Demo: Video-based Social Distancing Evaluation in the COSMOS Testbed Pilot Site

wim.net Wireless & Mobile Networking Lab

Columbia University in the City of New York

Mahshid Ghasemi[1], Zhengye Yang[2], Mingfei Sun[1], Hongzhe Ye[1], Zihao Xiong[1], Javad Ghaderi[1], Zoran Kostic[1], Gil Zussman[1]

[1]Electrical Engineering, Columbia University, [2]Computer and Systems Engineering, RPI

NEAL Network Algorithms

COSMOS

Abstract

Social distancing can reduce the spread of coronavirus. Hence, we designed two approaches for social distancing analysis, **Auto**mated video-based **S**ocial **D**istancing **A**nalyzer (**Auto-SDA**), and **B**ird's eye view **S**ocial **D**istancing **A**nalyzer (**B-SDA**).

- Designed to measure pedestrians' compliance with social distancing policies using street-level and bird's eye view cameras.
- Auto-SDA offers high accuracy which is not sensitive to the dynamics of the scene and the camera's tilt angle.
- B-SDA provides comparable accuracy while preserving pedestrian's privacy.
- Evaluated the pipelines using **COMOS testbed** deployed in West Harlem, NYC.
- Used real-world videos recorded from cameras deployed on the 2nd and 12th floor of **Columbia's Mudd building**, looking at the COSMOS site, before and during the pandemic.
- The results represent how the impact of social distancing policies on pedestrian's social behavior.

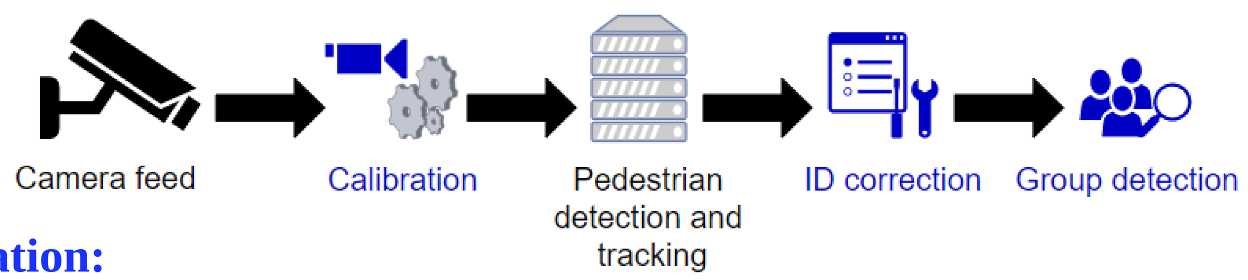
COSMOS Testbed Pilot Site

The NSF PAWR COSMOS site at 120th St. and Amsterdam Ave. intersection, NYC.

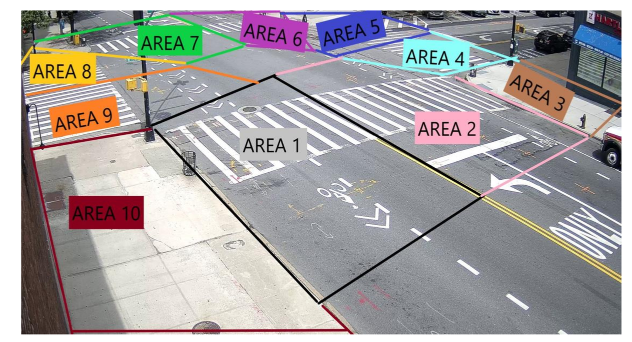
Viewpoint of the camera deployed on the 2nd floor of the Columbia's Mudd building.

Viewpoint of the camera deployed on the 12th floor of the Columbia's Mudd building.

Automated Social Distancing Analyzer (Auto-SDA)

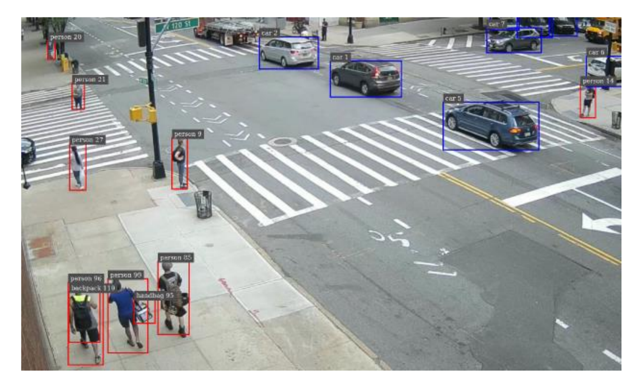


- **Calibration:**
  - Converts 2D on-image distances to 3D on-ground distances.
  - Standard camera calibration methods lead to **inaccurate on-ground distance** computation.
  - Auto-SDA's calibration module **splits the view** of the camera into **multiple areas**.
  - Photogrammetry parameters are calculated for each area individually.
  - Auto-SDA's calibration module can obtain on-ground distances with **less than 10 cm error**.



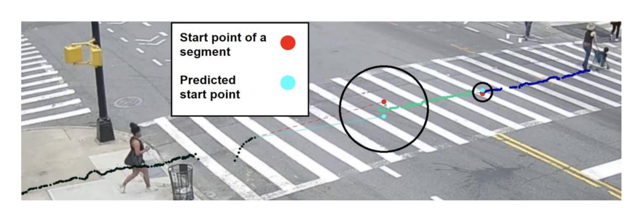
The view of the camera is split to 10 areas. The extrinsic parameters are calculated for each area individually.

- **Object detection and tracking:**
  - Uses **Nvidia DeepStream SDK** platform which is an **optimized architecture** built using the **Gstreamer** framework.
  - Uses object detector, **YOLOv4**, and **Nvidia DCF-based** tracker models.



Results of object detection (YOLOv4) and tracking (Nvidia DCF).

- **ID correction:**
  - **High altitude** and **oblique** view of the COSMOS cameras, and **obstacles** degrade the performance of the tracker.
  - Compensates for the inaccuracies of the tracker.
  - **Removes the redundant IDs** and extracts pedestrians' **entire trajectory**.



ID correction detects and removes redundant IDs when the tracker assigns 3 IDs to a single pedestrian.

- **Group detection:**
  - Excludes social groups from social distancing violation.
  - **Off-the-shelf group detectors require detailed information** such as body and head orientation, velocity, exact trajectory, etc.
  - In realistic deployments, such as COSMOS cameras, **details cannot be obtained**.
  - Detects the social groups **with limited data** from the cameras.

Group detection using instantaneous distance between pedestrians.

Comparison of Prior Work to Auto-SDA

Framework	Object Detection	Tracking	Calibration Method	On-Ground Distance Computation Error	Correction	Group Detection	Real-World COVID-19 Pandemic Impact Analysis
[1, 2]	✓	X	Homography transformation	$\gg$ 10 cm	X	X	X
[3]	✓	✓	Pairwise L_2 norm	$\gg$ 10 cm	X	X	X
[4]	✓	✓	Planar camera persp. trans.	$\gg$ 10 cm	X	X	X
[4, 5, 6, 7]	✓	X	Planar camera persp. trans.	$\gg$ 10 cm	X	X	X
Auto-SDA	✓	✓	Multi-area calibration	< 10 cm	✓	✓	✓

Auto-SDA Algorithms

Algorithm 1 ID Correction

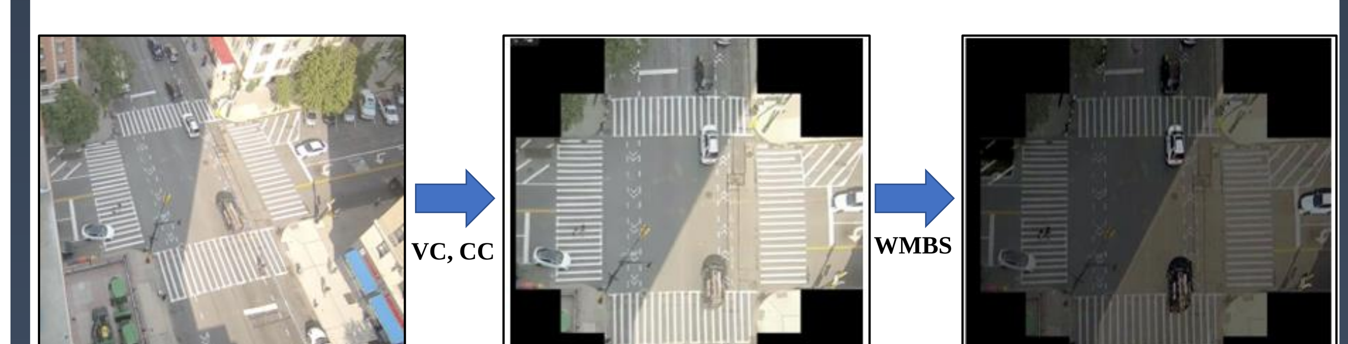
```
Input: ID_vec, e1, e2, n                      ▷ ID_vec is the output of NvDCF tracker
Output: corrected ID_vec
for id ∈ ID_vec do
    Compute id.Trj                             ▷ vector of points on id's path
    Compute id.TimeStamp.StartTime             ▷ detection time
    Compute id.TimeStamp.StopTime              ▷ Lost time
    Compute (id.TailEst, id.TailDir)           ▷ Linear Regression of id.Trj.tail(n)
    Compute (id.HeadEst, id.HeadDir)           ▷ Linear Regression of id.Trj.head(n)
for (id1, id2) ∈ ID_vec do
    t1 ← id1.TimeStamp.StopTime
    t2 ← id2.TimeStamp.StartTime
    p1 ← id1.TailEst.at(t = t2),  p2 ← id1.Trj.at(t2)
    v1 ← id1.TailDir,  v2 ← id2.HeadDir
    if t2 − t1 < e1 && |p1 − p2| < e2 && ∠(v1, v2) < 90° then
        id1 and id2 belongs to same person
```

Algorithm 2 Group Detection

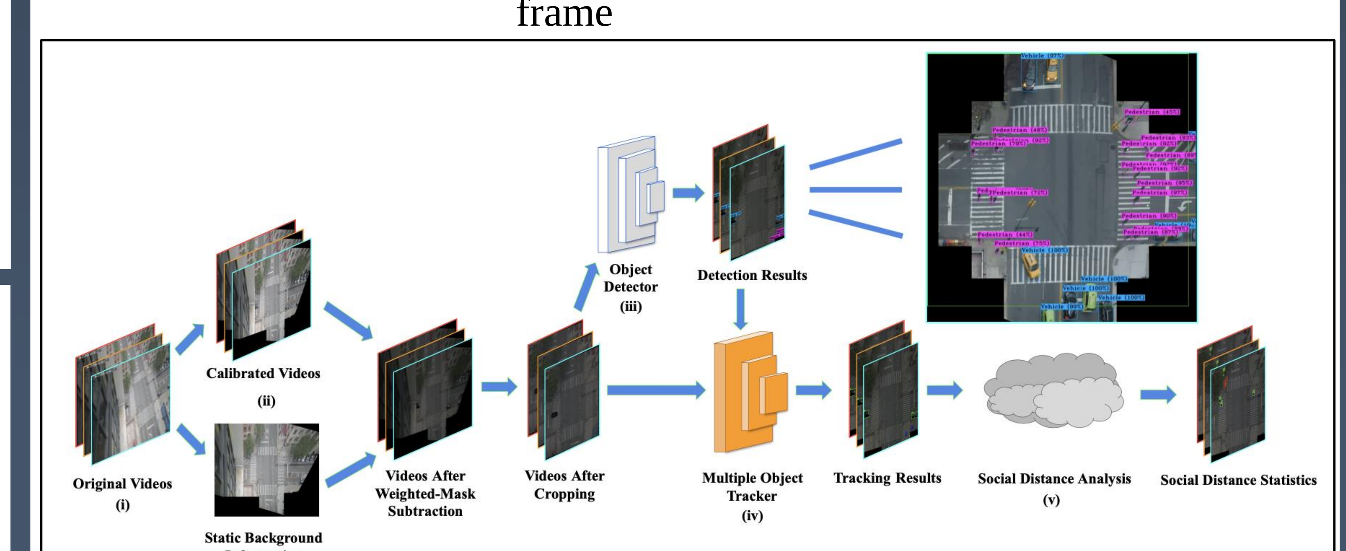
```
Input: ID_vec, d_max, d_max, σ_max
Output: ID_vec Pdestrians belong to a group
for id ∈ ID_vec do
    id.TimeTrj = map(id.TimeStepVec, id.Trj)
for (id1, id2) ∈ ID_vec do
    n = 0
    for t = 1 : T do
        pos1 = id1.TimeTrj(t),  pos2 = id2.TimeTrj(t)
        d = ||pos1 − pos2||2
        if d > d_max then
            n++, continue
        Corr_vec(id1, id2).append(d)
    if n > N_max then
        continue
    d̄ = mean(Corr_vec(id1, id2))     ▷ calculate the mean distance between two pedestrians
    σ = std(Corr_vec(id1, id2))       ▷ calculate the standard deviation of instantaneous distances between two pedestrians
    if d̄ < d̄_max && σ < σ_max then
        id1 and id2 belongs to the same group
```

Bird's Eye View Social Distancing Analyzer (B-SDA)

- **Auto-SDA** detects social distancing violations with **high accuracy**.
- Street-level cameras' challenges:
  - Surveillance area is limited.
  - Temporal tracking of pedestrians is challenging due to occlusions.
  - Street-level cameras may raise privacy concerns.
- **B-SDA modules**:
  - **Pre-processing**
    - Video Calibration (**VC**): Transforms original (slanted) view to perpendicular to the ground.
    - Center Cropping (**CC**): Crops the salient part of the image.
    - Weighted-Mask Background Subtraction (**WMBS**): Background image is subtracted from the original frames to sharpen the boundaries between pedestrians and the static background.
  - Pedestrian detection and tracking:
    - **Customized YOLOv4** (detector) to enhance the detection of small pedestrians.
    - SORT (tracker).
  - **Group detection**: Similar to the group detection module in Auto-SDA.



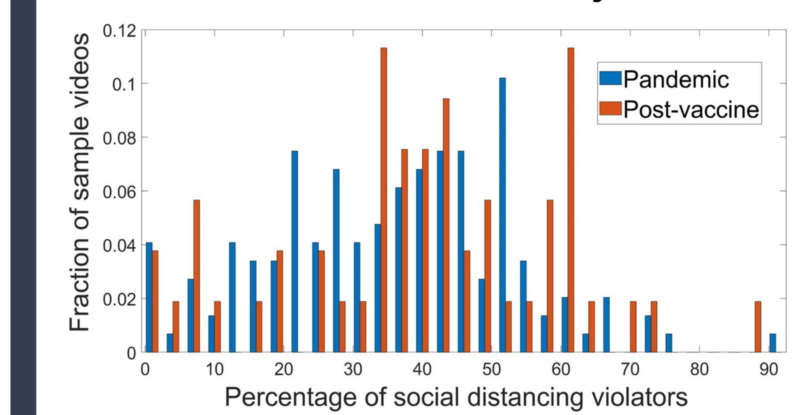
Original frame — Calibrated/cropped frame — Frame after WMBS



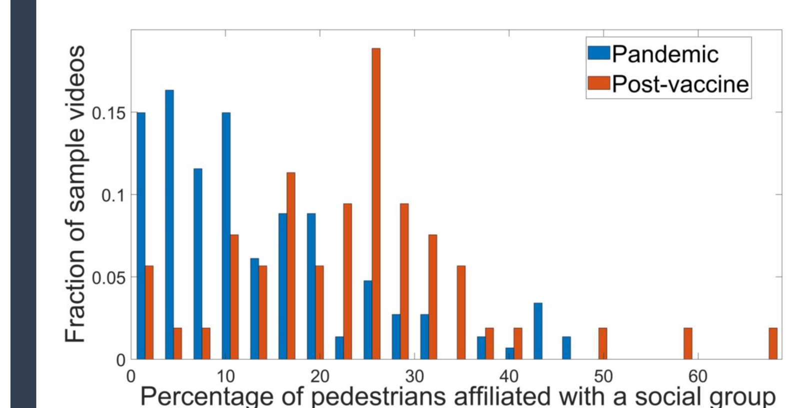
B-SDA pipeline: (i) collect raw videos from a bird's-eye view camera; (ii) do calibration and background subtraction to alleviate the effect of sub-optimal sensor quality; (iii) get pedestrian detection results; (iv) get pedestrian tracking information; (v) analyze pedestrian movement behavior with social distancing analysis algorithm.

Measurements and Evaluation

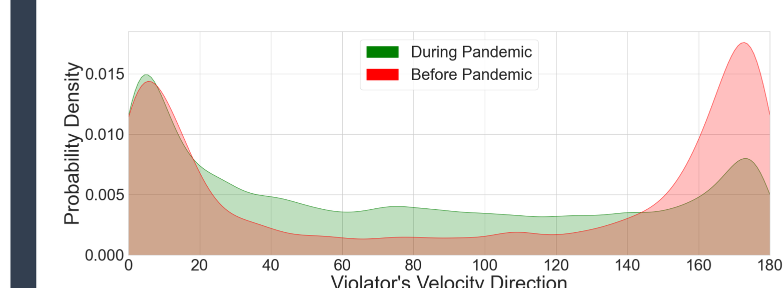
- Evaluated the **impact of COVID-19 on pedestrians behavior** using recorded videos from the COSMOS cameras.
- Recorded videos consist of:
  - **Pre-pandemic** videos recorded in **June 2019.**
  - **Pandemic** videos recorded between **June 17 to July 20, 2020 (soon after the lockdown).**
  - **Post-vaccine** videos recorded in **May 2021 (after the vaccine became broadly available).**



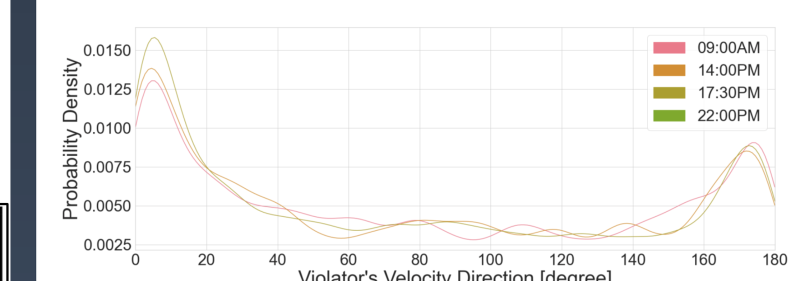
Auto-SDA: Normalized histogram of the percentage of pedestrians considered social distancing violators in the recorded videos.



Auto-SDA: Normalized histogram of the percentage of pedestrians affiliated with a social group.



B-SDA: Distribution of the angle between moving directions of two pedestrians in a violation pair.



B-SDA: Distributions of the angle between moving directions of two pedestrians in a violation pair at different times of day (estimated by KDE with Gaussian kernel).

References

1. D. Yang, E. Yurtsever, V. Renganathan, K. Redmill, and Ü. Özgüner. A vision-based social distancing and critical density detection system for COVID-19. Sensors (2021).
2. S. Saponara, A. Elhanashi, and A. Gagliardi. Implementing a real-time, AI-based, people detection and social distancing measuring system for Covid-19. JRIP (2021)
3. N. Singh Punn, S. Kumar Sonbhadra, and S. Agarwal. Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLOv3 and DeepSort techniques. arXiv preprint arXiv:2005.01385 (2020).
4. A. Koneru, P. Ragini, M. Sri Vastav, and K. Jashnavi. A real-time solution for social distance detection in COVID-19 pandemic. ICICCS (2021).
5. J. Betancourt. Social distancing analyser. https://github.com/JohnBetaCode/Social-Distancing-Analyser (2020).
6. D. Birla. Social Distancing AI. https://github.com/deepak112/Social-Distancing-AI (2020).
7. M. Cristani, Alessio Del Bue, Vittorio Murino, Francesco Setti, and Alessandro Vinciarelli. The visual social distancing problem. IEEE Access (2020).
8. T. Farrand. Social distancing. https://github.com/FarrandTom/social-distancing (2020).
9. M. Ghasemi, Z. Kostic, J. Ghaderi, and G. Zussman., Auto-SDA: Automated video-based social distancing analyzer. In Proc. ACM Hot-EdgeVideo (2021).
10. Z. Yang, M. Sun, H. Ye, Z. Xiong, G. Zussman, and Z. Kostic, "Bird's-eye view social distancing analysis system," In Proc. IEEE ICC Workshop Edge Learn. 5G Mobile Networks Beyond (2022) (to appear).
11. D. Raychaudhuri, I. Seskar, G. Zussman, T. Korakis, D. Kilper, T. Chen, J. Kolodziejski, M. Sherman, Z. Kostic, X. Gu, H. Krishnaswamy, S. Maheshwari, P. Skrimponis, and C. Gutterman. Challenge: COSMOS: A city-scale programmable testbed for experimentation with advanced wireless. In Proc. ACM MobiCom (2020).

This work was supported in part by NSF grants CNS-1827923, OAC-2029295, CNS-2038984, ARO grant W911NF1910379, and AT&T VURI award.