

The dangers of social distancing: How COVID-19 can reshape our social experience

Kevin Sikai<sup>1</sup>

► Author information ► Article notes ► Copyright and License information ► Disclaimer

## Social distancing decreases an individual's likelihood of contracting COVID-19

Russell H. Fazio   , Benjamin C. Ruisch  , Courtney A. Moore,   , and Jesse T. Ladanyi [Authors Info & Affiliations](#)

February 4, 2021 | 118 (8) e2023131118 | <https://doi.org/10.1073/pnas.2023131118>

## Insufficient social distancing may contribute to COVID-19 outbreak: The case of Ijuí city in Brazil

Thiago Gomes Heck   , Rafael Z. Frantz  , Matias Nunes Frizzo, Carlos Henrique Ramires François, Mirna Stela Ludwig, Marilia Arndt Mesenburg, Giovano Pereira Buratti, Lígia Beatriz Bento Franz, Evelise Moraes Berlezi

Published: February 17, 2021 • <https://doi.org/10.1371/journal.pone.0246520>

# BIRD'S EYE VIEW AI HOGGERS

## Failure to wear face masks and not social distancing prime reasons for COVID-19 spread

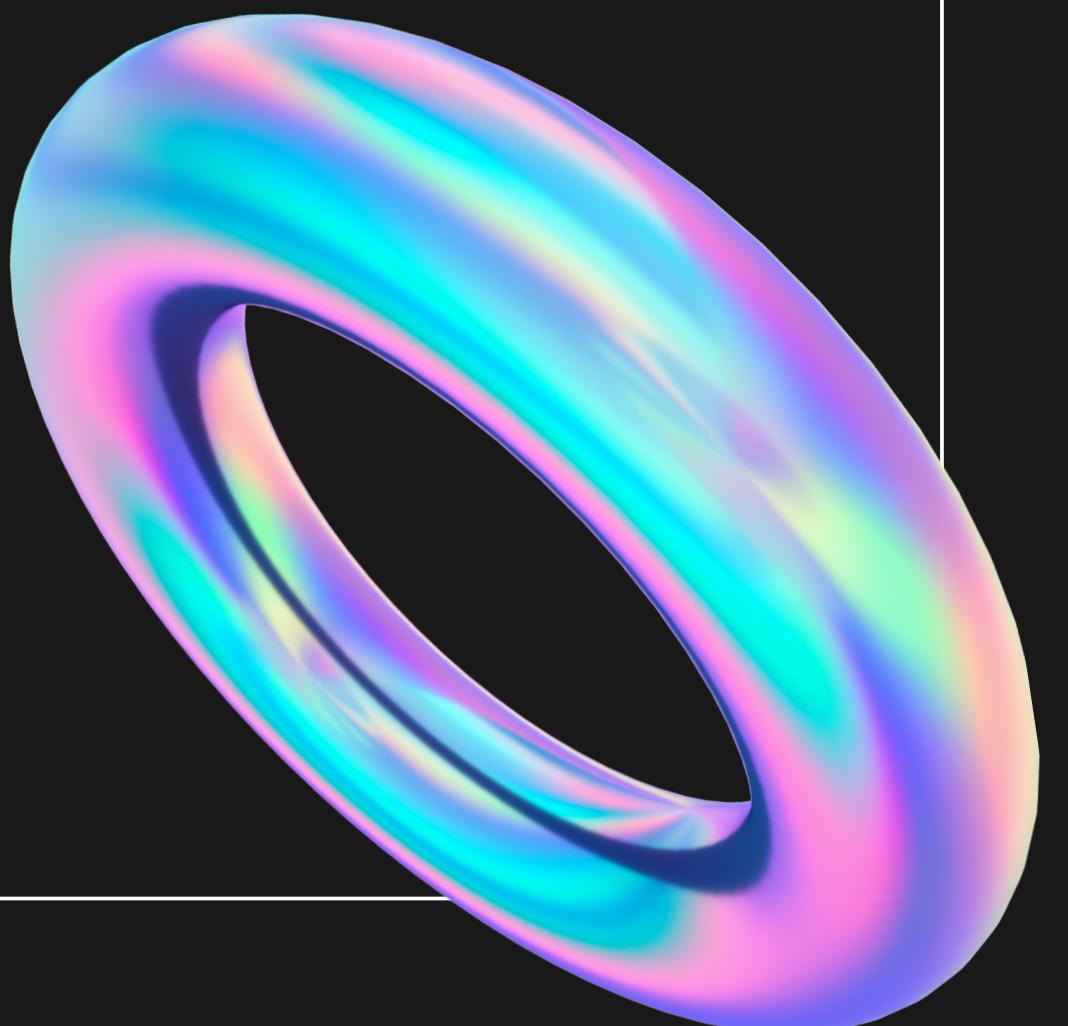
 THIS STORY IS FROM JULY 13, 2020

### 'Public fail to practise social distancing'



# A STEP TOWARDS SAFETY

A LOOK AT HOW TECHNOLOGY CAN HELP SAVE  
LIVES DURING THIS PANDEMIC



# DISCUSSION POINTS

Key topics covered  
in this presentation

The problem

The solution

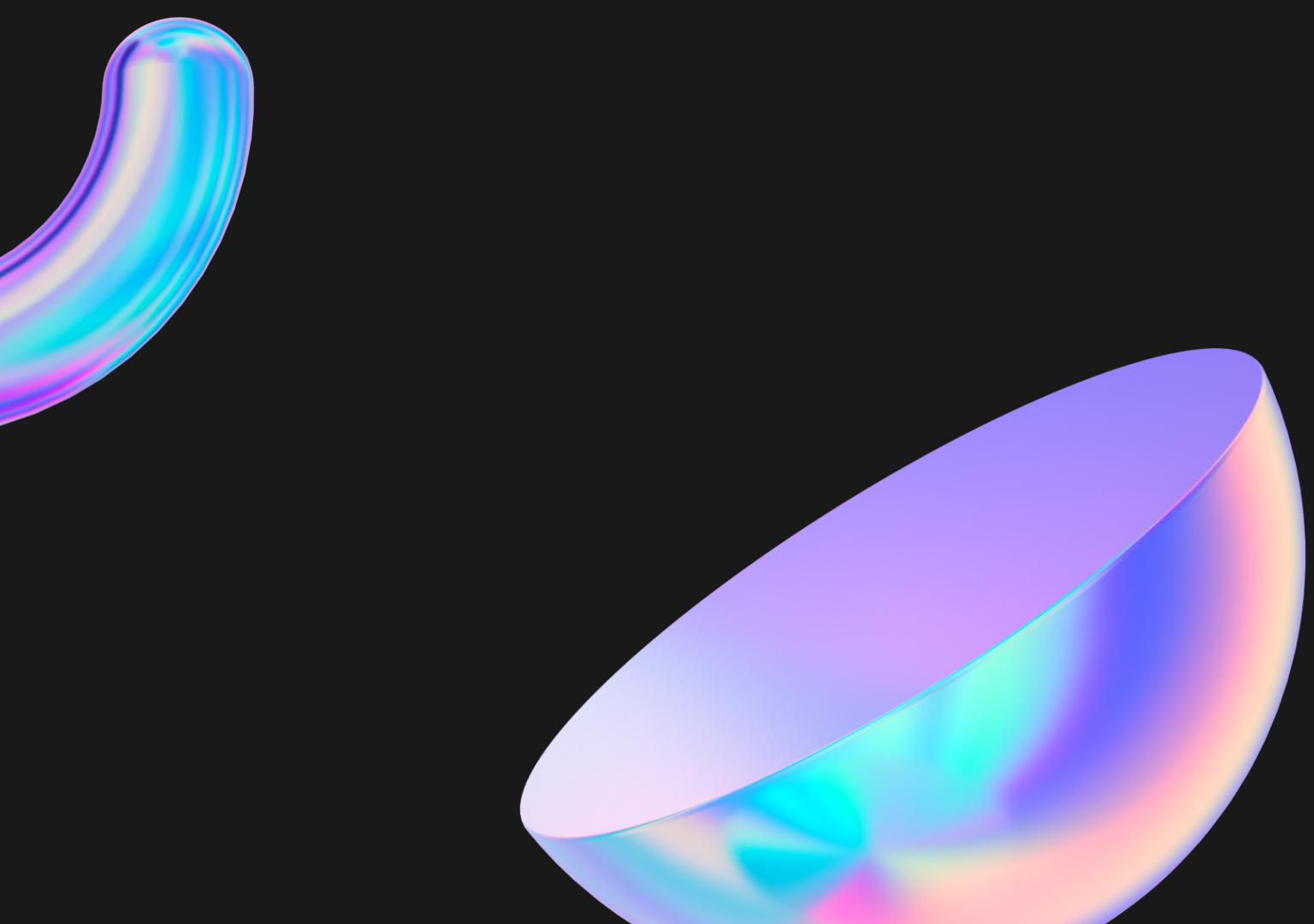
Technologies used

USP & Features

Market Validity & Target Market

Prototype Implementation

Business model



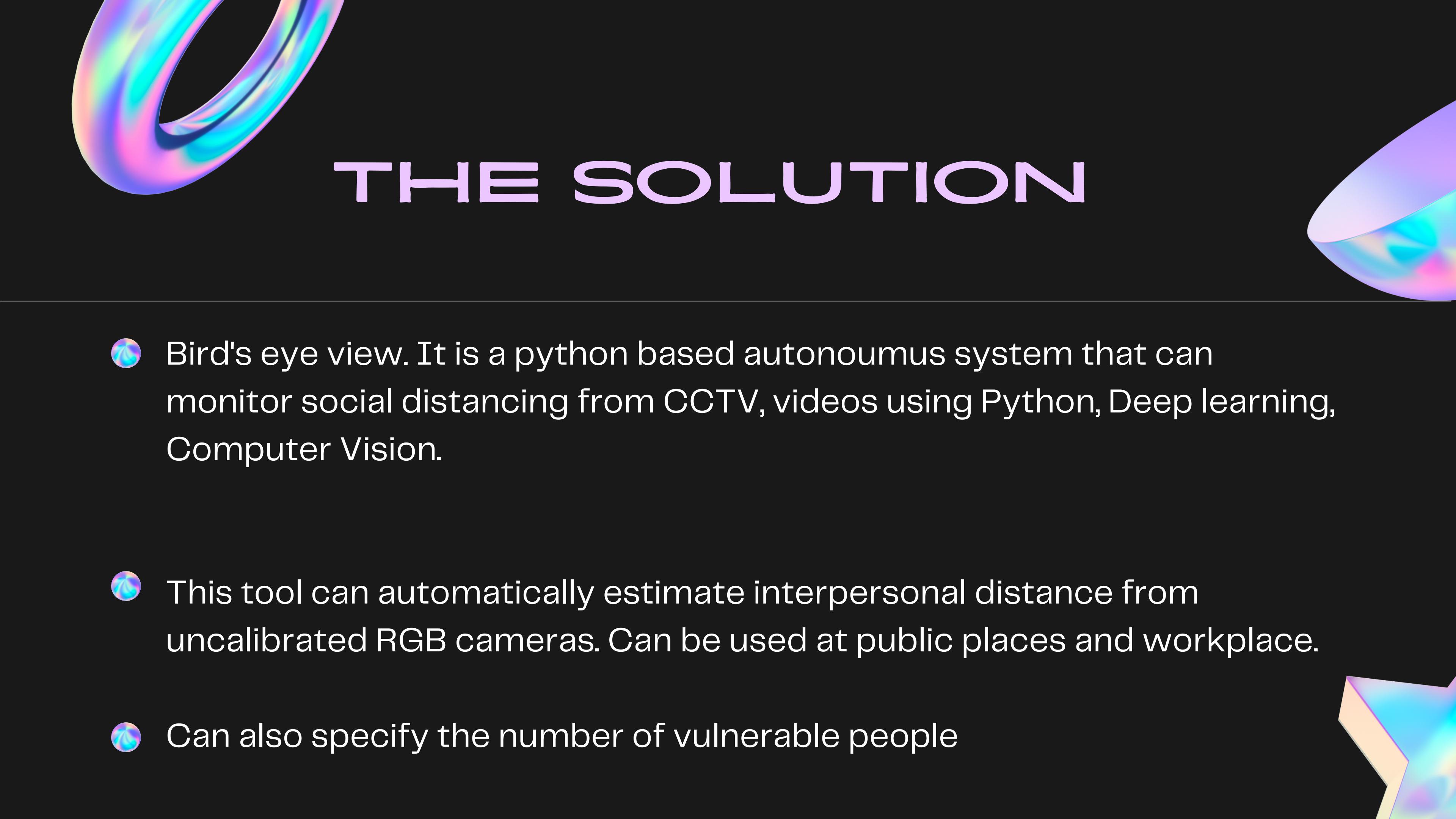
# THE PROBLEM



If social distancing is implemented at the initial stages, it can perform a pivotal role in overcoming the virus spread and preventing the pandemic disease's peak.



However due to the large population, it becomes impossible to control the crowd. Or moreover recognize instances where social distancing isn't observed.



# THE SOLUTION

- Bird's eye view. It is a python based autonomous system that can monitor social distancing from CCTV, videos using Python, Deep learning, Computer Vision.
- This tool can automatically estimate interpersonal distance from uncalibrated RGB cameras. Can be used at public places and workplace.
- Can also specify the number of vulnerable people

# THE SOLUTION



# TECHNOLOGIES USED

## FOR BACKEND

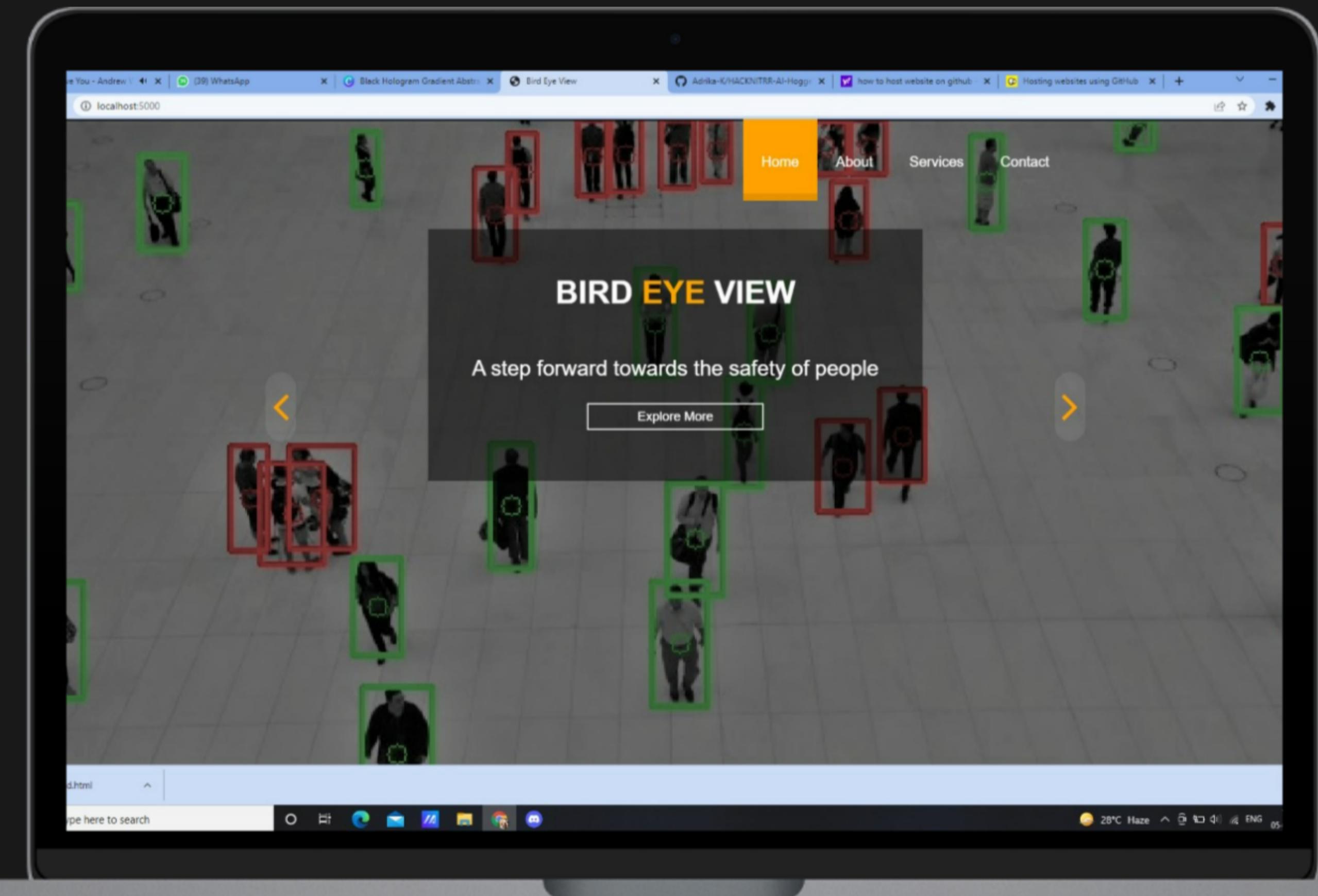
## FOR FRONTEND

### Libraries used for the model

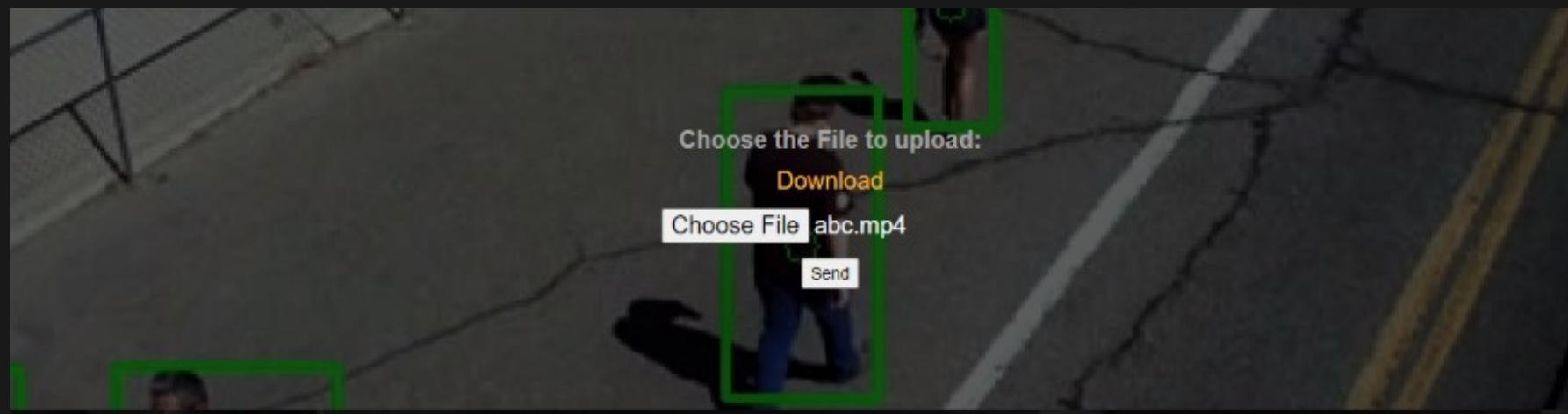
- astroid==2.4.2
- colorama==0.4.3
- imutils==0.5.3
- isort==4.3.21
- lazy-object-proxy==1.4.3
- mccabe==0.6.1
- numpy==1.18.5
- opencv-python==4.2.0.34
- pylint==2.5.3
- scipy==1.4.1
- six==1.15.0
- toml==0.10.1
- typed-ast==1.4.1
- wrapt==1.12.1

We're connecting the python based model to a website (made with html, css and js) using flask

# PROTOTYPE IMPLEMENTATION



# PROTOTYPE IMPLEMENTATION



```
File Edit Selection View Go Run Terminal Help
social_distancing_detector.py detection.py Web_connect.py index.html Settings
E: Hackathon > HackNITRR > social-distancing-detector-master > social-distancing-detector-master > configs > detection.py > ...
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
```

```
File Edit Selection View Go Run Terminal Help
social_distancing_detector.py Web_connect.py index.html Settings
E: Hackathon > HackNITRR > social-distancing-detector-master > social-distancing-detector-master > social_distancing_detector.py > ...
19 # construct everything from new
20 apsz = argparse.ArgumentParser()
21 apsz.add_argument("-i", "--input", type=str, default="", help="path to (optional) input video file")
22 apsz.add_argument("-o", "--output", type=str, default="", help="path to (optional) output video file")
23 apsz.add_argument("-d", "--display", type=int, default=1, help="whether or not output frame should be displayed")
24 artgys = vars(apsz.parse_args())
25
26 # initializing YOLO
27 Pathlabels_name = os.path.sep.join([cng.MODEL_PATH, "coco.names"])
28 LABELS = open(Pathlabels_name).read().strip().split("\n")
29
30 # finding paths
31 pathweightsinitialize = os.path.sep.join([cng.MODEL_PATH, "yolov3.weights"])
32 configPath = os.path.sep.join([cng.MODEL_PATH, "yolov3.cfg"])
33
34
35 print("IF you see this that means code is working ....")
36 TrackerPY = cv2.dnn.readNetFromDarknet(configPath, pathweightsinitialize)
37
38 # you need gpu for good performance
39 if cng.Needtheuseofgpu:
40     TrackerPY.setPreferableBackend(cv2.dnn.DNN_BACKEND_CUDA)
41     TrackerPY.setPreferableTarget(cv2.dnn.DNN_TARGET_CUDA)
42
43 notlt = TrackerPY.getLayerNames()
44 notlt = [notlt[i - 1] for i in TrackerPY.getUnconnectedOutLayers()]
45
46 # loading video stream
47 print("if you see this that means everything is working fine")
48 # no video ?!! no problem it loads webcam also
49 vs = cv2.VideoCapture(artgys["input"] if artgys["input"] else 0)
50 VideoWriter = None
51
52
53 while True:
54     (gand, ftrrate) = vs.read()
55     if not gand:
```



## USP AND FEATURES

---

Can be further altered to send an email alert in real-time. For eg: If the total number of violations exceeded in a store/building, we simply alert the staff.

---

No upfront cost.

98% accuracy.

Enhance the pre-existing CCTVs cameras.

# MARKET VALIDITY & TARGET MARKET

---

Can be used in collaboration with the government.

---

Can be used for tracking other objects too. This might range from a javelin to a missile

---

Can be used by different workplaces too.

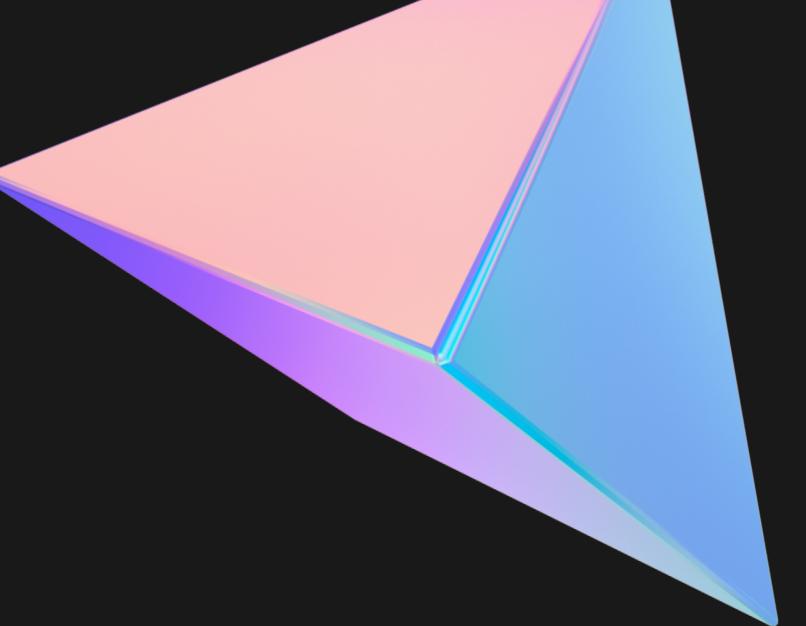
---

Can be used by private car manufacture enterprises.

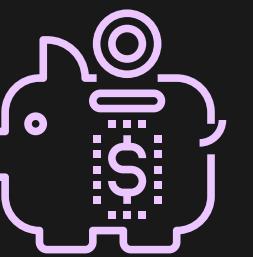




# BUSINESS MODEL



Niche  
market; not  
much  
competition



Highly  
cheaper  
than the  
competition



More  
accurate  
than the  
competition



Doesn't collect  
any data of a  
particular person

# THANK YOU!

Adrika, Aaditya, Kanupriya