Assisting Blind Person With Computer Vision

UNDER GUIDANCE OF Prof. Dr. Srijith P.K, Athul A.F
Dr. P. K. SRIJITH & ATHUL

- BOJJA VARAHA SAI GNANESWAR
- KONDAPALKALA RITHVIK
- MAHIDHAR ALAM

OBJECTIVES

- DATA SET COLLECTION
- OBJECT DETECTION
- DEPTH CALCULATION
- EDGE DETECTION
- MOVEMENT ASSISTANCE

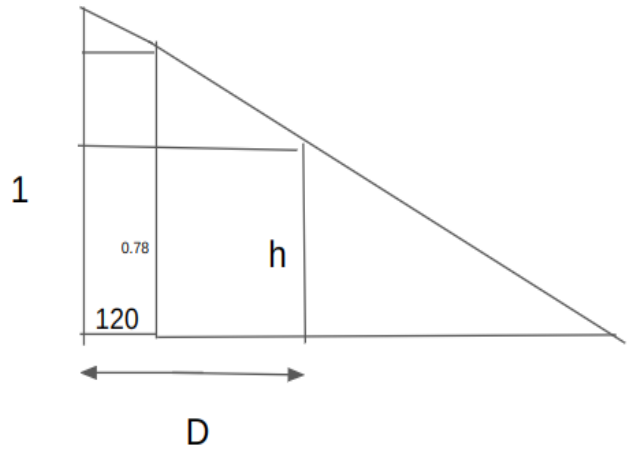
STAGES/PHASES

- **PHASE_1 :** OBJECT DETECTION (OBJECT NAMES)
- **PHASE_2 :** ADDED DEPTH ESTIMATION FOR PERSONS & IMPROVED OBJECT DETECTION
- **PHASE_3 :** EDGE DETECTION ADDED, IMPROVED DEPTH ESTIMATION, OUPUT AS VOICE & MOVEMENT ASSISTANCE

OBJECT DETECTION

- Algorithm: YOLOv5
- Recognizing the presence of an object: if it is with-in 300cms range
- Alert message
  - Danger: if any object comes with-in 100cms range
  - Safe: if no object is with-in the range of 100cms range
- Alert messages are invoked only when there is a change from Danger to Safe or change from Safe to Danger.
- To avoid continuous fluctuation (when a object moves from into or away from the 100cms range), If an object is in danger region, untill it goes more than 150cms farther we declare it as danger.

Depth Detection

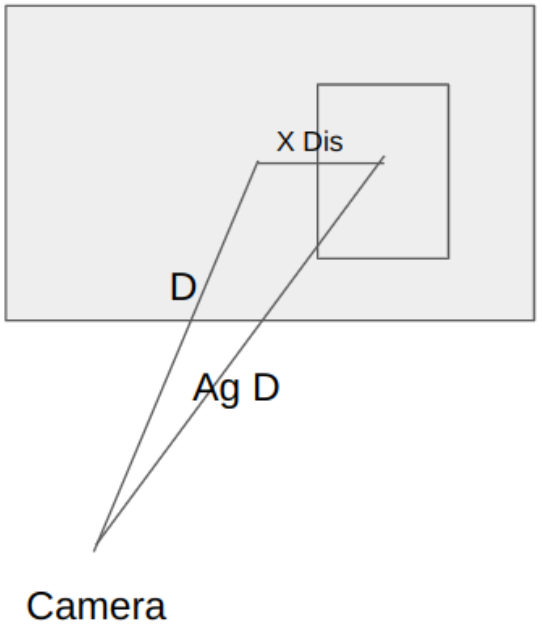


Using Similar Triangles:

120/(1-0.78)=D/(1-h)

D=(120/(1-0.78)) *(1-h)

Modified Depth Detection using Angular View



AgD=sqrt(D^2+ XDis^2)

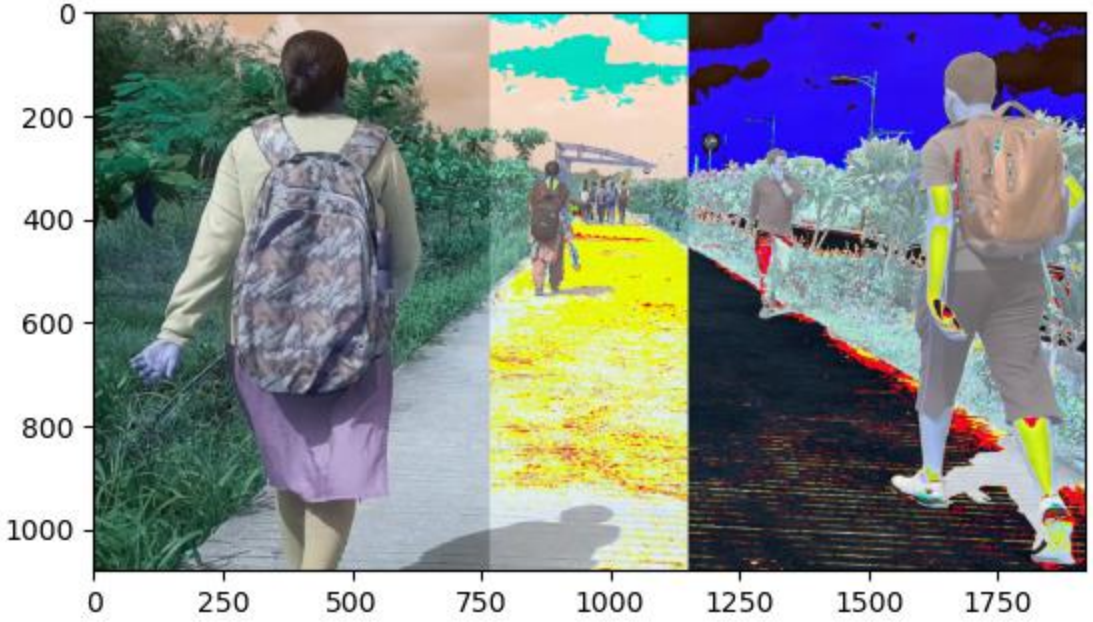
Movement Assistance

- Step 1: Frame division and predicting the occupied and un-occupied sub-frames
- Step 2: Predicting the edge flags based on distance from boundaries of pedestrian path
- Step 3: Predicting the action for present frame.
- Step 4: Assisting the movement based on the present and past actions

Step 1: Frame division

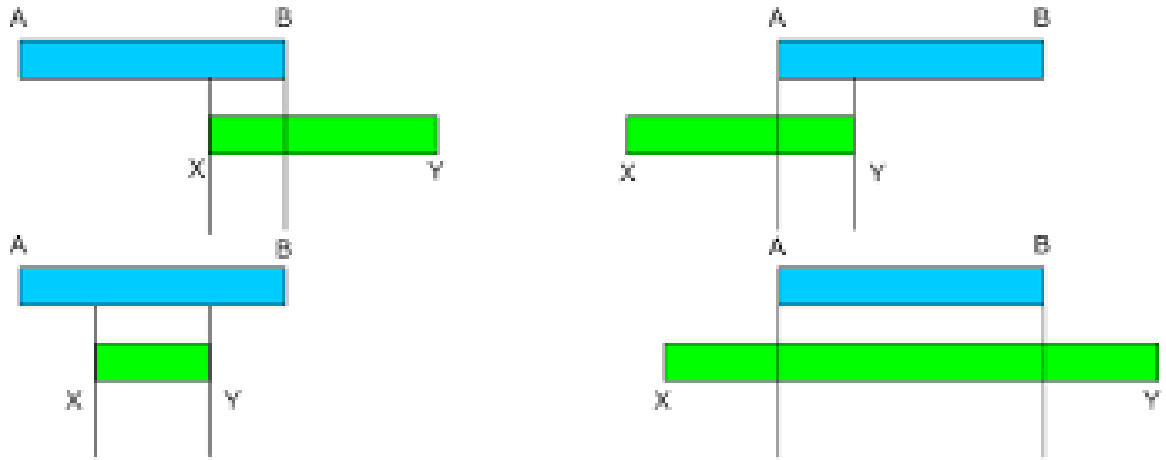
- Divided a frame into 3 sub frames and assigned flags to each sub-frame:
  - (L:M:R = 0.40:0.20:0.40)
  - Left frame: flag – f_L , range = (0.0, 0.40)
  - Right frame: flag – $f_{R,}$ range = (0.41, 0.60)
  - Middle frame: flag – $f_{M,}$ range = (0.61, 1.0)

  Where,
  - $objf_k$ = 0 if the k^{th} frame is not occupied by any person with-in 100cm radius
  - $objf_k$ = 1 if the k^{th} frame is occupied by any person with-in 100cm radius



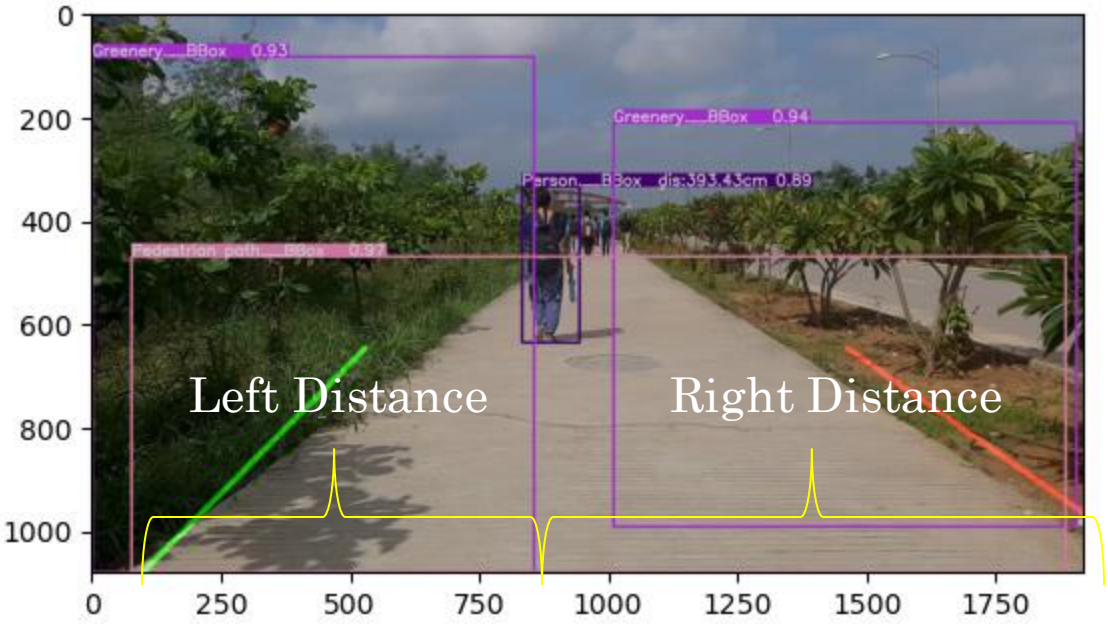
Step 1: Predicting objflag value

- Initially, flag values of 3 sub-frames are initialized to 0.
- For each detected person with-in 100cm radius:
  - Based on the co-ordinates of bounding box, we determine whether the person belongs to either left frame (or) right frame (or) middle frame.
    - Ex:
      - Boundaries of bounding-box (X, Y) (in green color)
      - Let boundaries of sub-frame be (A, B) (in blue color)
      - For overlapping on x-axis:
        - If (B>=X && A<=Y) :
          - $f_{sub\text{-}frame} = 1$



Step 2: Predicting the edgeflags based on distance from boundaries

- For pedestrian path boundaries detection, we are using edge detection and we are getting the line equations of the 2 edges of the pedestrian path L_1 AND L_2.
- Now we can get the distance of the person from both the edges by intersecting the line y=0 with L_1 AND L_2.
- if left distance < 0.1*frame width:
  - $edgef_L$ =1, $edgef_M$ = 1
- if right distance <0.1*frame width:
  - $edgef_M$ =1, $edgef_R$ =1
- Finally, flags = edgeflags | objflags



Step 3: Predicting the action for present frame

- The video we have taken is 30fps.
- We tried to predict the action for every 5 frames.
- We are taking 2 lists cummflags(for past cumulative flags), flags(for present frame flags) consisting of 3 entries and combining the flags with cummflags as:
  - cummflags = cummflags | flags
  - So that if in any of the past 5 flags, if the sub-frame is blocked, in the final cummflags it is set to be 1 indicating it is blocked.
- Based on the observations from the past 5 frames we determine the action as:
  - if $cummf_M = 0$:
    - action = straight
  - elif $cummf_L = 0$:
    - action = left
  - elif $cummf_R = 0$:
    - action =right
  - else:
    - action = stop

Step 4: Predicting the movement

- Based on the action we predicted for past 5-frames (say, PA) and the action we predicted for the present 5-frames (say, A), we suggest the movement as:

PA → A ↓	Straight	Left	Right	Stop
Straight	-	Straight	Straight	Straight
Left	Keep left	-	Keep left	Keep left
Right	Keep right	Keep right	-	Keep right
Stop	Stop	Stop	Stop	-

Flow chart for Movement Assistance