

Pauli Korhonen

# Lennoxin langaton tiedonkeruujärjestelmä

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tietotekniikan koulutusohjelma

Insinöörityö

30.9.2015

Tekijä Otsikko	Pauli Korhonen Lennokin langaton tiedonkeruujärjestelmä
Sivumäärä Aika	43 sivua 30.9.2015
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Tietotekniikka
Suuntautumisvaihtoehto	Ohjelmistotekniikka
Ohjaajat	Lehtori Keijo Länsikunnas Insinööriksi Mika Nieminen
<p>Insinööriyön tarkoituksena oli suunnitella ja toteuttaa langaton tiedonkeruujärjestelmä Puolustusvoimien maalilennokkiin. Järjestelmään sisältyy kamera, asento- ja jänniteanturit sekä näiden tietojen esittäminen maassa olevalle lennättäjälle reaaliajassa. Työ kattaa myös tämän järjestelmän kiinnittämisen lennokkiin ja vastaanottavan maa-aseman suunnittelun ja toteutuksen.</p> <p>Työssä hyödynnettiin kolmiulotteista tulostusta itse mallinnettujen koteloiden ja kiinnikkeiden tekoon. 3D-tulostuksen avulla kiinnitysratkaisut pystyttiin räätälöimään tarkasti käytettyille komponenteille. Maa-aseman tiedot esittävä ohjelma on ohjelmoitu käyttäen C-ohjelmointikieltä. Sekä maa-asemalla että lennokissa on käytössä Raspberry Pi -minitietokone. Tietoliikenneyhteytenä käytetään kaupallista mobiilidataverkkoa suojatun VPN-tunnelin (Virtual Private Network) kanssa.</p> <p>Valmista järjestelmää testattiin lennätystilaisuudessa, jonka aikana kaikki toimi hyvin. Järjestelmä kesti rikkoontumatta rajutkin laskeutumiset yhteyden katkeamatta. Järjestelmää voidaan käyttää jatkossa esimerkiksi maastouttamisharjoituksissa, joissa lennokin avulla voidaan havaita ilmasta käsin, onko maastoutus onnistunut.</p> <p>3D-tulostaminen mahdollistaa ongelmistaan huolimatta erilaisten pienten kappaleiden, kuten koristeiden tai varaosien, tulostamisen. Tulevaisuudessa laitteiden kehittyessä on mielenkiintoista nähdä, millaisia kappaleita voidaan tulostaa. Tietokoneiden pienentyminen ja hintojen lasku ovat mahdollistaneet tietokoneiden hyödyntämisen mitä erilaisimpiin tehtäviin. Vielä muutama vuosi sitten insinööriyön tekeminen ei olisi onnistunut nykymuodossaan. On mielenkiintoista nähdä, toteutuuko esimerkiksi esineiden internet, kun tietokoneet jatkavat pienenemistään.</p>	
Avainsanat	videonsiirto, 4G, 3D-tulostus, vektorigrafiikka

Author Title	Pauli Korhonen Wireless data gathering system for the drone
Number of Pages Date	43 pages 30 September 2015
Degree	Bachelor of Engineering
Degree Programme	Information and Communications Technology
Specialisation option	Software Engineering
Instructors	Keijo Länsikunnas, Lecturer Mika Nieminen, Captain Engineer
<p>The goal of the final year project was to design and implement wireless data gathering system for a drone of the Finnish Defence Forces. The system includes a camera and angle and voltage meters. Gathered information is then transferred in real time to the pilot at the ground station. This thesis goes through the installation of the system on the drone as well as designing and implementing the receiving ground station.</p> <p>3D printing was used for manufacturing the cases and clamps designed for the system. It was possible to create optimal, tailored and unique cases by using 3D printing. The code presenting the gathered information at the ground station was programmed in the C programming language. A Raspberry Pi minicomputer was used at the ground station and on the drone. Data is transferred from the drone via commercial mobile data carrier by using a VPN tunnel.</p> <p>The system was tested in practice and everything worked as expected. The system remained fully functional during and after hard landings and the connection was not lost during the tests. In the future the system can be used in practices to determine how well tents and other equipment are camouflaged.</p> <p>3D printing enables manufacturing of different kind of spare parts and other small objects. It is interesting to see what kind of objects it is possible to print in the future when printers are more advanced than previously. Computers have been getting more inexpensive and smaller in size which has led to the usage of computers in many applications. It would have been impossible to make the final year project without recent advancements of computers. It will be exciting to see if the Internet of Things will become common when computers are getting smaller and smaller.</p>	
Keywords	video transferring, 4G, 3D printing, vector graphics

# Sisällys

## Lyhenteet

1	Johdanto	1
2	Tiedonkeruujärjestelmän suunnittelu	2
2.1	CAD-ohjelmistot	2
2.2	Maa-asema	4
2.3	Keinohorisontti	7
2.4	Videokuva	12
2.5	Lenkki	12
2.6	Tietoliikenne	13
3	Koteloiden tulostus ja ohjelmiston kehitys	15
3.1	3D-tulostus	15
3.2	Maa-asema	22
3.3	Lenkki	28
3.4	Raspberry Pi 2 -minitietokoneen päivityksen ongelmat	31
4	Havaitut ongelmat ja niiden korjaaminen	32
4.1	Ohjelmiston muutokset	33
4.2	Lenkin värinäongelma	34
4.3	Lenkitys Lohtajalla	37
5	Yhteenveto	38
	Lähteet	40

## Lyhenteet ja käsitteet

4G	Neljännän sukupolven mobiilidatayhteys.
ABS	Akryylinitriilibutadieenistyreeni. Muovilaatu, jota käytetään 3D-tulostuksessa. Haastava tulostettava, onnistuessaan hyvin rasitusta kestävä tuloste.
Bittikarttakuva	Digitaalinen kuva, joka muodostetaan tallentamalla määrätyn kokoisen suorakulmion alueelta jokaisen kuvapisteen väriarvot erikseen. Kuvan esittäminen eri koossa vaatii häviöllistä skaalaamista.
CAD	Computer aided design. Tietokoneavusteinen mallinnus.
FDM	Fused deposition modeling. Pursotustekniikkaan perustuva 3D-tulostintekniikka.
h.264	Videon häviöllinen pakkaustekniikka, joka hyödyntää pakkauksessa edellisen ruudun informaatiota.
Teräväpiirtotaso	Tarkkuudeltaan vähintään 1280 x 720 oleva video, jonka pakkaus ei ole liian voimakas, jotta tarkkuuden tuoma hyöty ei häviäisi pakkausvirheisiin.
Kill	Linux-käyttöjärjestelmän komento, jolla voidaan sulkea ajossa olevia ohjelmia.
MJPEG	Motion JPEG. Valokuvaan kehitettyä häviöllistä JPEG-standardia käyttävä videopakkaustekniikka. Ei hyödynnä edellisten ruutujen informaatiota.
NUC	Proessorivalmistaja Intelin käyttämä termi 10 x 10 senttimetrin kokoiselle tietokoneelle.
Omxplayer	Linux-käyttöjärjestelmän videosoitin, joka tukee näytönohjaimella tapahtuvaa kiihdytystä.

OpenVG	Vektorigrafiikan piirtoon ja käsittelyyn tarkoitettu kirjasto, joka tukee näytönhajaimella tapahtuvaa kiihdytystä.
PLA	Polylaktidi. Muovilaatu, jota käytetään 3D-tulostuksessa. Kova, mutta hauras materiaali.
QT	Kirjasto, jonka avulla voidaan luoda grafiikkaa helposti. Tukee näytönhajaimella tapahtuvaa kiihdytystä grafiikan esittämisessä.
Raspbian	Raspberry Pi -minitietokoneelle suunniteltu Linux-käyttöjärjestelmäjakeluversio.
Rc.local	Linux-käyttöjärjestelmän tiedosto, jonka sisällä olevat komennot suoritetaan aina järjestelmän käynnistyessä.
Ristikääntäminen	Englanniksi cross compiling. Ohjelman kääntäminen toisella tietokoneella, kuin millä ohjelmaa tullaan käyttämään.
SLS	Selective Laser Sintering. 3D-tulostintekniikka, jolla voidaan tulostaa liikkuvia osia. Käyttää tulostusmateriaalina pulverimaista muoviraetta.
SSH	Secure Shell. Turvallisen terminaalietäyhteyden tarjoava protokolla.
Struct	C-ohjelmointikielen säiliötietotyyppi, johon voidaan kerätä toisiinsa liittyviä muuttujia.
Terminaali	Linux-käyttöjärjestelmän tekstimuotoinen komentoikkuna.
Vektorikuva	Digitaalinen kuva, joka muodostetaan matemaattisesti viivoista, kaarista ja muista muodoista. Voidaan esittää minkäkokoisena tahansa, koska kuva voidaan laskea ja piirtää suoraan esitystarkkuudella.
VPN	Virtual Private Network. Suojattu tunneli internetin kautta tapahtuvaan kahden laitteen väliseen kommunikointiin.

## 1 Johdanto

Insinööri työ tehtiin yhdessä Metropolia Ammattikorkeakoulun insinööriopiskelijan Ari Rajamäen kanssa. Alkuperäisenä tavoitteena oli suunnitella multikopteri kahdella kameralla ja eleohjauksella. Kameroiden kuvaama videokuva oli tarkoitus esittää virtuaalilaseilla lennättäjälle. Insinööri työ aihe kiinnosti molempia osapuolia, koska siinä oli mahdollista yhdistää käytännön suunnittelu ohjelmointiin.

Kopterin suunnittelun aikana tilaajaksi vaihtui kuitenkin Puolustusvoimat. Työ päätettiin tehdä mieluummin Puolustusvoimille kuin koululle tilaajalle, koska silloin työn lopputulos tulee oikeaan käyttöön. Puolustusvoimat halusi maalilennokkiinsa tiedonkeruujärjestelmän, jonka avulla voisi seurata jatkuvasti, missä asennossa lennokki lentää. Lisäksi lennokkiin haluttiin kamera kuvaamaan eteenpäin, jolloin voidaan katsoa sekä maastouttamisen onnistumista että lennokkia kohti ammuttavia ammuksia. Nämä tiedot haluttiin saada näkyviin mahdollisimman pienellä viiveellä ja suurella resoluutiolla.

Puolustusvoimat tutki järjestelmän avulla, olisiko lennokkiin asennettavalla tiedonkeruujärjestelmällä mahdollista saada lisäarvoa lennätysharjoituksiin. Järjestelmän avulla ei ollut tarkoitus pystyä lennättämään lennokkia, vaan tarkoitus oli keskittyä maastossa olevien kohteiden havainnointiin.

Työn tavoitteena oli suunnitella ja toteuttaa Puolustusvoimille sekä lennokkiin asennettava tiedonkeruujärjestelmä että tietojen vastaanottoon käytettävä maa-asema. Lennokkiin oli tarkoitus tehdä mahdollisimman vähän muutoksia järjestelmää varten. Työn yksi osa-alue olikin suunnitella järjestelmälle sellainen kiinnitysratkaisu, että järjestelmä on helppo siirtää lennokista toiseen. Kiinnitysratkaisussa hyödynnettiin kolmiulotteista tulostusta.

Työtä tehdessä käytiin neuvotteluja Puolustusvoimien alueella Santahaminassa ja Tikkakoskella. Tikkakoskella myös koekäytettiin järjestelmää ennen varsinaista lennätystä Lohtajalla.

## 2 Tiedonkeruujärjestelmän suunnittelu

### 2.1 CAD-ohjelmistot

Tietokoneavusteinen suunnittelu eli Computer Aided Design (CAD) alkoi yleistyä 1970-luvun lopulla, kun suunnitteluohjelmistoja kehittävien yritysten määrä kasvoi (1). Aluksi CAD oli vain harvojen osaajien käytettävissä, mutta vähitellen yhä useampi insinööri siirtyi käyttämään tietokonetta paperilla suunnittelun sijaan. Aluksi ohjelmistot olivat kaksiulotteisia, mutta myöhemmin saataville tulivat 3D-pohjaiset ohjelmistot, joilla kappaletta voitiin pyörittää mihin tahansa asentoon kuvaruudulla (2). Näissä järjestelmissä kappale muistutti esikuvaansa eikä ollut ainoastaan joukko kaksiulotteisia kuvia.

CAD-ohjelmistoja käytetään sekä peli- ja elokuvateollisuudessa että teollisessa suunnittelussa. Kolmiulotteiset kappaleet koostuvat kolmion muotoisista palasista, joita kutsutaan polygoneiksi. Peleissä tärkeintä on saada luotua mahdollisimman realistinen, mutta mahdollisimman vähällä polygonimäärällä varustettu malli, joten nämä ohjelmistot eroavat teolliseen suunnitteluun tarkoitetuista ohjelmista. Animointi ja materiaalien tarkka mallinnus on toteutettu monipuolisemmin pelien ja elokuvien tekoon keskittyvissä ohjelmissa. Teollisessa suunnittelussa sen sijaan tärkeintä on saada mitoiltaan ja ominaisuuksiltaan tarkka malli, mutta polygonien määrällä ei ole niin suurta merkitystä. Polygonien määrää halutaan rajoittaa peleissä, koska jokaisen polygonin piirtäminen vie näyttöohjaimen prosessoriaikaa. Jos jokainen pelimaailman kappale sisältäisi miljoonia polygoneja, olisi laajoja pelimaailmoja mahdotonta päivittää pelaajalle riittävän nopeasti.

Ohjelmat ovat hintatasoltaan vaihtelevia: jotkin ohjelmat ovat avointa lähdekoodia ja näin ollen ilmaisia, kun taas jotkin kaupalliset, suljetun lähdekoodin ohjelmat voivat lisenssistä riippuen maksaa jopa tuhansia euroja (3). Maksullisista fotorealistista 3D-grafiikkaa luovista ohjelmista AutoDesk Maya on yksi suosituimmista vaihtoehdoista. Siihen saa lisäosana Pixarin kehittämän RenderMan-kirjaston, jolla on tehty muun muassa kaikki Pixarin omat animaatioelokuvat (4). Tällä yhdistelmällä voidaan luoda ammattilaistasoista 3D-grafiikkaa, mutta varsinaiseen teolliseen mallintamiseen se ei ole tarkoitettu.



Ilmaisista avoimen lähdekoodin 3D-mallinnusohjelmista esimerkkinä on Blender (5), jolla yleensä tehdään graafisesti näyttäviä mallinnuksia, joiden ei kuitenkaan tarvitse olla millimetrin tarkkoja, joskin tarkkakin mallintaminen on mahdollista (6). Blenderillä onnistuu myös kokonaisen maailman mallinnus ja kuvaus, joista esimerkkeinä ovat vapaaehtoisvoimin tehdyt OpenMovie-projektit (7; 8). Yhdysvaltain avaruushallinto NASA on tehnyt Blenderillä joitakin julkisesti saatavilla olevista 3D-malleistaan (9). Blenderiä on käytetty myös elokuvien teossa, mistä esimerkkinä Spider Man 2, jossa animaattori Anthony Zierhut käytti sitä luodakseen elokuvan eri kohtauksia 3D-maailmaksi, jotta muu tiimi pystyisi ymmärtämään ohjaaja Sam Raimin vision paremmin (10).

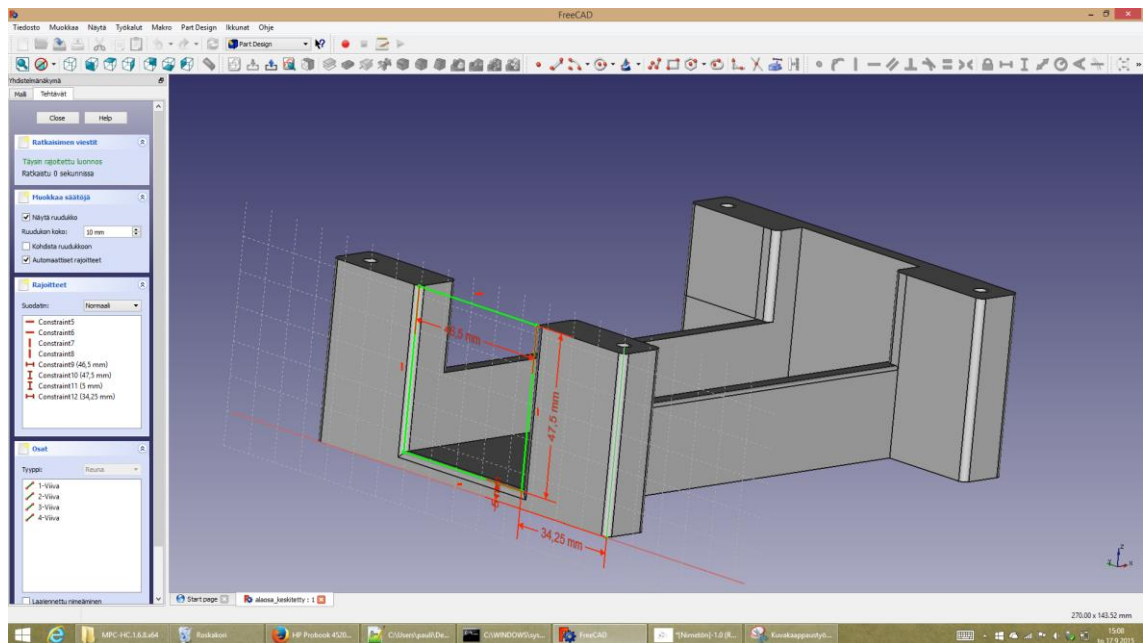
Yksi esimerkki varsinaiseen 3D-kappaleiden mallintamiseen tarkoitetuista kaupallisista ohjelmista on AutoDesk AutoCAD (11). Sitä on käytetty muun muassa Brasilian kansallisen urheilustadionin Castro Mellon suunnittelussa (12). AutoCADilla voidaan luoda tarkkoja 3D-malleja, joilla on tarkkoja rajoituksia (13). Esimerkiksi mallinnettaessa taloa voidaan sen avautuville oville asettaa rajat, jotta ovet avautuvat vain tiettyyn suuntaan ja määrätyn verran. Ohjelmalla voidaan myös simuloida erilaisten materiaalien kestävyttä ja mallintaa esimerkiksi vaihteisto, jossa osat liikkuvat kuten oikeassa vaihteistossa.

Ilmaisista, tarkkaan 3D-mallintamiseen tarkoitetuista ohjelmista voidaan ottaa esimerkiksi tässäkin työssä käytetty FreeCAD (14). Se mahdollistaa mallien muokkaamisen myös jälkikäteen, koska se sisältää jokaisen muokkauksen vaiheen osapuussaan. Työtä tehdessä kuitenkin huomattiin, ettei muokkaus aina ole jälkikäteen mahdollista, vaan uudemmat vaiheet pitää poistaa, jotta muutoksia voi tehdä. Tämä oli joidenkin kappaleiden kohdalla ongelmallista, koska ohjelman rajoitusten vuoksi kappaleen mallinnuksen saattoi joutua aloittamaan uudelleen alusta.

Varsinainen syy joidenkin kappaleiden muokkauksen epäonnistumiseen ei täysin selvinnyt. Todennäköisesti ongelmat johtuivat uudempien muutosten riippuvuuksista edellisissä työvaiheissa, jolloin ohjelma ei osannut päivittää uudempia muutoksia vanhempiin työvaiheisiin. Esimerkkinä tällaisesta tilanteesta voidaan ottaa tapaus, jossa kotelon seinien paksuutta olisi haluttu ohentaa, mutta seiniin oli paksuuden määrittelyn jälkeen lisätty läpivientireikiä. Näin ollen uudempi työvaihe riippui edellisestä, eikä edellisen vaiheen muuttaminen, eli seinien paksuuden määrittely uudelleen, onnistunut.

FreeCADilla mallinnettiin kaikki työssä käytetyt osat kiinnikkeistä koteloihin. Kolmiulotteinen mallinnus ja -tulostus avasi monia mahdollisuuksia valmistaa juuri sellaiset kapaleet, jotka parhaiten soveltuvat tähän työhön. Kolmiulotteisia malleja suunniteltaessa huomattiin, ettei kaikkia yksityiskohtia osattu huomioida aluksi, mikä johti siihen, että malleja jouduttiin tulostamaan useaan kertaan, ennen kuin kaikki ominaisuudet saatiin korjattua mallista. Tämä johtui, ainakin osittain, aiemman 3D-mallinnustaidon puutteesta.

Kuvassa 1 näkyy lennokin puolella käytetty akkukotelo mallinnuksen aikana. Vihreä, kuvan keskellä etualalla oleva neliö määrittelee akun koon kotelossa. Vasemmalla puolella näkyvät rajoitteet, joilla neliö on määritelty oikeaan kohtaan mallia ja oikean koiseksi. Yläreunassa näkyy erilaisia työkaluja, joilla voi joko tehdä rajoitteita tai uusia muotoja.



Kuva 1. FreeCAD-ohjelmisto, jossa on mallinnettavana akkukotelo.

## 2.2 Maa-asema

Työtä varten piti suunnitella jonkinlainen tapa esittää kerätty data käyttäjäystävällisessä muodossa. Ensin ajatuksena oli hankkia jokin säänkestävä tablet-tietokone, koska sitä olisi tarkoitus käyttää myös pienessä sateessa. Ongelmana oli kuitenkin vedenkestävien laitteiden korkea hintataso ja rajallinen valikoima. Käytännössä kaikki sopivat lait-

teet olivat Android-käyttöjärjestelmällä toimivia, eikä Android-sovelluskehitykseen haluttu lähteä.

Tablet-tietokoneesta ideaa kehitettiin hieman eri suuntaan. Ajatuksena oli hankkia jokin erillinen, noin 20-tuumainen näyttö tai televisio, joka kiinnitettäisiin salkun kanteen. Salkun toisessa puolikkaassa olisi tietokone vastaanottamassa ja esittämässä tietoa. Tähän soveltuisi jokin erittäin pienikokoinen ja vähävirtainen PC-tietokone. Suoritinvalmistaja Intelillä on käytössä NUC-termi (15), jolla tarkoitetaan tällaista fyysisesti pientä ja vähävirtaista laitetta. Kyseessä on noin 10 x 10 senttimetrin kokoinen PC-tietokone. Tällaiseen tietokoneeseen voisi asentaa joko Linux- tai Windows-käyttöjärjestelmän suorittamaan vastaanottavaa ohjelmaa. Ideaa pidemmälle ajateltaessa kuitenkin todettiin, ettei näin järeää tietokonetta kannattaisi hankkia tähän tarkoitukseen. Halvimmatkin NUC-tietokoneet maksoivat useita satoja euroja kaikkine osineen. (16.)

Tarkoitukseen päätettiin hankkia jokin minitietokone, koska sellainen riittää ominaisuuksiltaan työn tarpeisiin, mutta on hinnaltaan huomattavasti edullisempi (17). Tästä ajatuksesta päädyttiin lopulta käyttämään Raspberry Pi -minitietokonetta, jonka virrankulutus tyypillisessä käytössä on vain muutamia watteja. Myös sen fyysinen koko on paljon pienempi kuin NUC-tietokoneen. Lisäksi sen suorituskyky on huomattavasti vaatimattomampi, mutta kuitenkin riittävä tähän käyttöön. Raspberry Pin valinnasta kerrotaan tarkemmin Ari Rajamäen insinöörityössä (17).

Näytöksi valittiin 22-tuumainen televisio, jota voi käyttää sekä 12 että 230 voltilla. Kahdentoista voltin jännitteellä toimiminen oli etu, koska näyttöä haluttiin olevan mahdollista käyttää myös akulla. Huonona puolena näytössä on se, ettei se kestä vettä, joten vedenpitävyyden osalta jouduttiin tekemään kompromissi.

Lisäksi tarvittiin näppäimistö ja hiiri, jotta ohjelman sai käynnistettyä. Aluksi ohjelma oli asetettu käynnistymään automaattisesti, mutta testauksessa huomattiin tämän aiheuttavan vikatilanteissa ongelmia, joten toimintaa muutettiin. Tähän vähäiseen käyttöön löytyi sopiva medianäppäimistö, jossa hiiren tilalla oli kosketuslevy. Erillinen, tason vaativa hiiri olisi ollut kenttäolosuhteissa erittäin hankala, koska sopivan tasaisen pinnan löytyminen voi olla vaikeaa.

Elektroniikan lisäksi tarvittiin salkku, johon kaikki tavarat saadaan mahtumaan tiiviiksi paketiksi helppoa kuljetusta varten. Tähän löytyikin pienen hakemisen jälkeen juuri

sopivankokoinen salkku, jonka kanteen näyttö saatiin kiinnitettyä. Muut osat olisivat mahtuneet pienempäänkin tilaan, joten näyttö oli määräävä tekijä valintaa tehtäessä.

Kaikkien osien löydyttyä tutkittiin, mikä olisi paras tapa kiinnittää osat salkkuun. Kiinnityksen piti olla siisti ja kestävä, mutta sen verran pieni ja kevyt, että näyttö olisi mielekästä asentaa käsin liikuteltavaan salkkuun. 3D-mallintamista ja tulostamista hyödynnettiin keskusyksikön suojaamiseen ja kiinnittämiseen salkkuun. Myös näytön kiinnike tulostettiin muovista. Kuvassa 2 näkyy osien sijoittelu salkkuun, kun maa-asema on valmis.



Kuva 2. Lennättäjälle videon ja lentotiedot näyttävä maa-asema osineen.

Kun kaikki osat oli valittu, aloitettiin ohjelmistosuunnittelu eli se, miten kaikki osat saadaan toimimaan yhteen ja tarvittava tieto esitettyä käyttäjälle. Ensimmäinen osa-alue oli suunnitella, kuinka lennokin asentotieto näytetään mahdollisimman selkeästi käyttäjälle. Tässä päädyttiin käyttämään oikeista lentokoneista tuttua keinohorisonttia, joka näyttää lentokoneen asennon suhteessa horisonttiin. Piti siis löytää tapa esittää keinohorisontti ja liikkuva videokuva yhtä aikaa kuvaruudulla. Tämä osoittautui hankalaksi Raspberry Pin alhaisen laskentatehon vuoksi; käytännössä Raspberry Pissa on suhteellisen tehokas näytönohjain, mutta sen prosessorin suorituskyky on huomattavan alhainen.

### 2.3 Keinohorisontti

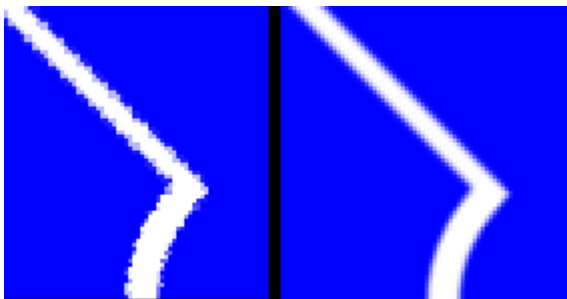
Linux-käyttöjärjestelmässä on tekstimuotoinen toimintatila, johon voidaan syöttää komentoja. Sama tekstinäkymä on saatavilla myös graafisen käyttöliittymän puolella omana ohjelmanaan. Tätä tekstimuotoista toimintatilaa kutsutaan terminaaliksi. Aluksi horisonttia suunniteltiin piirrettävän suoraan terminaaliin tekstimuotoisena oman koodin avulla, mutta tässä tavassa oli muutamia ongelmia. Ensinnäkin tekstimuotoinen esittäminen ei ole nykyään havainnollisin tapa, eikä tällä tavalla saavutettu esitystarkkuus olisi ollut riittävä. Toisin sanoen varsinkin lentokoneen kallistamisen esittäminen olisi ollut hankalaa, koska terminaalissa ei varsinaisesti voi kiertää tekstiä, vaan kallistaminen olisi pitänyt hoitaa vaihtamalla piirrettävää riviä. Kuten kuvasta 3 voi havaita, eri asentoja olisi mielekkäästi voinut esittää noin viisi, tai jos koneen siivet olisi piirretty tavuviivoilla, olisi alaviivojen avulla saatu nostettua esitettävien asentojen määrä yhdeksään. Kuvassa on esitetty tämä tarkempi tapa, jolla asentoja on käytettävissä neljä suuntaansa ja tämän lisäksi kone voi olla suorassa. Kuvassa konetta kallistetaan vasemmalle.



Kuva 3. Terminaalissa esitetty lentokoneen kallistus suorasta asennosta ääriasentoon vasemmalle.

Koska tekstimuotoinen esittäminen osoittautui hieman huonoksi, selvitettiin, olisiko horisontti mahdollista piirtää kuvien avulla, joita pyöritetään akseliensa ympäri. Tässä tavassa etuna olisi havainnollisempi ja tyylikkäämpi esitystapa, koska ulkonäkö määräytyisi käytettävän kuvan mukaan. Lisäksi kuvia voisi kiertää huomattavasti vapaammin kuin tekstiä, jolloin lentokoneen asennon voisi esittää tarkemmin.

Aluksi kokeiltiin piirtää itse summittainen kuva halutusta keinohorisontista ja pyörittää sitä GIMP-kuvankäsittelyohjelmassa (18), jotta nähdään, kuinka kuva käyttäytyy pyöritettäessä. Nopeasti kuitenkin selvisi, että bittikarttakuvan pyörittäminen mihin tahansa muuhun kulmaan kuin yhdeksälläkymmenellä jaollisiin lukuihin johti kuvanlaadun heikkenemiseen. Bittikarttakuvan käyttäminen horisonttina ei tuntunut parhaalta vaihtoehdolta, kun otetaan huomioon heikko kuvanlaatu ja Raspberry Pin heikko laskentateho. Kuvassa 4 näkyy osasuurennos tehdystä pyöritystestistä, jossa on pyöritetty sekä vektori- että bittikarttakuvaa 45 astetta. Kuvasta nähdään, kuinka pyöritys on huonontanut merkittävästi bittikarttakuvan laatua.



Kuva 4. Vasemmalla 45 astetta kierretty bittikartta-, oikealla vektorikuva. Kuvia on suurennettu 200 %, jotta erot näkyvät selkeämmin.

Vektorikuvat ovat matemaattisesti luotuja kuvia, joita voidaan suurentaa ja pyörittää näytöllä vapaasti kuvan laadun heikkenemättä. Vektorikuvat tallennetaan tekstimuotoisena tiedostona, joka sisältää viivojen ja pisteiden koordinaatit. Tämän vuoksi vektorikuvat ovat käyttökelpoisia lähinnä yksinkertaisissa muodoissa, kuten logoissa tai kartoissa, joiden muodot voidaan selkeästi muuttaa kaariksi ja viivoiksi (19). Monet kartta-palveluiden tarjoajat, kuten Google ja Apple, siirtyivät muutama vuosi sitten käyttämään kartoissaan vektorigrafiikkaa (20). Tämä mahdollisti karttojen sujuvamman liikuttelun sekä lähentämisen ja loitontamisen aiempaa sulavammin. Ennen kaikkea karttoja kuitenkin pystyi muutoksen myötä pyörittämään vapaasti, mitä aiemmilla bittikarttakuviin perustuvilla kartoilla olisi ollut vaikea toteuttaa. Vektorikarttoihin siirtymisessä oli muitakin etuja, mutta ne eivät ole merkityksellisiä tämän työn kannalta. Kartat olivat kuitenkin hyvä osoitus siitä, miten kuvanlaadullisesti korkeatasoinen pyöritys on mahdollista toteuttaa.

Vektorikarttojen innoittamana selvitettiin, olisiko mahdollista toteuttaa myös keinohorisontti vektorigrafiikan avulla eli riittääkö Raspberry Pin teho sen käsittelyyn. Ensimmäisenä vaihtoehtona löytyi lupaavan oloinen QT-kirjasto, jonka avulla pystytään hyödyntämään Raspberry Pin näytönohjainta grafiikan käsittelyyn. Kirjaston avulla on saatu jopa tehtyä yksinkertaisia 3D-pelejä, joten kirjastoa päätettiin kokeilla käytännössä. (21.)

Ennen kuin kirjastoa kuitenkaan pääsi kokeilemaan, oli vektorigrafiikka luotava jollain keinolla. Internetistä (22) löytyi vapaasti käytettävissä olevia toteutuksia mittareista, joita päätettiin käyttää kokeilussa hyväksi. QT-kirjasto ladattiin Raspberry Pille ja alettiin kääntää sitä ohjeiden mukaan (23). Kirjasto täytyy kääntää sillä tietokoneella, jolla sitä aiotaan käyttää, jotta siihen tulee mukaan tarvittavat osat juuri kyseiselle kokoonpanolle. Teoriassa on mahdollista kääntää ohjelma tai kirjasto myös eri koneella, kuin missä sitä tullaan käyttämään, asettamalla sopivat parametrit käännösvaiheessa. Tätä operaatiota kutsutaan ristikäntämiseksi (englanniksi cross compiling). Tällaisen käännösympäristön asettamisessa on kuitenkin omat ongelmansa, joten kirjasto päätettiin kääntää suoraan Raspberry Pillä. Ristikääntämisessä on muun muassa monimutkainen asennus ja kirjastojen lisäys käännösympäristöön (24). Ainoa konkreettinen hyöty on ohjelmien nopeampi käännösaika, mutta koska käännösympäristön asennus vie runsaasti aikaa, jää kokonaishyöty melko pieneksi.

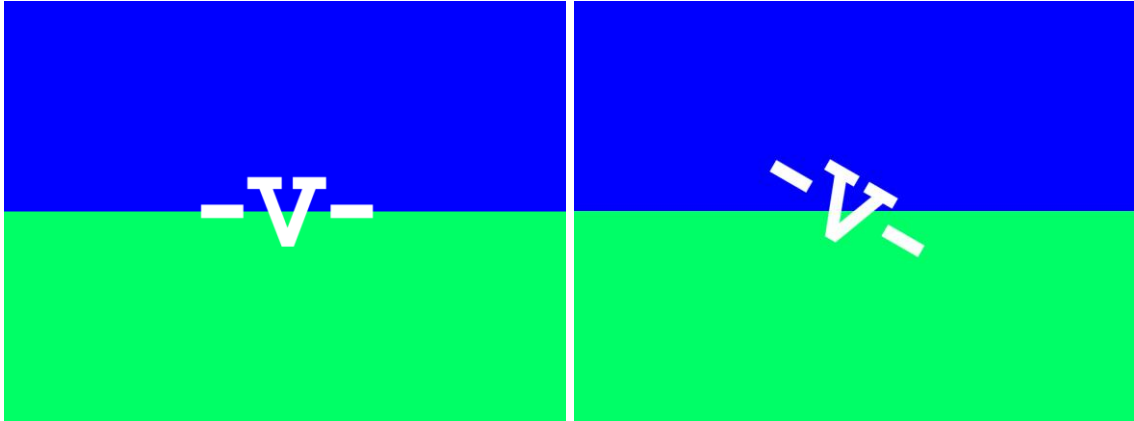
Keskustelualueiden ja ohjeiden (25) mukaan kääntämisessä saattaa kestää useita päiviä heikon prosessoritehon vuoksi. Muutama tunti kääntämisen aloittamisen jälkeen tuli kuitenkin toinen ongelma vastaan. Raspberry Pin muistikortti tuli täyteen, eikä kääntämistä voinut tämän vuoksi enää jatkaa. Vaihtoehtona olisi ollut ostaa suuremmalla tallennustilalla varustettu muistikortti, mutta kirjaston monimutkaisuus ja laajuus alkoi epäilyttää muutenkin, sillä käytössä oli jo melko suuri, kahdeksan gigatavun muistikortti. Ongelmaan alettiin etsiä uudelleen ratkaisua, mutta tällä kertaa haluttiin löytää mahdollisimman yksinkertainen kirjasto. Lopulta lupaavimmaksi osoittautui OpenVG-kirjasto (26).

OpenVG-kirjaston avulla voidaan piirtää vektorigrafiikkaa hyödyntäen näytönohjainta, joten suorituskyky pysyy hyvänä. Githubiin ladattu ajstarks-nimimerkin tekemä OpenVG oli huomattavasti pienempi ja yksinkertaisempi paketti kuin ensin kokeiltu QT, joten sen lataus ja kääntäminen onnistui alle kymmenessä minuutissa. Paketin mukana on muutamia valmiiksi tehtyjä ohjelmia, joilla voi kokeilla helposti kirjaston mahdollisuuksia ja suorituskykyä. Näiden demo-ohjelmien perusteella kirjasto vaikutti työhön sopivalta.

Seuraava askel oli kokeilla keinohorisontin piirtämistä kirjaston avulla. Aiemmin mainituista valmiista mittaristopohjista luovuttiin ja päätettiin tehdä omat mittarit alusta asti. Alun perin ajatuksena oli vain kokeilla toimintaa nopeasti omalla mittarilla, mutta myöhemmin selvisi, että valmiit mittarit olivat liian monimutkaisia ja täten raskaita pyöriäkseen yhtä aikaa koko muun koodin kanssa. Näin ollen omatekoiset mittarit olivatkin välttämättömät.

Ensimmäiseksi yritettiin piirtää yksinkertaisesti muutama merkki, joita pyöritettiin ja liikuteltiin taustaa vasten. Tämä testi osoitti, että OpenVG oli sopiva kirjasto työn tarkoitukseen. Suorituskyky ja grafiikan laatu olivat hyviä. Kuvassa 5 on testeissä käytetty versio keinohorisontista. Tämä versio oli esillä myös ensimmäisessä esittelyssä tilaajalle, jolloin tilaaja sai esittää muutostoiveita ulkonäköön.





Kuva 5. Keinohorisontin ensimmäinen testiversio. Vasemmalla lennokka on suorassa ja oikealla kallellaan 30 astetta oikealle.

Kuten kuvasta 5 näkee, päätettiin lennokin asento esittää keinohorisontissa siten, että tausta pysyy paikallaan ja lennokin siluettia liikutellaan sen päällä. Oikeissa lentokoneissa olevissa keinohorisonteissa tieto esitetään toisin päin, eli horisontti liikkuu lentokoneen siluettin pysyessä paikallaan. Toisenlaiseen ratkaisuun päädyttiin, koska tämä tuntui havainnollisemmalta tavalta näyttää lennokin asento. Oikeassa lentokoneessa lentäjä näkee horisontin tuulilasin läpi siten, että se näyttää kallistuvan lentokoneen kojelaudan pysyessä suorassa. Näin ollen on luontevaa, että myös mittari näyttää tilanteen samalla lailla. Lennokin tapauksessa lennättäjä sitä vastoin näkee lennokin kallistuvan horisontin pysyessä suorassa. Tätä tilannetta pyrittiin esittämään myös keinohorisontilla.

Maa-asemalla esitetään myös lennokinpuoleisen tiedonkeruujärjestelmän akun jännite-taso, jotta akun loppumiseen osataan varautua ajoissa. Tieto esitetään kahden desimaalin tarkkuudella ja lisäksi tekstin väriä vaihdetaan vihreästä keltaisen kautta punaiseen akun varaustason laskiessa. Lennokissa on käytetty LiFePo<sub>4</sub>-akkaa, jonka jännite laskee melko hitaasti, kunnes se romahtaa hieman ennen tyhjenemistään. Tämän vuoksi varoitusvärit ovat vain suuntaa-antavia ja käytännössä akun varauksesta saa parhaan käsityksen ottamalla aikaa, kuinka kauan järjestelmä on ollut käynnissä. Akun jännitteen ollessa yli 11,16 voltia näytetään vihreää. Akun jännitteen laskiessa alle tämän näytetään keltaista, ja jännitteen edelleen laskiessa alle 10,45 voltin näytetään punaista (27). Punaisen värin lisäksi jännitetiedon yläpuolelle piirretään teksti, joka kehottaa lataamaan akun välittömästi.

## 2.4 Videokuva

Keinohorisontin oli tarkoitus näkyä yhdessä videon ja lennokin tiedonkeruujärjestelmän akun jännitteen kanssa maa-aseman kuvaruudulla. Jännite ja horisontti piirretään molemmat OpenVG-kirjaston avulla, mutta video piirretään erillisellä, Raspberry Pin Raspbian-käyttöjärjestelmästä vakiona löytyvällä Omxplayer-videosoittimella. Omxplayerin etuna on, että se purkaa videota näytönohjaimella, mikä mahdollistaa sulavan videotoiston. Näiden ohjelmien yhtäaikainen käyttö kuitenkin aiheutti ongelmia.

Ongelmana oli, että joko video tai itse tehdyn ohjelman elementit peittivät koko kuvaruudun, riippuen siitä, kumpi ohjelmista käynnistettiin ensin. Aluksi omaa ohjelmaa yritettiin rajata siten, että se piirtyisi vain rajatulle alueelle. Vaikka ohjelman piirrettävät elementit olivat melko pieniä kooltaan, täytti ohjelma silti koko ruudun, koska sille täytyi määritellä taustaväri. Tämä taustaväri piirtyi koko ruudulle. Väriksi valittiin musta, jotta se ei häiritsisi käyttäjää.

OpenVG:n piirtoalueen rajauksen osoittauduttua mahdottomaksi päätettiin sen sijaan yrittää rajata videota. Se onnistui antamalla toisto-ohjelmalle haluttu ikkunan koko parametrina. Jos ensin käynnistetään mittarit piirtävä ohjelma ja sen jälkeen videota toistava ohjelma, peittää video osan mittarit piirtävästä ohjelmasta, jolloin molemmat saadaan näkymään yhtä aikaa.

## 2.5 Lennokki

Lennokin päässä piti tehdä valinta NUC-tietokoneen ja minitietokoneen välillä. Lopulta lennokin päässä päädyttiin samaan ratkaisuun kuin maa-aseman päässä eli Raspberry Pihin. Sen ominaisuudet riittivät myös lennokin päähän, joskin siitä puuttui sekä jännitteen mittausmahdollisuus että kiihtyvyyssanturi. Nämä kuitenkin pystyttiin lisäämään suunnittelemalla oma piirilevy, johon toiminnollisuudet pystyi lisäämään. Piirilevy suunniteltiin peittämään Raspberry Pi alleen, ja tietojenvälitys piirilevyjen välillä onnistui GPIO-liitännän avulla (17). Piirilevyjen kehitys oli Ari Rajamäen vastuulla, ja se käsitellään hänen työssään.

Lennokki on Puolustusvoimien maalilennokki, johon ei voinut tehdä muutoksia, mikä aiheutti tiedonkeruujärjestelmän kiinnittämisen suhteen haasteita. Lopulta lennokkiin

kuitenkin saatiin erillinen levy, joka tulee lennokin rungon ja irrotettavan siiven väliin. Tähän levyyn sai tehdä minkäläisen kiinnityksen tahansa, koska se ei ollut osa lennokkia.

Lennokkiin suunniteltiin ja tulostettiin muoviset kotelot mobiilidatayhteyttä, kameraa ja itse keskusyksikköä varten. Osien suunnittelussa otettiin huomioon lennokin suuret kiihtyvyydet noustessa ja laskeutuessa sekä säänkestävyys. Lisäksi osien täytyy kestää sekä pölyä että vesisadetta, mutta niiden pitää kuitenkin olla avattavissa. Kameran kotelo on ainoa, joka on silikonilla kiinni ja täten sitä ei voi avata.

Modeemissa on tavallinen dataliittymä, joten siihen täytyy päästä käsiksi SIM-kortin vaihtoa varten. Tätä varten kotelo suunniteltiin siten, että sen voi avata irrottamalla kulmissa olevat neljä ruuvia. Myös keskusyksikön kotelo on kulmistaan ruuveilla kiinni, joten myös sen pystyy avaamaan huoltoa varten. Kameralle ei voi tehdä mitään korjauksia, ja näin ollen sen kotelo suunniteltiin ennen kaikkea vesitiiviiksi.

## 2.6 Tietoliikenne

Maa-aseman ja lennokin välille piti saada radiolinkki, jonka kantama olisi viisi kilometriä, ja videon tulisi olla tarkkuudeltaan vähintään 1280 x 720 kuvapistettä eli teräväpiirrotasoista. Pelkkä korkea tarkkuus ei kuitenkaan riitä, vaan myös pakkauksen tulee olla riittävän lievää, jotta hyvä tarkkuus ei peity pakkaushäiriöihin. Toivomuksena oli myös, että radiolinkki olisi salattu, jotta ulkopuoliset eivät pystyisi katsomaan videota omilla laitteillaan. Toivottu kantama oli kuitenkin erittäin vaikea saavuttaa, koska kuluttajille avoimet taajuudet ovat tarkasti säädeltyjä lailla (17). Käytännössä ainoa soveltuva taajuusalue on 2,4 GHz enintään 100 milliwatin lähetysteholla, mutta tällä taajuudella on vaikea toteuttaa pitkän kantaman radiolinkkiä, sillä 2,4 GHz:n taajuus vaimenee voimakkaasti etäisyyden kasvaessa (17).

Erilaisten radiolinkkien selvityksen jälkeen (17) päädyttiin tutkimaan WLAN-linkkejä tarkemmin, koska WLAN tarjoaa valmiin ratkaisun salaukseen, jota on helppo käyttää. Lisäksi modeemin valmistaja on jo huolehtinut, ettei korkeinta sallittua lähetystehoä ylitetä. WLAN-verkon kantama oli kuitenkin liian lyhyt, joten vaihtoehtoja piti tutkia lisää.

Lisätutkimuksen jälkeen kaikki tavoitteet täyttäväksi tekniikaksi todettiin kaupallisen neljännen sukupolven mobiilidataverkon (4G) käyttäminen suojatun VPN-tunnelin kanssa (Virtual Private Network) (17). VPN mahdollisti tehokkaan salauksen käytön, ja kaupallinen mobiiliverkko tarjosi riittävän tiedonsiirto kapasiteetin ja kantaman. Kantama nousi tulokinnasta riippuen koko internetin laajuiseksi, koska VPN-tunnelin avulla videota pystyy katsomaan missä tahansa, kunhan tiedonsiirtonopeus vain on riittävä.

Videon siirrossa käytettiin MotionJPEG-pakkausta vähemmällä bittivirralla pärjäävän h.264-pakkauksen sijaan, koska MJPEG toimii vaihtelevassa verkkoympäristössä paremmin. Toisin sanoen h.264-pakkauksen tehokkuus perustuu siihen, ettei jokainen videon ruutu sisällä kaikkea informaatiota, jolloin yhden ruudun katoaminen matkalla saa koko videon korruptoitumaan hetkeksi. Näin ollen kokonaislaatu saatiin tasaisemmaksi ja paremmaksi käyttämällä tehottomampaa pakkausta.

Teräväpiirtotasaisen videon toisto edellytti testien mukaan (17) vähintään yhden megabitin tiedonsiirtonopeutta sekunnissa. h.264-video olisi teoriassa ollut parempilaatuinen tällä tiedonsiirtonopeudella, mutta käytännössä se ei ollut käyttökelpoinen 4G-verkon häiriöiden vuoksi. Häiriöitä voisi verrata digitaalisen television häiriöihin heikossa kentässä, jolloin kuvassa näkyy palikoitumista ja pätkimistä. Osa digitaalisen television häiriöistä johtuu tosin liian voimakkaasta pakkauksesta, mikä ilmenee etenkin paljon yksityiskohtia sisältävissä kohdissa tai liikettä sisältävissä kohtauksissa (28). MJPEG-pakkauksen avulla videosta saatiin lopulta riittävän hyvälaatuinen, eikä siinä ole havaittavissa palikoitumista. Videokuvan siirrosta voi lukea lisää Ari Rajamäen työstä (17).

Lennoista piti videon lisäksi saada asento- ja jännitetieto maa-asemalle, mutta telemetriatiedot ovat tekstipohjaisuutensa vuoksi erittäin vähän tiedonsiirtokaistaa tarvitsevia ja ne kulkevatkin vaivattomasti videon rinnalla. Molempien tietojen välittämiseen käytetään itse suunniteltua ohjelmistoa. Akun jännitetieto saadaan SPI-väylän kautta.

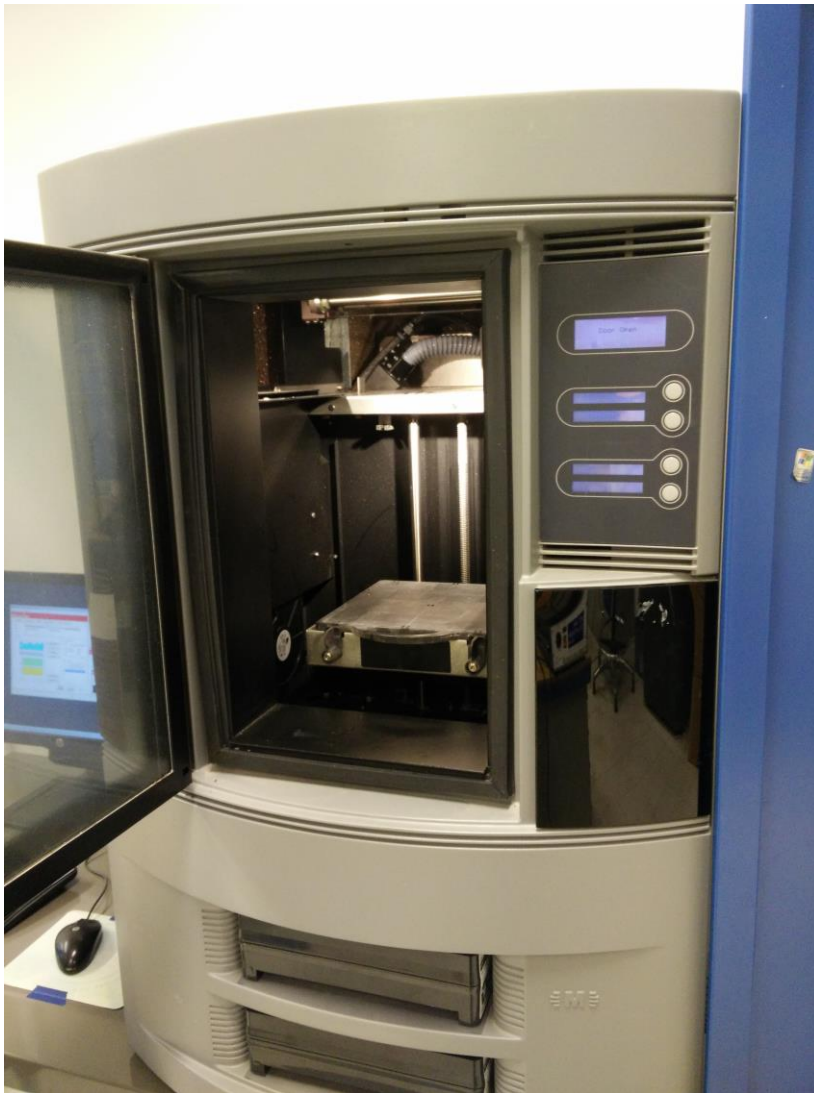
Akun jännitteen mittausta ja lennokin asentotietoa varten tarvittiin jonkinlaista anturointia. Tätä varten suunniteltiin piirilevy, johon sijoitettiin tarvittavat komponentit (17). Näiltä komponenteilta saatiin tieto raakamuodossa ulos. Tiedot piti muokata käyttäjälle luettavampaan muotoon, minkä lisäksi käytettiin suodatusta, jotta häiriöt mittausdatassa eivät näkyisi käyttäjälle.

### 3 Koteloiden tulostus ja ohjelmiston kehitys

#### 3.1 3D-tulostus

3D-tulostus on muutaman viime vuoden aikana saavuttanut suuren suosion. Tähän on syynä patenttien raukeaminen ja näin ollen kilpailun lisääntyminen ja hintojen lasku. Nykyään tulostimia saakin jo muutamalla sadalla eurolla ja hieman laadukkaamman laitteen saa hieman yli tuhannella eurolla (29; 30). Laitteiden käyttö on myös helpottunut, mikä nostaa tulostimien suosiota entisestään. Valmiita malleja on internetissä runsaasti ja esimerkiksi suosituimmille puhelinmalleille on tehty valmiita 3D-tulostettavia malleja aina suojakuorista erilaisiin telineisiin. Valmiita malleja voi ladata esimerkiksi [thingiverse.com](http://thingiverse.com)- ja [grabcad.com](http://grabcad.com)-sivustoilta.

PLA eli polylaktidi on biomateriaalista valmistettu muovilaatu. PLA-muovia voidaan valmistaa esimerkiksi sokeriruo'osta tai maissitärkkelyksestä (31). Se on kovempaa mutta hauraampaa kuin raakaöljystä valmistettu ABS-muovi eli akryylinitriilibutadieenisyreeni (32; 33). PLA-muovi murtuu kovassa rasituksessa helpommin verrattuna ABS-muoviin, joka venyy ja joustaa enemmän. 3D-tulostuksessa PLA-muovi on hieman ABS-muovia helpompi materiaali, koska se ei vääntyile niin helposti tulostuksen aikana. ABS-muovi vaatii korkeamman tulostuslämpötilan kuin PLA-muovi, ja sen tulostaminen vaatii lämpösäädellyn kaapin, jollainen on tässäkin insinööriyössä käytetyssä Stratasyksen Dimension tulostimessa (kuva 6). Jos ympäröivän ilman lämpötila on liian alhainen, ABS-tuloste irtoaa helposti tulostusalustasta ja muuttaa muotoaan tulostuksen aikana. Tulosteeseen saattaa myös syntyä repeämiä, jotka johtuvat tulosteen osittaisesta ja epätasaisesta jäähtymisestä. (34.)



Kuva 6. Stratasys Dimension -3D-tulostin, jossa on suljettu tulostusalue tarkempaa lämmön-säätelyä varten.

Yleensä edullisemmat, kuluttajakäyttöön tarkoitetut tulostimet käyttävät PLA- tai ABS-muovia, joista PLA on helpompi tulostettava, koska se ei tarvitse niin tarkkaa lämmön-hallintaa. Myös hartsia tulostavia malleja on, mutta ne ovat kalliimpia, eivätkä hartsitu-losteet ole kokemusten mukaan yhtä kestäviä kuin muoviset. Hartsin käsitteleminen on myös haasteellisempaa, koska se vaatii huolellista työskentelyä kemikaalien kanssa (29).

Liikkuvien osien, kuten vaihteiston, valmistaminen on mahdollista valikoivaa laser-sintraustekniikkaa (Selective Laser Sintering (SLS)) käyttävällä tulostimella. SLS-tulostimet levittävät pulverimaista muovia (polyamidi-12) ohuen kerroksen koko tulos-tettavalle alueelle, ja lasersäde kuumentaa ja sulattaa kerroksesta ne alueet, joihin

halutaan kiinteää materiaalia. Tekniikka mahdollistaa monimutkaisten muotojen tulostamisen, koska pulveri tukee kappaletta tulostuksen aikana, jolloin kappale ei pääse valumaan muodottomaksi ennen jähmettymistään. Ylimääräinen muovipulveri puhalletaan pienillä lasikuulilla lopuksi pois, jolloin pinnasta tulee sileä. SLS-tulostimia ei ole juurikaan kuluttajakäytössä niiden kalliin hinnan ja tulostamisen sotkuisuuden vuoksi. Muovipulveri on hienojakoista, ja se põlisee helposti. (35.)

Edullisimmat ja täten yleisimmät tulostimet ovat pursotustekniikkaan (Fused deposition modeling (FDM)) perustuvia laitteita, jotka pursottavat sulaa muovia ohuina säikeinä kerros kerrokselta tulostusalustalle. Tulostimissa on moottori, joka vetää lankaa rullalta ja työntää sitä kuumaan suuttimeen. Muovi sulaa suuttimen sisällä ja tulee ulos ohuena sulana nauhana pienestä suuttimen reiästä. Muovisäikeet tarttuvat ja jähmettyvät toisiinsa kiinni jäähtyessään, ja lopputuloksena on kiinteä ja vahva muovikappale. Yksittäinen kerros on hyvin ohut, parhaissa tulostimissa vain 20 mikrometriä eli 0,02 millimetriä.

Ohuen kerrospaksuuden ansiosta tulostimilla saa tehtyä tarkkojakin tulosteita, mutta parhaalla laadulla pienenkin esineen tulostaminen kestää kauan. Itse tulostimet eivät sisällä älyä, vaan ne vain toteuttavat tarkasti saamiaan ohjeita. Ohjeet sisältävä määrittiedosto tehdään sopivalla ohjelmalla tietokoneella, ja syntyvä tiedosto siirretään tulostimelle muistikortilla. Jotkin tulostimet ovat jatkuvasti liitettynä tietokoneeseen USB-johdolla, jolloin tietokone antaa tulostuksen edetessä käskyjä tulostimelle. Määrittiedosto luodaan eri ohjelmilla eri tulostimille. Tällaisia ohjelmia ovat esimerkiksi Ultimakerin käyttämä Cura ja MiniFactoryn käyttämä Repetier-Host (36).

Kappaleet tulostetaan yleensä osittain onttoina, mikä säästää sekä tulostusmateriaalina käytettävää muovia että tulostuksen vaatimaa aikaa. Kappaleen sisärakenteet tulostetaan ristikkorakenteena, joka on melko lujaa. Tämän ristikkorakenteen saa luotua automaattisesti määrittiedoston tekevällä ohjelmalla. Rakenteen tiheyttä voi säätää ohjelman asetuksista, mikä vaikuttaa suoraan kappaleen jäykkyyteen, mutta luonnollisesti myös tulostusaikaan (36). Tässä työssä on käytetty 30 prosentin täyttöastetta, mikä tarjoaa riittävän kestävyuden ja kohtuullisen tulostusajan.

Itse 3D-malli suunnitellaan erillisellä ohjelmalla, joka yleensä vieään STL-tiedostomuodossa vaikkapa Curaan. Cura muuttaa STL-tiedoston mallin tulostimen ymmärtämään muotoon ja samalla se analysoi, onko malli tulostettavissa. Ohjelmalla

voi myös lisätä automaattisesti tukirakenteita, jotka poistetaan tulosteen ollessa valmis. Osassa tulostimia on kaksi tulostuspäätä, joista toisella voidaan tulostaa tukimateriaalia. Erillinen tukimateriaali on vesiliukoista PLA-muovia, joten sen poistaminen valmiista mallista viimeistelyvaiheessa on helpompaa (36). Tulostaminen ei onnistu tyhjän päälle, vaan uusi kerros voidaan tulostaa vain vanhan päälle.

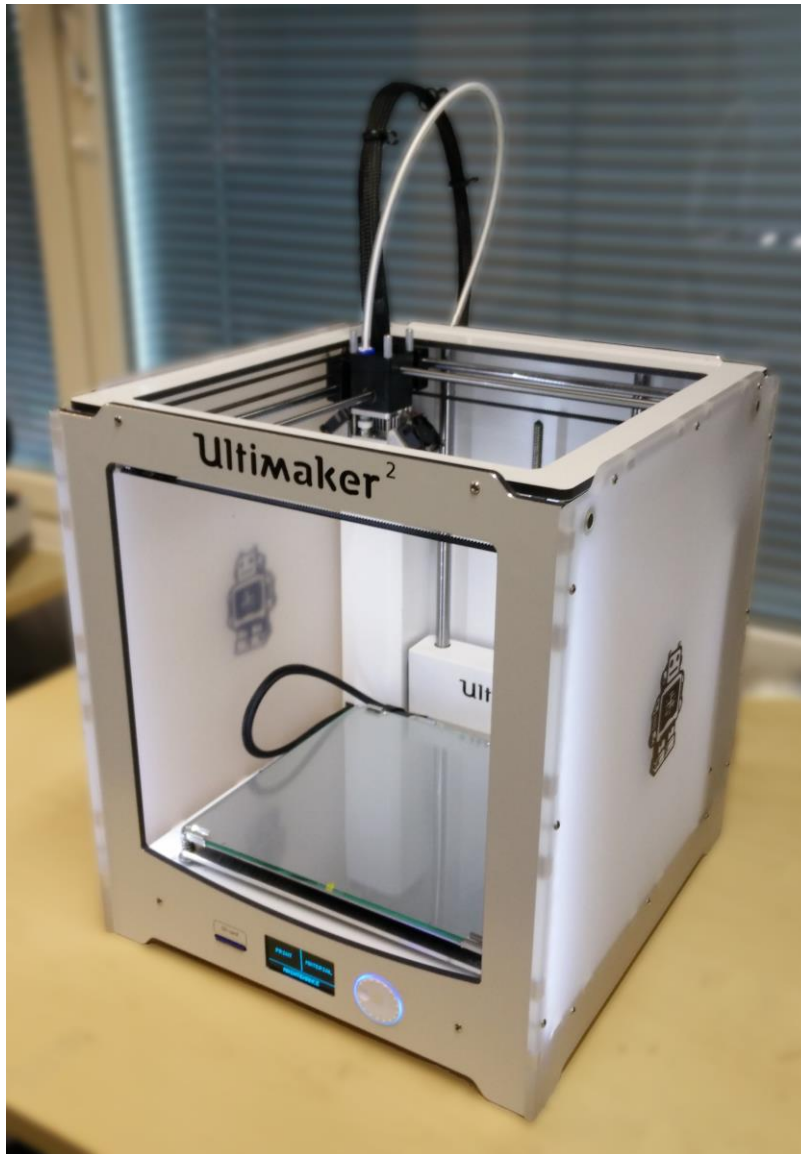
Tyypillisesti 45 astetta on suurin kulma, jossa ulkonevien rakenteiden tulostaminen onnistuu. Tätä jyrkemmät kulmat vaativat tuen, jotta muovi ei valuisi muodottomaksi. Kuitenkin myös yhden tulostuspään tulostimissa on mahdollista käyttää tukirakenteita. Ne ovat irrotettavissa joko sormin tai puukolla, riippuen kappaleen tukirakenteiden pak-suudesta (36).

3D-tulostimia on hankittu muutamiin pääkaupunkiseudun kirjastoihin, joissa ne ovat vapaasti jokaisen kirjaston asiakkaan käytettävissä. Tulostimien käyttöön saa myös tarvittaessa apua kirjastossa, joten alkuun pääsee helposti. Tulostimen ominaisuuksia on hyvä testata tulostamalla aluksi jokin valmis malli. Ainakin toistaiseksi 3D-tulostaminen on kirjastoissa ilmaista, joten kokeilun kynnyks on matala. (37.)

Tässä työssä käytetyt 3D-mallit on tulostettu samanlaisella Ultimaker-merkkisellä (kuva 7) tulostimella, kuin mikä on ainakin Sellon ja Ison Omenan kirjastoissa Espoossa (37). Tällainen tulostin sattui löytymään valmiiksi Ari Rajamäen työpaikalta, ja sen käyttö oli vapaampaa kuin kirjastoissa. Osa työssä käytetyistä malleista on sen verran isokokoisia, ettei niiden tulostaminen olisi onnistunut kirjastossa.

Kirjastossa pisin tulostamiseen käytettävissä oleva aika on työpäivän pituus, eli yön yli tulostinta ei voi jättää työskentelemään. Isoimman tässä työssä käytetyn mallin tulostus kesti jopa 60 tuntia, joten oman tulostimen käyttö oli pakollista. Joitakin malleja myös tulostettiin useaan kertaan, koska niihin haluttiin tehdä muutoksia joko suunnitteluvirheen tai jonkin muun syyn takia. Useiden viikkojen yhtäjaksoinen työskentely suurten mallien parissa ei olisi ollut mahdollista kirjaston laitteilla, sillä tulostimet on tarkoitettu tasapuolisesti kaikkien käyttöön.



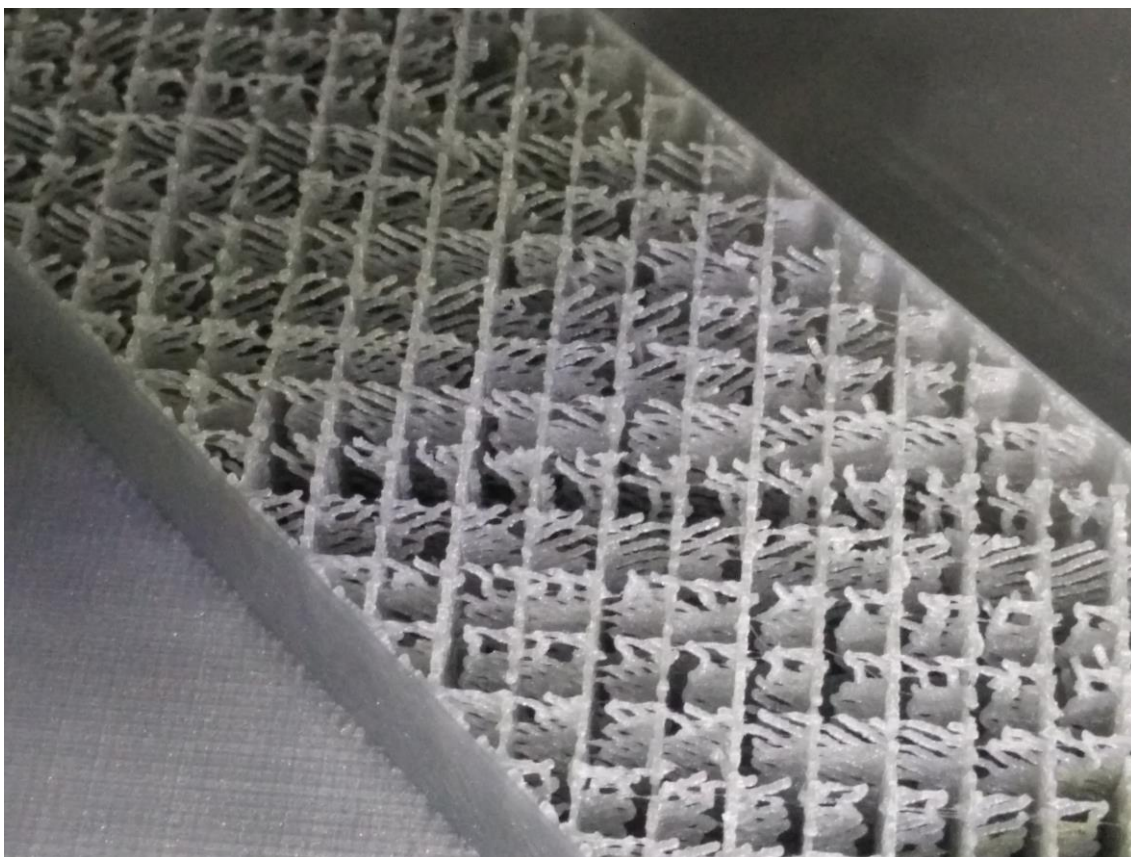


Kuva 7. Ultimaker-3D-tulostin, jolla tulostettiin suurin osa työn malleista.

Kuten kuvasta 7 havaitaan, on Ultimaker-3D-tulostimen tulostusalue avonainen, jolloin ainoa tulostettavaa kappaletta lämmittävä osa on tulostusalusta. Tämä ei ole ongelma pienillä kappaleilla eikä suuremmillakaan, jos materiaalina käytetään PLA-muovia. ABS-muovi kuitenkin tarvitsisi lämpimämmän tulostusalueen, eikä sen tulostaminen onnistu kokeilujen perusteella kunnolla tällä tulostimella.

Työssä käytetyssä 3D-tulostimessa oli heti uutena viallinen materiaalin syöttömoottori, joka ei jaksanut työntää materiaalia riittävästi suuttimen läpi. Tämän vuoksi tulosteet joko epäonnistuivat kokonaan tai jäivät huonolaatuisiksi ja heikoiksi. Etenkin tulosteiden kennorakenne jäi helposti hauraaksi liian vähäisen muovin määrän takia, kuten

kuvasta 8 voi huomata. Ongelmaa pystyi lieventämään hidastamalla tulostusnopeutta, jolloin muovivaursotettiin hitaammin suuttimen läpi. Hidastuksen haittapuolena melko pienenkin kappaleen tulostaminen kesti kymmeniä tunteja. Jos mallintamisessa oli tapahtunut virhe tai tulostus epäonnistui muusta syystä, se hidasti merkittävästi työn etenemistä, koska tulostin ei ollut pelkästään tämän työn käytössä.

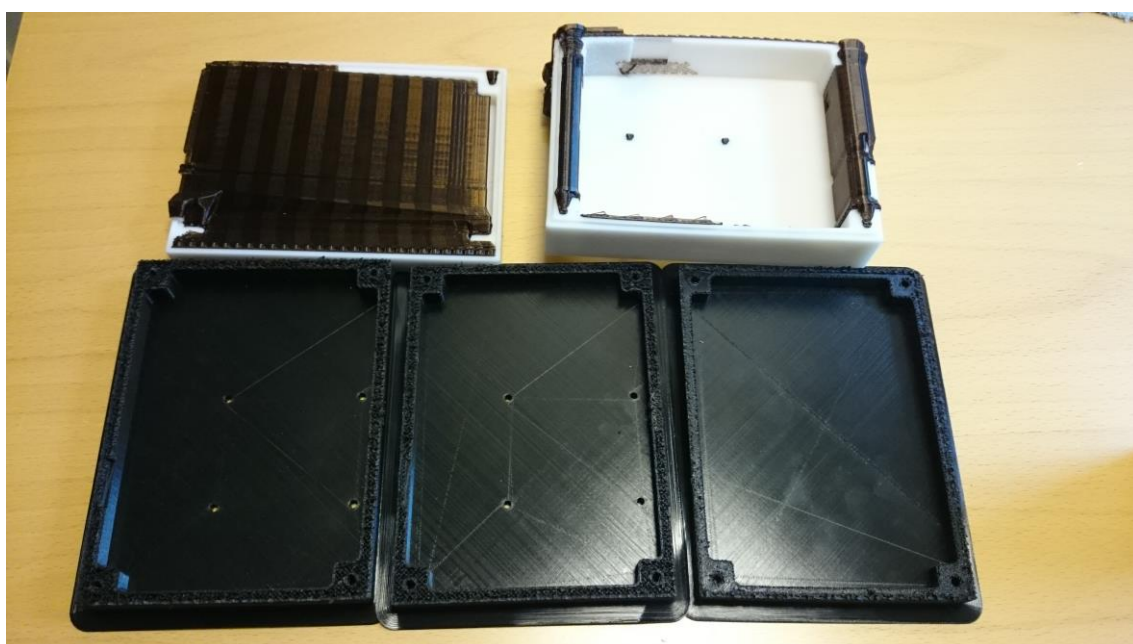


Kuva 8. Epäonnistunut kennorakenne liian vähäisen muovin syötön takia.

Tulostimella yritettiin tulostaa koekappale, jossa tulostusnopeutta kiihdytetään mallin edetessä. Testi osoitti, että ongelmat alkavat noin 7 kuutiomillimetriä sekunnissa nopeudella ja tätä nopeammat vauhdit ovat käyttökelvottomia. Suurin testissä käytetty nopeus oli  $10 \text{ mm}^3/\text{s}$ , jonka pitäisi onnistua käytetyllä Ultimaker-tulostimella. Käytössä olleella tulostimella tämä nopeus oli kuitenkin liikaa, ja tuloste oli täynnä reikiä. Kaikki työssä käytetyt 3D-mallit on tästä syystä tulostettu hidastetulla vauhdilla ja tiheämmällä kennorakenteella, jotta tulosteista saatiin riittävän kestäviä.

Loppuvaiheessa työtä, kun jäljellä oli enää varakoteloiden tulostaminen, tulostimen suutin meni osittain tukkoon eikä sitä saatu edes purkamalla enää kuntoon. Tulosti-

meen tilattiin varaosana uusi suutin, mutta se ei ehtinyt tulla ajoissa työn luovutusta ajatellen. Suutin ei mennyt täysin tukkoon, mikä hidasti vian toteamista. Kappaleet onnistuivat osittain, mutta jossain vaiheessa suuttimesta ei enää tullut riittävästi materiaalia. Tämä johti keskeytyneisiin tulosteisiin, joita on kuvassa 9. Varakappaleet tulostettiin Metropolian Tikkurilan-toimipisteen Stratasys Dimension -tulostimella. Tällä tulostimella tulostetaan ABS-muovia, eikä ontto tulostus ole tuettuna. Hyvänä puolena tulostimella voidaan tulostaa erillistä tukimateriaalia, jolloin sillä voidaan tulostaa monimutkaisempia esineitä. Työn kannalta tällä ei tosin ollut merkitystä, koska kappaleet oli suunniteltu tulostettaviksi Ultimaker-tulostimella eli ilman tukimateriaalia. Stratasyksen tulostusalue on suljettu kaappi, mikä auttaa ABS-tulosteiden onnistumista.



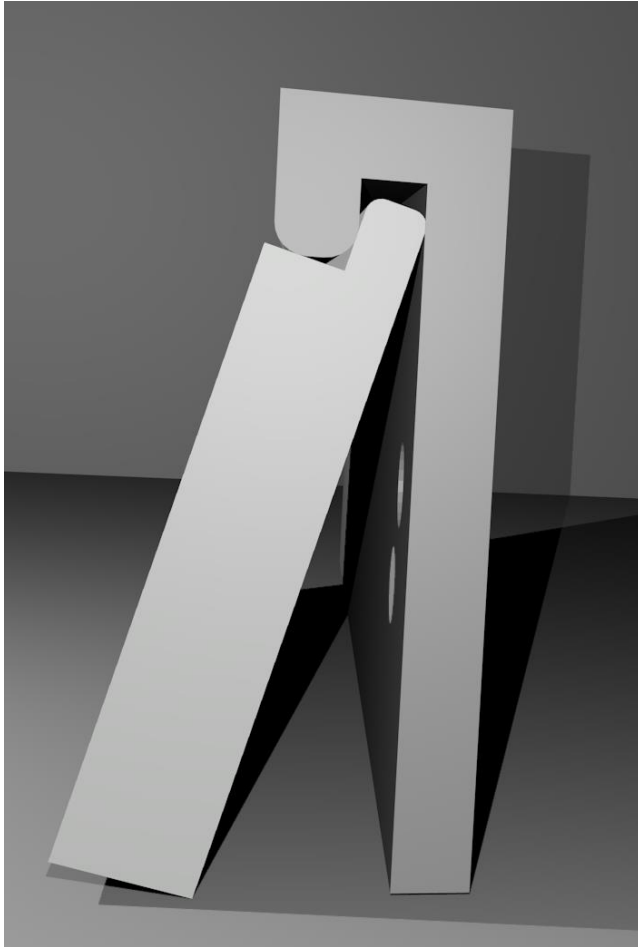
Kuva 9. Epäonnistuneita tulosteita. Mustat, etualalla olevat kappaleet on tulostettu Ultimakerilla ja valkoiset Stratasyksellä. Valkoisissa tulosteissa olevat tummat osat ovat tukimateriaalia.

Osa kappaleista epäonnistui myös Stratasyksen tulostimella, mikä johtui siitä, että tulostamisen nopeuttamiseksi kappaleet tulostettiin pystyasennossa. Tämä johti siihen, että jos tukimateriaali on vähänkään irti kappaleesta, alkaa kappale valua muodottomaksi. Myös ABS-muovin herkkä taipumus käyristyä tulostusvaiheessa oli varmasti osasyllinen tulostuksen epäonnistumiseen. Tämä näkyy kuvassa 9 yläoikealla olevassa valkoisessa kappaleessa, jossa vasen reuna on litistynyt.

Uusi suutin saapui Ultimaker-tulostimeen, kun varakappaleet oli jo tulostettu ja työ luovutettu tilaajalle. Tulostimella kokeiltiin tulostaa uudelleen sama testikappale, jossa nopeutta kiihdytetään kappaleen edetessä. Uusi suutin paransi tulosta huomattavasti, ja nopein, 10 mm<sup>3</sup>/s -vauhti onnistui nyt ongelmitta. Suutin oli siis todennäköisesti uudesta asti jollain tavalla viallinen, sillä uusi suutin paransi muovin pääsyä ulos.

### 3.2 Maa-asema

Maa-aseman salkkuun piti saada kiinnitettyä kaikki tarvittavat osat eli näyttö, keskusyksikkö, näytön kaukosäädin, virtalähde, näppäimistö ja mobiilidatayhteyden modeemi. Kiinnityksessä käytettiin monenlaisia ratkaisuja, kuten 3D-tulostusta. Näyttö on yläreunastaan kiinni 3D-tulostetulla koukun muotoisella kappaleella, joka pitää näytön yläreunan kiinni. Alareunastaan näyttö on kiinni puulistalla, jonka keskellä on tappi, joka menee näytön alareunassa olevan reiän läpi. Lopullisesti näyttö kiinnittyy salkkuun salpojen avulla. Kuvassa 10 havainnollistetaan, kuinka näytön yläreuna lukittuu paikalleen. Kuvassa kappaleet ovat kulmassa toisiinsa nähden, mikä kuvaa näytön irrottamistilannetta. Koukkuun on mallinnettu pyöritykset, jotta näytön irrottaminen ja uudelleen kiinnittäminen sujuisi helpommin. Malli on melko korkea ja leveä, jotta se tukisi näyttöä laajalta alueelta kuljetuksen aikana.

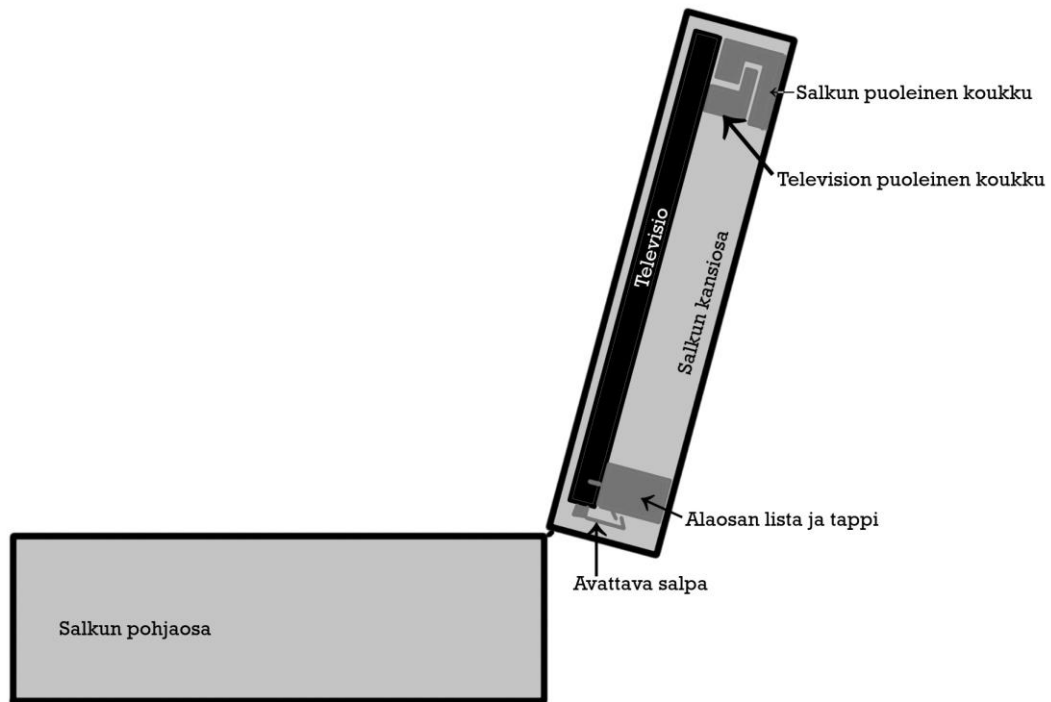


Kuva 10. Lennättäjälle lentotiedot näyttävän maa-aseman näytön yläreunan kiinnitysmekanismi. Oikealla on maa-aseman salkun ja vasemmalla näytön puoleinen kappale.

Näytön kiinnitykseen suunnitellut kappaleet kiinnitetään näyttöön ja salkkuun ruuvamalla. Salkunpuoleinen osa on leveämpi, ja sen kiinnitysreiät ovat aivan kappaleen reunoilla. Näytönkiinnike on sen verran kapeampi, että salkunpuoleisen kiinnikkeen ruuvit mahtuvat olemaan sen sivuilla. Näytönpuoleiseen kiinnikkeeseen ruuveille tehtiin syvennykset, jolloin ne jäivät samalle tasolle kiinnikkeen pinnan kanssa. Salkunpuoleiseen kiinnikkeeseen ei tehty samanlaisia upotuksia, koska se kiinnittyy salkun kannen läpi menevillä pulteilla. Salkun ulkopuoli haluttiin mahdollisimman siistiksi, jolloin ruuvien kanta jätettiin ulkopuolelle. Näin ollen pultti jäi kiinnikkeen puolelle ja pultin kiristystä varten pultin ympärille piti saada kiintoavain tai hylsy. Upotuksesta olisi pitänyt tehdä tämän vuoksi paljon isompi, kuin itse pultti on. Tätä ei haluttu tehdä, koska se heikentää kiinnikkeen rakennetta. Paremmaksi ratkaisuksi todettiin kiinnikkeen leventäminen, jolloin pultin kiristäminen oli helppoa.

Näyttö oli salkun osista vaikein ja monimutkaisin kiinnitettävä, koska se oli suurikokoinen ja painava sekä herkkä rikkoontumaan, jos se pääsisi heilumaan kuljetuksen aikana salkun sisällä. Näyttö haluttiin kuitenkin irrotettavaksi, jotta sen liitännät ovat tarvittaessa saatavilla. Testauksen aikana näytöstä rikkoontuikin yksi liitin, jolloin irrotettavuudesta oli hyötyä. Näytön kiinnitykseen mietittiin monenlaisia ratkaisuja, mutta toteutettu ratkaisu todettiin helpoimmaksi ja toimivimmaksi tehdä. Näyttö piti saada hieman irti salkun kannesta, koska liittimet osoittavat suoraan näytön taakse. Kuvanvälittämiin tarkoitettu HDMI-kaapelissa käytettiin 90 asteen kulmapalaa, mutta virtajohtoliittimestä oli saatavilla vain suoriamalleja. Suora malli olisi vaatinut noin seitsemän senttimetriä tilaa näytön taakse, joten liitintä päätettiin muokata itse siten, että virtajohto lähtee sivulle.

Näytön alaosassa oli valmiiksi halkaisijaltaan noin 9 millimetrin kokoinen reikä, eikä sitä ollut mahdollista suurentaa. Näytön alaosan kiinnitys toteutettiin poikkeuksellisesti puulistalla, koska tulostamalla olisi ollut vaikeaa tehdä riittävän kestävää tappia kannattelemaan koko näytön paino. Kun salkun kansi on auki, näyttö pysyy paikoillaan käytännössä pelkästään alaosan tapin avulla. Lisäksi puuhun oli helpompi porata ruuvit, jotka pitävät salvat paikoillaan. Kun salkku on kiinni ja sen pohjan on maata vasten eli kun maa-aseman käyttö lopetetaan eikä salkkua nosteta pystyyn, pitää näytön yläosastaan paikallaan aiemmin mainittu J-kirjaimen muotoinen kappale ja alaosastaan näyttö pysyy kiinni avattavien salpojen avulla. Salvat puristavat näytön tiukasti puulistaa vasten ja estävät näytön irtoamisen tapista, joka osoittaa salkun ollessa tässä asennossa suoraan alaspäin. Kuva 11 havainnollistaa kiinnitysratkaisua.



Kuva 11. Salkun poikkileikkaus. Osien mittasuhteita on muutettu, jotta yksityiskohdat näkyisivät paremmin.

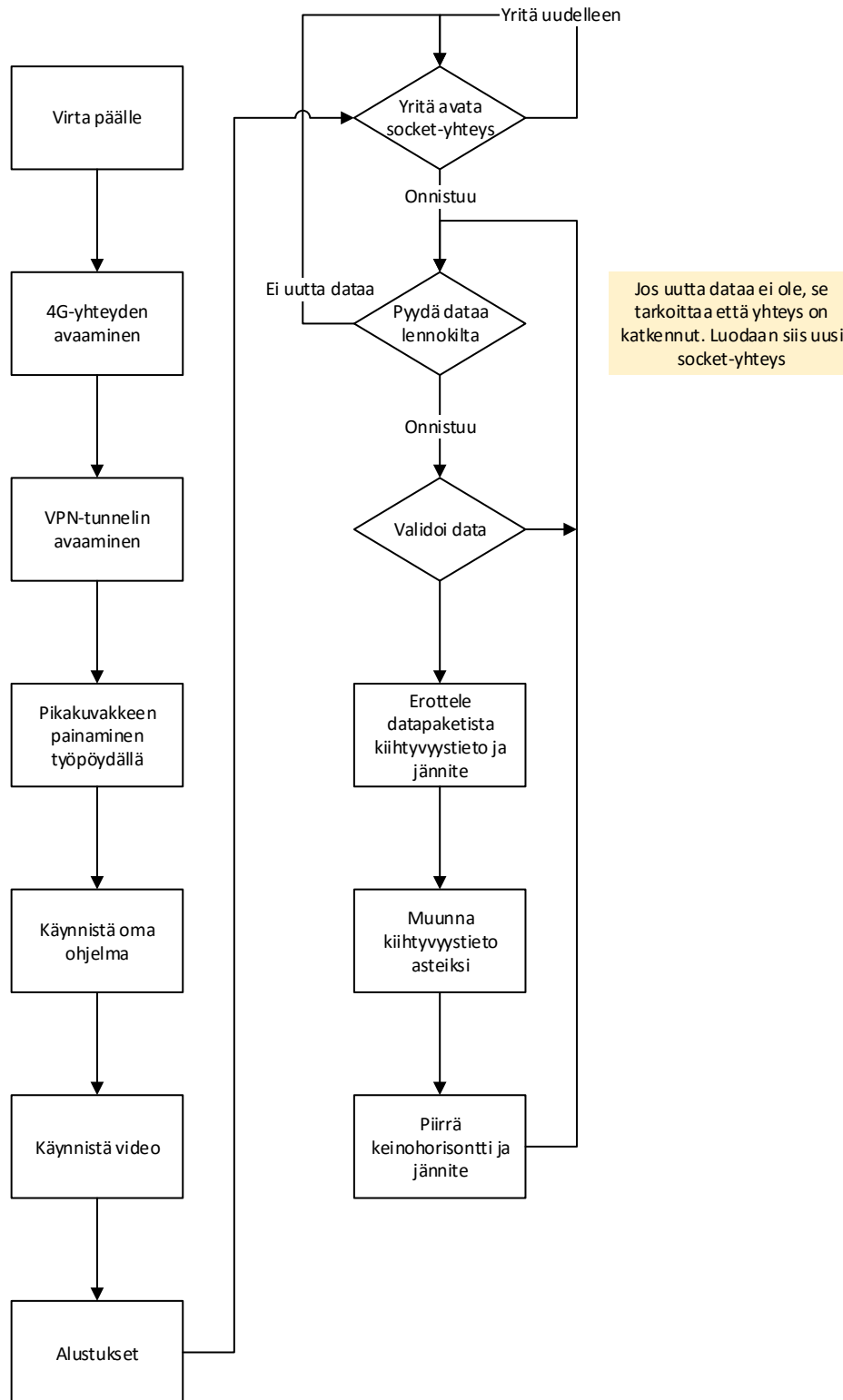
Keskusyksikkö on omassa kotelossaan, mutta itse kotelo on kiinni vain salkun vaahtomuovin avulla. Myös muut osat eli 4G-modeemi, virtalähde, näppäimistö ja näytön kaukosäädin ovat kiinni vain vaahtomuovin avulla. Nämä kaikki osat ovat sen verran kevyitä, että ne pysyvät riittävän hyvin paikoillaan pelkästään sopivan kokoiseksi leikattujen kolojen sisällä. Kaikki johdot kulkevat vaahtomuovin sisällä, mutta virtajohdolle ja 4G-modeemille jätettiin hieman vapaata johtoa. Tämä ylimääräinen johto on kääritty siistille rullalle omaan koloonsa, josta sen voi purkaa tarpeen vaatiessa.

Näytön kaukosäädintä tarvitaan harvoin, koska näyttö on asetettu käynnistymään automaattisesti muun järjestelmän kanssa. Kaukosäädin piti kuitenkin saada mukaan mahdollisia asetusten tekoa varten, esimerkiksi jos tulosignaalia halutaan vaihtaa tai näytön kirkkautta muuttaa.

Testausvaiheessa huomattiin, että käyttäjän täytyy pystyä avaamaan ja sulkemaan ohjelma ongelmatilanteissa. Näin ollen maa-aseamalla avataan automaattisesti vain 4G-datayhteys ja VPN-putki. Videokuva ja mittarit näytetään vasta käyttäjän painaessa työpöydällä olevaa pikakuvaketta. Videokuva siirretään erillisenä yhteytenään VPN-

putken sisällä ja kaikki telemetriatieto omana datavirtanaan. Putkessa liikkuu siis kaksi toisistaan riippumatonta datavirtaa, joista videokuva vie suurimman osan käytettävissä olevasta tiedonsiirtokapasiteetista. Jommankumman datavirran katketessa käynnistetään kyseisen datan lähetys automaattisesti uudelleen. Videokuvassa ei käytetä virheidentarkistusta, joten kuvassa saattaa esiintyä virheitä ajoittain. Telemetrille käytetään yksinkertaista virheentunnistusta, jolla varmistetaan paketin tietojen oikeellisuus. Jos tietoja puuttuu, paketti hylätään ja odotetaan seuraavaa pakettia ennen näytön päivitystä. Kuvio 1 havainnollistaa maa-aseman toimintaa.





Kuvio 1. Maa-aseman toimintaa kuvaava vuokaavio.

### 3.3 Lennokki

Lennokin kotelot haluttiin mahdollisimman vesitiiviiksi, mutta kuitenkin avattaviksi. Helppoin tapa olisi ollut liimata kotelot kiinni, jolloin vesitiiviys olisi ollut taattu, mutta tällöin minkä tahansa vian ilmaantuessa koteloa ei olisi voitu avata rikkomatta sitä. Sen vuoksi koteloihin suunniteltiin jonkinlaista tiivistettä, joka pitäisi kotelon riittävän tiiviinä vesiroiskeita vastaan. Imakr.com-verkkokauppa myy pehmeää, kumimaista materiaalia (38), jonka arveltiin soveltuvan hyvin tiivisteeksi, joten sitä tilattiin kokeeksi.

Materiaali osoittautui pehmeähköksi, mutta tiivisteenä se ei kuitenkaan ollut riittävän pehmeää. Materiaalin pitäisi antaa periksi melko paljon, eikä tulostettava materiaali pysty siihen. Vaihtoehtoiksi jäi erillisen kumisen tiivisteiden liimaaminen saumakohtaan, sauman liimaaminen kiinni tai jonkinlaisen uran teko, joka vastustaa veden pääsyä koteloon. Erillisestä tiivisteestä kuitenkin luovuttiin, koska se olisi tehnyt rakenteesta hieman huteran. Näin päädyttiin suunnittelemaan ura, joka estää veden suoran pääsyn koteloon. Jos lennokki tekee pakkolaskun järveen, tällainen ura ei pysty pitämään vettä poissa kotelosta, mutta se suojaa sadevedeltä.

Kuten kuvasta 12 näkyy, keskusyksikkö on asennettu lennokin rungon sisään, jonne saattaa päästä hieman vettä, mutta kotelo ei kuitenkaan ole suorassa vesisateessa. Näin ollen pelkkä ura arvioitiin riittäväksi eristykseksi. Jos tiiviyttä halutaan parantaa myöhemmin, on kotelo mahdollista tiivistää saumoistaan silikonilla. Tämä kuitenkin estää kotelon avaamisen, joten esimerkiksi kameran vaihtaminen sisällä olevaan tietokoneeseen ei ole mahdollista. Johtojen läpivientien aukot täytettiin silikonilla, jolloin kotelossa ei ollut avoimia reikiä lainkaan, mutta avaaminen on kuitenkin vielä mahdollista.

Kameran kotelon koko etupuoli on lasia, jotta kameran objektiivilla olisi esteetön näkymä ulos kotelosta. Kotelo on lennokin ulkopuolella, joten sen täytyy kestää pölyä ja vettä. Näin ollen kotelon kiinnitysreiät tiivistettiin silikonilla, samoin kuin kameran johdon läpivienti. Myös kameran kotelon lasi on kiinni reunoistaan silikonilla. Näin koko kotelosta saatiin täysin tiivis ja sen pitäisi kestää jopa upotus veteen. Haittapuolena on, että melko helposti rikkoutuvaa kameran lattakaapelia ei voi vaihtaa rikkomatta koteloa ja todennäköisesti samalla myös kameraa. Kamera ja mobiilidatayhteyden modeemi näkyvät kuvassa 12, jossa lennokin oikealla puolella on modeemin kotelo ja vasemmal-

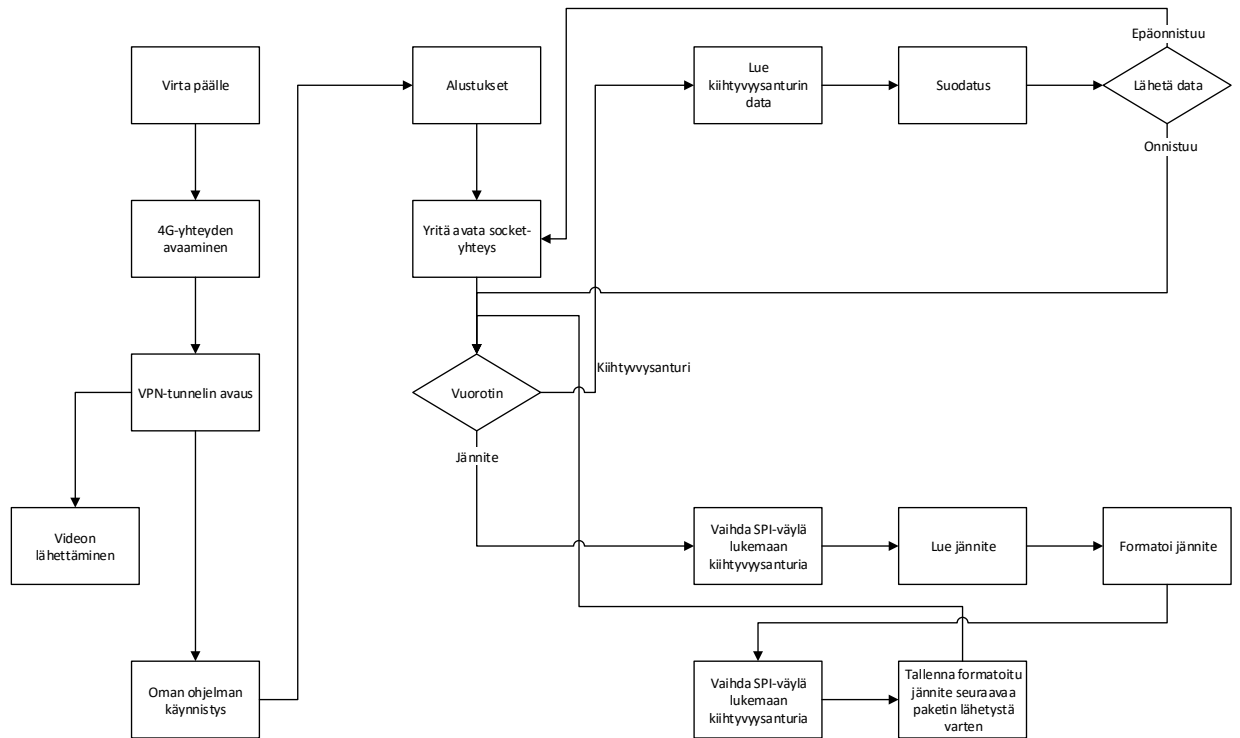
la kamera. Kuvassa kamera on vielä alkuperäisessä sijoituspaikassaan alumiinikiskon päässä, josta se siirrettiin testauksessa havaitun värinäongelman takia siiven alle.



Kuva 12. Lennokki ja tiedonkeruujärjestelmä.

Lennokin koodi suunniteltiin käynnistymään automaattisesti, jotta pelkkä sähköjen kytkeminen riittäisi järjestelmän käynnistämiseksi. Aluksi ohjelma käynnistää datayhteyden käynnistämällä 4G-modeemin ja suorittamalla sille oikeat alustukset. Tämän jälkeen käynnistetään kamera ja sen kuvaa ruvetaan siirtämään 4G-yhteyden kautta maa-asemalle. Viimeiseksi käynnistetään kiihtyvyyss- ja jännitteenmittausanturit, joiden tiedot välitetään maa-asemalle. Videokuva ja telemetriatiedot ovat VPN-putkessa omi-na datavirtoinaan, jolloin toisen datavirran katkeaminen ei häiritse toista.

Ohjelmointi aloitettiin tekemällä koodi, joka hakee tiedon kiihtyvyyssanturilta SPI-väylän avulla ja antaa sen eteenpäin merkkijonona. Tämä merkkijono tallennetaan viiden alkion suuruiseen taulukkoon, minkä mittausarvoille voidaan tehdä suodatusta. Mittausarvoille lasketaan liukuva keskiarvo, minkä jälkeen tulokset ajetaan vielä alipäästösuotimen läpi (kuvio 2). Suodatuksen jälkeen arvo lähetetään maa-asemalle.



Kuvio 2. Maa-aseman toiminnan kuvaava vuokaavio.

OpenVG on erittäin matalan tason kirjasto, joka piirtää käyttöjärjestelmän ohi suoraan grafiikkapiirille. Tämä mahdollistaa hyvän suorituskyvyn, mutta vaikeuttaa samalla tekstin esittämistä yhtä aikaa muun tiedon kanssa. Aluksi jännite ja mahdolliset virheviestit aiottiin esittää terminaalissa, mutta koska terminaali jää ohjelman alle, täytyi kaikki tekstit lisätä omaan ohjelmaan osaksi piirtoa. Lopulta tämä oli kuitenkin selkeämpi ratkaisu, koska fontti saatiin isommaksi ja värejä pystyttiin hyödyntämään.

Keinohorisonin koodaus aloitettiin piirtämällä mahdollisimman yksinkertainen toteutus aiheesta, eli puolet näytöstä piirrettiin vihreällä, mikä esitti maata ja puolet näytöstä sinisellä. Tämän taustan päälle piirrettiin lentokone, jota alkuvaiheessa esitti merkkijono "-v-", jossa väliviivat esittivät lentokoneen siipiä ja kirjain "v" runkoa. Tätä lentokonetta sitten yritettiin pyörittää ja liikuttaa taustan päällä.

OpenVG-kirjaston origo on näytön vasen alakulma, joten kaikki käytettävät koordinaatit ovat positiivisia. Aluksi lentokoneen pyöritys tuntui mahdottomalta, koska kuva pyöri origon ympäri, jolloin kuvio liikkui laajassa kaaressa pitkin näyttöä kallistuen samalla. Kirjastossa olevalla funktiolla origoa voi siirtää haluamaansa paikkaan näytöllä ja saada se asetettua keskelle näyttöä. Muutoksen jälkeen lentokone pysyi pyöritettäessä

paikallaan eikä enää tehnyt laajaa kaartaa samalla. Seuraavaksi lentokonetta siirrettiin vertikaalisesti, mikä esitti lentokoneen nokan kulmaa suhteessa horisonttiin.

Lennoikin piirtotestin onnistuttua alettiin miettiä, miten näytölle saisi videon ja jännitetiedon samaan aikaan horisontin kanssa. Tässä päädyttiin skaalaamaan horisontti pienemmäksi näytön oikeaan alakulmaan, vasempaan alareunaan jännitetieto ja vasempaan yläkulmaan video. Videolle varattiin suurin tila, jotta siitä saisi mahdollisimman hyvin selvää. Horisontti oli suunniteltava uudelleen, koska tässä piti ottaa huomioon näytön resoluutio ja fyysinen koko, jotta horisontista saisi selvää.

Keinohorisonttia käytiin kehitysvaiheessa näyttämässä tilaajalle Santahaminassa, jolloin havaittiin, ettei h.264-pohjainen video vastaa viiveiltään toiveita. Viive oli noin sekunnin luokkaa, mutta välillä se saattoi kasvaa muutamaan sekuntiin puskuroinnin takia. Keinohorisontin ulkonäkö ei ollut lopullinen, ja siihen tulikin parannusehdotuksia, kuten asteiden lisääminen näkymään. Pyyntöjen pohjalta tehtiin muutoksia keinohorisontin ulkonäköön ja tutkittiin, olisiko videon viivettä mahdollista lyhentää entisestään.

### 3.4 Raspberry Pi 2 -minitietokoneen päivityksen ongelmat

Työssä käytetystä Raspberry Pi -minitietokoneesta julkaistiin tehokkaampi versio ohjelmistokehityksen aikana. Uuden mallin luvattiin olevan huomattavasti, eli keskimäärin noin kuusi kertaa, aiempaa nopeampi. Uudempaan malliin haluttiin päivittää, koska aluksi käytetty Raspberry Pi model B oli todettu melko hitaaksi sekä videon siirrossa että keinohorisontin piirtämisessä. Uudemman Raspberry Pi 2 -tietokoneen piti olla sekä laitteistoltaan että ohjelmistoiltaan taaksepäin yhteensopiva, joten päivityspäätös oli helppo. Ikävä kyllä päivitys ei kuitenkaan ollut ongelmaton.

Käyttöjärjestelmän ytimelle täytyy kertoa jollakin tavalla, minkälaisia laitteita järjestelmään on kytketty ja mitkä moduulit ytimen täytyy ladata käynnistyksen aikana. USB-laitteet tunnistuvat automaattisesti, mutta esimerkiksi prosessorin tyyppi täytyy määrittellä käsin. Tässä määrittelyssä käytettiin aikaisemmissa Raspbian-käyttöjärjestelmän jakeluversioissa struct-tietotyyppimuuttujia, mutta uudemmissa koontiversioissa käyttöjärjestelmässä otettiin Device Tree oletuksena käyttöön. (39.)

Struct-muuttujat ovat c-ohjelmointikielen säiliöitä, joihin voi lisätä monenlaisia toisiinsa liittyviä muuttujia. Tavallisen käyttäjän auttamiseksi käyttöjärjestelmä on aikaisemmin ladannut oletuksena lukuisia tarpeettomia moduuleita, mikä vie turhaan resursseja ja hidastaa järjestelmää. Device Tree on hieman XML-merkintäkielen kaltainen tiedosto, joka ratkaisee tämän ongelman siten, että käyttäjän täytyy lisätä haluamansa moduulit siihen, jolloin ne ladataan käyttöön. Device Treehen on vaivatonta tehdä muutoksia muokkaamalla vain yhtä tiedostoa, joten uusien laitteiden lisääminen ei ole ongelma. Struct-pohjaisessa toteutuksessa käyttöjärjestelmän ydin täytyy kääntää muutosten jälkeen uudelleen, mikä kestää etenkin pienitehoisella minitietokoneella kauan. Struct-pohjainen toteutus on hankala ylläpidettävä, ja pidemmällä aikavälillä määritykset voivat olla erittäin monimutkaisia. Device Tree pysyy loogisena paremmin, koska se on hierarkkinen järjestelmä. (40.)

Käyttöjärjestelmä oli pakko päivittää samalla, kun käyttöön otettiin uusi versio minitietokoneesta, koska vain uusien versio Raspbian-käyttöjärjestelmästä sisälsi tuen Raspberry Pi 2 -minitietokoneelle. Ikävä kyllä, aiemmin mainittu Device Tree aiheutti SPI-väylän toimimattomuuden, ja täten kiihtyvyyssanturiin ei saatu enää yhteyttä. Ongelma saatiin ratkaistuksi lisäämällä tarvittavat moduulit Device Treehen käyttäen Raspberry Pin asetusohjelmaa.

Toinen päivityksen aiheuttama ongelma oli mobiilidatayhteyden toimimattomuus. Tämä johtui siitä, että modeemi otetaan käyttöön automaattisesti skriptillä järjestelmän käynnistyessä. Tämän skriptin ajoitukset oli suunniteltu toimimaan vanhemman ja hitaamman prosessorin mukaan, jolloin nopeampi prosessori kävi skriptin läpi liian nopeasti. Modeemi ei siis ollut vielä valmis suorittamaan pyydettyjä komentoja, kun tietokone alkoi suorittaa niitä. Ratkaisu oli yksinkertaisesti suunnitella skriptiin uudet ajoitukset, jolloin modeemi pysyi perässä. Korjauksen jälkeen modeemi lähti toimimaan jälleen automaattisesti, kun järjestelmä oli uudelleen käynnistetty.

#### **4 Havaitut ongelmat ja niiden korjaaminen**

Kehityksen aikana testattiin vain yhtä osa-aluetta kerrallaan, koska se oli ainoa tapa löytää toimivin ratkaisu lukuisista eri vaihtoehdoista. Suurin osa testauksesta painottui loppupuolelle, kun koko järjestelmän oli valmis. Järjestelmää testattiin lennokissa vain

kerran ennen lopullista lennätystä. Tämä johtui siitä, ettei lennokkia voinut käynnistää ilman oikeaa henkilökuntaa ja varusteita, joten lennokkia testattiin Tikkakoskella.

Testauksen aikana Tikkakoskella havaittiin muutosten tarve sekä mekaanisella että ohjelmallisella tasolla. Tässä testauksessa ilmeni suunnittelusta riippumaton ongelma. Maa-aseman näytön kuva vilkkui testauksen aikana oudosti ja välillä katosi kokonaan. Selvityksen jälkeen ongelma paikallistettiin näytön vialliseen HDMI-porttiin. Näytössä on kaksi HDMI-liitintä, joten ongelma ratkaistiin yksinkertaisesti vaihtamalla HDMI-johto toiseen porttiin. Vian selvityksessä auttoi näytön helppo irrotettavuus salkusta.

#### 4.1 Ohjelmiston muutokset

Ensimmäinen laajempi testaus tehtiin, kun järjestelmän kaikki osat oli saatu kommunikoimaan keskenään ja mobiilidatayhteys toimimaan. Tässä vaiheessa myös huomattiin, että itse tehdyn keinohorisontin ja jännitteen piirtävän ohjelman toteutuksessa oli ohjelmointivirhe, joka aiheutti ohjelman kaatumisen datapaketin hävitessä. Tämä ei tullut LAN-yhteydellä ilmi, koska LAN-verkossa paketteja ei hävinnyt lainkaan.

Odottamattoman ongelman sai korjattua muokkaamalla tietoliikenneyhteyttä molemmista päistä siten, että ohjelma ei sammu paketin hävitessä, vaan se yrittää muodostaa yhteyden automaattisesti uudelleen. Alun perin ajatuksena oli, että lennokin päässä suoritettava ohjelma sulkeutuisi yhteyden katketessa maa-asemalle. Yhteyden katkaisemiseen olisi riittänyt maa-asemalla suoritettavan ohjelman sulkeminen, eli maa-aseman ohjelman sulkeminen olisi sammuttanut myös lennokissa suoritettavan ohjelman. Muutoksen jälkeen ohjelmaa ei saa suljettua lennokin päästä muuten kuin Linux-käyttöjärjestelmän *kill*-komentoa käyttäen. Sillä saa sammutettua minkä tahansa suorituksessa olevan ohjelman. Komennon antaminen vaatii SSH-terminaaliyhteyden (Secure Shell) lennokin Raspberry Pihin, joten ohjelman sulkeminen ja uudelleen käynnistäminen eivät ole yksinkertaisia operaatioita. Käytännössä helpointa on käynnistää koko järjestelmä uudelleen suoraan virtakytkimestä. SSH tarjoaa turvallisen terminaaliyhteyden kahden laitteen välillä (41).

Myös maa-asema suunniteltiin aluksi käynnistymään suoraan omaan ohjelmaan siten, että video ja keinohorisontti käynnistyvät heti, kun järjestelmä on käynnistynyt. Testattaessa järjestelmää ongelmaksi havaittiin, että jos ohjelma halutaan sammuttaa syystä

tai toisesta, se ei ole mahdollista ilman SSH-yhteyttä. Täten mukaan pakattu näppäimistö olisi ollut turha. Yleensä terminaalissa avatut ohjelmat voidaan sulkea näppäinoikopollulla, joka on monesti Control-C. Tämä ei kuitenkaan toiminut tässä tapauksessa, koska käyttäjä ei ollut käynnistänyt ohjelmaa, vaan se käynnistetään käyttöjärjestelmän *rc.local*-tiedoston avulla. Tähän tiedostoon voi kirjoittaa komentoja, jotka suoritetaan aina järjestelmän käynnistyessä. Tämän takia käyttäjän antama lopetuskomento ei mene perille, koska *rc.local*-tiedoston avulla käynnistetyt ohjelmat suorittaa eri käyttäjä kuin joka käyttää näppäimistöä.

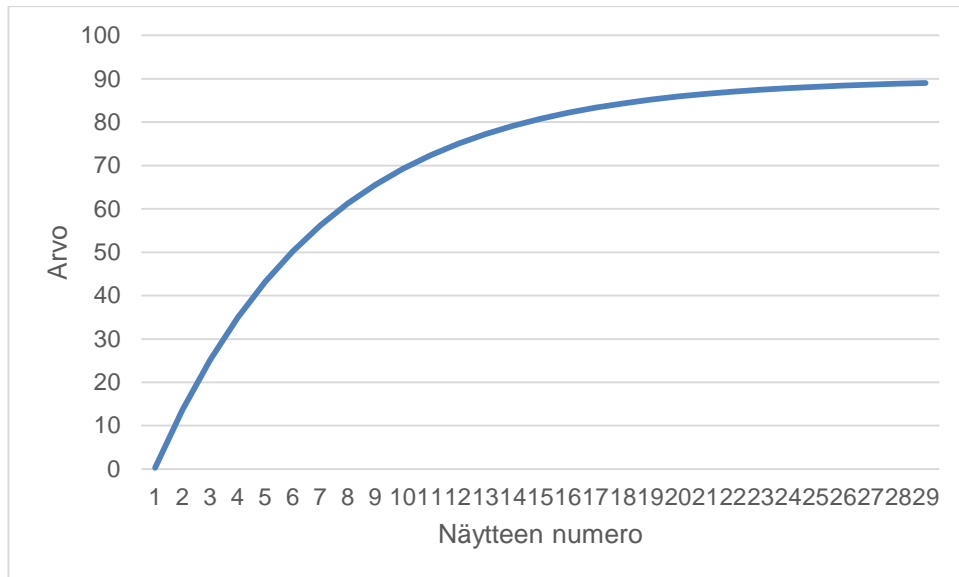
Ohjelman sulkeminen haluttiin mahdollistaa, jotta käyttäjä voi suorittaa muita komentojaan, kuten ottaa SSH-yhteyden lennokkiin. Myös maa-aseman asetuksia olisi hyvä voida muuttaa suoraan näppäimistöllä, ilman erillistä SSH-yhteyttä. Näin ollen Raspberry Pin asetuksia muutettiin siten, että tietokone käynnistyy työpöydälle eikä omaan ohjelmaan. Työpöydälle luotiin pikakuvake, jota painamalla oman ohjelman voi käynnistää. Pikakuvake avaa sekä keinohorisontin ja jännitteen piirtävän ohjelman että videoyhteyden. Koska ohjelma on täten käyttäjän avaama, sen pystyy sulkemaan Control-C-näppäinyhdistelmällä.

#### 4.2 Lennokin tärinäongelma

Muutosten jälkeen tiedonkeruujärjestelmää käytiin testaamassa Tikkakoskella. Järjestelmä asennettiin lennokkiin, ja sen polttomoottori käynnistettiin. Tässä vaiheessa lennokilla ei vielä lennetty, mutta jo pelkkä moottorin käynnistäminen paljasti erittäin pahan ongelman: sekä videokuva että keinohorisontti tärisivät niin pahasti, ettei kummasakaan saanut mitään selvää.

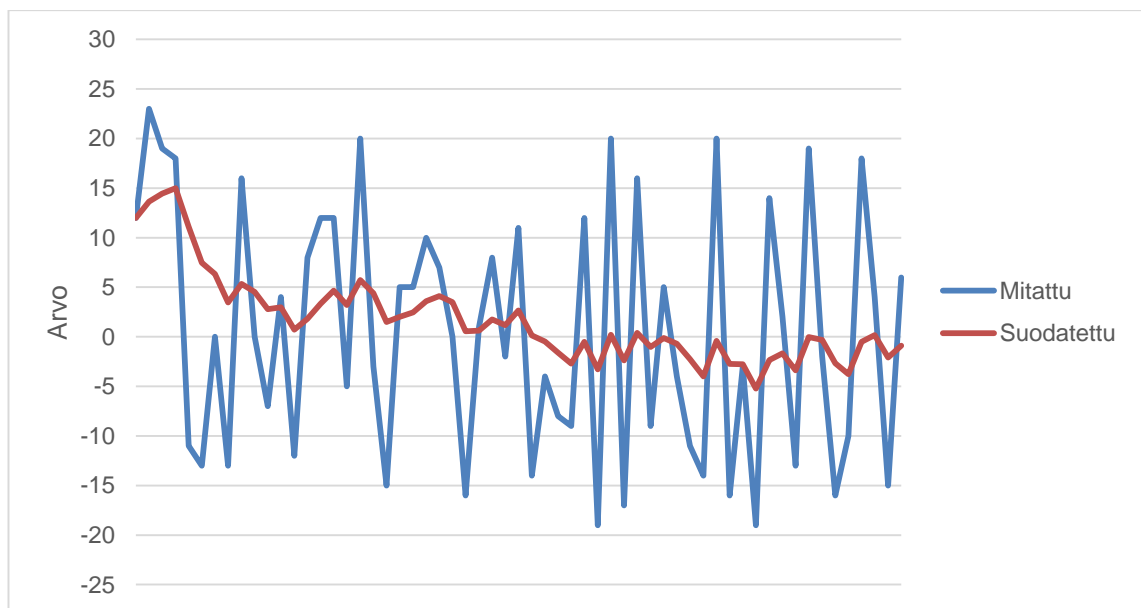
Kiihtyvyyssanturi, jolta keinohorisontti saa asentotietonsa, päätettiin suodattaa ohjelmallisesti, koska tärinän estäminen mekaanisesti olisi ollut hankala. Kuvassa 13 esitetään alipäästösuodattimen askelvaste eli se, kuinka suodatettu arvo käyttäytyy antoarvon muuttuessa nolasta yhdeksäänkymmeneen. Kuvassa antoarvo muuttuu ajanhetkellä nolla ja sininen käyrä kuvaa, kuinka suodatettu arvo muuttuu. Käytännössä suodatus siis hidastaa arvon muuttumista nopeasti arvosta toiseen, mikä on haluttu ominaisuus, jotta lentokoneen moottorin aiheuttama tärinä ei näkyisi virtuaaliorisontissa.





Kuva 13. Suodatuksen askelvaste.

Kuvassa 14 esitetään, miten suodattamaton ja suodatettu arvo käyttäytyvät simuloitussa tilanteessa, jossa lentokone on suorassa eli arvojen pitäisi olla nollassa. Moottorin aiheuttama värinä kuitenkin aiheuttaa arvojen poukkoilua nollan molemmin puolin. Suodatettu arvo tasaantuu noin nollan kohdalle, mutta suodattamaton hyppii laajalla alueella. Kuvassa on otettu vain kuusikymmentä arvoa, joten pidemmällä aikavälillä suodatetut arvot tasaantuisivat entistä lähemmäs nollaa, mutta mitatut arvot jatkaisivat vaihtelua nollan molemmin puolin.



Kuva 14. Mitattu ja suodatettu arvo.

Kameran tärinää päätettiin vähentää sen sijaan mekaanisesti, koska videon tärinän vakautus ohjelmallisesti olisi vaatinut huomattavasti enemmän laskentatehoa kuin mihiin Raspberry Pi pystyy. Kamera oli aluksi asennettu alumiinikiskon toiseen päähän, minkä epäiltiin tärinän ilmettyä olevan huono sijoituspaikka. Kamera siirrettiin lähemmäs runkoa siten, että se on edelleen kiskossa kiinni, mutta suoraan kiinnityslevyn kohdalla. Yllättäen tärinä kuitenkin vain paheni tällä ratkaisulla. Alumiinikisko siis sitoi matkalla osan tärinästä.

Lopulta kisko poistettiin kokonaan ja kamera kiinnitettiin suoraan hiilikuitulevyyn. Tämäkään ei poistanut tärinää, mutta siiven ja kiinnityslevyn väliin asennettiin vielä palanen vaahtomuovia, joka painoi levyä kovemmin siipeä vasten. Tämä vähensi tärinää huomattavasti, ja kuvasta alkoi saada selvää. Lisäksi koneen tärinän pitäisi vaimentua entisestään lennokin ollessa ilmassa, joten asennukseen oltiin tyytyväisiä. Kuvassa 15 näkyy kameran lopullinen sijoituspaikka ja tärinän vaimennukseen käytetty vaahtomuovin palanen.



Kuva 15. Kameran lopullinen sijoituspaikka siiven alla.

Kameran värinäongelma pystyttiin korjaamaan paikanpäällä, mutta kiihtyvyyssanturin arvot suodatettiin vasta testauksen jälkeen. Suodatusta ei näin ollen voinut kokeilla lennokissa, vaan lopullinen testaus jäi varsinaiseen lennätukseen Lohtajalla. Värinän vaikutusta kiihtyvyyssanturiin simuloitiin ravistelemalla laitetta käsissä ja katsomalla keinohorisontista, millaisella tärityksellä se näyttää samalta kuin lennokkiin asennettuna. Ravistelemalla laitetta ensin jonkin aikaa suodattamattomalla koodilla ja sitten suodatetulla saatiin melko hyvä kuva suodatuksen toimivuudesta.

#### 4.3 Lennätys Lohtajalla

Varsinainen lennätys ja työn lopullinen testaus tehtiin Lohtajalla. Koska sää oli aurinkoinen, koteloiden vesitiivyyttä ei testattu. Kaikki järjestelmät toimivat odotusten mukaisesti ja tilaaja oli tyytyväinen erityisesti videokuvan laatuun ja viiveettömyyteen. Lenno-kin järjestelmä kesti testauspäivänä suoritettua kolme nousua ja laskeutumista hyvin, eikä mitään vaurioita havaittu. Yksi laskuista oli erityisen raju lennokin tullessa lasku- varjolla alas. Kuva ei kuitenkaan katkennut hetkeksikään tässä testissä. Kuten kuvasta 16 näkyy, on kameran nykyinen sijoituspaikka muuten esteetön, mutta lenno-kin potkuri näkyy kuvassa. Kuvassa moottoria ei kuitenkaan ole vielä käynnistetty, ja potkuri häviää lähes täysin lennon aikana, kun se pyörii kovilla kierroksilla.



Kuva 16. Valmis maa-asema.

Kuvassa 16 näkyy, että suuri osa näytön pinta-alasta on varattu videokuvalle, jotta siitä saisi mahdollisimman hyvin selvää. Video kuvataan ja esitetään 1280 x 720 kuvapisteen tarkkuudella, jolloin sitä ei tarvitse skaalata missään vaiheessa. Tämä on tarpeen, jotta laatu ei kärsisi tarpeettomasti. Koska käytetty näyttö on erottelukyvyltään 1600 x 900 kuvapistettä, jää muille esitettävillä tiedoille riittävästi tilaa, vaikka video varaakin suuren osan näyttöpinta-alasta.

## 5 Yhteenveto

Insinööriyössä tehtiin langattoman tiedonkeruujärjestelmän suunnittelu, toteutus ja testaus. Tiedonkeruujärjestelmä sisältää lennokkiin asennettavan kameran ja asentaja jänniteanturit. Lennokin keräämät tiedot esitetään lennättäjälle reaaliajassa. Alun suunnitteluvaiheessa mietittiin erilaisia ratkaisuja järjestelmän toteuttamiseksi, ja siinä

huomioitiin toteutuksen helppous, hinta ja soveltuvuus käyttötarkoitukseensa. Erilaisia ratkaisuja testattiin jo suunnitteluvaiheessa, jotta huonot vaihtoehdot saatiin karsituiksi pois. Seuraavaksi suunnitteluvaiheessa tehdyt ratkaisut toteutettiin käytännössä. Toteutuksen aikana ilmeni monia ongelmia, mutta ne onnistuttiin ratkaisemaan. Lopuksi järjestelmää testattiin, ja silloin havaittiin esimerkiksi lennokin värinäongelma, joka ratkaistiin osittain mekaanisia ja osittain ohjelmallisia keinoja käyttäen.

3D-tulostus tarjosi joustavan tavan suunnitella ja toteuttaa komponenttien kiinnitys lennokkiin ja maa-asemaan. 3D-suunnitteluun käytetyssä ohjelmassa huomattiin muutamia puutteita, jotka hidastivat osien suunnittelua. 3D-tulostimessa oli ongelmana muun muassa tukkeutunut suutin, jonka takia osa tulosteista epäonnistui. Tulevaisuudessa laitteiden ja ohjelmistojen kehittyessä 3D-tulostus tarjoaa mielenkiintoisia mahdollisuuksia erilaisten pienten kappaleiden valmistukseen. Esimerkiksi matkapuhelimelle voi tulostaa suojakuoren ja kylpyhuoneeseen uuden pyyhekoukun rikkoontuneen tilalle. 3D-tulostimella voi myös valmistaa persoonallisia lahjoja tuttavilleen.

Tietokoneet ovat kehittyneet nopeasti, ja nykyään pienimmät tietokoneet ovat pinta-alaltaan vain luottokortin kokoisia ja vähävirtaisia. Nämä minitietokoneet ovat halpoja, mikä avaa uusia mahdollisuuksia suunnitella erilaisia käyttösovelluksia laitteille. Tässä työssä käytetty Raspberry Pi -minitietokone on erittäin suosittu harrastelijakäytössä, ja sen avulla on toteutettu erilaisia laitteita, kuten seinäkelloja, kotipalvelimia ja mediasoitimia. Se tarjosi myös hyvän alustan sekä tiedon keräämiseen että esittämiseen.

Järjestelmää käytettiin yhden päivän ajan lennätysharjoituksessa, ja tänä aikana järjestelmän toiminnasta saatiin riittävän hyvä kuva. Järjestelmä toimi koko lennätyksen ajan hyvin, ja tilaaja oli tyytyväinen erityisesti videokuvan laatuun ja viiveettömyyteen. Järjestelmä kesti laskeutumiset vahingoittumattomana, joten kiinnitys todettiin riittäväksi. Järjestelmää voidaan hyödyntää jatkossa esimerkiksi maastouttamisen onnistumisen tarkistukseen.

## Lähteet

1. CAD Chronology: The 70's. Verkkodokumentti. M.Bozdoc  
<<http://mbinfo.mbdesign.net/CAD1970.htm>>. Luettu 20.7.2015.
2. Computer1984. Verkkodokumentti. Archive.org.  
<[https://ia802308.us.archive.org/27/items/Computer1984\\_6/Computer1984\\_6\\_512kb.mp4](https://ia802308.us.archive.org/27/items/Computer1984_6/Computer1984_6_512kb.mp4)>. Luettu 28.7.2015.
3. Buy Maya 2016. Verkkodokumentti. Autodesk.  
<<http://www.autodesk.com/products/maya/buy/>>. Luettu 26.7.2015.
4. RenderMan. Verkkodokumentti. Pixar. <<http://renderman.pixar.com/view/p-renderman/>>. Luettu 26.7.2015.
5. About Blender. Verkkodokumentti. Blender Foundation.  
<<http://www.blender.org/about/>>. Luettu 25.7.2015.
6. Blender features. Verkkodokumentti. Blender Foundation.  
<<http://www.blender.org/features/>>. Luettu 25.7.2015.
7. Blender demo reels. Verkkodokumentti. Blender Foundation.  
<<http://www.blender.org/features/reels/>>. Luettu 25.7.2015.
8. Blender open movies. Verkkodokumentti. Blender Foundation.  
<<https://www.blender.org/features/projects/>>. Luettu 25.7.2015.
9. 3D Resources NASA. 2012. Verkkodokumentti. NASA.  
<<http://nasa3d.arc.nasa.gov/detail/jpl-vtad-aqua/>>. Päivitetty 1.5.2012. Luettu 25.7.2015.
10. Blender Testimonials. 2007. Verkkodokumentti. Blender Foundation.  
<<http://web.archive.org/web/20070221025521/https://www.blender.org/features-gallery/testimonials/>>. Päivitetty 21.7.2007. Luettu 25.7.2015.
11. AutoCAD features. Verkkodokumentti. Autodesk.  
<<http://www.autodesk.com/products/autocad/features/all/gallery-view/>>. Luettu 26.7.2015.
12. Autodesk - AutoCAD - Archived Customer Stories. Verkkodokumentti. Autodesk.  
<<http://usa.autodesk.com/autocad/customers/>>. Luettu 26.7.2015.
13. 2D CAD | AutoCAD LT Features | Autodesk. Verkkodokumentti. Autodesk.  
<<http://www.autodesk.com/products/autocad-lt/features/all/gallery-view>>. Luettu 26.7.2015.

14. About FreeCAD – FreeCAD Documentation. 2014. Verkkodokumentti. FreeCAD Community. <[http://www.freecadweb.org/wiki/index.php?title=About\\_FreeCAD](http://www.freecadweb.org/wiki/index.php?title=About_FreeCAD)>. Päivitetty 27.9.2014. Luettu 16.8.2015.
15. Mini PC: Intel® NUC. Verkkodokumentti. Intel. <<http://www.intel.com/content/www/us/en/nuc/overview.html>>. Luettu 27.7.2015.
16. Intel NUC. Verkkodokumentti. Verkkokauppa.com. <<http://www.verkkokauppa.com/fi/search?q=intel+nuc&s=price&o=A>>. Luettu 27.7.2015.
17. Rajamäki, Ari. 2015. Maalilennokin langaton tiedonkeruujärjestelmä. Insinööriyö. Metropolia Ammattikorkeakoulu.
18. GIMP - The GNU Image Manipulation Program. Verkkodokumentti. GIMP. <[gimp.org](http://gimp.org)>. Luettu 19.8.2015.
19. Scalable Vector Graphics (SVG). 2004. Verkkodokumentti. W3 Schools. <<http://www.w3.org/Graphics/SVG/About.html>>. Päivitetty 29.10.2004. Luettu 22.9.2015.
20. How Apple's new vector-based Maps leave Google Maps looking jittery. 2012. Verkkodokumentti. Apple Insider. <[http://appleinsider.com/articles/12/08/03/inside\\_apples\\_new\\_vector\\_based\\_maps\\_in\\_ios\\_6.html](http://appleinsider.com/articles/12/08/03/inside_apples_new_vector_based_maps_in_ios_6.html)>. Päivitetty 3.8.2012. Luettu 19.8.2015.
21. Qt5 and the Raspberry Pi. 2015. Verkkodokumentti. Raspberry Pi Foundation. <<https://www.raspberrypi.org/blog/qt5-and-the-raspberry-pi/>>. Päivitetty 25.9.2015. Luettu 24.8.2015.
22. sebmattton/jquery-flight-indicators · GitHub. 2014. Verkkodokumentti. Sebmattton. <<https://github.com/sebmattton/jquery-flight-indicators>>. Päivitetty 26.9.2014. Luettu 18.8.2015.
23. Category:QtonPi. 2015. Verkkodokumentti. QT. <<https://wiki.qt.io/Category:QtonPi>>. Päivitetty 20.5.2015. Luettu 19.8.2015.
24. Setting Up Windows Eclipse Programming of the RPi – Raspberry Pi Projects. Verkkodokumentti. IBEX. <<http://www.raspberry-projects.com/pi/programming-in-c/compilers-and-ides/eclipse/programming-the-rpi-in-windows-using-eclipse>>. Luettu 23.8.2015.
25. Native Build of Qt5 on a Raspberry Pi. 2015. Verkkodokumentti. QT. <[https://wiki.qt.io/Native\\_Build\\_of\\_Qt5\\_on\\_a\\_Raspberry\\_Pi](https://wiki.qt.io/Native_Build_of_Qt5_on_a_Raspberry_Pi)>. Päivitetty 27.6.2015. Luettu 19.8.2015.
26. ajstarks/openvg · GitHub. 2015. Verkkodokumentti. Github. <<https://github.com/ajstarks/openvg>>. Päivitetty 7.7.2015. Luettu 19.8.2015.

27. LiFePO<sub>4</sub> (3.2V) project, within a 3 x AA battery holder! Charge/Discharge curves. Verkkodokumentti. Manuka. <<http://www.instructables.com/id/Single-AA-LiFePo4-cell-powered-project-in-a-parti/step2/ChargeDischarge-curves/>>. Luettu 19.5.2015.
28. Le Dinhand, Phuc-Tue & Patry, Jacques, 2006. Video compression artifacts and MPEG noise reduction. Verkkodokumentti. <<http://www.embedded.com/design/real-time-and-performance/4013028/Video-compression-artifacts-and-MPEG-noise-reduction>>. Päivitetty 24.2.2006. Luettu 15.9.2015.
29. 2015 Best 3D Printers Guide. Verkkodokumentti. 3D Hubs. <<https://www.3dhubs.com/best-3d-printer-guide>>. Luettu 23.8.2015.
30. 3D-tulostuksen oppii kuten lukutaidon tai mopon virittämisen. Verkkodokumentti. 2014. eOppimiskeskus. <<http://eoppimiskeskus.fi/en/dcl/item/411-3d-tulostuksen-oppii-kuten-lukutaidon-tai-mopon-viritt%C3%A4misen>>. Päivitetty 5.5.2014. Luettu 20.9.2015.
31. PLA monomere (Polylactic Acid). Verkkodokumentti. Matbase. <<http://www.matbase.com/material-categories/natural-and-synthetic-polymers/agro-based-polymers/material-properties-of-polylactic-acid-monomere-pla-m.html#general>>. Luettu 20.9.2015.
32. About PLA. Verkkodokumentti. Makerbot. <[http://www.makerbot.com/support/new/3D\\_Printing/Knowledge\\_Base/PLA](http://www.makerbot.com/support/new/3D_Printing/Knowledge_Base/PLA)>. Luettu 20.9.2015.
33. Acrylonitrile Butadiene Styrene (ABS). Verkkodokumentti. Dynalon. <[https://dynalon.com/PublicStore/pages/ABS\\_Properties.aspx](https://dynalon.com/PublicStore/pages/ABS_Properties.aspx)>. Luettu 20.9.2015.
34. About ABS. Verkkodokumentti. Makerbot. <[http://www.makerbot.com/support/new/3D\\_Printing/Knowledge\\_Base/ABS](http://www.makerbot.com/support/new/3D_Printing/Knowledge_Base/ABS)>. Luettu 20.9.2015.
35. 3D-tulostus. Verkkodokumentti. Materflow Oy. <<http://www.materflow.com/fi/tekniikka>>. Luettu 20.9.2015.
36. Ylä-jääski, Vesa. 2014. Kolmiulotteista pursotusta. Tekniikan maailma 20, s. 76–81.
37. Kirjasto Omenan Pajassa on tekemisen meininki!. 2015. Verkkodokumentti. HelMet. <[http://www.helmet.fi/fi-FI/Kirjastot\\_ja\\_palvelut/Kirjasto\\_Omena/Palvelut/Kirjasto\\_Omenan\\_Pajassa\\_on\\_tekemisen\\_mei\(62611\)](http://www.helmet.fi/fi-FI/Kirjastot_ja_palvelut/Kirjasto_Omena/Palvelut/Kirjasto_Omenan_Pajassa_on_tekemisen_mei(62611))>. Päivitetty 25.3.2015. Luettu 22.9.2015.
38. NinjaFlex™ - SemiFlex 2.85mm. Verkkodokumentti. iMakr. <<http://www.imakr.com/en/filament/312-ninjaxflex-semiflex-285mm.html>>. Luettu 27.9.2015.



39. Raspberry Pi • View topic - I2C, SPI, I2S, LIRC, PPS, stopped working? Read this. 2015. Verkkodokumentti. Raspberry Pi Community. <<https://www.raspberrypi.org/forums/viewtopic.php?t=97314>>. Päivitetty 21.1.2015. Luettu 17.9.2015.

40. Device Tree. Verkkodokumentti. eLinux. <[http://elinux.org/Device\\_Tree](http://elinux.org/Device_Tree)>. Luettu 27.9.2015.

41. RFC 4251 - The Secure Shell (SSH) Protocol Architecture. 2006. Verkkodokumentti. IETF. <<https://tools.ietf.org/html/rfc4251>>. Päivitetty 1/2006. Luettu 12.7.2015.

