3. Two input images to the system [top and bottom].

Fig. 4. Segmented and Extracted Windows.

D. Window Objects Recognition and Scrambling

Now that the coordinates of the window objects in the frames are known, they can be easily located and recognized. The next step is scrambling all windows to eschew any look-through. The scrambling process could be done using simple image denaturing processes but we have opted to employ a more robust and reversible chaotic scrambling.

The scrambling is then done using chaotic images generated by solving a system of differential equations stated in Eq. (3)

and Eq. (4) [9].

$$\frac{dy}{dt^2} = 2\alpha\frac{dx}{dt} - x \tag{3}$$

$$\frac{d^2x}{dt^2} - 2\alpha\frac{dx}{dt} + x = 0 \tag{4}$$

Eq. (4) is a homogeneous equation of the form described in Eq. (5) where $f(x) = 0$.

$$a\frac{d^2x}{dt^2} + b\frac{dx}{dt} + cx = f(x) \tag{5}$$

It produces chaos only if its solutions are complex. Then, solving Eq. (4) using this requirement and simplifying it using Euler's formula, it produces the system of solutions stated in Eqs. (6) and (7).

$$x(t) = e^{\alpha t}[x_0 cos(\beta t) + (\frac{y_0 - \alpha x_0}{\beta})sin(\beta t)] \tag{6}$$

$$y(t) = e^{\alpha t}[y_0 cos(\beta t) + \frac{\alpha}{\beta}(y_0 - \alpha x_0 - \frac{\beta^2}{\alpha}x_0)sin(\beta t)] \tag{7}$$

where x_0 and y_0 are initial conditions, and α and β are constants related to each other by Eq. (8):

$$\beta = \sqrt{1 - \alpha^2} \tag{8}$$

The α and β values must be carefully selected so as to produce a complex solution and then a random enough chaos. After researching on a range of values, we eventually chose $\alpha = 0.005$, and then β was computed to be 0.999 using Eq. (8). With these values, we were able to produce a randomized chaos. The good thing about a chaotic scrambling is that it could be reproduced using the same initial conditions (x_0 and y_0) and constant values (α and β) whenever deemed necessary. The chaotic generator is more like the pseudo-random generators where the initial conditions and constants serve as the seed value.

Now the privacy-conserving mechanism is ready to detect window objects on any image or video frame by means of morphological transformation, semantic segmentation, and the Suzuki contouring algorithm. Then, it performs the scrambling of window objects through chaotic mixing for privacy reasons. The chaotic scrambling technique is embedded into the privacy-preserving mechanism and it automatically scrambles once a window object has been detected on the input frames.

IV. Experimental Settings and Results

A. Experimental Setup

At last, we carried out a number of experiments and tests treating the drones as edge devices with constrained resources to verify our proposed model. Then, in terms of the experimental setup, a single board computer (SBC) is employed. It is a Raspberry PI 3 Model B$^+$ with 1.4GHz Cortex-A53 (ARMv8) 64-bit quad core CPU and 1GB DDR2 RAM. It is also fitted with a camera for real-time video analytics testing. Besides, we run our model on an old laptop with Intel(R) Core(TM) i5-3337U CPU @ 1.80GHz and 3938MB System memory in an effort to observe how the performance varies with respect to computing device changes. The testing was conducted on a Raspberry Pi 3 B$^+$ connected to a live camera to produce the result on Fig. 6.

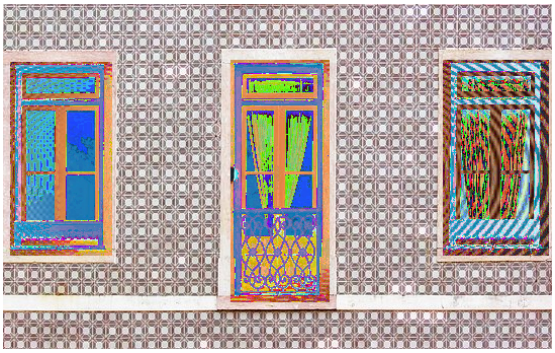

Fig. 5. Sample image test result.

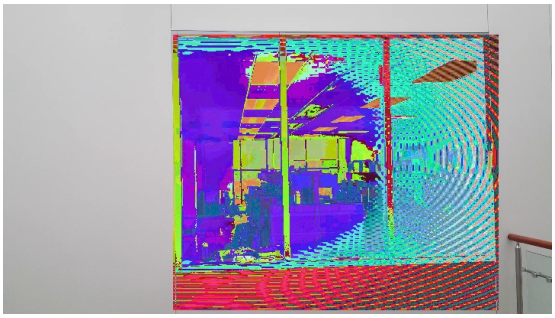

Fig. 6. Sample real-time video test result.

B. Results and Discussion

To verify the functionality, speed, and accuracy of our proposed model, we conducted extensive tests using 17,400 images, recorded and real-time videos of buildings containing various types of windows. Out of the 17,400 images, 11,600 of them are positive images that contain windows of different style, color, size, and varying positions. The rest 5,800 images are dummy walls or negative images that don't contain windows. For demonstration purpose, the video frame and image in Fig. 3 are used. The test image (bottom) is one of the many different types of windows used and the video is one of the many different videos recorded at different scenes and tested live in real-time. As can be seen from the sample test image result in Fig. 5, all of the three windows were effectively detected and scrambled. Likewise, all window objects in every frame of a video captured in a scene in our campus were successfully detected and scrambled. The sample result is portrayed in Fig. 6. Our method is primarily designed to make window objects detection and automatic scrambling in real-time video analytics.

In the MASP scheme, a window object is detected based on basic image processing, morphological transformation, semantic segmentation, and contouring process. The results are very promising. It can process more than 30 frames per second (FPS) on average on the old laptop described in section IV-A and more than 8 FPS on the Raspberry PI 3 B$^+$ as portrayed in table I. The weights used in our experiments for the window and background image classes are $w_{win} = 0.5$ and $w_{bgd} = 0.5$, respectively. The only shortcoming observed is that it considers a window object in an image or video frame captured from very slant angles as multiple windows. We found out that a window object captured at angle greater than 60^o clockwise or counter-clockwise from the normal line to the

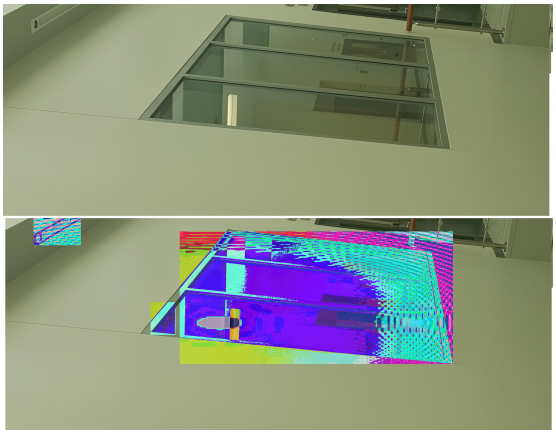

Fig. 7. A window object captured at slant angle and scrambled multiple times.

window surface is treated as if comprising multiple windows and perform the scrambling multiple times as portrayed in Fig. 7. Small window-like non-window shapes are also detected as windows and scrambled as illustrated in Fig. 7. But it is still a very decent method because when the distance from which such windows are shot becomes longer, the small window-like shapes are less likely to be detected. In case of drones, the minimum distance as required by law from any property is 50 meters. We also measured the accuracy of the window objects detection based on the tests carried out on the 17,400 images and a number of real-time videos. It detects windows 100% at 8.69 FPS but we found out that it has an average false positive rate (FPR) of 10.67%.

TABLE I
NUMBER OF FPS REQUIRED FOR LAPTOP AND RASPBERRY PI

Machine	**Laptop**	**Raspberry PI 3 B$^+$**
FPS	*33.25*	*8.69*
FPR	N/A	*10.67%*

For comparative analysis, we have researched on preexisting algorithms including histogram oriented gradients (HOG) features based methods like Support Vector Machine (SVM) classifier coupled with HOG features [21], Convolutional neural networks (CNN) based methods like Faster Region-based CNN (Faster R-CNN) [25] and Spatial Pyramid Pooling(SPP-net) [11], regression-based object detectors like You only Look Once version 3 (YOLOv3)[24] and the Single Shot Detector(SSD) [16], and the the Haar-cascade network [37]. They are very promising methods; however, they are compute-intensive in that they cannot run well at the edge, a resource-constrained environment.

We discovered that most of these methods can process images or video frames at less than 2 frame per second (fps) on a Raspberry Pi 3 B$^+$. A benchmark made on how much time each of the models can take to make a prediction on a new image shows that the fastest method is the SDD_mobilenet_v1 which can process 1.39 frames per second or it takes 0.72 seconds for processing an image on the Raspberry Pi. However, there is a more recently (in 2018) developed light-weight CNN-based object detector [19] where the fps is about 1.79 when run on an edge device. Harkening back to our experiments, we trained the HOG+SVM, and Haar-Cascade networks using the dataset we created and found out the results portrayed in table II. The results substantiate that our proposed model is much faster on the edge than any object-detection method made available in the public domain of the Internet. It is computing resource-conscious method of window-object-detection and scrambling. But there is a room for improving its FPR.

TABLE II
PERFORMANCE MEASURES IN TERMS OF FPR AND FPS.

Methods	**False Positives Rate**	**Frames Per Second**
MASP	*10.67%*	*8.69*
HOG+SVM	*13.23%*	*0.65*
Haar	*22.71%*	*1.93*

In summary, our experimental results corroborate the feasibility and suitability of our proposed model for drones, considered as as edge devices. It is capable of processing a decent FPS on a resource-constrained environment, where there are only a CPU of 1.4GHz and a memory of 1GB.

V. CONCLUSIONS AND FUTURE WORKS

Passing laws and imposing regulations are vital for safe operation of drones and for creating non-flying sky zones. However, they cannot efficiently address the invasion and violation of the privacy rights of individuals in relation to the prolific use of drones. They give the slightest concern to privacy. The best solution is, therefore, to incorporate a privacy-conserving mechanism into the drones by design. This has the potential to play very pronounced role in solving the issues of privacy without impeding the technological advancement of drones. Thus, this paper proposes a method to address the the privacy issues by design.

The method we proposed for conserving privacy in personal drones take advantages of the nature of windows, image processing, and simple edge detection techniques. As a result, it is faster. The light-weight MASP can process more than 8 fps on a resource-constrained Raspberry PI 3 B$^+$.

The contents reported in this paper are results of our preliminary study. It does not handle windows with stylish shapes, i.e. with arc, or being embedded in a more complex walls or houses. Also, it suffers from a relatively high false positive rate (10.67%). Being aware of these weaknesses, our on-going efforts are exploring a machine learning approach that leverages the powerful convolutional neural networks to improve the detection accuracy in more complex scenarios and a great deal of windows with different styles. For this purpose, a focus is being given to the design of a lightweight machine learning algorithm that can tackle the resource constraints on the drones. Beyond windows, the detection and denaturing of human faces or other sensitive parts of the human body are investigated to address other privacy concerns, such as personal identities, children protection, etc.

REFERENCES

[1] M. Al-Rubaie and J. M. Chang, "Privacy-preserving machine learning: Threats and solutions," *IEEE Security & Privacy*, vol. 17, no. 2, pp. 49–58, 2019.

[2] C. Arthur, "Skygrabber: the $26 software used by insurgents to hack into us drones," *The Guardian*, vol. 17, 2009.

[3] A. Cavoukian, *Privacy and drones: Unmanned aerial vehicles*. Information and Privacy Commissioner of Ontario, Canada Ontario, 2012.

[4] K. Chaudhuri and C. Monteleoni, "Privacy-preserving logistic regression," in *Advances in neural information processing systems*, 2009, pp. 289–296.

[5] A. Cuadra and C. Whitlock, "How drones are controlled," *The Washington Post*, vol. 20, 2014.

[6] F. Dufaux, "Video scrambling for privacy protection in video surveillance: recent results and validation framework," in *Mobile Multimedia/Image Processing, Security, and Applications 2011*, vol. 8063. International Society for Optics and Photonics, 2011, p. 806302.

[7] J. Fifield, "How drones raised privacy concerns across cyberspace," 2016.

[8] A. Fitwi, Y. Chen, and N. Zhou, "An agent-administrator-based security mechanism for distributed sensors and drones for smart grid monitoring," in *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVIII*, vol. 11018. International Society for Optics and Photonics, 2019, p. 110180L.

[9] A. H. Fitwi and S. Nouh, "Performance analysis of chaotic encryption using a shared image as a key," *Zede Journal*, vol. 28, pp. 17–29, 2011.

[10] M. Gharibi, R. Boutaba, and S. L. Waslander, "Internet of drones," *IEEE Access*, vol. 4, pp. 1148–1162, 2016.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[12] M. Hooper, Y. Tian, R. Zhou, B. Cao, A. P. Lauf, L. Watkins, W. H. Robinson, and W. Alexis, "Securing commercial wifi-based uavs from common security attacks," in *MILCOM 2016-2016 IEEE Military Communications Conference*. IEEE, 2016, pp. 1213–1218.

[13] A. Y. Javaid, W. Sun, V. K. Devabhaktuni, and M. Alam, "Cyber security threat analysis and modeling of an unmanned aerial vehicle system," in *2012 IEEE Conference on Technologies for Homeland Security (HST)*. IEEE, 2012, pp. 585–590.

[14] R. Kinge, P. Gawande, A. S. ingle, and S. Badhe, "Internet of drones," *nternational Journal of Research in Advent Technology (IJRAT) (E-ISSN: 2321-9637)*, 2017.

[15] M. R. Koerner, "Drones and the fourth amendment: Redefining expectations of privacy," *Duke LJ*, vol. 64, p. 1129, 2014.

[16] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[17] N. McKelvey, C. Diver, and K. Curran, "Drones and privacy," in *Unmanned Aerial Vehicles: Breakthroughs in Research and Practice*. IGI Global, 2019, pp. 540–554.

[18] G. S. McNeal, "Drones and aerial surveillance: Considerations for legislators," *Brookings Institution: The Robots Are Coming: The Project on Civilian Robotics*, 2014.

[19] S. Y. Nikouei, Y. Chen, S. Song, R. Xu, B.-Y. Choi, and T. Faughnan, "Smart surveillance as an edge network service: From harr-cascade, svm to a lightweight cnn," in *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*. IEEE, 2018, pp. 256–265.

[20] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[21] Y. Pang, Y. Yuan, X. Li, and J. Pan, "Efficient hog human detection," *Signal Processing*, vol. 91, no. 4, pp. 773–781, 2011.

[22] S. M. M. Rahman, M. A. Hossain, H. Mouftah, A. El Saddik, and E. Okamoto, "A real-time privacy-sensitive data hiding approach based on chaos cryptography," in *2010 IEEE International Conference on Multimedia and Expo*. IEEE, 2010, pp. 72–77.

[23] B. Rao, A. G. Gopi, and R. Maione, "The societal impact of commercial drones," *Technology in Society*, vol. 45, pp. 83–90, 2016.

[24] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[25] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[26] A. Roder, K.-K. R. Choo, and N.-A. Le-Khac, "Unmanned aerial vehicle forensic investigation process: Dji phantom 3 drone as a case study," *arXiv preprint arXiv:1804.08649*, 2018.

[27] P. J. Salembier Clairon and M. Pardàs Feliu, "Hierarchical morphological segmentation for image sequence coding," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 639–651, 1994.

[28] C. Schlag, "The new privacy battle: How the expanding use of drones continues to erode our concept of privacy and privacy rights," *Pitt. J. Tech. L. & Pol'y*, vol. 13, p. i, 2012.

[29] J. Serra and P. Soille, *Mathematical morphology and its applications to image processing*. Springer Science & Business Media, 2012, vol. 2.

[30] J. Stanley, C. Crump, and A. Speech, *Protecting Privacy From Aerial Surveillance*. American Civil Liberties Union.(December 2011), 2011, vol. 6, no. 6.

[31] C. Streiffer, A. Srivastava, V. Orlikowski, Y. Velasco, V. Martin, N. Raval, A. Machanavajjhala, and L. P. Cox, "eprivateeye: To the edge and beyond!" in *Proceedings of the Second ACM/IEEE Symposium on Edge Computing*. ACM, 2017, p. 18.

[32] S. Suzuki *et al.*, "Topological structural analysis of digitized binary images by border following," *Computer vision, graphics, and image processing*, vol. 30, no. 1, pp. 32–46, 1985.

[33] T. Takahashi, "Drones and privacy," 2012.

[34] E. Vattapparamban, İ. Güvenç, A. İ. Yurekli, K. Akkaya, and S. Uluağaç, "Drones for smart cities: Issues in cybersecurity, privacy, and public safety," in *2016 International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, 2016, pp. 216–221.

[35] J. Wang, B. Amos, A. Das, P. Pillai, N. Sadeh, and M. Satyanarayanan, "Enabling live video analytics with a scalable and privacy-aware framework," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 14, no. 3s, p. 64, 2018.

[36] J. Wang, Z. Feng, Z. Chen, S. George, M. Bala, P. Pillai, S.-W. Yang, and M. Satyanarayanan, "Bandwidth-efficient live video analytics for drones via edge computing," in *2018 IEEE/ACM Symposium on Edge Computing (SEC)*. IEEE, 2018, pp. 159–173.

[37] Y. Xu, G. Yu, X. Wu, Y. Wang, and Y. Ma, "An enhanced viola-jones vehicle detection method from unmanned aerial vehicles imagery," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 7, pp. 1845–1856, 2016.

[38] J. Yu, B. Zhang, Z. Kuang, D. Lin, and J. Fan, "iprivacy: image privacy protection by identifying sensitive objects via deep multi-task learning," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 5, pp. 1005–1016, 2017.

[39] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, 2016.

[40] X. Zhang and X. Wang, "Chaos-based partial encryption of spiht coded color images," *Signal Processing*, vol. 93, no. 9, pp. 2422–2431, 2013.