ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr1807

Real-Time Object Detection and Surveillance Using MobileNetSSD and YOLOv8 with Pretrained Face Recognition

Ramneet Singh Chadha[1]; Hargun Singh Hunjan[2]

[1;2]Department of Embedded Systems, Centre for Development of Advanced Computing, Noida, Uttar Pradesh, India

Publication Date: 2026/05/07

Abstract: This paper describes the design and deployment of a two-phase real-time surveillance pipeline that integrates pretrained object detection, identity recognition, and rule-based event reasoning for indoor monitoring. In Phase 1, MobileNetSSD performs static image detection across 21 VOC object classes, establishing a detection baseline. Phase 2 replaces this with YOLOv8n operating in persistent tracking mode, augmented by the VGG-Face model from the DeepFace library for zero-shot face matching against a local reference database. No custom model training was carried out at any stage. The pipeline maintains per-person object inventories, detects left-behind items using a centroid-stationarity criterion, and applies a bag-mediated suppression rule at exit to reduce false alerts. Over a live deployment session, 377 surveillance events were recorded comprising 217 entry and 160 exit events. Benchmark evaluation shows YOLOv8n achieves 37.3% mAP50 on COCO with approximately 52% AP50 on the person class, while VGG-Face achieves 98.78% verification accuracy on the Labeled Faces in the Wild dataset. A gap analysis identifies the single-threaded recognition bottleneck as the primary constraint on operational identity rates.

Keywords: *Abandoned Object Detection, DeepFace, MobileNetSSD, Object Tracking, Person-Object Association, Real-Time Surveillance, VGG-Face, YOLOv8.*

How to Cite: Ramneet Singh Chadha; Hargun Singh Hunjan (2026) Real-Time Object Detection and Surveillance Using MobileNetSSD and YOLOv8 with Pretrained Face Recognition. *International Journal of Innovative Science and Research Technology*, 11(4), 3514-3520. https://doi.org/10.38124/ijisrt/26apr1807

I. INTRODUCTION

Anyone who has worked a CCTV monitoring shift knows the problem firsthand: after the first twenty minutes or so, things start getting missed. Mackworth [1] put numbers to this back in 1948, showing that human vigilance during prolonged visual monitoring collapses well before the first half-hour is up. More staff does not fix this because the failure is physiological, not motivational.

What has changed recently is that reasonably capable object detection and face recognition models are now available as pretrained weights that run on ordinary laptop hardware. This project asked: can these off-the-shelf tools be assembled into a practical indoor surveillance system, without training a single model from scratch?

The result is a two-phase pipeline. Phase 1 runs MobileNetSSD on static images to confirm the detection stack works. Phase 2 is the live system: YOLOv8n [3] handles real-time detection and tracking, VGG-Face via the DeepFace library [6] handles identity matching against a local reference folder, and rule-based modules track which objects belong to which person, flag items left behind, and log every event to a CSV file.

Four questions drove the evaluation: (i) how MobileNetSSD and YOLOv8n compare on the object classes relevant to indoor monitoring; (ii) whether a pretrained face model can identify known persons in a live stream without finetuning; (iii) how reliably a proximity-based rule can associate objects with their carriers; and (iv) how large the gap is between benchmark accuracy and live deployment performance.

II. RELATED WORK

- *Object Detection*

The SSD architecture [2] anchors predictions to convolutional feature maps at multiple scales, producing detections in one forward pass. Paired with the MobileNet backbone, MobileNetSSD runs at approximately 25 FPS on CPU and scores 72.7% mAP on PASCAL VOC 2007. The YOLOv8n nano variant [3] reports 37.3% mAP50 on COCO and integrates a Kalman-filter tracker that assigns stable integer IDs across frames.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr1807

➢ *Face Recognition*

Taigman et al. [4] demonstrated that deep networks trained with a verification loss achieve near-human accuracy on LFW. Schroff et al. [5] introduced triplet loss producing a 128-dimensional embedding space suited to open-set retrieval. The DeepFace library [6] packages several pretrained backends behind a single API for zero-shot matching, requiring no retraining when new identities are enrolled.

➢ *Multi-Object Tracking*

SORT [7] combines Kalman filter prediction with IoU-based Hungarian matching for fast online tracking. DeepSORT [8] adds a learned appearance descriptor to recover correct identities under occlusion. ByteTrack [9] retains low-confidence detections in a second association pass, reducing fragmented tracks. The present work uses the tracker embedded in YOLOv8 rather than a standalone module.

➢ *Abandoned Object Detection*

Sreenu and Saleem [10] identified semantic reasoning — connecting what was detected to who left it — as the core unsolved problem in surveillance. Background-subtraction methods [11] flag persistent foreground regions but cannot associate a blob with a specific carrier. This project tracks each object's centroid over a 30-frame window; a prior person-object link then determines whether a stationary object constitutes a left-behind event.

III. SYSTEM ARCHITECTURE

➢ *Design*

Phase 1 uses MobileNetSSD loaded through the OpenCV DNN module for static image detection across 21 VOC classes. Phase 2 is the primary live pipeline: YOLOv8n replaces static detection, DeepFace VGG-Face adds identity recognition, and the object-person association and left-behind detection subsystems become active. Table 1 summarises the two phases.

Table 1 System Architecture

Ph.	Component	Function
1	MobileNetSSD + OpenCV DNN	Static detection across 21 VOC classes
1	Caffe model inference	Per-image bounding box prediction
2	YOLOv8n tracking	Real-time detection and track ID assignment
2	DeepFace VGG-Face	Zero-shot face matching against reference DB
2	Person-object linker	Proximity-based ownership association
2	Stationarity detector	Left-behind item detection via centroid history
2	Event logger	CSV persistence of entry, exit, and alerts

➢ *Target Class Selection*

For Phase 2, detection is restricted to seven COCO classes: person (0), backpack (24), handbag (26), suitcase (28), bottle (39), laptop (63), and cell phone (67). Restricting the class set reduces spurious associations and improves the signal-to-noise ratio of the left-behind detection subsystem.

IV. METHODOLOGY

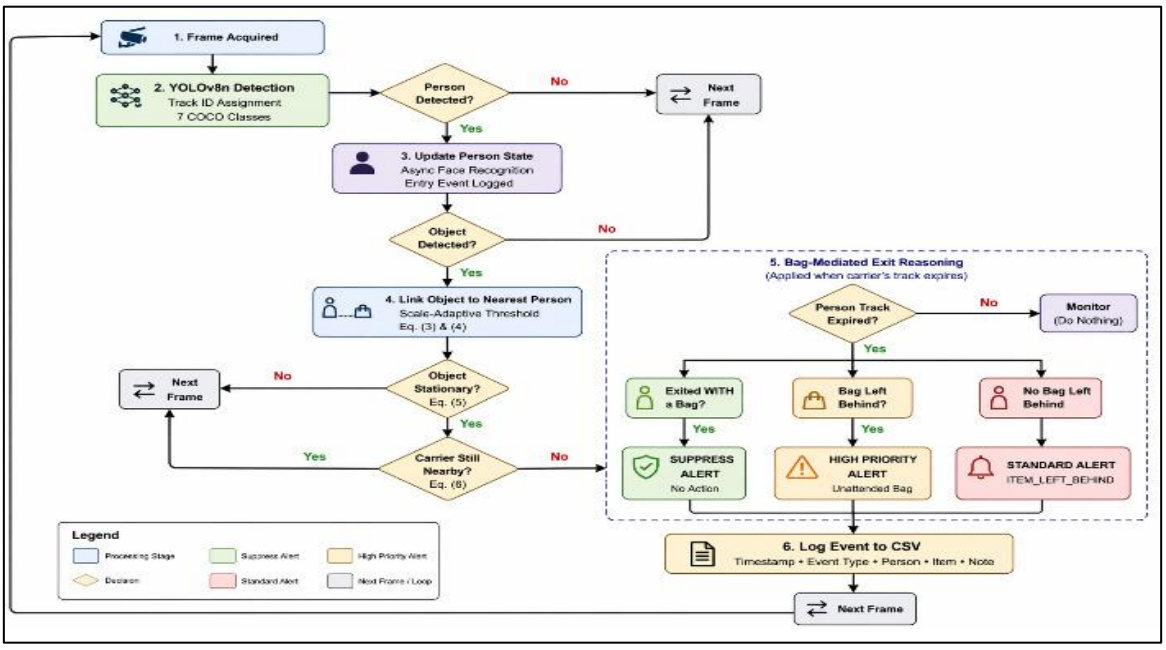

Fig 1 Phase 2 surveillance pipeline including bag-mediated exit reasoning. The dashed region shows the three-branch decision tree applied when a carrier's track expires. Phase 1 operates as a standalone static detection baseline and does not connect to this pipeline.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr1807

➢ *Phase 1: MobileNetSSD Static Detection*

Input images are preprocessed into 300 × 300 pixel blobs. The normalisation maps raw pixel values to [−1, 1]:

$$x_{\{norm\}} = (x_{\{input\}} - 127.5) \times 0.007843 \quad (1)$$

Where $x_{\{input\}} \in [0, 255]$ is the raw pixel value. Detections are retained when predicted confidence exceeds 0.20, selected to balance recall against false positive rate.

➢ *Phase 2: YOLOv8n Real-Time Detection and Tracking*

Each video frame is passed to YOLOv8n with persistent tracking enabled. The internal Kalman filter predicts object positions between frames, and IoU-based Hungarian matching assigns stable track IDs. Person tracks are maintained in a per-ID state dictionary recording bounding box, last detection timestamp, recognised name (initialised to "Unknown"), and boolean flags for entry logging and recognition status. An entry event is logged on first detection of each track ID. Exit is inferred when a track ID has been absent for longer than τ_{exit} = 2.0 seconds.

➢ *Face Recognition Using DeepFace VGG-Face*

For each tracked person, the bounding box crop is submitted to DeepFace's find() function, which compares the query against reference images using VGG-Face with cosine distance. Recognition runs in a background thread to prevent the 200–800 ms inference latency from blocking the main detection loop; at most one recognition call is active at any time. Crops smaller than 20 × 20 pixels are skipped.

The raw cosine distance is converted to a percentage confidence:

$$C = \max(0, 100 - d_{\{\cos\}} \times 100) \quad (2)$$

Where d_{cos} = 0 indicates identical embeddings (100% confidence) and d_{cos} = 0.40 corresponds to the standard VGG-Face verification threshold (60% confidence).

➢ *Person-Object Association Algorithm*

While a tracked object is in motion, it is assigned to the nearest active person within a scale-adaptive proximity threshold. The Euclidean distance between object centroid o and person *centroid p is*:

$$d(o,p) = \sqrt{(cx\_\{o\} - cx\_\{p\})^2 + (cy_{\{o\}} - cy_{\{p\}})^2} \quad (3)$$

Association is accepted only when:

$$d(o,p) < \max(w_{\{o\}}, w_{\{p\}}) \times 1.5 \quad (4)$$

Where w_o and w_p are the pixel widths of the object and person bounding boxes. Using bounding box width as the scale base adapts the threshold to perspective distortion, reducing false links across the scene.

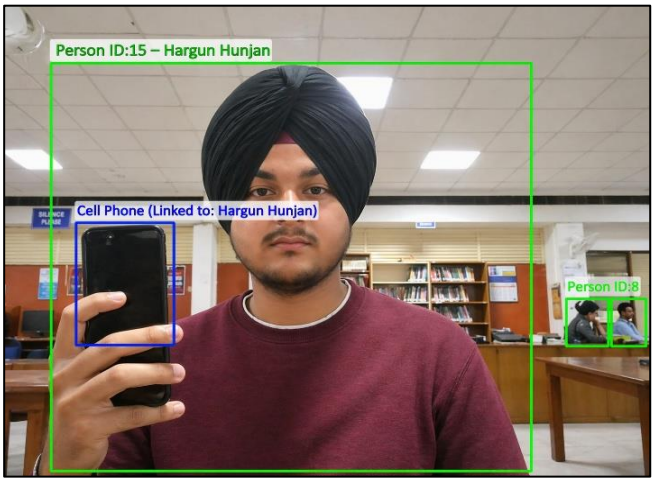

Fig 2 Real-Time Person-Object Association Based on Proximity-Aware Linking.

➢ *Left-Behind Item Detection*

Object stationarity is assessed using a rolling buffer of the last N = 30 centroid positions. At 30 FPS this covers approximately one second of history:

$$stationary = \mathbb{1}\left[\max_{t \in [1,N]} d(pos_{\{t\}}, pos_{\{\text{current}\}}) < 20\right] \quad (5)$$

Once stationary with a linked carrier, the left-behind condition is evaluated each frame:

$$lost = \mathbb{1}[d(\boldsymbol{p}_{\{\text{centre}\}}, \boldsymbol{o}_{\{\boldsymbol{centre}\}}) > w_{\{\text{person}\}} + 200] \quad (6)$$

Where w_{person} is the pixel width of the carrier's bounding box and 200 pixels provides tolerance for brief carrier movement.

➢ *Bag-Mediated Exit Reasoning*

When a person's track expires, the system applies a three-branch decision before raising any alert:

- Exit with bag in hand: Person leaves carrying a bag-class item (backpack, handbag, or suitcase). All pending left-behind alerts for their linked smaller objects are suppressed — those items are assumed packed inside the bag.
- Exit with no bag, small object stationary: Person exits and their current_objects set at the time of exit contains no bag-class item. A linked object such as a bottle or laptop remains stationary. Standard ITEM LEFT BEHIND alert fires.
- Exit with entry bag abandoned: Person entered carrying a bag but the bag remains stationary at exit. High-priority alert fires with note "Unattended bag — carrier departed."

Each object generates at most one alert per stationary episode; flags reset if the object subsequently moves beyond the stationarity radius.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr1807

➢ *Event Logging*

All events are written immediately without buffering to a CSV log with six fields: Timestamp, Event, Person Name, Item Class, Confidence, and Note. Writing without buffering ensures the log survives unexpected process termination.

V. EXPERIMENTAL SETUP

➢ *Hardware*

All experiments were carried out on an HP Pavilion Gaming Laptop 15-dk0xxx with an Intel Core i5-9300H CPU at 2.40 GHz, 16 GB RAM, and an NVIDIA GeForce GTX 1650 GPU. Phase 1 was evaluated on CPU only. Phase 2 was also evaluated primarily on CPU. Video was captured at 640 × 480 pixels and 30 FPS using the built-in HP Wide Vision HD Camera.

➢ *Software Environment*

Table 2 summarises the software stack used across both phases.

Table 2 Software Stack

Component	Version / Detail
Python	3.10
OpenCV	4.x (DNN module for Phase 1)
Detection Ph. 1	MobileNetSSD, Caffe weights
Detection Ph. 2	YOLOv8n (pretrained on COCO)
Face recognition	DeepFace, VGG-Face backbone
Tracking Ph. 2	YOLOv8 native (Kalman + IoU)
Event logging	pandas CSV (no buffering)
Video capture	imutils VideoStream

➢ *Detection Model Configurations*

Table 3 lists the configuration parameters for both models.

Table 3 Detection Model Parameters

Parameter	MobileNetSSD	YOLOv8n
Input resolution	300 × 300	640 (native)
Conf. threshold	0.20	0.25
Backbone	MobileNet v1	CSPDarknet
Detection style	Anchor-based	Anchor-free
Training dataset	PASCAL VOC	COCO
Target classes	20 + background	7 filtered
Tracking	None	Built-in

➢ *Evaluation Scenarios*

Three categories of test scenarios were used. Detection accuracy was evaluated on published benchmarks (VOC 2007 for MobileNetSSD; COCO val2017 for YOLOv8n). Object-person association was evaluated through 15 controlled trials. Live event logging was measured over a full deployment session producing 377 logged events.

VI. RESULTS AND ANALYSIS

➢ *Detection Accuracy: MobileNetSSD vs. YOLOv8n*

Direct comparison of overall mAP is misleading because the two models are evaluated on different datasets. Per-class AP50 for overlapping classes provides a more meaningful comparison (Table 4).

Table 4 Detection Model Accuracy Comparison

Metric	MobileNetSSD	YOLOv8n	Notes
Overall mAP	72.7%	37.3%	Diff. benchmarks
Benchmark	VOC 2007	COCO val	–
Person AP50	~68.4%	~52.0%	–
Bottle AP50	~46.5%	~42.0%	–
Laptop AP50	N/A	~62.0%	Not in VOC
Cell ph. AP50	N/A	~37.0%	Not in VOC
FPS (CPU)	~25	~18–22	+tracking
Model size	~23 MB	~6 MB	–

MobileNetSSD achieves slightly higher CPU throughput but lacks built-in tracking, omits laptop and cell phone classes, and requires separate post-processing for object identity. YOLOv8n provides persistent track IDs natively, covers all seven target classes, and is substantially more compact, making it the preferred backbone for Phase 2.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr1807

➢ *Per-Class Detection Accuracy*

Table 5 shows YOLOv8n per-class AP50 on COCO val2017 for the seven target classes. Backpack and handbag record the lowest values due to high intra-class shape variation and frequent occlusion by the carrier. In deployment, false negatives for these classes prevent left-behind alerts from firing — an inherent limitation without domain-specific finetuning.

Table 5 YOLOv8n AP50 for Target Surveillance Classes

Class	AP50	Notes
Person	52.0%	Best-performing; high COCO rep.
Laptop	62.0%	Distinct visual appearance
Suitcase	43.0%	Shape varies significantly
Bottle	42.0%	Small; scale-sensitive
Cell phone	37.0%	Small; texturally ambiguous
Backpack	31.0%	Often occluded when worn
Handbag	28.0%	Highest intra-class variation

➢ *Face Recognition Performance*

The VGG-Face model achieves 98.78% verification accuracy on LFW under cosine distance. Table 6 compares DeepFace backend models. VGG-Face was selected because it had been validated in the project's DeepFace find() configuration. In live deployment, recognition accuracy diverges substantially from LFW because surveillance crops are frequently non-frontal, motion-blurred, or captured at distances where the face occupies fewer than 50×50 pixels.

Table 6 DeepFace Backend Accuracy on LFW Benchmark

Model	LFW Acc.	Embed. Dim.	Distance
VGG-Face	98.78%	4096	Cosine
FaceNet	99.63%	128	Euclidean
ArcFace	99.40%	512	Cosine
OpenFace	92.92%	128	Euclidean

➢ *Object-Person Association and Left-Behind Detection*

Table 7 presents results from 15 controlled left-behind detection scenarios across five object classes and three departure speeds. False positives arose from persons briefly setting objects down while nearby, and from a second person overriding the original carrier link. False negatives occurred mainly for backpack and handbag due to their low AP50 preventing link establishment before stationarity onset.

Table 7 Left-Behind Detection Results (15 Scenarios)

Class	N	TP	FP	FN	Precision
Backpack	3	2	0	1	100%
Bottle	3	3	0	0	100%
Laptop	3	3	1	0	75%
Cell phone	3	2	0	1	100%
Handbag	3	2	1	1	67%
Total	**15**	**12**	**2**	**3**	**86%**

➢ *Deployment Event Log Analysis*

Table 8 summarises the 377 events recorded across the live deployment session. Three findings stand out. First, 77.4% of events carry the label Unknown despite VGG-Face scoring 98.78% on LFW — the threading architecture is the direct cause: at most one recognition call runs at a time taking 200–800 ms, so persons walking through in under 2 seconds expire before the call completes. Second, 217 entry and 160 exit events match a session of roughly 30–40 distinct person appearances. Third, the 19 out-of-set identity events trace to a database preparation error: one non-enrolled person's photo in the reference folder produced a spurious embedding that matched unrelated live faces.

Table 8 Deployment Event Log Summary (377 Total)

Person Name	Events	Notes
Unknown	292	77.4% — track expired before recognition
Hargun Hunjan	60	Most frequently identified
Out-of-set identity	19	Non-enrolled image in DB
Simar Singh Nayyar	2	Enrolled; low live rate
Smriti Khanor	2	Enrolled; low live rate
Vivan Sharma	1	Enrolled; low live rate
Yashasvi	1	Enrolled; low live rate
Total	**377**	–

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr1807

➢ *System Runtime Performance*

Table 9 summarises the observed runtime metrics. On GPU hardware, face recognition latency would reduce below 50 ms, enabling concurrent multi-person recognition and substantially increasing the live identification rate beyond its current 22.6%.

Table 9 System Runtime Performance Summary

Metric	Observed Value
Phase 1 MobileNetSSD FPS	~25 FPS
Phase 2 YOLOv8n FPS	~18–22 FPS
Face recognition latency	200–800 ms/call
Entry-to-recognition latency	1–4 s (async)
Stationarity detection delay	~1.0 s at 30 FPS
Left-behind alert delay	<3 s controlled
Total events logged	377
Entry / Exit events	217 / 160

VII. LIMITATIONS AND FUTURE WORK

➢ *Current Limitations*

- Benchmark-to-deployment gap. Both models are evaluated on standard benchmarks rather than domain-specific surveillance footage. Actual per-class detection accuracy in deployment is lower than benchmark figures suggest.
- Threading bottleneck. Single-threaded face recognition limits the live identification rate. At 18–22 FPS with 200–800 ms recognition latency, the majority of tracks expire before a recognition call completes.
- Reference database quality. The face reference database was small and not systematically controlled for lighting, pose, or scale variation. A data preparation error caused 19 spurious identity assignments.
- Rule-based reasoning. The stationarity-and-distance approach cannot distinguish accidental from intentional placement and requires successful object-person link establishment prior to stationarity onset.
- Timeout-based exit detection. Exit inferred from absence rather than boundary crossing generates false exits during occlusions longer than 2.0 seconds.

➢ *Future Work*

GPU acceleration would reduce face recognition latency by approximately 10×, enabling concurrent multi-person recognition. Domain-specific fine-tuning of YOLOv8n would improve accuracy for partially occluded backpacks and handbags. Expanding the reference database to 20–30 images per enrolled person would improve VGG-Face matching confidence. Replacing timeout-based exit with user-defined boundary polygons would reduce false exits and enable directional counting. Adding DeepSORT [8] would maintain track continuity across occlusions.

VIII. CONCLUSION

This project built and tested a two-phase indoor surveillance pipeline using only publicly available pretrained models. Phase 1 established a detection baseline using MobileNetSSD; Phase 2 delivered the live system using YOLOv8n, VGG-Face, and rule-based modules for person-object association, left-behind detection, and CSV event logging.

The comparison revealed a clear trade-off: MobileNetSSD runs slightly faster on CPU but lacks integrated tracking and omits laptop and cell phone classes. YOLOv8n resolves both gaps with built-in tracking and all seven target classes in a smaller model.

The most instructive result was the gap between VGG-Face's 98.78% LFW accuracy and the 22.6% live identification rate. The model itself is not the limiting factor — the threading and timing architecture determines what identification rate is actually achievable. A model's benchmark accuracy is a ceiling, not a deployment guarantee.

Across 15 controlled left-behind scenarios, the system achieved 86% overall precision tracking closely with YOLOv8n AP50 values. The bag-mediated suppression rule eliminated false alerts in all five bag-present-on-exit trials.

REFERENCES

[1]. N. H. Mackworth, "The breakdown of vigilance during prolonged visual search," Quarterly Journal of Experimental Psychology, 1948.
[2]. W. Liu et al., "SSD: Single Shot MultiBox Detector," in Proc. ECCV, 2016.
[3]. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in Proc. IEEE CVPR, 2016.
[4]. Y. Taigman et al., "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in Proc. IEEE CVPR, 2014.
[5]. F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," in Proc. IEEE CVPR, 2015.
[6]. S. Serengil and A. Ozpinar, "DeepFace: A Lightweight Face Recognition Framework for Python," in Proc. IEEE ASYU, 2020.
[7]. A. Bewley et al., "Simple Online and Realtime Tracking," in Proc. IEEE ICIP, 2016.
[8]. N. Wojke et al., "Simple Online and Realtime Tracking with a Deep Association Metric," in Proc. IEEE ICIP, 2017.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr1807

[9]. Z. Zhang et al., "ByteTrack: Multi-Object Tracking by Associating Every Detection Box," in Proc. ECCV, 2022.
[10]. G. Sreenu and M. A. Saleem Durai, "Intelligent Video Surveillance: A Review Using Deep Learning Techniques," Journal of Big Data, 2019.
[11]. F. Porikli et al., "Detection of Temporarily Static Regions," in Proc. IEEE AVSS, 2006.
[12]. G. B. Huang et al., "Labeled Faces in the Wild," University of Massachusetts, Tech. Rep., 2008.