\*\*\*\*\*\*\*\*In short:

Copy 3 files (cameras, images, points3D) from COLMAP sparse model text results, and run in the following order:

move the text file generated by SFM sparse reconstruction to xu\_mono\_loc folder (image\_and\_camera\_id.txt, images.txt, points3D.txt)

for\_SFM\_xu\_run\_track.py (two pkl files generated:

counted\_fruits\_and\_last\_frame\_whole\_list.pkl, **tracks\_frame\_idx\_whole\_list.pkl(first element: all tracks in frame, second element: frame index e.g. ‘0001’))**

read\_SFM\_text\_results\_save\_valid\_points.py (generate files: image\_and\_camera\_id.txt which contains image position information ; valid\_points.npy, a list contain all valid points for every frame(every element is a list, where first element is array of all valid points in this frame and second element is float frame number such as 8.0))

read\_fruits\_and\_frame\_and\_find\_position.py (generate files: pos\_area\_frind\_of\_fruits.pkl, pos\_area\_frind\_of\_fruits\_text.txt(only for visualization: element order: X,Y,Z,area,fruit frame number))

find\_camera\_pos.py (generate files: img\_id\_position.npy and camera\_id\_position.txt (only for visualization \_id(frame\_id,X,Y,Z of 3D position of projective center for every frame)

world\_to\_camera\_frame.py (generate files: pos\_cam\_frame\_area\_fridx\_of\_fruits.npy and pos\_cam\_frame\_area\_fridx\_of\_fruits.txt(only for visualization: element order: X,Y,Z(all in camera frame), area(=image area \* Z^2), fruit frame number))

viasualize\_final\_results (Visualize results and generate oulier files: size\_outliers, double\_counted, distance\_outliers)

\*\*\*\*\*\*\*\*other details

I changed fruit\_tracker.py in Traking\_my\_implementation/scpye/track/ folder to save tracks as .pkl file here

Directly run for\_SFM\_xu\_run\_track the .pkl file will be generated

Do not delete frame images here!

1st version:

read\_tracks\_for\_four\_frames\_and\_savetxt: read .pkl tracks to .npy four frame tracks. Output is .npy files containing x,y positions of features (a list of features, pos is an array : [[x, y]])

list\_to\_text\_from\_read\_tracking\_result: # TODO: load the npy files in, after loading they are 2D array containing feature positions

2nd version:

read\_fruits\_and\_frame\_and\_find\_position: read in the fruits and the last frame they appear. Find the position according to the output of SFM.

!!important: bounding boxes extracted are with margin pixels!! can set the value (margin\_pixels) in fruit\_tracker.py

Important changes in fruit\_tracker.py:The final counted\_fruits\_and\_last\_frame is a list, every element is: [[x,y,w,h],[array(xc,yc)]], which saves the counted fruits and last frame for our localization!!!! bbox\_pre\_frame is (x, y, w, h)with margin\_pixels and t.prev\_pos is center position of bbox [xc,yc]

counted\_fruits\_and\_last\_frame\_whole\_list is a list containing the first element is counted\_fruits\_and\_last\_frame and the second element is frame number, e.g., frame0000