METODOLOGIA IN OUT

OUT

Por cada zona:

- Pasamos el objeto zona a espacio (4 puntos, un rectángulo)
- Entonces pasamos la variable memory que tiene una lista de listas de bounding boxes a una lista de listas de los ids de esas bounding boxes
- Creamos otra variable donde se listan todas las bounding boxes que hay en memory y cogemos solo las bounding boxes que hayan estado en la zona
- Entonces de esas bounding boxes que han pasado por la zona anteriormente, cogemos solo el ID y suprimimos los duplicados, así tendremos una lista de IDs que en la memoria han estado en la zona
- Y después cogemos todos los IDs que estamos viendo ahora
- Entonces que creamos una variable (ids_oprob_out) que es la intersección entre los ids que veíamos en la zona y los que vemos ahora, asi tenemos los IDs que están ahora y que han pasado previamente por la zona.
- Esos son todos los ID que tienen probabilidades de que sean un OUT
- Generamos un for loop que irá por cada ID que es probable que sea OUT.
- Entonces, para cada probable out ID:
    - Creamos un array igual de grande que el array memory, y la inicializamos a 0s. Esta array es para saber en que frames ese ID ha pasado por la zona. Entonces por cada previous frame (curr_frame-1, curr_frame-2, ..., curr_frame-(len(memory)) si ese ID ha estado en ese frame en la zona, le ponemos a ese array un 1. Ósea que los únicos valores que puede tener es 0 o 1, 0 → en ese frame ese Bounding Box no ha estado en la zona, 1 → en ese frame ese bounding box ha estado en la zona.
    - Nosotros buscamos arrays de la forma: [000011] que empiece con 0s y cuando aparezca el primer 1 no deje de ser 1s. Esto significa que antes no estaba y que de pronto ha aparecido por primera vez en la zona (esta saliendo de la tienda)
    - Así que comprobamos que sea así, y si encontramos un id que tenga una array de esta forma entonces, cogemos los bounding boxes de memory que sean igual al que es probable y lo ordenamos por frames, en orden ascendiente. La primera posición para el frame más antiguo.
    - Entonces tenemos una array de bounding boxes de forma que tenemos, la primera vez que vimos la bounding box, la segunda, ..., y así hasta el anterior frame.
    - Calculamos la distancia que hay de cada bounding box de ese array a la línea de la zona de interés. Con la fórmula:

      Line defined by two points [edit]

      If the line passes through two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ then the distance of (x_0, y_0) from the line is:[4]

      $$\operatorname{distance}(P_1, P_2, (x_0, y_0)) = \frac{|(x_2 - x_1)(y_1 - y_0) - (x_1 - x_0)(y_2 - y_1)|}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}.$$

      The denominator of this expression is the distance between P_1 and P_2. The numerator is twice the area of the triangle with its vertices at the three points, (x_0, y_0), P_1 and P_2. See: Area of a triangle § Using coordinates. The expression is equivalent to $h = \frac{2A}{b}$, which can be obtained by rearranging the standard formula for the area of a triangle: $A = \frac{1}{2} bh$, where b is the length of a side, and h is the perpendicular height from the opposite vertex.

    - Entonces si la array de distancias (original, sin aplicarle ninguna función de ordenación) es igual a una ordenada de forma ascendiente. Ósea que se está alejando de la línea, añadimos el id a la lista de counted ids, y le sumamos uno a los total out de esa zona.

IN

Por cada zona:

- Pasamos el objeto zona a espacio (4 puntos, un rectángulo)
- Entonces pasamos la variable memory que tiene una lista de listas de bounding boxes a una lista de listas de los ids de esas bounding boxes
- Creamos otra variable donde se listan todas las bounding boxes que hay en memory y cogemos solo las bounding boxes que hayan estado en la zona
- Entonces de esas bounding boxes que han pasado por la zona anteriormente, cogemos solo el ID y suprimimos los duplicados, así tendremos una lista de IDs que en la memoria han estado en la zona
- Y después cogemos todos los IDs que estamos viendo ahora
- Si la lista de personas que estamos viendo ahora es mayor o igual a la lista de personas que veíamos anteriormente eso significa que no se ha ido nadie (entrado en la tienda) y que no hace falta continuar con el algoritmo en esa zona.
- Entonces creamos una variable (ids_prob_in) que es la diferencia entre los ids que veíamos en la zona y los que vemos ahora, así tenemos los IDs que previamente han pasado por la zona, y ahora ya no estan.
- Por cada ID probable de ser un in (lista anterior):
    - Primero comprovamos que no se haya contado previamente y si no es el caso, proseguimos
    - Cogemos los bounding boxes de memory que sean igual al que es probable y lo ordenamos por frames, en orden ascendiente. La primera posición para el frame más antiguo.
    - Entonces tenemos una array de bounding boxes de forma que tenemos, la primera vez que vimos la bounding box, la segunda, ..., y así hasta el anterior frame.
    - Calculamos la distancia que hay de cada bounding box de ese array a la línea de la zona de interés. Con la fórmula:

      Line defined by two points [edit]

      If the line passes through two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ then the distance of (x_0, y_0) from the line is:[4]

      $$\operatorname{distance}(P_1, P_2, (x_0, y_0)) = \frac{|(x_2 - x_1)(y_1 - y_0) - (x_1 - x_0)(y_2 - y_1)|}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}.$$

      The denominator of this expression is the distance between P_1 and P_2. The numerator is twice the area of the triangle with its vertices at the three points, (x_0, y_0), P_1 and P_2. See: Area of a triangle § Using coordinates. The expression is equivalent to $h = \frac{2A}{b}$, which can be obtained by rearranging the standard formula for the area of a triangle: $A = \frac{1}{2}bh$, where b is the length of a side, and h is the perpendicular height from the opposite vertex.

    - Entonces si la array de distancias (original, sin aplicarle ninguna función de ordenación) es igual a una ordenada de forma descendiente. Ósea que se está acercando a la línea, añadimos el id a la lista de counted ids, y le sumamos uno a los total in de esa zona.