

SUMMARY

The field of logistical robotics is largely dominated by research oriented towards well-established larger companies looking to maximize efficiency and profits. This leaves small and medium enterprises lagging behind in the same areas, and has created a need for development of lower scale robotics solutions to be deployed in the workspaces of these companies. Since the robots would be working in spaces developed with human workers in mind, they need to be able to navigate intelligently, in which a large part is played by machine vision. While solutions for simple machine vision tasks, like object detection, are readily available in open projects, more difficult or specialized tasks require a case by case solution development. The task solved by the thesis at hand concerned the precise positional detection of a carrying frame for use in robot navigation.

To solve this task, firstly a theoretical outline of the implemented algorithm was devised. A YOLOv3 machine learning model would be used for carrying frame detection and region selection. This region would then be used by task specific code to calculate the position and relative angle of the carrying frame based on depth information from an on-board depth camera. Finally, a navigational target would be calculated based on the previously acquired position. This theoretical outline was implemented in Python, and communications between the robot and solution code were handled with ROS publishers and subscribers. The YOLOv3 machine learning model used for frame detection was re-trained to a batch loss of 1% from a model pre-trained on a different carrying frame by compiling a dataset of 62 images from data recorded from the robot's camera.

The solution code was first tested away from the robot on ideal test data, confirming the successful detection and position calculation for the frame and thus the completion of the main thesis goal. Then, live-testing on the robot was conducted for three specific cases, looking for consistent detection of the facing of the frame entrance to the right, left and head-on. Here it was discovered that the noise in the camera's depth measurements was too large to accurately detect the head-on case consistently. Possible solutions for this issue are described.

The results of this thesis can be further developed by testing the outlined three possible solutions to the noisy depth measurements: using a different camera model from the same manufacturer, not using ROS, or using a different depth sensing camera altogether. Furthermore, once the mentioned issue is eliminated, the machine vision solution can be fully integrated into the robot's navigational stack to be used in autonomous navigation.

KOKKUVÕTE

Praegusel hetkel domineerivad logistikarobotite arenduses suurfirmad, kes soovivad suurendada tootmises efektiivsust ja kasumeid. Seetõttu jäävad nendes valdkondades maha väikesed ja keskmise suurusega ettevõtted, kellel ei ole suuremahuliste robotikalahenduste kasutamiseks võimalusi, ning on tekkinud vajadus väikesemahuliste lahenduste arendamiseks just neile. Kuna robotid, mida kasutatakse nendes lahendustes, töötaksid tootmisruumis, mis on arendatud pidades silmas inimtöötajaid, peavad need oskama keskkonnas liikuda intelligentselt, milles mängib suurt rolli masinnägemine. Kuigi lihtsamad masinnägemise lahendused, nagu näiteks objektituvastus, on juba laialt levinud vabavaraprojektides, siis säärase spetsiifilised masinnägemise probleemid tuleb endiselt lahendada olukorrapõhiselt. Selles töös ongi käsil üks selline probleem, milleks on kandraami ja selle täpse positsiooni tuvastamine läbi masinnägemise, kasutamiseks roboti navigeerimiseks.

Antud probleemi lahendamiseks loodi esmalt teoreetiline kavand arendatavast algoritmist. Esiteks on kasutusel YOLOv3 masinõppe mudel kandraami tuvastamiseks ning piirkonnaavalikuks. Tuvastatud piirkonna ning roboti sügavustundliku kaamera pildi abil on seejärel võimalik arvutada kandraami suhteline nurk ning asukoht. Lõpuks arvutatakse saadud positsiooniinfo abil ka navigeerimissihtmärk robotile. Algoritm rakendati Python programmeerimiskeele ja ROS robotikatarkvara abil. Kasutatud YOLOv3 masinõppe mudel oli eeltreenitud erineva kandraami tuvastamiseks ning taastreeniti 62-pildilise andmestikuga.

Lahendust testiti esmalt robotist eemal, lindistatud andmetega, et kinnitada edukas positsiooni ja nurga tuvastamine ning töö peamise eesmärgi täitmine. Seejärel testiti lahendust otse robotilt tuleval andmevool, et kinnitada ühtlane kandraami sissekäigu suuna tuvastamine kolmel juhul: suunaga vasakule, paremale ja otse. Viimasel juhul tuli katsetamise käigus välja, et saadav sügavusandmestik ei ole küllalt täpne ühtlase tulemuse saavutamiseks. Kirjeldatud on ka võimalikud lahendused sellele probleemile.

Selle töö tulemuseks olevat lahendust võiks edasi arendada katsetades kolme välja toodud võimalikku lahendust eelmainitud probleemile: kasutada erinevat kaamera mudelit samalt firmalt, mitte kasutada ROS tarkvara, või kasutada täiesti teist kaamerat erinevalt firmalt. Kui märgitud probleem on lahendatud, saab lahenduse integreerida roboti navigeerimissüsteemi, kasutamiseks autonoomses navigatsioonis.