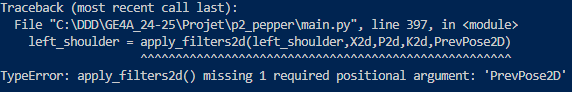
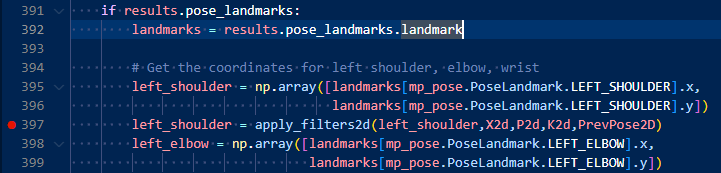
Logs:

1. TypeError: apply\_filters2d() missing 1 required positional argument: 'PrevPose2D'



Code:



Prototype:



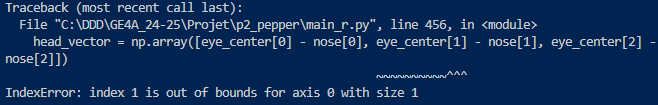


Solution:

Add the missing argument ‘jointnum’ when calling



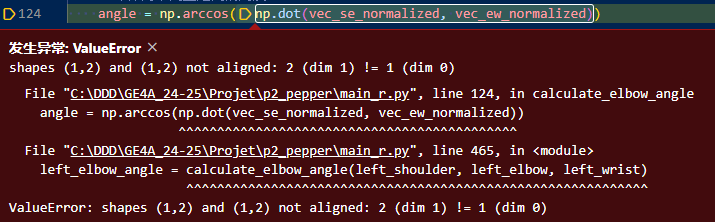
1. IndexError: index 1 is out of bounds for axis 0 with size 1



Through debug, we can find that the array ‘eye\_center’ is in shape of (1,3), which is a 2-dimension variable, so we should use the eye\_center[0][0], eye\_center[0][1] …

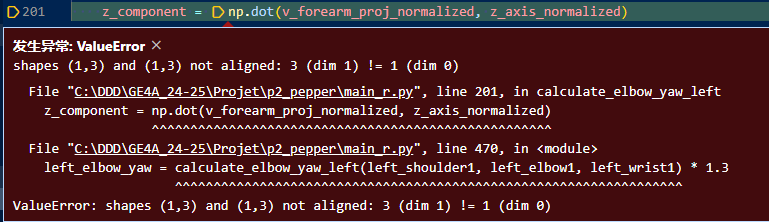
Correction:



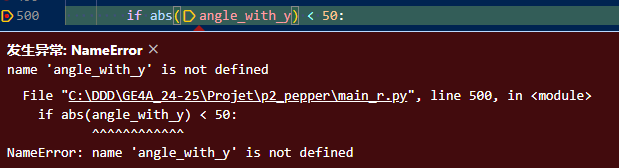


In this sentence, we use np.not() to multiply two matrices, but vec\_se\_normalized (size(1,2)) and vec\_ew\_normalized (size(1,2)) cannot multiply directly. We transpose the second matrix to fix the problem.

1. same problem with 3



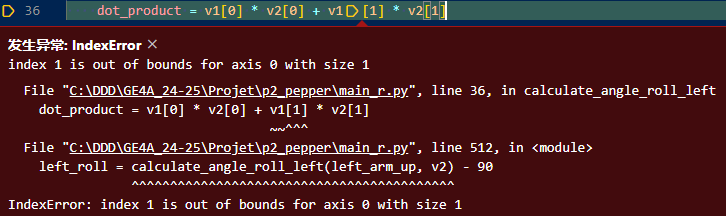
1. variable not defined before using



Angle\_with\_y and angle\_with\_x are not defined before using as parameters, so it has to give values before using.



1. IndexError: index 1 is out of bounds for axis 0 with the size 1



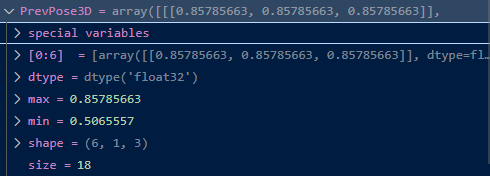
Variables:



Shape of V1 not the right for v1[1]

7. Correction: The function KalmenFilter returns wrong parameter.

The variable PrevPose3D is in shape (6, 1, 3),



The return value is a 2-dimesion array



While when using the returning value, there are too many errors about the size of value, so it could be the problem of return value.

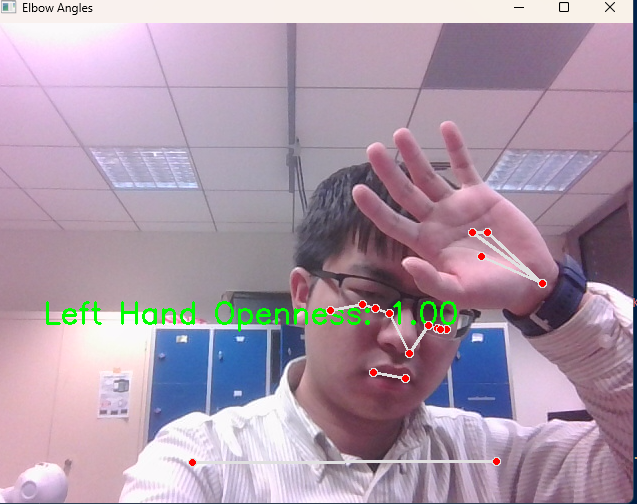
According to the knowledge above, we need a 1-demension array, so we should change the sentence as below:



The same situation applied to the function apply\_filters2d():



In the end, the program works.



Conclusion:

1 Lack of parameter jointnum for apply\_filters2d(), apply\_filters3d()

2 Incorrect use of variable PrevPose3D and PrevPose2D, which are 3-dimension variables.

3 Calling the variable before the definition