

Harri Tolkkinen

KÄYTTÖLIITTYMÄ FMCW-TUTKALLE JA PAIKKATIEDON SYNKRONOINTI TUTKADATAAN

Informaatioteknologian ja viestinnän tiedekunta

Kandidaatintyö

Toukokuu 2019

TIIVISTELMÄ

Harri Tolkkinen: KÄYTTÖLIITTYMÄ FMCW-TUTKALLE JA PAIKKATIEDON SYNKRONOINTI TUTKADATAAN

Kandidaatintyö

Tampereen yliopisto

Tietotekniikka

Toukokuu 2019

Automaation ja robotiikan kasvu on toiminut viime aikoina ajurina ympäristöä kartoittaville mittavälineille. Informaatiota kerätään eri teknologioilla koko sähkömagneettisen spektrin alueelta. Radioaalloilla toimivat tutkat eivät ole herkkiä sääolosuhteille ja kykenevät tuottamaan tietoa myös hyvin etäältä. Tutkalla voidaan tuottaa hyvin monentyyppistä dataa riippuen käytetystä menetelmästä ja tutkatyyppistä.

Synteettisen apertuurin tutkalla voidaan tuottaa valokuvamaista tutkadataa, jota on tyypillisesti tuotettu lentokoneeseen tai matalan kiertoradan sateliittiin asennetulla tutkalla. Viime vuosina markkinoille on tullut kohtuuhintaisia tutkia, joilla on mahdollista tuottaa SAR-kuvaa lyhyiltä etäisyyksiltä. SAR-kuvan muodostaminen vaatii antennilla vastaanotetun signaalin lisäksi tietoa antennin paikasta ja asennosta mittausten aikana. Mikäli antennin rataa ei pystytä määrittelemään etukäteen riittävällä tarkkuudella, voidaan antennin kulkema rata määrittää myös antenniin kiinnitetyllä inertianavigointiyksiköllä.

Koneoppimismenetelmillä on toistaiseksi toteutettu runsaasti hahmontunnistusta näkyvän valon aallonpituudella otetuille kuville, mutta hahmontunnistusta voidaan tehdä myös esimerkiksi SAR-kuvasta. Koneoppimisjärjestelmien toteutuksessa on usein haasteena opetusdatan kerääminen. Tämän työn tarkoitus on hahmotella lyhyen kantaman SAR-kuvaa tuottava järjestelmä, sekä kehittää graafinen käyttöliittymä hahmontunnistuksessa hyödynnettävän datan keräämisen helpottamiseksi.

Avainsanat: tutka, radar, FMCW, SAR, ISAR, Doppler, kohteentunnistus, Libusb, antenna, aperture, antenni

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck -ohjelmalla.

ALKUSANAT

Tämä työ oli varsin käytännönläheinen vilkaisu tutkatekniikan maailmaan, mutta toisaalta antoi minulle uudenlaista näkökulmaa myös signaalinkäsittelyn keskeisimpien tekniikoiden taustaan. Tahdon kiittää työni ohjannutta Juha Jylhää, sekä tutkija Minna Väilää antamastanne ohjauksesta ja tuesta projektini aikana. Kiitos kuuluu myös kämppekaverilleni Heikki Virtaselle, tekstinkäsittelytyökalun vaihtaminen kannatti.

Työn aikana kertynyt henkinen pääoma osoitti jälleen, että maailma on täynnä mahdollisuuksia kokea oppimisen elämyksiä. Täytyy vain löytää hyviä syitä tarttua niihin.

Tampereella, 6. toukokuuta 2019

Harri Tolkkinen

SISÄLLYSLUETTELO

1	Johdanto	1
2	Tutkateoria	2
2.1	Pulssitutka ja kantoaaltoitutka	2
2.2	Taajuusmoduloitu kantoaaltoitutka	3
2.3	Kulmaresoluutio ja resoluutiosolu	5
2.4	Micro-doppler	5
2.5	SAR	6
2.6	ISAR	7
3	FMCW-tutkan ohjausohjelmisto	8
3.1	Tutkan C++ -ohjelmistorajapinta	8
3.2	Libusb	8
3.3	Käyttöliittymä	10
3.4	Mittaukset ja data-analyysi	10
4	SAR-toteutus	14
4.1	Paikkatieto	14
4.1.1	Satelliittinavigaatio	14
4.1.2	Inertianavigaatio	15
4.2	Tiedonsiirto	15
4.3	Paikkatieto SAR laskennassa	16
5	Yhteenveto	18
	Lähdeluettelo	19

KUVALUETTELO

2.1	Periaatekuva tutkasta.[1]	2
2.2	FMCW-tutkan lähetetty ja vastaanotettu signaali.[7]	4
2.3	Kulmaresoluutioiden ja etäisyysresoluution muodostama solu.[7]	5
3.1	Työssä kehitetyn käyttöliittymän ulkoasu.	10
3.2	Soppiheijastimen testimittauksen järjestely.	11
3.3	Kokeellinen mittaus saha-aallolla, bandwidth=2Ghz, sweeptime=4ms, tal- lennusaika=10s.	12
3.4	STFT laskettuna yhdelle etäisyysolulle.	13
4.1	Inertianavigaatiojärjestelmän päätyypit.[5]	16

LYHENTEET JA MERKINNÄT

C_0	valonnopeus väliaineessa
D	fyysisen antennin leveys
S_i	pulssin i kompleksinen kaiku
S_r	etäisyysresoluutio
τ	pulssin pituus
f_D	dopplertaajuus
f_R	vastaanottotaajuus
f_T	lähetystaajuus
t_R	pulssin vastaanottoaika
t_T	pulssin lähetysaika
AD-muunnin	analogia-digitaalimuunnin
CSV	tiedostotyyppi, jossa tietokentät on erotettu pilkulla (comma separated value)
DGPS	korjausdataa käyttävä GNSS-toteutus(differential global positioning system)
FMCW	taajuusmoduloitu kantaalto (frequency modulated continuous wave)
ISAR	käänteinen synteettisen apertuurin tutka (inverse synthetic aperture radar)
isochronous	isokroninen, tasaisin väliajoin tai säännöllisesti tapahtuva
LiDAR	optinen tutka (Light Detection and Ranging)
PID	USB-ohjaimen laitetunnus(Product ID)
RTK	reaaliaikaista korjausdataa käyttävä GNSS-toteutus(real time kinematics)
SAR	synteettisen apertuurin tutka (synthetic aperture radar)
SD-radar	ohjelmistotutka(software defined radar)
STFT	ikkunoittain laskettu fourier muunnos(short-time fourier transform)
VID	USB-ohjaimen valmistajatunnus(Vendor ID)

1 JOHDANTO

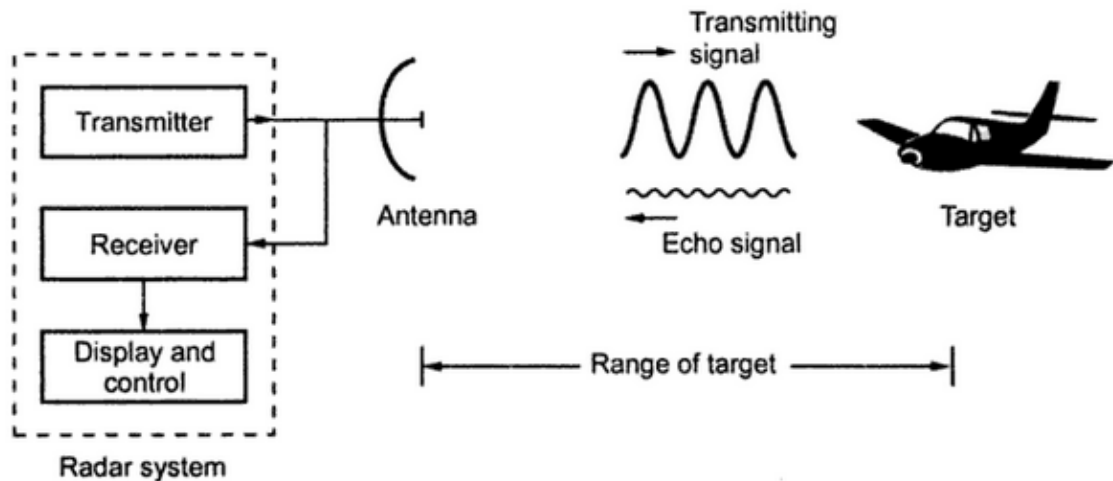
Automaation ja robotisaation lisääntyminen on viime aikoina lisännyt kiinnostusta havainnoida, mitata ja kartoittaa ympärillämme olevaa maailmaa entistä tarkemmin. Teknologiairytykset ovat suorastaan kaataneet rahaa lähes muoti-ilmiöksi nousseisiin koneoppimisteknologioihin ja itseajaviin autoihin. Infrapuna-alueella toimiva LiDAR-teknologia on nostanut päätään uudelleen, ja löytänyt tiensä esimerkiksi Googlen omistaman Waymon itseajavien autojen ytimeen[10]. Lähes jokaisen auton puskureista löytyy kaikuluotaimen tavoin toimivat peruutustutkan sensorit, ja kamerateknologioissakin on edetty muuttuvissa sääolosuhteissa kuvaamisen saralla.

Radioaalloilla toimiva tutkateknologia ei ole poikkeus. Perinteisesti viranomaistyökaluksi mielletty tutkateknologia on muodostanut jalansijaa muun muassa autojen adaptiivisissa vakionopeussäätimissä sekä hätäjarrutusavustimissa[4]. Tutkalta vaaditaan toisaalta erittäin korkeaa resoluutiota erityisesti etäisyysuunnassa, mutta toteutuksen on oltava myös taloudellisesti perusteltavissa henkilöautokäyttöön. Pääsääntöisesti tieliikenteen ongelmia on ratkaistu FMCW-tutkalla, jolla saavutetaan muun muassa hyvät etäisyys- ja Doppler-resoluutiot. Sivutuotteena FMCW-tutkien yleistymisen on tuonut saataville keino tuottaa lyhyen kantaman SAR-kuvaa esimerkiksi koneoppimiseen perustuvilla kohteentunnistusjärjestelmille.

Tässä työssä tutustutaan FMCW-tutkan toimintaperiaatteeseen, esitellään tutkan käyttöön liittyvät parametrit sekä kehitetään datan keräämistä helpottava graafinen käyttöliittymä erään valmistajan FMCW-tutkalle. Työssä tutustutaan myös pintapuolisesti SAR- ja ISAR-kuvaan, sekä hahmotellaan teoreettinen SAR-toteutus hyödyntäen FMCW-tutkaa. Järjestelmää on tarkoitus hyödyntää SAR- ja ISAR-hahmontunnistuksen kehityksessä. Toisessa luvussa käsitellään muutamaa erilaista tutkatyyppiä, pohditaan tutkien resoluutiota ja selitetään SAR-kuvantamisen idea. Kolmannessa luvussa tutustutaan työssä käytettyyn FMCW-tutkaan, perehdytään tutkan ohjaamiseen ja esitellään tutkalla suoritetuista mittauksista. Neljännessä luvussa pohditaan FMCW-tutkan soveltamista SAR-kuvan tuottamisessa. Viides luku sisältää työn aikana vedetyt johtopäätökset sekä yhteenvedon työn tuloksista.

2 TUTKATEORIA

Tutka on radioaaltoja hyödyntävä, perinteisesti puolustusteollisuudessa, lennonjohdossa tai merenkulussa käytetty havainnointilaitte, jolla pyritään tuottamaan tietoa etäällä olevan kohteen paikasta ja nopeudesta. Maaliin lähetetään radioaaltoja, jotka kohteesta sirotesaan muodostavat vastaanottimella havainnoitavan tutkavasteen (Kuva 2.1). Tutkan etuna esimerkiksi näkyvän valon aallonpituudella tehtävään havainnointiin on suuri kantama ja säänkesto.[8]



Kuva 2.1. Periaatekuva tutkasta.[1]

2.1 Pulssitutka ja kantoaaltotutka

Karkealla tasolla tutkia voidaan jaotella esimerkiksi käytettävän lähetetyypin perusteella pulssi- ja kantoaaltotutkiin. Kantoaaltotutka lähettää tasaista kantoaaltoa, ja vastaanottaa maalista palaavaa kaikua toisella antennilla. Mikäli maali on liikkeessä, tapahtuu heijastuvassa signaalissa Doppler-ilmiöstä johtuva taajuussirto, joka pystytään erottamaan, kun vastaanotettua taajuutta verrataan lähetettyyn taajuuteen.

Pulssitutka puolestaan lähettää lyhyitä pulsseja. Mittaamalla pulssin lähetyksen ajanhetken t_T ja takaisin heijastuneen pulssin vastaanoton ajanhetken t_R välistä aikaa

$$t_T - t_R = \Delta t \quad (2.1)$$

pystytään päättämään kohteen etäisyys d kaavalla

$$d = \frac{\Delta t \cdot c_0}{2}, \quad (2.2)$$

kun tiedetään valonnopeus c_0 väliaineessa. Etäisyysresoluutio S_r saadaan ideaalisessa tapauksessa kaavasta

$$S_r = \frac{c_0 \cdot \tau}{2}, \quad (2.3)$$

jossa τ on lähetetyn pulssin pituus.[3]

Puhtaan kantoaaltotutkan ongelmana on etäisyystiedon puuttuminen. Tutkalla pystytään havaitsemaan Doppler-ilmiöstä johtuva taajuussiirto, mutta toisaalta on mahdotonta päätellä kuinka etäällä taajuussiirron aiheuttama maali on.[3] Pulssitutkan ongelmana puolestaan on, että hyvään etäisyysresoluutioon vaadittavasta lyhyestä pulssista on hankala tehdä taajuusmittausta, joka mahdollistaisi nopeuden määrittämisen Doppler-ilmiön avulla. Nopeusmittausta voidaan kuitenkin tehdä riittävällä resoluutiolla mittaamalla referenssin ja vastaanotetun signaalin välistä vaihe-eroa usean pulssin yli, jolloin edellytyksenä on tutkan oskilaattorien riittävä stabiilius. Tutka saa tyypillisesti yksittäisestä pulssista yhden näytteen, jolloin kohteen nopeuden yksikäsitteinen mittaaminen edellyttää tiettyä näytetiheyttä, eli pulssintoistoväliä. Suurempi pulssintoistotaajuus mahdollistaa nopeamman kohteen yksikäsitteisen nopeuden mittaamisen. Toisaalta yksikäsitteinen etäisyyden mittaaminen rajoittaa pulssintoistotaajuutta siten, että etäisyydellä d pulssien välillä on oltava vähintään aika ΔT etäisyyden kaavan 2.2 mukaisesti. Yksikäsitteiseen nopeuden määrittämiseen tarvitaan tietty määrä näytteitä, ja toisaalta yksikäsitteinen etäisyysikkunan koko rajoittaa näytteistystaajuutta.

Lyhyeen pulssiin voi myös toisaalta olla hankalaa saada riittävää lähetystehoa. Ongelmaa kompensoidaan lähettämällä pidempi pulssi, jonka taajuutta tai vaihetta moduloidaan siten, että lähetetty signaali sisältää ajoitukseen liittyvää informaatiota myös pulssin sisällä.[3] Nyt vastaanotetun signaalin käsittelyssä pystytään mittaamaan lähetetyn ja heijastuneen pulssin aikaeroa ΔT myös pulssin sisäisten markkereiden välillä, eikä ainoastaan pulssin nousevasta ja laskevasta reunasta.

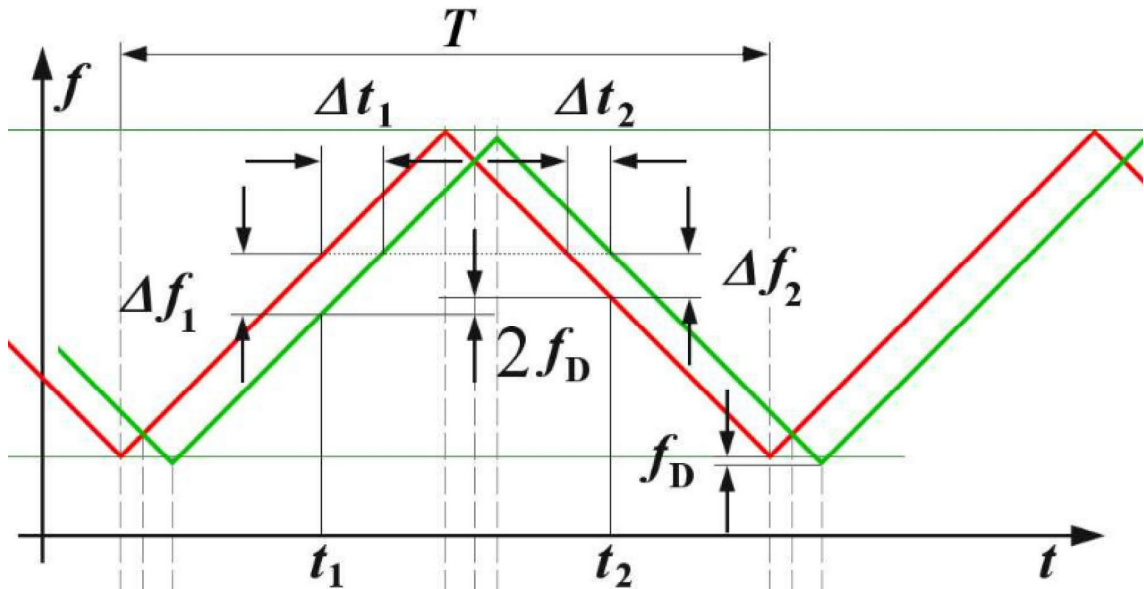
2.2 Taajuusmoduloitu kantoaaltotutka

FMCW-tutkassa yhdistyy sekä pulssitutkan, että kantoaaltotutkan ominaisuuksia. FMCW-tutkan kantoaallon taajuusmodulaatio muistuttaa pulssitutkan pulssikompressiossa käytettävää pulssin taajuusmodulaatiota. Tutkan lähettämää kantoaaltotaajuutta moduloidaan esimerkiksi kolmioaallolla siten, että kantoaaltotaajuus vaihtelee kaistanleveydellä f_B . Kolmioaallon tapauksessa yhden taajuuspyyhkäisyn aikana lähetystaajuus nousee lineaarisesti kaistan alhaalta ylös ja palaa takaisin alas kuvan(2.2) mukaisesti. Tutkan lähettämän hetkellisen kantoaallon taajuuden f_T , ja maalista heijastuvan, vastaanotetun

aallon taajuuden f_R välille muodostuu taajuusero

$$|f_T - f_R| = \Delta f \quad (2.4)$$

jota mittaamalla kyetään määrittämään radioaallon kulkema matka, kun moduloivan funktion pyyhkäisy nopeus tunnetaan.[3] Liikkuvan maalin tapauksessa vastaanotetun signaa-



Kuva 2.2. FMCW-tutkan lähetetty ja vastaanotettu signaali.[7]

lin taajuuteen vaikuttaa etäisyyden lisäksi kohteen nopeuden aiheuttama Doppler. Käytännössä koko vastaanotetun signaalin kaista siirtyy taajuustasossa (f_D , Kuva 2.2), minkä seurauksena lähetetyn ja vastaanotetun signaalin pyyhkäisykaistat eivät ole täysin päällekkäin, ja myös erotajuus Δf muuttuu. Dopplerin ja etäisyyden aiheuttamat taajuussirrot kyetään erottamaan hyvin toisistaan, kun taajuusmoduloivana funktiona käytetään esimerkiksi kolmioaaltoa, ja mitataan erotajuutta sekä aallon nousevalla, että laskevalla reunalla (Kuva 2.2). Nousevan reunan erotajuuden Δf_1 ja laskevan reunan erotajuuden Δf_2 välille syntyy liikkuvalla maalilla taajuusero

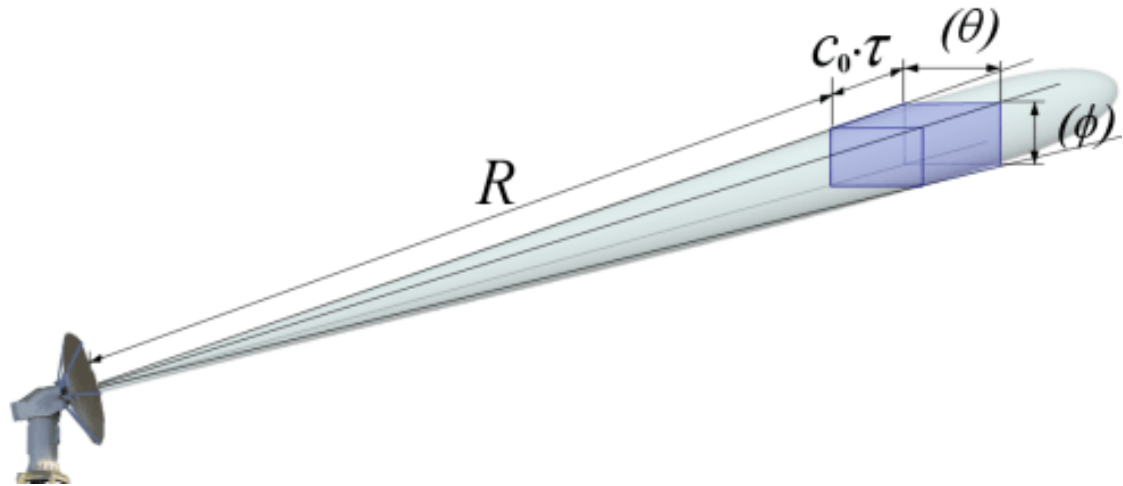
$$\Delta f_1 - \Delta f_2 = 2f_D \quad (2.5)$$

jonka perusteella myös kohteen nopeus saadaan määritettyä taajuusmittausta hyödyntäen. Yksittäinen taajuuspyyhkäisy voi kuitenkin olla niin lyhyt, ettei siitä saada riittävää taajuusresoluutiota f_D määrittämiseen, jolloin resoluutiota voidaan kasvattaa tekemällä mitausta usean modulaatiojakson yli. Yleisesti ottaen FMCW-tutkan etu pulssitutkaan verrattuna on toiminnan perustuminen suhteellisen helposti toteutettavissa olevaan taajuusmittaukseen, kun taas pulssitutkassa vaadittava ajoitusmittaus on teknisesti vaativampi[7]. Puutteena on toisaalta yksikäsitteisen etäisyysikkunan rajoittuminen muutamaankilometriin tällä hetkellä markkinoilla olevilla edullisilla AD-muuntimilla.

2.3 Kulmaresoluutio ja resoluutiosolu

Maalin etäisyyden ja nopeuden lisäksi on myös hyvin usein oleellista kyetä erottelemaan kohteiden suunta suhteessa havainnoijaan, jolloin etäisyyden avulla tiedetään kohteen paikka pallokoordinaatistossa. Usein koordinaatisto määritellään siten, että maapallon pinnalla olevalla tarkkailijalla horisontin suuntaisessa tasossa olevaa kulmaa kutsutaan atsimuuttikulmaksi, sekä horisontin ja kappaleen välistä kulmaa taivaalla elevaatiokulmaksi.

Tutkan antenni säteilee esimerkiksi kartion muotoisen keilan, jolloin käytännössä kulmaresoluutiot määrittää antennin suuntaavuus. Pallokoordinaatisto voidaan nyt jakaa kulmaresoluutioiden ja etäisyysresoluution avulla alkioihin, joissa tilavuus jakaantuu kuvan 2.3 mukaisesti resoluutiosoluihin.[7] Huomioitavaa on, että vaikka resoluutiosolun tilavuus riippuu vahvasti etäisyydestä, on teoreettinen etäisyysresoluutio kuitenkin sama maalin etäisyydestä riippumatta.



Kuva 2.3. Kulmaresoluutioiden ja etäisyysresoluution muodostama solu.[7]

Etäisyys- ja kulmaresoluutoista muodostuvaan kolmeen dimensioon voidaan liittää lisäksi neljäs dimensio kuvaamaan tarkastelutiheyttä, eli resoluutiota aikatasossa, sekä viides dimensio kuvaamaan Doppler-resoluutiota.

2.4 Micro-doppler

Tutkalla keilattavan kappaleen nopeudesta aiheutuvan Doppler-siirron lisäksi Doppler-ilmiötä aiheuttaa esimerkiksi kappaleen pyöriminen tai pinnan värähtely. Esimerkiksi lentävällä helikopterilla voi olla massakeskipisteen nopeudesta johtuva Doppler-taajuus, sekä pyörivistä roottorin lavoista aiheutuva efekti, josta käytetään usein nimitystä micro-Doppler tai lyhennettä μD . Helikopterin tapauksessa vastaanotetusta signaalista laskettuun Doppler-taajuuteen voi sisältyä vaihtelua, joka kyetään huomaamaan esimerkiksi Doppler-taajuuden vaiheen vaelteluna. Käytännössä micro-Dopplerin mittaaminen vaatii

siis riittävän määrän näytteitä näytteistettynä riittävän tiheästi, jotta mahdollisesti hyvin nopeat Doppler-taajuuden muutokset olisivat havaittavissa.

2.5 SAR

Kulmaresoluutioita voidaan kasvattaa fyysisen antennin suuntaavuuden parantamisen lisäksi myös hyödyntämällä useasta avaruuden paikasta mitattua dataa. Oletetaan teoreettinen kaksiulotteinen tapaus, jossa kohde on paikoillaan tien vieressä, ja tietä pitkin ajavan auton kyydissä on kohtisuoraan sivulle osoittava tutka (stripmap SAR). Tutkassa on antenni, jonka keilan leveys on esimerkiksi 16° . Tutka ajaa kohteen ohi tallettaen samalla mittauksia eri paikoista. Kohde on havaittavissa useammassa eri mittauksessa, jolloin kohteen paikan määrittystä saadaan tarkennettua yhden mittauksen resoluutiosolusta usean resoluutiosolun leikkauksen pinta-alaan. Useasta mittauksesta koostettavan synteettisen apertuurin muodostuksessa on luonnollisesti edellytyksenä, että kohde on kaikkien yhdistettävien mittausten keilassa, jolloin synteettisen apertuurin maksimaalisen pituuden määrittää fyysisen antennin suuntaavuus ja etäisyys kohteesta[3].

Suuri etu fyysiseen anteeniin on, että synteettisesti koostamalla voidaan luoda valtavia apertuureja teoriassa ainoastaan yhdellä antennielementillä, kun fyysisellä antennilla toteutus olisi käytännössä mahdoton. Toinen synteettisen apertuurin merkittävä etu on, että periaatteessa sillä voidaan saavuttaa sama resoluutiosolun koko kaikilta etäisyyksiltä. Vaikka suurempi etäisyys maalista kasvattaakin yksittäisen mittauksen resoluutiosolun kokoa vakiona pysyvän antennin keilanleveyden takia, on maali myös vastaavasti näkyvissä useammassa mittauksessa, ja resoluutiosolun koko voidaan pitää samana muodostamalla pidempi apertuuri. Käytännössä rajoitteita asettaa kuitenkin antennin kulkemaan rataa ja asentoon liittyvien virheiden kasaantuminen apertuurin pidentyessä.[3]

Apertuurin koostamisessa on huomioitava, että apertuurin keskellä antennielementillä on lyhyempi matka maaliin kuin apertuurin reunalla. Tavallisen, tasaisesti syötetyn fyysisen antennin tapauksessa kaikki antennielementit vastaanottavat kaikkien elementtien lähettämien signaalien heijastukset, ja heijastukset summautuvat reaaliaikaisesti. Synteettisen apertuurin tapauksessa vain yksi antennielementti kerrallaan lähettää ja vastaanottaa maalista syntyvän heijastuksen, jolloin toisten antennielementtien lähettämä signaali ei vaikuta mittaukseen, eivätkä heijastukset summaudu reaaliaikaisesti, vaan summaminen voidaan tehdä myöhemmin vaihekoherentisti.[3] Heijastuksen kulkeman matkan vaihtelusta apertuurin eri osissa johtuva vaihesiirto voidaan korjata lähetetyn signaalin aallonpituuden avulla, jolloin kompleksisia näytteitä tietyltä etäisyydeltä tallentavan tutkan heijastus on

$$S = \sum S_i \cdot e^{j \cdot \Delta \phi} \quad (2.6)$$

jossa S_i on mitatus apertuurin indeksistä i ja $\Delta \phi$ on etäisyyden vaihekorjaus. Etäisyysalkioiden tasolla tehtävää karkeampaa korjausta kutsutaan liikekompensaatioksi, ja etäi-

syysalkion sisällä tapahtuvaa vaihekorjausta autofokukseksi.[9]

2.6 ISAR

Käänteisessä SAR-kuvaamisessa tutka pysyy paikallaan ja kohde liikkuu ja/tai pyörii. Kohteen pyöriessä heijastuneen aallon kokema Doppler muuttuu kappaleen pinnanmuodoista ja pyörimisestä riippuen, jolloin kappaleen siroittajille muodostuu oma vaihehistoria. Kunkin siroittajan vaihehistoriasta muodostetaan pikseli pyrkimällä etsimään sen vaihekäyttäytymisen kanssa parhaiten korreloiva integraalin kantafunktio. Oleellista ISAR-laskennassa on kuitenkin huomata, että heijastuneen kaiun vaiheeseen vaikuttaa kappaleen pyörimisestä johtuvien pinnanmuutosten lisäksi myös varsinainen koko kappaleen massakeskipisteen liikkeestä aiheutuva Doppler, jolloin inversio-ongelmassa täytyy kyetä kompensoimaan sekä kappaleen massakeskipisteen liiketila, että ratkaisemaan mitatun micro-Dopplerin aiheuttava pyörimisliike ja kappaleen pinnanmuodot.

3 FMCW-TUTKAN OHJAUSOHJELMISTO

Työssä käytetty FMCW-tutka on Ancortekin valmistama, 24 – 26 GHz taajuuskaistalla toimiva ohjelmistotutka. Tutkaa on mahdollista käyttää neljällä, kahdella, tai yhdellä vastaanottavalla antennilla ja yhdellä lähettävällä antennilla. Aaltomuodoista on valittavissa saha-aalto, kolmioaalto, kanttiaalto tai jatkuva kantoaalto. Taajuuspyyhkäisylle on seitsemän mahdollista vaihtoehtoa 0,125 – 8 ms väliltä. Näytteistykselle on neljä erilaista vaihtoehtoa, joista käytettävissä olevia rajoittaa vastaanottavien kanavien määrä.

3.1 Tutkan C++ -ohjelmistorajapinta

Valmistaja toimitti tutkan mukana GPLv3-lisensoidut lähdekoodit sekä Matlab-rajapinnalle että C++-rajapinnalle. Matlab-rajapinta on käytännössä wrapperi C++-koodin ympärille ja on toteutettu kääntämällä C++-funktioita Matlab-koodista kutsuttaviksi MEX-tiedostoiksi. C++-rajapinnan muodostaa SDRadar-luokka.

C++-rajapinnan käyttö on suhteellisen yksinkertaista. Alustetaan SDRadar-luokan olio, jonka jälkeen oliolla on aaltomuotoa, näytteistystä ja pyyhkäisyä koskevat julkiset jäsenmuuttujat, joihin halutut parametrit sijoitetaan, ja päivitetään tutkaan erillisellä funktiokutsulla. Seuraavaksi lasketaan tutkasta tulevan datan määrä, varataan keskusmuistista riittävä tila ja pyydetään tutkalta tallenne. Lopuksi saatu data kirjoitetaan tiedostoon oheistietoineen. Ancortek toimitti tutkan mukana esimerkkikoodin binääritiedoston luomisesta, sekä myös Matlab-koodin, jolla tiedosto voidaan ladata Matlabiin. Esimerkinomainen karstittu koodi C++ -luokan käytöstä on esitetty algoritmissa 3.1.

Käytännössä tutkan ohjelmallisessa käytössä täytyy lisäksi huomioida, että USB-puskurin koon täytyy muodostua käytettävän USB-paketin koon monikerraksi. Funktiokutsuihin täytyy myös liittää jonkin verran virheenkäsittelyyn liittyviä parametrejä, sekä suorittaa konversioita tietotyyppien välillä. Pääpiirteissään luokan käyttö on kuitenkin kohtuullisen suoraviivaista ja selkeää.

3.2 Libusb

Valmistajan toteuttama, tutkan mukana toimitettu SDRadar-luokka toteuttaa muun muassa tietokoneen ja tutkan välisen kommunikaation. USB-kommunikaatio on toteutettu useal-

```

1 {
2     // Alustetaan olio ja muodostetaan yhteys tutkaan
3     SDRadar4 *sdr = new SDRadar4 ();
4     sdr->initialize ();
5
6     sdr->Parameters.options.modulation = waveform;
7     sdr->Parameters.options.channel = channels;
8     sdr->Parameters.options.sweep_time = sweeptime;
9     sdr->Parameters.options.sampling_number = sampling_mode;
10
11     // Lähetetään asetukset tutkaan
12     sdr->update_radar_parameters ();
13     sdr->bandwidth_parameters_udpate(Freq_start , Freq_stop , SweepTime);
14
15     // Lasketaan USB buffer
16     NTS = max_data_rate * NTS_dec_factor * SweepTime / CH_Num;
17     buffer_size = 2 * CH_Num* (NTS / SweepTime) * desiredSeconds * 2;
18
19     // Suoritetaan mittausta ja kirjoitetaan tiedosto
20     buffer_char = new unsigned char[buffer_size];
21     bytes_received = sdr->request_data_synchronous (buffer_size ,
22                                                     buffer_char);
23     write_file (bytes_recived , metadata);
24 }

```

Algoritmi 3.1. Esimerkki SDRadar-luokan käytöstä.

- 1 Alustetaan libusb
- 2 Listataan USB-laitteet
- 3 Varataan halutun VendorID:n ja ProductID:n omaavan laitteen rajapinta
- 4 Asetetaan laitteen USB-ohjain resetoititilaan
- 5 Ladataan mahdollinen haluttu ohjaimen firmware laitteeseen
- 6 Palautetaan laite reset-tilasta
- 7 Claimataan halutun laitteen USB-rajapinta uudelleen
- 8 Suoritetaan haluttu siirto
- 9 Vapautetaan rajapinta

Algoritmi 3.2. USB-laitteen käyttö ohjelmallisesti.

la eri alustalla tuetun Libusb-kirjaston avulla. Kirjasto sisältää funktiot muun muassa kytettyjen USB-laiteiden listaamiseen, laitteiden USB-ohjaimen päivitykseen sekä datasiirtojen toteuttamiseen. USB-siirron toteuttaminen onnistuu pääosin algoritmissa 3.2 kuvalla tavalla.

USB-kommunikaation toteuttamisessa Libusb-kirjastolla on kuitenkin pieniä eroja eri käyttöjärjestelmien välillä, eikä myöskään Ancortekin toteuttama SDRadar-luokka toiminut Linux-ympäristössä. Tutkaa haluttiin kuitenkin käyttää Linux-pohjaisesta käyttöjärjestelmästä Windows-pohjaisen käyttöjärjestelmän sijaan, ja SDRadar-luokan toteuttamaan USB-kommunikaation jouduttiin työn aikana tekemään pieniä muutoksia.

Linux-pohjaisessa käyttöjärjestelmässä USB-laitteen rajapinta täytyy Windowsista poike-
ten vapauttaa käyttöjärjestelmäajurilta, ennen kuin se voidaan varata sovellusohjelmal-
le. Libusb-kirjasto sisältää funktion, jolla Linux-pohjaisessa järjestelmässä käyttöjärjestel-
mäytimen USB-ajuri voidaan irrottaa ennen sovellusohjelman kytkemistä USB-endpointtiin.
Vastaavasti käyttöjärjestelmän USB-rajapinta täytyy palauttaa käyttöjärjestelmän USB-
ajurille siirron jälkeen, jotta käyttöjärjestelmä pysyy ajantasalla kytketyistä USB-laitteista.
Käytännössä SDRadar luokan koodiin lisättiin algoritmossa 3.2 kohdan 2 jälkeen ajurin ir-
rottava funktiokutsu, ja loppuun kohdan 9 jälkeen käyttöjärjestelmän USB-ajurin takaisin
kiinnittävä funktiokutsu.

3.3 Käyttöliittymä

Algoritmossa 3.1 määritellyn toiminnallisuuden toteuttavan funktion lisäksi työssä kehitet-
tiin mittauksien tekemistä varten myös yksinkertainen graafinen käyttöliittymä Qt-grafiikkakirjastolla,
jolla käyttäjä voi asettaa tutkan parametrit ja suorittaa mittauksia. Käyttöliittymään sisäl-
lytettiin SAR-toteutusta ajatellen myös komponentit paikkatietoa tuottavaa moduulia var-
ten, sekä varattiin mahdollisuus CSV-tiedostossa määriteltävän useamman mittauksen
sarjan suorittavalle toiminnallisuudelle. Käyttöliittymän ulkoasu on esitetty kuvassa 3.1.

The screenshot displays the SDRadar application's user interface, organized into three main panels: RADAR, CAPTURE, and POSITION. The RADAR panel contains several dropdown menus and input fields for configuring radar parameters: Waveform (set to FMCW sawtooth), Channels (4), Sweeptime (125_us), Start_freq (24000), End_freq (26000), and Sampling number per sweep (max). The CAPTURE panel features a text input for Capture length (5), a radio button for Embedded position data, and two buttons: 'Execute capture from CSV' and 'Start capture'. The POSITION panel includes a series of input fields for tracking location and orientation: Number of satellites, Current position, GPS position, Yaw, Pitch, and Roll. At the bottom of the interface, a progress bar indicates that the current operation is 24% complete.

Kuva 3.1. Työssä kehitetyn käyttöliittymän ulkoasu.

3.4 Mittaukset ja data-analyysi

Tutkalla suoritettiin sisätiloissa testimittauksia. Tutkan edessä pöydällä muutaman metrin
päässä oli soppiheijastin, jota lähdettiin liikkuttamaan tutkasta poispäin mittauksen aika-
na. Mittausjärjestely on nähtävissä kuvassa 3.2. Soppiheijastin liikkui mittauksen aikana
muutamia kymmeniä senttejä siirtyen kokonaisuudessaan muutaman etäisyysalkion ver-

ran.



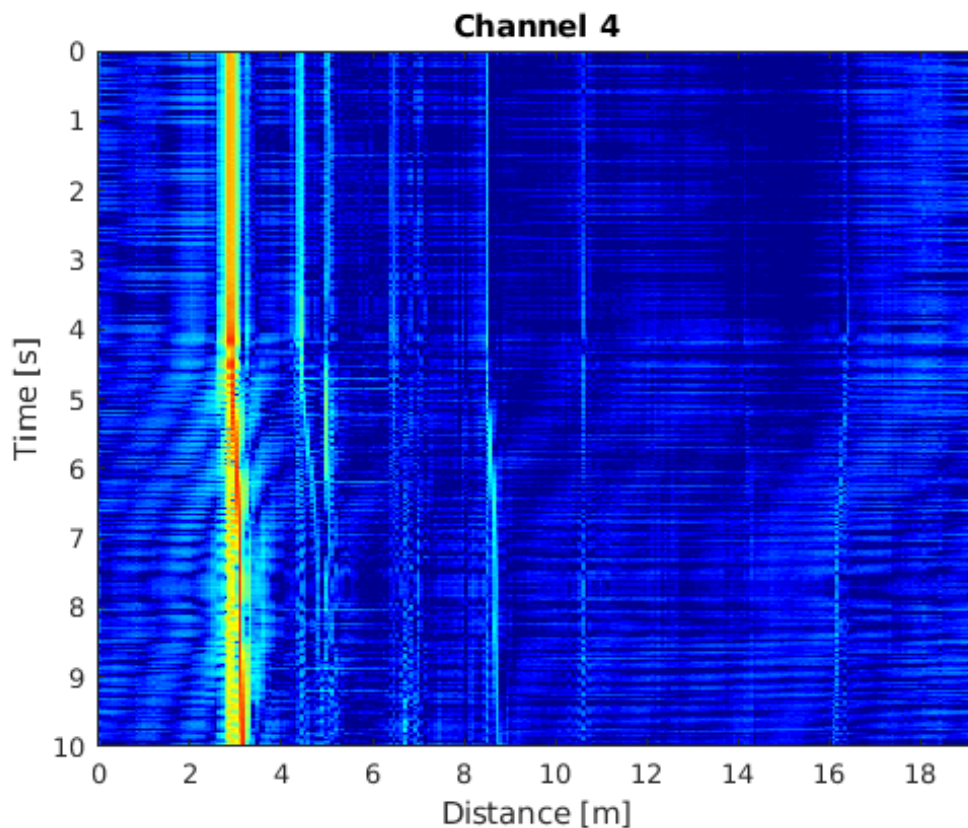
Kuva 3.2. Soppiheijastimen testimittauksen järjestely.

Yhden vastaanottokanavan data koostuu [$SamplesPerSweep$, $Sweepcount$] matriisista kompleksilukuja, jotka kuvaavat kullakin ajanhetkellä vastaanotetun signaalin amplitudia ja vaihetta. Kanavan datasta voidaan suodattaa häiriötä vähentämällä ensin keskiarvo yli koko

mittauksen. Suodatuksen jälkeen kuva 3.3 voidaan piirtää laskemalla käänteinen fouriermuunnos taajuuspyyhkäisyittäin. Aika-akseli muodostuu kuvaan taajuuspyyhkäisyjen määrästä sekä kestosta, ja etäisyysakseli taajuuspyyhkäisyksen kaistanleveydestä ja näytemäärästä. Esimerkkinä olevassa kuvassa 3.3 tutkalla mitattiin siis 2500 taajuuspyyhkäisyä 2 GHz kaistalla, jotka kestivät kukin 4 ms. Etäisyysikkunan kooksi muodostuu

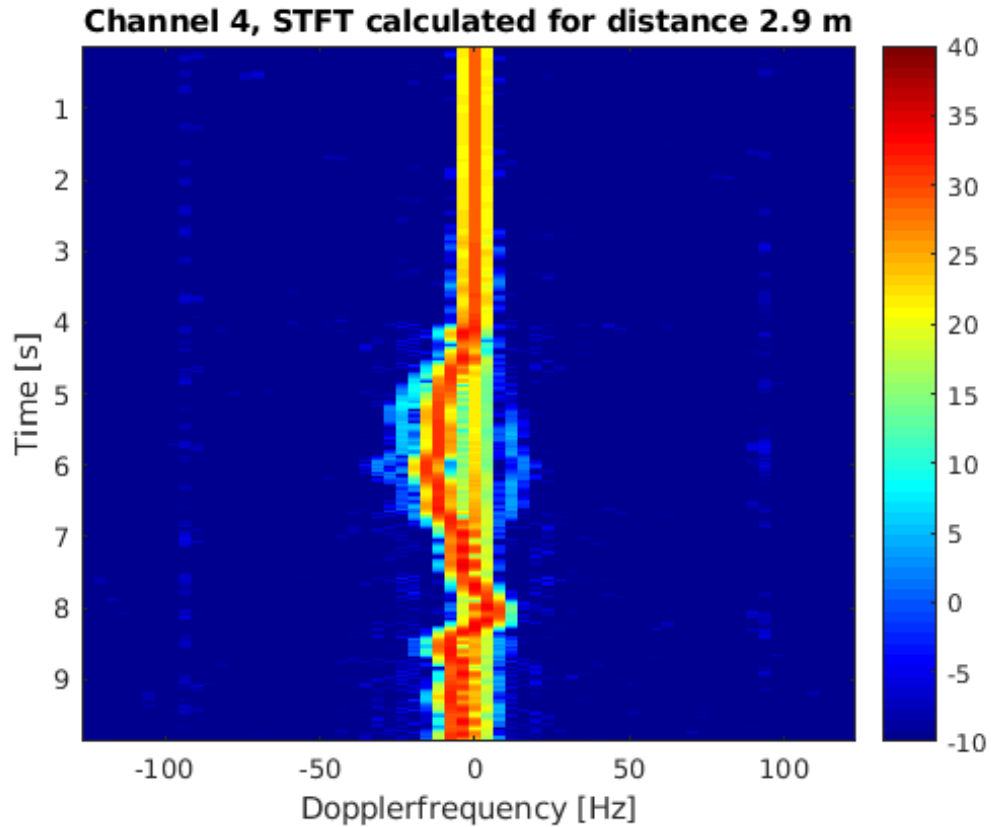
$$\frac{299792458m/s}{\frac{2GHz}{256}} \approx 19,2m. \quad (3.1)$$

Taajuuspyyhkäisyjen aikana tehtiin vastaanottavalla antennilla 256 mittausta, jolloin etäisyysikkuna jakaantuu vastaavaan määrään etäisyyssoleja. Tutkan vastaanottamaa häiriötä 24 – 26 GHz kaistan ulkopuolelta vähentää oleellisesti myös fyysisen antennin vahvistuksen heikkeneminen suunnitellun taajuuskaistan ulkopuolella.



Kuva 3.3. Kokeellinen mittaus saha-aallolla, bandwidth=2Ghz, sweeptime=4ms, tallennusaika=10s.

Toinen tapa tulkita dataa on laskea lyhyen ajan fourier muunnosta yksittäiselle etäisyyssolelle esimerkiksi Hamming-ikkunoina. Kompleksisesta matriisista tarkastellaan nyt siis taajuussisältöä tietyllä etäisyydellä aika-ikkunan sisällä. Kuva 3.4 on muodostettu laskemalla STFT kuvan 3.4 noin kolmen metrin päässä olevalle etäisyysalkiolle, eli STFT-ikkuna liukui sarakkeen ylhäältä alas kaikkien taajuuspyyhkäisyjen yli. Soppiheijastimen



Kuva 3.4. STFT laskettuna yhdelle etäisyyskolle.

liikkeellelähtö on nähtävissä kuvassa noin puolessa välissä mittausta. Karkealla tasolla kuvan 3.4 muodostus voidaan selittää muunnoksessa tehtävällä STFT:n muotoa $e^{j\omega t}$ olevien kantafunktioiden korreloinnilla soppiheijastimen etäisyysalkion vaihehistorian kanssa. Staattisessa tilanteessa mittauksen alussa datan kanssa korreloivat taajuudeltaan nollan lähellä olevat kantafunktiot, kun taas soppiheijastimen liikkeen aiheuttama lineaarinen vaihemuutos STFT-ikkunassa tarkasteltavassa vaihehistoriassa korreloi vahvasti Doppler-taajuudella olevan kantafunktion kanssa.

4 SAR-TOTEUTUS

SAR-kuvan laskentaa varten antennin paikka ja asento täytyy tietää mittauksen aikana, jotta synteettisen apertuurin muodostaminen olisi mahdollista. Mikäli SAR-kuvauksessa antennin liike on tavoitteena toteuttaa esimerkiksi ajoneuvolla tai ilma-aluksella, on hankala toteuttaa mittausta, jossa antenni kulkee riittävän täsmällisesti etukäteen määritellyn radan. Vaikka vaadittava apertuurin koko onkin työssä käytetyllä 24 – 26 GHz järjestelmällä lyhyiltä etäisyyksiltä vain metrien luokkaa, toteutus vaatisi vähintäänkin antenniryhmälle jonkinlaista vakautusjärjestelyä, joka lisää järjestelmään painoa ja voi osoittautua hankalaksi toteuttaa.

Työssä toteutusta lähestytään toisesta näkökulmasta. Antenniryhmän yhteyteen kiinnitetään mittaussyksikkö, joka tuottaa mittauksen aikana paikka- ja asentotietoa antenniryhmältä. Kun antennin kulkema rata ja asento tiedetään ajan funktiona, voidaan apertuuri määritellä vasta kuvaa laskettaessa. Haittapuolena on toisaalta syntyvän apertuurin muodostamisen monimutkaisuus, sillä mittauksien aikana valaistun alueen geometria voi olla hyvinkin erikoinen.

4.1 Paikkatieto

Paikkatiedon tuottamista käsitellessä on hyvä huomata, että vaikka erilaisia navigointitekniikoita käsitelläänkin erillisinä järjestelminä, on navigoinnissa usein järkevää hyödyntää useamman järjestelmän tuottamaa tietoa. Satelliitti- ja inertianavigaation tuottamaa dataa voidaan yhdistää sensorifuusion avulla, ja usein inertianavigaatiojärjestelmien(INS) yhteydessä puhutaankin satelliittivusteisesta inertianavigaatiosta(GNSS-Aided INS).[5]

4.1.1 Satelliittinavigaatio

GNSS(global navigation satellite system) on satelliitteihin perustuva navigaatiojärjestelmä, jossa navigaatiosatelliitit kiertävät maata keskimatkan radalla, noin 23000 km korkeudella. Järjestelmästä on useita rinnakkaisia toteutuksia kuten esimerkiksi eurooppalainen Galileo, kiinalainen Baidou, yhdysvaltalainen GPS ja venäläinen GLONASS. Käytännössä esimerkiksi älypuhelimissa toteutettavalla yksikanavaisella vastaantotolla saavutettava tarkkuus on maanpinnan tasossa muutama metrin ja korkeussuunnassa 2 – 3 kertaa heikompi. Tarkkuutta pystytään lisäämään jonkin verran esimerkiksi kuuluvilla olevien sa-

teliittien määrän kasvattamisella, ja osa GNSS-vastaanottimista pystyykin vastaanottamaan yhtäaikaaisesti esimerkiksi sekä GPS- että GLONASS-satelliittien signaalia, jolloin puhutaan monikanavaisesta vastaanotosta.[5] GNSS-paikannuksella voidaan määritellä myös asentoa, jolloin saavutettava asentotieto perustuu eri sateliiteista vastaanotettavien signaalien voimakkuuksiin ja vaatii sateliiteilta sopivaa geometriaa, sekä on toisaalta riippuvainen vastaanottimen antennigeometriasta.

Paikannukseen aiheuttaa epätarkkuutta signaalien vääristymät ilmakehässä. Virhettä voidaan korjata hyödyntämällä useampaa vastaanotinta. Majakkana toimiva vastaanotin pysyy paikallaan, jolloin se kykenee keskiarvoistamaan satelliittien signaaleista määritellyn paikkansa ajan kuluessa. Nyt mikäli majakka ja liikkuva vastaanotin ovat suhteellisen lähellä toisiaan, ilmakehän läpi tulevien reaaliaikaisten satelliittisignaalien voidaan olettaa vääristyvän kutakuinkin samalla tavalla molempien vastaanottimien näkökulmasta. Majakalta voidaan laskea reaaliaikaisen signaalin virhe keskiarvoistettuun paikkaan, ja lähettää radioteitse korjausdata liikkuvalla vastaanottimelle. Korjausdataa käyttäviä tekniikoita ovat esimerkiksi DGPS(differential GPS) ja RTK(real time kinematics), joista jälkimmäisellä kyetään muutamien senttimetrin tarkkuuteen.[6]

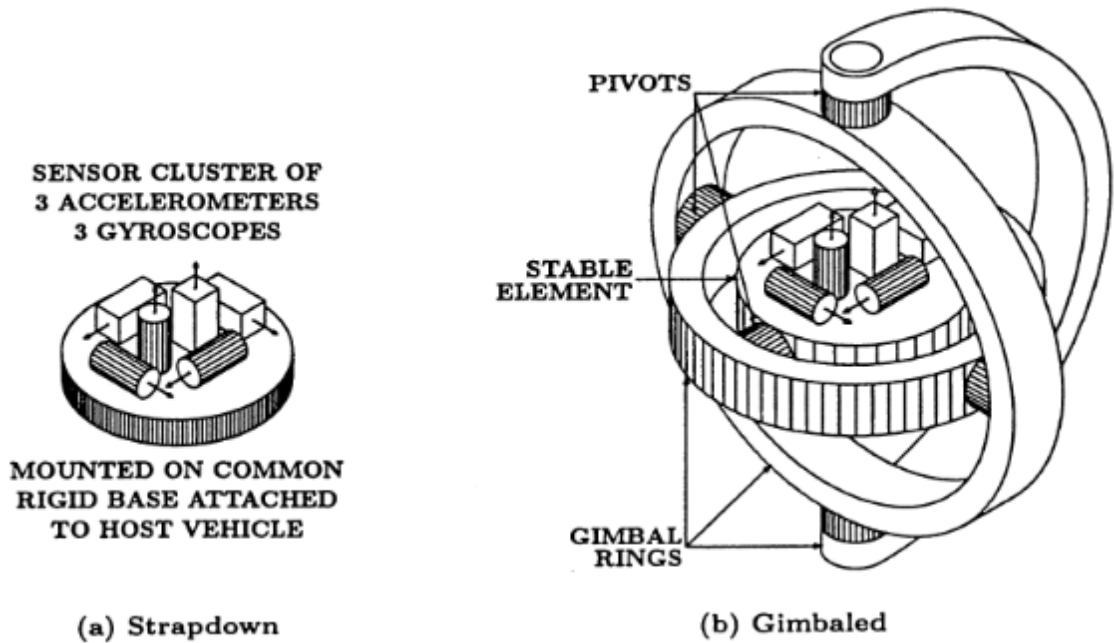
4.1.2 Inertianavigaatio

Inertianavigaatio muodostuu inertiamittausyksikön ja navigointitietokoneen yhdistelmästä [5]. Pääpiirteissään järjestelmät voidaan jakaa laitteen suoraan runkoon kiinnitettyä (strapdown) sensoripakettia käyttävään sekä gimbaloituun järjestelmään, jossa anturit on asennettu vakaajan sisälle. Periaatekuva järjestelmistä on esitetty kuvassa 4.1. Tyypillisesti modernit järjestelmät ovat gimbaloitonta tyyppiä.

Inertianavigaation etuja verrattuna satelliittinavigointiin ovat autonomisuus ja toisaalta erityisesti asentotiedon tarkkuus suhteessa edellisten ajanhetkien asentoon. Suurena etuna voi myös pitää käyttökohteesta riippuen suurempaa päivitysnopeutta. Ongelmana on, että ilman ulkoista referenssiä järjestelmän epävarmuus kasvaa lineaarisesti ajan kasvaessa virheiden integroitumisen myötä. Inertianavigaatiota käytettiin muun muassa Apollo-lennoilla.[5]

4.2 Tiedonsiirto

Järjestelmän kannalta oleellista on antennin paikka- ja asentotiedon synkronointi mitattuun tutkadataan. Ancortekin tutkan USB-kommunikaatio on toteutettu niin sanotulla synkronoidulla bulkkisiirrolla, mikä voi osoittautua ongelmaksi SAR-kuvantamisen toteuttamisessa. Bulkkisiirto mahdollistaa suuren siirtonopeuden ja virheettömän datan, mutta toisaalta ei anna minkäänlaista ajoitustakuuta datalle[2]. Käytännössä siis tutkalta pyydetään halutun kokoista datasiirtoa, jonka jälkeen sovellusohjelman suoritus pysähtyy siirron ajaksi, ja jatkaa seuraavalle riville kun suoritus palaa siirtokäskyltä. Bulkkisiirtois-



Kuva 4.1. Inertianavigaatiojärjestelmän päätyypit.[5]

sa paketeille lasketaan muun muassa tarkistesumma, ja virheelliselle paketille pyydetään uudelleenlähetyksi. Tyypillisiä käyttökohteita bulkkisiirrolle ovat esimerkiksi tiedostojen lähettäminen tulostimilla tai kuvien siirto skannerilta[2]. Videon tai äänen suoratoisto(stream) USB-kanavan läpi on yleisesti toteutettu isokroonisella siirrolla, jossa taataan tietyt rajat siirtonopeudelle ja viiveelle [2].

4.3 Paikkatieto SAR laskennassa

Tutkan ja tietokoneen välinen USB-siirto ei kuitenkaan välttämättä tarkoita sitä, etteikö tutkan vastaanottimilta tuleva datavirta olisi tasaista. Samalla suunnitellun SAR-järjestelmän kannalta ei välttämättä ole kriittistä USB-pakettien ajoitusten muuttuminen siirtopuskurissa, vaan olennaisempaa on, että kokonaisen siirron valmistuttua tutkan vastaanottamat näytteet ovat oikeassa järjestyksessä, ja näytteiden väli on kutakuinkin vakio. Paikkatiedon synkronoinnin kannalta taas tärkeää on, että tutkan tallennuksen aloitus saataisiin päätettyä jossain määrin tarkasti, jotta paikkatietomoduilta tulevasta aikaleimatusta asento- ja paikkadatavirrasta olisi valittavissa vastaava ajankohta.

Käytännössä ohjelman täytyy toteuttaa kahta erilaista toimintoa samanaikaisesti. Paikkatietomodulin tuottamia viestejä täytyy tallentaa tietorakenteeseen, ja toisaalta suorittaa tutkalta USB-siirtoa. Bulkkisiirron säikeen suorituksen keskeyttävästä luonteesta johtuen toteutus vaatii siis vähintäänkin kahden rinnakkaisen säikeen käyttöä. Alustava idea toteutukselle on käyttää kahta säiettä, joista toinen vastaanottaa paikkatietomoduilta esimerkiksi NMEA-viestejä sekä päivittää käyttöliittymää, ja toinen säie toteuttaa USB-siirron tutkalta.

Työssä käytetyn tutkan antennin keilan suuntaavuus on noin 16° . SAR-teorian mukaisesti maksimaalinen apertuurin pituus on keilan leveys mittausetäisyydellä. Esimerkiksi 75m päässä apertuurin maksimaalinen pituus on

$$L = 75m \cdot \sin 16^\circ \approx 20m, \quad (4.1)$$

mikä käytännössä tarkoittaa, että apertuurin pituus tulee kuljettua kävelen suoritettuna joissain kymmenissä sekunneissa. Kun otetaan lisäksi huomioon, että käytännössä usein ei ole tarpeellista käyttää maksimaalista apertuurin kokoa, vaan esimerkiksi etäisyys-suunnan 7,5 cm resoluutio saavutetaan liikesuunnassa jo noin 3,5 m pitkällä antennilla, tieto antennin liikeradasta on realistista tuottaa inertianavigointiyksiköllä hyvinkin taraksti.

Antennin liikerataan liittyvää epävarmuutta voidaan kuitenkin kompensoida myös SAR-kuvan tapauksessa esimerkiksi ISAR-kuvalla käytetyillä menetelmillä. Useamman etäisyys-solun tasolla tapahtuvaa virhettä voidaan korjata liikekompensaatiomenetelmillä, ja etäisyys-solun sisäistä virhettä autofokus-menetelmillä.

5 YHTEENVETO

Työssä kehitettiin käyttöliittymä eräälle FMCW-tutkalle, suoritettiin tutkalla testimittauksia, sekä esiteltiin mittausten analysointia. Opinnäytetyössä myös hahmoteltiin tutkalla SAR-kuvausta toteuttavan ohjelman rakenne. Kehitetty käyttöliittymä mahdollistaa mittausten suorittamisen sekä Windows- että Linux-ympäristöissä. Tutkan mukana toimitettua rajapintaluokkaa analysoitiin, ja todettiin, että toteutustapa voi aiheuttaa hankaluuksia paikkatiedon ja tutkadatan synkronoinnissa SAR-toteutusta ajatellen. Työn aikana kartoitettiin myös erilaisia paikka- ja asentotiedon tuottamiseen liittyviä tekniikoita, ja pääteltiin, että inertianavigaatiolla tuotettava paikkatieto soveltuu lyhyen kantaman SAR-kuvaamiseen hyvin. Työ on osa laajempaa kokonaisuutta, jossa kehitetään lyhyen kantaman SAR-kuvaan perustuvaa kohteentunnistusjärjestelmää. Kokonaisuutena tuloksia voidaan pitää varovaisen rohkaisevina, ja projektissa voidaan siirtyä eteenpäin sopivan inertianavigointiyksikön hankinnan jälkeen.

LÄHDELUETTELO

- [1] V. S. Bagad. *Radar Systems*. Technical Publications Pune, 2008.
- [2] *Beyond Logic. USB in a NutShell*. 2019. URL: <https://www.beyondlogic.org/usbnutshell/usb4.shtml> (viitattu 17.04.2019).
- [3] O. Klemola ja A. Lehto. *TUTKATEKNIikka*. Otatieto Oy, 1998.
- [4] I. Komarov ja S. Smolskiy. *Fundamentals of short range FM radar*. ARTECH HOUSE, INC, 2003.
- [5] S. G. Mohinder, R. W. Lawrence ja P. A. Angus. *Global Positioning Systems, Inertial Navigation, and Integration*. A John Wiley & Sons, Inc., 2004.
- [6] *PennState College of Earth and Mineral Sciences. Real-Time Kinematic and Differential GPS*. 2019. URL: <https://www.e-education.psu.edu/geog862/node/1828> (viitattu 18.04.2019).
- [7] *Radartutorial. Frequency-Modulated Continuous-Wave Radar (FMCW Radar)*. 2018. URL: <http://www.radartutorial.eu/02.basics/Frequency%20Modulated%20Continuous%20Wave%20Radar.en.html> (viitattu 26.03.2019).
- [8] M. I. Skolnik. *Introduction to Radar Systems*. McGRAW-HILL BOOK COMPANY, 2001.
- [9] R. Vehmas. *Liikekompensatio- ja autofokusmenetelmät SAR- ja ISAR-tutkien signaalinkäsittelyssä*. Puolustusvoimien tutkimuslaitos, 2016.
- [10] *Waymo. Lidar*. 2019. URL: <https://waymo.com/lidar/> (viitattu 18.04.2019).