

KOKKUVÕTE

Käesoleva lõputöö eesmärk oli ehitada numbrituvastuse ringraja ajavõtu lahenduse prototüüp, mis tuvastab automaatselt numbrimärke ja mõõdab ringraja aegu.

Lõputöö esimeses peatükis uuris autor erinevaid numbrituvastus ja ajavõtusüsteeme. Peatükis tutvustatakse, mis on masinõpe, sügavõpe ning automaatne numbrimärgi tuvastussüsteem. Autor ei kohanud uurimise käigus ajavõtusüsteemi, mis kasutab numbrituvastust ringraja aegade võtmiseks. Lisaks võrdles autor erinevaid ajavõtusüsteeme, arutades nende puudusi ja eeliseid.

Teises peatükis uuris autor erinevaid alternatiivseid lahendusi tarkvarale ja riistvarale. Riistvara analüüsimise tulemuste põhjal tegi autor otsuse parima ühe pardaarvuti ning kaamera valiku osas. Prototüübi ehitamisel kasutati Raspberry Pi 4 numbrimärgi objekti tuvastamiseks ja reaajas pilditöötamiseks ning Raspberry Pi V2 kaamerat video sisendina. Üks lõputöö väljakutsetest oli numbrimärgi tuvastamise kiirus. Tuvastamise kiirust katsetati läbi kahe erineva objekti tuvastamisalgoritmi ja analüüsiti numbrimärgi lugemise mootorite kiirust ja täpsust. Lähtudes tulemustest valiti kiirem numbrimärgi tuvastamise algoritm ning kiirem ja täpsem numbrimärgi lugemise mootor.

Prototüübi tarkvara on kirjutatud C++ ja Pythoni keeles, kasutades Raspberry Pi operatsioonisüsteemi. Kaameralt video edastamiseks kasutatakse võrgu voogedastust. Prototüübi elektroonika tehti makettplaadi peale, mille abil saab juhtida LCD ekraani RGB LED-i. Lisaks on ühendatud ventilaator ja klahvlüliti makettplaadiga. Prototüübi detailide printimisel kasutas autor PLA plastikut, mille mõõtmed on 139 mm x 110 mm x 80,90 mm.

Lõputöö eesmärgid saavutati ning loodud süsteem töötab korrektselt. Autoril puudus eelnev kogemus masinnägemise süsteemide kohta, kuid on lõputöö tulemusega rahul. Süsteemi peamine puudus on numbrimärgi tuvastamise kiirus. Tulevikus selle parandamiseks, oleks vaja optimeerida numbrituvastuse algoritmi.

SUMMARY

This thesis aimed to develop a prototype for license plate recognition with a timing solution for race tracks, capable of automatically detecting license plates and measuring race times.

The first chapter reviews various license plate number recognition and timing systems. It introduces what machine learning, deep learning, and automatic license plate systems are. During the research, no existing system combining both license plate recognition and timing was found. Additionally, the author compared various timing systems, discussing their shortcomings and advantages.

In the second chapter, the author explored different alternative solutions for software and hardware. Based on the results of the hardware analysis, the author selected the most suitable single-board computer and camera. The prototype was built using Raspberry Pi 4 for license plate object detection and real-time image processing, along with a Raspberry Pi V2 camera for video input. One of the challenges of the thesis was the speed of license plate recognition. This was tested using two different object detection algorithms, and both the speed and accuracy of the license plate reading engine were analysed. Based on these results, a faster license plate recognition algorithm and a faster and more accurate license plate reading engine were chosen.

The prototype's software is written in C++ and Python, operating on the Raspberry Pi operating system. Network streaming is used for transmitting video from the camera. The electronics of the prototype were assembled on a breadboard, enabling control of an LCD screen, RGB LED, fan, and a key switch. The author used PLA plastic for printing the details of the prototype, with dimensions of 139 mm x 110 mm x 80,90 mm.

The objectives of the thesis were achieved, and the created system functions correctly. Despite having no prior experience with machine vision, the author is satisfied with the thesis results. The main drawback of the system is the speed of the license plate recognition. In the future, in order to improve this, it would be necessary to optimize the number recognition algorithm.