IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 5, May 2017

Optimizing Cloud Base Video – CrowdSensing

Anitha J[1], Sapna B Kulkarni[2]

M.Tech 4th Sem, Dept. of CSE, RYMEC, Ballari, VTU University[1]

Associate Professor, Dept. of CSE, RYMEC, Ballari VTU University[2]

Abstract: Wearable and mobile devices are widely used for crowdsensing, as they come with many sensors and are carried everywhere. Among the sensing data, videos annotated with temporal-spatial metadata contain huge amount of information, but consume too much precious storage space. In this paper, we solve the problem of optimizing cloud-based video crowdsensing in three steps. First, we study the optimal transcoding problem on wearable and mobile cameras. We propose an algorithm to optimally select the coding parameters to fit more videos at higher quality on wearable and mobile cameras. Second, we empirically investigate the throughput of different file transfer protocols from wearable and mobile devices to cloud servers. We propose a real-time algorithm to select the best protocol under diverse network conditions, so as to leverage the intermittent WiFi access. Last, we look into the performance of cloud databases for sensor-annotated videos, and implement a practical algorithm to search videos overlapping with a target geographical region. Our measurement study on three popular opensource cloud databases reveals their pros and cons. The three proposed algorithms are evaluated via extensive simulations and experiments. The evaluation results show the practicality and efficiency of our algorithms and system. For example, our proposed transcoding algorithm outperforms existing approaches by 12 dB in video quality, 87% in energy saving, and one-quarter in delivery delay. Another example is, by intelligently choosing a proper cloud database; our system may reduce the insertion time by up to one-third, or the lookup time by up to one-fourth.

Keywords: Crowd Sensing and Crowd Sourcing, Cloud Ser-vices, Mobile and Ubiquitous Systems, Efficient Communications and Networking.

I. INTRODUCTION

Cloud-based video delivery becomes more popular, the demands placed upon the service provider's resources have dramatically increased. Service providers typically provision for the peak demands of each service across the subscriber population. However, provisioning for peak demands leaves resources underutilized at all other periods. This is particularly evident with Instant Channel Change (ICC) requests in video crowdsensing.

In video crowdsensing, Live TV is typically multicast from servers using IP Multicast, with one group per TV channel. Video-on-Demand (VoD) is also supported by the service provider, with each request being served by a server using a unicast stream. When users change channels while watching live TV, we need to provide additional functionality so that the channel change takes effect quickly. For each channel change, the user has to join the multicast group associated with the channel, and wait for enough data to be buffered before the video is displayed; this can take some time. As a result, there have been many attempts to support instant channel change by mitigating the user perceived channel switching latency [1], [2]. With the typical ICC implemented on video crowdsensing systems, the content is delivered at an accelerated rate using a unicast stream from the server. The playout buffer is filled quickly, and thus keeps switching latency small. Once the playout buffer is filled up to the playout point, the set top box reverts back to receiving the multicast stream.

Instant Channel Change (ICC) adds a demand that is proportional to the number of users concurrently initiating a channel change event [1]. Operational data shows that there is a dramatic burst load placed on servers by correlated channel change requests from consumer's .This results in large peaks occurring on every half-hour and hour boundaries and is often significant in terms of both bandwidth and server I/O capacity. Currently, this demand is served by a large number of servers grouped in a data center for serving individual channels, and are scaled up as the number of subscribers increases. However this demand is transient and typically only lasts several seconds, possibly up to a couple of minutes. As a result, majority of the servers dedicated to live TV sit idle outside the burst period.

Our goal in this paper is to take advantage of the difference in workloads of the different VIDEO CROWDSENSING services to better utilize the deployed servers. For example, while ICC workload is very busty with a large peak to average ratio, VoD has a relatively steady load and imposes "not so stringent" delay bounds. More importantly, it offers opportunities for the service provider to deliver the VoD content in anticipation and potentially out of- order, taking advantage of the buffering available at the receivers. We seek to minimize the resource requirements for supporting the service by taking advantage of statistical multiplexing across the different services - in the sense; we seek to satisfy the

DOI10.17148/IJARCCE.2017.65125

peak of the sum of the demands of the services, rather than the sum of the peak demand of each service when they are handled independently. Virtualization offers us the ability to share the server resources across these services.

II. LITERATURE SURVEY

Crowdsensing [5] has attracted significant attentions in smart city and urban computing communities. For example, Salim and Haque [7] survey large scale participation and citizen engagement in urban computing field deployments. Cesario et al. [8] present a high-level design of cloud-based urban computing architecture, and use it to analyze large scale mobile user trajectories. Lea and Blackstock [9] share similar view with us and propose a city-wide data storage to bridge sensors and smart city (or urban computing) applications. However, none of these studies consider the unique properties of video crowdsensing: videos are huge and take a lot of time to be uploaded. Thus, collecting videos are much more challenging compared to collecting other sensing data.

Video collection in smart cities and urban computing: Wearable (or mobile) devices are resource constrained and have to transcode videos from time to time. Several studies propose to transcode in resourceful clouds [10], [11]. In particular, Li et al. [10] design a cloud transcoder to facilitate video representations on diverse mobile devices. Jokhio et al. [11] focus on the resource allocation and scheduling problems of the transcoding process in data centers. These studies are not applicable to us, as we do not have the luxury of always-connected Internet and have to perform transcoding on wearable (or mobile) devices [12], which is much more challenging, due to less powerful CPU, limited storage space, and constrained battery capacity.

Video summarization is more aggressive on reducing video sizes, compared to video transcoding. Classical summarization algorithms [13], [14] are too heavy for wearable (or mobile) devices, and we only survey video summarization algorithms designed for wearable (or mobile) devices. Spriggs et al. [15] use Inertial Measurement Units (IMU) to temporally segment continuous human motion for activity classification. Kitani et al. [16] propose to summarize the action categories using an unsupervised learning algorithm. Lee et al. [17] propose to summarize videos based on the most important people and objects in them. Zhang et al. [18] propose to leverage sensing data for better video summarization. These video summarization algorithms are complementary to our proposed solution: video summarization can be added to our system when the resources are extremely scarce.

Opportunistic uploading videos: Opportunistic uploading files when network conditions are good have been shown to achieve up to 50% energy saving when using WiFi to offload cellular data [19], [20]. Wang et al. [21] study a transfer scheduling algorithm based on Markov Decision Process. Motivated by these studies, we leverage the widely-deployed public WiFi networks both in infrastructure and ad-hoc mode [22], [23] to upload transcoded videos. There exist complicated tradeoffs among storage space, upload time, energy consumption, and computation power, which require us to jointly consider the transcoding and uploading problem. This in turn makes our work different from others in the literature.

Video cloud databases: Opensource databases are widely used for online shopping, blog management, bigdata operation, and so on. In addition, geo-spatial friendly databases are becoming popular since geo-spatial data continues to play an important role in several applications, including smart cities and urban computing. Patroumpas etal [24] design evaluation methodologies of geo-spatial data management, including range searching and spatial indexing. In addition, they conduct performance measurements on several state-of-the-art semantic data management, such as OWLIM-SE, Strabon, and PostGIS. The data management of sensor-rich videos that provide information on viewable scenes is important for smart cities and urban computing. Arslan et al. Propose a view model of sensor-rich videos, which provide users sufficient information of videos for query videos. Lu et al. propose an efficient data management designed for querying the locations, the orientations, and the distances of the videos, respectively. Kim et al. Build a media management system with crowd sourced video collection with 4 dimensional meta-data management (where, who, what, and when). Our work lever-ages the existing databases for storing the collected sensor-rich videos. Moreover, we quantitatively and qualitatively evaluate the performance of the video databases using wearable (or mobile) cameras.

III. PROPOSED SYSTEM

Describes a system where a digital television service is delivered using the Internet Protocol over a network infrastructure, which may include delivery by a broadband connection. For residential users, video crowdsensing is often provided in conjunction with Video on Demand and may be bundled with Internet services such as Web access and VoIP. The commercial bundling of video crowdsensing, VoIP and Internet access is referred to as a Triple Play. Adding the mobile voice service leads to the Quadruple Play denomination. Video crowdsensing is typically supplied

DOI10.17148/IJARCCE.2017.65125

by a broadband operator using a closed network infrastructure. This closed network approach is in competition with the delivery of TV content over the public Internet. This type of delivery is widely called TV over Internet or Internet Television. In businesses, video crowdsensing may be used to deliver television content over corporate LANs and business networks. Perhaps a simpler definition of IPTV would be television content that, instead of being delivered through traditional formats and cabling, is received by the viewer through the technologies used for networks.

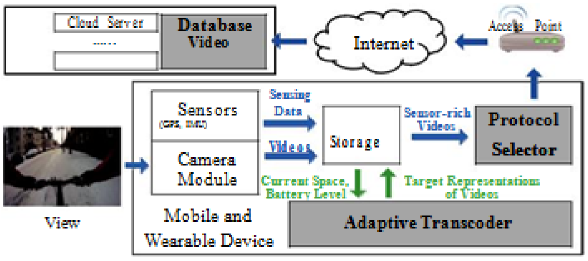

Fig. 1. Overview of our proposed system.

IV. RESULT AND DISCUSSION

A. Algorithms used in proposed system

- The Proposed Adaptive Transcoding Algorithm (ATA)

We develop a heuristic algorithm for our adaptive transcoding problem. The algorithm is based on two intuitions. The first one is to find, for each segment, the best representation that sacrifices the least video quality, yet releases the most storage space. More specifically, we identify the best representation of each segment in terms of the ratio of video quality over used storage space. The second intuition is to gradually cut down the storage usage of the segment that leads to the least quality degradation after saving the same amount of storage space. More specifically, we perform fine-grained adjustments with a step size δ between the current and the best representations chosen by the first intuition. Fig. 8 illustrates that different segments may lead to different quality degradation per unit saved segment size. In this figure, segment 1 has a flatter slope and is the best candidate to be pushed toward the best representation. The second intuition is repeatedly leveraged until enough storage space is freed.

Optimal Video Upload Protocol Selection

- Experimental Setup for Throughput Modeling

We conduct actual experiments to quantify the throughput achieved by different protocols under diverse network conditions In particular, we build a tested consisting of a sender, a receiver, and a Dummy net box. The sender and receiver are configured to support several data transfer protocols for the measurement study. Dummy net is a network emulator to recreate different network conditions, including bandwidth w, network round-trip delay r, and Packet Loss Rate (PLR) l. To systematically adjust the network conditions, we implement a controller program on the Dummy net box in C. The sender sends a command to the controller, which in turn emulates the specified network condition. The sender then transfers a file via the Dummy net box to the receiver using various file transfer protocols, and measures their throughput.

B. Video Cloud Base Design

- Overview

Fig.2 gives a high-level overview on the architecture of our proposed video cloud database. The database contains 4 components: (i) the video database that stores the meta-data of the uploaded videos, (ii) the intersection detection module that checks if the captured scenes of the videos overlap with the query region, (iii) the database manager that performs insertion, deletion, and query, and (iv) the smart city and urban computing applications that leverage lookup operations to satisfy users' needs. The database manager performs insertion once receiving the videos with the meta-data from the mobile and wearable devices. The database manager may delete the most outdated videos when the video database is full.

- Design

Arslan et al. propose Filed-Of-View (FOV) model to describe sensor-rich videos, and we utilize it. Each FOV record contains several components: (i) location p that represents the current location from GPS sensor, (ii) orientation d obtained from IMU, (iii) visual angle θ that describes the angular extent of the captured scene by camera, and (iv) range R that is the maximum distance for the camera to capture a recognizable object. We currently adopt FOV in 2D for

DOI10.17148/IJARCCE.2017.65125

brevity, but it can be readily extended to 3D FOV. FOV is very useful to provide users accurate information of a video or image. We put videos along with FOV records and other metadata and store them into the database. Table V shows the adopted metadata of videos in our system. Among them, the sample time is the timestamp of current sample and the time position is the time between the current time and the time when the video starts. Storing time position allows us to extract user- preferred segment in the videos instead of providing the whole video to the users.

We aim to efficiently store, manage, and lookup the sensor-rich videos. We look into three popular opensource databases:
(i) PostgreSQL (PostGIS), (ii) MySQL, and (iii) MongoDB, are summarized below:
PostgreSQL (with PostGIS extension) is designed for geo-spatial storage, which allows more complicated geo-location queries to be run in SQL.
MySQL is one of the most popular databases used in many website applications such as online shopping and blogging. It is designed for general purpose. Newer version of MySQL adds initial supports on geo-spatial queries.
MongoDB is a Not-only-SQL (NoSQL) database which is document-oriented. It is suitable for big-data operations, which may contain unstructured and schema less data.

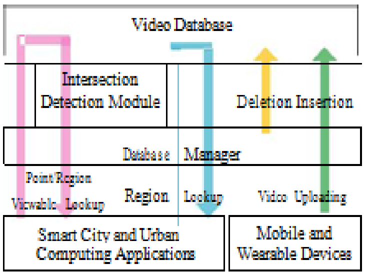
Video Database
Intersection Detection Module
Deletion Insertion
Database Manager
Point/Region Viewable Lookup
Region Lookup
Video Uploading
Smart City and Urban Computing Applications
Mobile and Wearable Devices

Fig. 2. The overview of the components in our video cloud database.

➢ Evaluation Setup
We conduct experiments using the considered database and make comparisons among them. Our adopted video dataset contains 2531 videos (286756 records). We consider the following performance metrics:

- **Insertion time**, which is the time consumed for performing insertion (VideoCollect).
- **Deletion time**, which is the time consumed for performing deletion (VideoDelete).
- **Region lookup time**, which is the time consumed for querying the videos recorded in a particular region (RegionLookup).
- **Point viewable lookup time** (ViewableLookup), which is the time consumed for looking up the videos overlapping with a particular location.
- **Region viewable lookup time** (ViewableLookup), which is the time consumed for lookup the videos overlapping with a particular region.
- **Storage overhead**, which is the space used by the database.

C. Future Enhancement

Optimizing cloud-based Video Crowdsensing future enhancements are really important to make this system as a platform for video-sharing, uploading videos in diverse environment and maintaining temporal-spatial properties of the video more diverse to improve ranking & indexing feature to speed up search of these videos.
Above goal can be achieved by increasing the accuracy of the time and energy models, which is one of future task of this project. Than investigating alternatives for better platform than existing one to store online videos. Currently existing databases poses limitation like delay & latency issues while storing or retrieving videos. Also, storing technique is designed for storing contents in text forms. Therefore, these databases are architecturally limited by its design to store or manipulate videos in their data structures. Therefore, looking new technologies like NoSql might be the better fit for storing videos. Which stores data with no relation but before that we have to know the scope and limitation of the new databases too.

VI. CONCLUSION

Optimizing cloud-based video crowdsensing service providers can leverage a virtualized cloud infrastructure and intelligent time-shifting of load to better utilize deployed resources. Using Instant Channel Change and VoD delivery as examples, we showed that we can take advantage of the difference in workloads of video crowdsensing services to schedule them appropriately on virtualized infrastructures. By anticipating the LiveTV ICC bursts that occur every half hour we can speed up delivery of VoD content before these bursts by prefilling the set top box buffer. This helps us to dynamically reposition the VoD servers to accommodate ICC bursts that typically last for a very short time. Our paper provided generalized framework for computing the amount of resources needed to support multiple services with deadlines. We formulated the problem as a general optimization problem and computed the number of servers required according to a generic cost function. We considered multiple forms for the cost function and solved for the optimal number of servers that are required to support these services without missing any deadlines. We implemented a simple time-shifting strategy and evaluated it using traces from an operational system. Our results show that anticipating ICC bursts and time-shifting VoD load gives significant resource savings. We also studied the different parameters that affect the result and show that their ideal values vary over time and depend on the relative load of each service mechanisms as part of our future work.

REFERENCES

[1] S. Wang, C. Fan, Y. Huang, and C. Hsu, “Toward optimal crowdsens-ing video quality for wearable cameras in smart cities,” in Proc. of IEEE International Workshop on Smart Cities and Urban Informatics (SmartCity’15), Hong Kong, China, April 2015, pp. 624–629.

[2] “World Urbanization Prospects,” 2011, http://http://www.un.org/en/ development/desa/news/population/world-urbanization-prospects-2014. html/.

[3] H. Schaffers, N. Komninos, M. Pallot, B. Trousse, M. Nilsson, and A. Oliveira, “The future Internet,” J. Domingue, A. Galis, A. Gavras, T. Zahariadis, and D. Lambert, Eds. Berlin, Heidelberg: Springer-Verlag, 2011, ch. Smart Cities and the Future Internet Towards Cooper-ation Frameworks for Open Innovation, pp. 431–446.

[4] Y. Zheng, L. Capra, O. Wolfson, and H. Yang, “Urban computing: Con-cepts, methodologies, and applications,” ACM Transaction on Intelligent Systems and Technology, vol. 5, no. 3, pp. 1–55, 2014.

[5] R. Ganti, F. Ye, and H. Lei, “Mobile crowdsensing: current state and future challenges,” IEEE Communications Magazine, vol. 49, no. 11, pp. 32–39, 2011.

[6] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, “En-ergy consumption in mobile phones: A measurement study and implica-tions for network applications,” in Proc. of ACM SIGCOMM conference on Internet measurement conference (IMC’09), Chicago, IL, November 2009, pp. 280–293.

[7] F. Salim and U. Haque, “Urban computing in the wild: A survey on large scale participation and citizen engagement with ubiquitous computing, cyber physical systems, and Internet of Things,” International Journal of Human-Computer Studies, vol. 81, pp. 31–48, March 2015.

[8] E. Cesario, C. Comito, and D. Talia, “Towards a cloud-based framework for urban computing, the trajectory analysis case,” in Proc. of IEEE International Conference on Cloud and Green Computing (CGC’13), Karlsruhe, Germany, September 2013, pp. 16–23.

[9] R. Lea and M. Blackstock, “City Hub: A cloud-based IoT platform for smart cities,” in Proc. of IEEE International Conference on Cloud Com-puting Technology and Science (CloudCom’14), Singapore, December 2014, pp. 799–804.

[10] Z. Li, Y. Huang, G. Liu, F. Wang, Z. Zhang, and Y. Dai, “Cloud transcoder: Bridging the format and resolution gap between Internet videos and mobile devices,” in Proc. of ACM International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV’12), Toronto, Canada, June 2012, pp. 33–38.

[11] F. Jokhio, A. Ashraf, S. Lafond, I. Porres, and J. Lilius, “Prediction-based dynamic resource allocation for video transcoding in cloud computing,” in Proc. of Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP’13), Belfast, UK, February 2013, pp. 254–261.

[12] B. Shen, W. Tan, and F. Huve, “Dynamic video transcoding in mobile environments,” IEEE MultiMedia, vol. 15, no. 1, pp. 42–51, 2008.

[13] Y. Ma, L. Lu, H. Zhang, and M. Li, “A user attention model for video summarization,” in Proc. of ACM International Conference on Multimedia (MM’02), Juan-les-Pins, France, Decenber 2002, pp. 533– 542.

[14] C. Ngo, Y. Ma, and H. Zhang, “Video summarization and scene detection by graph modeling,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 15, no. 2, pp. 296–305, February 2005.

[15] E. Spriggs, F. Torre, and M. Hebert, “Temporal segmentation and activity classification from first-person sensing,” in Proc. of IEEE Com-puter Society Workshops On Computer Vision and Pattern Recognition (CVPR’09), Miami, FL, June 2009, pp. 17–24.

[16] K. Kitani, T. Okabe, Y. Sato, and A. Sugimoto, “Fast unsupervised ego-action learning for first-person sports videos,” in Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR’11), Colorado Springs, CO, June 2011, pp. 3241–3248.

[17] Y. Lee, J. Ghosh, and K. Grauman, “Discovering important people and objects for egocentric video summarization,” in Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR’12), Providence, Rhode Island, June 2012, pp. 1346–1353.

[18] Y. Zhang, H. Ma, and R. Zimmermann, “Dynamic multi-video summa-rization of sensor-rich videos in geo-space,” Advances in Multimedia Modeling, vol. 7732, pp. 380–390, 2013.

[19] A. Balasubramanian, R. Mahajan, and A. Venkataramani, “Augmenting mobile 3G using WiFi,” in Proc. of ACM International Conference on Mobile Systems, Applications, and Services (Mobisys’10), San Fran-cisco, CA, June 2010, pp. 209–222.

[20] K. Lee, J. Lee, Y. Yi, I. Rhee, and S. Chong, “Mobile data offloading: How much can wifi deliver?” IEEE/ACM Transactions on Networking, vol. 21, no. 2, pp. 536–550, 2013.

[21] Y. Wang, X. Liu, A. Nicoara, T. Lin, and C. Hsu, “Smarttransfer: Transferring your mobile multimedia contents at the right time,” in Proc. of ACM International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV’12), Toronto, Canada, June 2012, pp. 71–76.

[22] T. Karkk¨ainen,¨ M. Pitkanen,¨ and J. Ott, “Enabling ad-hoc-style commu-nication in public wlan hot-spots,” ACM SIGMOBILE Mobile Computing and Communications Review, vol. 17, no. 1, pp. 4–13, 2013.

[23] S. Trifunovic, B. Distl, D. Schatzmann, and F. Legendre, “WiFi-opp: ad-hoc-less opportunistic networking,” in Proc. of ACM Workshop on Challenged Networks (CHANTS’11), Las Vegas, NV, September 2011, pp. 37–42.

[24] K. Patroumpas, G. Giannopoulos, and S. Athanasiou, “Towards geospa-tial semantic data management: strengths, weaknesses, and challenges ahead,” in Proc. of ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL’14), Dallas, TX, November 2014.