Для использования алгоритма коррекции навигации

1. Скачайте пакет Python “Open\_VO” с репозитория на гитхабе используя “python/build/Open\_VO.cp312-win\_amd64.pyd” или “python/build/open\_vo-1.0.0-py3-none-any.whl”.
2. Создайте проект в PyCharm (или другой IDE) и создайте \*.py файл. Установите “Open\_VO”:
   1. “.pyd”: положите пакет “Open\_VO.cp312-win\_amd64.pyd” в ваш проект и импортируйте библиотеку “import Open\_VO”.
   2. “.whl”: установите пакет “pip install python/build/open\_vo-1.0.0-py3-none-any.whl” и импортируйте библиотеку “import Open\_VO”.

3. Импортируйте следующие модули:

from Open\_VO import (MapAnalysis, Detection, DetectorPy, CameraParams, PosAngle, Pos\_i2, Pos\_d3, Pos\_f2, SurfaceData, LocalData)

4. Инициализируйте класс MapAnalysis:

map\_analyzer\_full = MapAnalysis(  
 input\_file="input\_tmp2.txt",  
 img\_folder="./",  
 output\_file="output\_tmp2.txt",  
 labels\_file="labels.txt",  
 model\_path="best.torchscript", # Or .pt, .onnx etc.  
 cuda\_enabled=True,  
 img\_w=640, img\_h=640,  
 sc\_thres=0.45, nms\_thres=0.50, max\_d=100,  
 gl\_model\_path="best.torchscript",  
 gl\_cuda\_enabled=True,  
 gl\_img\_w=1280, gl\_img\_h=1280,  
 gl\_sc\_thres=0.3, gl\_nms\_thres=0.45, gl\_max\_d=300  
)

\* настройте параметры под ваш проект

5. Обработайте карту спутниковых снимков подстилающей поверхности:

map\_analyzer\_full.calculate\_map\_objects()  
map\_analyzer\_full.calc\_map\_edges()

6. Найдите локальные объекты на кадре (переменная “frame”):

map\_analyzer\_full.calculate\_local\_objects(frame, identity\_delta=10,  
cam\_params=cam\_p, curr\_angles=angles,  
curr\_offset=offset, meter\_in\_pixel=m\_in\_pix)

7. Проверьте, что агент находится внутри границ карты спутниковых снимков:

is\_inside = map\_analyzer\_full.location\_verification(curr\_x=10.0, curr\_y=10.0, fov\_x=5.0, fov\_y=5.0, delta\_fov=1.0)  
print(f"Is (10,10) inside map? {is\_inside}")

8. Если агент находится внутри, запустите сопоставления локально найденных объектов с объектами на подстилающей поверхности, получите смещение (в метрах) и обновите координаты локально найденных объектов:

deltas\_found = map\_analyzer\_full.object\_matcher(  
 curr\_x=50.0, curr\_y=50.0, fov\_x=100.0, fov\_y=100.0, delta\_fov=10.0,  
 delta\_offset=20.0, match\_delta=15, conf\_overlap=2, obj\_per\_class\_thresh=1, scale=0.1  
)  
print(f"Object matcher found deltas: {deltas\_found}")

При удачном сопоставлении, используя подсчитанное смещение для корректировки позиции агента.

(Вы можете использовать этот модуль без детекции объектов на спутниковых снимках, если они уже размечены и сохранены в файл “input\_file”).