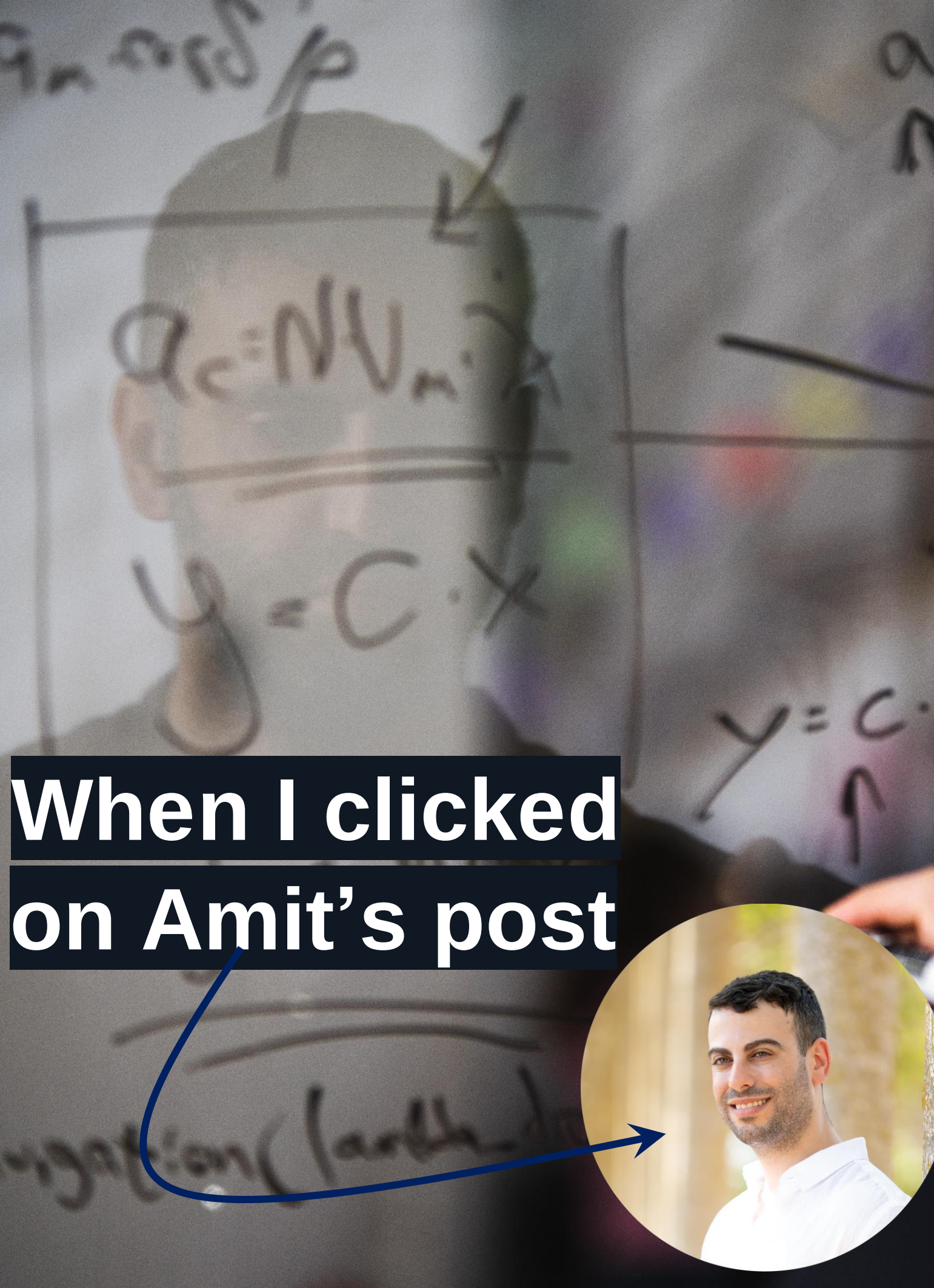

When I clicked
on Amit’s post

A post of Amit jumped in my feed:

Amit Elbaz • 1st
M.Sc. Mechanical engineering graduate. @ Ben-Gurion University | Seeking algo...
1w •

Hello LinkedIn community! I'm excited to announce my latest project focused on multiple object tracking (MOT) using the combination of the YOLOv3 object detection algorithm and Kalman Filter. In this endeavor, I employed a pre-trained YOLOv3 [illegible] for object detection and a set of Kalman filters for robust tracking. Through this project, I aimed to gain firsthand experience with the challenges of MOT using actual camera feeds.
This proj[illegible]s a testament to my passion for continuous learning and pushing my [illegible]ndaries. If you find my project intriguing, I would be delighted to connect and exchange ideas. You can find the YouTube video demonstration of my project at [https://lnkd.in/dFDyGwvh]. In order for me to improve further, your feedback and constructive criticism are invaluable.
#ObjectTracking #YOLOv3 #KalmanFilter #ComputerVision

You and 41 others 11 comments

YOLO

Kalman

...

C4dynamics

These words caught my eyes:

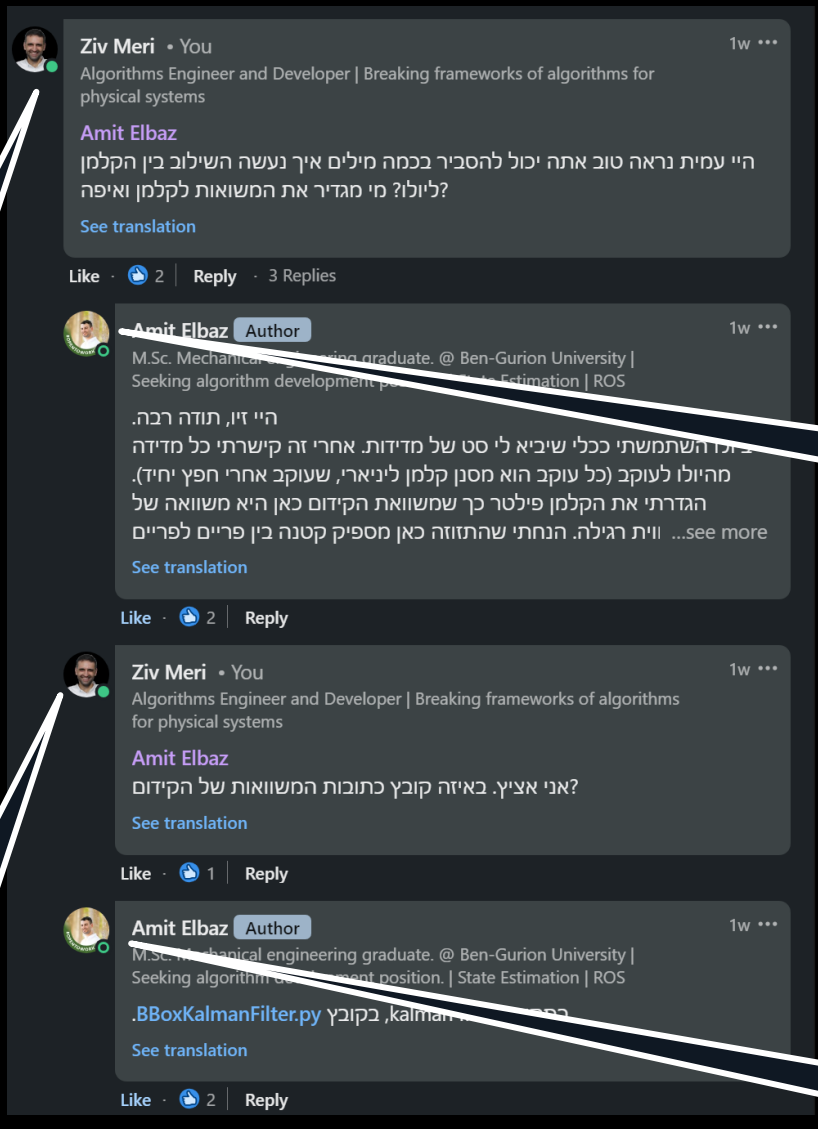
“YOLOv3 object detection algorithm”

“Kalman Filter”

“actual camera feeds”

C4dynamics

After small talk in public



Looks Great!

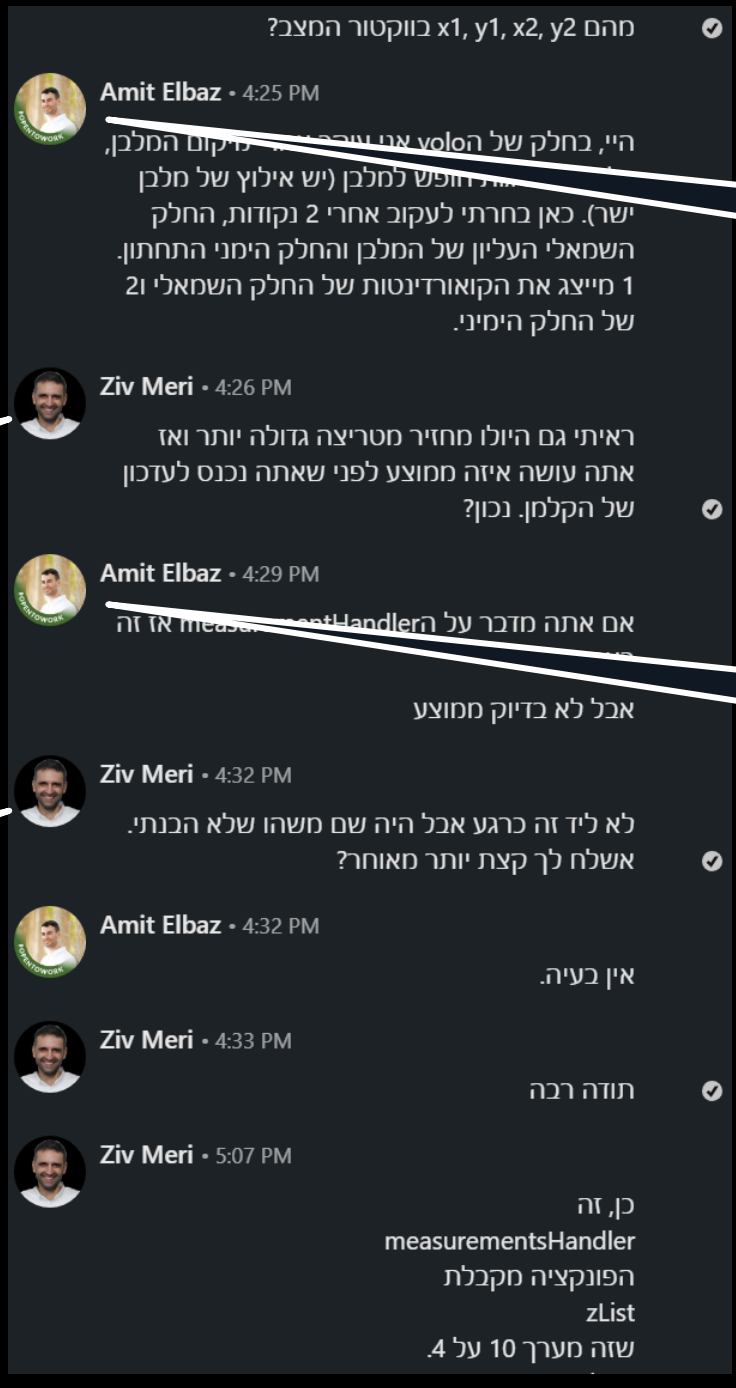
Thank you

Blah blah..

Of course

C4dynamics

And little longer in private



detection

objection

(-;

| -:

C4dynamics

I figured out his project may be a perfect fit for C4dynamics!

An opportunity to show how it supports algorithms of camera inputs

C4dynamics

So I took his original (great!) program

```python
import numpy as np
import cv2
from yoloDetector import yoloDetector
from Kalman_Filter.BBoxKalmanFilter import BBoxKalmanFilterDictionary
from sklearn.neighbors import NearestNeighbors

from functions import measurementsHandler , trackerMaker , RemoveDoubles


import argparse


class YoloTracker():

    def __init__(self , name
        if saveSolution: sav
        if name == '0': name
```

```python
MODEL_SIZE = (416, 416,3)
NUM_OF_CLASSES = 80
CLASS_NAME = './data/coco.names'
MAX_OUTPUT_SIZE = 40
MAX_OUTPUT_SIZE_PER_CLASS= 20
IOU_THRESHOLD = 0.5
CONFIDENCE_THRESHOLD = 0.5

class yoloDetector():

    def __init__(s
```

```python
class KalmanFilterBBox():
    # x(t) = A(t) * x(t-1) + e(t) || e(t) ~ N(0,Q(t))
    # z(t) = H(t) * x(t) + r(t)   || r(t) ~ N(0,R(t))

    # x(t) = [x1 y1 x2 y2 vx vy]

    def __init__(self , dt = 0.5):
        R = np.random.randint(0,255)
        G = np.random.randint(0,255)
        B = np.random.randint(0,255)
        self.color = [R,G,B]

        self.dt = dt
        self.A = np.array([[1,0,0,0,dt,0],
                           [0,1,0,0,0,dt],
                           [0,0,1,0,dt,0],
                           [0,0,0,1,0,dt],
                           [0,0,0,0,1,0],
                           [0,0,0,0,0,1]
                          ])
```

C4dynamics

with his support

And re-implemented it with C4dynamics

datapoint is a core element in c4d

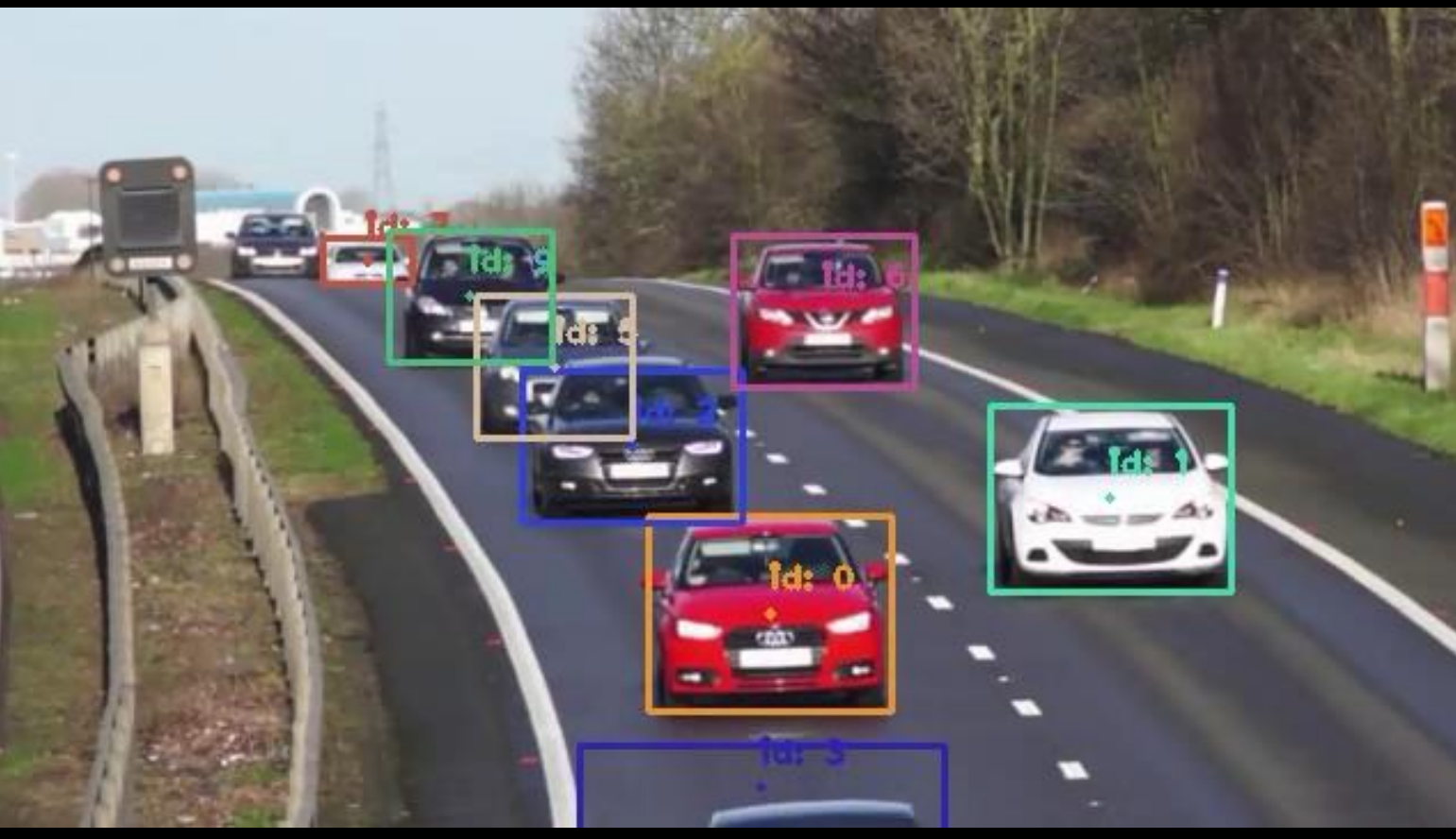
Import C4dynamics as c4d
…

```
class tracker(c4d.datapoint):
    # a datapoint
    # kalman filter
    # display color
    ##

    def __init__(obj, z):

        super().__init__()  # Call the constructor of c4d.datapoint

        obj.filter = c4d.filters.kalman(np.hstack((z, np.zeros(2))), P, A, H, Q, R)

        obj.counterSame = 0
        obj.counterEscaped = 0
        obj.appear = 0
```

And I used the built-in Kalman Filter

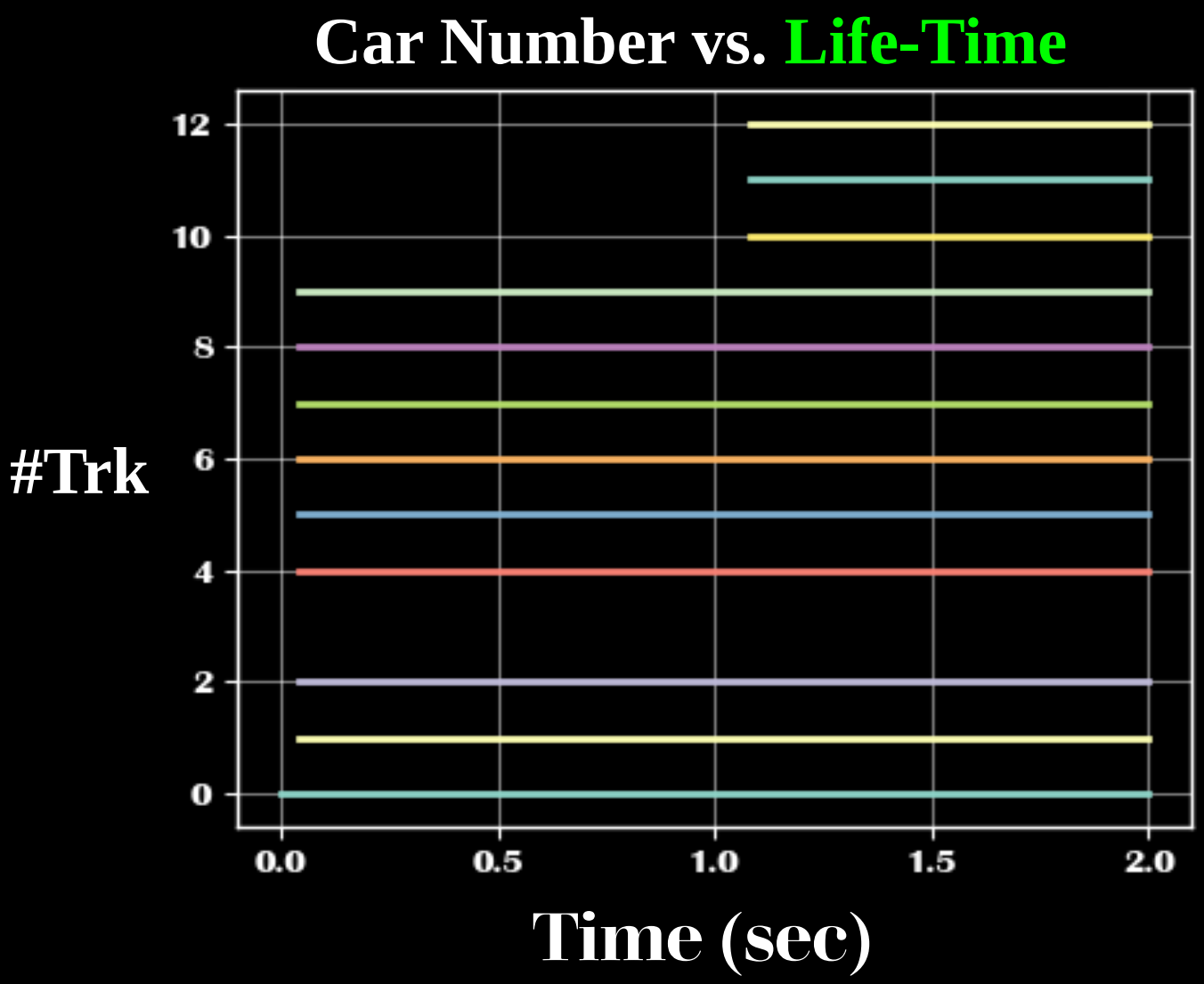
C4dynamics

So now, it’s not only doing cars detection and tracking



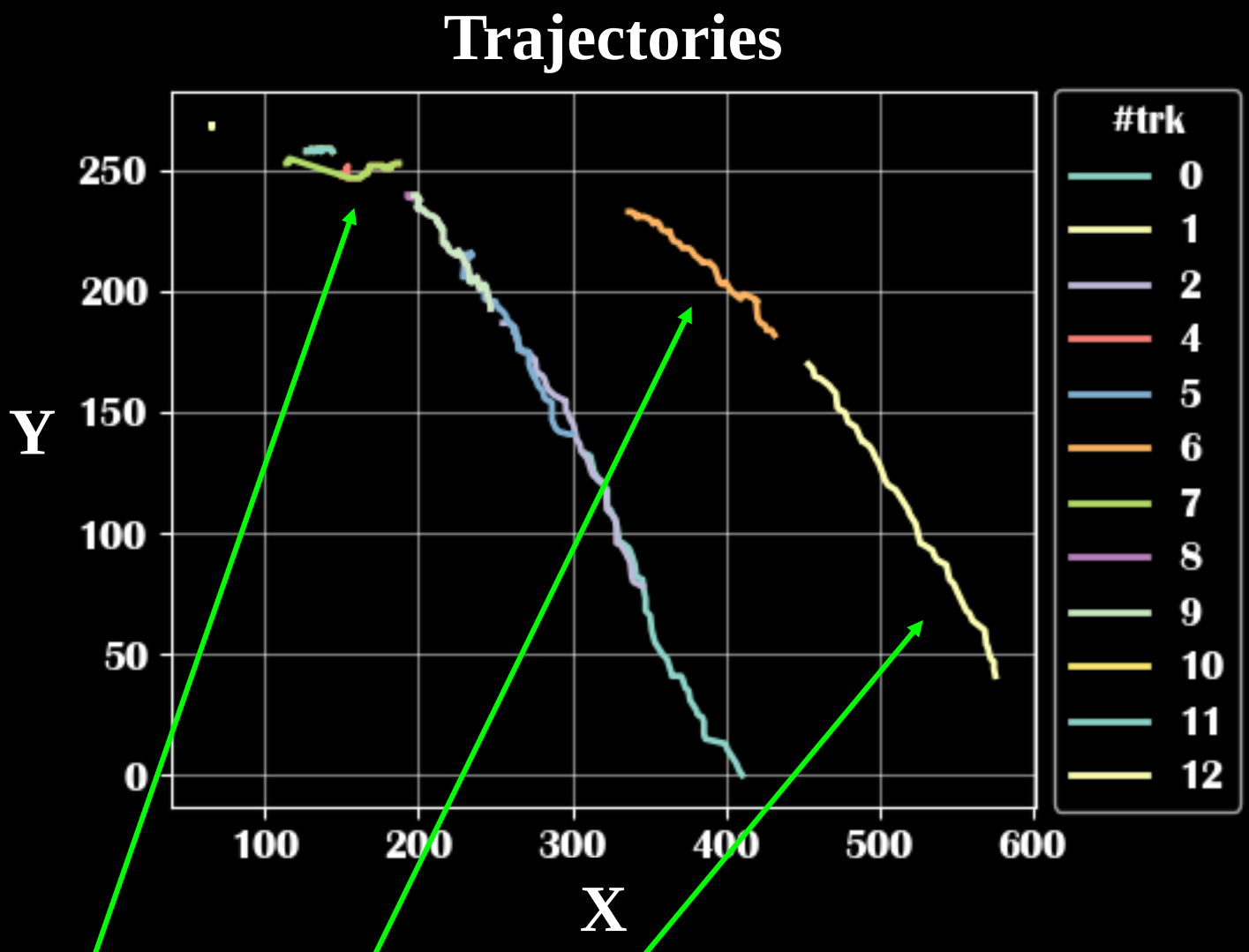
C4dynamics

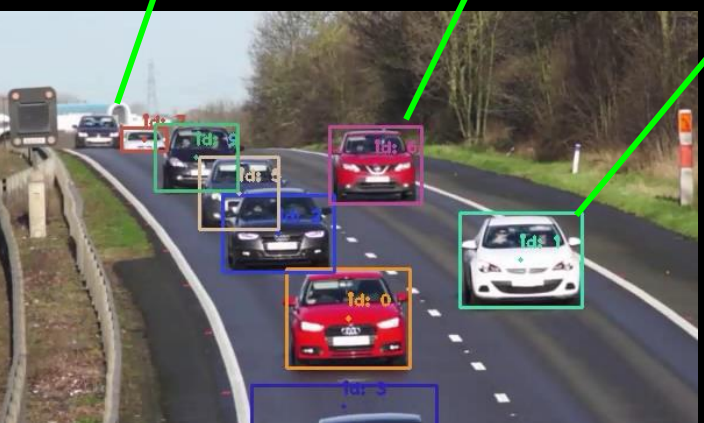
But also, has time-series capabilities



C4dynamics

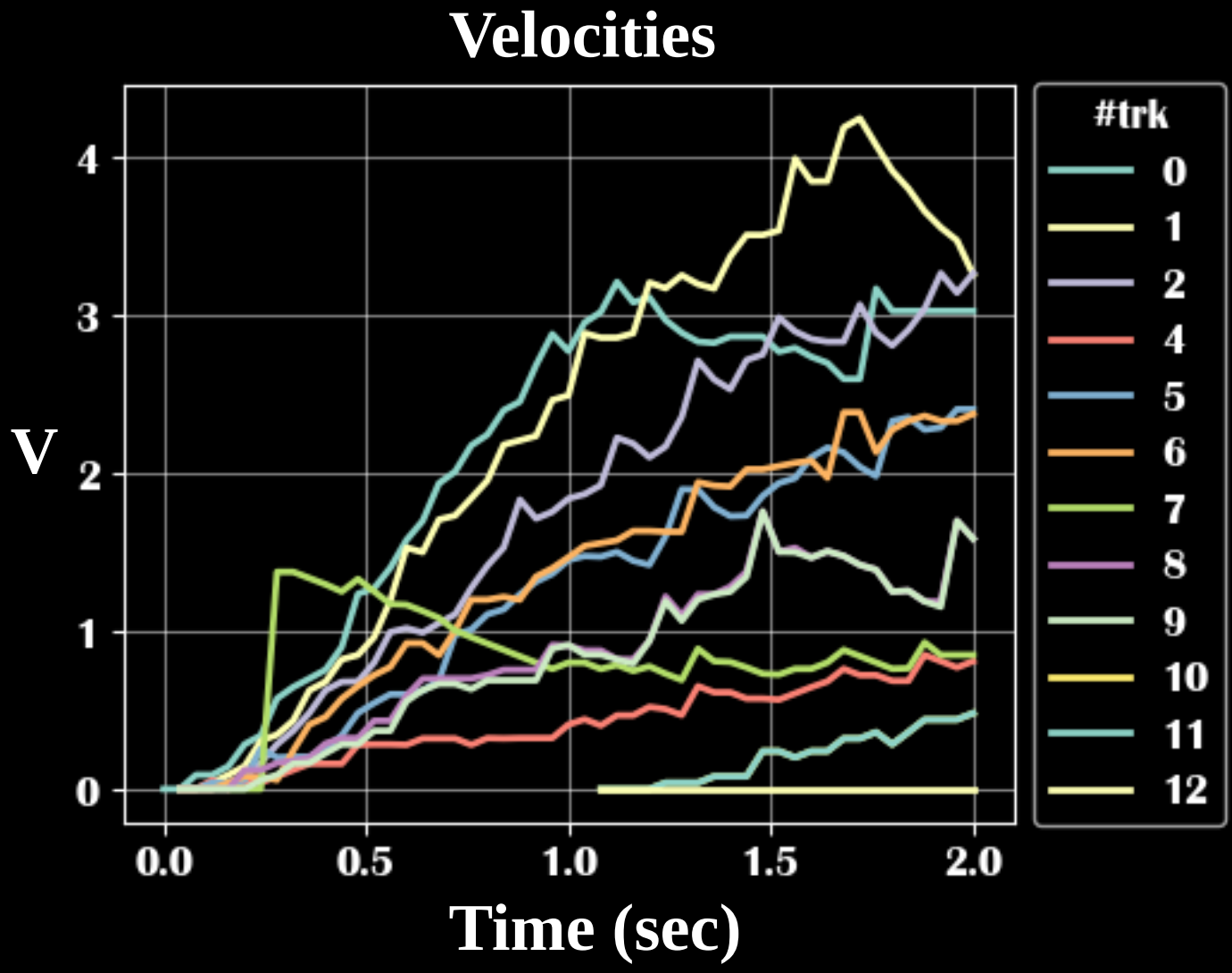
Spatial operations





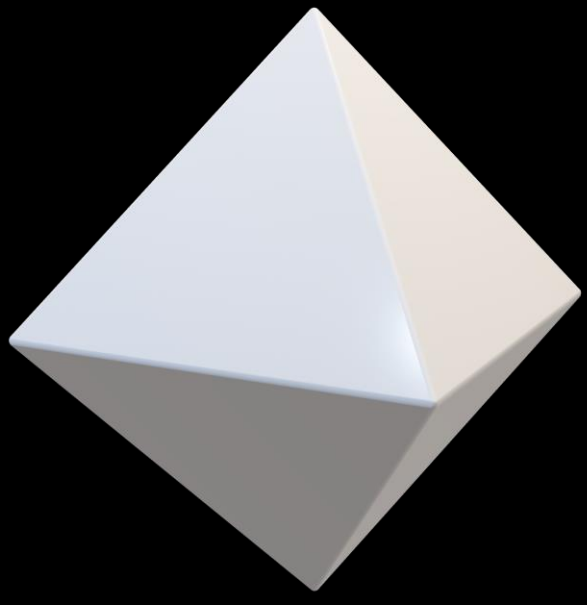
C4dynamics

And other dynamical manipulations



C4dynamics

Then never underestimate your networks and connections!



A diamond opportunity may be waiting just for you.

C4dynamics

Steal this talent, computer vision algorithms engineer

Lucky for you, he's looking for his next challenge

C4dynamics

Want to collaborate on C4dynamics too?

DM me with a short description of your project

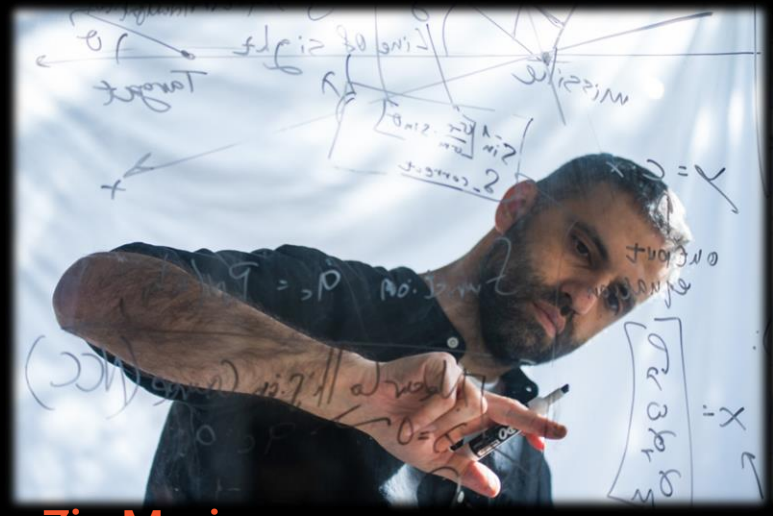
C4dynamics

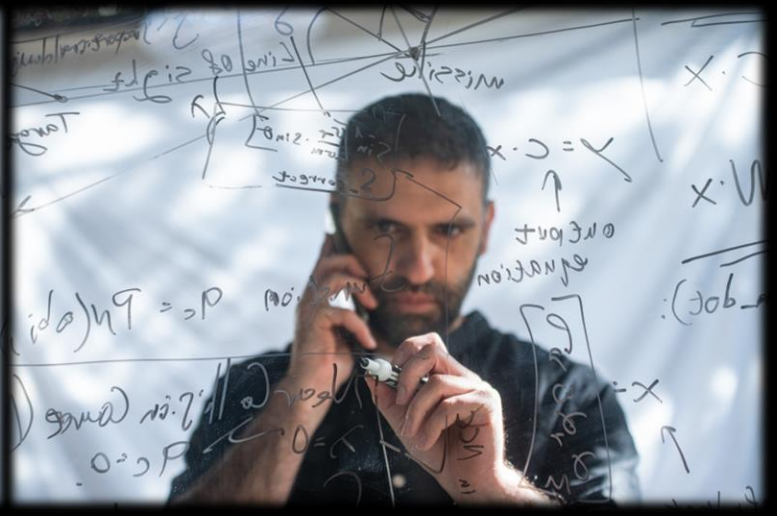
Framework for algorithms engineering of physical systems

Now with a new example:

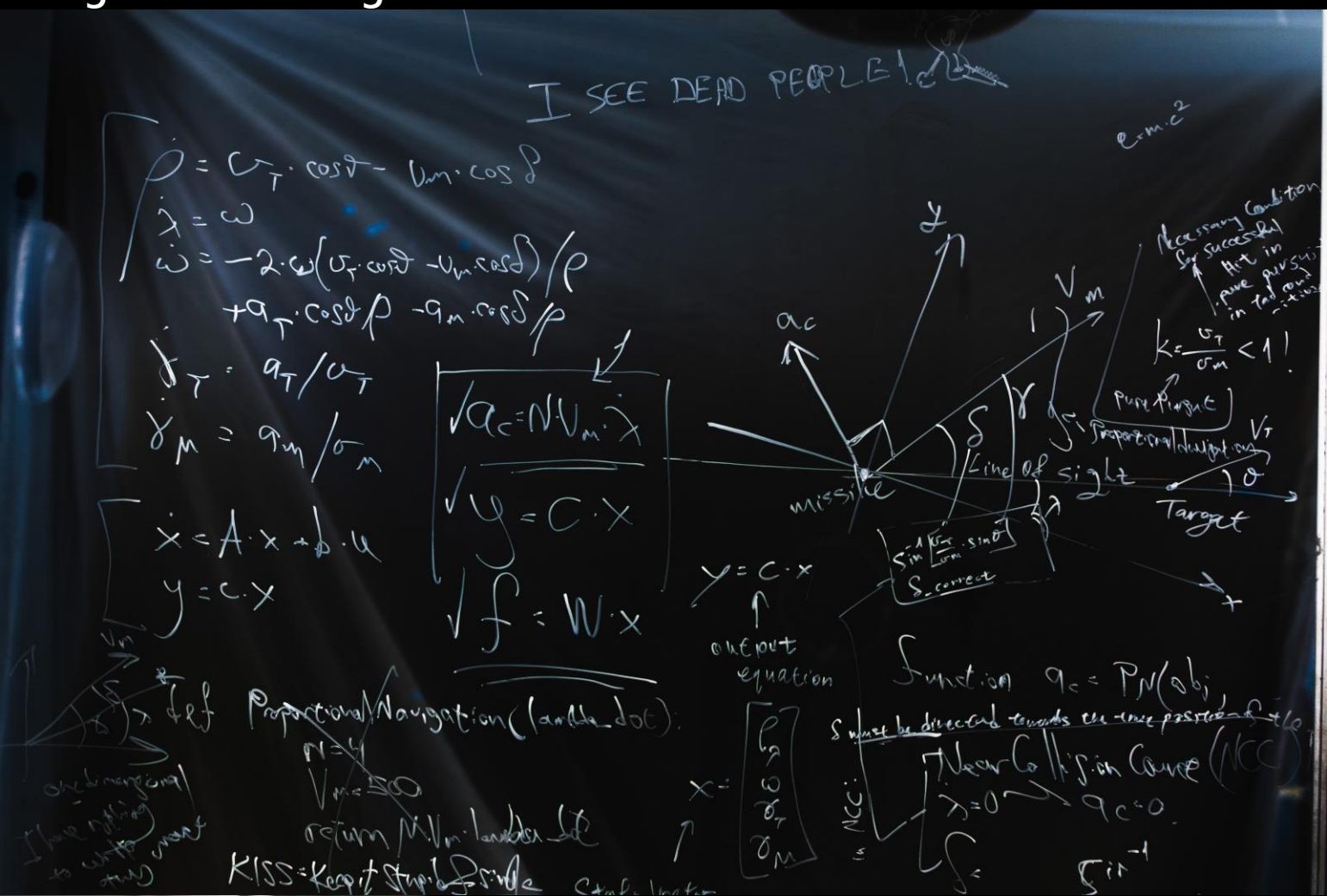
- ✓ Objects detection
- ✓ YOLO
- ✓ Kalman Filter

C4dynamics





Ziv Meri
Algorithms Engineer



Gavriel Weinberger
Visual Content Creator

C4dynamics