



Osaamista
ja oivallusta
tulevaisuuden
tekemiseen

Joonas Kouhia

Autonomisen ajoneuvon kaistantunnistus- ohjelmiston testaus

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Ajoneuvotekniikka

Insinöörityö

6.5.2020

Tekijä Otsikko	Joonas Kouhia Autonomisen ajoneuvon kaistantunnistusohjelmiston testaus
Sivumäärä Aika	26 sivua 6.5.2020
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Ajoneuvotekniikka
Ammatillinen pääaine	Ajoneuvosuunnittelu
Ohjaajat	Lehtori Pasi Oikarinen
<p>Tämän insinööriyön aiheena oli tutkia autonomisen ajoneuvojärjestelmän havainnointimenetelmiä ja komponentteja sekä tutustua autonomisten ajoneuvojen simulointiin tarkoitettuihin ohjelmistoihin.</p> <p>Työssä tarkasteltiin autonomisen ajamisen historiaa, autonomian eri tasoja ja antureita, joita ajoneuvo käyttää ympäristön havaitsemiseen ja itsensä paikallistamiseen. Lisäksi tutustuttiin autonomisten ajoneuvojen simulointiin tarkoitettuihin ohjelmistoihin ja lopuksi testattiin kameraan perustuvaa havainnointiohjelmistoa, joka tunnistaa kamerasensorin datasta ajokaistat ja muut tienkäyttäjät sekä arvioi näiden etäisyydet. Työssä perehdyttiin myös siihen, mitä eri tekniikoita ja vaiheita kameraan perustuva havainnointijärjestelmän vaatii ja testattiin ohjelmistoa eri olosuhteissa kuvattuun ajodataan.</p> <p>Insinööriyö antaa katsauksen autonomisiin ajoneuvoihin, sen antureihin ja niiden toimintaan. Tehty ohjelmiston testaus havainnollisesti, millaisia haasteita autonomisen ajoneuvon havainnointijärjestelmä kohtaa. Vaihtelevat sääolosuhteet, tiemerkitöiden ja teiden kunto vaikeuttavat ajoradan kaistamerkitöiden tunnistusta.</p>	
Avainsanat	Autonominen ajaminen, älykäs liikenne

Author Title	Joonas Kouhia Testing of Autonomous Vehicle Perception Software
Number of Pages Date	26 pages 6 May 2020
Degree	Bachelor of Engineering
Degree Programme	Automotive Engineering
Professional Major	Automotive Design
Instructors	Pasi Oikarinen, Lecturer
<p>The objective of this Bachelor's thesis was to study the perception methods of the autonomous vehicles, the components it uses and study what kind of simulation software there are for simulating and testing autonomous vehicles.</p> <p>The thesis examined the history of autonomous driving, the different levels of autonomy and what type of sensors the vehicle uses to perceive the environment and locate itself. Different software for simulating autonomous vehicles are discussed as well as, tested camera-based perception software that use camera data to identify lane markings, other road users and estimate their distances. The thesis examines the different techniques and steps required by a camera-based perception system and tested the software in different driving data. The thesis gives an overview of autonomous vehicles and how they use sensors to perceive the surrounding world. It was found out that there are several software available for autonomous driving simulation purposes, e.g. o Matlab's automated driving toolbox. The camera-based perception software test illustrated what kind of challenges the perception system of autonomous vehicles faces. Varying weather conditions, state of the lane markings and road conditions, all affect how well the system detects the lane markings and other vehicles.</p>	
Keywords	Autonomous driving, intelligent cars

Sisällys

Lyhenteet

1	Johdanto	1
2	Autonominen ajoneuvo	2
2.1	Autonomisen ajamisen historia	2
2.2	Autonomisen ajamisen tasot	3
2.3	Anturit	5
2.3.1	Teräväpiirtokartta	6
2.3.2	GPS	7
2.3.3	Inertiaalinen mittauslaite (IMU)	7
2.3.4	Tutka	8
2.3.5	Kaikuluotain	9
2.3.6	Lasertutka	10
2.3.7	Kamera	11
3	Ohjelmistot	12
3.1	NVIDIA Drive Constellation	12
3.2	CARLA Simulator	13
3.3	AirSim	13
3.4	Matlab Automated Driving Toolbox	14
3.4.1	Ground truth labeler	14
3.4.2	Driving scenario designer	15
4	Testaus	16
4.1	Kameran kalibrointi	17
4.2	Kuvan prosessointi	18
4.3	Kaistamerkintöjen tunnistus	19
4.4	Ajoneuvojen tunnistus	21
4.5	Ohjelmiston testaus	22
5	Yhteenveto	25

Lyhenteet

RANSAC	Random sample consensus
SAE	Society of Automotive Engineers
GPS	Global Positioning System
SLAM	Simultaneous Localization and Mapping
IMU	Inertialinen mittauslaite
HIL	Hardware-in-the-loop
ACF	Aggregate Channel Features

1 Johdanto

Autonomiset ajoneuvot lupaavat liikenteeseen lukuisia parannuksia: autonomisten ajoneuvojen ennakoiva ajotyyli parantaa liikennevirtaa, vähentää päästöjä sekä niiden älykkäät turvajärjestelmät parantavat liikenneturvallisuutta. Jotta nämä ajoneuvot voivat turvallisesti toimia päivittäisessä liikenteessä, on lukuisia havaitsemisen, navigoinnin ja hallinnan ongelmia ratkaistava. Näitä ongelmia yrittävät ratkaista niin isot ajoneuvovalmistajat kuin pienemmät teknologiayhtiöt.

Tässä työssä tarkastellaan autonomisen ajoneuvojärjestelmän peruskomponentteja ja tutkitaan autonomiseen ajamiseen tarkoitettuja simulointiohjelmistoja ja testataan kamerapohjaisen havainnointijärjestelmän toimintaa. Autonomisiin ajoneuvoihin liittyy monia eri tekniikan aloja, sillä autonomisessa ajoneuvossa yhdistyvät niin mekaaniset kuin sähköiset järjestelmät. Tietokoneiden laskentatehon kasvaessa on tekoälyn ja syväoppimisen käytöstä tullut olennainen osa autonomisten ajoneuvojen havainnointijärjestelmää. Tekoälyä tai syväoppimista ei tässä työssä kuitenkaan käsitellä.

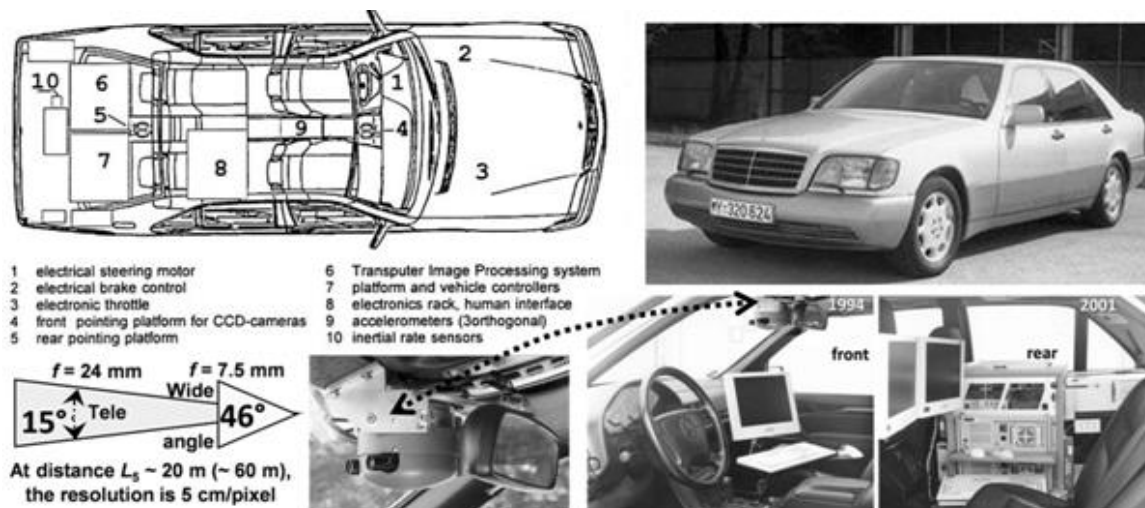
Työssä kuvataan ensin autonomisten ajoneuvojen historiaa, autonomian eri tasoja ja antureita, minkä jälkeen tutkitaan eri autonomisten ajoneuvojen simulointiin tarkoitettuja ohjelmistoja ja niiden ominaisuuksia. Lisäksi tutustutaan syvemmin yhteen simulointiohjelmistoon ja esitellään sen testausta. Lopuksi kootaan työn ja testauksen tulokset.

2 Autonominen ajoneuvo

Autonominen ajoneuvo on ajoneuvo, joka kykenee suoriutumaan ajotehtävästä ilman kuljettajaa ja ilman yhteyttä muihin ajoneuvoihin tai infrastruktuuriin [1, s. 2]. Autonominen ajoneuvo käyttää antureita ympäristön havainnointiin ja itsensä paikallistamiseen sekä tietokonetta reitinsuunnitteluun ja ohjauskomentoihin.

2.1 Autonominen ajamisen historia

Ensimmäisiä autonominen ajamisen kokeiluja tehtiin jo vuonna 1925, jolloin Houdina Radio Control kehittämä radio-ohjattu auto ajoi kuljettajattomana New Yorkin kaduilla. Sen jälkeen on esitelty erilaisia versioita niin sähköistetyistä moottoriteistä aina nykypäivän antureilla suunnistaviin ajoneuvoihin. Ensimmäinen antureihin ja konenäköön turvautuva autonominen auto esiteltiin vuonna 1994. Ajoneuvovalmistaja Daimler-Benz kehittämä ajoneuvo VaMP (kuva 1) käytti ympäristön havainnointiin neljää kameraa ja anturidatan käsittelyyn ja päätöksentekoon senaikaista super-tietokonetta. Se pystyi ajamaan liikenteessä pitkiä matkoja ilman ihmisen puuttumista ajoneuvon hallintaan ja järjestelmä osasi tunnistaa muut ajoneuvot, liikennemerkkit ja tarvittaessa pystyi ohittamaan muita tienkäyttäjiä. [2, s. 45.]



Kuva 1. Daimler-Benzin kehittämä autonominen ajoneuvo VaMP [2, s. 12].

Nykypäivän tieliikenteeseen valmistetuissa autoissa on jo monia kuljettajaa avustavia järjestelmiä, jotka osaavat hoitaa osan ajotehtävästä, mutta täydelliseen autonomiaan ei vielä olla päästy. Nykyissä järjestelmissä on kuljettaja vastuussa ympäristön havainnoinnista, ja ne edellyttävät kuljettajan olevan valmiina ottamaan ajoneuvo takaisin haltuun järjestelmän sitä vaatiessa. Tällä hetkellä suurin osa suurimmista autoteollisuuden valmistajista samoin kuin muut teknologiayhtiöt, kuten Google, Uber ja monet startup-yritykset, kehittävät omia autonomisia järjestelmiään. Vaikka näiden suunnittelukonseptit eroavat toisistaan, on periaate kaikissa sama. Kaikki järjestelmät käyttävät erityyppisiä antureita ympäristön havainnointiin ja ajoneuvon paikallistamiseen, ohjelmistoa datan käsittelyyn, tietokonetta päätöksentekoon ja reitin suunnitteluun sekä ajoneuvon toimilaitteita ajoneuvon hallintaan.

2.2 Autonomisen ajamisen tasot

Autonomiset järjestelmät voidaan jakaa kuuteen eri automaation tasoon (kuva 2). Tasot ovat SAE:n (Society of Automotive Engineers) määrittelemiä, ja ne ovat yleisesti käytössä. Tasolla nolla ei ole automatisaatiota ollenkaan, ja tasolla viisi on ajoneuvo täysin automatisoitu. SAE-standardin mukaan kuljettaja on vastuussa ympäristön havainnoinnista tasolle kaksi asti ja ajoneuvon hallinnasta tasolle kolme asti. [3, s. 19.]

SAE level	Name	Narrative Definition	Execution of Steering and Acceleration/Deceleration	Monitoring of Driving Environment	Fallback Performance of Dynamic Driving Task	System Capability (Driving Modes)
Human driver monitors the driving environment						
0	No Automation	the full-time performance by the <i>human driver</i> of all aspects of the <i>dynamic driving task</i> , even when enhanced by warning or intervention systems	Human driver	Human driver	Human driver	n/a
1	Driver Assistance	the <i>driving mode</i> -specific execution by a driver assistance system of either steering or acceleration/deceleration using information about the driving environment and with the expectation that the <i>human driver</i> perform all remaining aspects of the <i>dynamic driving task</i>	Human driver and system	Human driver	Human driver	Some driving modes
2	Partial Automation	the <i>driving mode</i> -specific execution by one or more driver assistance systems of both steering and acceleration/deceleration using information about the driving environment and with the expectation that the <i>human driver</i> perform all remaining aspects of the <i>dynamic driving task</i>	System	Human driver	Human driver	Some driving modes
Automated driving system ("system") monitors the driving environment						
3	Conditional Automation	the <i>driving mode</i> -specific performance by an <i>automated driving system</i> of all aspects of the <i>dynamic driving task</i> with the expectation that the <i>human driver</i> will respond appropriately to a <i>request to intervene</i>	System	System	Human driver	Some driving modes
4	High Automation	the <i>driving mode</i> -specific performance by an <i>automated driving system</i> of all aspects of the <i>dynamic driving task</i> , even if a <i>human driver</i> does not respond appropriately to a <i>request to intervene</i>	System	System	System	Some driving modes
5	Full Automation	the full-time performance by an <i>automated driving system</i> of all aspects of the <i>dynamic driving task</i> under all roadway and environmental conditions that can be managed by a <i>human driver</i>	System	System	System	All driving modes

Kuva 2. Autonomisten ajoneuvojen eri tasot [3, s. 19].

Ei automaatiota (SAE Taso 0)

Kuljettajan on tarkkailtava ympäristöä, hallita ajoneuvoa ja on vastuussa kaikista ajotehtävistä. Ajoneuvossa voi olla kuljettajaa varoittavia järjestelmiä kuten kuolleen kulman varoitin tai kaistalta suistumisvaroitin, mutta nämä eivät puutu ajoneuvon hallintaan.

Kuljettajan avustimet (SAE Taso 1)

Järjestelmä pystyy säädellä ajoneuvon nopeutta tai hallita ajoneuvon ohjausta, mutta järjestelmä ei pysty hallitsemaan molempia samanaikaisesti. Tähän tasoon kuuluvat erilaiset kaistavahdit ja mukautuvat vakionopeudensäätimet, jotka voivat puuttua ajoneuvon hallintaan.

Osittainen automaatio (SAE Taso 2)

Järjestelmä pystyy samanaikaisesti säädellä ajoneuvon nopeutta ja hallita ajoneuvon ohjausta. Tähän tasoon kuuluvat pysäköinti- ja ruuhka-avustimet, jotka voivat hallita ajoneuvon nopeutta ja ohjausta samanaikaisesti.

Ehdollinen automaatio (SAE Taso 3)

Järjestelmä pystyy ajamaan itsenäisesti rajatuissa olosuhteissa kuten moottoritiellä tai ruuhkassa. Tasolla kolme ei kuljettajan tarvitse enää havainnoida ympäristöä, mutta hänen täytyy olla valmiina ottamaan ajoneuvo takaisin hallintaan järjestelmän sitä pyytäessä

Korkea automaatio (SAE Taso 4)

Järjestelmä pystyy ajamaan itsenäisesti rajatuissa olosuhteissa kuten moottoritiellä tai ruuhkassa. Tasolla 4 järjestelmä voi pyytää kuljettajaa ottamaan ajoneuvon hallintaan muttei oleita sitä. Tähän tasoon kuuluvat esimerkiksi robottibussit ja itseajavat taksit.

Täysi automaatio (SAE Taso 5)

Ajoneuvon kuljettajaa ei tarvita. Järjestelmä on täysin automatisoitu ja suoriutuu kaikista ajotehtävistä itsenäisesti, olosuhteista riippumatta.

2.3 Anturit

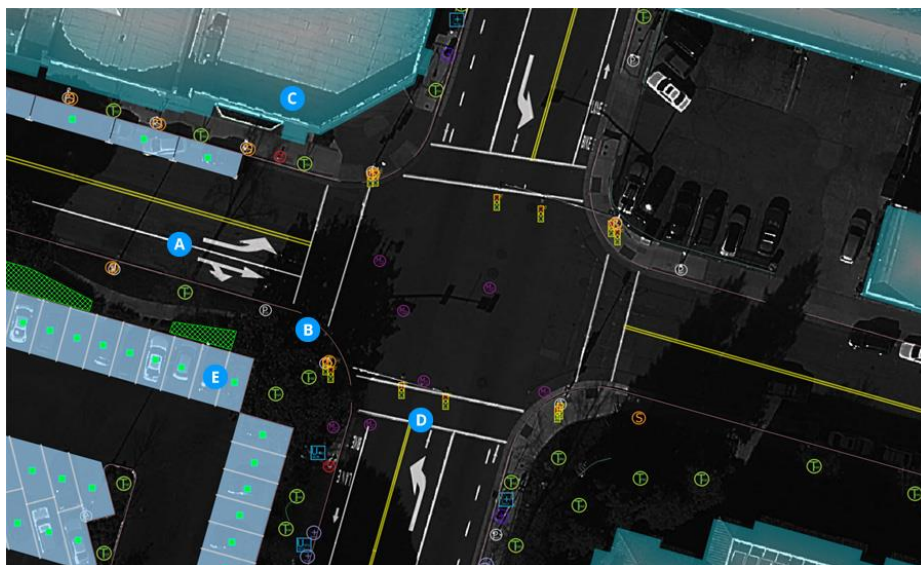
Koska autonominen ajoneuvo toimii ainakin osittain tuntemattomassa ja dynaamisessa ympäristössä, on sen samanaikaisesti rakennettava tästä ympäristöstä kartta ja paikallistettava itsensä tähän karttaan. Autonomisissa ajoneuvoissa käytetään SLAM-menetelmää (Simultaneous Localization and Mapping), joka tarkoittaa, että samaan aikaan sekä paikannetaan kohde että kartoitetaan ympäristöä. Ajoneuvon lasertutkien, kameroiden ja muiden sensorien antamasta informaati-

tiosta ratkaistaan yhtäaikaaisesti ajoneuvon suhteellinen liike sekä luodaan sensorimittauksiin pohjautuva kartta lähialueesta [4, s.1]. Jokaisella anturityypillä on vahvuutensa ja heikkoutensa, ja jotta saavutettaisiin vaikeissakin olosuhteissa toimiva havainnointijärjestelmä, on järjestelmän su-lautettava monen anturin dataa yhteen ja luotava tarkka kuva ajoneuvon ympäristöstä ja sijain-nista. Ajoneuvon on kyettävä välttämään läheisiä esineitä havaitsemalla samalla kaukana olevia esineitä. Sen on kyettävä toimimaan erilaisissa ympäristö- ja tieolosuhteissa sekä haastavissa valo- ja sääolosuhteissa. [5, s. 16]

Ympäristön havainnointiin voidaan käyttää kuvapohjaisia sensoreita, kuten monokulaari- ja ste-reokamerat. Etäisyyden arviointiin voidaan käyttää erityyppisiä tutkia kuten lasertutka. Ajoneuvon paikallistamiseen käytetään useita komponentteja paikannusvirheen minimoimiseksi; yleisimmin paikallistamiseen käytetään satelliittipaikannusta, teräväpiirtokarttoja ja inertiaalista mittauslai-tetta.

2.3.1 Teräväpiirtokartta

Nykyään navigointiin käytetään laajalti karttapalveluita kuten Google Maps. Autonomiset ajoneu-vot tarvitsevat kuitenkin huomattavasti tarkemman kartan pystyäkseen paikallistamaan ja navi-goimaan tarpeeksi tarkasti. Teräväpiirtokartta eroaa normaalista kartasta yksityiskohtaisuudel-laan: siihen on mallinnettu senttimetrien tarkkuudella tiemerkinnet, ajokaistat, liikennemerkit ja muu ympäristö (kuva 3). Kolmioimalla etäisyyksiä teräväpiirtokartan tunnettuihin kohteisiin, voi-daan ajoneuvon tarkka sijainti määrittää. Teräväpiirtokarttojen haasteina on niiden luominen, ajan-tasalla pitäminen ja suuri tallennustilan tarve. [6, s. 1.]



Kuva 3. Tyypillinen teräväpiirtokartta [3, s. 1].

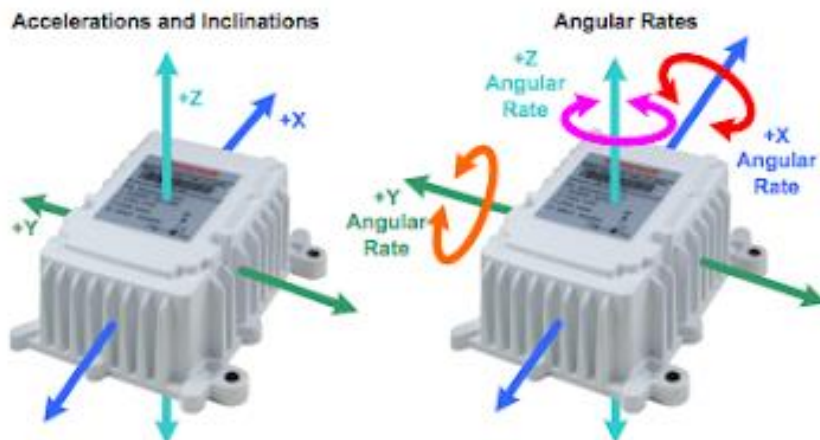
2.3.2 GPS

GPS eli Global Positioning System on maailmanlaajuinen paikallistamisjärjestelmä, jonka toiminta perustuu n. 20 200 kilometrin korkeudessa Maan kiertoradalla sijaitseviin satelliitteihin. GPS-järjestelmän toiminta perustuu tarkkaan ajansiirtoon. Jokaisessa satelliitissa on atomikello, jonka perusteella ne lähettävät maahan aikaleiman, satelliitin sijainnin ja muuta oheisdataa. Signaalista saatavan aikaleiman ja vastaanottimen sisäisen kellon antaman vastaanottoajan erotuksella voidaan se muuttaa etäisyysmittaukseksi ja paikallistaa GPS vastaanotin. [7, s. 1.] Satelliittipaikannuksella on mahdollista päästä jopa alle metrin tarkkuuksiin, mutta tämä ei ole aina mahdollista erilaisten häiriöiden vuoksi, joten autonomisessa ajoneuvokäytössä järjestelmän epätarkkuutta kompensoidaan käyttämällä satelliittipaikannuksen lisäksi muita järjestelmiä kuten inertia-mittausta ja ajoneuvon odometriaa.

2.3.3 Inertiaalinen mittauslaite (IMU)

Inertiaalinen mittauslaite (IMU) on yhdistelmä gyroskooppeja, kiihtyvyys ja voima-antureita, joka mittaa laitteeseen kohdistuvia voimia (kuva 4). Ajoneuvo käytössä tämä mahdollistaa ajoneuvon nopeuden, etenemissuunnan ja kaltevuuden seuraamisen. Inertiaalinen mittauslaite yhdistettynä

satelliittipaikannuksen ja teräväpiirtokarttojen dataan mahdollistaa ajoneuvon tarkan paikallistamisen. Esimerkiksi tunneliin ajettaessa tai jos ajoneuvo menettää yhteyden GPS:ään, pystyy järjestelmä rajoitetun ajan jatkamaan ajoneuvon paikallistamista IMU:n, odometrian ja teräväpiirtokarttojen avulla.



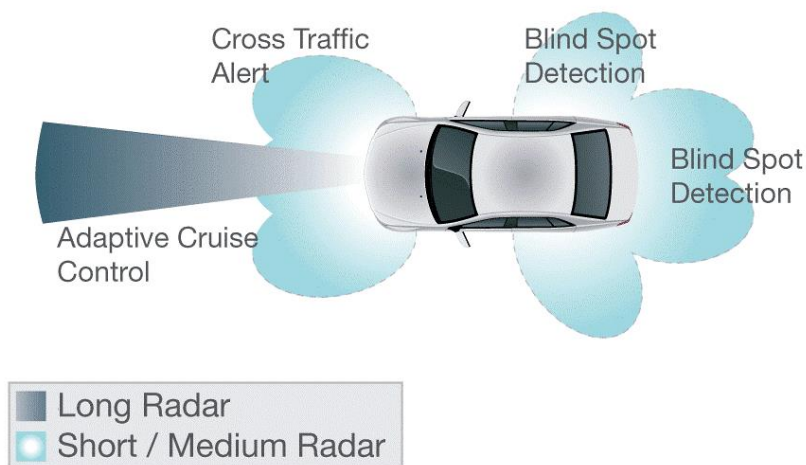
Kuva 4. Inertiaalinen mittausyksikkö ja mitattavat suureet [9, s. 3].

2.3.4 Tutka

Tutka (engl. radar) on radiotekninen mittauslaite, jonka toiminta perustuu radioaaltojen avulla tapahtuvaan ilmaisuun ja mittaamiseen. Tutkan lähetin lähettää radioaaltoja taajuusalueella 3 Hz – 3 GHz, radioaaltojen heijastukset vastaanotetaan usealla antennilla, ja niiden perusteella on mahdollista tulkita kohteen sijainti, nopeus sekä etenemissuunta. Yleisimpiä tutkan käyttökohteita ovat ilmailu, merenkulku ja maanmittaus. Tutkalla voidaan havaita kohteen nopeus hyödyntämällä Doppler-efektiä. Vertaamalla lähtevän ja tulevan elektromagneettisen taajuuden eroa voidaan laskea havaitun kohteen etenemissuunta ja nopeus. Ajoneuvokäytössä tätä tietoa voidaan käyttää apuna erottamaan, onko havaittu kohde nopeasti liikkuva ajoneuvo vai jalankulkija.

Tutka sopii hyvin autonomiseen ajoneuvokäyttöön, sillä sen pitkä kantama mahdollistaa kohteen sijainnin, nopeuden ja suunnan havaitsemisen jopa 200 metriin asti, se on edullinen sekä sen toimintakyky säilyy vaikeissakin olosuhteissa. Tutkan rajoituksena on sen huono sivuttaissuuntai-

nen resoluutio, joka vaikeuttaa kohteiden luokittelua sekä useiden kohteiden erottamista toisistaan. Nykyautoissa tutkaa käytetään mukautuvissa vakionopeudensäätimissä (kuva 5), joissa se mittaa muiden autojen sijaintia ja nopeutta, minkä perusteella järjestelmä säättää ajoneuvon nopeuden sopivaksi. [5, s. 22].



Kuva 5. Tutkien käyttö ajoneuvossa [10, s. 2].

2.3.5 Kaikuluotain

Kaikuluotain (engl. SONAR, sound navigation and ranging) on laite, joka käyttää 40–180 kHz:n ääniaaltoja kohteen havaitsemiseen ja paikallistamiseen. Kaikuluotain havaitsee kohteen sijainnin ja nopeuden perustuen pinnoista heijastuvien ääniaaltojen aikaan, taajuuteen ja muotoon. Ääniaaltojen nopeus ilmassa on noin 320 m/s, joten mittaus on hidas. Ääniaaltojen vaimenemisen vuoksi myös mittausetäisyys on lyhyt, minkä vuoksi se soveltuu käytettäväksi lähellä olevien kohteiden havainnointiin. Kaikuluotaimia käytetään ajoneuvojen parkkiavustimissa sekä kuolleen kulman varoittimissa.

2.3.6 Lasertutka

Lasertutka (LIDAR, light detection and ranging) on optinen tutka, joka käyttää laservaloa ympäristön havainnointiin ja kartoittamiseen. Lasertutka keksittiin vuonna 1960, ja Apollo 15 käytti lasertutkaa kuun pinnan kartoittamisessa vuonna 1971. Ajoneuvokäyttöön se otettiin ensimmäistä kertaa vuonna 2005 osana DARPA Challengea, autonomisten ajoneuvojen kilpailua.

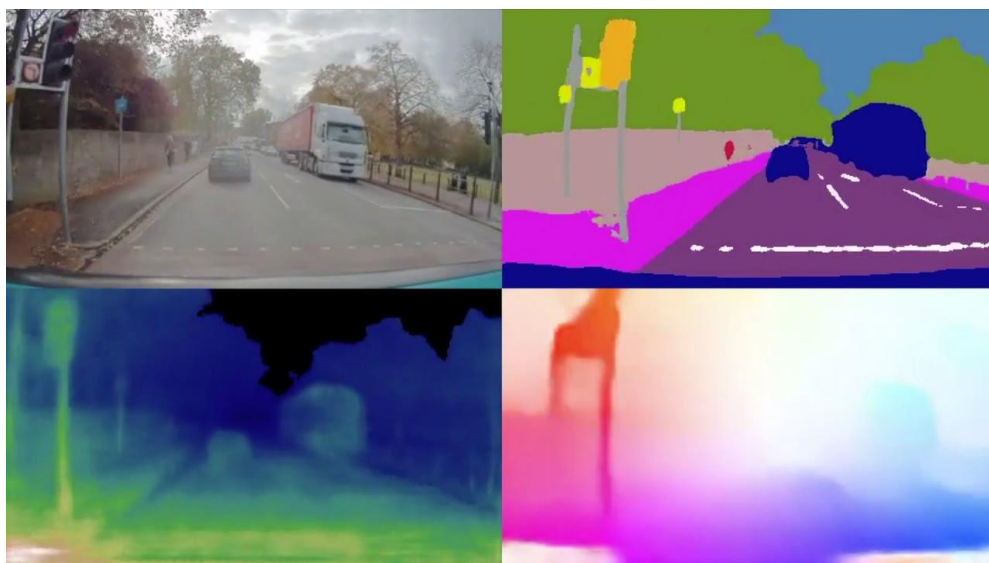
Toimintaperiaate on sama kuin perinteisen tutkan: lasertutka lähettää lyhyen laservalopulssin ja mittaa sen heijastumisen kulkuaikaa. Yleisimmin autonomisissa ajoneuvoissa käytetään mekaanista lasertutkaa, jossa käytetään pyörivää laseria tai pyöriviä peilejä, joilla laservalopulssi heijastetaan ympäristöön [5, s. 25]. Ympäristöstä heijastuneista pisteistä luodaan kolmiulotteinen pisteilvi (kuva 6), joka mahdollistaa kohteiden kartoittamisen muutamien senttimetrien tarkkuudella. Lasertutkan etuja on sen suhteellisen laaja näkökenttä: sillä voidaan saavuttaa jopa 360 asteen näkökenttä riippuen lasertutkan tyypistä. Lisäksi sillä on pitkä kantama, ja sen sisältämä etäisyystieto on tarkempi verrattuna muihin antureihin. Sen heikkouksina on heikko resoluutio ja korkea hinta. [5, s. 26.].



Kuva 6. Lasertutkan luoma kolmiulotteinen kuva [5, s, 24].

2.3.7 Kamera

Autonomisissa järjestelmissä voidaan käyttää useita erityyppisiä kameroita. Kameran toiminta perustuu kuvattavan kohteen valonsäteiden kirkkauden ja värierojen mittaamiseen. Kamerassa on valoherkkä kenno, johon kameran objektiivi kokoaa valonsäteet. Autonomisen ajoneuvon systeemissä kameraa käytetään havaitsemaan tiemerkinnot, liikennemerkkit sekä muut tienkäyttäjät. Kameroiden etuna on muihin antureihin verrattuna niiden ominaisuus hahmottaa kohteen väri, kontrasti ja tekstuuri. Yhdistelemällä useita kameroita saadaan muodostettua stereokuva, joka mahdollistaa etäisyyksien mittauksen. Kameroiden, tietokoneiden laskentatehon ja syväoppimisen kehittyessä, on kameroista tullut tärkeä komponentti autonomisiin ajoneuvoihin. Kamera datan tärkeänä etuna on se, että sitä voidaan tehokkaasti käyttää syväoppimiseen (kuva 7). Joissain ajoneuvoissa käytetään lämpökameraa, joka helpottaa elävien kohteiden havainnoinnin pimeässä. Ajoneuvon sisällä olevia kameroita voidaan käyttää kuljettajan monitorointiin ja vireystilan arviointiin. [4, s. 15; 11.]



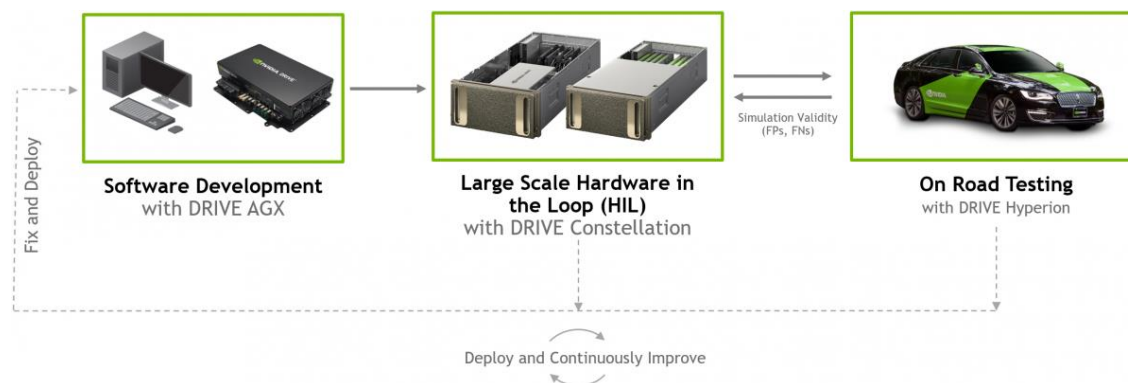
Kuva 7. Kameradata käsiteltynä erilaisilla kuvatunnistusalgoritmeilla [5, s. 38].

3 Ohjelmistot

Autonomisten ajoneuvojen algoritmien kehittäminen ja testaaminen on kallis ja aikaa vievä prosessi, ja jotta autonomisen ajoneuvon toiminnasta saataisiin luotettavaa, tarvitaan algoritmeille paljon testikilometrejä. Erilaiset simulointiohjelmistot tarjoavat mahdollisuuden analysoida ja validoida ajoneuvon toimintaa virtuaalisessa ympäristössä. Simuloinnit mahdollistavat eri liikenneskenaarioiden ja olosuhteiden testaamisen, joita olisi vaikea testata oikeassa liikenteessä. Simuloimalla voidaan testata niin järjestelmän päätöksenteko- ja suunnittelualgoritmeja kuin konenäön suoriutumista. [12] Monia simulointiohjelmistoja voidaan käyttää niin sanottuna hardware-in-the-loop (HIL) -järjestelmänä, joka tarkoittaa, että simulointijärjestelmä käyttää samaa ohjauslaitteistoa ja ohjelmistoa kuin oikea ajoneuvo. Seuraavassa osassa tutkitaan autonomisten ajoneuvojen simulointiin tarkoitettuja ohjelmistoja.

3.1 NVIDIA Drive Constellation

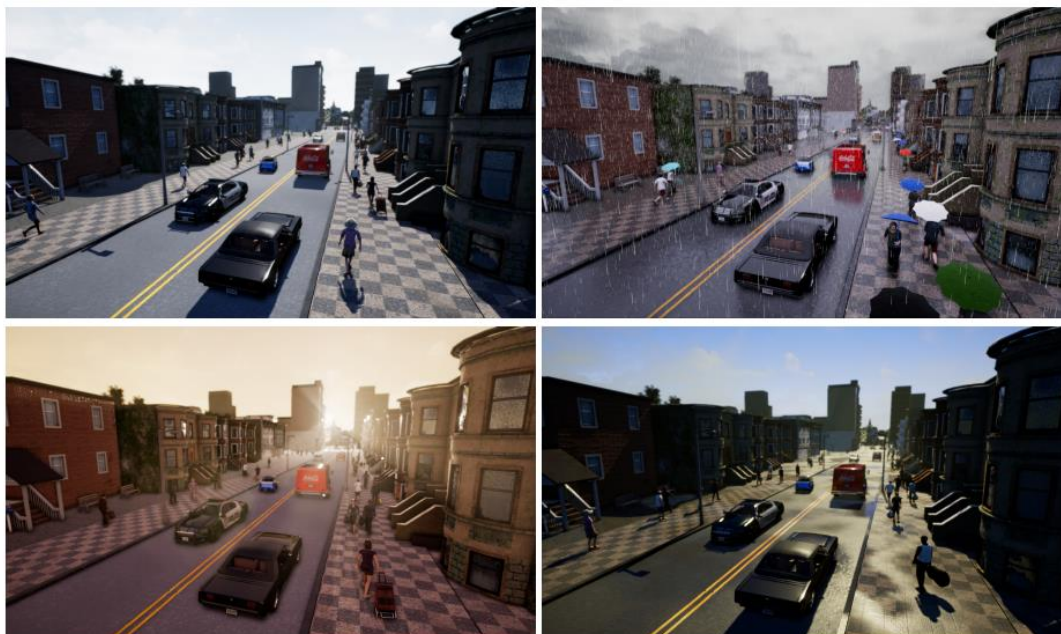
Drive Constellation on Nvidian kehittämä autonomisten ajoneuvojen testaamiseen ja validointiin tarkoitettu alusta. Sen toiminta perustuu kahteen palvelimeen. Ensimmäinen palvelin simuloi virtuaalisen ympäristön, ajoneuvot ja anturit, joista se generoi anturidataa. Simuloitu anturidata lähetetään toiselle palvelimelle, joka prosessoi datan ja lähettää ajoneuvon ohjaus päätökset takaisin simulointipalvelimelle. Näin saadaan täysi hardware-in-the-loop, joka mallintaa samanlaista järjestelmää kuin oikeassakin autonomisessa ajoneuvossa. [13]



Kuva 8. NVIDIA Drive -järjestelmä [13].

3.2 CARLA Simulator

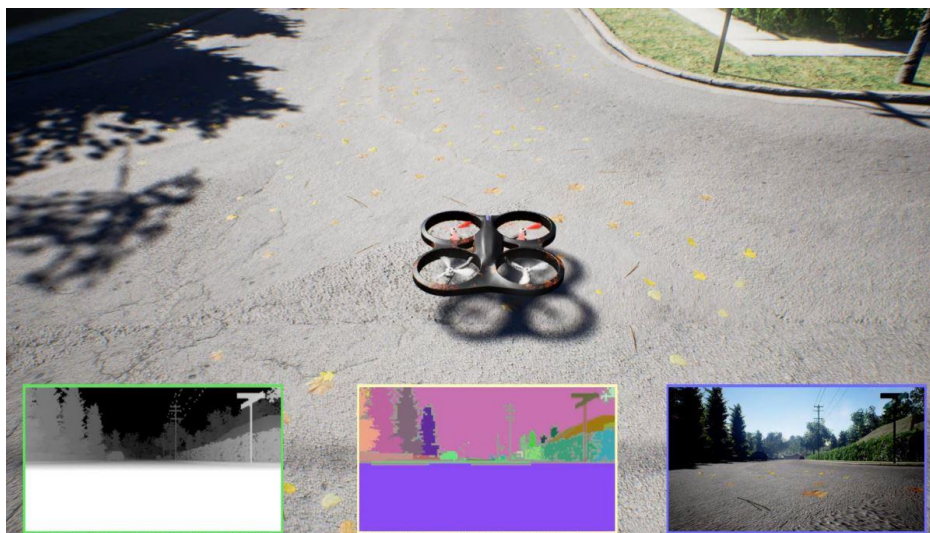
CARLA on avoimen lähdekoodin ajosimulaattori, joka on suunniteltu autonomisten algoritmien suunnitteluun, kouluttamiseen ja validointiin. Ohjelmisto käyttää ympäristön luomiseen Unreal Enginen -pelimoottoria (kuva 9). Simulaattori sisältää valmiita ympäristöjä ja skenaarioita, mutta käyttäjä voi myös luoda ympäristöjä käyttämällä työkaluja kuten RoadRunner. [15, s. 1.]



Kuva 9. Carlassa simuloitu kaupunkiympäristö eri sääolosuhteilla [15, s. 2].

3.3 AirSim

AirSim on avoimen lähdekoodin simulaattori erilaisille ajoneuvoille kuten autoille ja droneille. AirSim-simulaattorin tavoitteena on toimia alustana tekoälyalgoritmien kehittämiseksi ja tutkimiseksi. Se sisältää työkaluja opetusdatan generointiin ja konenäön kehittämiseen. Simulaattori toimii Unreal Enginen lisäosana, joka mahdollistaa sen käytön kaikissa Unreal Enginen ympäristöissä. [16, s. 1.]



Kuva 10. Dronen ja konenäön simulointi AirSim-simulaattorissa [16, s. 2].

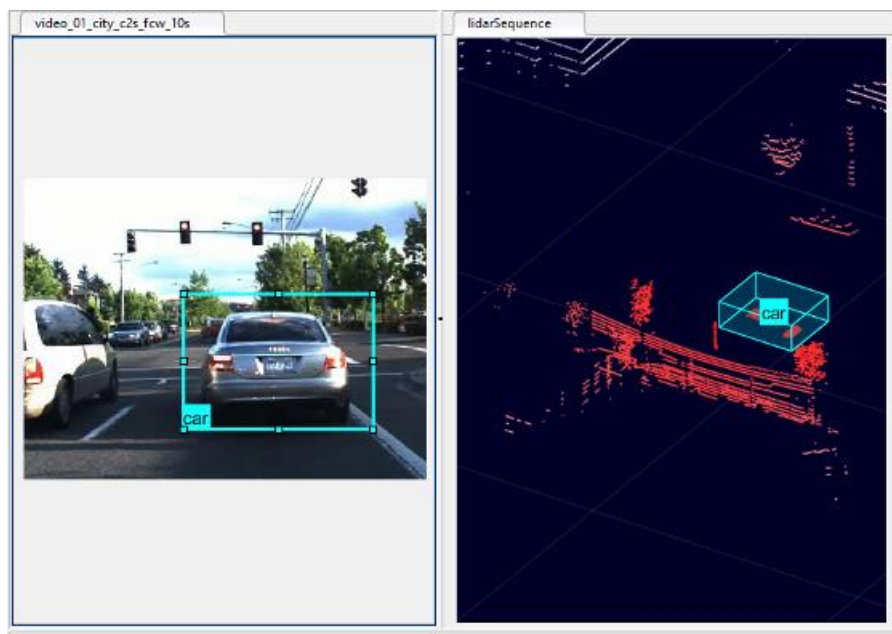
3.4 Matlab Automated Driving Toolbox

Matlab Automated Driving Toolbox on Matlabin yhteyteen julkaistu laajennus, joka sisältää algoritmeja ja työkaluja autonomisten järjestelmien suunnitteluun ja testaamiseen. Matlab on MathWorksin numeeriseen laskentaan tarkoitettu ohjelmointiympäristö, joka sisältää työkalut matriisien käsittelyyn, funktioiden ja datan visualisointiin, algoritmien toteuttamiseen, käyttöliittymien luomiseen. Matlab soveltuu hyvin käytettäväksi tieteellisen ja teknisen laskennan tehtäviin. Cleve Moler kirjoitti vuonna 1978 alkuperäisen Matlab-version, jota levitettiin yliopistoiden käyttöön. Julkaisun jälkeen ohjelmistoon on tullut useita laajennuksia eri sovellusaloihin, mikä mahdollistaa Matlabin käytön moneen eri tarkoitukseen [14, s. 6].

3.4.1 Ground truth labeler

Hyvä perusteellinen totuus (Ground truth) on ratkaisevan tärkeää, kun kehitetään autonomisten ajoneuvojen algoritmeja ja arvioidaan niiden suoriutumista. Tarkan ja monipuolisen datan luominen vaatii kuitenkin paljon aikaa ja resursseja. Ground truth labeler -työkalulla pystyy merkitsemään ja luokittelemaan dataa, ja työkalu sisältää erityyppisiä algoritmeja merkkkausprosessin nopeuttamiseksi. Työkalussa voi samanaikaisesti käsitellä videodataa, kuvasarjoja ja lasertutkan

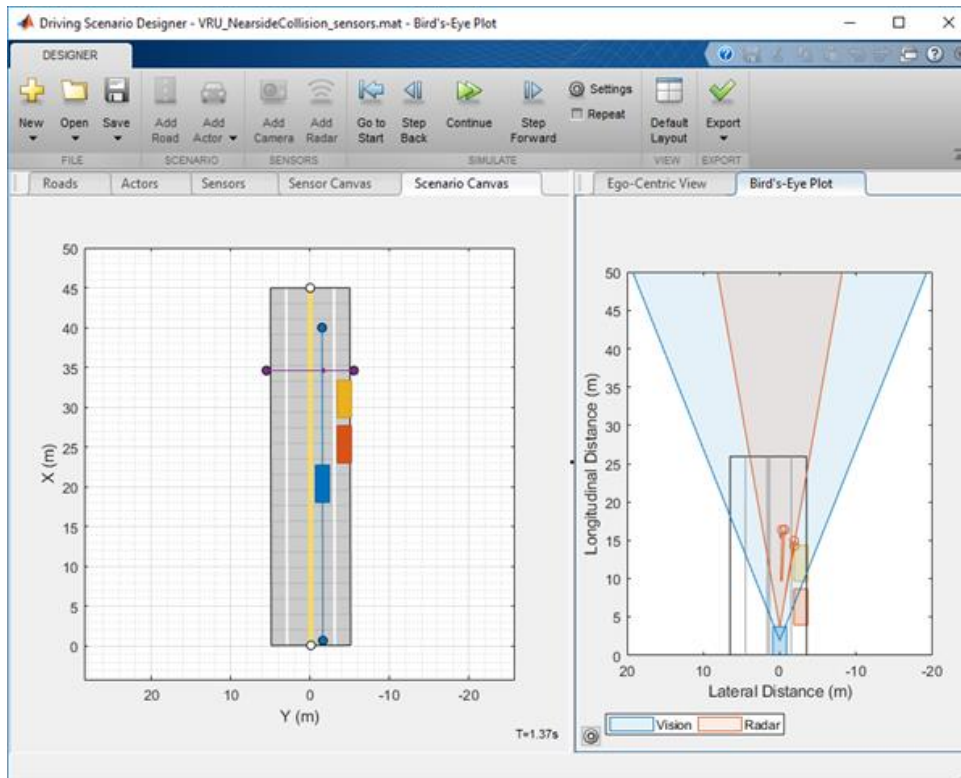
signaalia kuten pistepilviä (kuva 11). Merkittävää dataa voidaan hyödyntää koulutusdatana koneoppimisessa ja syväoppimisen malleissa, kuten kohteiden tunnistimissa tai semanttisissa segmentointiverkoissa. [17]



Kuva 11. Ground truth labeler -työkalu

3.4.2 Driving scenario designer

Driving scenario designer on työkalu, joka mahdollistaa erilaisten ajotilanteiden luomisen ja simuloinnin yksinkertaistetussa ympäristössä (kuva 12). Työkalussa voidaan simuloida eri anturityyppejä ja kerätä synteettistä anturidataa, jota voidaan käyttää apuna algoritmien suunnittelussa ja kehityksessä. Simuloitujen anturien sijaintia, näkökenttää ja päivitystaajuutta voidaan muuttaa, mikä helpottaa havainnollistamaan antureiden sijoittelua ajoneuvoon ja sitä, kuinka nämä kattavat konenäön näkökenttää. Skenaarioiden generointi mahdollistaa sellaisten vaikeiden ajotilanteiden simuloinnin, jotka voivat olla vaikeita testata liikenteessä. Työkalu sisältää myös valmiita skenaarioita kuten erikokoiset risteykset ja käännökset sekä eri EURO NCAP -testejä. Työkalu myös mahdollistaa skenaarioiden luomisen jo olemassa olevasta ajodatasta. [18, s. 5]



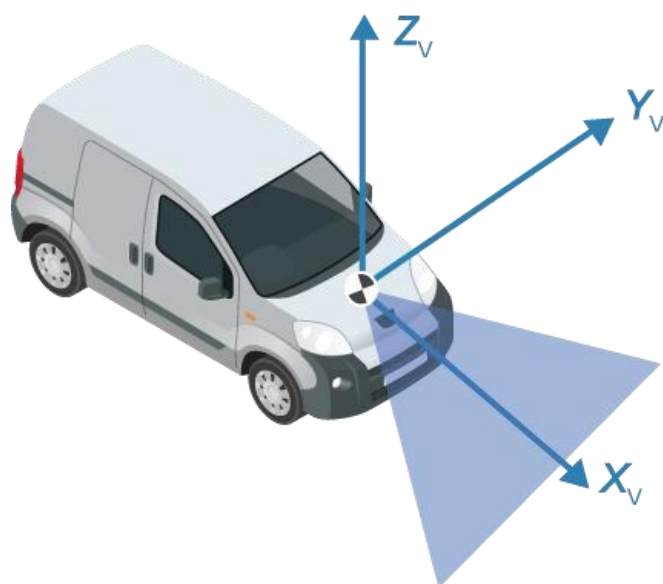
Kuva 12. Driving scenario designer -työkalun käyttöliittymä

4 Testaus

Tässä osiossa kuvataan sellaisen kameraan perustuvan havainnointiohjelmiston testausta, joka tunnistaa kameradatasta ajokaistat ja muut tienkäyttäjät sekä arvioi näiden etäisyydet. Esimerkissä käydään läpi, mitä eri tekniikoita ja vaiheita kameraan perustuva havainnointijärjestelmän vaatii ja testataan ohjelmistoa eri olosuhteissa kuvattuun ajodataan. Lopuksi arvioidaan ohjelmiston suoritumista ja testauksien tuloksia. Testaus suoritettiin käyttämällä Matlabin autonomisen ajamisen työkaluja.

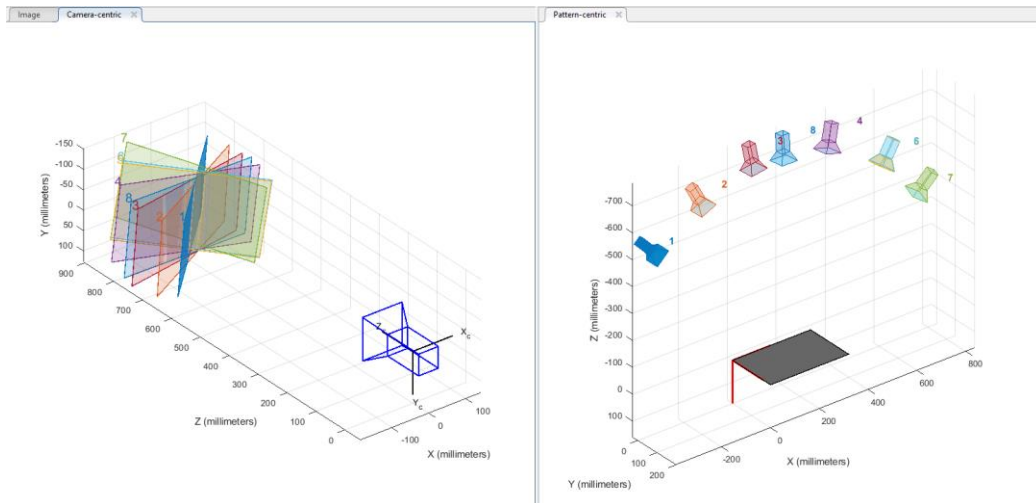
4.1 Kameran kalibrointi

Jotta kameraa voidaan käyttää osana autonomista ajoneuvoa ja sen havainnointijärjestelmää, täytyy kameran sisäiset ja ulkoiset parametrit kalibroida. Kameran sisäisiä parametreja ovat objektiivin polttoväli, keskipiste ja resoluutio. Kameran sisäisten parametrien lisäksi täytyy tietää ulkoiset parametrit kuten kameran sijainti ja kaltevuuskulmat ajoneuvon koordinaatistoon nähden. Tässä esimerkissä koordinaatiston origo sijaitsee suoraan kameran keskipisteen alapuolella (kuva 13). Kameran parametrien tietäminen on olennaista, jotta saadaan tarkka muunnos pikselien ja ajoneuvon koordinaatiston välillä.



Kuva 13. Ohjelmiston käyttämä koordinaatistojärjestelmä [18, s. 7].

Hyödynsin Matlabista löytyvää kameran kalibrointisovellusta käyttämäni kameran sisäisten parametrien määrittämiseen (kuva 14). Parametrien määrittystä varten tarvitsee ottaa 10–20 kuvaa kalibrointikuvista ja syöttää ne kalibrointisovellukseen. Aluksi oli vaikeuksia saada parametrien määrittäminen toimimaan, koska en ollut huomionnut kuvatun ajodatan ja kalibrointikuvien kuvasuhteiden eroa. Otin uudet kalibrointikuvat samalla kuvasuhteella kuin ajodata ja sain parametrit määritettyä oikeiksi. Ohjelmistoon täytyi myös määrittää kameran ulkoiset parametrit kuten kameran kiinnityskorkeus ja kaltevuuskulma.



Kuva 14. Kameran kalibrointityökalu

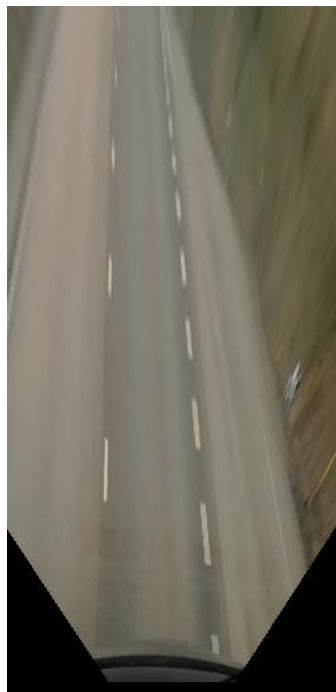
4.2 Kuvan prosessointi

Ennen kuin käsitellään kokonaista videota, ladataan videosta yksi kuva (kuva 15). Yhden kuvan prosessointi on nopeampaa kuin kokonaisen videon, ja tämä havainnollistaa paremmin systeemin suunnitteluun liittyviä vaiheita.



Kuva 15. Prosessointia varten ladattu kuva

Tiemerkintöjen segmentointiin ja tunnistukseen voidaan käyttää monia eri menetelmiä kuten lintuperspektiivialgoritmia tai reunantunnistusta. Reunantunnistusalgoritmilla voidaan ilmaista datassa tapahtuvia jyrkkiä sävynmuutoskohtia ja tällä menetelmällä tunnistaa kohteita. Tässä esimerkissä käytetään lintuperspektiivialgoritmia, joka muuntaa alkuperäisen datan lintuperspektiiviin (kuva 16). Tämän menetelmän yhtenä etuna on, että tiemerkinnot ovat lintuperspektiivistä paksuudeltaan yhtenäisiä, mikä helpottaa niiden tunnistusta ja segmentointiprosessia.

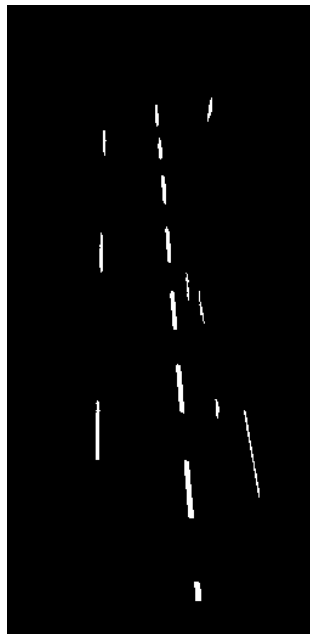


Kuva 16. Lintuperspektiiviin muutettu kuva.

4.3 Kaistamerkintöjen tunnistus

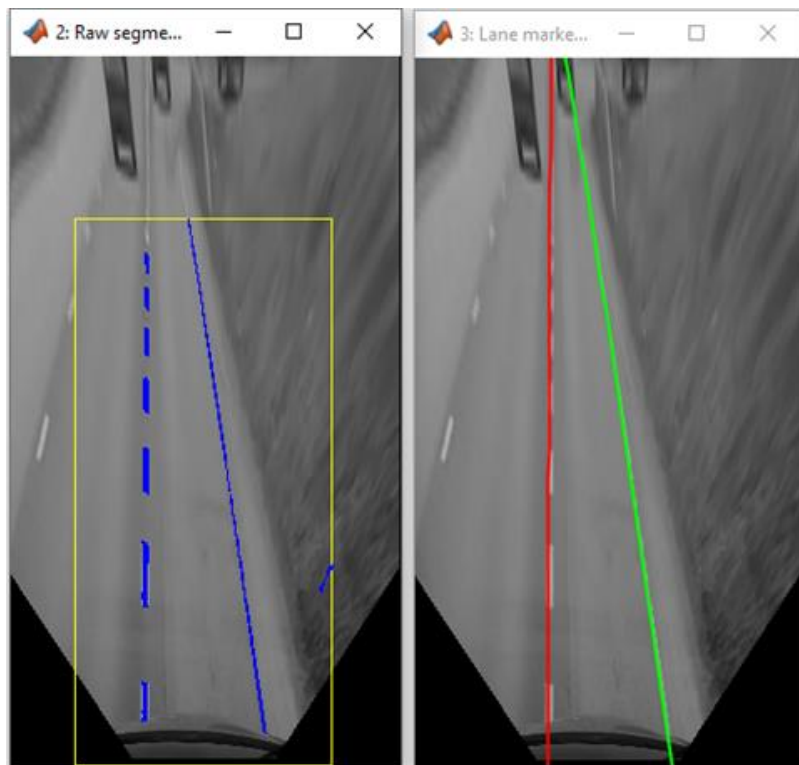
Kun kuva on saatu prosessoitua lintuperspektiiviin, voidaan siitä helpommin tunnistaa kaistamerkinnot. Tässä työssä käytetään Matlabin `segmentLaneMarkerRidge` funktiota, joka erottaa kaistamerkintäpikselit tienpinnasta (kuva 17). Funktiota varten täytyy data muuttua harmaasävyasteikkoon, josta funktio etsii sille määrätyn levyisiä kaistamerkintöjä. Tämä tekniikka on yksinkertainen ja tehokas eikä vaadi paljoa laskentatehoa. Funktio käyttää suurimmaksi osaksi maailman-

yksiköitä (metrejä), mikä helpottaa eriresoluutioisen datan käyttöä, verrattuna siihen, että käytettäisiin pikseleihin sidottuja yksiköitä. Tämä mahdollistaa eri sensoreiden testaamisen ja tekee havainnointijärjestelmän suunnittelusta joustavampaa.



Kuva 17. Funktion löytämät kaistamerkintäehdokkaat.

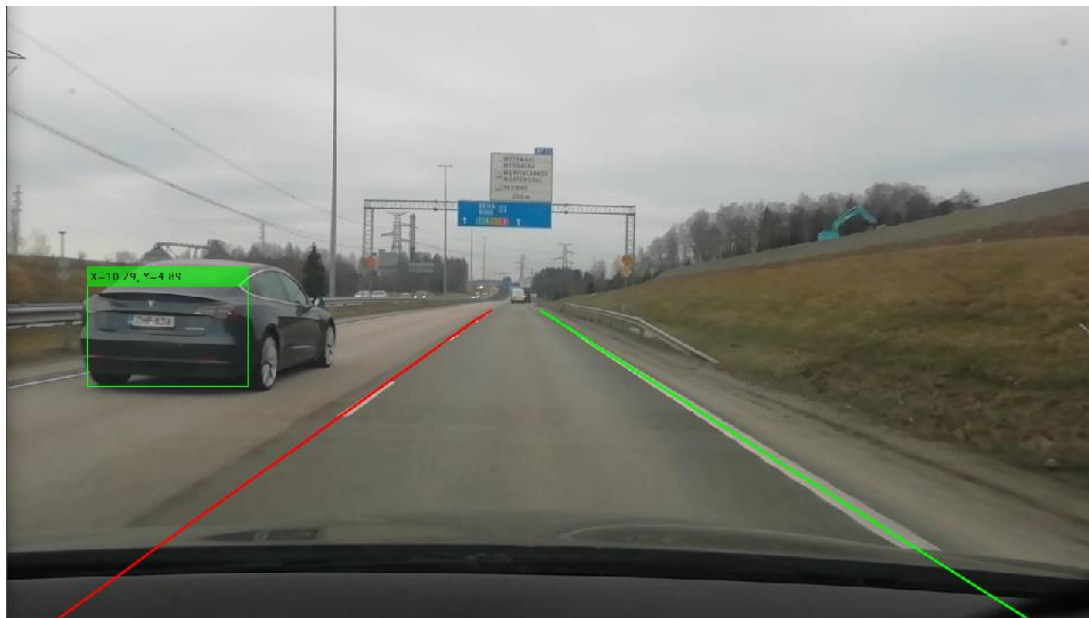
Koska funktio etsii sille määrätyn levyisiä kaistamerkintöjä, täytyi sille määrittää Suomen tiemerkinntöjen mitat. Suomessa ajoradan kaistat erottava keskiviivan, ajoradan reunaviivan ja ajokais-taviivan leveys on joko 10 cm tai 20 cm riippuen siitä, onko kyseessä taajama, maantie vai moot-toriväylä [17, s. 12]. Kaistamerkintöjä etsitään vain ajoneuvon suuntaisesti ja segmentoiduista pikseleistä hylätään ajoradan yli menevät pikselit sekä ajoradalle maalatut liikennemerkkit. Seg-mentoidut pikselit muutetaan ajoneuvon koordinaatistoon ja ne sovitetaan käyrälle käyttämällä RANSAC-algoritmia (kuva 18).



Kuva 18. Tunnistetut kaistamerkinnot (vasen) ja sovitetut egokaistat (oikea).

4.4 Ajoneuvojen tunnistus

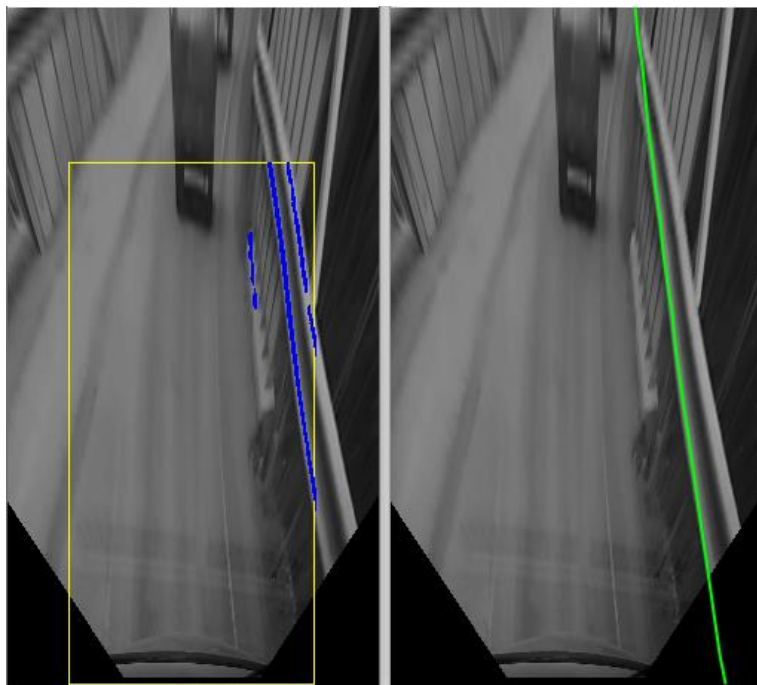
Muiden ajoneuvojen havaitseminen ja niiden seuranta on kriittinen osa autonomisen ajoneuvon havainnointijärjestelmää. Tässä ohjelmistossa käytetään ajoneuvon tunnistukseen ACF-tunnistinta, joka on ennalta koulutettu tunnistamaan ajoneuvojen etu- ja takaosan piirteitä (kuva 19). ACF-tunnistimelle täytyy määrittää tunnistettavien kohteiden mitat ja se, miltä alueelta tunnistin kohteita etsii. Tunnistin etsii kohteita vain tien pinnan suuntaisesti, mikä säästää laskennallista aikaa ja vähentää virheellisten tunnistusten määrää.



Kuva 19. Algoritmin tunnistama ajoneuvo.

4.5 Ohjelmiston testaus

Kun ohjelmistoon oli määritelty kameran parametrit ja kaistojen tunnistukseen vaadittavat asetukset sekä varmistettu, että ohjelmisto toimii, oli tavoitteena testata sitä eri olosuhteissa kuvattuun ajodataan. Testauksissa käytetty kamera ei kuitenkaan ollut resoluutioltaan ja objektiivin parametreiltä optimaalinen kyseiseen käyttötarkoitukseen. Esimerkiksi kehätiellä ajoradan keskiviivojen väli on suurempi kuin taajamassa eikä käytetyn kameran tarkkuus ollut riittävä, ja tämä vaikeutti kaistaviivan tunnistamista. Mallia testatessa huomattiin, että kaistamerkkien tunnistuksessa tapahtui paljon virheitä. Esimerkiksi osassa ajotilanteista algoritmi segmentoi tien reunassa olevan suojakaiteen ajoradan reunamerkinäksi (kuva 20). Tämänkaltaisia virheellisiä tunnistuksia sai vähennettyä muuttamalla segmentointiherkkyyttä ja rajaamalla aluetta, josta algoritmi etsii kaistamerkkien.



Kuva 20. Segmentointialgoritmin virheellinen tunnistus.

Lintuperspektiivimenetelmän heikkoutena on, että ilman ajoneuvon korin liikkeiden kompensointia vääristyy algoritmin tuottama kuva ajoneuvon korin kallistellessa. Esimerkiksi epätasaisuuksiin ajettaessa saattoi algoritmi menettää kaistojentunnistuksen. Ohjelmiston suoriutumista verrattiin kahteen eri sääolosuhteissa kuvattuun dataan. Aurinkoisella säällä kaistamerkintöjen segmentointi toimi huomattavasti paremmin kuin pilvisellä mutta ajoneuvojen tunnistus vaikeutui. Kirkkaampi valotus luo voimakkaamman varjon ajoneuvon ympärille, mikä vaikeuttaa ajoneuvon piirteiden tunnistamista (kuva 21). Kuvasta voidaan nähdä, miten paljon valotuksella on vaikutusta ajoneuvon piirteisiin. Tämä havainnollistaa hyvin, millaisia haasteita autonomiset autot ja sen kokenäkö kohtaavat.



Kuva 21. Valotuksen vaikutus ajoneuvon piirteisiin.

Kuvassa 22 ohjelmisto ei tunnistanut edellä ajavaa pakettiautoa, minkä takia pelkän kameraan perustuvan havainnointijärjestelmän käyttö on riski. Lasertutka tai normaali tutka olisi havainnut pakettiauton. Useasti risteysalueilla on ajoradan reunaviivan tilalla kiveys, joka on huomattavasti vaikeampi tunnistaa kuin maalattu kaistaviiva. Tehty ohjelmiston testaus havainnollisti hyvin, millaisia haasteita autonomisen ajoneuvon havainnointijärjestelmä kohtaa. Vaihtelevat sääolosuhteet sekä tienpinnan ja tiemerkinntöjen kunto vaikeuttavat tiemerkinntöjen ja muiden ajoneuvojen tunnistusta. Kun sääolosuhteet ovat optimaaliset ja tiemerkinntöjen kunto hyvä, toimii työssä testattu kaistantunnistusohjelmisto hyvin.



Kuva 22. Tilanne, jossa ohjelmisto ei tunnistanut edellä ajavaa pakettiautoa.

5 Yhteenveto

Työn tavoitteena oli tarkastella autonomisia ajoneuvoja, sen havainnointijärjestelmää ja siihen kuuluvia komponentteja. Lisäksi työssä tutustuttiin autonomisten ajoneuvojen simulointiin tarjolla oleviin ohjelmistojä ja perehdyttiin tarkemmin Matlabin autonomisen ajamisen lisäosaan. Työssä käytiin läpi kaistatunnistusohjelmiston eri vaiheita ja tekniikoita sekä testattiin kyseistä ohjelmistoa ja sen toimintaa.

Kaistatunnistusohjelmiston testaus havainnollisti hyvin, kuinka haasteellista kaistamerkintöjen ja muiden ajoneuvojen tunnistaminen on. Ohjelmiston testissä käytetty ajodata oli kuvattu suhteellisen hyvissä olosuhteissa, mutta silti osassa tilanteissa ei kaistamerkintöjen tunnistus toiminut. Ajoneuvojen tunnistuksessa suurimpana haasteena oli auringon luoma varjostus. Kirkas valotus luo kohteiden ympärille varjon, joka muuttaa kohteen ulkonäköä. Erityisesti tummat ajoneuvot sulautuvat tummaa taustaa vasten. Testissä käytetty kamera ei välttämättä ollut optimaalinen kyseiseen käyttötarkoitukseen, ja sopivammalla kameralla olisi voitu päästä parempiin tuloksiin.

Työn tutkimusta voisi jatkaa vertailemalla eri kaistatunnistustekniikoita kuten syväoppimisella koulutettua tunnistinta. Aihe oli haastava, ja ilman suurempaa ohjelmointikokemusta on monien simulointiin tarkoitettujen ohjelmistojen käyttö vaikeaa. Työssä käytetty Matlabin lisäosa, Automated Driving Toolbox, sisältää paljon mielenkiintoisia autonomiseen ajamiseen liittyviä työkaluja, ja nämä työkalut ovat helposti lähestyttäviä ja sopivat hyvin jollekin, joka haluaa syvemmin tutustua autonomisiin ajoneuvojärjestelmiin.

Lähteet

- 1 Innamaa, Satu; Kanner, Heikki; Rämä, Pirkko & Virtanen, Ari. 2015. Automaation lisääntymisen vaikutukset tieliikenteessä. Verkkoaineisto. <https://arkisto.trafi.fi/filebank/a/1461576365/fdb4c6b311fb1da01cf40bdf8fd33b5c/20473-Trafi_tutkimuksia_01-2015_-_Automaattiajaminen.pdf>. Luettu 21.2.2020.
- 2 Dickmanns, Ernest. 2007. Dynamic Vision for Perception and Control of Motion
- 3 Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. 2018. Society of Automotive Engineers.
- 4 Autonominen ajaminen. Verkkoaineisto. Maanmittauslaitos. <<https://www.maanmittauslaitos.fi/tutkimus/teematietoa/autonominen-ajaminen>>. Luettu 16.4.2020.
- 5 Autonomous vehicle technology report. Verkkoaineisto. Wevolver. <<https://www.wevolver.com/article/2020.autonomous.vehicle.technology.report>>. Luettu 25.3.2020.
- 6 Building maps for a self-driving car. Verkkoaineisto. Waymo. <<https://medium.com/waymo/building-maps-for-a-self-driving-car-723b4d9cd3f4>>. Luettu 17.4.2020.
- 7 High definition maps for autonomous vehicles. Verkkoaineisto. Sanborn. <<https://www.sanborn.com/highly-automated-driving-maps-for-autonomous-vehicles/>>. Luettu 22.4.2020.
- 8 Satelliittipaikannus. Verkkoaineisto. Maanmittauslaitos. <<https://www.maanmittauslaitos.fi/tutkimus/teematietoa/satelliittipaikannus>>. Luettu 20.3.2020.
- 9 Inertial measurement unit. Verkkoaineisto. Honeywell. <<https://sensing.honeywell.com/honeywell-sensing-inertial-measurement-unit-6df-productsheet-000741-2a-en.pdf>>. Luettu 20.3.2020.
- 10 Daft, Chris. 2016. Self-Driving Cars Expert Witness: Physics Drives the Technology. Verkkoaineisto. <<https://riversonicsolutions.com/self-driving-cars-expert-witness-physics-drives-the-technology/>>. Luettu 15.4.2020.
- 11 Fridman, Lex. 2017. Large-Scale Naturalistic Driving Study of Driver Behavior and Interaction with Automation. Massachusetts: Massachusetts Institute of Technology.

- 12 Smith, J. 2019. Why Simulation is the Key to Building Safe Autonomous Vehicles. Verkkoaineisto. <<https://www.electronicdesign.com/markets/automotive/article/21808661/why-simulation-is-the-key-to-building-safe-autonomous-vehicles>>. Luettu 10.4.2020
- 13 Drive constellation. Verkkoaineisto. Nvidia. <<https://www.nvidia.com/en-us/self-driving-cars/drive-constellation/>>. Luettu 10.4.2020.
- 14 Hahn, Brian & Valentine Daniel. 2019. Essential MATLAB for Engineers and Scientists.
- 15 Dosovitskiy, Alexey. 2019. CARLA: An Open Urban Driving Simulator. New York: Cornell University.
- 16 Shah, Shital. 2017. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. Washington: Microsoft Research.
- 17 Ground truth labeler. Verkkoaineisto. Matlab. <<https://se.mathworks.com/help/driving/ref/groundtruthlabeler-app.html>>. Luettu 15.2.2020
- 18 Sensor Fusion and Tracking for Autonomous Systems. 2019. Massachusetts: Mathworks.
- 19 Tiemerkintöjen suunnittelu. 2015. Ohje. Helsinki: Liikennevirasto.