

Egy geometriai sávkövető algoritmus autonóm járművekhez

A geometric lane line tracking algorithm for autonomous vehicles

SÜTŐ József^{1,2}, Byron Fabricia Sarabia MORALES¹

¹ Informatikai Rendszerek és Hálózatok Tanszék, Informatikai Kar, Debreceni Egyetem, 4028, Debrecen, Kassai út 26.

² Információtechnológia Tanszék, Informatikai Kar, Eszterházy Károly Katolikus Egyetem, 3300 Eger, Leányka út 4/A.

Abstract

Autonomous vehicles constitute a technological convergence of automatic control, computer vision, and embedded systems. Their development has been made possible through significant advances in computer science, pattern recognition, and intelligent control methodologies. A fundamental prerequisite for autonomous driving is the precise determination of the vehicle's position relative to the roadway, as well as the reliable tracking of lane boundaries that define the navigable path. Among the established approaches to addressing this challenge, lane detection techniques based on computer vision have received particular attention. This study presents a lane-following methodology that integrates computer vision, geometric modeling, and proportional–integral–derivative (PID) control. The proposed method represents an enhanced version of a previously published algorithm and enables autonomous indoor navigation of vehicles along predefined trajectories.

Keywords: lane detection, machine vision, steering angle, PID controller.

Kivonat

Az önvezető járművek olyan különféle technológiákat integrálnak, mint az automatikus vezérlés, a gépi-látás és a beágyazott rendszerek. E járművek létrejötte a számítástechnika, a mintafelismerés és az intelligens irányítórendszerek területén elért fejlődésnek köszönhető. Ahhoz, hogy az önvezetés megvalósuljon ismerni kell a jármű útpályához viszonyított helyzetét és meg kell oldani az útpályát definiáló sávok követését. A gépi-látás alapú útsáv detektálás ennek a kihívásnak az egyik jól ismert megoldási módszere. Jelen tanulmány egy olyan sávvonal-követő módszert mutat be, amely gépi-látásra, geometriai számításokra és PID-szabályozásra épül. A cikkben bemutatott módszer egy korábban publikált algoritmusnak továbbfejlesztett változata. A javasolt módszer lehetővé teszi a járművek autonóm beltéri navigációját előre definiált útvonal mentén.

Kulcsszavak: sávdetektálás, gépi-látás, kormányzási szög, PID vezérlő.

1. BEVEZETÉS

Az autonóm járművek (AV-k) megjelenése potenciálisan paradigmaváltást idézhet elő a közlekedési infrastruktúrában. A technológia alkalmazása jelentős hatást gyakorolhat a közlekedésbiztonságra, a forgalmi torlódások mértékére, valamint a vezetési szokásokra, miközben társadalmi előnyöket kínál, többek között a közúti balesetek számának, az utazási időnek, az üzemanyag-fogyasztásnak és a parkolási igénynek a mérséklése révén. Az elmúlt évtizedben a számítógépes rendszerek fejlődése lehetővé tette az automatizált működés beépítését a közlekedésbe, amely korábban kizárólagosan emberi tevékenységnek számított. Ennek eredményeként több vállalat olyan önvezető járműveket fejlesztett ki, amelyek képesek a meglévő úthálózaton önálló üzemmódban közlekedni, alkalmazkodva a különböző úttípusokhoz és környezeti feltételekhez [1].

Az önvezető járművek különféle technológiák, mint például a szabályozás, a mesterséges intelligencia és a gépi-látás integrációjának eredményeként jöttek létre. Fejlesztésük a számítástechnika, a mintafelismerés és az intelligens vezérlőrendszerek területén elért fejlődésre épül.

Az autonóm járművek működését négy alapvető komponens határozza meg: a környezetérzékelés, az útvonaltervezés, a navigáció és a járművezérlés. A pontos döntéshozatal érdekében a járműnek képesnek kell lennie környezete észlelésére. A környezetészlelés meghatározó módszerei közé tartozik a lézeres, a vizuális és a radar alapú. A vizuális érzékelés kulcsfontosságú az autonóm közlekedésben, amelyet a gépi-látás és a gépi-tanuló modellek egészítenek ki olyan feladatokban, mint a lokalizáció, a navigáció és a mozgásdetektálás.

A jármű sávhoz viszonyított pozíciójának meghatározása alapvető követelmény az autonóm közlekedési rendszerek forgalomba történő integrálásához. Az autonóm járművek kutatási területén több sávérzékelési technika került kidolgozásra, amelyek két fő kategóriába sorolhatók: a geometriai modellezés és gépi-látás ötvözésén alapuló módszerek, valamint a mesterséges intelligencia alapú megközelítések [2,3]. Mindkét megközelítésnél a fő cél, hogy a sávok detektálását követően, a modell határozza meg a jármű kerekeinek elfordulás szögét, amely biztosítja a jármű útsávon belül tartását.

Az elfordulás szög meghatározásán túl a járművezérlés is egy fontos komponens, amelynek feladata a sebesség és a kormányzás szabályozására irányul. A vezérlés központi funkciója a jármű aktuális állapotának érzékelése, valamint olyan vezérlési stratégia kidolgozása, amely meghatározza a sebességet és az irányt, majd ezeket a jármű vezérlőrendszere számára továbbítja [4]. A járművezérlés során számos megközelítés alkalmazható. Habár számos korszerű szabályozási módszer áll rendelkezésre, a PID-vezérlés egyszerű struktúrája, megbízható teljesítménye, robusztus felépítése és egyszerű implementációja miatt továbbra is a legelterjedtebb megoldás [5].

Jelen tanulmány a geometriai modellezés és gépi-látás ötvözésén alapuló módszerekre fókuszál. A tanulmányban egy javított verzióját mutatjuk be egy korábban publikált sávkövető algoritmusnak [6]. Az algoritmus korábbi verziója egy költségkímélő, könnyen implementálható megoldást kínált zárt térben kialakított pályán, ahol a kormányzási szög geometriai modellezéssel lett meghatározva. A jelen tanulmányban ismertetett módszer két fő pontban módosította a korábbi verziót: dinamikus ROI-t (Region of Interest) használ a statikus helyett és az aktuális elfordulás szög meghatározásához PID szabályozót alkalmaz a korábbi súlyozott átlagolás helyett.

2. ALKALMAZOTT MÓDSZEREK

2.1. Képfeldolgozás

A képfeldolgozás célja a sávvonalak detektálása, valamint a sávok metszéspontja koordinátáinak meghatározása a kormányzási szög kiszámításához. A képfeldolgozás folyamata a következő lépésekre bontható: ROI kinyerés, Gaussian szűrő, Sobel éldetektálás és Hough-transzformáció.

A képkockák felbontása akár 3280×2464 képpont is lehet, azonban a felbontás csökkentése javasolt, mivel a sávfelismerés eredményei lényegében változatlanok, ha a képfelbontás egy bizonyos küszöbérték fölött van [7]. Tapasztalataink szerint ajánlott 160×120 képpontos felbontással dolgozni, mivel a sávvonalak még alacsonyabb felbontás mellett is jól láthatóak, mindemellett a kép feldolgozásának ideje drasztikusan csökken. A legtöbb esetben nincs is szükség a teljes képterületre és annak csak egy részével kell foglalkoznunk, amely a sávok detektálása szempontjából kritikus fontosságú. Erre a képterületre hivatkozunk ROI-ként. A referenciaként használt algoritmus [6] fix méretű ROI-val működik, míg a [7] tanulmányban bemutatott módszer adaptív ROI-t alkalmaz, amely a sávok metszéspontja alapján kerül meghatározásra. A metszéspont fölötti terület eltávolításra kerül, hogy a feldolgozás közvetlenül a jármű előtt lévő útszakaszra fókuszáljon. Ezekkel szemben, mi egy új megközelítést vezetünk be, a „Dinamikus ROI”-t, amely a kormányzási szögtől függ. Ennek az ötlete azon alapszik, hogy egyenes úton, a számított kormányzási szög relatíve kicsi, nulla körül szóródó érték. Ilyen esetben a ROI felfelé tolódik, hogy a sávot „távolabb tekintve” vizsgálja, lehetővé téve a rendszer gyorsabb reagálását kanyarokban. Amikor a kormányzási szög egy előre meghatározott küszöbértéket meghalad, jelezve egy kanyart, a ROI lefelé mozog, hogy a sávvonalakra összpontosítson, javítva a detektálás pontosságát és biztosítva a kormányzási szög precízebb meghatározását. Miután a kormányzási szög csökken és a jármű visszatér egyenes pályára, a ROI ismét felfelé mozog.

Az általunk javasolt algoritmusnál, a ROI mérete 160×60 képpont, ami egyenes szakaszon 250 ms alatt 20 képponttal mozog felfelé, míg kanyarokban lefelé tolódik. A kormányzási szög alapján történő ROI-módosítás során a kamera túlzott mozgásának elkerülése érdekében hisztérezis-ablakot alkalmazunk. A ROI felfelé történő eltolásához szükséges minimális szögérték a jármű maximális kanyarodási szögének felére van állítva. Az általunk használt tesztjármű esetében a maximális kanyarodási szög 30° , így az aktiválási szög 15° . A ROI fent marad, ha a kormányzási szög az aktiválási szög mínusz 8.5° alatt van, és lefelé

mozog, ha a kormányzási szög az aktiválási szöget több mint 10° -kal meghaladja. A Dinamikus ROI viselkedését az 1. ábra szemlélteti.

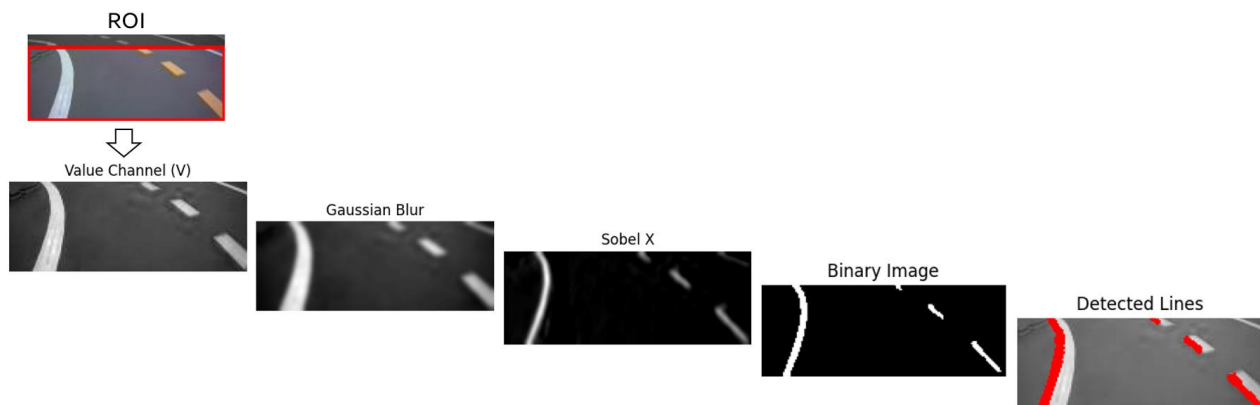


1. ábra. A dinamikus ROI működésének szemléltetése az elfordulási szög alapján

A kép előkészítésének következő lépése az, hogy az RGB (red, green, blue) színtérből átváltunk HSV (hu, saturation, value) színtérbe. Az RGB-vel ellentétben, amely erősen korrelált színtér, és érzékeny a megvilágításra, a HSV színtér H és S csatornái kevésbé befolyásolhatók a fényviszonyok által, így az emberi színészleléshez is jobban illeszkedik [8]. A csatornák közül különös figyelmet kap a V-csatorna, ami a kép szürkeárnyaltos reprezentációját adja, mivel a további képfeldolgozási lépésekhez ezt használjuk fel. A V-csatornára fókuszálva, a következő lépés a képben jelenlévő zaj csökkentése. A Gauss-zaj a leggyakoribb zajtípus a képfeldolgozásban [9]. A zaj jelenléte – különösen az éldetektálás során – ronthatja a teljesítményt, mivel a színintenzitás ingadozások miatt hamis élek jelenhetnek meg. Ebből adódóan, szükség van a kép simítására, ami fontos szerepet játszik az élek és vonalak detektálásában [10].

A kép simítását követi az élek detektálása Sobel operátorral, amely azon a feltételezésen alapul, hogy az élek ott fordulnak elő, ahol az intenzitásfüggvény folytonossága megszakad, vagy ahol az intenzitás gradiensenek meredeksége nagyon nagy. Erre alapozva, ez a szűrő egy gradiensvektort definiál, amelynek komponensei azt mérik, hogy a pixelek értékei milyen gyorsan változnak [11].

Végül a sávdetektálás utolsó lépéseként Hough-transzformációt alkalmazunk, amin keresztül az út sávjait egyenesekkel modellezzük. A Hough-transzformációt hatékony eszköz az egyenesek detektálására még zajjal terhelt képen is, ezért ezt az eljárást széles körben alkalmazzák ebben a témakörben [12]. A sávok modellezését követően már minden információ adott az elfordulási szög kiszámításához, aminek a folyamata a javított algoritmusban is megegyezik a korábban használttal [6]. Szemléltetésként a 2. ábra mutatja be a képfeldolgozás fő lépéseit, amelyet az ROI-ra alkalmazunk.



2. ábra. A kamerától kapott kép feldolgozásának fő lépései

2.2. Szabályozás

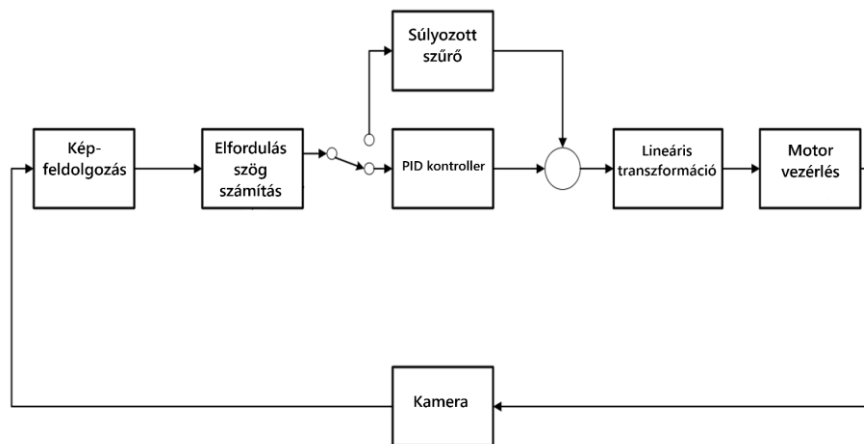
A sávok detektálása és az elfordulási szög kiszámítását követően, az algoritmus utolsó fázisában figyelmet kell fordítani a felmerülő pontatlan elfordulási szögeértékek kezelésére és a jármű egyenletes mozgására. A korábbi algoritmusban ezt egy súlyozott átlagoló szűrő kezelte, amely az aktuális és a korábbi elfordulási szögeértékek súlyozott átlagaként állította be a tényleges elfordulási szöget [6]. A továbbfejlesztett algoritmusban ezt egy széles körben használt PID (proportional, integral, derivative) szabályozóval helyettesítettük. A szabályozó, a rendszer tényleges kimenetét összehasonlítja a referencia bemenettel (beállított értékkel), majd kiszámítja a kettő közötti eltérést. Ezen eltérés alapján olyan vezérlő

jelet generál, amelynek célja a hiba minimalizálása, lehetőség szerint annak teljes megszüntetése, vagy legalábbis egy rendkívül alacsony tartományra korlátozása [13,14].

3. EREDMÉNYEK

A korábbi és a javított sávkövető módszerek objektív összehasonlításához egy modellautó használtunk tesztjárműként, amelynek egy NVIDIA Jetson Nano volt a központi egysége. A rendszerhez csatlakoztatott IMX219 kamera a jármű elején került elhelyezésre, amely 20 Hz mintavételi frekvenciával szolgáltat képkockákat a Jetson Nano számára. A vizsgálat során a jármű sebességét állandó értéken tartottuk, hogy a vizsgálat a jármű mozgásának kinematikai jellemzőire összpontosíthatson.

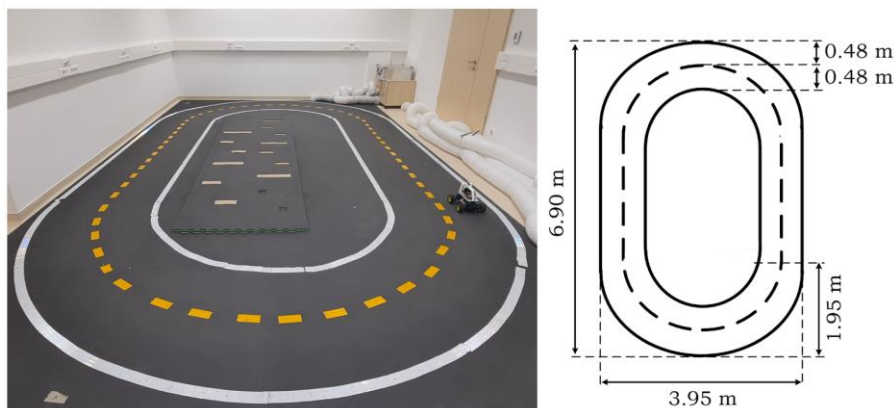
Tudomásunk szerint, jelenleg nincs általánosan elfogadott metrika a sávkövető vagy önvezető módszerek objektív összehasonlítására. Ezért egy saját metrikát állítottunk fel és kidolgoztunk egy platformot, amelynek a részletes blokkvázlatát mutatja be a 3. ábra.



3. ábra. A sávkövető algoritmusok összehasonlításához felállított rendszer blokkdiagramja

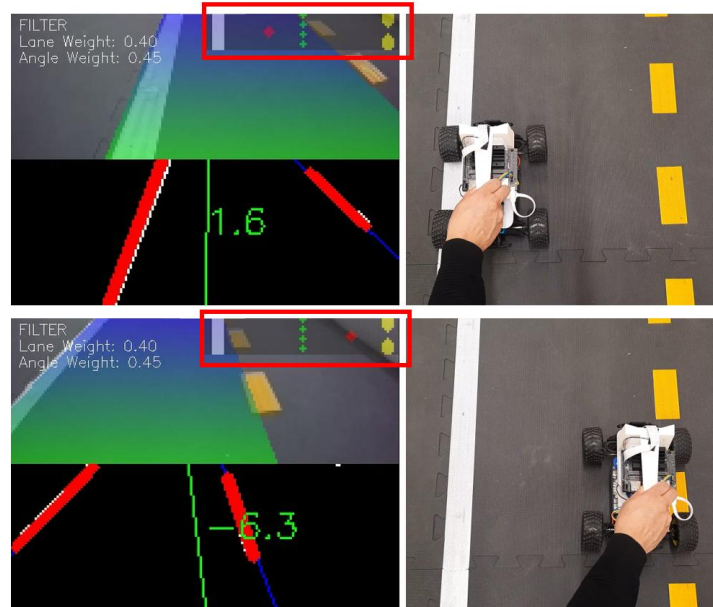
Az ábrán a képfeldolgozás blokk a kamera nyers képkockáit veszi át, és alkalmazza azokat a módszereket, amelyek a 2.1. alfejezetben kerültek ismertetésre. A képfeldolgozás eredményeként az elfordulás szög számításáért felelős blokk számára előáll két halmaznyi (jobb és bal) lehetséges sávvonal, amelyek végpontjaikkal vannak definiálva. Ezen a ponton a lehetséges sávvonalak halmazát vesszük alapul, és ebből állítjuk elő a pillanatnyi elfordulási szöget. A kapott szögérték továbbításra kerül a súlyozott szűrőhöz, vagy a PID szabályozóhoz. A vezérlési folyamat következő lépéseként egy egyszerű transzformációt, alkalmazunk, amely a szögértéket egy előre meghatározott tartományra képezi le. Ennek eredményeként egy vezérlőjelt kapunk és ezzel állítható be a tesztjárművön elhelyezett PCA9685 modullal generált PWM jel kitöltési szélessége. A PWM jel feladata a tesztjármű szervomotorjának működtetése, amely mechanikai úton határozza meg a jármű kerekeinek elfordulási szögét.

Az algoritmus hatékonyság vizsgálatát egy ovális alakú pályán végeztük, amely a 4. ábrán látható. A kísérletek során a megvilágítási viszonyokat szabályozott és állandó módon tartottuk fenn. A pálya fekete felületének köszönhetően nem jelentkeznek olyan fényvisszaverődések, amelyek befolyásolási tényezőként jelennek meg.



4. ábra. Az algoritmusok hatékonyság vizsgálatához használt tesztpálya és annak méretei

A sávkövető algoritmusok összehasonlítása a jármű, sávon belüli pozíciójának becslésén alapult, amely a detektált sávvonalakat veszi figyelembe. A módszer lényege, hogy a kép alján kiszámítjuk a sávvonalak közötti felezőpontot és ez adja az egyik referenciapontot. A másik referencia pont a képkocka alsó középpontja. A két pont között vett átlagos négyzetes hibát (MSE) használtuk metrikaként arra, hogy megbecsüljük az „ideális” pozíciótól (a sáv közepe) való eltérést. Ennek a megoldásnak az a legfőbb előnye, hogy nem igényel további hardverelemeket. Az algoritmus működésének monitorozása érdekében létrehoztunk egy felületet, amin nyomon lehet követni az aktuálisan kiszámított szögelfordulási értéket és a jármű pozíciójának becsült helyzetét a sávon belül (5. ábra).



5. ábra. A tesztjármű pozícióbecslésének vizsgálata az ehhez kidolgozott felületen

A sávkövető algoritmusok összehasonlítását követően a korábbi, súlyozott átlagolót és a fix ROI pozíciót használó algoritmus estében az MSE érték 169.21 volt, míg a javított dinamikus ROI és a PID szabályozót magába foglaló javított algoritmusnál az MSE érték 96.72. Az eredmények alapján kijelenthetjük, hogy a jármű sávkövetési pontossága a PID szabályozóra épülő algoritmus esetében hatékonyabb a korábbihoz képest, mivel a hibaarány megközelítőleg 57%-al csökkent.

A további vizsgálatok során növeltük a jármű sebességét és ismét elvégeztük az összehasonlítást. Arra kerestük a választ, hogy a két algoritmusnak az egymáshoz viszonyított hiba arányát mennyire befolyásolja a sebesség változása. Ahogyan az várható volt, a megnövelt sebesség hatására mindkét algoritmusnál megnövekedett az MSE érték a korábbihoz képest. Ekkor a javított sávkövető módszer MSE értéke 126.39 volt, míg a korábbi verzió hibaértéke 233.78. Annak ellenére, hogy a hibaérték növekedett, a tesztjármű képes volt mindkét algoritmussal, önvezető módban végig haladni a pályán. Továbbá, a megnövelt sebességnél is a javított módszer közel 54%-kal alacsonyabb hibával működött, a korábbi verzióhoz képest. A sávkövetés demonstrálásához videót is készítettünk, ami a következő linken érhető el: <https://youtu.be/tMCL0sXioSI>.

4. KONKLÚZIÓ

A tanulmány egy önvezető járművekhez alkalmazható, geometriai és képfeldolgozási eljárásokon alapuló sávkövető algoritmust mutat be. Az algoritmus célja, a jármű útpályához viszonyított pozíciójának pontos meghatározása és a sávvonalak megbízható követése, amely az autonóm közlekedés egyik alapfeltétele. Az algoritmus két fő újítást tartalmaz egy korábban publikált eljáráshoz képest: a kormányzási szögtől függően dinamikusan változó ROI alkalmazását, valamint a kormányzás szabályozásához a súlyozott átlag helyett PID-szabályozó integrálását.

A tanulmányban ajánlott algoritmus hatékonyságának szemléltetése modellautóval, szabályozott beltéri környezetben történtek. Az összehasonlító vizsgálatok eredményei szerint a továbbfejlesztett módszer

jobb teljesítményt nyújtott. Az átlagos négyzetes hiba (MSE) közel 76%-kal csökkent a korábbi algoritmushoz képest, magasabb sebességnél pedig a hibacsökkenés mértéke elérte a 85%-ot.

A kutatás igazolta, hogy a dinamikus ROI és a PID-szabályozás kombinációja stabilabb és pontosabb sávkövetést tesz lehetővé, amely kizárólag szoftveralapú megközelítéssel, további hardverelemek nélkül is megvalósítható.

IRODALMI HIVATKOZÁSOK

- [1] Fagnant D.J., Kockelman K. *Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations*. Transp. Res. Part A. Policy Pract., 2015, 77, 167–181.
- [2] Zakaria, N.J., Shapiai M.I., Ghani R.A., Yassin M.N.M., Ibrahim M.Z., Wahid N. *Lane detection in autonomous vehicles: a systematic review*. IEEE Access, 2023, 11, 3729–3765.
- [3] Saleem H., Riaz F., Mostarda L., Niazi M.A., Rafiq A., Saeed S. *Steering angle prediction techniques for autonomous ground vehicles: a review*. IEEE Access, 2021, 9, 78567–78585.
- [4] Zhao J., Liang B., Chen Q. *The key technology toward the self-driving car*. Int. J. Intell. Unmanned Syst., 2018, 6(1), 2–20.
- [5] Farag W. *Complex trajectory tracking using PID control for autonomous driving*. Int. J. Intell. Transp. Syst. Res., 2019, 18(2), 356–366.
- [6] Suto J. *Real-time lane line tracking algorithm to mini vehicles*. Transp. Telecommun. J., 2021, 22(4), 461–470.
- [7] Lee C., Moon J.H. *Robust lane detection and tracking for real-time applications*. IEEE Trans. Intell. Transp. Syst., 2018, 19(12), 4043–4048.
- [8] Jackl B., Metz Y., Schlegel U., Keim D.A., Fischer M.T. *Leveraging color channel independence for improved unsupervised object detection*. arXiv preprint, 2024, arXiv:2412.15150.
- [9] Bovik A. *The essential guide to image processing*. Academic Press, USA, 2009.
- [10] Basu M. *Gaussian-based edge-detection methods—a survey*. IEEE Trans. Syst. Man. Cybern. Part C Appl. Rev., 2002, 32(3), 252–260.
- [11] Vincent O.R., Folorunso O.A. *A descriptive algorithm for Sobel image edge detection*. Proceedings of Informing Science & IT Education Conference (InSITE), 2009, 40, 93–107.
- [12] Fernandes L.A.F., Oliveira M.M.O. *Real-time line detection through an improved Hough transform voting scheme*. Pattern Recognition, 2008, 41(1), 299–314.
- [13] Uzunovic T., Zunic E., Badnjevic A., Miokovic I., Konjicija S. *Implementation of digital PID controller*. The 33rd International Convention MIPRO, Opatija, Croatia, 2010, 1357–1361.
- [14] Kocur M., Kozak S., Dvorscak B. *Design and implementation of FPGA - digital based PID controller*. Proceedings of the 15th International Carpathian Control Conference (ICCC), Velke Karlovice, Czech Republic, 2014, 233–236.