Edge Analytics in the Internet of Things

Mahadev Satyanarayanan[†], Pieter Simoens[‡], Yu Xiao[•],
Padmanabhan Pillai[⋆], Zhuo Chen[†], Kiryong Ha[†],
Wenlu Hu[†], Brandon Amos[†]

[†]Carnegie Mellon University, [‡]Ghent University - iMinds, [•]Aalto University, [⋆]Intel Labs

ABSTRACT

High data rate sensors such as video cameras are becoming ubiquitous in the Internet of Things. This paper describes *GigaSight*, an Internet-scale repository of crowd-sourced video content, with strong enforcement of privacy preferences and access controls. The GigaSight architecture is a federated system of VM-based *cloudlets* that perform video analytics at the edge of the Internet, thus reducing demand for ingress bandwidth into the cloud. *Denaturing*, which is the owner-specific reduction in fidelity of video content to preserve privacy, is one form of analytics on cloudlets. Content-based indexing for search is another form of cloudlet-based analytics.

1. Ubiquitous Imaging and Sensing

Many of the "things" in the Internet of Things (IoT) are video cameras. These are proliferating at an astonishing rate. In 2013, it was estimated that there was one surveillance cameras for every 11 people in the UK [2]. Video cameras are now common in police patrol cars [7, 16] and almost universal in Russian passenger cars [3]. The emergence of commercial products such as Google Glass and GoPro point to a future in which body-worn video cameras will be commonplace. Many police forces in the US are now considering the use of body-worn cameras [10]. The report of the 2013 *NSF Workshop on Future Directions in Wireless Networking* [1] predicts that "It will soon be possible to find a camera on every human body, in every room, on every street, and in every vehicle."

What will we do with all this video? Today, most video is stored close to the point of capture, on local storage. Its contents are not easily searched over the Internet, even though there are many situations in which timely remote access can be valuable. For example, at a large public event such as a parade, a lost child may be seen in the video of someone recording the event [12]. Surveillance videos were crucial for discovering the Boston Marathon bombers in 2013. In general, an image captured for one reason can be valuable for some totally unrelated reason. Stieg Larsson's fictional work "The Girl with the Dragon Tattoo" embodies exactly this theme: a clue to solving a crime is embedded in the backgrounds of old crowd-sourced photographs [9]. This richness of content and possibility of unanticipated value distinguishes video from simpler sensor data that have historically been the focus of the sensor network research community. The sidebar presents many hypothetical use cases for crowd-sourced video. An Internet-scale searchable repository for crowd-sourced video content, with strong enforcement of privacy preferences and access controls, would be a valuable global resource. In this paper, we examine the technical challenges involved in creating such a repository. For reasons that will become clear in Section 2, this will be a physically distributed and federated repository rather than a monolithic entity.

Video is not the only data type that could benefit from such a global repository. Any data type in the IoT that has high data rate can potentially benefit. For example, each GE jet aircraft engine generates 1 TB of sensor data every 24 hours [6]. The airframe of the Boeing 787 generates half a terabyte of sensor data on every flight [5]. Modern automobiles, especially the emerging self-driving family of cars, generate comparable amounts of sensor data. These are just a few of examples of IoT data sources that have high data rates. Real-time analytics on such data could be valuable for early warning of imminent failures, need for preventive maintenance, fuel economy, and a host of other benefits. Longer-term studies of historical data from multiple units could reveal model-specific shortcomings that could be corrected in future designs. Many of the points made in this paper apply directly to this broad class of sensors. However, for simplicity and ease of exposition, we will focus on video cameras as sensors in our discussion.

2. GigaSight: a Reverse CDN

A key challenge for the cloud is the high cumulative data rate of incoming videos from many cameras. Without careful design, this could easily overwhelm metro area networks and ingress Internet paths into centralized cloud infrastructure such as Google's large data centers or Amazon's EC2 sites. In 2013, roughly one hour's worth of video was uploaded to YouTube each second. That corresponds to only 3600 concurrent uploads. Scaling well beyond this to millions of concurrent uploads from a dense urban area is going to be very difficult. Today's high-end metro area networks are only 100 Gbps links. Each such link can support 1080p streams from only 12000 users at YouTube's recommended upload rate of 8.5 Mbps. A million concurrent uploads would require 8.5 Tbps.

To solve this problem, we propose GigaSight, a hybrid

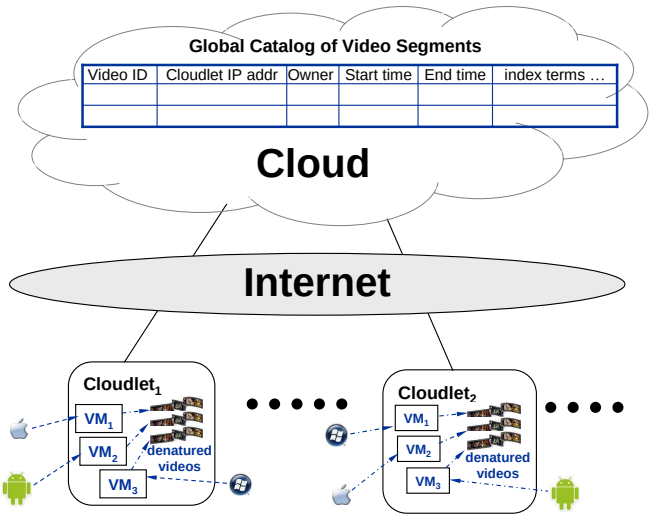


Figure 1: GigaSight Architecture

cloud architecture that is effectively a content delivery network (CDN) in reverse. This architecture, shown in Figure 1, uses decentralized cloud computing infrastructure in the form of VM-based *cloudlets* [13]. A cloudlet is a new architectural element that arises from the convergence of mobile computing and cloud computing. It represents the middle tier of a 3-tier hierarchy: mobile device – cloudlet – cloud. A cloudlet can be viewed as a "data center in a box" that "brings the cloud closer." While cloudlets were originally motivated by reasons of end-to-end latency for interactive applications, the use of cloudlets in GigaSight is based solely on bandwidth considerations.

The GigaSight architecture is described in detail by Simoens et al [15]. In this architecture, video from a mobile device only travels as far as its currently-associated cloudlet. Computer vision analytics are run on the cloudlet in near real-time, and only the results (e.g. recognized objects, recognized faces, etc.) along with meta-data (e.g., owner, capture location, timestamp, etc.) are sent to the cloud. The tags and meta-data in the cloud can guide deeper and more customized searches on the content of a video segment during its (finite) retention period on a cloudlet.

An important type of "analytics" supported on cloudlets is automated modification of video streams to preserve privacy. For example, this may involve editing out frames and/or blurring individual objects within frames. What needs to be removed or altered is highly specific to the owner of a video stream, but no user can afford the time to go through and manually edit video that is captured on a continuous basis. This automated, owner-specific lowering of fidelity of a video stream to preserve privacy is called *denaturing,* and is discussed further in Section 3.

It is important to note in Figure 1 that cloudlets are not just temporary staging points for denatured video data en route to the cloud. With a large enough number of cameras and continuous video capture, the constant influx of data at the edges will be a permanent stress on the ingress paths to the cloud. Just buffering data at cloudlets for later transmission to the cloud won't do — because video will be streaming 24/7, there will never be a "later" when ingress paths are unloaded. The potential bandwidth bottleneck is at the access and aggregation networks, and not in the core network with its high-speed links. Preprocessing videos on cloudlets also offers the potential of using content-based storage optimization algorithms to retain only one of many similar videos from co-located cameras. Thus, cloudlets are the true home of denatured videos. In a small number of cases, based on popularity or other metrics of importance, some videos may be copied to the cloud for archiving or replicated in the cloud or other cloudlets for scalable access. But most videos reside only at a single cloudlet for a finite period of time (typically on the order or hours, days or weeks). In a commercial deployment of GigaSight, how long videos remain accessible will depend on the storage retention and billing policies.

Notice that Figure 1 is agnostic regarding the exact positioning of the cloudlets in the network. One option could be to place numerous small cloudlets at the very network edge. An alternative is to place fewer but larger cloudlets deeper in the network, e.g., at metropolitan scale. The analysis by Simoens et al [15] suggests that small cloudlets close to the edge is the better alternative.

Marketing and Advertising: Crowd-sourced videos can provide observational data for questions that are difficult to answer today. For example, which are the billboards that attract the most user attention? How successful is a new store window display in attracting interest? Which are the clothing colors and patterns that attract most interest in viewers? How regional are these preferences?
Theme Parks: Visitors to places like DisneyWorld can capture and share their experiences throughout the entire day, including rides. With video, audio and accelerometer capture, the re-creation of rides can be quite realistic. An album of a family's visit can be shared via social networks such as Facebook or Google+.
Locating people, pets and things: A missing child was last seen walking home from school. A search of crowd-sourced videos from the area shows that the child was near a particular spot an hour ago. The parent remembers that the child has a friend close to that location. She is able to call the friend's home and locates the child there. When a dog owner reports that his dog is missing, search of crowd-sourced videos captured in the last few hours may help locate the dog before it strays too far.
Public safety: Which are the most dangerous intersections, where an accident is waiting to happen? Although no accident has happened yet, it is only a matter of time before a tragedy occurs. Video analytics can reveal dangerous intersections, leading to timely installation of traffic lights. Many other public safety improvements are also possible: uneven sidewalks that are causing people to trip and fall; timely detection of burned-out street lights that need to be replaced; newly-appeared potholes that need to be filled; and the iciest road surfaces and sidewalks that need immediate attention.
Fraud detection: A driver reports that his car was hit while it was parked at a restaurant. However, his insurance claims adjuster finds a crowd-sourced video in which the car is intact when leaving the restaurant. Other law and order opportunities abound. For example, when a citizen reports a stolen car, his description could be used to search recent crowd-sourced video for sightings of that car and help to locate it.

Sidebar: Example Use Cases of Crowd-Sourced Video

3. Denaturing

Denaturing has to strike a balance between privacy and value. At one extreme of denaturing is a blank video: perfect privacy, but zero value. At the other extreme is the original video at its capture resolution and frame rate. This has

the highest value for potential customers, but also incurs the highest exposure of privacy. Where to strike the balance is a difficult question that is best answered individually, by each user. This decision will most probably be context-sensitive.

Denaturing is a complex process that requires careful analysis of the captured frames. From a technical viewpoint, state-of-the-art computer vision algorithms enable face detection, face recognition, and object recognition in individual frames. In addition, activity recognition in video sequences is also possible. However, preserving privacy involves more than blurring (or completely removing) frames with specific faces, objects, or scenes in the personal video. From other objects in the scene, or by comparing with videos taken at the same place and/or time from other users with different privacy settings, one might still deduce which object was blurred and is hence of value to the person who captured the video. The user's denaturing policy must also be applied to videos that were captured by others at approximately the same time and place. Simply sending the denaturing rules to the personal VMs of other parties is undesirable; as this would expose at a meta level the sensitive content. One possible solution is provided in [4], where weak object classifiers are sent to a central site where they are combined to a global concept model. This model could then be returned to the personal VMs. Of course, this approach requires videos to be temporarily saved in the personal VM until the central site has received any video uploaded at the same time and place. Any video that is uploaded later could then simply be discarded, to avoid keeping videos in the personal VM for too long.

In its full generality, denaturing may not only involve content modification but may also involve meta-data modification. For example, the accuracy of location meta-data associated with a sequence of video frames may be lowered to meet the needs of k-$anonymity$ in location privacy [11]. Whether the contents of the video sequence will also have to be blurred depends on its visual distinctiveness — a scene with the Eiffel Tower in the background is obviously locatable even without explicit location meta-data.

In the future, guidance for denaturing may also be conveyed through social norms that deprecate video capture in certain locations and of certain types of scenes. A system of tagging locations or objects with visual markers (such as QR codes) could indicate that video capture is unwelcome. One can imagine video capture devices that automatically refrain from recording when they recognize an appropriate QR code in the scene. In addition, denaturing algorithms on a cloudlet may also strip out scenes that contain such a code. In the long run, one can envision the emergence of an ethics of video capture in public. Many broader societal issues, such as the ability to subpoena captured but encrypted video, add further complexity. Clearly, denaturing is a very deep concept that will need time, effort and deployment experience to fully understand. GigaSight opens the door to exploring these deep issues and to evolving a societally acceptable balance between privacy and utility.

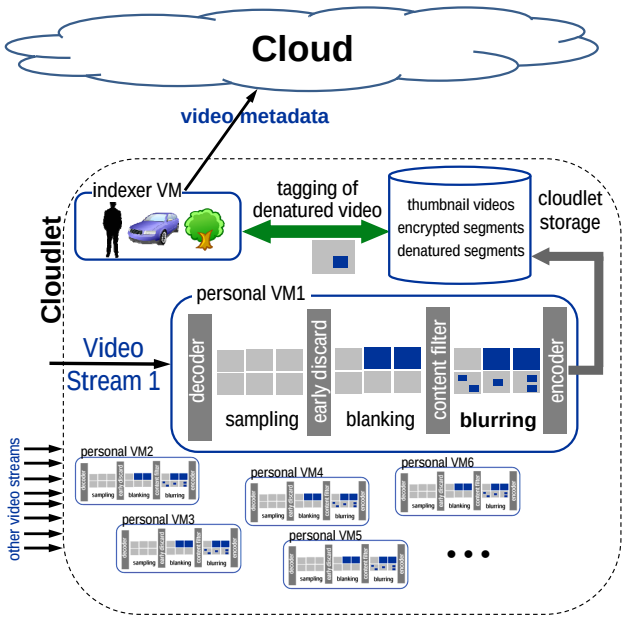


Figure 2: Cloudlet Processing in GigaSight Prototype

In the GigaSight prototype, a $personal\ VM$ on the cloudlet denatures each video stream in accordance with its owner's expressed preferences. This VM is the only component, apart from the mobile device itself, that accesses the original, non-denatured video. Figure 2 illustrates the processing with a personal VM. Denaturing is implemented as a multi-step pipeline. In the first step, a subset of the video frames is selected for actual denaturing. Our initial experiments [15] showed that denaturing is too compute-intensive to perform at the native video frame rate. The output of the denaturing process is therefore in two parts. One part is a low-framerate video file, called its "thumbnail video," that provides a representative overview of video content for indexing and search operations. The second part is an encrypted version of the original video. Both outputs are stored on the cloudlet, outside the personal VM. The encryption of the full-fidelity video uses a per-session AES-128 private key that is generated by the personal VM. If a search of the thumbnail video suggests that a particular segment of the full-fidelity video might of interest, its personal VM can be requested to decrypt and denature that segment. This newly-denatured video segment can then be cached for future resuse.

After sampling of video frames, metadata-based filters with low computational complexity are applied. This early-discard step is a binary process: based on time, location or other metadata, the frame is either completely blanked or passed through unmodified. Then, we apply content-based filters that are part of the preference specifications for denaturing. For example, face detection and blurring using specified code within the personal VM may be performed on each frame. Figure 3 illustrates the output of such a denatured frame.

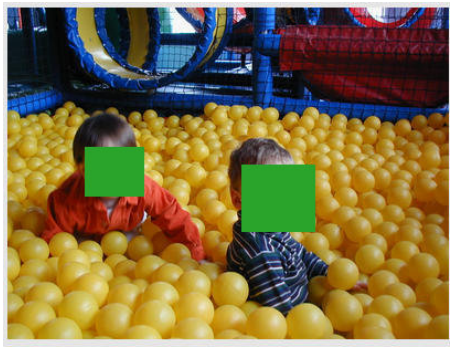


Figure 3: Example of a Denatured Video Frame

4. Indexing and Search

The indexing of denatured video content is a background activity that is performed by a separate VM on a cloudlet. To handle searches that are time-sensitive (such as locating a lost child) or to search for content that is not indexed, custom search code encapsulated in a VM can directly examine denatured videos. For each tag produced by the indexer, an entry is created in a dedicated tag table of the cloudlet database. Each entry contains the tag, the ID of the video segment, and a confidence score. For example, an entry "dog, `zoo.mp4`, 328, 95" indicates that our indexer detected with 95% confidence a dog in frame 328 of the video `zoo.mp4`. After extraction, these tags are also propagated to the catalog of video segments in the cloud. The throughput of indexing depends on the number of objects that must be detected. As this number is potentially very high, we propose to apply at first stage only classifiers for the most popular objects sought in the database. Classifiers of less popular objects could be applied on an ad-hoc basis if needed. As a proof of concept, the GigaSight prototype uses a Python-based implementation of Shotton et al.'s image categorization and segmentation algorithm [14] with classifiers trained on the MSRC21 data set mentioned in that work. This enables detection and tagging of 21 classes of common objects such as aeroplanes, bicycles, birds, boats, etc.

GigaSight uses a two-step hierarchical workflow to help a user find video segments relevant to a specific context. First, the user performs a conventional SQL search on the cloud-wide catalog. His query may involve metadata such as time and location, as well as tags extracted by indexing. The result of this step is a list of video segments and their denatured thumbnails. The identity (i.e., the host names or IP addresses) of the cloudlets on which those video segments are located can also be obtained from the catalog.

Viewing all the video segments identified by the first step may overwhelm the user. We therefore perform a second search step that filters on actual content to reduce the returned results to a more relevant set. This step is very computationally intensive but can be run in parallel on the cloudlets. This step uses *early discard,* as described by Huston et al. [8], to increase the selectivity of a result stream. Using a plugin interface, image processing code fragments called *filters* can be inserted into the result stream. These code fragments allow user-defined classifiers to examine video segments and to discard irrelevant parts of them, thus reducing the volume of data presented to the user. We provide a suite of filters for common search attributes such as color patches and texture patches. For more complex image content the user can train his own filters offline and insert them into the result stream.

To illustrate this two-step workflow, consider a search for *"any images taken yesterday between 2pm and 4pm during a school outing to the Carnegie Science Center in Pittsburgh, showing two children in a room full of yellow balls and one of the children wearing his favorite blue plaid shirt."* The first step of the search would use the time and location information and the "face" tag to narrow the search. The result is a potentially large set of thumbnails from denatured videos that cover the specified location. From a multi-hour period of video capture by all visitors, this may only narrow the search to a few hundred or few thousand thumbnails. Using a color filter tuned to yellow, followed by a composite color/texture filter tuned to blue and plaid, most of these thumbnails may be discarded. Only the few thumbnails that pass this entire bank of filters are presented to the user. From this small set of thumbnails, it is easy for the user to pick the result shown in Figure 3.

GigaSight only processes video that is voluntarily shared. Datasets gathered via crowd-sourcing often exhibit a sampling bias towards popular events, news, etc. and the "long tail" is much less covered. We believe this content bias will be lower in GigaSight since many of its data sources (e.g. police on-body cameras, automobile dashboard cameras, etc.) involve continuous capture of video. Conversely, pruning the collection of videos at locations and times where much redundant footage is available would limit the richness of data collected by GigaSight. An "uninteresting" detail that is eliminated in the pruning could be exactly the crucial evidence for an important future investigation, such as one of the scenarios in the sidebar.

5. Automotive Environments

The GigaSight architecture is especially relevant to automobiles. For the foreseeable future, cloud connectivity from a moving automobile will be 3G or 4G. An important question is whether cloudlets should be placed within automobiles or at cell towers. We see value in both alternatives, as shown in Figure 4. This architecture can be viewed as a mapping of Figure 1 to the automotive context.

Continuous capture and real-time analytics of car-mounted video cameras can help to improve road safety. For example, if the computer vision analytics on one automobile's cloudlet recognizes a pothole, dead animal, or fallen tree branch, it can transmit the coordinates of the hazard (including a brief video segment) to its cell tower cloudlet. The cloudlet can share this information promptly with other cars associated with that cell tower. With advance warning of the hazard, those cars can proactively shift lanes to avoid the hazard.

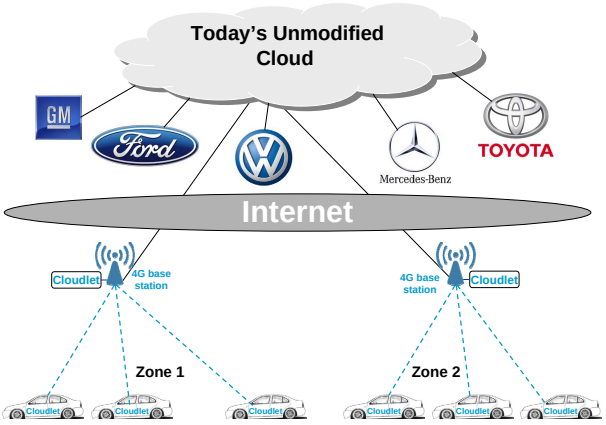
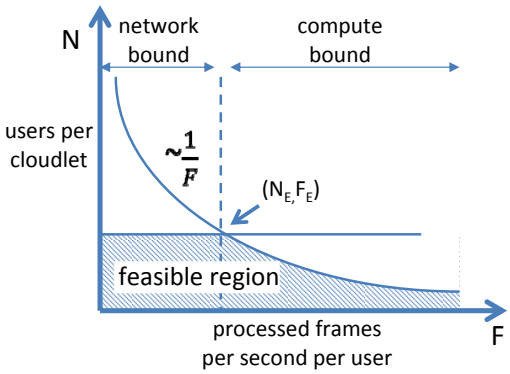


Figure 4: GigaSight for Cars



This figure illustrates the tradeoff between number of users (N) and processed framerate per user (F). The shaded region represents the range of feasible choices with fixed network and processing capacity.

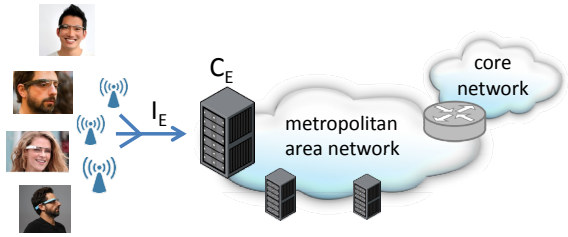
Figure 5: Cloudlet Sizing Tradeoff

Such transient local knowledge can also be provided when an automobile first associates with a cell tower.

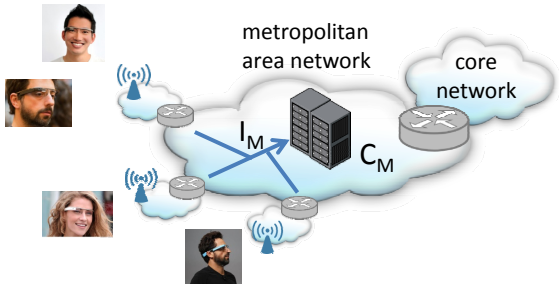
An automobile cloudlet could also perform real-time analytics of high data rate sensor streams from the engine and other sources. They can alert the driver to imminent failure or to the need for preventive maintenance. In addition, such information can also be transmitted to the cloud for integration into a database maintained by the vehicle manufacturer. Fine-grain analysis of such anomaly data may reveal model-specific defects that can be corrected in a timely manner.

6. Cloudlet Size and Placement

The scalability of GigaSight depends on the specific configurations of cloudlets and their locations in the Internet. Simoens et al [15] analyze the tradeoff between cloudlet computational capacity and number of cloudlets, with the goal of maximizing both the number N of simultaneous users per cloudlet, and the number F of denatured and indexed frames that each user contributes per unit of time. This conceptual tradeoff is illustrated in Figure 5. For values of $F < F_E$, the number of users supported is limited to



(a) Smaller Edge Cloudlets



(b) Larger MAN Cloudlets

Figure 6: Cloudlet Placement

N_E users. For values $F > F_E$, the architecture is compute bound and $N < N_E$.

Using measurements from the GigaSight prototype and extrapolating hardware improvements over a five-year timeframe, the analysis compares the two alternative design strategies that are illustrated in Figure 6. In Figure 6(a), many small cloudlets are deployed close to the edge of the Internet. In Figure 6(b), a single large cloudlet covers a city-sized area. The analysis concludes that edge cloudlets (Figure 6(a)) scale better than MAN cloudlets (Figure 6(b)) from the viewpoint of performance. In 5 years, the analysis estimates that one edge cloudlet would be able to support ~120 users with real-time denaturing and indexing at the rate of 30 fps. However, since management costs decrease with centralization, a more holistic analysis may suggest an optimum size that is somewhat larger than that suggested by performance considerations alone.

7. Conclusion

The central theme of GigaSight is processing edge-sourced data close to the point of capture in space and time. As the density of high data rate sensors in the IoT grows, it will become increasingly difficult to sustain the practice of shipping all captured data to the cloud for processing. The decentralized and federated architecture of GigaSight, using VMs on cloudlets for flexibility and isolation, offers a scalable approach to data collection. Sampling and denaturing data immediately after capture enables owner-specific lowering of fidelity to preserve privacy. Performing edge analytics (e.g. indexing) in near real-time on freshly-denatured data greatly improves the time-to-value metric of this data. The raw data at full fidelity is still available (during the fi-

nite storage retention period) for on-demand denaturing and "big data" processing. Finally, GigaSight supports interactive, content-based time-sensitive searches that were not anticipated by the indexer.

Acknowledgements

This paper is based on material that originally appeared in *"Scalable Crowd-Sourcing of Video from Mobile Devices"* [15].

This research was supported by the National Science Foundation (NSF) under grant number IIS-1065336, by an Intel Science and Technology Center grant, by DARPA Contract No. FA8650-11-C-7190, and by the Department of Defense (DoD) under Contract No. FA8721-05-C-0003 for the operation of the Software Engineering Institute (SEI), a federally funded research and development center. This material has been approved for public release and unlimited distribution (DM-0000276). Additional support was provided by IBM, Google, Bosch, Vodafone, and the Conklin Kistler family fund. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and should not be attributed to their employers or funding sources.

8. REFERENCES

[1] S. Banerjee and D. O. Wu. Final report from the NSF Workshop on Future Directions in Wireless Networking. National Science Foundation, November 2013.

[2] D. Barrett. One surveillance camera for every 11 people in Britain, says CCTV survey. *Daily Telegraph*, July 10, 2013.

[3] A. Davies. Here's Why So Many Crazy Russian Car Crashes Are Caught On Camera. *Business Insider*, December 2012.

[4] J. Fan, H. Luo, M.-S. Hacid, and E. Bertino. A novel approach for privacy-preserving video sharing. In *Proceedings of the 14th ACM international conference on Information and knowledge management*. ACM, 2005.

[5] M. Finnegan. Boeing 787s to create half a terabyte of data per flight says Virgin Atlantic. *ComputerWorld*, March 2013.

[6] J. Gertner. Behind GE's Vision for the Industrial Internet of Things. *Fast Company*, July/August 2014.

[7] H. Haddon. N.J. Mandates Cameras for Its New Police Cars. *Wall Street Journal*, September 11, 2014.

[8] Huston, L., Sukthankar, R., Wickremesinghe, R., Satyanarayanan, M., Ganger, G.R., Riedel, E., Ailamaki, A. Diamond: A Storage Architecture for Early Discard in Interactive Search. In *Proceedings of the 3rd USENIX Conference on File and Storage Technologies*, San Francisco, CA, April 2004.

[9] S. Larsson. *The Girl with the Dragon Tattoo*. Alfred A. Knopf, 2010.

[10] L. Miller, J. Toliver, and Police Executive Research Forum 2014. *Implementing a Body-Worn Camera Program: Recommendations and Lessons Learned*. Office of Community Oriented Policing Services, Washington, DC, 2014.

[11] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-Anonymity and its enforcement through generalization and suppression. Technical Report SRI-CSL-98-04, SRI Int., 1998.

[12] M. Satyanarayanan. Mobile Computing: The Next Decade. *SIGMOBILE Mobile Computing and Communication Review*, 15(2), August 2011.

[13] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies. The Case for VM-Based Cloudlets in Mobile Computing. *IEEE Pervasive Computing*, 8(4), October-December 2009.

[14] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, June 2008.

[15] P. Simoens, Y. Xiao, P. Pillai, Z. Chen, K. Ha, and M. Satyanarayanan. Scalable Crowd-Sourcing of Video from Mobile Devices. In *Proceedings of the 11th International Conference on Mobile Systems, Applications, and Services (MobiSys 2013)*, Taipei, Taiwan, June 2013.

[16] L. Westphal. The In-Car Camera: Value and Impact. *The Police Chief*, 71(8), August 2004.