Enhancing Lifelogging Privacy by Detecting Screens

Mohammed Korayem [*] CareerBuilder
Robert Templeman [*] Naval Surface Warfare Center, Crane Division
Dennis Chen [*] Olin College
David Crandall **Apu Kapadia** School of Informatics and Computing, Indiana University
{mkorayem, retemple, djcran, kapadia}@indiana.edu, dennis.chen@students.olin.edu

ABSTRACT

Low-cost, lightweight wearable cameras let us record (or 'lifelog') our lives from a 'first-person' perspective for purposes ranging from fun to therapy. But they also capture private information that people may not want to be recorded, especially if images are stored in the cloud or visible to other people. For example, recent studies suggest that computer screens may be lifeloggers' single greatest privacy concern, because many people spend a considerable amount of time in front of devices that display private information. In this paper, we investigate using computer vision to automatically detect computer screens in photo lifelogs. We evaluate our approach on an existing in-situ dataset of 36 people who wore cameras for a week, and show that our technique could help manage privacy in the upcoming era of wearable cameras.

Author Keywords

Lifelogging; wearable cameras; privacy; computer vision; deep learning; convolutional neural networks

ACM Classification Keywords

K.4.1. Public Policy Issues: Privacy; I.4.8. Scene Analysis

INTRODUCTION

Our world is filled with cameras, from surveillance systems to cameras built into phones, laptops, and gaming systems, and wearable cameras like Google Glass [13], Narrative Clip [29], and Autographer [2] will make them even more pervasive. These wearable devices let people record or 'lifelog' visual diaries of their lives from a 'first-person' perspective, in order to treat memory loss and dementia [14], to enhance public security and accountability [8], or just for fun.

But lifelogging cameras collect *thousands* of images every day, including photos with embarrassing, sensitive, or private information. Of course, what is considered private differs between people and contexts, but recent studies on lifelogging have found that the presence of certain objects, especially computer monitors, raises heightened concern [15,16]. Given that the average American spends over five hours a day on digital devices [10], displays are very common in most lifelogs. For example, Figure 1 shows images sampled uniformly at random from one of our personal lifelogs; three happened to capture digital displays with private information including an e-mail, a web search, and financial records.

Figure 1. Random images from an author's lifelog, showing that computer and phone displays with private information are common.

The huge volume of images collected by wearable cameras makes it difficult for people to maintain control over whether and with whom these images are shared. Private image content may be subtle (e.g., account numbers visible only after zooming in), causing people to accidentally share private information through a 'misclosure' [4]. Many wearable devices automatically upload images to the cloud to help people store and share images, which further amplifies privacy concerns. Uploading photos that capture even incidental academic or health records may violate the law, since there are often strong legal protections against disclosing such records.

In this paper, we study the feasibility of using automated computer vision techniques to identify monitors appearing in lifelogging images, so that these images can be flagged for user review. Such a system could avoid the need for people to manually tag each image [21], or to annotate or modify their physical spaces in order to make computer monitors easy to detect [31, 33], and would complement systems that detect photos taken in sensitive places like bathrooms or bedrooms [36] or when people are performing certain activities [6]. Automatically detecting objects in images is a difficult task, and is even more difficult for wearable camera images that are often poorly composed, blurry, and out-of-focus. We investigate the extent to which emerging state-of-the-art computer vision techniques based on deep learning [23] can identify monitors in real lifelogs, using a dataset collected by 36 participants during a week-long, *in situ* study [16]. To our

[*]This work was performed while authors were at Indiana University.

CHI'16, May 07–12, 2016, San Jose, CA, USA
©2016 ACM. ISBN 978-1-4503-3362-7/16/05…$15.00
DOI: http://dx.doi.org/10.1145/2858036.2858417

knowledge, we are the first to attempt monitor detection in first-person images, and among the first to apply deep learning to lifelogging. The experiments suggest that despite the difficulty of the task and the data, we can detect monitors with high enough accuracy to provide useful filtering tools.

RELATED WORK

Wearable cameras have been studied for many years [14, 24, 28, 37–39] but their recent consumer availability has raised questions about their societal and privacy impact. Hoyle et al. [16] explore privacy issues and concerns of lifeloggers, while Denning et al. [11] consider the privacy implications of the bystanders who are photographed by wearable cameras. Thomaz et al. [37] study the balance between eating behavior and privacy information in first person images. Roesener et al. [32] consider privacy and security in augmented reality devices as a special case of wearable cameras. Caine [4] more generally explores how people mistakenly share electronic information with unintended people.

Some techniques have been proposed for restricting where, when, and what image and sensor data is collected. For instance, Virtual Walls [20] suppresses sharing sensor data based on the physical metaphor of transparency, while Langheinrich [25] uses "beacons" to push sharing preferences to nearby devices. Specific to cameras, Roesner et al. [33] embed policies in the environment using physical tags that mark objects and places that should not be photographed, while Templeman et al. [35] propose a conceptual framework in which people specify policies based on automatically detected image content and context. In the embedded system security domain, Scanner Darkly [18] and OS Recognizers [17] control how image data is released to untrusted apps to protect against leaks. Caine et al. study how older adults can control video monitoring of their activities at home [5].

We focus on the complementary problem of automatically recognizing the image content needed to implement these frameworks. Perhaps the most related papers are Templeman et al. [36], which uses computer vision to detect when lifeloggers enter private spaces like bathrooms or bedrooms, and Castro et al. [6], which recognizes which activities the lifelogger is performing. Like us, both of these papers have to contend with the unique challenges of computer vision in first-person imagery, including blurry, poorly composed images, and training datasets that are small and difficult to collect. While those papers estimate *where the photo was taken* and *what the lifelogger was doing*, we consider the complementary problem of detecting *what is in the image itself*.

As a first step, we investigate the particularly important [16] problem of detecting monitors in lifelogging images. Like Castro et al. [6], we apply deep learning based on Convolutional Neural Networks [26], which have recently emerged as the state-of-the-art for object recognition [23] by significantly outperforming traditional visual features and classifiers [7, 9, 27] like those used by Templeman et al. [36]. Also as in Castro et al., we test on real lifelogging images collected by in-situ user studies, although their dataset was collected by a single participant over several months whereas ours was collected by 36 participants over a single week. Thus the datasets are also complementary: ours is perhaps more representative of how the system would perform across a diverse set of people and environments, whereas theirs is likely more representative of temporal variations.

OUR APPROACH

Detecting computer screens is a specific instance of 'visual object category detection,' in which the goal is to recognize instances of broad classes of objects (e.g., all cars, not one particular make and model). Different screens have different appearances, of course, and even the same display looks different across images due to changes in lighting, camera angle, and screen content. A key challenge is to build recognition models that are robust against such variations while still sensitive to features that differentiate monitors from similar objects like picture frames, windows, and paper documents.

We apply Convolutional Neural Networks (CNNs), which have recently emerged as the state-of-the-art technique in visual recognition [23]. Instead of using hand-crafted algorithms that convert pixel values into statistical features (e.g., SIFT or HOG [9, 27]) for input into classifiers like Support Vector Machines [7], CNNs analyze the raw pixel values themselves, in effect learning low-level features automatically and simultaneously with the high-level classifier. These CNNs are similar to the feed-forward neural networks that have been studied for decades [26], but are much deeper (a dozen or more layers) with many more parameters (hundreds of millions), and training them requires greater computational resources (a high-end workstation with a GPU).

Datasets

We collected two datasets to train and test our monitor detector. *Author* was collected from our personal lifelogs taken by a mixture of devices including Google Glass, Narrative Clip, and Autographer, and consisted of 18,798 images. *User study* is from our previous *in situ* study in which 36 undergraduate students wore wearable cameras for a week [16] and consists of 2,742 images. (This study was approved by IRB, with careful controls to protect participant and bystander privacy and to address the ethical and legal concerns of a study of this type; please see our previous publication [16] for details.) Both datasets were collected under realistic, uncontrolled conditions; for example, *User study* includes a wide variety of types, brands, and models of displays that reflects the diversity of computers that students own and use. To generate ground truth labels, we manually reviewed each of the over 20,000 images. An image was labeled as a positive exemplar even if only a small portion of a screen was visible (since private information could be revealed in any part of a screen). We defined "computer monitor" to include desktop and laptop computer displays, but not to include phones, tablets, TVs, or other electronic devices.

Training the models

We outline our approach here; see our technical report [22] for details. We used the open-source Caffe software [19] to create and train our monitor detector. We adopted the network structure of Krizhevsky et al. [23], which takes raw RGB

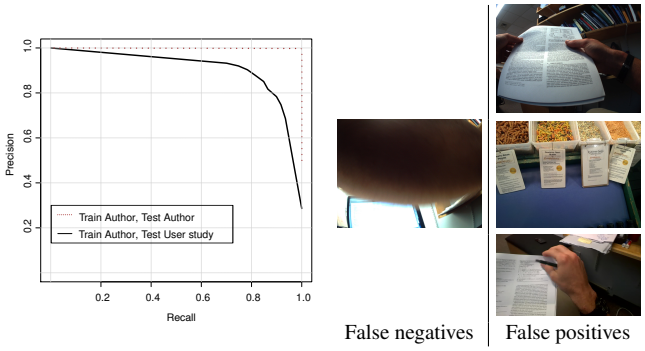

Figure 2. Left: Precision-Recall curves for screen detection. Right: All of the incorrectly classified *Author* images.

pixel values as input, and passes them through five convolutional layers, three fully connected layers, and several max-pooling layers. The complete network includes about 2.3 million neurons and 60 million learned parameters. Learning so many parameters requires a huge training dataset, typically millions of images versus the thousands in ours. We tried several techniques to address this, including downloading Flickr images tagged "monitor," but found that these images were so different from lifelogging images that they did not help. In the end, we applied fine-tuning [30] by starting with a CNN pre-trained on ImageNet [34], even though it consists of consumer images instead of lifelogging photos, and then used those parameters as initialization to re-train using our relatively small dataset. The intuition behind this is that there is enough structure in the visual world that a CNN trained for one problem learns useful low-level features that apply to other unrelated problems. Our trained models are available at `http://vision.soic.indiana.edu/screenavoider/`.

EXPERIMENTS

We first used the *Author* dataset, which we partitioned into 9,986 training images and 1,842 test images and sampled to have an equal class distribution. The classifier achieved 99.8% accuracy compared to the 50% baseline on its task of deciding whether each image contained a monitor. In fact, there was only one false negative and three false positives, all shown in Figure 2. The false negative image is of poor quality, such that no private information is visible, while two of the false positives include hardcopy documents that look similar to screens of text. This experiment has particularly high accuracy because while the CNN must classify unseen test images based on an independent set of training images, the training and test images are sampled from photo streams of the same participants and thus include similar images.

In a much more challenging experiment, we trained on 9,986 images from the *Author* dataset, but tested on the completely independent *User study* dataset from 36 lifeloggers. The class distribution in this case reflects the true distribution in the study, with 28.6% of images having screens. Our CNN achieved 91.5% accuracy, compared with 71.4% for the majority class baseline (that always predicts 'no-monitor'). We cannot publish the IRB-controlled user study images themselves in order to show incorrect classifications, so we manually characterize them instead. Of the 117 false negatives, the top four (not mutually exclusive) failure modes were computer screens that: (1) displayed photorealistic video games (49.6%); (2) were less than half visible (48.9%); (3) were significantly out of focus (35.0%); and (4) were showing other photorealistic content like movies, sports, and TV (12.8%). Even humans have difficulty discriminating some of these (e.g., between a photo displayed on a monitor and a photo print in a physical frame). Of the false negatives, only 6.8% (about 0.3% of the full test dataset) actually displayed applications that potentially contain sensitive content (specifically Skype (n=1), Microsoft Word (n=2), Facebook (n=3), and Adobe Illustrator (n=2)). The top causes of the 116 false positives were physical windows (33.6%), framed objects like photos and mirrors (32.8%), and screens of devices like TVs and tablets (16.4%) which we counted as false positives, but in practice might be considered true positives.

Alternatively, we can view this problem as a retrieval rather than a classification task. Figure 2 (left) adopts this view, presenting Precision-Recall curves for the two experiments described above. The curves explicitly show the trade-off between precision and recall so that the best point could be selected depending on the application. For example, in the difficult *User study* dataset, we can retrieve 88% of screen images with a precision of 80%, or be more conservative and catch 95% of screen images with a precision of about 60%.

DISCUSSION

Classifier accuracy and performance

Our goal in this paper is to test the feasibility of automatically detecting screens. While far from perfect, our results suggest that modern classifiers are accurate enough to help manage lifelogging privacy, especially because they allow balancing between precision and recall. For example, while finding most photos with monitors (high recall) may be important, moderate precision is often acceptable, since deleting a few of the thousands of non-monitor photos captured each day may not matter. Even if recall is not perfect, we agree with Raval et al. [31] that simply reducing the number of private photos can significantly improve privacy for most people. Image filtering could be performed in various ways, including in device hardware to decide whether to take a photo, in device firmware to control whether to share a photo with an untrusted app, or in a trusted cloud service for flagging or censoring photos.

A system deployed at larger scale could improve our performance significantly. We tested two extreme scenarios: training and test data sampled from the same lifelogs, and training and test data from lifelogs of unrelated people. A practical system could use the latter when a person first begins lifelogging to 'bootstrap' the classifier, but then use feedback over time to adapt to a specific person's environment and lifestyle. A deployed cloud-based system could also pool training images across users, building larger training sets than we could.

Our models were trained on a single workstation with a 16-core CPU and NVidia Tesla K20 GPU in about 5 hours. Classifying an image required 0.12 seconds with a GPU and about 1.5 seconds with just a CPU. These computation requirements would push the limits of current wearable devices, but

		Test on *Author*				Test on *User study*				
		Predicted class				Predicted class				
		Other	IM	FB	Gmail	No screen	Other	IM	FB	Gmail
Actual	No screen	—	—	—	—	1882	59	6	0	12
	Other	1165	378	24	150	157	243	143	35	158
	IM (Messages)	148	1403	0	166	0	2	0	0	0
	Facebook	379	635	524	179	4	12	11	5	3
	Gmail	460	602	23	632	0	7	3	0	2

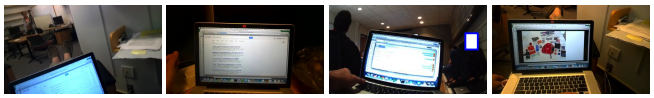

Figure 3. ***Top:*** **Confusion matrices for application classification trained on *Author*.** ***Bottom:*** **Some correctly recognized *Author* images, from left: Gmail, Other, Messages, and Facebook. (Faces obscured for privacy.)**

GPUs are beginning to appear in mobile devices, and in the meantime most processing could be performed in the cloud (similar to speech recognition on current mobile phones [1]).

Detecting screen content

Some people may want finer-grained control over which screen images are flagged. For example, Hoyle et al. [16] found that some gamers enjoy sharing lifelogged photos that capture key moments in video games, but may wish to suppress photos displaying privacy-sensitive applications. As a first step, we trained and tested CNNs to classify whether a given lifelogging image of a screen is displaying Facebook, GMail, Mac OS Messages, or another application. We used 9,986 *Author* images for training and a disjoint set of 6,868 *Author* images for testing, sampled to have equal class distribution. The CNN achieved an accuracy of 54.2% compared to the 25.0% baseline, and Figure 3 presents the confusion matrix. This classification task is quite difficult, especially because screens are often not fully visible; Figure 3 presents some correctly classified sample images.

We also considered a more difficult five-way task in which the classifier must both determine whether an image contains a screen and if so, which application it is displaying, using *Author* for training and *User study* for testing. Because the test set consists of real lifelogging data, the class distribution is highly unbalanced; the accuracy for this experiment was 77.7% compared to the 71.4% baseline (always predicting no monitor). From the confusion matrix in Figure 3, we see that the classifier performs well at the coarse task of inferring whether a screen is present, but classification amongst sensitive applications is much more difficult.

Easing the computer vision problem

A complementary direction for improving performance is to mark screens themselves with labels that could be more readily recognized by a vision algorithm. As a first step, we investigated rendering a real-time, machine-readable QR barcode on the corner of the lifelogging person's screen that embeds information about which applications are currently visible. Lifelogging images could then be scanned for this QR code, which is a much easier problem than recognizing the monitors themselves. We implemented a prototype system for Mac OS X, using a 120x120 pixel QR codes configured for maximum readability and error correction rendered by QRencode [12], and ZBar [3] to recognize them in lifelogging images.

We evaluated the system on a separate set of 535 lifelogging images collected by the authors, and found that the QR code was successfully recognized in 85.6% of them. There were no false positives and all successfully detected QR codes were read with 100% accuracy, because it is virtually impossible that a background image region spuriously satisfies the QR specification. This approach could thus be a complement to the fully automatic approaches described above, allowing higher recognition rates but requiring special software to be installed on any computers used by the lifelogger.

Generalizing our classifiers

We focused on displays of computers because they have been identified as a greater privacy risk [16] than displays of other devices (like phones or tablets). This is in part because monitors are larger, and thus both more likely to be captured by wearable cameras and more likely to be displaying private content. However, our vision techniques are general and can learn classifiers for other devices given sufficient training data. Our classifiers could also be combined with complementary work on detecting where photos are taken [36] and what lifeloggers are doing [6] to allow finer-grained privacy controls based on both image content and context.

CONCLUSION

Wearable cameras are opening up exciting new applications, but will also require new techniques to help people preserve their privacy. In this paper we investigate whether modern computer vision techniques could be used to automatically detect private content in images. As a first step, we investigate detecting monitors, since prior work has shown that private content displayed on monitors is among the greatest privacy concerns of lifeloggers. We show that policies based on the presence of computer screens in images can accurately be enforced at a coarse level. While fine-grained policies defined on the type of screen content would be more challenging to enforce, we remain optimistic based on our initial results, especially when characterizing 'sensitive' vs. 'non-sensitive' applications.

Much work remains to be done in this area, given the wide variety of questions, challenges, and applications for HCI that wearable cameras will introduce. Our paper is a first step towards demonstrating that privacy challenges can be at least partially addressed with easy-to-use automated solutions, by demonstrating their feasibility in one specific but interesting, important, and timely domain (detecting monitors in lifelogs). We hope our paper will inspire more work in this emerging area at the intersection of HCI, vision, and privacy.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (CNS-1408730, IIS-1253549), Google, and the IU FRSP program, and used compute facilities provided by NVidia, the Lilly Endowment through support of the IU Pervasive Technology Institute, and the Indiana METACyt Initiative. We thank Denise Anthony for her helpful comments.

REFERENCES

1. Apple. *Siri*. `https://www.apple.com/ios/siri/`.
2. Autographer. *Autographer Wearable Camera*. `http://autographer.com`.
3. Jeff Brown. *ZBar bar code reader*. `http://zbar.sourceforge.net/index.html`.
4. K. E. Caine. 2009. Supporting privacy by preventing misclosure. In *CHI'09 Extended Abstracts on Human Factors in Computing Systems*. ACM, 3145–3148.
5. Kelly E. Caine, Celine Y. Zimmerman, Zachary Schall-Zimmerman, William R. Hazlewood, Alexander C. Sulgrove, L. Jean Camp, Katherine H. Connelly, Lesa L. Huber, and Kalpana Shankar. 2010. DigiSwitch: Design and Evaluation of a Device for Older Adults to Preserve Privacy While Monitoring Health at Home. In *Proceedings of the 1st ACM International Health Informatics Symposium*. 153–162.
6. D. Castro, S. Hickson, V. Bettadapura, E. Thomaz, G. Abowd, H. Christensen, and I. Essa. 2015. Predicting daily activities from egocentric images using deep learning. In *Proceedings of the ACM International Symposium on Wearable Computers*.
7. Ken Chatfield, Victor Lempitsky, Andrea Vedaldi, and Andrew Zisserman. 2011. The devil is in the details: an evaluation of recent feature encoding methods. In *Proceedings of the British Machine Vision Conference*.
8. Fanny Coudert, Denis Butin, and Daniel Le Metayer. 2015. Body-worn cameras for police accountability: Opportunities and risks. *Computer Law & Security Review* 31 (2015), 749–762.
9. Navneet Dalal and Bill Triggs. 2005. Histograms of Oriented Gradients for Human Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 886–893.
10. C. Delo. 2013. U.S. Adults Now Spending More Time on Digital Devices Than Watching TV. *Advertising Age* (2013).
11. T. Denning, Z. Dehlawi, and T. Kohno. 2014. In Situ with Bystanders of Augmented Reality Glasses: Perspectives on Recording and Privacy-Mediating Technologies. In *Proceedings of the International Conference on Human Factors in Computing Systems*.
12. K. Fukuchi. *libqrencode*. `http://http://fukuchi.org/works/qrencode/`.
13. Google. *Google Glass*. `https://plus.google.com/+GoogleGlass/posts`.
14. Steve Hodges, Lyndsay Williams, Emma Berry, Shahram Izadi, James Srinivasan, Alex Butler, Gavin Smyth, Narinder Kapur, and Ken Wood. 2006. SenseCam: a Retrospective Memory Aid. In *Proceedings of the ACM International Conference on Ubiquitous Computing*.
15. Roberto Hoyle, Robert Templeman, Denise Anthony, David Crandall, and Apu Kapadia. 2015. Sensitive Lifelogs: A Privacy Analysis of Photos from Wearable Cameras. In *ACM SIGCHI Conference on Human Factors in Computing Systems*.
16. R. Hoyle, R. Templeman, S. Armes, D. Anthony, D. Crandall, and A. Kapadia. 2014. Privacy behaviors of lifeloggers using wearable cameras. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 571–582.
17. S. Jana, D. Molnar, A. Moshchuk, A. M. Dunn, B. Livshits, H. J. Wang, and E. Ofek. 2013a. Enabling Fine-Grained Permissions for Augmented Reality Applications with Recognizers. In *USENIX Security*. 415–430.
18. S. Jana, A. Narayanan, and V. Shmatikov. 2013b. A Scanner Darkly: Protecting User Privacy From Perceptual Applications. In *Proceedings of the IEEE Symposium on Security and Privacy*.
19. Yangqing Jia. 2013. Caffe: An Open Source Convolutional Architecture for Fast Feature Embedding. `http://caffe.berkeleyvision.org/`. (2013).
20. Apu Kapadia, Tristan Henderson, Jeffrey J. Fielding, and David Kotz. 2007. Virtual Walls: Protecting Digital Privacy in Pervasive Environments. In *Proceedings of the Fifth International Conference on Pervasive Computing*, Vol. 4480. 162–179.
21. P. Klemperer, Y. Liang, M. Mazurek, M. Sleeper, B. Ur, L. Bauer, L.F. Cranor, N. Gupta, and M. Reiter. 2012. Tag, you can see it!: using tags for access control in photo sharing. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*. ACM, 377–386.
22. Mohammed Korayem, Robert Templeman, Dennis Chen, David Crandall, and Apu Kapadia. 2014. *ScreenAvoider: Protecting Computer Screens from Ubiquitous Cameras*. arXiv 1412.0008.
23. A. Krizhevsky, I. Sutskever, and G.E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.
24. Michael S. Lam, Suneeta Godbole, Jacqueline Chen, Melody Oliver, Hannah Badland, Simon J. Marshall, Paul Kelly, Charlie Foster, Aiden Doherty, and Jacqueline Kerr. 2013. Measuring Time Spent Outdoors Using a Wearable Camera and GPS. In *Proceedings of the 4th International SenseCam Pervasive Imaging Conference*. 1–7.
25. Marc Langheinrich. 2002. A Privacy Awareness System for Ubiquitous Computing Environments. In *Proceedings of the ACM International Conference on Ubiquitous Computing*. Vol. 2498. 237–245.
26. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 11, 86 (1998), 2278–2324.

27. David G. Lowe. 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60, 2 (Nov. 2004), 91–110.

28. Gabriela Marcu, Anind K. Dey, and Sara Kiesler. 2012. Parent-driven Use of Wearable Cameras for Autism Support: A Field Study with Families. In *Proceedings of the ACM International Conference on Ubiquitous Computing*. 401–410.

29. Narrative. *Narrative Clip: a wearable, automatic lifelogging camera*. `http://getnarrative.com`.

30. M. Oquab, L. Bottou, I. Laptev, and J. Sivic. 2014. Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

31. Nisarg Raval, Landon Cox, Animesh Srivastava, Ashwin Machanavajjhala, and Kiron Lebeck. 2014. MarkIt: privacy markers for protecting visual secrets. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. 1289–1295.

32. Franziska Roesner, Tadayoshi Kohno, and David Molnar. 2013. Security and Privacy for Augmented Reality Systems. *Commun. ACM* (2013).

33. Franziska Roesner, David Molnar, Alexander Moshchuk, Tadayoshi Kohno, and Helen J Wang. 2014. World-Driven Access Control for Continuous Sensing. *Microsoft Tech Report* (2014).

34. Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2014. *ImageNet Large Scale Visual Recognition Challenge*. arXiv 1409.0575.

35. R. Templeman, A. Kapadia, R. Hoyle, and D. Crandall. 2014a. Reactive Security: Responding to Visual Stimuli from Wearable Cameras. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. 1297–1306.

36. R Templeman, M. Korayem, D. Crandall, and K. Kapadia. 2014b. PlaceAvoider: Steering First-Person Cameras away from Sensitive Spaces. In *Network and Distributed System Security Symposium*.

37. Edison Thomaz, Aman Parnami, Jonathan Bidwell, Irfan Essa, and Gregory D. Abowd. 2013. Technological Approaches for Addressing Privacy Concerns when Recognizing Eating Behaviors with Wearable Cameras. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 739–748.

38. Nancy A Van House and Marc Davis. 2005. The social life of cameraphone images. In *Proceedings of the Pervasive Image Capture and Sharing: New Social Practices and Implications for Technology Workshop at the Seventh International Conference on Ubiquitous Computing*.

39. Jing Wang, Grant Schindler, and Irfan Essa. 2012. Orientation-aware Scene Understanding for Mobile Cameras. In *Proceedings of the ACM International Conference on Ubiquitous Computing*. 260–269.