Scalable Crowd-Sourcing of Video from Mobile Devices

Pieter Simoens , Yu Xiao , Padmanabhan Pillai , Zhuo Chen, Kiryong Ha, Mahadev Satyanarayanan

School of Computer Science
Carnegie Mellon University

PRESENTATION MATERIAL & PRESENTER:

SAMARTH V SHASTRI

Roadmap for the presentation

- ✓ Motivation
- ✓ Scenario
- ✓ Author Published Model
- ✓ Architecture
- ✓ Evaluation
- ✓ Future Work
- ✓ Conclusion

Motivation for crowd sourcing of videos from Mobile Devices

- Marketing and Advertising
- Locating people, pets and thing
- Public safety
- Fraud detection

Components used: Head-up displays(Google glasses), Mobile

Author Published Model

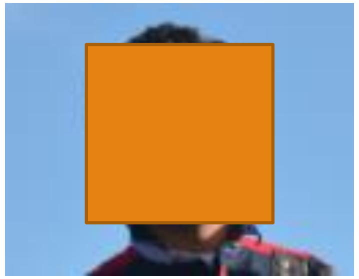
- Goal to publish a plausible incentive model
- Crowd sourced videos must be treated as authored content
- Create a business relation with the service provider that invests in the video capture infrastructure. Like a book publisher, the service provider monetizes individual’s authored content and revenues

Incentive model- *"you reap rewards for capturing and sharing scenes that others find to be of value"*

Denaturing

- Editing to reflect personal preference before submitting
- Privacy vs Value
- Computer vision algorithms- face detection, face recognition and object recognition
- Meta data modification

I don’t want my
face to be seen



Scenario

What do I do☹

Giga Sight

To be continued.....

Architecture

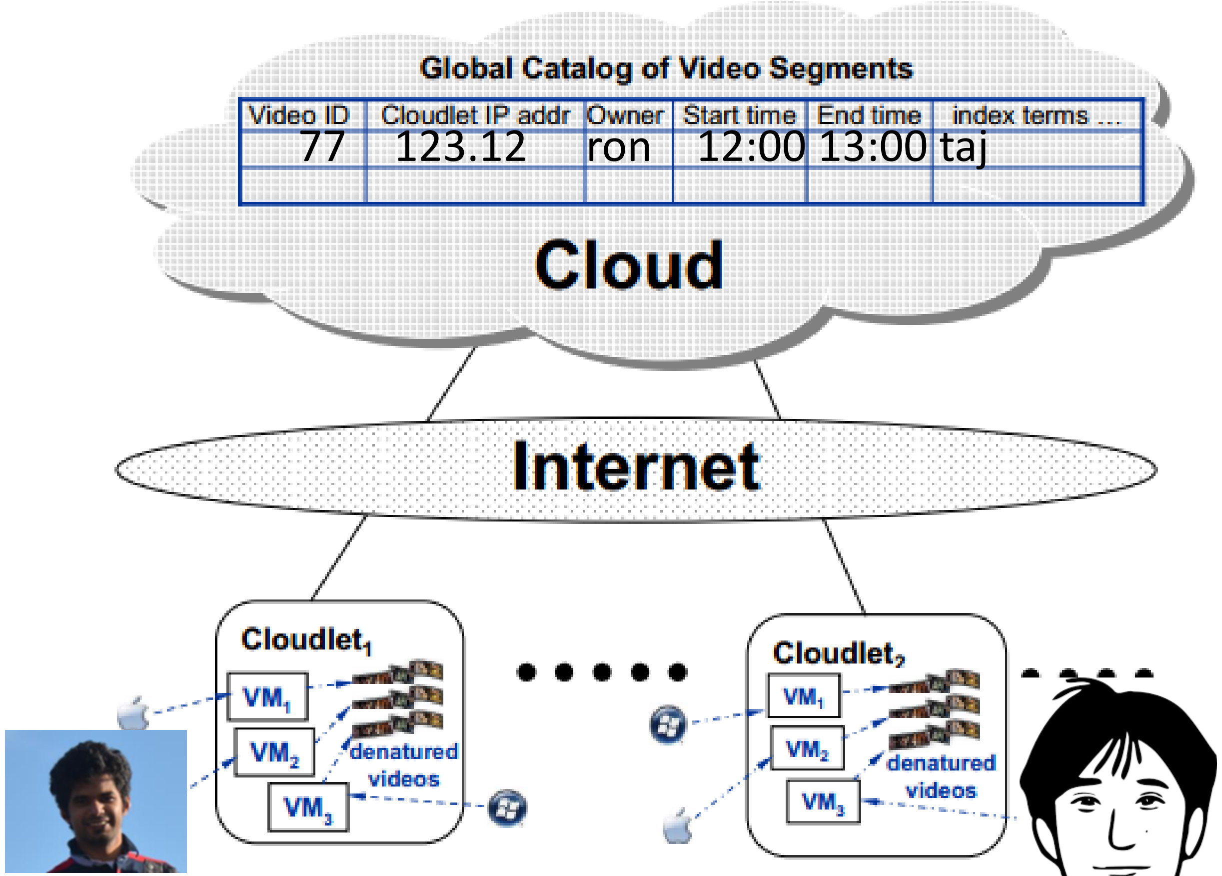


Figure 3: GigaSight Architecture.

More about architecture….

- Similar to CDN in reverse
- Denatured videos are stored on the cloudlets
- Metadata about video segments(such as owner, location of capture, start and end time of capture, cloudlet where stored and index terms) is stored in the global catalog of cloud
- Denaturing is performed on the cloudlets
- Cloudlets encapsulate image processing code to perform background indexing of video-segments. Users can also manually add tags to their video segments.

GigaSight Implementation

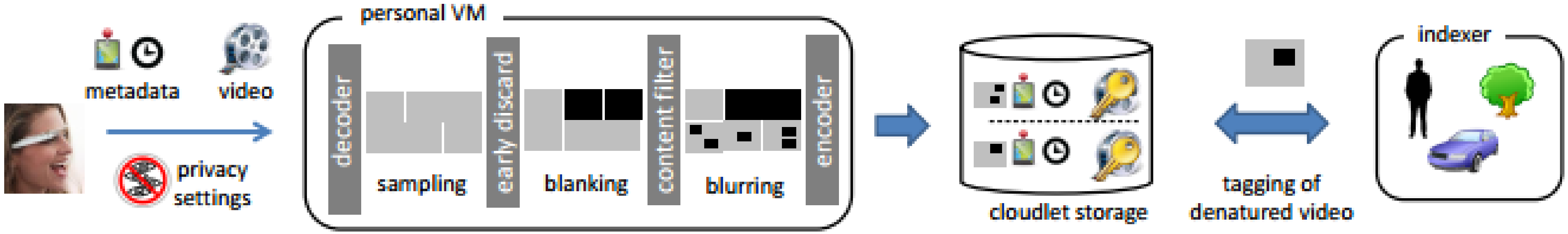
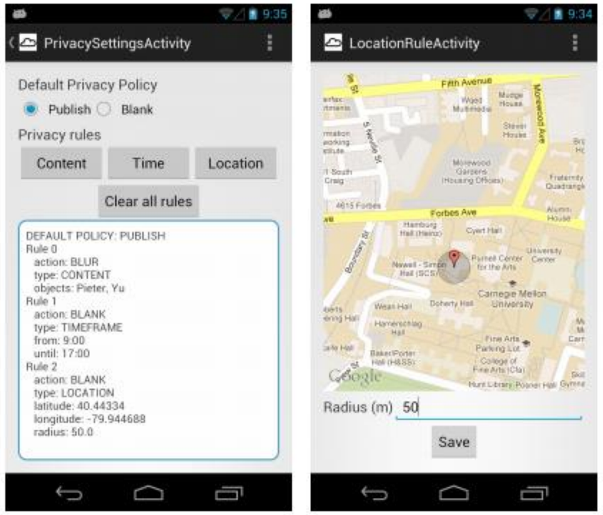


Figure 4: Overview of the GigaSight implementation.

Mobile Client

- Smartphone act as a proxy to the GigaSight framework where video segments are buffered during periods without network connectivity
- User specific privacy settings(publish or blank all his video segments)
- Captured video are cached and uploaded as soon as Wi-Fi connections becomes available through REST-based API



(a) Example privacy rule (b) Example location filter

Figure 5: Screenshots of the GigaSight app.

Personal VM

- Denatures the video uploaded by mobile client following user defined privacy rules
- Requests data manager to allocate space for video to be stored
- Denaturing involves:

    1)Subset of the video frames is selected for actual denaturing

    2) Early discard: based on time and location, frame is completely blanked or kept unmodified

    3)Complex content based filters (decoded to RGB frames->Grey Scale->Histogram->face detection)

- Output is a low-framerate video file that is stored on the cloudlet storage system
- Encrypted version of the original video is stored

Figure 6: Example of a denatured video frame.

Data Manager

- Runs in an individual VM on the cloudlet
- It manages the storage of the video segments and the database with associated metadata
- All video and metadata are stored on local disk storage in the cloudlet that is made accessible to local VM’s as Network File System(NFS) mounted(access through REST- interface)
- Users connected to a specific cloudlet will upload videos that are mostly captured in the same geographic area of the cloudlet

Video Content Indexer

- It is a background activity performed by a separate VM on the cloudlet
- Computer vision code(Clapping, falling and other actions)
- For each indexed frame an entry is created in the table of the cloudlet database
- Tags are exported from the denaturing process
- Tags of a frame are the union of those obtained during denaturing by the user's image processing code within his personal VM and those obtained during indexing by cloudlet's image processing code

Search Workflow

- This involves two steps
- First step: User performs a conventional SQL search on the cloud-wide catalog which may involve metadata such as time and location as well as tags extracted by indexing
- First step may be overwhelming to the user
- Second step: Using image processing code such as filters to reduce the returned results to a more relevant set (discard irrelevant parts of video)

"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."

Continued….

Evaluation

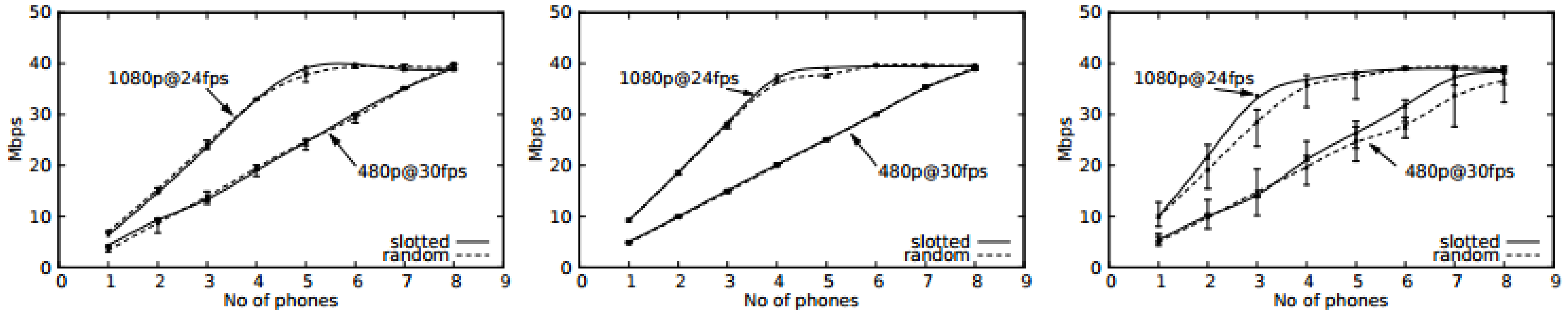
Experiment Set up

Table 1: Hardware used for the experiments

smartphone	Samsung Google Galaxy Nexus I9250 Android v4.0.4 Dual-core 1.2 GHz Cortex-A9 Wi-Fi 802.11 a/g/n 1 GB RAM, 16 GB internal memory
cloudlet	Intel® Core™ i7-3770 CPU @ 3.40 GHz Linux 3.2.0 x86_64 CPU MHz 1600, 4 CPU cores 32 GB RAM, 900 GB hard disk
Wi-Fi AP	Belkin N750 DB Wi-Fi Dual-Band N+ 2.4 GHz / 5 GHz (link rate up to 450 Mbps)

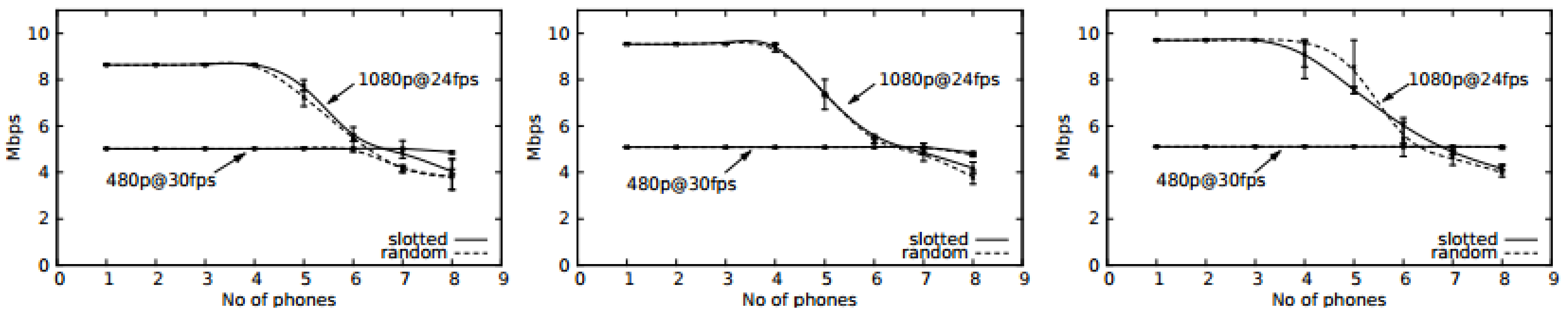
Cost of Capture

- Experiments conducted with actual video upload to the cloudlet.(No overhead involving virtualization, denaturing or indexing considered)
- Two methods were used for upload: Random and slotted
- Random: Smartphones start uploading after random delays in the interval [0,n] with n equal to the length of video segments used.
- Slotted: Interval of n seconds is divided inti a number of slots equal to the number of participating phones. This mechanism tries to avoid as much as collisions between other phones.



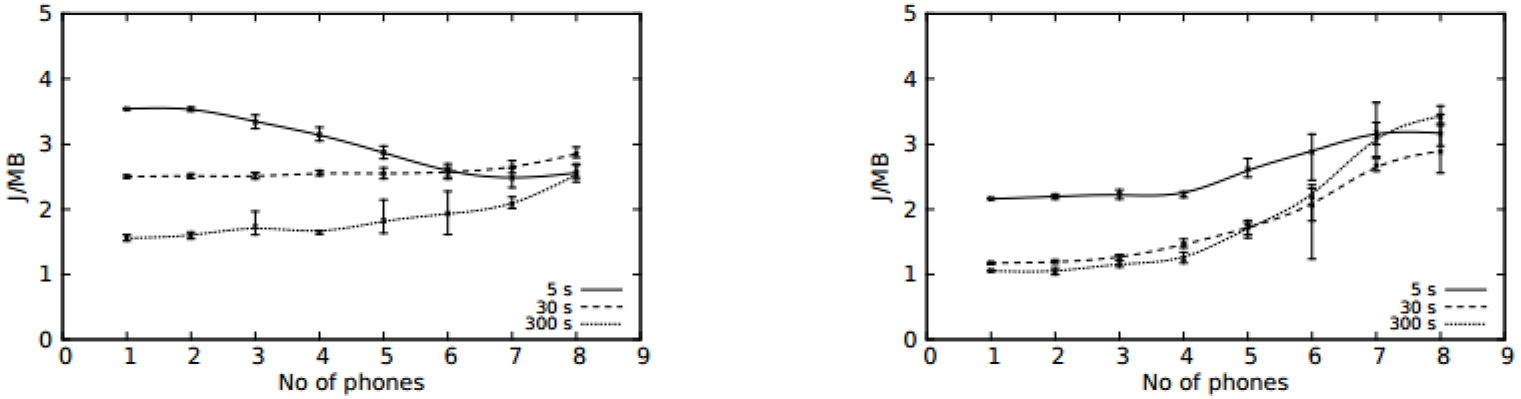
(a) 5s (b) 30s (c) 300s

Cumulative throughput



(d) 5s (e) 30s (f) 300s

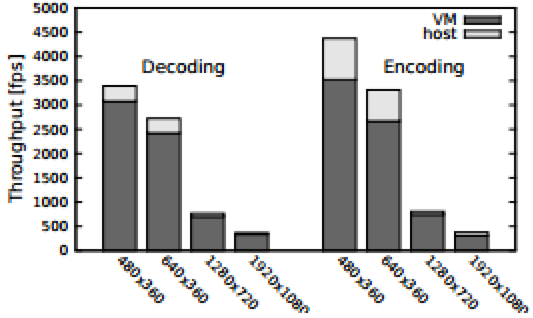
Net throughput achieved by each phone



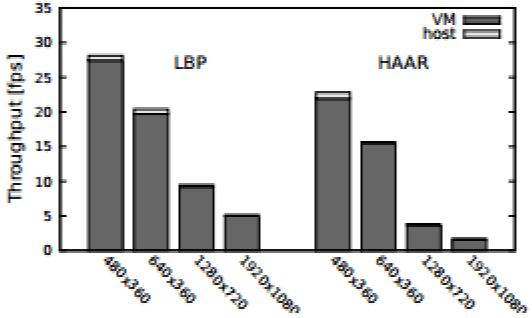
(a) 480p@30fps (b) 1080p@24fps

Average Energy consumption of smartphone

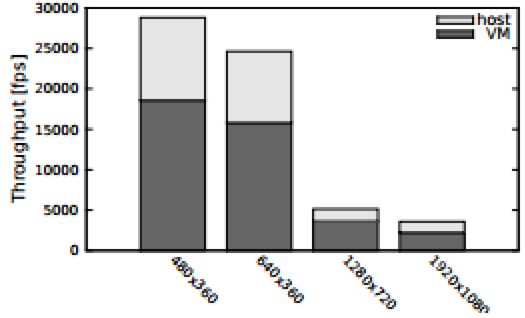
Cost of preserving privacy



(a) Video Decoding and Encoding



(b) Face Detection and Blurring



(c) AES Encryption

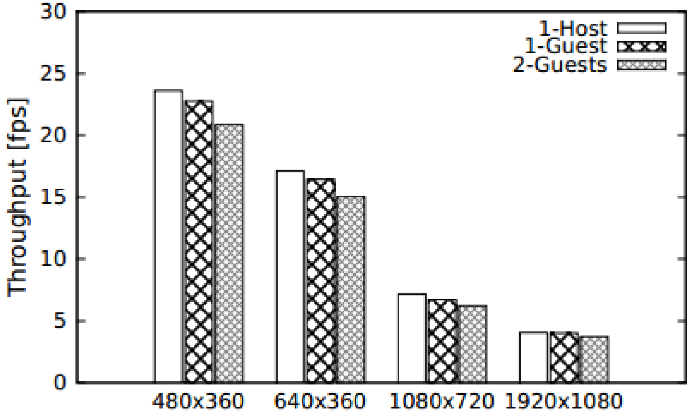
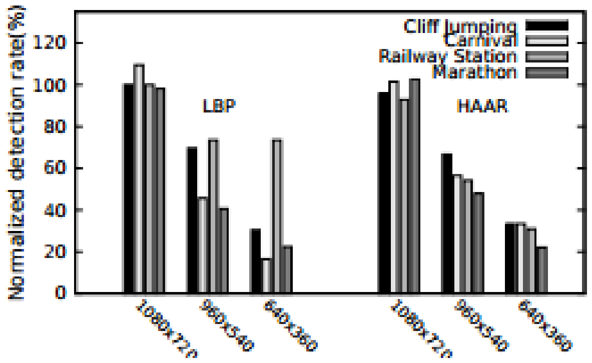
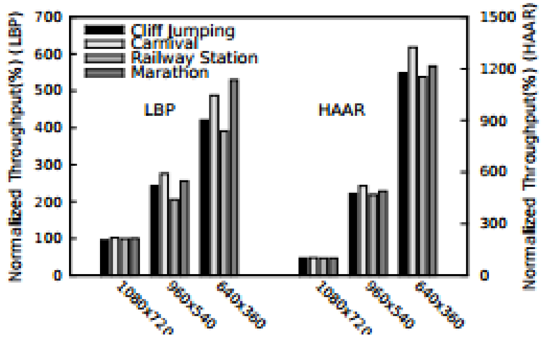


Figure 10: Overall throughput of the personal VM

- By enabling hyper threading each machine can provide up to 8 virtual CPUs that must be distributed between VM running on host
- Content based filtering is the only step which is paralyzed
- Computer vision algorithms are computationally expensive
- Reducing resolution is a clear path to increase the throughput of personal VM
- Scaling down the resolution prior to denaturing results in a higher throughput but has a clear impact on the accuracy



(a) Accuracy vs. Resolution



(b) Throughput vs. Resolution

Cost of content search

Classes	Total Tags	720p		480p		360p	
		TP (rate)	FP	TP (rate)	FP	TP (rate)	FP
aeroplane	4	4 (100.0%)	0	4 (100.0%)	0	4 (100.0%)	0
bicycle	0	0 (N/A)	2	0 (N/A)	2	0 (N/A)	11
bird	68	63 (92.6%)	4	59 (86.8%)	7	57 (83.8%)	15
boat	0	0 (N/A)	0	0 (N/A)	0	0 (N/A)	0
body	2408	2352 (97.7%)	19	2284 (94.9%)	24	2185 (90.7%)	23
book	74	74 (100.0%)	20	74 (100.0%)	60	74 (100.0%)	92
building	875	870 (99.4%)	137	856 (97.8%)	219	836 (95.5%)	273
car	122	118 (96.7%)	52	111 (91.0%)	136	108 (88.5%)	176
cat	573	560 (97.7%)	10	552 (96.3%)	15	546 (95.3%)	24
chair	22	22 (100.0%)	1	20 (90.9%)	2	19 (86.4%)	5
cow	0	0 (N/A)	0	0 (N/A)	0	0 (N/A)	0
dog	1004	965 (96.1%)	14	921 (91.7%)	25	882 (87.8%)	35
face	523	504 (96.4%)	32	468 (89.5%)	59	464 (88.7%)	65
flower	12	11 (91.7%)	0	10 (83.3%)	0	10 (83.3%)	0
grass	131	126 (96.2%)	0	120 (91.6%)	0	116 (88.5%)	0
road	370	351 (94.9%)	24	317 (85.7%)	41	291 (78.6%)	50
sheep	2	2 (100.0%)	0	2 (100.0%)	0	1 (50.0%)	0
sign	125	124 (99.2%)	10	120 (96.0%)	12	115 (92.0%)	17
sky	1409	1338 (95.0%)	1	1210 (85.9%)	1	1074 (76.2%)	3
tree	1080	1066 (98.7%)	97	1040 (96.3%)	189	984 (91.1%)	234
water	1290	1208 (93.6%)	4	1145 (88.8%)	6	1102 (85.4%)	8
total	10092	9758 (96.7%)	427	9313 (92.3%)	798	8868 (87.9%)	1031

s are based on the indexing of 3583 frames obtained by sampling 1 frame per 2 seconds from 12

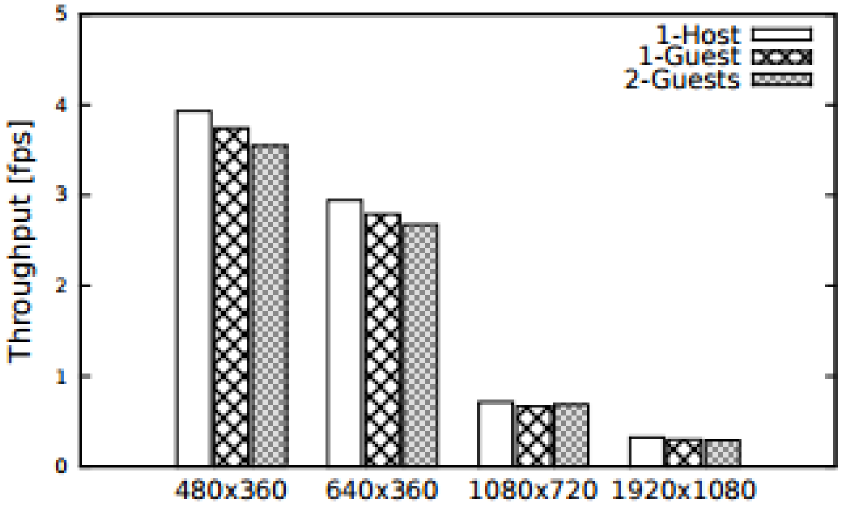


Figure 12: Throughput of the indexing process

Future Work

- Performance of Computer Vision Algorithms
- Storage management and reclamation
- Semantic deduplication
- Use of more distributed databases
- Visual content tagging(Social Network)

Conclusion

- Crowd-sourced videos pave way for many future applications
- GigaSight provides a framework to implement crowd-sourcing from initial capture of video to visual content search
- GigaSight protects privacy of people to a certain extent
- Gigasight extends the capability of HUDs such as google glasses

Github link: https://github.com/cmusatyalab/GigaSight

Comments

- Dash Camera(Driving recorder) can be used to record video
- There is no comment as to how to achieve optimized uploading of huge cached video data captured during no internet connectivity
- Authors give importance to privacy aspect of users sharing videos
- Good platform for future work in video crowdsourcing
- Not suitable for mobile devices as continuous video capturing will have a negative affect on mobile battery

Thank you