A Contributory Public-Event Recording and Querying System

Arun Joseph, Nikita Yadav, Vinod Ganapathy
(Indian Institute of Science, Bangalore, India)
Dushyant Behl (IBM Research, India)

Computer Systems Security Laboratory
Indian Institute of Science, Bangalore

How many CCTV cameras are in town?

- CCTV (Closed-Circuit Television) technology is widely used for security and surveillance
- Delhi, India, has **1446 cameras per square mile**, with both public and private installations
- London, UK boasts over 942,562 CCTV cameras, including private cameras. In London, a person can be captured on CCTV up to **70 times a day**



Number of CCTV cameras in some of the world's populated cities

City	Cameras	Per Square Miles	Per 1000 People
Delhi, India	436600	1446	26.7
Chennai, India	282126	614	24.53
Singapore, Singapore	108981	281	18.04
Seoul, South Korea	77814	333	7.8
Moscow, Russia	213000	219	16.85
London, UK	127373	210	13.35
New York, USA	56190	187	6.87
Cities of China*	54000000	1000	372

Why private parties should join such a network?

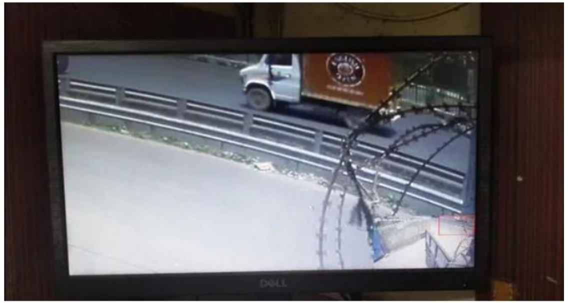
- Sharing public CCTV footage enhances overall surveillance and security in the area
- Lead to cost savings as multiple organizations share the infrastructure
- To prove video recording is tamper proof.
- Other cameras can provide continuous surveillance coverage even when our camera is
  - damaged
  - malfunctioning
  - technical problems
- Reducing blind spots

HT Hindustan Times

Accident response team solves 10 hit-and-run cases in 10 days

A six-member accident response team (Art) of the Gurugram Police is revisiting black spots and hit-and-run cases to identify drivers who fled the spot after hitting motorists and pedestrians this year. The police said on Tuesday that the team has solved 10 blind cases in the last 10 days

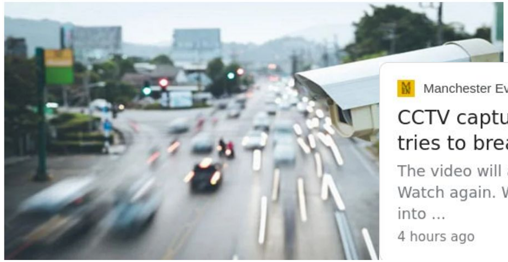
By Leena Dhankhar, Gurugram
PUBLISHED ON JUL 06, 2021 11:23 PM IST



The police analyse CCTV footage to track the suspects. (Sourced)

Mumbai: New camera system will detect stolen vehicles entering city

By Sachin Gaad



Manchester Evening News

CCTV captures moment burglar posing as delivery ... tries to break into home

The video will auto-play soon8Cancel. Play now. Video will play in. Watch again. Watch as burglar uses fake parcel in attempt to break into ...

4 hours ago

If we create a public network including all these parties

- Public authorities can make use of extra video footage, especially from private cameras, to support their investigations
- Help to identify the fleeing vehicle in a hit-and-run accident
- With multiple sources of footage cross-reference is possible to verify information

Challenges to build a contributory CCTV network

1. Untrustworthy Video Feed:
    a. Video feeds can be compromised by technical errors or malicious activities, leading to damaged or edited footage that is unreliable for investigations
    b. Organizations may intentionally share incorrect footage for their benefit or to deceive others
2. Privacy of Video Feed:
    a. Private organizations reluctant to share raw video feeds due to data security and privacy concerns
    b. Unauthorized access and misuse of footage hinder their participation in a contributory CCTV network
3. Variable Quality of Video Feed:
    a. CCTVs with varying qualities, viewing angles, aspect ratios, and color calibrations complicates the analysis
    b. Inconsistencies can arise due to unsynchronized timestamps on different cameras, and the sheer volume of data generated poses challenges in processing and analysis
4. Availability of CCTV Footage:
    a. The absence of CCTV cameras in certain areas results in surveillance blind spots.
    b. Camera malfunctions can also lead to gaps in footage

Untrustworthy Video Feed

Fake insurance claim

Hide the mistake/flaw lead to crime/accident

Doing some illegal activities

Destroying the evidence for crime

Create scam videos/ fake proofs

4 WWL-TV

Husband, wife who got surgeries for fake injuries in 18-wheeler insurance scam sentenced

... case that has exposed a network of scam artists who staged auto accidents with 18-wheelers in order to file fraudulent insurance claims.

2 weeks ago

Daily Express

'Be vigilant and be aware': Rise in ‘crash for cash’ scams expected after lockdown easing

Robin Challand, the Claims Director at Ageas Insurance, ... start to re-emerge and put innocent motorists at risk by causing accidents.

14 hours ago

TNIE The New Indian Express

CCTVs help unearth fake chain-snatching claims in Hyderabad

Analysis of CCTV footage didn't reveal any such incident. When confronted with the evidence, the women admitted to have lodged false complaints, ...

2 weeks ago

TH The Hindu

Form SIT to probe fake motor insurance claims, HC tells DGP

The judge ordered that if any false insurance claim petitions get ... motor accident claims having been made against it using fake insurance ...

10-Feb-2021

THE HINDU

Public Event Recording and Querying System (PERQS)

- PERQS is an innovative **distributed network of CCTV cameras**
- PERQS is a collaborative CCTV system allows users to submit queries
- The system utilizes the entire network of camera feeds, leveraging **advanced video analytics techniques** to provide responses
- Eliminates the need for manual review of individual camera feeds
- PERQS introduces **a novel consensus-based video analysis** concept to address trust and privacy challenges in a contributory CCTV network
- Multiple cameras participate in query execution to reach results, enhancing accuracy and reducing the risk of false positives

How PERQS address the four challenges

I. Tamper-Proof Video Feed:
   - Video-hash commitment and verification to ensure video authenticity

II. Private Video Storage:
   - PERQS shares only video analysis results, preserving raw video privacy
   - Eliminates the need for vast central storage facilities

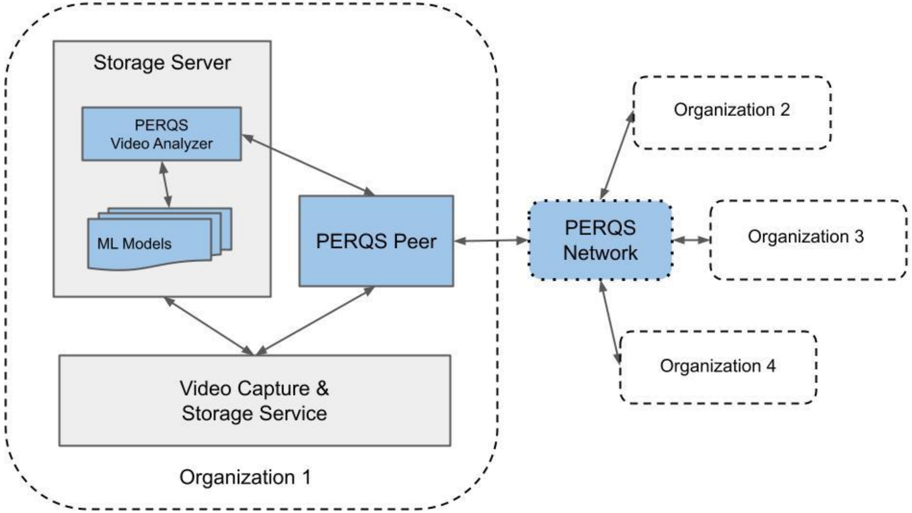
III. Consensus-Based Video Analysis:
   - The use of consensus to enhance reliability by involving multiple cameras in query execution
   - Faults, damage, or malicious behavior of individual cameras do not affect the output

IV. Time-Travel Consensus:
   - Introduction of a time-travel consensus mechanism to address blind spots and camera faults
   - Uses footage from other cameras to answer queries and fill gaps in recordings
   - "**Travels through time and space**" to access pre-event or post-event footage using nearby cameras for consensus

PERQS Design

- Blockchain-Based Architecture:
  - The necessity for mutual distrust among participants, video feed hash commitment, and consensus led to the adoption of a blockchain-based architecture
  - Consensus and hash verification establish collective correctness, ensuring system accuracy and reliability
- PERQL Query Language:
  - Developed the PERQL query language, similar to SQL, which is an extension of FRAMEQL
  - Provides users with a flexible and customizable queries
- Participant Analysis Facilities:
  - Each participant maintains their analysis facility for video analysis within the system
- Integration of Features in PERQS:
  - PERQS combines novel query consensus, time-travel consensus, tamper-proof videos, and the PERQL query language to address the challenges discussed

PERQS Architecture



- PERQS Network:
  - A permissioned blockchain network that connects all participating CCTV owner organizations in PERQS
  - Membership is verified, ensuring that only actual CCTV owner organizations can participate
- PERQS Client:
  - The blockchain peer that facilitates organization participation in the PERQS network
  - Multiple roles include committing video hashes to the blockchain, decoding and sending queries for execution, and reaching consensus on query results
- Video Capture & Storage Service:
  - Manages CCTV cameras within an organization and handles video storage through a central system
- PERQS Video Analyzer:
  - Responsible for conducting video analysis on recorded videos
  - Receives queries, decodes and executes them on individual video feeds

Query Language PERQL

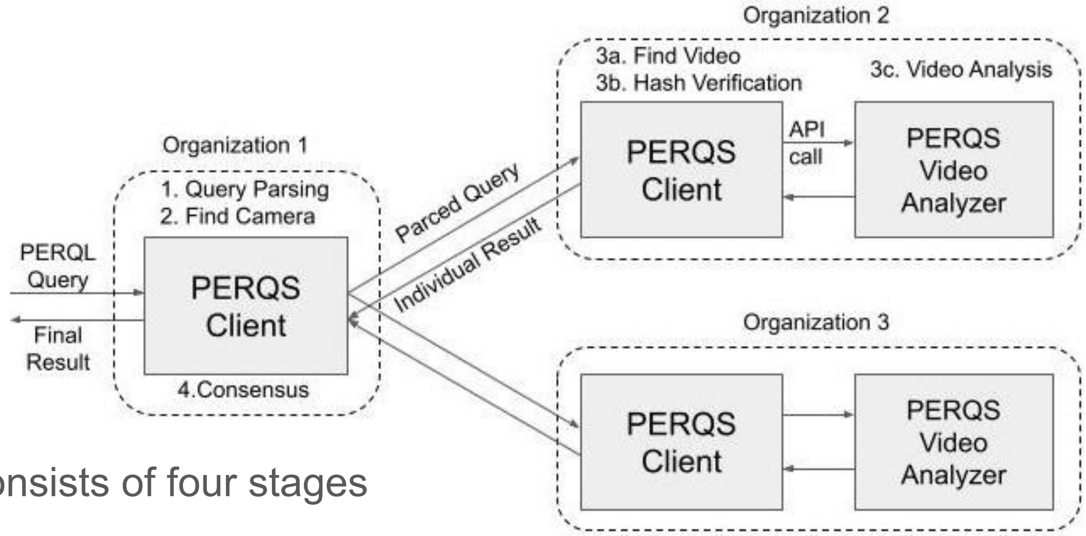
- CREATE MODEL
  - To add a Machine Learning (ML) or vision model
- ALTER MODEL
  - To changes or updates existing models
  - It is used to propagate these modifications to all participating organizations
- SELECT
  - Used to extract information from the collective video database within PERQS
  - Narrows down video feeds using GPS location and time data
  - Video analyzer server uses an ML/vision model for analysis, generating a table with results

```
CREATE MODEL <model_name>
(
<data_field1> <data_type>,
<data_field2> <data_type>,
...)
```

```
ALTER MODEL <model_name>
ADD <data_filed> <data_type>
```

```
SELECT <fields>
FROM <model_name>
WITH <feature_parameters> -- optional
AT <gps_loc>
WHERE <conditions> -- optional
TIME <start_time> TO <end_time>
```

Query Execution



A single query execution in PERQS consists of four stages

- Query Parsing:
  - Decode keywords and parameters from the query string
- Find the Camera:
  - Identify cameras of interest based on GPS location
  - “camera-finder" oracle returns the camera ID given GPS location and range
- Owner Query Execution:
  - CCTV owners locate the relevant video using the time parameter provided with the query
  - The client verifies the video hash against the committed hash in the blockchain
  - If the hashes match, an API call is made to the video analytics server with the video and decoded query parameters
  - The video is analyzed, and the result is returned in the form of a table as per query options
- Consensus:
  - Combine all received tables following consensus rules and policies

Time Travel Consensus

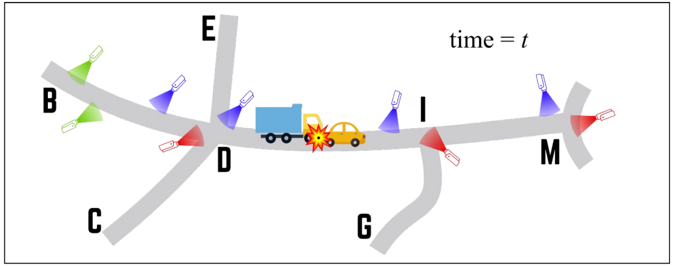
PERQL query execution reaches consensus when multiple cameras are available at a location, with the majority operating honestly

Majority consensus will not work situations with no cameras, insufficient cameras for consensus, or malicious camera owners

Time-travel consensus provides a solution to these limitations

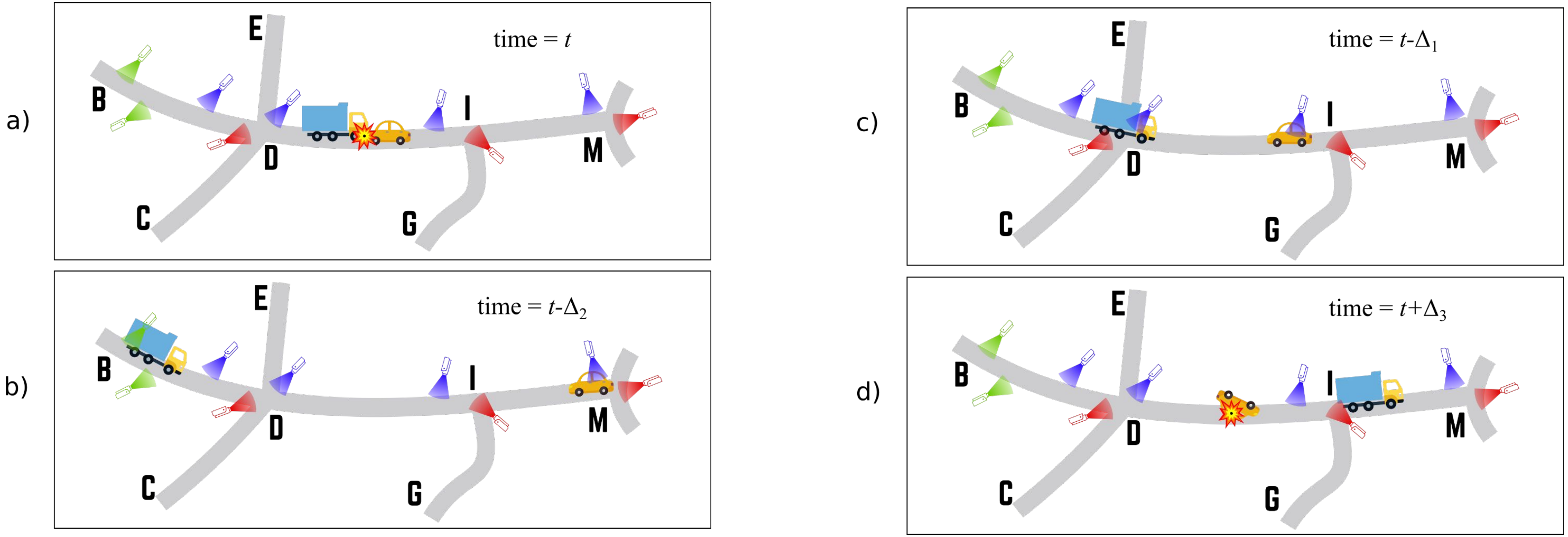
Time-travel consensus allows tracing events related to the event of interest across different locations and times

a)



Hit-and-Run Example

- An accident in a camera blind spot can be traced back in time to locations where related vehicles were captured
- It also allows traveling forward in time to identify vehicles involved in a hit-and-run accident



Case Study 1 - Majority consensus

Find a blue truck passed through junction one at around 11:30 AM

```
SELECT timeAppeared, vehicleType, vehicleColor
FROM vehicleDetector
AT 42.525678, -90.723601
WHERE vehicleType="truck", vehicleColor="#0000FF"
TIME 2023-01-25 11:30:00.005 TO 2023-01-25 11:35:00.005
```

Time sync screenshots of 5 cameras when a blue SUV truck crossing the junction at 11:30:05.50.

Results

timeAppeared	vehicleType	vehicleColor
Camera 1 result table		
11:30:04.90 11:30:07.10	truck	#012AB4
11:30:36.30 11:30:38.70	truck	#1336B8
11:32:08.40 11:32:09.90	truck	#15247F
11:33:12.60 11:33:14.50	truck	#3655AA
Camera 2 result table		
11:30:04.30 11:30:06.70	truck	#103BB8
11:30:35.90 11:30:37.70	truck	#1538B9
11:32:07.90 11:32:08.60	truck	#20247C
11:33:12.20 11:33:14.30	truck	#26456A
Camera 3 result table		
11:30:05.10 11:30:17.30	truck	#002CAD
11:30:36.50 11:30:48.70	truck	#1A35B4
11:32:08.20 11:32:19.30	truck	#25249B
11:33:12.90 11:33:23.20	truck	#36559A
Camera 4 result table		
11:30:01.80 11:30:06.80	truck	#003AB3
11:30:31.40 11:30:38.40	truck	#1254A4
11:33:09.20 11:33:14.10	truck	#36557A
Camera 5 result table		
11:30:04.80 11:30:09.20	truck	#001261
11:33:12.30 11:33:16.60	truck	#26356A
Final result table after consensus		
11:30:05.10 11:30:06.70	truck	#032CA2
11:30:36.50 11:30:37.70	truck	#1234B4
11:32:07.90 11:32:08.60	truck	#15245F
11:33:12.90 11:33:14.10	truck	#36558A

Case Study 2 - Time-travel consensus

The execution process is similar to Case Study 1, except that junction eight lacks overlapping camera views. Only two of the six nearby cameras capture the junction's view. Therefore, we utilize time-travel consensus to merge the outcomes from the six cameras.

```
SELECT timeAppeared, vehicleType
FROM vehicleDetector
AT 42.525678, -90.723601
WHERE vehicleType="cycle"
TIME 2023-01-25 10:00:00.000 TO 2023-01-25 10:05:00.000
```

Results

timeAppeared	vehicleType
10:00:14.20 10:01:13.10	cycle
10:01:02.30 10:01:45.40	cycle
10:01:04.50 10:01:42.30	cycle

Summary

- PERQS utilizes blockchain technology to ensure video feed integrity
- Enables multiple organizations to collectively execute CCTV video analysis
- Ensure data privacy and security while allowing efficient querying
- Limitations of PERQS:
  - Inability to perform effective joint video comparisons. Joint video analysis is crucial for tracking objects, identifying patterns, and conducting synchronized monitoring
  - Current architecture does not support direct video stream sharing for concurrent video analysis
- Future Possibilities:
  - Introduction of a secure hardware-based solution for multi-video joint analysis. Participants can send private encrypted videos to a secure execution environment for joint analysis
  - Enhancement of PERQS experience: Addition of a interactive UI for easy querying
  - Integration of a natural language interpreter to convert questions into PERQL query


THANK YOU

Publication

*“**A CONTRIBUTORY PUBLIC-EVENT RECORDING AND QUERYING SYSTEM**,” ARUN JOSEPH*, NIKITA YADAV, VINOD GANAPATHY, AND DUSHYANT BEHL
Proceedings of *SEC'23*, the **8th ACM/IEEE Symposium on Edge Computing**, Wilmington, Delaware, USA, December 2023.

Additional Slides

Implementation of Time-Travel Consensus

- Used when the *camera-finder* oracle returns a limited or zero count of cameras at a specified GPS location
- The search radius "R" and time offset "Δ" are specified as configuration parameters, with values determined by geographical context
- "R" is expanded iteratively if enough cameras are not found
- Dispatches queries to initiate video analytics within the estimated time interval
- Results from all cameras are received, and related events are identified to substantiate consensus, multiple supporting cameras are required for a final consensus
- The effectiveness of time-travel consensus relies on machine learning and vision models; they are treated as black boxes and can be updated as technology advances

Limitations of Time-Travel Consensus

- Many events cannot be supported with corroborative activities observed by nearby cameras
- Irrelevant nearby cameras may participate in the consensus, which can complicate the process.
- The search for cameras is confined to a defined radius "R" from the event-of-interest, and not all cameras within this range may have supporting events.
- Compatibility issues can arise, as not all video analytic models are suitable for time-travel consensus.
- Example: In the case of the example car accident, if only an accident detection model is available, and none of the cameras captured the accident, this model will not contribute to reaching a consensus. Estimations like speed and direction of vehicles may be possible with other ML/vision models.

Spatiotemporal Context

Acknowledges that real-world events are often interconnected across time and space

A single event may be causally related to activities at other locations and times

Time-travel consensus aims to support and validate the initial event occurrence

Conducts a meticulous search of events in the spatiotemporal vicinity when camera coverage is insufficient

Examines nearby cameras at different time frames to piece together causally related event patterns

PERQS Prototype

Our PERQS prototype was built on Hyperledger Fabric v2.2, a well-established permissioned blockchain framework.

Deployment was carried out on a Kubernetes cluster, utilizing Kubernetes for container orchestration.

Video analytics servers within the prototype were developed using the Python Django framework. The implementation used Django v4.1.4, Python v3.10.6, and TensorFlow v2.8.0.

These servers are responsible for conducting video analysis, a crucial component of the PERQS system.

Smart contracts were utilized to implement essential functionalities, including video hash commitment and query executions.

Consensus Study Summary

J	Cameras IDs	Find Query	#Out	Consensus Type
1	1,2,3,4,5	Blue Truck	4	Simple Majority
2	6,7,8,9	Green Car	1	Simple Majority
3	36,38,39,40	White Car	2	Simple Majority
4	34,35,36,37	White SUV	1	Time-travel
5	29,30,31,32	Blue Car	2	Time-travel
6	22,23,24,25,26	White Truck	1	Time-travel
7	18,19,20,21	Silver Car	1	Time-travel
8	10,11,12,13,14,15	Cycle	3	Time-travel

Evaluation

We have used the AICITY21 benchmark (CityFlowV2), captured by 46 cameras in a real-world traffic surveillance environment. A total of 880 vehicles are annotated in 6 different scenarios. There are 215.03 minutes of videos in total.

Within the PERQS video analyzer, a dedicated model named *vehicleDetector* was defined. This model leveraged YOLOv3 for real-time object detection, examining each frame to identify vehicle types (car, truck, bike, or SUV).

The machine learning model's output structure encompassed critical information, including *vehicleID, timeAppeared, vehicleType, vehicleColor, vehicleModel,* and *featureVector*.

References