

Kohteen paikannustarkkuus käytettäessä interpolointiin
perustuvaa alipikselitason konenäkömenetelmää

Risto Sippola

11. tammikuuta 2008

Esipuhe

Tämä Pro gradu -tutkielma on tehty Jyväskylän yliopiston Matematiikan ja tilastotieteen laitoksella vuonna 2007. Työn ohjaajana toimi yliassistentti Pasi Koikkalainen.

Tutkielma on tehty Jyväskylän yliopiston Data-analyysin laboratoriossa osana TEKESin rahoittamaa Ruiskuvaluprosessin hallinta ja laadun parantaminen (IMPROC) -projektia. Tämän vuoksi haluan kiittää kaikkia projektiin osallistuneita yrityksiä. Erityisesti haluan kiittää projektin johtoryhmään kuuluneita Harri Savoa (Oy Juho Jussila), Mika Sievi-Kortetta (TEKES), Mikko Silvennoista (PERLOS Oyj) ja Pasi Tarvaista (Numerola Oy).

Projektin rahoittajien lisäksi haluan kiittää Data-analyysin laboratorion henkilökuntaan kuuluneita Michael Harasta, Ismo Horppua ja Anssi Lensua heidän avustaan työtä koskien. Erityiskiitokset osoitan työn ohjaajalle Pasi Koikkalaiselle, jonka tuki ja neuvot olivat korvaamattomia.

Jyväskylässä

19.12.2007

Risto Sippola

Tiivistelmä

Konenäköön liittyvä laaduntarkastus yleistyy jatkuvasti. Tästä johtuen olemme kehittäneet menetelmän, jolla kohteita voidaan paikantaa kohinaisesta kuvasta toistuvasti ja luotettavasti alipikselin tarkkuudella. Menetelmää on tarkasteltu sekä yksi-, että kaksiulotteisissa tapauksissa, käyttäen mahdollisimman yksinkertaista mallia tutkittavasta ilmiöstä.

Menetelmä perustuu interpolointiin, suodattamiseen ja toistuviin kuvanottoihin, eli toistokuviin. Interpolointimenetelmällä diskreetistä kuvasta voidaan muodostaa jatkuva sovite, mikä mahdollistaa kuvan resoluution parantamisen, ja sitä kautta alipikselintarkkuuteen pääsemisen. Kohteen sijainti havaitaan valitun suotimen ja interpoloidun sovitteen konvoluution avulla. Tarkasteltavat suotimet ovat gradienttiin perustuvat suotimet ja templaatti. Toistokuvien avulla pyritään kohinan vaikutuksen pienentämiseen. Menetelmän toimivuutta on tarkasteltu sekä teoreettisissa analyyseissa, että empiirisissä kokeissa.

Teoreettisissa tarkasteluissa esitellään työssä käytettävät tunnusluvut sekä yksi-, että kaksiulotteisissa tapauksissa. Lukuun sisältyy myös kirjallisuuskatsaus aiemmin tehdyistä paikallistamiseen liittyvistä julkaisuista. Tämän lisäksi olemme johtaneet kohteen paikallistamisessa syntyvän varianssivirheen tapauksessa, jossa interpolointimenetelmänä on yksiulotteinen lineaarinen interpolointi ja suotimena on mikä tahansa työssä tarkasteltavista suotimista.

Empiirisissä tarkasteluissa vertaillaan suotimia eri olosuhteissa ja erilaisten kohteiden kanssa. Tulokset esitetään muodossa, jossa ne ovat suoraan verrannollisia pikselikokoon, mikä mahdollistaa tulosten helpon tarkastelun alipikselintasolla. Tämän lisäksi tutkimme harhaa, joka syntyy kun toistokuvia ei käytetä ja esitämme menetelmän, jolla tämä harha on mahdollista eliminoida.

Sisältö

1 Johdanto	1
1.1 Kuvan taustalla oleva rakenne	3
1.2 Dimensiomittaus ja kohteiden sijainnin estimointi	4
1.3 Kohteen sijainnin määrittäminen diskreetistä aineistosta	6
1.4 Käytettävissä oleva kuva-aineisto ja tunnistustehtävät	7
1.5 Menetelmälliset vaihtoehdot alipikselintarkkuuden saavuttamiseksi	9
2 Kohteen sijainnin mittaaminen alipikselin tarkkuudella	14
2.1 Lyhyt kirjallisuuskatsaus sijainnin estimointiin diskreetistä aineistosta ilman interpolointia	16
2.2 Interpolointiin perustuvat menetelmät	20
2.2.1 Interpolointi	20
2.2.2 Interpolointi ja kohinainen aineisto	23
2.2.3 Resoluution kasvattaminen interpoloinnilla	25
2.3 Konvoluutio kohteen sijainnin mittaamisessa	26

2.4	Reunansuodattimen ja templaattien muodostaminen	29
2.4.1	Cannyn ja Marr-Hildrethin suodattimien muodostaminen	31
2.4.2	Derivaattasuotimet diskreetille aineistolle	33
2.4.3	Templaattimaskien muodostaminen	37
2.5	Interpolointiin perustuvien reunansuodattimien toteutuksesta	38
2.5.1	Fourier-muunnoksen käyttö konvoluution laskemiseksi	40
3	Kohteen sijainnin tunnistusmenetelmien teoreettinen tarkastelu	42
3.1	Mittauksen harha ja varianssi	42
3.2	Arvioita reunantunnistimilla saavutettavasta mittaustarkkuudesta	44
3.3	Cannyn analyysi löydetyn reunan sijaintivirheelle	52
3.3.1	Diskretisoinnin ja lineaarisen interpoloinnin vaikutus kohteen paikannusvirheisiin	55
3.3.2	Harha lineaarisen interpoloinnin alipikselintarkkuudenmenetelmissä	62
4	Työssä tutkittavien menetelmien evaluointi empiirisin kokein	65
4.1	Yksiulotteisen sigmoidisen reunan sijainnin havaitseminen	68

4.1.1	Yhteenveto yksiulotteisen sigmoidisen reunan havaitsemisesta	71
4.2	Kaksiulotteisen sigmoidisen reunan sijainnin havaitseminen	73
4.2.1	Yhteenveto kaksiulotteisen sigmoidisen reunan havaitsemisesta	76
4.3	Kiinteään diskretisoinnin alkupisteeseen liittyvä harha	80
4.4	Mittausstrategia ruuvitorneja hyväksikäyttäen	84
5	Johtopäätökset	87
A	Yksiulotteisen sigmoidisen reunan empiiristen tarkastelujen tulokset	1
A.1	Mediaani reunatoistoista templaattia käyttäen	1
A.2	Mediaani reunatoistoista DoG-suodinta käyttäen	2
A.3	Mediaani reunatoistoista LoG-suodinta käyttäen	2
A.4	Keskiarvo reunatoistoista templaattia käyttäen	3
A.5	Keskiarvo reunatoistoista DoG-suodinta käyttäen	3
A.6	Keskiarvo reunatoistoista LoG-suodinta käyttäen	4
A.7	Mediaani pikselitoistoista templaattia käyttäen	4
A.8	Mediaani pikselitoistoista DoG-suodinta käyttäen	5

A.9	Mediaani pikselitoistoista LoG-suodinta käyttäen	5
A.10	Keskiarvo pikselitoistoista templaattia käyttäen	6
A.11	Keskiarvo pikselitoistoista DoG-suodinta käyttäen	6
A.12	Keskiarvo pikselitoistoista LoG-suodinta käyttäen	7
B	Kaksiulotteisen sigmoidisen reunan empiiristen tarkastelujen tulokset	8
B.1	Keskiarvo pikselitoistoista käyttäen reunantunnistusmenetelmänä kaksisuuntaista lineaarista interpolointia ja templaattisovitusta.	8
B.2	Keskiarvo pikselitoistoista käyttäen reunantunnistusmenetelmänä kaksisuuntaista lineaarista interpolointia ja Laplacian of Gaussiania.	8
B.3	Keskiarvo reunatoistoista käyttäen reunantunnistusmenetelmänä kaksisuuntaista lineaarista interpolointia ja templaattisovitusta.	9
B.4	Keskiarvo reunatoistoista käyttäen reunantunnistusmenetelmänä kaksisuuntaista lineaarista interpolointia ja LoG-suodinta.	9
B.5	Keskiarvo pikselitoistoista käyttäen reunantunnistusmenetelmänä yksisuuntaista lineaarista interpolointia ja templaattisovitusta.	10
B.6	Keskiarvo pikselitoistoista käyttäen reunantunnistusmenetelmänä yksisuuntaista lineaarista interpolointia ja LoG-suodinta.	10
B.7	Keskiarvo reunatoistoista käyttäen reunantunnistusmenetelmänä yksisuuntaista lineaarista interpolointia ja templaattisovitusta.	11

B.8 Keskiarvo reunatoistoista käyttäen reunantunnistusmenetelmänä yksisuuntaista lineaarista interpolointia ja LoG-suodinta.	11
C Reunan sijainnin varianssin osien johtaminen	12
C.1 Kokonaisresponsi	12
C.2 Ilmiöstä johtuva kokonaisresponsin osa	14
C.3 Kohinasta johtuva kokonaisresponsin osa	15
C.4 Kokonaisresponsin kohinasta johtuvan osan neliön odotusarvo	16

Kuvat

1	Konenäköjärjestelmä ottaa kuvan tutkittavasta kappaleesta. Tuloksena saadusta kuvasta voidaan tarkkailla haluttuja ominaisuuksia.	1
2	Harmaasävykuva matkapuhelimen kuoresta, johon merkitty mitattavat dimensiot: Leveys1, Leveys2, Leveys3, Pituus1 ja Pituus2.	2
3	(a) Ideaalinen kohde $r(z) \subset f(x)$ pisteessä x_0^* . (b) Valaistuksen takia vääristynyt kuva. (c) Diskreetti kohinainen mittaus.	6
4	Vasemmalla: Ideaalinen ilmiö $f(x, y)$, jossa kohteena, joka halutaan havaita on ruuvitorni. Oikealla: Kohinaisesta ilmiöstä $F(x, y)$ otettu diskreetti kuva $I_{xy}(t) = \{F_{ij}(t)\}_{i=1, j=1}^{n_p, n_p}$	8
5	Kuvassa esitettyinä kaksi eri toistostrategiaa, joissa alemmassa tunnusluku lasketaan silotetusta kuvasta ja ylemmässä se saadaan keskilukuna joukosta siloittamattomista kuvista koostettuja sijainteja.	11
6	Havaitusta aineistosta saatu interpoloitu sovite voi vaihdella huomattavasti riippuen käytettävästä interpolointimenetelmästä: a) vakio, b) lineaarinen, c) epälineaarinen.	21
7	(a): Havaittu digitaalinen 2×2 kuva. (b) Intensiteetit diskretisointipisteissä $\{f_{ij}\}_{i,j=1}^2$, jotka ovat pikselien keskipisteet. (c) vierekkäiset diskretisointipisteet yhdistettyinä yksiulotteisella lineaarisella interpoloinnilla. (d) Lineaarinen taso suuntaan x . (e) Lineaarinen taso suuntaan y	23
8	(a) Interpoloitava kaksiulotteinen aineisto. (b) Yksiulotteisesti suuntaan x lineaarista interpolointia käyttäen. (c) Yksiulotteisesti suuntaan y lineaarista interpolointia käyttäen. (d) Kaksiulotteisesti interpoloitu aineisto.	24

9	Esimerkki ilmiön $f(x)$ sovitteesta $[\hat{F}(x)](t)$, johon on päästy, kun aineistona on ollut diskreetti ja kohinainen kuva $I_x(t)$	24
10	Mahdollinen 95%:n luottamusväli $\hat{F}(x)$:lle lineaarisessa interpoloinnissa, kohinalla ε	25
11	(a) Ilmiöstä saatu diskreetti aineisto $\{F_i(t)\}_{i=1}^{n_p}$. (b) Aineistosta lineaarisesti interpoloimalla saatu jatkuva sovite $[\hat{F}(x)](t)$. (c) Jatkuvasta sovitteesta muodostettu diskreetti aineisto $\hat{I}_x(t)$, jonka koko n_{sim} määräytyy interpolointivälin Δ_{sim} mukaan.	26
12	Korrelaatio $m(x) = \int_{-a}^a f(z+x)r(z)dz$, jossa funktio $r(z)$ on liu'utettu todellisen ilmiön $f(x)$ yli.	27
13	Derivative of Gaussian: a) yksiulotteinen, b) kaksiulotteinen	32
14	Laplacian of Gaussian operaattori: a) yksiulotteinen, b) kaksiulotteinen	32
15	Derivaatan approksimointi maskia $\mathbf{M}' = [-1, 1]$ käyttäen.	34
16	Kaksiulotteinen kuva voidaan konvoloida käyttäen yksiulotteisia maskeja rivi, tai sarake kerrallaan.	34
17	Kaksiulotteisella reunalla on suunta θ	35
18	Kaksiulotteiset derivaattaa approksimoivat maskit.	36
19	Reunantunnistus diskreeteillä Sobelin maskeilla 4×4 ikkunasta, sekä x -, että y -akselin suuntaisesti.	36

20	Kohdetta $r(z)$ sovitetaan funktioon $f(x)$ liu'uttamalla sen yli määrittelyalueen. Vasteessa $m(x)$ näkyvät suurina arvoina kohdat, joissa $f(x)$:n muoto vastaa $r(z)$:n muotoa.	38
21	Jos sovitettavan pinnan $r(z)$ integraali ei kohteen määrittelyalueella ole nol-la, saa korrelaation responssi eri arvot samanlaisilla kohteilla volyymin vaihtumisen johdosta. Jos sovitettava pinta on määritelty siten, että $\int_{-a}^a r(z)dz = 0$, ei tätä ongelmaa synny vaan kohteet havaitaan samalla tavoin volyyymista riippumatta.	39
22	Harha ja varianssi kaksiulotteisessa aineistossa.	44
23	Yhteenvedo teoreettisista analyyseista	45
24	Kohinattomaan sigmoidiseen reunaan on sovitettu kolme eri lineaarista sovitetta. Kun diskretisointi pisteet x_i osuvat tiettyyn kohtaan tulee sovitteesta harhainen reunan todellisen sijainnin x_0^* suhteen.	63
25	Vasemmalla: Yksiulotteinen ilmiö $f(x) = \frac{1}{1+e^{-x+0.5}}$. Oikealla: Ilmiöstä havaittu aineisto $D = \{I_x(t)\}_{t=1}^4$	69
26	Pikselivälin ja signaali-kohinasuhteen ($SNR = 50$ ympyröity viiva, $SNR = 20$ katkoviiva, $SNR = 10$ yhtenäinen viiva) vaikutus harhan ja keskihajonnan arvoihin templaattisovitusta, DoG-suodinta ja LoG-suodinta käyttäen.	72
27	Kuvassa esitettyinä tarkasteltava kaksiulotteinen reunafunktio $f(x, y) = \frac{1}{1+e^{-\beta_1 x + \beta_2 y + \beta_0}}$ (vasemmalla) ja siitä havaittu kuva $I_{xy}(t) = \{F_{ij}(t)\}_{i=1}^{n_p} = \{f(x_i, y_j) + \varepsilon_{ij}(t)\}_{i=1}^{n_p}$ (oikealla)	74
28	Templaatilla ja Laplacian of Gaussianilla havaittavat kaksiulotteisen sigmoidisen reunan sijainnin pikselihajonnat $\hat{sd}(\alpha_{xy})$ signaali-kohinasuhde ($SNR = 50$ ympyröity viiva; $SNR = 20$ katkoviiva; $SNR = 10$ yhtenäinen viiva) ja reunan leveyden suhde pikselikokoon $\frac{\Delta e}{\Delta p}$ huomioonottaen.	77

29	Templaattisovituksella ja LoG-suotimella havaittavat kaksiulotteisen sigmoidisen reunan sijainnin x -akselin suuntaiset keskineliövirheet $ Bias_x + (sd_x)$ signaali-kohinasuhde ($SNR = 50$ viiva, jossa pisteet päällä; $SNR = 20$ katkoviiva; $SNR = 10$ yhtenäinen viiva) ja reunan leveyden suhde pikselikokoon $\frac{\Delta e}{\Delta p}$ huomioonottaen.	78
30	$\frac{\Delta e}{\Delta p} = 2$: a) Lineaarinen interpolointi ja templaatti, b) Spline-interpolointi ja templaatti, c) Regressiospline ja templaatti, d) Lineaarinen interpolointi ja LoG.	82
31	Kuvassa on kaksi lineaarista sovitetta kohinattomalle sigmoidiselle reunalle. Molemmassa soviteissa reunan sijainnin ja lähinnä olevan diskretisointipiste etäisyys $\min\{ x_i - x_0^* \} = \frac{\Delta p}{2}$	83
32	Reunan sijainti y -akselilla perustuu ruuvitornien avulla havaittavaan pisteeseen 2.	84

Taulukot

1	Lyhyet kuvaukset julkaisujen sisällöistä	47
2	Luvussa tarkasteltavat olosuhteet ja sijainnin paikallistamismenetelmät. . .	67
3	Mittausstrategian avulla saadut reunan sijainnit yksisuuntaista lineaarista interpolointia ja templaattia käyttäen.	86

Notaatio

Työssä esitetyt symbolit ja operaattorit esitysjärjestyksessä:

Symboli/operaattori	Selitys
(x_0^*, y_0^*)	Kaksiulotteisen kohteen todellinen sijainti
$r(z, q)$	Kaksiulotteinen kohdefunktio
$f(x, y)$	Kohinaton kappaleen taustalla oleva kaksiulotteinen reaali maailman ilmiö
$r(x - x_0^*, y - y_0^*)$	Kohde sijoitettuna pisteeseen (x_0^*, y_0^*)
$F(x, y)$	Kohinainen havainto $f(x) + \varepsilon$
ε	Mittauskohina
(\hat{X}_0, \hat{Y}_0)	Kohteen sijainnin satunnaisestimaattori
(\hat{x}_0, \hat{y}_0)	Kohteen sijainnin havaittu estimaatti
$\hat{x}_0 y_0^*$	Kohteen sijainnin estimaatti x -akselilla ehdolla, että y -sijainti tunnetaan
$r(z)$	Yksiulotteinen kohdefunktio
x_0^*	Kohteen sijainti x -akselilla
\tilde{r}	Vääristynyt kohdefunktio
ε_i	Mittauskohina pisteessä x_i
I_x	Satunnainen kuvavektori $I_x = \{F_i\}_{i=1}^{n_p}$
n_p	Pikseleiden lukumäärä yksiulotteisessa tapauksessa
F_i	Diskreetti satunnainen ilmiön arvo pisteessä x_i , $F_i = F(x_a + (i - 1) \cdot \Delta p)$
X_a	Ensimmäinen satunnainen diskretisoinnin piste x -akselilla $X_a = X_1$.
Δp	Pikseliväli $\Delta p = X_i - X_{i-1}$
$I_{xy}(t)$	Satunnaisen kuvan t :s realisaatio
$n_p \times n_p$	Kuvamatriisin $I_{xy}(t)$ koko
$F_{ij}(t)$	Kuvan t :nnen realisaation pikseli, jonka sijainti (i, j) . $F_{ij}(t) = f(x_i, y_j) + \varepsilon_{ij}(t)$.
$\varepsilon_{ij}(t)$	Kuvan t :nnen realisaation pikselin (i, j) kohinatermi. Kohinatermit i.i.d

Symboli/operaattori	Selitys
$\mathbb{E}()$	Odotusarvo-operaattori
D	Käytettävissä oleva aineisto $D = \{I(t)\}_{t=1}^s$
s	Toistokuvien lukumäärä
$I(t)$	Havaittu t :s kuvamatriisi $I(t)$ (joko $I_x(t)$ tai $I_{xy}(t)$).
α	Alipikselintarkkuus/Pikselivirhe
$\hat{X}_0(t)$	Kuvasta $I_x(t)$ havaittu kohteen sijainnin estimaatti
$X_a(t)$	Kuvan t :nnen realisaation diskretisoinnin alkupiste x -akselilla
x_a^*	Kiinteä diskretisoinnin alkupiste $X_a = x_a^*$
X_a	Satunnainen diskretisoinnin alkupiste $X_a \sim U[x_a^* - \frac{\Delta p}{2}, x_a^* + \frac{\Delta p}{2}]$
$Bias()$	Harha
σ^2	Mittauskohinan varianssi
$T()$	Keskiluku
\hat{x}_0	Aineistosta havaittu kohteen sijainnin estimaatti
\bar{x}	Keskiarvo
$Med(\{x_i\}_{i=1}^m) = x_{r(\lceil \frac{m}{2} \rceil)}$	Mediaani joukosta $\{x_i\}$
$[a]$	Seuraavaan suurempaan kokonaislukuun pyöristävä operaattori
μ	Odotusarvo
$Var()$	Varianssi
$\hat{F}(x)$	Diskreetin kuvan $\{F_i\}_{i=1}^{n_p}$ pohjalta muodostettu jatkuva sovitte
$\min_{x_0}()$	Minimioperaattori x_0 :n suhteen
$[\hat{F}(x)](t)$	Interpoloinnin t :s realisaatio
$In()$	Interpolointimenetelmä
\hat{F}_i	Interpoloitu diskreetti piste $\hat{F}_i = \hat{F}(X_a + (i - 1) \cdot \Delta sim)$.
$\{\hat{F}_i\}_{i=1}^{n_{sim}}$	Interpoloitu diskreetti aineisto
n_{sim}	Interpoloidun diskreetin aineiston koko
Δsim	Interpolointiväli
\hat{I}_x	Yksiulotteinen interpoloitu kuva $\hat{I}_x = \{\hat{F}_i\}_{i=1}^{n_{sim}}$
n_{sim}	Yksiulotteisen interpoloidun aineiston \hat{I}_x koko
\hat{I}_{xy}	Kaksiulotteinen interpoloitu aineisto $\hat{I}_{xy} = \{\hat{F}_{i,j}\}_{i=1,j=1}^{n,m}$
$\Phi_k(x)$	Ortogonaali kantafunktio (Momenttifunktio)
α_k	Momenttien painoarvot
$\beta_k(x_0)$	Momenttien painoarvot x_0 :n suhteen
$\hat{F}_X(x y_k)$	Interpoloitu sovitte x -akselin suuntaisesti kohdassa y_k

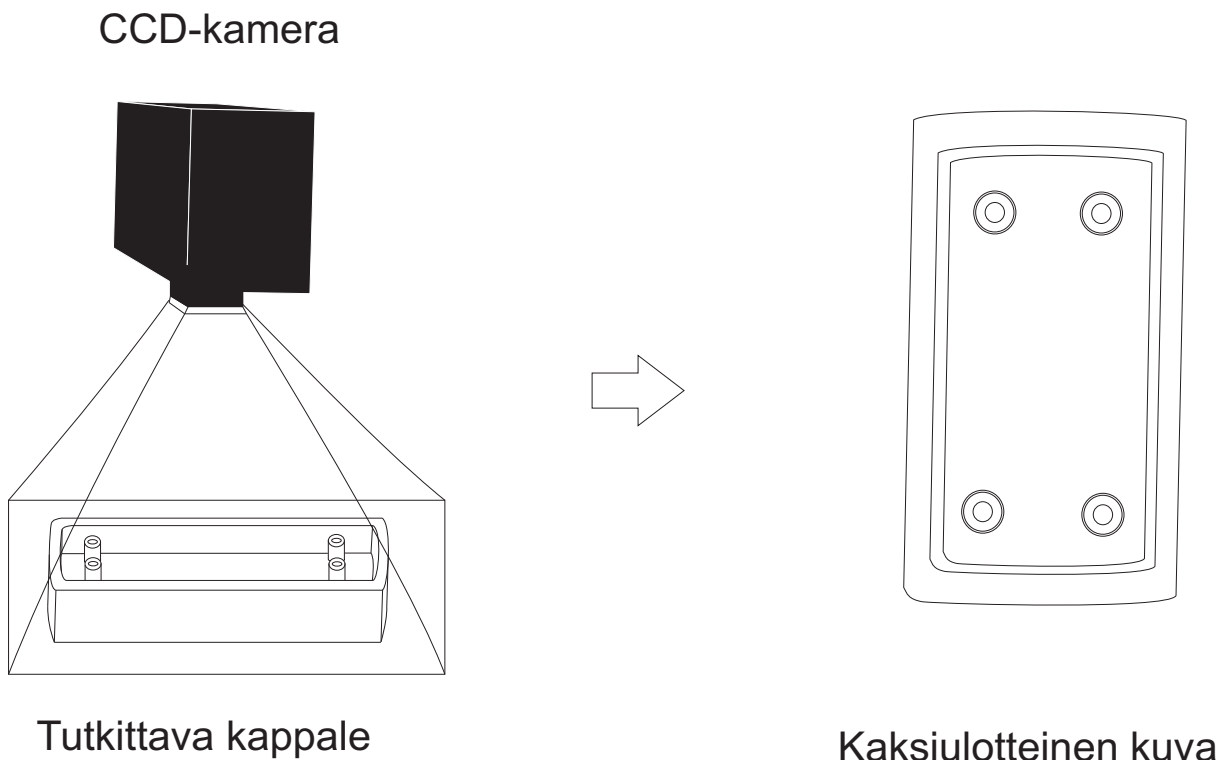
Symboli/operaattori	Selitys
f'	Funktion f ensimmäinen derivaatta
f''	Funktion f toinen derivaatta
$f * g$	Funktioiden f ja g konvoluutio-operaatio
g	Reunan paikallistamiseen käytettävä suodin
$m(x)$	Funktioiden konvoluution responsi $m(x) = (f * g)(x)$ Mitta, joka kertoo kuinka hyvin piste x edustaa pistettä x_0^*
$f \circ g$	Funktioiden f ja g korrelaatio
$\nabla()$	Differentiaalioperaattori
$m(x_i)$	Diskreetti konvoluution responsi
$\mathbf{F}(u)$	Fourier-muunnos funktiolle $f(x)$.
$\mathfrak{F}()$	Fourier-operaattori
$\mathfrak{F}^{-1}()$	Käänteinen Fourier-muunnos.
n_g	Yksiulotteisen diskreetin suotimen kuvapisteiden lukumäärä
k^0	Yksiulotteisen diskreetin suotimen kohdistuspiste
	Kaksiulotteisen diskreetin suotimen kohdistuspiste suuntaan x
k^0	Kaksiulotteisen diskreetin suotimen kohdistuspiste suuntaan y
\mathbf{M}'	Ensimmäistä derivaattaa approksimoiva maski
\mathbf{M}''	Toista derivaattaa approksimoiva maski
θ	Reunan suunta kaksiulotteisessa avaruudessa.
S	Gaussin funktiosta diskretisoitu silottava maski
\mathbf{T}'	Templaattimaski
G	Diskreetti reunasuodatin
$\partial()$	Osittaisdifferentiaalioperaattori.
\mathbf{M}_x	x -akselin suuntainen maski
FFT	nopea Fourier-muunnos
MSE	Keskineliövirhe
sd	Mittausten keskihajonta
$ Bias $	Absoluuttinen harha
$pixerr$	Pikselivirhe
$sd(\alpha)$	Pikselihajonta
$Bias(\alpha)$	Pikseliharha
h	Intensiteetin muutos kynnsreunassa Korkeampi intensiteetin arvo kynnsfunktiossa

Symboli/operaattori	Selitys
SNR	Signaali-kohinasuhde $\frac{h}{\sigma}$ Kohinaherkkyyden mitta
$CRLB$	Alempi Cramer-Rao raja
NMS	Non-maxima Suppression menetelmä
iid	Riippumattomasti samoinjakautuneita
T_y	Puolet reunasuotimen leveydestä y -akselin suunnassa
b	Alhaisempi intensiteetin arvo kynnsfunktiossa
ρ	Ideaalisen reunan etäisyys pikselin keskipisteestä
$m_T(x)$	Konvoluution totaaliresponsi
$m_N(x)$	Konvoluution responsin kohinasta johtuva osa
$m_f(x)$	Konvoluution responsin ideaalisesta ilmiöstä johtuva osa
$Err()$	Taylorin approksimaation virheosa
$S()$	Lineaarisen interpoloinnin virheen skaalatekijä
\bar{f}_k	Keskitetty funktion arvo $\bar{f}_k = f_k - f(\bar{x})$
$\hat{s}d$	Otoskeskihajonta
$\hat{\alpha}$	Otospikselivirhe
\hat{Bias}	Otosharha
\hat{ME}	Mediaanivirhe

1 Johdanto

Konenäön käyttö teollisissa tuotantoprosesseissa on yleistynyt viimeisten kahdenkymmenen vuoden aikana. Esimerkiksi ruiskuvaluteollisuudessa voidaan konenäön avulla tehdä reaaliaikaisia mittauksia tuotetuista kappaleista. Tämä taas mahdollistaa nopean ja halvan laaduntarkastusprosessin suhteessa eräisiin perinteisiin menetelmiin, kuten laboratoriomittauksiin ja ihmisnäköön perustuvaan laaduntarkastukseen. Verrattuna laboratoriomittauksiin on konenäköön perustuva laaduntarkastus sekä ajallisesti että rahallisesti tehokasta. Kun ajatellaan ihmisnäköön perustuvaa laaduntarkastusta, on koneellisesti mahdollista, ainakin teoreettisesti, päästä tarkempaan mittaustulokseen. Konenäköjärjestelmään perustuvan vaihtoehdon hyödyntämisen mahdollisuudet ovat kuitenkin jossain määrin rajalliset, koska tarkastelun kohde on pystyttävä määrittelemään tarkasti, mikä saattaa olla haasteellista monissa visuaalisen laaduntarkastuksen tehtävissä.

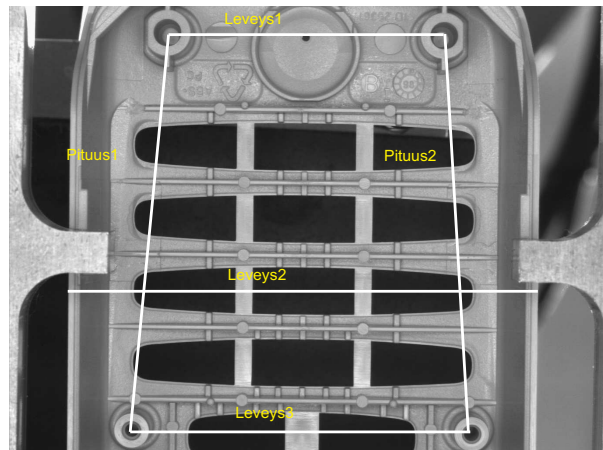
Konenäkö perustuu digitaaliseen kuvanottoon, jonka tuloksena saadaan kaksiulotteinen projektio kolmiulotteisesta reaalimaailman kappaleesta. Tämä tapahtuma on havainnollistettu kuvassa 1. Siihen kuinka tämä havainnointi tapahtuu tutustutaan työssä myöhemmin.



Kuva 1: Konenäköjärjestelmä ottaa kuvan tutkittavasta kappaleesta. Tuloksena saadusta kuvasta voidaan tarkkailla haluttuja ominaisuuksia.

Kuvasta oletetaan, että se on harmaasävykuva, eli siinä ei ole väritaaajuuskaistoja, vaan tietty rajattu määrä harmaasävyn arvoja, jotka muodostavat kuvan intensiteetin. Tämän työn yhteydessä tulemme olettamaan, että kuvat muodostuvat sileistä intensiteettifunktioista. Lisäksi kuvaa havainnoidaan diskreetillä hilalla kuvapisteitä, eli pikseleitä.

Kuvan intensiteettifunktion synnyttävät pääasiallisesti neljä eri tekijää, jotka ovat kappaleen geometria, heijastuvuus kappaleen pinnasta, valaistus ja kuvauskulma. Kuvassa nämä tekijät ovat sekoittuneina toisiinsa. Eräänä konenäköön liittyvänä ongelmana pidetään näiden tekijöiden erottamista toisistaan, sillä osaa näistä tekijöistä voidaan pitää jonkinasteisina häirtatekijöinä. Kiinnostavin kuvan intensiteettiin vaikuttava tekijä, josta tässä työssä halutaan tietoa, on kappaleen geometria. Ideaalissa tilanteessa kaikkien muiden tekijöiden vaikutus olisi minimoitavissa, jolloin kohteet pystyttäisiin havaitsemaan täydellisesti. Tämä on kuitenkin mahdotonta kuvanotossa syntyvien satunnaisten ilmiöiden, kuten kohinan, vuoksi [14].



Kuva 2: Harmaasävykuva matkapuhelimen kuoresta, johon merkitty mitattavat dimensiot: Leveys1, Leveys2, Leveys3, Pituus1 ja Pituus2.

Tässä työssä tutkittava kohde on ruiskuvalutekniikalla tuotettu muovikappale, jonka laatua pyritään tarkastamaan konenäön avulla. Eräs tehtävä on muovikappaleiden dimensioimittojen kuten pituuden ja leveyden tarkastaminen. Kuvassa 2 on esitettyä harmaasävykuva tuotetusta matkapuhelimen kuoren osasta, josta on tavoitteena mitata kolme leveys- ja kaksi pituusmittaa. Tässä työssä nämä mitat on havaittava noin kymmenen mikrometrin tarkkuudella. Kun käytettävän konenäkökameran resoluutio on maksimissaan pari tuhatta pikseliä ja tutkittavan kappaleen koko noin 5cm (tai 50mm), tulee mittaustehtävästä haastava. Järjestelmän on pystyttävä kameran resoluutiota tarkempaan mittaukseen. Tätä kutsutaan alipikselimittaukseksi (engl. subpixel).

1.1 Kuvan taustalla oleva rakenne

Konenäön hyödyntäminen mittaustehtävissä perustuu kuvankäsittelyn tekniikoihin. Nämä tekniikat vaihtelevat sovelluksesta riippuen, sillä konenäköongelman ratkaisu on riippuvainen taustaoletuksista.

Marr [14] lähestyy kuvankäsittelyä luonnollisen näön ja siihen liittyvien ominaisuuksien kannalta. Jotta muutosten havaitseminen kuvan geometriassa olisi mahdollista, tulisi kuvan intensiteetin muodostua siten, että geometria pystytään erottamaan muista tekijöistä, kuten suunnasta ja etäisyydestä. Kuvan siis tulisi muodostua kohteista, jotka voidaan havaita luotettavasti ja toistuvasti. Kohteilla voi olla ominaisuuksia (attribuutteja) kuten suunta, kirkkaus, koko ja sijainti. Lisäksi on tärkeää, että kuvasta havaitut ilmiöt johtuvat kohteiden todellisista fyysisistä muutoksista, eivätkä ole kuvankäsittelyn tuloksia.

Marr olettaa, että visuaalinen maailma koostuu sileistä heijastavista pinnoista, joiden spatiaalista rakennetta voidaan käsitellä yksityiskohtaisesti intensiteettifunktion kautta. Tämä rakenne on hierarkkinen, josta esimerkkinä olkoon kissan karva. Sitä voidaan tarkastella yksittäisenä visuaalisena oliona, mutta katsottaessa kauempaa karvat muodostavat turkin, joka sekin on käsiteltävissä rakenteeksi. Tämä tarkoittaa, että spatiaalinen rakenne koostuu useista eri prosesseista, jotka operoivat omassa skaalassaan. Työssä esitettävien periaatteiden tulee toimia samoin kaikilla skaaloilla. Toisaalta tulee ottaa huomioon, että tietyt kohteet tulee kyetä havaitsemaan vain määrättyllä skaalalla.

Kohteita, jotka määrittelevät kuvan rakenteen, ovat muun muassa suunnatut reuna- tai rajasegmentit; epäjatkuvuuskohdat suunnassaan; palkit tai niiden päättymiset ja pisarat. Näitä kohteita voidaan pitää primitiiveinä, eli alkeellisimpina pinnan rakenteen muodostavina muotoina, jotka voidaan määritellä joko abstraktisti tai konkreettisesti, kuten jatkumattomuuskohtien avulla. Konenäkö perustuu usein tällaisten kohteiden havaitsemiseen ja erottamiseen. Tästä tyypillinen esimerkki on kohteiden välisten etäisyyksien mittaaminen.

Eräs tyypillinen kohde, joka kuvasta halutaan havaita, on reuna. Digitaaliseen kuvankäsittelyyn liittyvää reunantunnistuksen ongelmaa onkin lähestytty useista eri näkökulmista viimeisten neljänkymmenen vuoden aikana. Reunantunnistuksen pyrkimyksenä on löytää kuvissa esiintyvät reunat. Tällöin reuna on pystyttävä määrittelemään yksikäsitteisesti.

Tehtävän ongelmallisuutta kuvaa hyvin, että vaikka reunantunnistuksen ongelmaa on tutkittu vuosikymmenten ajan, ei reunalle ole olemassa yleisesti hyväksyttyä määritelmää. Eri tutkijoiden näkemykset eroavat ainakin jossain määrin toisistaan, vaikka joissain perusasioissa on havaittavissa yhtymäkohtiakin. Muun muassa Hueckel [8], Torre ja Poggio [31], Marr [14], sekä Canny [2] ovat esittäneet näkemyksiään reunantunnistukseen.

Hueckelin [8] mukaan luonnolliset kuvat koostuvat yleensä objekteista, joissa on tarkasti määriteltyjä pintoja. Pintojen kaksiulotteiset projektiot koostuvat usein reunojen erottamista intensiteetin vaihteluista. Reunoilla on perustavaa laatua oleva rooli kuvan rakenteessa. Koska ihmisenäkö itsessään pystyy havaitsemaan reunat jo ennen informaation siirtymistä aivoihin, saattaa ihmisen tietoisuus aiheuttaa reunantunnistuksen ongelman vähättelyn. Tämä ei kuitenkaan muodostu ongelmaksi, sillä työssämme lähdemme liikkeelle oletuksesta, että kohde, jota etsitään, on funktiomuodoltaan ennalta tunnettu ja että reuna on aina havaittavissa intensiteetin vaihtelun kautta.

Suuri osa tunnistusprosesseista keskittyy kohteiden havaitsemiseen, mutta tässä yhteydessä emme ole niinkään kiinnostuneita reaalimaailman kohteiden havaitsemisesta ja niiden tunnistamisesta, vaan kuvista löydettävien kohteiden tai reunojen paikallistamistarkkuudesta.

1.2 Dimensiomittaus ja kohteiden sijainnin estimointi

Työmme tavoitteena on siis onnistuneen dimensiomittauksen suorittaminen. Käytännössä dimensiomitat määritellään kahden mittauspisteen välisenä etäisyytenä. Mittauspisteet puolestaan edustavat joitain kappaleen kuvasta tunnistettavia kohteiden sijainteja kuten reunoja, ruuvitorneja jne. Tämän johdosta dimensiomittausongelma onkin käytännössä kohteiden sijainnin estimointia kuvista.

Teoriassa kohteen sijainti on kaksiulotteisen reaaliakseliston piste (x_0^*, y_0^*) , joka on löydettävissä kohteen muodon perusteella. Formaalisti tämä voidaan esittää kohdetta kuvaavalla funktiolla $r(z, q)$, joka on osa laajempaa kappaletta kuvaavaa funktiota $f(x, y)$ siten, että

$$r(z, q) = r(x - x_0^*, y - y_0^*), \text{ kun } x_a \leq x \leq x_b, y_a \leq y \leq y_b.$$

Idealisessa tapauksessa sijainnin haku siis tarkoittaa sen pisteen (x_0^*, y_0^*) hakemista ideaalisesta aineistosta, jolle pätee

$$\int_{y_a}^{y_b} \int_{x_a}^{x_b} |f(x, y) - r(x - x_0^*, y - y_0^*)| dx dy = 0. \quad (1)$$

Todellisuudessa kohdetta ei voida löytää tarkasti, sillä ideaalista kappaleen muotoa $f(x, y)$ ei ole mahdollista havaita sellaisenaan. Tuotetut kappaleet ovat aina jossain määrin vääristyneitä, minkä lisäksi havainnoinnin yhteydessä ideaalinen muoto $f(x, y)$ korvautuu kohinallisella havainnolla $F(x, y) = f(x, y) + \varepsilon$, missä ε edustaa mittauskohinaa. Tästä johtuen kohteen sijainnin havaitsemisessa on kyse estimaattorista (\hat{X}_0, \hat{Y}_0) , josta havaitaan kohteen sijainnin estimaatti (\hat{x}_0, \hat{y}_0) havaitun aineiston perusteella. Tämä voidaan määrittellä esimerkiksi minimointina

$$(\hat{X}_0, \hat{Y}_0) = \arg \min_{x_0, y_0} \int_{y_a}^{y_b} \int_{x_a}^{x_b} |F(x, y) - r(x - x_0, y - y_0)| dx dy. \quad (2)$$

Kohteen sijainnin havaitsemisen tehtäviä on periaatteessa kolmea eri tyyppiä

- a)** Yksiulotteinen reunan hakutehtävä, jossa yksi dimensio on ennalta kiinnitetty. Yksinkertaisimmillaan esimerkiksi y_0^* on tunnettu ja tehtävänä on määrätä estimaatti

$$\hat{x}_0 | y_0^*$$

- b)** Yksiulotteinen reunan hakutehtävä, jossa toinen koordinaatti jää auki. Tavoitteena on tällöin määrätä reunan suuntavektori

$$y = a_1 x + a_0,$$

eli estimaatit kertoimille a_0 ja a_1 .

- c)** Kaksiulotteisen kohteen sijainnin estimaatti, eli pari

$$(\hat{x}_0, \hat{y}_0).$$

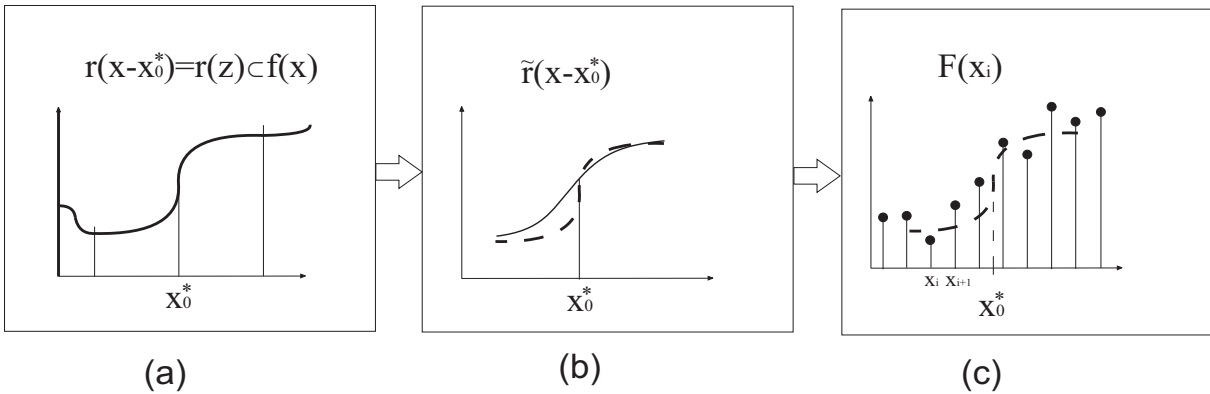
Tässä yhteydessä työ on päätetty rajata tilanteisiin **a)** ja **c)**, missä **a)** edustaa kohteen (reunan) yksiulotteista hakutehtävää ja **c)** on tilanne, missä kohteen sijainnin yksiulotteinen määrittely ei ole järjellinen, koska kohteelle ei voida asettaa suuntaa, kuten ruuvitornin tapauksessa (ks. kuva 4).

1.3 Kohteen sijainnin määrittäminen diskreetistä aineistosta

Kohteen sijainnin määrittäminen edellyttää, että kohde on riittävällä tarkkuudella identifiitavissa kuva-aineistosta. Reaalimaailman kuviin liittyy lisäksi useita tunnistusta vaikeuttavia seikkoja, jotka ovat muunmuassa

- i) kuvan diskreetti esitysmuoto,
- ii) kuviin liittyvä mittauskohina,
- iii) muotovääristymät, joiden taustalla ovat mm. heijastukset ja varjot.

Käytettävissä oleva data-aineisto voidaankin selittää kuvan 3 kaltaisen toimintaketjun kautta. Tässä tapauksessa tutkittava kohde on yksiulotteinen sigmoidin muotoinen reuna



Kuva 3: (a) Ideaalinen kohde $r(z) \subset f(x)$ pisteessä x_0^* . (b) Valaistuksen takia vääristynyt kuva. (c) Diskreetti kohinainen mittaus.

$r(z)$, joka sijaitsee pisteessä x_0^* , kun sijainti määritellään suhteessa kohteen pisteeseen $r(0)$. (kuva 3(a)) Mahdollisten vääristymien johdosta ideaalisen muodon $r(z)$ sijaan näemmekin muodon $\tilde{r}(z)$, mitä havainnollistetaan kuvassa 3(b). Käytössämme on lopulta kuvan 3(c) kaltainen diskreetti aineisto $F(x_i)$, $i = 1, \dots, n_p$, joka on mitattu pisteissä x_1, \dots, x_{n_p} ja joka sisältää mittauskohinaa siten, että $F(x_i) = \tilde{f}(x_i) + \varepsilon$, missä ε on kohinaa kuvaava satunnaismuuttuja. Jäljempänä työssä keskitytään diskretisoinnin ja kohinan vaikutuksiin ja vääristymien aiheuttamat vaikutukset sivuutetaan Tällöin malli on muotoa

$$F_i = f(x_i) + \varepsilon_i,$$

jossa mittauskohinatermit $\{\varepsilon_i\}$ ovat i.i.d., eli riippumattomasti poimittuja ja samoin jakautuneita.

Diskreetti aineisto on siis joukko intensiteettifunktion arvojen poimintoja reaaliakselilta (pikseleitä), jonka voimme yksiulotteisessa tapauksessa määritellä myös muodossa

$$I_x = \{F_i\}_{i=1}^{n_p} = \{F(X_a + (i-1) \cdot \Delta p)\}_{i=1}^{n_p},$$

missä X_a on diskretisoinnin ensimmäinen piste x -akselilla, n_p on pikselien lukumäärä ja Δp on kahden pikselin välinen etäisyys.

On hyvä huomata, että jos etsittävä kohde $r(z)$ sijaitsee pisteessä x_0^* , eli $r(x - x_0^*) = F(x)$, $x \in [x^a, x^b]$ niin on varsin epätodennäköistä, että sijainti x_0^* osuu tarkalleen jonkin diskretisointipisteen x_i kohdalle. Toisekseen diskretisointi suoritetaan kuvanotossa, jolloin diskretisoinnin alkupiste sijoittuu satunnaisesti ja diskretisointiväli Δp riippuu käytettävästä kamerasta. Tästä johtuen joudumme olettamaan, ettei diskretisointi riipu kohteesta.

1.4 Käytettävissä oleva kuva-aineisto ja tunnistustehtävät

Tunnistustehtävä voi olla joko yksi- tai kaksiulotteinen. Taustalla on kuitenkin aina kaksiulotteinen aineisto sillä digitaalisella kameralla tuotettu kuva-aineisto kuvanottohetkellä t muodostuu kaksiulotteisesta kuvapisteen matriisista

$$I_{xy}(t) = \{F_{ij}(t)\}_{i=1, j=1}^{n_p, n_p} = \{F(X_a + (i-1) \cdot \Delta p, Y_a + (j-1) \cdot \Delta p)\}_{i=1, j=1}^{n_p, n_p},$$

joka on työssä rajoitettu $n_p \times n_p$ neliömatriisiksi ja jonka kuvapisteen väli on vakio ja molempiin suuntiin sama, eli

$$X_{i+1} - X_i = Y_{j+1} - Y_j = \Delta p, \quad i, j = 1, \dots, n_p - 1.$$

Jos oletetaan, että reaali maailman ilmiö $f(x, y)$ ei muutu ajan t funktiona, kuvapisteen matriisi I voidaan tulkita näytteistykseksi ilmiöstä $f(x, y)$, missä poimintaan liittyy kohinaa, ja näyte pisteessä (x_i, y_j) edustaa keskiarvoista kohteen intensiteettiä kuvapisteen alueella kuvanottohetkellä. Näyte voidaan kirjoittaa

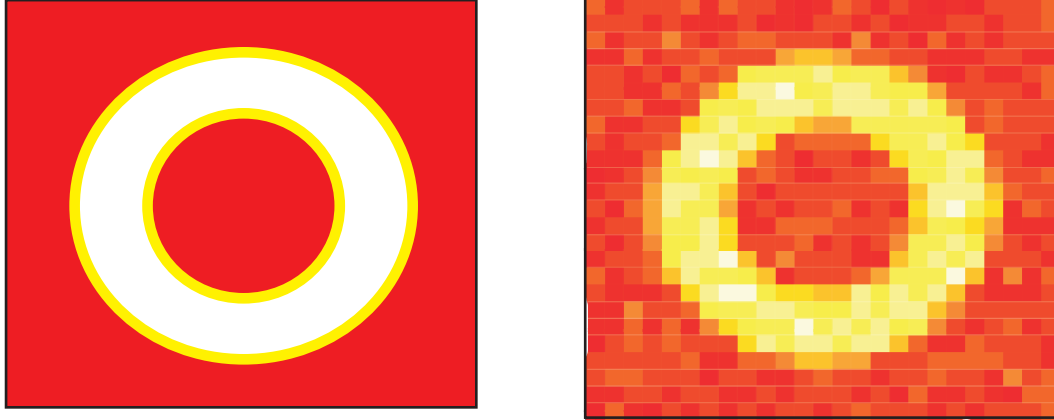
$$F_{ij}(t) = \frac{1}{\Delta p \cdot \Delta p} \int_{X_i}^{X_{i+1}} \int_{Y_j}^{Y_{j+1}} f(x, y) dx dy + \varepsilon_{ij}(t), \quad (3)$$

missä $\varepsilon_{ij}(t)$ on kuvanottoon liittyvä kohina, jonka voidaan olettaa olevan odotusarvoltaan nolla $\mathbb{E}[\varepsilon] = 0$. Näytteistysproseduuri löytyy tarkemmin selitettynä esimerkiksi Gonzalezin ja Woodsin teoksesta [7]. Tämän työn kannalta täsmällinen tieto kuvanmuodostuksesta on

kuitenkin epäolennaista ja voimme käyttää integraalin 3 sijaan yksinkertaisempaa oletusta näytteistykseksi, missä

$$F_{ij}(t) = f(x_i, y_j) + \varepsilon_{ij}(t).$$

Esimerkki tilanteesta, jossa ideaalisesta kaksiulotteisesta ilmiöstä on saatu kohinainen diskreetti kuva on esitetty kuvassa 4.



Kuva 4: Vasemmalla: Ideaalinen ilmiö $f(x, y)$, jossa kohteena, joka halutaan havaita on ruuvitorni. Oikealla: Kohinaisesta ilmiöstä $F(x, y)$ otettu diskreetti kuva $I_{xy}(t) = \{F_{ij}(t)\}_{i=1, j=1}^{n_p, n_p}$

Kohinan vaikutuksen eliminoimiseksi kohteesta voidaan ottaa useampia kuvia, eli käytettävissä oleva aineisto on muotoa

$$D = \{I(t)\}_{t=1}^s,$$

jossa $I(t)$ on joko yksi- tai kaksiulotteinen havaittu kuvamatriisi ($I_x(t)$ tai $I_{xy}(t)$).

Kohteen yksiulotteisen sijainnin tunnistustehtävä voidaan nyt määrittellä seuraavasti:

Olettaen, että ideaalinen kohde $r(z)$ on tunnettu, määritä sen sijainti x_0^* käyttäen diskreettiä havaittua aineistoa

$$D = \{I_x(t)\}_{t=1}^s = \{F_i(t)\}_{i=1, t=1}^{n_p, s}.$$

Tämän työn yhteydessä emme kuitenkaan tarkastele vääristymien vaikutusta tunnistukseen, vaan keskitymme pelkästään sijainnin x_0^* määrittämiseen diskreetistä aineistosta. Eri-

tyisenä haasteena on estimoida sijainti \hat{X}_0 suuremmalla tarkkuudella (pienemmällä ne-
liövirheellä) kuin mikä on kuvan pikseliväli Δp . Tämä tarkoittaa vaatimusta

$$\sqrt{\mathbb{E}[(\hat{X}_0 - x_0^*)^2]} \leq \Delta p \cdot \alpha,$$

missä α on tyypillisesti luokkaa 0.1, eli tunnistustarkkuuden on oltava pikselin kymme-
nesosaa parempi.

1.5 Menetelmälliset vaihtoehdot alipikselintarkkuuden saavuttamiseksi

Perinteisesti konenäössä kohteiden, esimerkiksi reunojen, sijainti on etsitty kuvapikse-
lin tarkkuudella. Perinteisiä reunantunnistusalgoritmeja löytyy konenäön perusteoksista
[18], [7], [24] ja katsauksia menetelmiin yleisemmin teoksista [31], [33], [15]. Kyseisten
menetelmien tarkkuus on riittävä, kun tavoitteena on tulkita kuvan sisältöä ja tunnistaa
siinä esiintyviä hahmoja. Tässä työssä tarkkuusvaatimus on suurempi, sillä tavoitteena
on hahmontunnistuksen sijaan mittaus. Kohteiden tunnistus siis oletetaan helpoksi, mut-
ta niiden sijainnista vaaditaan tietoa pikselitason menetelmiä suuremmalla tarkkuudel-
la. Tästä johtuen vain osa perinteisistä konenäössä käytettävistä kohteen- tai reunanet-
sintämenetelmistä soveltuu myös työssä tutkittaviin alipikselitason mittaustehtäviin.

Lähtökohtana oletetaan, että yhdestä tutkittavasta ilmiöstä $f(x)$ otetaan s kappaletta
kuvia ja niistä saadaan diskreetti aineisto $D = \{I_x(t)\}_{t=1}^s$. Tällöin diskretisoinnin alkupis-
teen voidaan olettaa olevan joko kiinteän $X_a(t) = x_a^*$, $t = 1, \dots, s$, tai satunnaisen X_a ,
noudattaen pikselin sisällä tasaista jakaumaa

$$X_a \sim \text{U}[x_a^* - \frac{\Delta p}{2}, x_a^* + \frac{\Delta p}{2}].$$

Jälkimmäisessä tapauksessa diskretisointipisteet vaihtuvat jokaisella mittauskerralla, eli
 $X_i(u) \neq X_i(v)$, $u \neq v$, minkä seurauksena myös alla oleva kohinaton ilmiö $f(X_i)$ tulee
satunnaiseksi.

Mitattaessa useita kappaleita, on luonnollista, että jokaisen uuden kappaleen osalta dis-
kretisoinnin alkupiste X_a on satunnainen. Sen sijaan saman kappaleen toistomittauksissa
tilanne riippuu kuvaolosuhteista kuten kameran tärinästä jne. Emme siis välttämättä

tiedä onko toistomittausten alkupiste satunnainen vai ei. Tällä on merkitystä myös lopputuloksen kannalta, sillä diskretisoinnin alkupiste saattaa aiheuttaa mittausharhaa

$$\text{Bias}(\hat{X}_0|X_a) = \mathbb{E}[\hat{X}_0|X_a] - x_0^*,$$

jota ei voida poistaa esimerkiksi kamerakalibroinnin avulla. Menetelmästä ja toistostrategiasta riippuen joudummekin tekemään oletuksia, joita käsittelemme seuraavaksi.

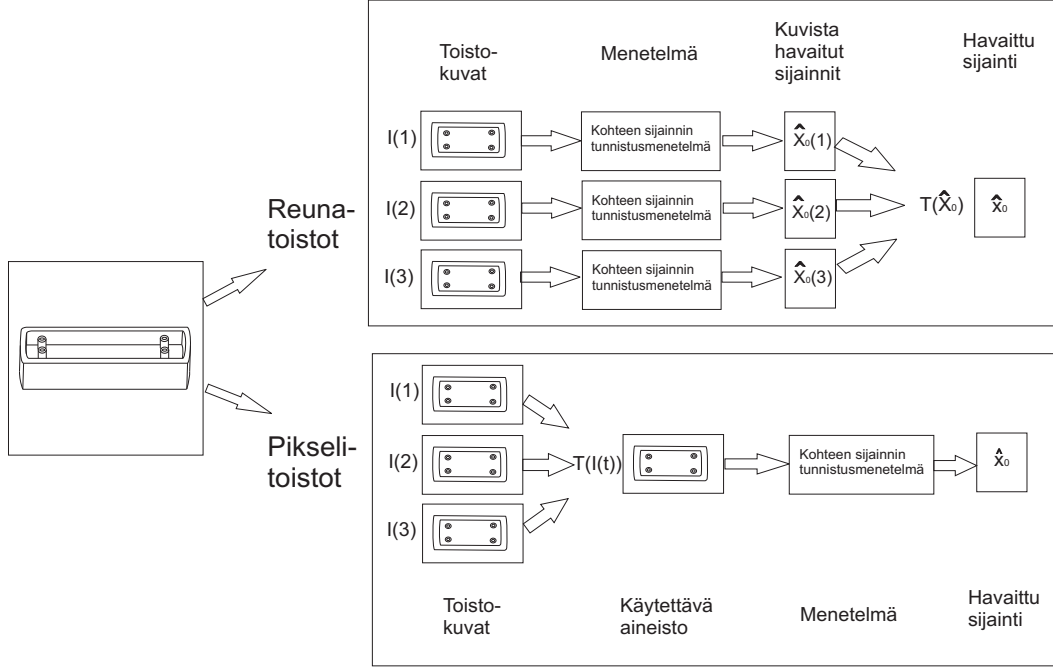
Määrittelemme kohteen sijainnin tunnistustehtävän siten, että sen tarkkuus määräytyy kahden eri menetelmän yhdistelmästä:

- kohinan vaikutusten minimointi toistomittausten avulla,
- sijainnin laskeminen pikseliresoluutiota tarkemmin interpoloinnin tai estimoinnin avulla.

Tässä työssä toteutettavat toistostrategiat on esitetty kuvassa 5. Mitattavasta kappaleesta otetaan joukko kuvia, joiden avulla mittauskohinan vaikutusta tuloksiin pyritään minimoimaan. Kuvassa näkyvä kohteen sijainnin tunnistusmenetelmä voidaan toteuttaa usealla tavalla, mikä esitellään tarkemmin luvussa 2. Nyt oletamme ainoastaan, että menetelmä suorittaa sijainnin tunnistuksen alipikselintarkkuudella. Tässä kappaleessa tutustumme sijainnin tunnistusmenetelmien vaihtoehtoihin vain pinnallisesti pääpainon ollessa toistostrategian esittelyssä.

Tapa, jolla suoritamme dimensiomittauksen, edellyttää siis sijainnihakumenetelmän sekä toistostrategian valinnan. Kohteen- tai reunan sijainnin tunnistusmenetelmä tulee siis valita siten, että sijainti pystytään havaitsemaan alipikselintarkkuudella ja toistostrategia siten, että se pienentää kohinan vaikutusta tunnistuksessa.

Kuten edellä on kerrottu, on yhdestä kappaleesta mahdollista ottaa useita kuvia, minkä avulla voidaan minimoida mittauskohinan vaikutusta tuloksiin. Jotta lisäinformaatiosta saatava hyöty olisi mahdollisimman suuri, tulee toistot hyödyntää käyttämällä jotain tilanteeseen sopivaa toistostrategiaa. Kuvassa 5 on esitetty kaksi tässä työssä sovellettavaa strategiaa. On hyvä huomata, että nämä eivät ole ainoat mahdolliset vaihtoehdot, sillä toistostrategiaa voidaan soveltaa myös sijainnin tunnistusmenetelmien eri vaiheissa.



Kuva 5: Kuvassa esitettyinä kaksi eri toistostrategiaa, joissa alemmassa tunnusluku laske-
taan silotetusta kuvasta ja ylemmässä se saadaan keskilukuna joukosta siloittamattomista
kuvista koostettuja sijainteja.

Kutsumme kuvan 5 alemmaa menetelmää **pikselitoistoiksi**. Olettamalla, että diskreti-
soinnin alkupiste $X_a(t)$ on vakio x_a kaikille toistokuville $t = 1, \dots, s$, kohdistuvat kaikki
pikselin i havainnot $\{F_i(t)\}_{t=1}^s$ samaan pisteeseen $x_i = x_a + \Delta p \cdot i$ tutkittavaa ilmiötä $f(x)$.
Kun lisäksi mittauskohina $\varepsilon_i(t)$ oletetaan riippumattomaksi mittauspisteestä i ja toistosta
 t , voidaan kuvissa olevaa kohinaa pienentää helposti korvaamalla alkuperäiset havainnot
 $F_i(t)$ silotetulla estimaatilla $T(\{F_i(t)\}_{t=1}^s)$, jossa $T()$ on jokin keskiluku Siis alkuperäinen
aineisto $D = \{F_i(t)\}_{i=1, t=1}^{n_p, s}$ korvataan silotetulla kuvalla

$$\bar{I}_x = \{T(\{F_i(t)\}_{t=1}^s)\}_{i=1}^{n_p}.$$

Tämän jälkeen reunan sijainti etsitään silotetusta aineistosta $D = \bar{I}_x$. Jos alkupiste ei
ole vakio, on pikselitoiston soveltaminen haastavampaa, sillä pikseli x_i edustaa eri mit-
tauskerroilla eri kohtaa ilmiöstä $f(x)$, jolloin $f(x)$:n vaihtelu ja mittauskohina sekoittu-
vat keskenään. Toisaalta kiinteän alkupisteen x_a tapauksessa joudumme olettamaan, että
 $Bias(\hat{X}_0|x_a) = 0$.

Kuvan 5 ylempi strategia, **reunatoisto**, on sovellettavissa suoraan myös, kun diskreti-
soinnin alkupiste X_a on satunnainen. Tässä aineiston $D = \{I_x(t)\}_{t=1}^s$, jokaisesta kuvasta
 $I_x(t)$ lasketaan reunan sijainti $\hat{X}_0(t)$. Tällä tavoin saadaan s toistoa reunan sijainnista
 $\{\hat{X}_0(t)\}_{t=1}^s$. Lopulliseksi reunan sijainniksi lasketaan silotettu tunnusluku $T(\{\hat{X}_0(t)\}_{t=1}^s)$,

joka on vähemmän kohinainen kuin alkuperäiset havainnot $\hat{X}_0(t)$. Jos diskretisoinnin alkupiste X_a on satunnainen ja sillä on vaikutusta löydettävään reunan sijaintiin $\hat{X}_0(t)$, on toistostrategialla myös havaitun sijainnin harhaa pienentävä vaikutus. Siis olettamalla diskretisoinnin alkupiste X_a satunnaiseksi on mahdollista saavuttaa harhaton reunatoistostrategia.

Toistostrategiasta riippumatta voidaan kohinan minimointiin käyttää samoja silottavia estimaattoreita. Olettaen, että silotettavat havainnot noudattavat jotain symmetristä jakaumaa, esimerkiksi

$$F_i = f_i + \varepsilon_i, \text{ missä } \varepsilon_i \sim N(0, \sigma^2), \text{ tai } \hat{X}_0 \sim N(0, \sigma^2)$$

voidaan havainnoista saatava informaatio tiivistää johonkin keskilukuun $T()$. Tässä työssä käytettävät keskiluvut ovat mediaani ja aritmeettinen keskiarvo. Toistostrategioilla on keskilukuja käytettäessä kohinaa poistava vaikutus. Kohinan suhteella käytettävään keskilukuun onkin tärkeä osa menetelmän onnistumisessa.

Mediaani on suuruusjärjestykseen asetettujen havaintojen keskimäinen arvo, eli

$$\text{Med}(\{x_i\}_{i=1}^m) = x_{r(\lceil \frac{m}{2} \rceil)},$$

missä $r(\lceil \frac{m}{2} \rceil)$ on järjestyksessä keskimäisen havainnon indeksi siten että

$$x_{r(1)} \leq x_{r(2)} \leq \dots \leq x_{r(\lceil \frac{m}{2} \rceil)} \leq \dots \leq x_{r(m-1)} \leq x_{r(m)},$$

kun $r()$ on havaintojen suuruusjärjestyksen osoittava järjestysfunktio ja $\lceil a \rceil$ tarkoittaa luvun a pyöristämistä ylöspäin seuraavaan kokonaislukuun. Yleensä oletetaan, että m on pariton, jolloin mediaanin molemmille puolille jää yhtä monta suurempaa sekä pienempää havaintoa. Mediaanin etuna on robustisuus, mikä tarkoittaa, että yksittäinen virrehavainto, oli se kuinka suuri tahansa, ei suuresti vaikuta estimaatin arvoon. Toisaalta mediaani ei yleensä ole yhtä tehokas kuin keskiarvo.

Toinen työssä tutkittava vaihtoehto on käyttää tunnuslukuna toistojen aritmeettista keskiarvoa. Kun X :stä otetaan s toistoa, on sen keskiarvo

$$\bar{X} = \frac{1}{s} \sum_{t=1}^s X(t). \quad (4)$$

Keskiarvo on yleisimmin käytössä oleva keskiluku. Aritmeettinen keskiarvo on lisäksi yksi tyhjentävistä tunnusluvuista useimmiten käytetyille jakaumille ja sen pohjalta tehtävä

tilastollinen päättely on hyvin tunnettua. Esimerkiksi, kun keskiarvo lasketaan toistuvasti samasta jakaumasta riippumattomasti poimituista havainnoista

$$X \sim N(\mu, \sigma^2)$$

on s :n havainnon kautta lasketun keskiarvon \bar{X} vaihtelun varianssi luokkaa

$$\text{Var}(\bar{X}) = \frac{\sigma^2}{s}.$$

Odotettavissa oleva mittausten keskihajonta siis pienenee tällöin vauhtia $\frac{1}{\sqrt{s}}$ toistojen määrää kasvatettaessa. Tämä on tehokkaampaa kuin mediaanilla. Toisaalta estimaattori ei ole robusti, vaan yksikin virheellinen havainto voi aikaansaada suuren virheen, vaikka toistojen määrä kasvaisi hyvin suureksi.

Yhdistelemällä edellä esitellyt menetelmät saadaan neljä erilaista toistostrategiaa: mediaani pikselitoistoista, keskiarvo pikselitoistoista, mediaani reunatoistoista ja keskiarvo reunatoistoista. Työn loppupuolella tutkimme, miten toistostrategiat vaikuttavat reunan havaittavuuteen, sekä teoreettisesti että empiirisesti. Kuten edellä on mainittu, ei pelkällä toistostrategialla päästä työssä vaadittavaan alipikselintarkkuuteen. Lisäksi tarvitsemme menetelmän, joka on mahdollista yhdistää toistostrategiaan ja joka mahdollistaa kohteen sijainnin havaitsemisen alipikselintarkkuudella. Seuraavassa luvussa tutustummekin menetelmiin, jotka mahdollistavat tämän.

2 Kohteen sijainnin mittaaminen alipikselin tarkkuudella

Kappaleessa 1.5 esitellyn alipikselitason mittaustehtävän ratkaisemiseksi tarvitsemme tarkoitukseen soveltuvan menetelmän. Useimmat menetelmät on kehitetty kohteentunnistukseen, joissa sijainnin tarkka mittaaminen on ollut toissijainen tehtävä. Tyypillisessä tunnistustehtävässä näet riittää kohteen sijaintitiedoksi kuvapikselin määräämä tarkkuus. Tässä työssä lähtökohta on päinvastainen. Oletamme, että tunnistustehtävä on pääosin ratkaistu päätehtävän ollessa kohteiden sijainnin mahdollisimman tarkka mittaaminen. On siis löydettävä tapa soveltaa perinteisiä pikselintarkkuuden menetelmiä alipikselin tarkkuudella tai kehittää täysin uusia menetelmiä.

Kun kohteena on reuna, voidaan tunnistus erään näkemyksen [21] mukaan tehdä kolmella eri tavalla: suodattamalla (filtering), templaattisovituksella tai stokastisella reunantunnistuksella. Näistä kaksi ensimmäistä toteutetaan yleensä pikselimaskin sovituksella kuvapikseleihin, jolloin edellä esitellyistä menetelmien tyypeistä ainoastaan stokastisessa reunantunnistuksessa päästään suoraan alipikselin tarkkuuteen. Suodattamalla ja templaattisovituksella alipikselin tarkkuuteen pääsemiseen tarvitaan yksi lisävaihe, interpolointi. Tästä johtuen jaamme käytettävät menetelmät karkeasti kahteen luokkaan:

- a) Interpolointiin perustuvat menetelmät, joissa kuvan resoluutiota kasvattamalla voidaan hyödyntää myös pikselitason menetelmiä.
- b) Suorat sijainnin estimointimenetelmät, joissa kohteen sijainti estimoidaan parametrisen mallin avulla.

Menetelmistä suodatus ja templaattisovitus kuuluvat joukkoon **a)** ja stokastiset menetelmät joukkoon **b)**. Seuraavassa kappaleessa esittelemme menetelmiä yksityiskohtaisemmin. Menetelmien esittelyn jälkeen luvuissa 3 ja 4 tutkimme tarkkuutta, johon menetelmillä on mahdollista päästä. Tarkasteluihin kuuluu sekä menetelmien teoreettinen että empiirinen analyysi. Tällä tavoin pyrimme luomaan lähtökohdat käytettävän menetelmän valitsemiseen siten, että ne toimivat vaihtelevissa olosuhteissa.

Luvun aiheena on siis kohteen, kuten reunan sijainnin, mittaaminen alipikselin tarkkuudella diskreetistä kuvasta. Lähtökohtana on yksi kuva $I_x = \{F_i\}_{i=1}^{n_p}$ ilman toistomittauksia.

Tavoitteena on määrätä mahdollisimman tarkasti reaalimuotoisen (ei-diskreetin) kohteen $r(z)$ sijainti \hat{X}_0 kuvan spatiaalisessa akselistossa $x \in \mathbb{R}$, kun kuva on mitattu akseliston diskreeteissä pisteissä $x_i \in \mathbb{R}$,

$$x_i = x_a + i \cdot \Delta p, \quad i = 1, \dots, n_p.$$

Tällöin on siis löydettävä reaaliarvoinen sijainti $\hat{X}_0 \in \mathbb{R}$ reaaliarvoiselle jatkuvamuotoiselle kohteelle $r(z) \in \mathbb{R}$, $z \in \mathbb{R}$ diskreetin aineiston I_x kautta.

Alipikselitason mittatarkkuuden voi saavuttaa periaatteessa kolmella tavalla:

1. Kohteen $r(z)$ sovituksella diskreettiin aineistoon, minkä voi tulkita estimointitehtäväksi. Tällöin kohde parametrisoidaan lokaatioparametrin x_0 kautta muotoon $r(x|x_0)$ ja kuvapisteen $\{F_i\}_{i=1}^{n_p}$ ymmärretään kohinaiseksi havainnoiksi kohteesta $r(x|x_0)$. Sovitus voidaan siten esittää sopivan optimointitehtävän kautta minimoimalla erotusta

$$(r(x_i|x_0) - F_i), \quad i = 1, \dots, n_p,$$

reaaliarvoisen sijainnin x_0 suhteen. Ajatusmalli soveltuu kaikkiin perinteisiin estimointimenetelmiin kuten pienimmän neliösumman estimointiin, uskottavuusfunktion käyttöön tai momenttien kautta tehtävään estimointiin.

2. Kohteen $r(z)$ haulla jatkuva-arvoisesta kuvafunktiosta $\hat{F}(x)$, joka muodostetaan diskreetin kuvan $\{F_i\}_{i=1}^{n_p}$ pohjalta. Kuvauksen $\hat{F}(x)$ muodostaminen edellyttää funktionsovitusten menetelmän käyttöä, kuten interpolointia, splinesovitusta tai kantafunktioiden kautta tehtyä sovitusta (katso esim. [4, 8]). Varsinainen sijainnin haku voidaan tämän jälkeen esittää minimointitehtävänä, esimerkiksi luvun 1.2 yhtälön (1) mukaisesti

$$\min_{x_0} \int_{x_a}^{x_b} \|\hat{F}(x) - r(x - x_0)\| dx,$$

mikä on toteutettavissa käyttäen jotain tehtävään soveltuvaa optimointimenetelmää (katso esim. [8]).

3. Diskreettinä hakuna interpoloidusta aineistosta $\{\hat{F}_i\}_{i=1}^{n_{sim}}$, jossa alkuperäisestä kuvasta $\{F_i\}_{i=1}^{n_p}$ on tehty tarkempi versio esimerkiksi interpoloinnin avulla. Jotta menetelmällä päästään alipikselin tarkkuuteen, on uuden aineiston pikselien lukumäärän n_{sim} on oltava huomattavasti suurempi kuin alkuperäisen aineiston. Tyypillisesti $n_{sim} \geq (10 \dots 100) \times n_p$, eli interpoloidun aineiston väli Δsim on $10 \dots 100$ kertaa pienempi, kuin kuvan pikseliväli Δp . Tämän menetelmän etuna on, että siinä voidaan

käyttää monia tunnettuja menetelmiä. Konenäön kirjallisuudessa [4], [7], [15], [24] on näet runsaasti lähdemateriaalia vastaaville sijainnin tunnistusongelmille, joissa tunnistus tehdään pikselin tarkkuudella. Siis löydetty sijainti \hat{X}_0 on jokin diskretisointipisteistä $\{x_i\}_{i=1}^{n_{sim}}$.

Tämän työn yhteydessä keskitymme pääosin diskreettiin esitystapaan 3) ongelman diskreetistä luonteesta johtuen. Muut vaihtoehdot sivuutamme lyhyellä kirjallisuuskatsauksella. Jaottelusta huolimatta on tärkeä huomata, että vaihtoehdot 1, 2 ja 3 eivät ole täysin poissulkevia. Valitsemalla tapaan 1 sopivan momenttipohjaisen menetelmän voimme tulkita tämän erääksi vaihtoehdon 2 toteutukseksi. Lisäksi tapa 3 voidaan useissa tapauksissa tulkita tavan 2 numeeriseksi toteutukseksi, etenkin jos uuden diskretisointivälin Δ_{sim} merkitys mittaustarkkuuteen on häviävän pieni.

2.1 Lyhyt kirjallisuuskatsaus sijainnin estimointiin diskreetistä aineistosta ilman interpolointia

Vaikka tämän työn kohteena ovatkin interpolointiin perustuvat menetelmät, on hyvä luoda lyhyt katsaus myös muihin vaihtoehtoihin. Alan kirjallisuudessa on yllättävän vähän mainintoja perinteisten estimointimenetelmien soveltamisesta kohteen sijainnin mittaamiseen alipikselin tarkkuudella. Syynä tähän saattaa olla tehtävän luonne, jossa kohdetta $r(z)$, $z \in [z^a, z^b]$ etsitään laajemmasta aineistosta $\{F_i\}_{i=1}^{n_p}$, $x_i \in [x^a, x^b]$, eli $x^b - x^a \geq z^b - z^a$. Perinteisessä estimoinnissa tilanne on usein päinvastainen, eli sovitettava malli (kohde) $g(x|\theta)$ on usein määritelty välillä $x \in [-\infty, \infty]$ ja aineisto sijoittuu tälle välille. Tämän vuoksi kirjallisuudessa esitetyt menetelmät rajaavatkin tunnistuksen usein vain sellaiseen osaan reaalimaailman ilmiötä $f(x)$, jossa data $\{F_i\}_{i=1}^{n_p}$ on sovitettavissa osaksi kohdetta $r(z)$. Havaintoaineiston rajaaminen puolestaan voidaan tehdä jollain karkeammalla menetelmällä.

Eräs varhaisimmista reunantunnistusmenetelmistä, johon suuri osa kirjallisuudessa esitetyistä menetelmistä perustuu, on Hueckelin [8] vuonna 1971 esittämä momenttiopeeraattori. Kyseinen työ on merkittävä myös siltä osin, että vaikka reunan sijainnille ei ole kvantitatiivisesti ja universaalisti hyväksyttyä määritelmää, on Hueckelin työssä esitettävä määritelmä hyvin lähellä sellaista [27]. Hueckel määrittelee reunan käyttäen termiä

”reuna-aineisto”, jolla hän kuvaa lineaarista rajaa kahden alueen välillä. Lyhyt leikkaus reunasta on ”reunaelementti” ja ”ideaalinen reunaelementti” viittaa tarkkaan kynnsfunktioon, jossa ei ole muun muassa kohinaa.

Momenttimenetelmän perusajatuksena on, että kohteesta $r(z)$ laaditaan sijainnin x_0 suhteen parametrisoitu approksimaatio ortogonaaleilla momenttifunktioilla. Samat momentit lasketaan myös havaintoaineistolle $\{F_i\}$ suhteessa tunnettuun pisteeseen (esimerkiksi pikseli 0) ja sovitus tehdään minimoimalla momenttiapproksimaatioiden ero lokalisaation x_0 suhteen.

Momenttiapproksimaatio funktiolle $f(x)$ voidaan kirjoittaa muodossa

$$f(x) \approx \sum_k \alpha_k \Phi_k(x),$$

jossa $\Phi_k(x)$ ovat ortogonaaleja kantafunktioita (momenttifunktioita) siten, että

$$\int \Phi_k(x) \cdot \Phi_l(x) dx = 0, \quad \text{kun } k \neq l$$

ja α_k ovat momenttien painoarvoja, jotka jatkuvalle funktiolle $f(x)$ voidaan määrätä integraalilla

$$\alpha_k = \int \Phi_k(x) f(x) dx.$$

Käytännölliseksi momenttimenetelmän tekee muun muassa se, että momenttien painotukset α_k on helppo laskea diskreetistä aineistosta $\{f(x_i)\}_{i=1}^n$ muodossa

$$\alpha_k = \sum_i \Phi_k(x_i) f(x_i),$$

mitä reunantunnistimien yhteydessä sovelletaan kuvadataan $I = \{F_i\}_{i=1}^{n_p}$.

Ongelmaksi jää kohteen momenttiapproksimaatio, joka on parametrisoitu sijainnin x_0 suhteen. Silloin on löydettävä tapa laskea kohteen $r(z)$ approksimaatio, eli integraali suljetussa muodossa x_0 :n funktiona

$$\beta_k(x_0) = \int r(x - x_0) \Phi_k(x) dx.$$

Tämä onnistuu helposti vain tietyille kohteen muodoille $r(z)$ ja sopivasti valitulle momenttifunktionalle $\Phi_k(x)$. Ongelma vaikeutuu lisäksi kaksiulotteisessa tapauksessa, kun integraaleista tulee kahden muuttujan funktioita.

Olettaen, että sopivat approksimaatiot ovat löydettävissä, voidaan reunan sijainti määrittää korvaamalla minimointitehtävä

$$\min_{x_0} \int (r(x - x_0) - \hat{F}(x))^2 dx \quad (5)$$

approksimointitehtävällä

$$\min_{x_0} \int [(\sum_k \beta_k(x_0) - \alpha_k) \Phi_k(x)]^2 dx,$$

mikä sopivien momenttifunktioiden tapauksessa on derivoitavissa x_0 :n suhteen ja kirjoitettavissa muotoon, josta x_0 saadaan ratkaistua.

Hueckelin reunantunnistimessa [8] kaksiulotteinen integraali rajataan R -säteiseen ympyrään ja momentteina käytetään yhdistelmää radiaali ja (x, y) koordinaateista, esimerkiksi

$$\Phi_1(x, y) = H_0(x, y) = \frac{5}{6} \pi Q(r)(1 - 2r^2)$$

ja

$$\Phi_7(x, y) = H_6(x, y) = \frac{32}{3\pi} Q(r) [(\frac{x - x_0}{R})^2 - \frac{y - y_0}{R}]^2,$$

missä $r = \frac{1}{R} \sqrt{(x - x_0)^2 - (y - y_0)^2}$ ja $Q(r) = \sqrt{1 - r^2}$.

Hueckelin operaattorin tavoitteena on nopea ja luotettava reunantunnistin. Se perustuu mallin sovittamiseen aineistoon kantafunktioiden avulla Hilbertin avaruudessa. Reuna estimoidaan pisteeseen, jossa malli sopii parhaiten tarkasteltavaan aineistoon. Operaattori tuottaa reunan sijainnin lisäksi informaatiota reunan korkeudesta ja suunnasta. Kokeellisten tulosten perusteella Hueckelin operaattori [8] toimii hyvin kohinaisessa ja karkeapintaisessa kuvassa. Ongelmana Hueckelin menetelmässä on sen kompleksisuus ja laskennallinen raskaus. Kompleksisuuden vuoksi tulosten teoreettinen analyysi on vaikeaa.

Eräs toinen momenttipohjainen menetelmä on Tabatabain ja Mitchellin [27] kehittämä momentteihin perustuva reunantunnistin. Tässä menetelmässä operaattori havaitsee reumat sovittamalla kolmea ensimmäistä harmaasävyn momenttia aineistoon. Se pystyy havaitsemaan reumat alipikselintarkkuudella sekä yksi- että kaksiulotteisessa tapauksessa ilman interpolointia tai iterointia. Etuna tälle operaattorille on se, että se on vähemmän herkkä kohinalle kuin Hueckelin operaattori. Vaikka operaattori kehitettiin havaitsemaan reunoja, pystyy se havaitsemaan myös muita muotoja.

Lyvers, Mitchell, Akey ja Reeves [12] rakensivat kehittyneemmän version Tabatabain

ja Mitchellin kolmeen ensimmäiseen harmaasävymomenttiin perustuvasta operaattorista. Operaattorin toiminta arvioidaan alipikselintarkkuudella kohinaisessa tapauksessa. Se käyttää reunan sijainnin määrittämiseen diskreetin datan spatiaalisia momenteja, joiden taustalla olevan ideaalin jatkuvan reunafunktion momentit ja reunan sijainti on määritelty. Sijainniksi saadut tulokset on sen jälkeen modifioitu korjaamaan näytteenotossa syntyviä deterministisiä virheitä. Operaattori pystyi havaitsemaan reunan suhteellisen pienillä harhalla ja hajonnalla.

Toisentyyppinen esiteltävä stokastisen reunantunnistuksen menetelmä perustuu suoraan kustannusten minimointiin, esimerkiksi pienimmän neliösumman estimointimenetelmään. Näissä menetelmissä sijainti \hat{X}_0 lasketaan suoraan yhtälöstä (5) ilman momenttiapproksimaatiota. Koska funktiosta $\hat{F}(x)$ on olemassa vain diskreettejä mittauksia $\hat{I} = \{\hat{F}(x_i)\}_{i=1}^{n_p}$, kirjoitetaan yhtälö esimerkiksi muotoon

$$\min_{x_0} \sum_i (r(x_i - x_0) - \hat{F}(x_i))^2.$$

Esimerkkinä tämänkaltaisista menetelmistä on Yen, Fun ja Poudelin [32] ehdottama algoritmi. Muita pienimmän neliösumman menetelmiä edustavat Nalwan ja Binfordin [13] sekä Kisworon, Venkateshin ja Westin [10] menetelmät. Tämän tyyppin menetelmät perustuvat siis parametrien sovitukseen, jossa minimoidaan todellisen kuvadatan ja mallin välinen virhe. Yleisesti ottaen pienimmän neliösumman menetelmien etuna on niiden yksinkertaisuus ja suora yhteys tilastolliseen estimointiin. Menetelmistä on momenttiapproksimaatioita helpompi kehittää muun muassa robusteja versioita korvaamalla pienimmän neliösumman minimointi jollain robustimmalla etäisyysmitalla. Tosin tällöin optimin haku voi olla haastavampaa.

Yen, Fun ja Poudelin työssä kohteena on niin kutsuttu gaussisesti usvainen eli kohinainen kynnsyreuna, joka on muodoltaan epälineaarinen. Tällöin sovitukseen tarvitaan korkean asteen polynomi, jolla määritetään reunan muoto. Menetelmän virhefunktiossa polynomimallia verrataan havaittuun aineistoon. Mallin parametrit, joista osa edustaa lokaatiota ja osa muotoa, optimoidaan parhaan sovituksen löytämiseksi. Menetelmä ei ole aivan ongelmaton. Mitä korkeamman asteen polynomi on käytössä, sitä enemmän tehtävässä on tuntemattomia parametreja, mikä vaikeuttaa ratkaisun löytämistä. Moniulotteinen optimisointiongelma joudutaan ratkaisemaan käyttämällä jotain numeerista etsintäalgoritmia. Käytettävän algoritmin tulisi olla laskennallisesti tehokas. Ye, Fu ja

Poudel käyttävät työssään erästä Newtonin menetelmän versiota, joka minimoi approksimaation käyttämällä numeerista derivointia.

2.2 Interpolointiin perustuvat menetelmät

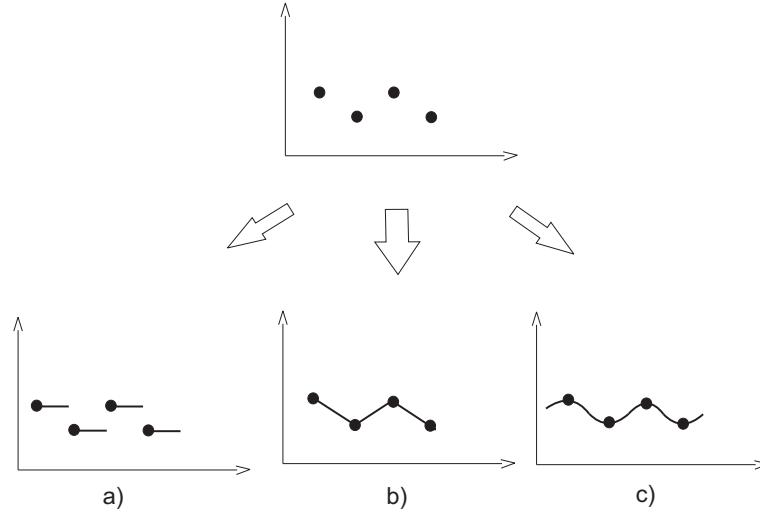
Kuvan pikselien määrää voidaan kasvattaa interpoloimalla, mikä mahdollistaa perinteisten reunantunnistimien kuten derivaattoihin tai templaattisovitukseen perustuvien pikselimaskien hyödyntämiseen alipikselin tarkkuudella. Näissä menetelmissä on kaksi osatehtävää:

1. Interpolointimenetelmä, joka pystyy diskreetin aineiston pohjalta jäljittelemään mahdollisimman hyvin todellisen taustalla olevan ilmiön $f(x)$ käyttäytymistä.
2. Konvoluutioon perustuva suodatin, joka pyrkii havaitsemaan kohteen sijainnin interpoloidusta aineistosta mahdollisimman tarkasti ja virheettömästi.

Näiden vaiheiden toteutukset voidaan valita toisistaan riippumatta. Aloitamme tarkastellun interpolointimenetelmistä ja palaamme konvoluutioon perustuviin suodattimiin osaluvuissa 2.3 ja 2.4

2.2.1 Interpolointi

Interpoloinnissa tyypillisesti muodostetaan sovite $\hat{F}(x)$ diskreetin aineiston perusteella väleille $[x_i, x_{i+1})$, $i = 1, \dots, (n_p - 1)$. Interpolointimenetelmiksi on useita vaihtoehtoja, joihin kuuluu lähimmän naapurin menetelmä. Yksinkertaisimmillaan lähimmän naapurin menetelmässä sovite on vakio ja saa arvonsa joko lähimmästä (edeltävästä tai seuraavasta) tai aina edeltävästä diskretisointipisteestä, kun taas monimutkaisimmillaan menetelmä kerää informaatiota useista pisteistä ja sovittaa niiden perusteella epälineaarisen käyrän. Kuvassa 6 on havainnollistettu interpolointimenetelmän valinnan vaikutusta sen tuottamaan tulokseen. Menetelmän toimivuus riippuu havaintopisteiden määrästä ja niiden informatiivisuudesta sekä siitä, kuinka hyvin menetelmä pystyy ennustamaan funktion todellista käyttäytymistä.



Kuva 6: Havaitusta aineistosta saatu interpoloitu sovite voi vaihdella huomattavasti riippuen käytettävästä interpolointimenetelmästä: a) vakio, b) lineaarinen, c) epälineaarinen.

Tässä yhteydessä sovellamme interpolointia kohinaiseen kuvaan $I = \{f(x_i) + \varepsilon_i\}_{i=1}^{n_p}$. Koska tarkoituksena on arvioida mahdollisimman hyvin ilmiön $f(x)$ käyttäytymistä tunnettujen pisteiden, esim. x_i ja x_{i+1} , välillä, voimme harkita kahta erilaista vaihtoehtoa interpoloinnin suorittamiseksi:

- Eksaktia interpolointia, jossa interpoloitu sovite $\hat{F}(x)$ kulkee kaikkien tunnettujen pisteiden $\{F(x_i)\}_{i=1}^{n_p}$ kautta.
- Regressiosovitetta, jossa sovituksen edellytetään olevan sileä (regularisoitu) ja jossa sovituksen ei välttämättä tarvitse kulkea havaintopisteiden kautta. (Sen on aina seurattava havaintopistemassaa!)

Kuten edellä on kerrottu, interpolointi tehdään lähes poikkeuksetta jonkin funktiosovituksen eli interpolaatiofunktion avulla. Tällöin on luonnollista arvioida interpolointimenetelmän ominaisuuksia käytettävän interpolaatiofunktion näkökulmasta kuvan 6 tapaan. Yksinkertaisin on paloittain vakio (kuva 6a) eli muotoa

$$\hat{F}(x) = F(x_i), \quad \text{kun } x_i \leq x < x_{i+1}$$

oleva interpolaatiofunktio. Linearisessa interpoloinnissa pistejoukko yhdistetään paloittain lineaarisella funktiolla (kuva 6b)

$$\hat{F}(x) = F(x_i) + \frac{[F(x_{i+1}) - F(x_i)][x - x_i]}{x_{i+1} - x_i}, \quad x_i \leq x \leq x_{i+1}. \quad (6)$$

Epälineaarista interpolointimenetelmistä tunnetuimpia ovat polynomisovitukset, joissa haetaan parametrit a_0, a_1, \dots, a_p siten, että p :nnen asteen polynomi (kuva 6c))

$$\hat{F}(x) = \sum_{i=0}^p a_i x^i, \quad x_i \leq x \leq x_{i+1},$$

saadaan sovitettua havaintodataan $\{(F(x_i), x_i)\}_{i=1}^{n_p}$. Polynomisovitteesta on hyvä huomata, että $p + 1$:n parametrin sovittaminen vaatii vähintään $p + 1$ havaintopistettä, joten kun $p \geq 1$, käyttää menetelmä välin $[x_i, x_{i+1})$ approksimaation määrittämiseen useampia pisteitä, esim. pisteitä $x_{i-p}, \dots, x_i, \dots, x_{i+p+1}$. Lisäksi lineaarinen vakio- ja lineaarinen interpolaatio ovat tulkittavissa polynomi-interpolaatioksi astetta $p = 0$ (vakio) ja $p = 1$ (lineaarinen). Muita yleisesti käytettyjä menetelmiä ovat mm. spline-interpolaatiot, joissa havaintopisteet yhdistävä polynomikäyrä muodostetaan siten, että käyrän derivaatat (ainakin ensimmäinen derivaatta) ovat jatkuvia myös havaintopisteissä.

Tässä työssä sovelletaan pääosin eksaktia lineaarista interpolointia, jonka ominaisuuksia on suhteellisen helppo analysoida. Työn empiirisessä osuudessa sovelletaan myös spline-interpolointia ja spline-interpolointia regressiosovituksella.

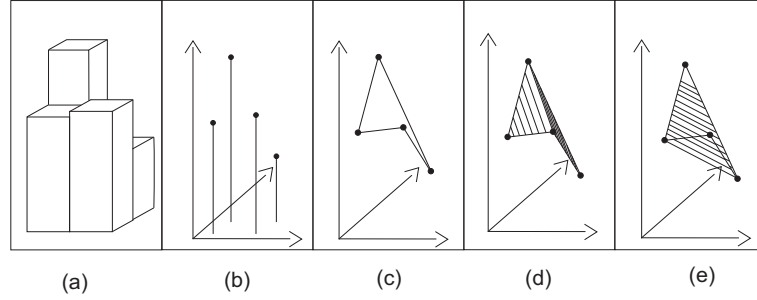
Interpolointi voidaan yleistää myös kahden tai useamman vastemuuttujan funktioille. Hankaluutena tosin on, että näin saatavat sovitteet ovat usein varsin monimutkaisia. Lisäksi ei ole yksikäsitteistä tapaa määrätä lineaarista sovitetta edes kahden muuttujan tapauksessa. Tätä on havainnollistettu kuvassa 7, missä tavoitteena on interpoloida 2×2 pisteen kuva lineaarisella interpolaatiolla. Koska lineaarinen interpolaatio kahden muuttujan funktiolle sovittuu kolmella pisteellä, on selvää, että neljän pisteen muodostamaa alaa ei voida interpoloida eksaktisti yhdellä lineaarisella tasolla, kun käytettävissä on funktion arvot pisteissä $(x_i, y_j), (x_{i+1}, y_j), (x_i, y_{j+1})$ ja (x_{i+1}, y_{j+1}) . Vaihtoehtoja on tällöin kaksi (kuvan kohdat d ja e):

- a) Tehdään interpolointi kahdella lineaarisella tasolla, jotka voidaan tehdä kahdella tavalla, joko pisteille (kuva 7d)

$$[(x_i, y_j), (x_{i+1}, y_j), (x_i, y_{j+1})] \text{ ja } [(x_{i+1}, y_j), (x_i, y_{j+1}), (x_{i+1}, y_{j+1})],$$

tai pisteille (kuva 7e)

$$[(x_{i+1}, y_j), (x_i, y_j), (x_{i+1}, y_{j+1})] \text{ ja } [(x_i, y_{j+1}), (x_i, y_j), (x_{i+1}, y_{j+1})].$$



Kuva 7: (a): Havaittu digitaalinen 2×2 kuva. (b) Intensiteetit diskretisointipisteissä $\{f_{ij}\}_{i,j=1}^2$, jotka ovat pikselien keskipisteet. (c) vierekkäiset diskretisointipisteet yhdistettyinä yksiulotteisella lineaarisella interpoloinnilla. (d) Lineaarinen taso suuntaan x . (e) Lineaarinen taso suuntaan y .

- b) Tehdään interpolointi lineaarisesti useana yksiulotteisena interpolointina sekä x -että y -akselin suuntaan kuten kuvassa 8, jolloin näiden funktioiden avulla esitettävä kaksiulotteinen pinta on epälineaarinen. Täsmällisesti esitettynä interpolaatio on tällöin muotoa

$$\hat{F}(x, y) = \hat{F}_X(x|y_i) + \frac{[\hat{F}(x|y_{i+1}) - \hat{F}(x|y_i)][y - y_i]}{y_{i+1} - y_i}, \quad (7)$$

missä

$$\hat{F}_X(x|y_k) = F(x_i, y_k) + \frac{[F(x_{i+1}, y_k) - F(x_i, y_k)](x - x_i)}{x_{i+1} - x_i}. \quad (8)$$

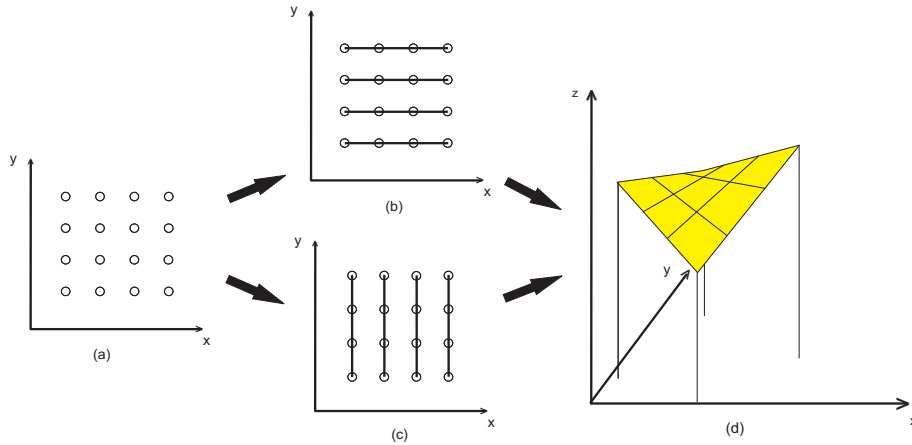
Tässä työssä kaksiulotteiset interpoloinnit on suoritettu jälkimmäisellä tavalla, mikä voidaan tulkita muotoa

$$F(x, y) = ax + by + cxy + d$$

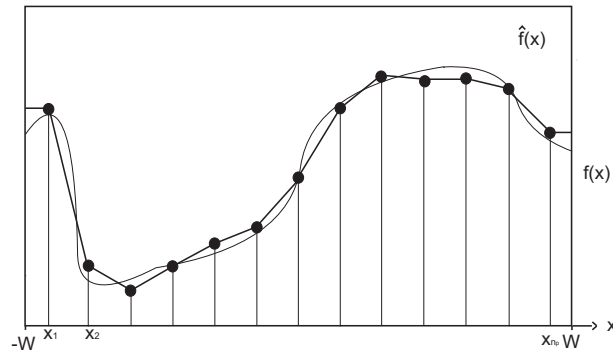
olevana polynomisovitteena, missä a, b, c ja d ovat havaintojen perusteella haettavia vakio kertoimia. Interpoloinnin periaatetta on vielä havainnollistettu kuvassa 8.

2.2.2 Interpolointi ja kohinainen aineisto

Tässä työssä interpolointimenetelmänä on pääosin lineaarinen interpolointi. Siinä sovitusta tehdään muodostamalla lineaarinen jono kahden peräkkäisen havaintopisteen välille. Eräs lineaarisen interpoloinnin toteutus on esitetty kuvassa 9. Aineistona kyseisessä tapauksessa on yksi kuva $I_x(t) = \{F_i(t)\}_{i=1}^{n_p}$ kohinaisesta ilmiöstä $F(x_i) = f(x) + \varepsilon_i$. Kuvasta 10 on nähtävissä, kuinka lineaarisesti interpoloitavien arvojen hajonta on suurimmillaan havain-



Kuva 8: (a) Interpoloitava kaksiulotteinen aineisto. (b) Yksiulotteisesti suuntaan x lineaarista interpolointia käyttäen. (c) Yksiulotteisesti suuntaan y lineaarista interpolointia käyttäen. (d) Kaksiulotteisesti interpoloitu aineisto.



Kuva 9: Esimerkki ilmiön $f(x)$ sovitteesta $[\hat{F}(x)](t)$, johon on päästy, kun aineistona on ollut diskreetti ja kohinainen kuva $I_x(t)$.

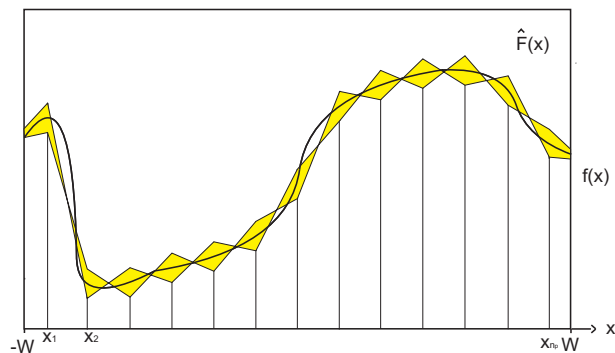
topisteissä, joissa menetelmä on kuitenkin harhaton, ja pienimmillään diskretisointipisteiden välin keskipisteessä, jossa menetelmä kuitenkin on mahdollisesti harhainen. Koska lineaarinen interpolointi on suorien rakentamista havaintopisteiden välille, on menetelmän käyttö perusteltua silloin, kun tiedetään, ettei reunafunktio $r(z)$ käyttäydy voimakkaan epälineaarisesti, jolloin tuloksena olisi harhainen sovite, kuten kuvasta 10 on havaittavissa.

Yleisesti tiedämme, että lineaarisen interpoloinnin tekemä virhe kohinattomalle aineistolle on luokkaa

$$|\hat{f}(x) - f(x)| \leq \frac{1}{8} \max_{x' \in [x_i, x_{i+1}]} [f''(x')] \cdot (x_{i+1} - x_i)^2,$$

eli se maksimoituu neliöllisesti etäisyytenä diskretisointipisteisiin ja on suhteessa kohdefunktion $f(x)$ toiseen derivaattaan, joka kuvaa funktion kaareutumista.

Kohinan tekemä virhe taas on suurimmillaan diskretisointipisteissä, missä se seuraa mit-



Kuva 10: Mahdollinen 95%:n luottamusväli $\hat{F}(x)$:lle lineaarisessa interpoloinnissa, kohinalla ε .

tapisteiden kohinaa ε , mutta niissä sen vaikutusta voidaan vähentää usean kuvan otolla. Mittavälillä lineaarisella interpoloinnilla on silottava vaikutus.

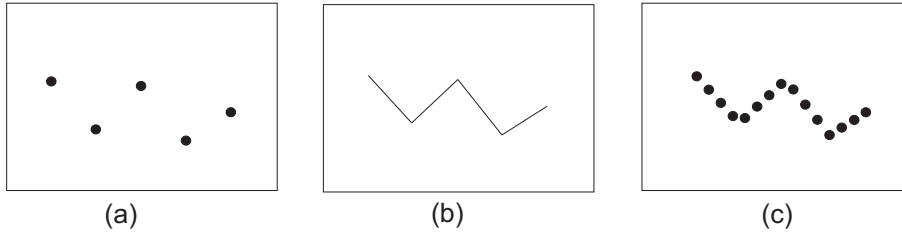
2.2.3 Resoluution kasvattaminen interpoloinnilla

Loppukommenttina interpoloinnista on syytä mainita, että tämän työn yhteydessä interpolointia käytetään kuvan resoluution kasvattamiseen. Ajatuksena on korvata aiempi aineisto, jonka pikseliväli on $\Delta p = x_{i+1} - x_i$, uudella tarkemmalla aineistolla, jonka pikseliväli Δsim on huomattavasti pienempi. Tällöin uuden aineiston pikselit ovat siis alkuperäisen aineiston alipikseleitä (pikselin osia). Formaalisimmin ilmaistuna havaitusta aineistosta kokoa n_p muodostetaan interpoloinnin avulla n_{sim} kokoinen aineisto $\hat{I}_x(t) = \{\hat{F}_i(t)\}_{i=1}^{n_{sim}}$, jossa

$$\hat{F}_i(t) = [\hat{F}(x_a + (i - 1) \cdot \Delta sim)](t), \quad i = 1, \dots, n_{sim},$$

missä x_a on kiinteä diskretisoinnin ensimmäinen piste x -akselilla, n_{sim} interpoloidun aineiston pisteiden lukumäärä ja Δsim kahden interpolointipisteen välinen etäisyys. Työssä pyritään siihen että interpolointiväli Δsim on sellainen, että siitä koitua virhe reunan havaittuun sijaintiin olisi mitätön verrattuna muihin virhelähteisiin.

Tätä menettelyä on pyritty havainnollistamaan kuvassa 11, jossa viidestä diskretisointipisteestä on lineaarisen interpoloinnin avulla muodostettu 15 pistettä, jolloin pisteiden väli on pienentynyt kolmasosaksi. Joissain interpolointimenetelmissä ei muodosteta jatkuva sovitetta, vaan sen tuloksena saadaan suoraan parannetulla resoluutiolla aineistoa estimoiva diskreetti sovite [4].



Kuva 11: (a) Ilmiöstä saatu diskreetti aineisto $\{F_i(t)\}_{i=1}^{n_p}$. (b) Aineistosta lineaarisesti interpoloimalla saatu jatkuva sovite $[\hat{F}(x)](t)$. (c) Jatkuvasta sovitteesta muodostettu diskreetti aineisto $\hat{I}_x(t)$, jonka koko n_{sim} määräytyy interpolointivälin Δ_{sim} mukaan.

2.3 Konvoluutio kohteen sijainnin mittaamisessa

Interpolointiin perustuvassa menetelmässä kohteen sijainti määritetään interpoloidun aineiston ja nk. tunnistussuotimen välisen konvoluution avulla. Menetelmä ja siihen liittyvä terminologia on pääosin lähtöisin signaalinkäsittelystä. Konvoluutio on kahden funktion välille määritelty matemaattinen operaatio. Konvoluutio saadaan funktioiden tulon integraalina, kun toinen funktioista on peilattu ja liu'utettu yli määrittelyalueen

$$(f * g)(x) = \int g(z)f(x - z)dz = \int g(x - z)f(z)dz.$$

Eräs tunnettu konvoluutiota käsittelevä teos on Bracewellin teos [1] vuodelta 1986. Siinä konvoluutio ja sen käyttömahdollisuuksia on esitelty perusteellisesti.

Signaalinkäsittelyn yhteydessä konvoluutiota kutsutaan myös suodattamiseksi, missä funktio $f(x)$ edustaa jotain reaali maailman ilmiötä ja $g(z)$ on nk. suodatin. Tässä yhteydessä $f(x)$ edustaa (jatkuva-arvoista) kuvaa ilmiöstä ja $g(z)$ on nk. kohteentunnistin, josta eri yhteyksissä käytetään myös nimityksiä maski tai templaatti. Suodatin $g(z)$ voi olla joko suoraan tutkittavan ilmiön muotoinen tai sillä on jokin muu muoto. Tavoitteena on, että konvoluutio antaa voimakkaan vasteen, kun suodattimen $g(z)$ kohdistuspiste osuu siihen kohtaan, x_0^* , ilmiötä $f(x)$, jonka ympäristössä tutkittava kohde sijaitsee.

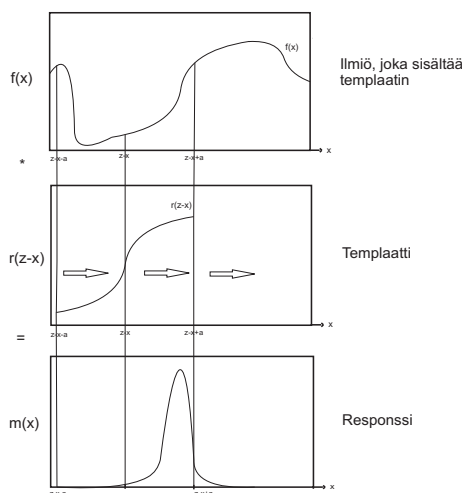
Terminologian selkiyttämiseksi kutsumme tutkittavan kohteen muotoa esittävää suodatinta templaatiksi. Toinen vaihtoehto on nk. reunasuodatin, eli reunamaski, joka reagoi vahvasti reunan kohdalla tapahtuviin kuvan intensiteetin muutoksiin. Reunamaskien etuna on yleiskäyttöisyys, sillä niiden oletetaan soveltuvan laajalle joukolle erityyppisiä reuna-funktioita. Toisaalta templaatti antaa usein reunamaskia vahvemman vasteen eli responsin tunnistuspisteessä x_0^* . Käsittelemme reunantunnistimien ja niihin liittyvien maskien

teoriaa tarkemmin aliluvussa 2.4.

Työssämme konvoluutio tulkitaan mitaksi $m(x)$, joka kertoo kuinka hyvin piste x edustaa kohteen sijaintia edustavaa pistettä x_0^* . Tätä on havainnollistettu kuvassa 12. Kuvassa on esitetty tilanne, jossa kohdetta esittävä templaatti on konvoloitu ilmiön kanssa. Huomataan, että kohdassa, jossa funktiot sopivat parhaiten toisiinsa, myös konvoluution responsi $m(x)$ on suurimmillaan. Tästä pääsemmekin konvoluution korrelaatio-ominaisuuteen. Korrelaatio $f \circ g$ on muotoa

$$(f \circ r)(x) = \int g(z)f(x+z). \quad (9)$$

Konvoluutio ja korrelaatio ovat käytännössä sama asia, mutta konvoluutiossa suodatin



Kuva 12: Korrelaatio $m(x) = \int_{-a}^a f(z+x)r(z)dz$, jossa funktio $r(z)$ on liu'utettu todellisen ilmiön $f(x)$ yli.

$g(z)$ on esitettävä kohdituspisteensä suhteen peilattuna. Korrelaatio funktiosta $g(z)$ on funktion $g(-z)$ konvoluutio, mikä nähdään seuraavasta kaavasta

$$\int g(z)f(x+z) dz = \int g(-z)f(x-z)dz. \quad (10)$$

Korrelaation siis voidaan ajatella olevan konvoluutio, joka mittaa miten hyvin pisteeseen x sijoitettu funktio $g(z)$ vastaa funktiota $f(x)$.

Mittaa $m(x)$ voi siten käyttää kohteen $r(z)$ sijainnin hakemiseen kuvasta $f(x)$ siten, että löydetty sijainti \hat{x}_0 on vasteen maksimissa

$$\hat{x}_0 = \arg \max_x m(x).$$

Konvoluution ja korrelaation ominaisuuksiin kuuluvat

1. assosiativisuus $(h * f) * g = h * (f * g)$
2. distributiivisuus $h * (f + g) = h * f + h * g$
3. kommutatiivisuus $f * g = g * f$
4. skalaarimonikerta $a(f * g) = (af) * g = f * (ag), \quad \forall a \in \mathbb{R}$

Näiden lisäksi eräs konvoluution käytännöllinen ominaisuus liittyy sen derivointiin. Laskehtaessa konvoluution derivaattaa riittää, että vain toinen funktioista derivoidaan. Konvoluution derivaatta saadaan, kun saatu funktion derivaatta konvoloidaan toisen funktion kanssa, eli

$$\nabla(f * g) = \nabla f * g = f * \nabla g.$$

Tätä ominaisuutta tulemme hyödyntämään myöhemmin.

Edellisten ominaisuuksien lisäksi konvoluutioon liittyy nk. konvoluutioteoreema [7], jonka mukaan konvoluutio voidaan tehdä Fourier-avaruudessa funktioiden $f(x)$ ja $g(z)$ taajuuksien tulona. Tarkemmin ilmaistuna, jos $\mathbf{F}(u)$ on $f(x)$:n Fourier-muunnos ja $\mathbf{G}(u)$ on $g(z)$:n Fourier-muunnos, voidaan konvoluutio ratkaista muodossa

$$\int g(z)f(x - z)dz = \mathfrak{F}^{-1}(\mathbf{G}(u) \cdot \mathbf{F}(u)),$$

missä \mathfrak{F}^{-1} on käänteinen Fourier-muunnos. Tämä osoittautuu hyödylliseksi konvoluution käytännön toteutuksissa.

Reaalimaailman aineiston ollessa kaksiulotteinen digitaalinen kuva, ei jatkuvaa konvoluutiota voida käyttää sellaisenaan. Tämän vuoksi tarvitsemme käyttöön diskreetin konvoluution. Diskreetti konvoluutio on muotoa

$$m(x_i) = \sum_k f_{k+i} \cdot g_k.$$

Tässä f_k ja g_k ovat siis diskreetissä tasavälisessä joukossa määritellyjä funktion arvoja, kuten esimerkiksi kuvamme pisteet

$$F_i = F(x_i) = F(x_a + \Delta p \cdot (i - 1)), \quad i = 1, \dots, n_p.$$

Kuvaan kohdistettuna suodatin on diskreetin konvoluution tapauksessa myös diskretisoituva, eli kirjoitettava muotoon

$$g_k = g(z_k) = g(z_a + \Delta p \cdot (k - 1)), \quad i = 1, \dots, n_g$$

missä n_g on suodattimessa käytettävien kuvapisteen määrä.

Diskreetin konvoluution antama vaste $m(x_i)$ vastaa diskreetin suodattimen $g(z_k)$, $k = 1, \dots, n_g$ ensimmäistä pistettä z_1 . Jos kohdistuspisteeksi valitaan jokin muu, esimerkiksi suodattimen piste k^0 , tulee vaste kirjoittaa muotoon

$$m(x_i) = \sum_{k=1}^{n_g} g_k f_{k+1-k^0}, \quad k^0 \leq i \leq n_p - k^0.$$

Konvoluutio voidaan yleistää myös kaksiuotteisille funktioille $f(x, y)$ ja $g(z, q)$, missä jatkuva-arvoinen konvoluutio on muotoa

$$m(x, y) = \int \int f(x, y) g(z - x, q - y) dz dq,$$

ja sen diskreetti toteutus on muotoa

$$m(x_i, y_j) = \sum_k \sum_l f_{k+i, l+j} \cdot g_{k, l}. \quad (11)$$

Työkaluina työssä käytämme ainoastaan diskreettejä toteutuksia konvoluutiosta, jolloin konvoluution avulla löydetty piste (x_0, y_0) määrätään vasteen $m(x_i, y_j)$ maksimista

$$(x_0, y_0) = \arg \max_{(x_i, y_j)} m(x_i, y_j).$$

Kuten yksiulotteisessa tapauksessa, vastaa yhtälön (11) mukaisesti laskettu konvoluution maksimi diskretisoidun suodattimen $g_{k, l}$ pistettä $k = 1, l = 1$. Jos kohdistuspiste on esim. piste k^0, l^0 , on konvoluutio laskettava muodossa

$$m(x_i, y_j) = \sum_k \sum_l f_{k+i-k^0, l+j-l^0} \cdot g_{k, l}.$$

2.4 Reunansuodattimen ja templaattien muodostaminen

Esimerkkejä interpoloinnin käytöstä alipikselintarkkuuden reunan sijainnin mittaamiseen ovat esittäneet muunmuassa Steger [26], Devernay [6] ja Rockett [20]. Nämä perustuvat jossain määrin perinteiseen reunantunnistukseen ja ainakin Devernayn ja Rockettin käyttämät menetelmät pohjautuvat Canny'n [2] reunantunnistukseen.

Perinteinen pikselitason reunantunnistus on derivaattojen approksimointia, sillä derivaatat mittaavat kuvan intensiteetin muutoksia, joilla perinteisesti määrittellään reunan sijainti [31]. Näissä menetelmissä reuna tunnistetaan derivaattaa approksimoivan suodattimen

avulla. Perinteisen reunantunnistuksen idea esitellään tarkemmin konenäön perusteoksissa [7, 15, 24].

Eräs yleisimmin käytössä oleva menetelmä reunansuodattimien muodostamiseksi perustuu J.F. Cannyn [2] luomien kriteerien optimointiin. Vaikka joissain yhteyksissä kerrotaan Cannyn luoneen yhden reunantunnistimen, ei tämä pidä paikkaansa. Todellisuudessa Canny loi metodin, jolla voidaan luoda useita reunantunnistimia. Aivan yleinen tämä metodi ei kuitenkaan ole, sillä Cannyn reunantunnistimet perustuvat ensimmäisen derivaatan eli gradientin approksimointiin. Koska derivointi korostaa kohinan vaikutusta, käsitellään aineisto gaussisella silotuksella. Cannyn menetelmää tarkasteltaessa tulee ottaa huomioon, ettei se sinällään pääse alipikselin tarkkuuteen. Tästä johtuen on ensin suoritettava aineiston interpolointi, jotta Cannyn menetelmällä päästään alipikselin tarkkuuteen.

Edellä esiteltyt kaksi reunantunnistusmenetelmää perustuvat kuvan intensiteetin eri asteisten derivaattojen approksimointiin. Nämä menetelmät ovat kuitenkin ongelmallisia kohteen sijainnin tunnistuksen kannalta, sillä kohde ei välttämättä ole sidoksissa intensiteetin monotoniseen muutokseen, vaan tämä kohde saattaa olla mikä tahansa funktion muodon avulla identifioitavissa oleva rakenne. Yleisesti ottaen emme pyri havaitsemaan kuvassa olevia intensiteetin muutoksia, vaan tarkoituksenamme on erilaisten intensiteetifunktiossa sijaitsevien, ennalta määriteltyjen osajoukkojen eli kohteiden tunnistaminen. Templaattisovituksen määritelmä (template matching, matching) sopiikin huomattavasti paremmin kohteen sijainnin tunnistamiseen kuin reunantunnistusmenetelmät. Tämä on perinteinen menetelmä, jonka toteutus löytyy konenäön perusteoksista [7, 15, 24].

Templaattisovituksessa käytettävät työvälineet ovat käytännössä samat kuin reunan suodatuksessa. Ero templaattisovituksen ja reunan suodatuksen välillä perustuukin siihen, että templaattisovitus ei liity kuvan differentiointiin, vaan siihen, kuinka hyvin tavoittepinta sopii kuvan tarkasteltavaan kohtaan. Tämän vuoksi sovitettavasta templaatista eli reunafunktiosta ja tarkasteltavasta kuvasta mitataan korrelaatio. Templaattisovitusta käsitellään lyhyesti luvussa 2.4.3

2.4.1 Canny'n ja Marr-Hildreth'in suodattimien muodostaminen

Sekä Canny'n, että Marr-Hildreth'in suodattimet perustuvat luvussa 2.4 esitettyyn konvoluution differentiointiominaisuuteen, jonka mukaan

$$\nabla(g * f) = \nabla g * f \quad \text{ja} \quad (12)$$

$$\nabla^2(g * f) = \nabla^2 g * f. \quad (13)$$

Tässä ∇ on ensimmäisen asteen derivaatat määräävä gradienttioperaattori ja ∇^2 on toisen asteen derivaatat määräävä Laplace-operaattori. Ideana on, että differentiaalioperaattorit ∇ ja ∇^2 havaitsevat intensiteetin muutoksia suodattimen g määräämällä leveydellä kuvasta f . Tällöin suodatin g toimii myös kohinan poistajana.

Canny'n reunantunnistimessa g on gaussinen funktio, jonka leveys optimoidaan. Tunnistin olettaa reunojen olevan kynnsfunktioita, joihin sisältyy normaalisti ja nollakeskeisesti jakautunutta kohinaa. Yksiulotteisessa tapauksessa suodatin, joka jatkuvassa joukossa optimoi Canny'n kriteerit, on Tagaren ja de Figueiredon [29] mukaan muotoa

$$\nabla g(z) = -ze^{\frac{-z^2}{2\sigma^2}}.$$

Tämä filtteri on siis nimeltään Derivative of Gaussian (DoG). Se on ensimmäisen derivaatan gaussinen filtteri, jossa σ on gaussisen osan keskihajonta. Kaksiulotteinen jatkuva toteutus filteristä z -akselin suuntaan on muotoa

$$\nabla g(z, q) = \frac{z}{2\pi\sigma^4} e^{\left\{-\frac{(z^2+q^2)}{2}\sigma^2\right\}}.$$

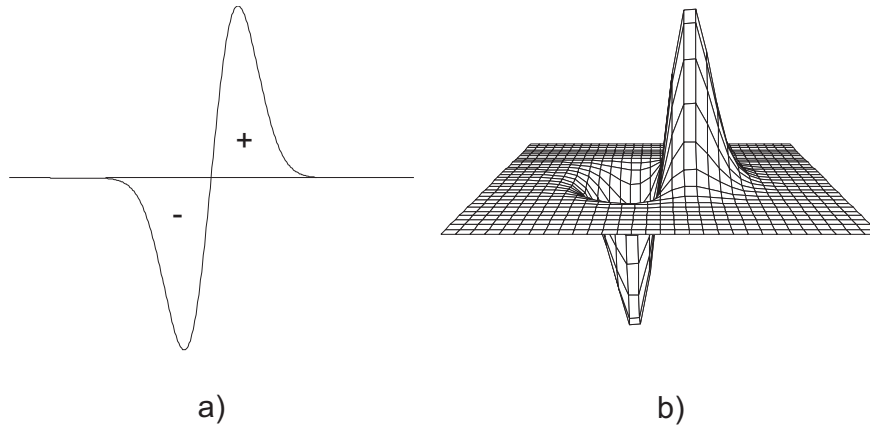
Hajontatermi σ määrää funktion skaalan, joten sen tulee olla sellainen, että se pystyy havaitsemaan reunan sijainnin, jota olemme etsimässä. Toinen asia, mikä tulee ottaa huomioon, on hajontatermin suhde filterin määrittelyalueeseen $[-a, a]$. Lisäksi suodattimien määrittelyalueen ylläskettujen integraalien tulee hävitä eli

$$\int_{-a}^a g(z) dz = 0$$

ja

$$\int_{-a}^a \int_{-a}^a g(z, q) dz dq = 0.$$

Derivative of Gaussianin muoto on esitetty kuvassa 13.



Kuva 13: Derivative of Gaussian: a) yksiulotteinen, b) kaksiulotteinen

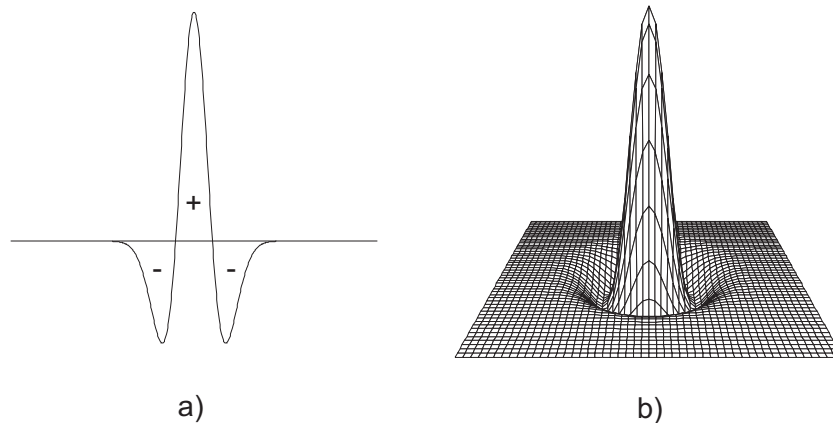
Marr-Hildrethin operaattorissa eli Laplacian of Gaussianissa (LoG) havaitaan kuvan toinen derivaatta gaussisen operaattorin g toisen differentiaalin ∇^2 ja ilmiön f konvoluutiosta. Kuten edellä on todettu, approksimoi Laplacian of Gaussian yhtälöä 13. Jatkuva yksiulotteinen Laplacian of Gaussian on muotoa

$$\nabla^2 g(z) = -\frac{z^2 - \sigma^2}{\sigma^4} e^{\{-\frac{z^2}{2\sigma^2}\}}$$

ja jatkuva kaksiulotteinen muotoa

$$\nabla^2 g(z, q) = \frac{1}{\pi\sigma^4} \left(1 - \frac{z^2 + q^2}{2\sigma^2}\right) e^{\{-\frac{z^2 + q^2}{2\sigma^2}\}},$$

joissa σ on filttarin gaussisen osan hajontatermi, kuten Derivative of Gaussianissa. Tätä on havainnollistettu kuvassa 14 Laplacian of Gaussiania kutsutaan usein ”Mexican hat”-



Kuva 14: Laplacian of Gaussian operaattori: a) yksiulotteinen, b) kaksiulotteinen

operaattoriksi [7] sen sombrero-maisen muodon vuoksi, kuten kuvasta 14 pystytään huomaamaan. Jos LoG on muodostettu kaikkiin suuntiin samalla tavoin jakautuneesta sym-

metrisestä Gaussin funktiosta, kuten Marr ja Hildreth tekivät, on operaattori suuntainvariantti, mikä tarkoittaa, että sitä käytettäessä ei tarvitse olla tietoa reunan suunnasta reunan sijainnin havaitsemiseksi. Tämä on etu verrattuna esimerkiksi Cannyn [2] operaattoriin, jossa reunan suunta tulee arvioida. Lisäksi nollakohtien havaitseminen eroaa paikallisen maksimin havaitsemisesta. Lisätietoa Laplacian of Gaussianin käyttäytymisestä ja nollakohtien havaitsemisesta löytyy Tabbonen, Alonson ja Zioun artikkelista [28].

2.4.2 Derivaattasuotimet diskreetille aineistolle

Kuten aiemmin on todettu, perinteinen reunantunnistus perustuu suurilta osin derivaatan approksimointiin. Diskreetisti derivaattaa voidaan approksimoida sitä mittaavilla maskeilla tai filtereillä. Derivaatan ollessa suurimmillaan, myös kontrasti, joka mittaa kahden peräkkäisen pikselin arvojen välistä etäisyyttä $|f(x_i) - f(x_{i-1})|$, on suurimmillaan. Tässä kappaleessa esittelemme, mihin tämäntyylinen reunantunnistus perustuu diskreettiä aineistoa käytettäessä.

Oletetaan, että on olemassa yksiulotteinen diskreetti ilmiö F_i , $i = 1, \dots, n_p$, jonka derivaatta on approksimaatio $F'_i \approx \frac{F_i - F_{i-1}}{x_i - x_{i-1}}$. Diskretisointivälin ollessa yksi, $\Delta p = 1$, derivaatta F'_i saadaan konvoluutiota käyttäen maskilla $\mathbf{M}' = [-1, 1]$. Olkoon aineisto on muotoa $I_x(t) = \{F_i(t)\}_{i=1}^{n_p}$. Tällöin kuvan vektorin $I_x(t)$ derivaattaa $I_x(t)'$ voidaan approksimoida konvoluutiolla

$$I_x(t)' \approx \mathbf{M}' * I_x(t) = \{F_i(t) - F_{i-1}(t)\}_{i=2}^{n_p}.$$

Toisaalta voidaan ajatella, että tämä approksimaatio tapahtuu pisteessä $\frac{x_i - x_{i-1}}{2}$, kuten kuvasta 15 on havaittavissa.

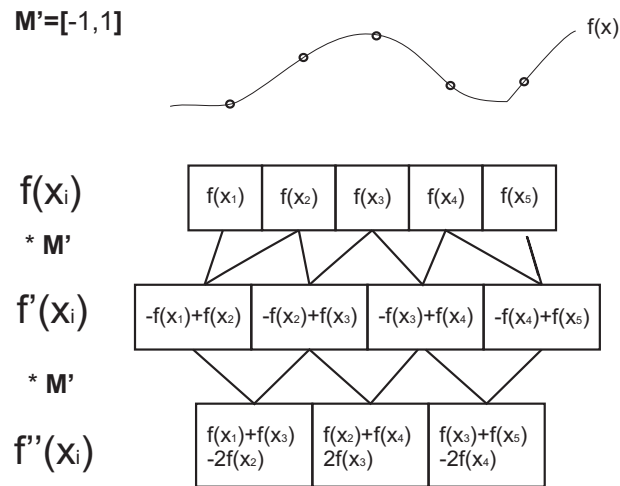
Jos tarvitaan myös toisen asteen derivaattaa, päästään siihen konvolvoimalla saadun ensimmäinen derivaatan vektori $I_x(t)'$ ensimmäisen derivaatan maskilla \mathbf{M}' . Tästä johtuen toinen derivaatta on muotoa

$$I_x(t)'' \approx \mathbf{M}'' * I_x(t)' = \{F_{i+1}(t) - 2F_i(t) + F_{i-1}(t)\}_{i=2}^{n_p-1}.$$

Myös tämä vaihe on esitetty kuvassa 15. Jos toiseen derivaattaan halutaan päästä ilman välivaiheita, voidaan konvoluutiossa käyttää toisen derivaatan maskia $\mathbf{M}'' = [1, -2, 1]$, jolloin

$$I_x(t)'' \approx \mathbf{M}'' * I_x(t) = \{F_{i+1}(t) - 2F_i(t) + F_{i-1}(t)\}_{i=2}^{n_p-1}.$$

Toisaalta, ensimmäisen derivaatan maski voi olla muotoa $\mathbf{M}' = [1, -1]$, jolloin toisen de-

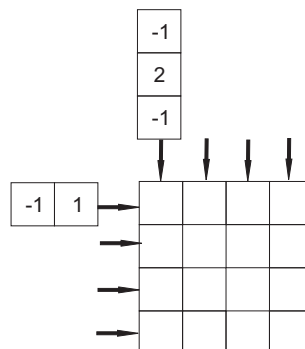


Kuva 15: Derivaatan approksimointi maskia $\mathbf{M}' = [-1, 1]$ käyttäen.

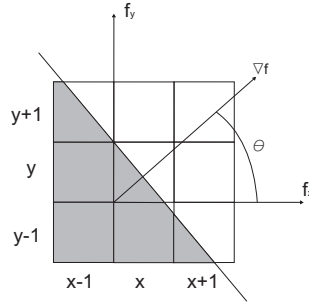
rivaatan maski olisi $\mathbf{M}'' = [-1, 2, -1]$. Sillä, mitä maskia kuvan analysoinnissa käytetään, ei ole väliä, kunhan tiedetään mitä ollaan havaitsemassa. Yhteistä derivoiville maskeille on se, että niiden alkiot summautuvat nolllaksi.

Ensimmäisen derivaatan maskit antavat konvoluution tuloksena suuria arvoja suuren kontrastin kohdalla, kun taas toisen derivaatan maskit antavat nollan suurimman kontrastin kohdalla. Tästä johtuen intensiteetin muutoksesta johtuvat reunat havaitaan ensimmäisen derivaatan maskeilla kuvan vektorin $I_x(t)$ ja maskin \mathbf{M}' konvoluution itseisarvon maksimista ja toisen derivaatan maskeilla konvoluution nolllakohdista.

Edellä esiteltyjä maskeja voidaan käyttää sellaisenaan myös kaksiulotteisessa aineistossa. Tällöin derivaatta lasketaan kuvan I_{xy} yhdelle riville, tai sarakkeelle kerrallaan konvoloida yksiulotteisen maskin kanssa, mikä on esitetty kuvassa 16.



Kuva 16: Kaksiulotteinen kuva voidaan konvoloida käyttäen yksiulotteisia maskeja rivi, tai sarake kerrallaan.



Kuva 17: Kaksiulotteisella reunalla on suunta θ

Kuvan taustalla olevan intensiteettifunktion ollessa muotoa $f(x, y)$, on intensiteetin suurimmalla muutoksella myös suunta θ , mikä ilmenee kuvasta 17. Suunta voidaan ratkaista kuvan intensiteettifunktion gradientin avulla: $\theta = [\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}]$. Kuvan I pisteen (x, y) kontrastia voidaan arvioida laskemalla $\frac{|f(x_{i-1}, y) - f(x_{i+1}, y)|}{2}$ ja $\frac{|f(x, y_{i-1}) - f(x, y_{i+1})|}{2}$, jotka ovat pikselin (i, j) ympäristössä x - ja y -akselien suuntaisesti tapahtuneet intensiteettien muutokset. Koska pikseleissä on kohinaa, ja koska reuna voi olla mihin suuntaan θ tahansa, suuremman ympäristön pikselien huomioonottaminen parantaa estimoinnin tulosta. Tällöin pisteen (i, j) ympäristön kontrastien estimointi suunnissa x ja y tapahtuisi seuraavasti:

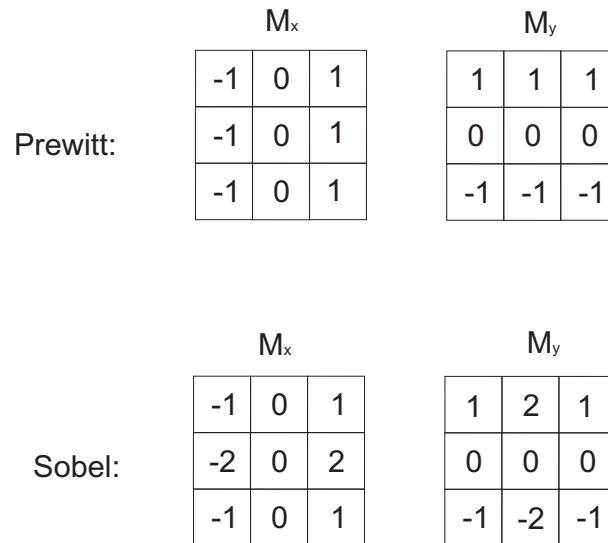
$$\begin{aligned} \frac{\partial f}{\partial x} \equiv f'_x &\approx \frac{1}{3} \left[\frac{|f(x_{i-1}, y_j) - f(x_{i+1}, y_j)|}{2} \right. \\ &+ \frac{|f(x_{i-1}, y_{j-1}) - f(x_{i+1}, y_{j-1})|}{2} \\ &\left. + \frac{|f(x_{i-1}, y_{j+1}) - f(x_{i+1}, y_{j+1})|}{2} \right] \end{aligned}$$

ja

$$\begin{aligned} \frac{\partial f}{\partial y} \equiv f'_y &\approx \frac{1}{3} \left[\frac{|f(x, y-1) - f(x, y+1)|}{2} \right. \\ &+ \frac{|f(x-1, y-1) - f(x-1, y+1)|}{2} \\ &\left. + \frac{|f(x+1, y-1) - f(x+1, y+1)|}{2} \right]. \end{aligned}$$

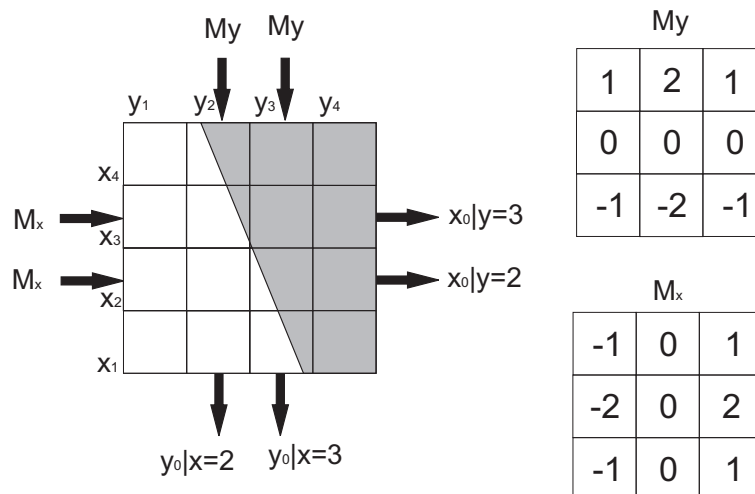
Useissa tapauksissa jakolaskuja ei suoriteta laskennallisen tehokkuuden saavuttamiseksi, mistä johtuen estimaatit eivät ole skaalattuja. Näiden kahden kontrastioperaattorin maskit on esitetty ylimpänä kuvassa 18 ja niitä kutsutaan Prewittin maskeiksi. Niiden alapuolella olevat maskit ovat Sobelin maskit, jotka saadaan saman periaatteen mukaisesti kuin Prewittin maskit, mutta niiden ideana on painottaa keskimmäisiä arvoja kaksinkertaisesti reunoihin verrattuna. Myös niiden tulkinnat ovat samat. Nämä kaksi maskityyppiä ovat yleisimmin käytössä olevien digitaalisten gradienttien laskentaoperaattoreiden joukossa. Verrattaessa näitä kahta maskityyppiä Prewittin maskit ovat helpompia implementoida,

mutta Sobelin maskit eivät ole niin herkkiä kohinalle, mikä on tärkeä ominaisuus derivoitaessa. [7]



Kuva 18: Kaksiulotteiset derivaattaa approksimoivat maskit.

Kuvassa 19 on esitetty kuinka reunantunnistus tapahtuu Sobelin maskien avulla 4×4 kokoisesta kuvasta, jossa reunan sijainti \hat{X}_0 , tai \hat{Y}_0 havaitaan ehdollisen konvoluution maksimisista. Edellä kuvattujen derivaattamaskien ongelma on, että ne ovat määriteltyjä vain pikselivälillä Δp leveydellä. Koska monet reaali maailman reunat ovat sileämpiä eli intensiteetti kasvaa usean pikselin alueella, soveltuvat derivaattamaskit M' ja M'' huonosti reunantunnistukseen. Derivaattamaski onkin erittäin herkkä reaali maailman kuvissa esiintyvälle kohinalle.



Kuva 19: Reunantunnistus diskreetillä Sobelin maskeilla 4×4 ikkunasta, sekä x -, että y -akselin suuntaisesti.

Ratkaisu ongelmaan löytyy silotuksen avulla, jossa lopullinen reunamaski muodostetaan konvolvoimalla derivaattamaski sopivalla silottimella, kuten Canny'n ja Marr Hildreth'in menetelmissä tehdään. Tällöin konvoluutioon lisätään Gaussin funktiosta diskretisoitu silottava maski

$$S = \{g(z_k)\}_{k=1}^{n_g},$$

jolloin ensimmäisen asteen derivaatta tulee muotoon

$$m(x_i) = S * [\mathbf{M}' * I_x] = \underbrace{[S * \mathbf{M}']}_{\text{DoG}} * I_x,$$

missä DoG on diskreetti versio äärimmäisen kapeasta Canny'n Derivative of Gaussian maskista. Vastaavasti Laplacian of Gaussian suodatin saadaan tapauksesta

$$m(x_i) = S * [\mathbf{M}'' * I_x] = \underbrace{[S * \mathbf{M}'']}_{\text{LoG}} * I_x,$$

missä LoG on diskreetti versio kapeasta Laplacian of Gaussian maskista.

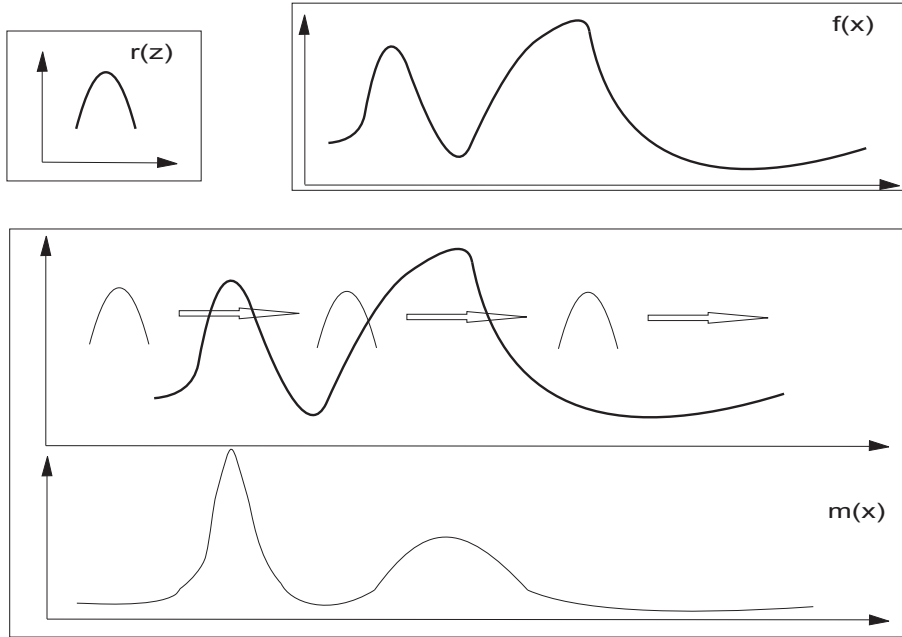
2.4.3 Templaattimaskien muodostaminen

Templaattisovitus perustuu tunnetun kohteen $r(z)$ hakuun ilmiöstä $F(x)$. Tämä mahdollistaa konvoluution korrelaatio-ominaisuuden, joka on esitetty yhtälössä (9). Korrelaatiohan mittaa asioiden riippuvuutta toisistaan, joka tässä tapauksessa voidaan ajatella yhteensopivuuden mitaksi. Piste, jossa korrelaatio on suurimmillaan, on sellainen, jossa etsittävä kohde $r(z)$ ja tutkittava ilmiö $F(x)$ ovat muodoiltaan samankaltaisimmat. Ideaa on pyritty havainnollistamaan graafisesti kuvassa 20.

Jotta templaattisovitus ei riippuisi funktion $f(x)$ skaalasta vaan ainoastaan tunnistettavan kohteen muodosta, tulee reunafunktio määritellä siten, että sen intensiteetin integraali määrittelyalueen yli laskettuna häviää eli

$$\int_{-a}^a r(z) dz = 0.$$

Jos tätä integraalia ei aseteta nolaksi, saa korkeamman intensiteetin alueella sijaitseva muoto eri responssin kuin alhaisen intensiteetin kohdalla sijaitseva samanlainen muoto. Tämä ilmiö on havainnollistettu kuvassa 21. Käytännössä intensiteetin summautuminen nolkaan saadaan aikaan vähentämällä intensiteetin odotusarvo $\mathbb{E}[r(z)]$ funktiosta $r(z)$.



Kuva 20: Kohdetta $r(z)$ sovitetaan funktioon $f(x)$ liu'uttamalla sen yli määrittelyalueen. Vasteessa $m(x)$ näkyvät suurina arvoina kohdat, joissa $f(x)$:n muoto vastaa $r(z)$:n muotoa.

Varsinainen maski on $r(z)$:n diskretisoitu ja keskitetty esitysmuoto eli

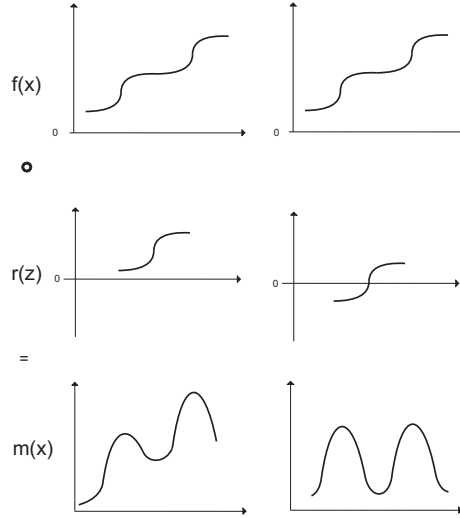
$$\mathbf{T}' = \{r(z_k)\}_{k=1}^{n_r} - \sum_{k=1}^{n_r} r(z_k).$$

Jos jokin jatkuvan templaatin $r(z)$ piste z^0 on valittu templaatin kohdistuspisteeksi, on diskretisoinnin yhteydessä järkevää valita diskretisoinnin alkupiste z_a siten, että kohdistuspiste z^0 tulee olemaan jokin diskretisointipisteistä z_1, \dots, z_{n_r} .

2.5 Interpolointiin perustuvien reunansuodattimien toteutuksesta

Edellä esitetyt interpolointiin perustuvat alipikselin tarkkuuden menetelmät ovat suhteellisen helppoja toteuttaa käytännössä. Toimintatapa on seuraava:

1. Interpoloidaan aineisto $I = \{F(x_i)\}_{i=1}^{n_p}$ tarkempaan resoluutioon $\hat{I} = \{F(x_l)\}_{l=1}^{n_{\text{sim}}}$, missä $n_{\text{sim}} \gg n_p$.
2. Valitaan reunasuodatin $g(z)$, esim.
 - a) Canny'n DoG suodin



Kuva 21: Jos sovitettavan pinnan $r(z)$ integraali ei kohteen määrittelyalueella ole nolla, saa korrelaation responssi eri arvot samanlaisilla kohteilla volyymin vaihtumisen johdosta. Jos sovitettava pinta on määritelty siten, että $\int_{-a}^a r(z)dz = 0$, ei tätä ongelmaa synny vaan kohteet havaitaan samalla tavoin volyymistä riippumatta.

b) Marr-Hildrethin LoG suodin, tai

c) templaatti

sekä määrätään sen mahdolliset parametrit.

3. Diskretisoidaan reunasuodatin uuden interpoloidun aineiston tarkkuuteen

$\Delta sim \ll \Delta p$, eli

$$G = \{g(z_a + k \cdot \Delta sim)\}_{k=1}^{n_r}$$

4. Tehdään konvoluutio interpoloidun kuvan ja diskretisoidun maskin välillä

$$m(x_i) = \hat{I} * G.$$

5. Tehdään varsinainen tunnistus konvoluution vastevektorin $m(x_i)$ avulla. Tämä riippuu käytettävästä suotimesta. Templaatin ja DoG suotimen yhteydessä reunan sijainniksi määrätään responssin maksimi

$$\hat{X}_0 = \arg \max_x m(x_i), \quad i = 1, \dots, n_{sim}.$$

Marr-Hildrethin suodin puolestaan vaihtaa etumerkkiä reunan kohdalla, joten vastaukseksi saadaan sijainti niiden kahden pikselin väliltä, jolla

$$m(x_i) \cdot m(x_{i+1}) < 0.$$

Tämä ei tapahdu välttämättä symmetrisesti, eli tyypillisesti $|m(x_i)| \neq |m(x_{i+1})|$, joka sisältää tuloksena saatavan sijainnin tarkennusmahdollisuuden.

2.5.1 Fourier-muunnoksen käyttö konvoluution laskemiseksi

Konvoluutioon perustuvat reunantunnistusmenetelmät ovat usein laskennallisesti raskaita, jos ne toteutetaan spatiaalisesti. Laskentaa voidaan onneksi keventää käyttämällä Fourier-muunnosta. Fourier-muunnoksessa funktio esitetään siniaaltoisten funktioiden integraalina. Fourier-muunnos $\mathbf{F}(u)$ funktiolle $f(x)$ on muotoa

$$\mathbf{F}(u) = \mathfrak{F}(f(x)) = \int_{-\infty}^{\infty} f(x)e^{-i2\pi xu} dx.$$

missä i on imaginääriluku $i = \sqrt{-1}$. Sen avulla funktiot voidaan muuntaa spatiaalisesta avaruudesta taajuusavaruuteen. Merkitään funktioiden f ja g Fourier-muunnoksia muodossa $\mathfrak{F}(f) = \mathbf{F}$ ja $\mathfrak{F}(g) = \mathbf{G}$.

Konvoluutioteoremana tunnettu ominaisuus perustuu siihen, että funktioiden f ja g konvoluutio saadaan taajuusavaruudessa Fourier-muunnosten vastinelementtien tulona

$$f * g = \mathfrak{F}(f) \cdot \mathfrak{F}(g) = \mathbf{F} \cdot \mathbf{G}$$

ja käänteinen Fourier-muunnos saadaan puolestaan muodossa

$$f(x) = \mathfrak{F}^{-1}(\mathbf{F}(u)) = \int_{-\infty}^{\infty} \mathbf{F}(u)e^{i2\pi xu} du.$$

Käänteinen Fourier-muunnos $\mathfrak{F}^{-1}(\mathbf{F} \cdot \mathbf{G})$ muuntaa konvoluution, tai minkä tahansa muun operaation, takaisin spatiaaliseen avaruuteen.

Fourier-muunnos pystytään esittämään myös diskreetissä muodossa. Yksiulotteinen diskreetti Fourier-muunnos on muotoa

$$\mathbf{F}(u) = \frac{1}{n} \sum_{x=1}^n f(x)e^{-\frac{i2\pi ux}{n}}, \quad u = 1, \dots, n$$

ja

$$f(x) = \frac{1}{n} \sum_{u=0}^{n-1} \mathbf{F}(u)e^{-\frac{i2\pi ux}{n}}, \quad x = 1, \dots, n.$$

Kaksiulotteinen diskreetti Fourier-muunnos on puolestaan muotoa

$$\mathbf{F}(u, v) = \frac{1}{n^2} \sum_{x=1}^n \sum_{y=1}^n f(x, y)e^{-i2\pi(\frac{ux + vy}{n})}$$

ja sen käänteismuunnos

$$f(x, y) = \sum_{u=1}^n \sum_{v=1}^n \mathbf{F}(u, v)e^{-i2\pi(\frac{ux + vy}{n})}$$

Hyödylliseksi Fourier-muunnoksen tekee Cooleyn ja Tukeyn [5] vuonna 1965 keksimä nopea Fourier-muunnosalgorithmi FFT, jolla diskreetin Fourier-muunnoksen voi laskea $n \log_2 n$ operaatiolla yksiulotteisessa tapauksessa.

Käytännössä reunantunnistus, joka käyttää apunaan Fourier-muunnosta tehdään siten, että havaitulle kuvalle $I(t)$ ja käytettävälle suotimelle g tehdään Fourier-muunnokset $\mathbf{I} = \mathfrak{F}(I(t))$ ja $\mathbf{G} = \mathfrak{F}(g)$, jolloin päästään taajuusavaruuteen. Konvoluutio lasketaan taajuusavaruudessa Fourier-muunnosten tulona $\mathbf{I} \cdot \mathbf{G}$. Tämän jälkeen siirrytään takaisin spataaliseen avaruuteen käyttämällä käänteistä Fourier-muunnosta \mathfrak{F}^{-1} . Konvoluution voi tällöin kirjoittaa muodossa

$$m(x) = \mathfrak{F}^{-1}(\mathbf{I} \cdot \mathbf{G}) = \mathfrak{F}^{-1}(\mathfrak{F}(I_x) \cdot \mathfrak{F}(G)).$$

3 Kohteen sijainnin tunnistusmenetelmien teoreettinen tarkastelu

Konenäköön perustuvassa mittauksessa on nähtävissä kaksi erityyppistä tehtävää:

- Mittauskohteiden tunnistus
- Sijainnin mitta

Ensimmäisessä vaiheessa etsitään mittauksen kohde, esimerkiksi kappaleen reuna, siten, että se ei sekoitu muihin kohteisiin. Tämän jälkeen voidaan soveltaa tarkempia menetelmiä kohteen sijainnin tarkkaan määrittämiseen. Tässä luvussa pääosin oletetaan, että tunnistustehtävä on jo ratkaistu, jolloin mielenkiinnon kohteena on tunnistustarkkuuden arviointi.

3.1 Mittauksen harha ja varianssi

Olettaen, että kohteen oikea sijainti on x -akselin pisteessä x_0^* ja havaittua sijaintia edustaa satunnaismuuttuja (reunan estimaattori) \hat{X}_0 . Tällöin mittausvirhettä voidaan karakterisoida sijaintiestimaattorin \hat{X}_0 jakauman tai siitä laskettujen tunnuslukujen avulla. Tässä työssä pääasiallisena tarkkuuden mittana käytetään keskineliövirhettä $MSE = \mathbb{E}[\hat{X}_0 - x_0^*]^2$, joka mittaa havaitun \hat{x}_0 sijainnin odotusarvoista neliöllistä etäisyyttä todellisesta sijainnista x_0^* . Keskineliövirhe on mahdollista jakaa harha- ja hajontakomponentteihin seuraavasti:

$$\begin{aligned} MSE &= \mathbb{E}(\hat{X}_0 - x_0^*)^2 && (14) \\ &= \mathbb{E}(\hat{X}_0 - \mathbb{E}\hat{X}_0 + \mathbb{E}\hat{X}_0 - x_0^*)^2 \\ &= \mathbb{E}(\hat{X}_0 - \mathbb{E}\hat{X}_0)^2 + 2\underbrace{\mathbb{E}(\hat{X}_0 - \mathbb{E}\hat{X}_0)(\mathbb{E}\hat{X}_0 - x_0^*)}_{=0} + (\mathbb{E}\hat{X}_0 - x_0^*)^2 \\ &= \underbrace{\mathbb{E}(\hat{X}_0 - \mathbb{E}\hat{X}_0)^2}_{\text{varianssi}} + \underbrace{(\mathbb{E}\hat{X}_0 - x_0^*)^2}_{\text{harha}^2}, \end{aligned}$$

jossa $\mathbb{E}\hat{X}_0 = x_0$, joka on kohteen havaittava sijainti. Hajotelmasta nähdään, että virhe koostuu harhasta ja varianssista. Näistä varianssivirhe on usein merkittävämpi, sillä harha voidaan monessa tapauksessa poistaa mittausjärjestelmän kalibroinnilla. Käytännössä

neliölliset mitat ovat hankalasti tulkittavia, joten niistä otetaan usein neliöjuuret. Lisäksi mitat voidaan suhteuttaa esimerkiksi tutkittavan kappaleen kokoon tai kameran resoluutioon. Tyypillisiä virhemittoja ovat tällöin esimerkiksi

- Mittausten keskihajonta

$$sd = \sqrt{\mathbb{E}(\hat{X}_0 - \mathbb{E}(\hat{X}_0))^2}$$

- Absoluuttinen harha

$$|Bias| = |\mathbb{E}(\hat{X}_0) - x_0^*|$$

- Pikselivirhe

$$pixerr = \alpha = \frac{\sqrt{MSE}}{\Delta p},$$

missä Δp on pikselin leveys.

- Pikselihajonta

$$sd(\alpha) = \frac{sd}{\Delta p}.$$

Pikselivirhe kertoo tarkkuuden, jolla havaitut kohteet ovat jakautuneet suhteessa diskretisointiväliin. Jos $\alpha = 1$ on havaintotarkkuus yksi pikseli. Jos $\alpha < 1$, päästään havainnoissa alipikselintarkkuuteen ja käytetystä alipikselinestimointimenetelmästä on hyötyä. Työssä pyritään tarkkuuteen, joka pikselivirheellä mitattuna olisi pienempää kuin kymmenesosa, $\alpha < 0.1$

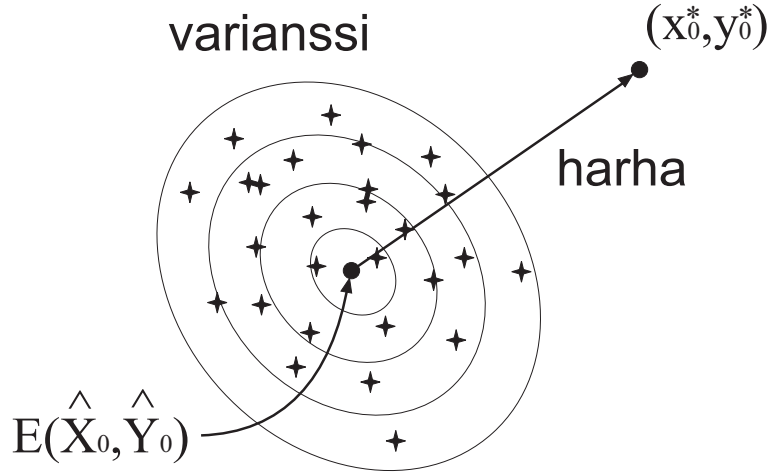
Empiirisiä kokeita tehtäessä voidaan odotusarvoon (keskiarvoon) perustuvat mitat korvata helposti esim. mediaanilla, joka on keskiarvoa robustimpi yksittäisille mittavirheille. Lisäksi keskiarvon ja mediaanin suhteesta on mahdollista tehdä päätelmiä mm. virhejakauman mahdollisista epäsymmetrioista.

Kaksiulotteisessa tilanteessa sijainnin virhemitat ovat havaitun sijainnin (\hat{X}_0, \hat{Y}_0) yhteisjaukauman tunnuslukuja todellisen sijainnin (x_0^*, y_0^*) suhteen. Tilannetta voidaan havainnollistaa graafisesti kuvan 22 tavalla, mistä nähdään, että harha voidaan esittää vektorina

$$Bias(\hat{X}_0, \hat{Y}_0 | x_0^*, y_0^*) = (\mathbb{E}(\hat{X}_0) - x_0^*, \mathbb{E}(\hat{Y}_0) - y_0^*),$$

josta useimmissa tapauksissa tunnusluvuksi riittää vektorin pituus

$$|Bias_{xy}| = \sqrt{(\mathbb{E}\hat{X}_0 - x_0^*)^2 + (\mathbb{E}\hat{Y}_0 - y_0^*)^2}. \quad (15)$$



Kuva 22: Harha ja variassi kaksiulotteisessa aineistossa.

Tämä voidaan edellä kuvattuun tapaan esittää myös pikseliharhana

$$Bias(\alpha_{xy}) = \frac{|Bias_{xy}|}{\Delta p}.$$

Varianssivirhe puolestaan vaihtelee eri suuntiin sen mukaan, mikä on x - ja y -akselien sijaintien \hat{X}_0 ja \hat{Y}_0 riippuvuusrakente. Jos \hat{X}_0 ja \hat{Y}_0 ovat riippumattomia, näin syntyvä virhe voidaan laskea esim. odotusarvoisena hajontana

$$sd(\hat{X}_0, \hat{Y}_0) = \mathbb{E} \left[\sqrt{(\hat{X}_0 - \mathbb{E}(\hat{X}_0))^2 + (\hat{Y}_0 - \mathbb{E}(\hat{Y}_0))^2} \right],$$

ja sen havainnoista $\{\hat{X}_0(i), \hat{Y}_0(i)\}_{i=1}^N$ laskettu estimaattori on

$$sd_{xy} = \frac{1}{N} \sum_{i=1}^N \sqrt{\left(\hat{X}_0(i) - \frac{1}{N} \sum_{k=1}^N \hat{X}_0(k) \right)^2 + \left(\hat{Y}_0(i) - \frac{1}{N} \sum_{k=1}^N \hat{Y}_0(k) \right)^2}.$$

Pikselin leveyden suhteutettuna mittausten hajonta on muotoa

$$sd(\alpha_{xy}) = \frac{sd_{xy}}{\Delta p},$$

josta kaavan (15) kanssa voimme yleistää kaksiulotteisten mittausten neliölliseksi keski-
virheeksi pikseliä kohden

$$\alpha_{xy} = \frac{\text{MSE}}{\Delta p} = Bias(\alpha_{xy})^2 + sd(\alpha_{xy})^2.$$

3.2 Arvioita reunantunnistimilla saavutettavasta mittaustarkkuudesta

Reunantunnistimien soveltuvuutta kohteiden sijainnin määrittämiseen on tutkittu jonkin verran. Tunnetuin esimerkki tästä sisältyy Cannyn alkuperäiseen työhön [2], jossa optimaalinen reunantunnistin määriteltiin maksimoimalla reunan havaitsemisen responsin ja

paikallistamisvirheen suhdetta. Koska Canny'n lokaatiovirhe mittaa nimenomaan löydetyn reunan sijainnin \hat{X}_0 ja todellisen reunan sijainnin x_0^* välistä eroa, sisältyy Canny'n algoritmiin automaattisesti myös arvio mittauksen tarkkuudesta. Käsittelemme Canny'n virheaprosimaatiota tarkemmin luvussa 3.3.1, mutta ensin teemme lyhyen katsauksen muihin kirjallisuudessa esitettyihin tuloksiin.

Luvussa 2 esitettyjä alipikselin tarkkuuden menetelmiä on analysoitu useammissa artikkeleissa. Menetelmät ovat jaettavissa kahteen eri osaan. Nämä osat ovat tämän lisäksi jakautuneet tarkastelluissa tutkimuksissa siten, että interpolointia käyttävissä menetelmissä tunnistimena on joko DoG- (DoG) tai LoG-suodin (LoG). Suoraan diskreetistä aineistosta alipikselin tarkkuuden saavuttavat menetelmät puolestaan voidaan jakaa momenttipohjaisiin (Momentti) tai pienimmän neliösumman menetelmiin (PNS). Koska aineisto voidaan ajatella jatkuvana (jatkuva) tai diskreettinä (Pikseli tai Interpoloitu), voidaan tulokset jaotella myös tämän mukaan. Diskreetin aineiston tapauksessa osa menetelmistä käyttää interpolointia alipikselin tarkkuuteen pääsemiseksi (Interpoloitu), mutta ne saattavat ottaa myös diskretisoinnin vaikutuksen huomioon (Pikseli). Loput esitellyistä menetelmistä arvioivat alipikselintarkkuuden sijainnin suoraan diskreetistä aineistosta (Pikseli). Havaittu sijainti on kaikissa tässä työssä tarkastelluissa tuloksissa jatkuva muuttuja. Tarkastelut on saatettu toteuttaa sekä yksi- (1D), että kaksiulotteisissa (2D) tapauksissa. Yhteenvedo

Tekijä	DoG	LoG	Momentti	Pns	Jatkuva	Pikseli	Interpoloitu	Teoreett.	Empiirinen	1-D	2-D
Canny	●				●			●	●	●	●
Tagare jne.	●	●			●			●		●	
Sarkar jne.	●				●			●	●	●	●
Koplowitz jne.	●	●			●			●		●	
Kakarala jne.	●				●			●		●	
Rohr				●	●			●	●	●	●
Devernay	●	●					●		●	●	
Ramdani jne.	●	●					●	●	●		●
Åström jne.	●	●				●	●	●	●	●	●
Steger	●						●	●	●	●	●
Rockett	●						●		●		●
Tabatabai jne.			●		●	●			●	●	●
Lyvers jne.		●	●		●	●		●	●	●	●
Ye jne.				●		●	●		●	●	●

Kuva 23: Yhteenvedo teoreettisista analyyseista

näissä tutkimuksissa esiintyvistä käsitteistä on tehty kuvassa 23. Tämän lisäksi olemme yleiskuvan kehittämiseksi sisällyttäneet töiden pääasiallisen sisällön taulukkoon 1.

Edellämainitut Canny ja Marr-Hildrethin algoritmit ovat reunantunnistusmenetelmistä ehkä yleisimmin käytössä. Molemmat algoritmit perustuvat derivaatan approksimointiin ja ovat alunperin tarkoitettuja luomaan reunakarttoja kuvista. Niitä on kuitenkin mahdollista käyttää myös kohteen sijainnin etsimiseen. Niiden yleisyydestä johtuen menetelmiä onkin tarkasteltu useista eri lähtökohdista, kuten kuvasta 23 ja taulukosta 1 nähdään. Seuraavaksi esittelemmekin tietyssä määrin derivaatan approksimointiin perustuvien menetelmien alipikselitason analyysiin liittyviä julkaisuja ([6], [9], [17], [20], [26], [34]).

Kakarala ja Hero [9] käyttävät tilastollisia parametrien estimointitekniikoita määrittämään rajat saavutettavissa olevalle tarkkuudelle reunantunnistuksessa. Nämä niin sanotut Cramér-Rao -rajat paljastavat joidenkin tekijöiden vaikutuksen reunantunnistuksessa. Heidän työsäään kuvan koordinaatit ja kuvan intensiteetti ovat jatkuvia muuttujia, mistä johtuen tutkimuksen voidaan olettaa tapahtuvan alipikselintarkkuudella.

Reunantunnistuksen tarkkuudelle johdetut rajat syntyvät seuraavien tekijöiden tuloksena: signaali-kohinasuhde (SNR), havaitun reunan leveys, silottavan suodattimen skaala ja ennakkotieto reunan intensiteetin luotettavuudesta. Kakarala ja Hero johtavat alemman Cramér-Rao -rajan ($CRLB$) MSE:lle, jota on mahdollista soveltaa kaikille sellaisille reunantunnistimille, joilla on tunnettu harha. $CRLB$ mittaa osuutta, johon Canny algoritmi pääsee parhaan saavutettavan reunantunnistuksen tarkkuuden suhteen.

Kakarala ja Hero ovat johtaneet Cramér-Rao -rajat gaussiselle kynnyksireunalle

$$hf(x) = h \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{z^2}{2}} dz, \quad (16)$$

jossa h on intensiteetin muutos reunassa. Havaittuun funktioon $F(x, y)$ on lisätty additiivista gaussista kohinaa $F(x, y) = f(x) + \varepsilon_{x,y}$. Kohinan varianssi on σ^2 ja $SNR = \frac{h}{\sigma}$.

Erona edeltäviin tutkimuksiin, he vastaavat seuraaviin reunantunnistukseen liittyviin kysymyksiin $CRLB$:n avulla:

1. Mikä on vaihtelevien SNR :n, havaitun reunan leveyden ja reunasta olevan ennako-

Canny [2], [3]	Cannyn algoritmi, jossa reunantunnistimen muodostaminen, jatkuvan tapauksen analyysit, sekä Cannyn kriteerit reunantunnistukselle
Tagare, deFigueiredo [29], [30]	Cannyn ja Marr-Hildrethin tunnistimien reunan paikallistamisen teoreettiset tarkastelut jatkuvalla yksiulotteiselle kynnsreunalle, muutosehdotukset Cannyn kriteereihin, sekä vastaus Sarkarin ja Boyerin kritiikkiin.
Sarkar, Boyer [21], [22]	Cannyn algoritmiin perustuvan suotimen luominen ja sen analysointi ja vertailu DoG-suotimeen, sekä kritiikki Tagaren ja deFigueiredon muokkaamiin kriteereihin.
Koplowitz, Greco [11]	Cannyn ja Marr-Hildrethin algoritmien reunan sijainnin paikallistamisen tarkastelut ja reunan sijainnin jakauman johtaminen jatkuvassa yksiulotteisessa tapauksessa
Kakarala, Hero [9]	Reunantunnistuksessa saavutettavan sijainnin tarkkuuden alaraja (Cramer-Rao alaraja) yksiulotteisen jatkuvan kynnsreunan tapauksessa, sekä Cannyn algoritmin tarkkuus suhteessa tähän rajaan.
Rohr [19]	Kakaralan ja Heron työn yleistys, jossa tarkastellaan useampien kaksiulotteisten kohteiden paikallistamisen alarajaa.
Devernay [6]	NMS-algoritmin suorittaminen alipikselin tarkkuudella kvadraattista tai lineaarista interpolointia käyttäen, sekä empiiriset yksiulotteiset tarkastelut.
Ramdani, et al. [17]	NMS-algoritmin teoreettiset ja empiiriset tarkastelut kaksiulotteisessa tapauksessa. (Devernayn työn yleistys)
Åström, Heyden [34]	Cannyn ja Marr-Hildrethin reunantunnistimien reunantunnistimien ja kuvanmuodostumisprosessin stokastinen analyysi ja empiiriset tarkastelut kaksiulotteisessa tapauksessa, jossa reunantunnistus suoritetaan alipikselin tarkkuudella
Steger [26]	Reunan ja viivan sijaintien, sekä varianssien tarkastelut alipikselin tarkkuudella kaksiulotteisessa tapauksessa yksinkertaisella kuvanmuodostumisprosessilla.
Rockett [20]	Cannyn algoritmin empiiriset tarkastelut alipikselin tarkkuudella.
Tabatabai, Mitchell [27]	Momenttipohjaisen tunnistimen ja Hueckelin tunnistimen alipikselin tarkkuuden empiiriset tarkastelut
Lyvers, et al. [12]	Momenttipohjaisten menetelmien analysointi, sekä useamman menetelmän (Hueckel,LoG) vertailu empiirisesti
Ye, et al. [32]	Pienimmän neliösumman menetelmän empiirinen vertailu

Taulukko 1: Lyhyet kuvaukset julkaisujen sisällöistä

tiedon vaikutus saavutettavaan MSE:hen?

2. Kuinka lähellä optimia Canny'n reunantunnistusalgoritmi on CRLB:n saavuttamisen suhteen?

Kakarala ja Hero tutkivat ainoastaan harhattomia estimaattoreita, sillä lineaariset suodattimet ja heidän käyttämänsä suurimman uskottavuuden suodatin ovat harhattomia kyseisen reunan tapauksessa. Harhattomassa tapauksessa he pystyvät tiivistämään Cramér-Rao-rajan lauseeseen

$$\mathbb{E}[(\hat{X}_0 - x_0^*)^2] \geq \frac{\sigma^2 \sqrt{\pi} \sigma_s}{h^2 T_y}, \quad (17)$$

jossa T_y on puolet reunasuotimen leveydestä y -akselin suunnassa ja σ_s on gaussisen reunan hajontaparametri. Tästä huomataan, että kasvattamalla reunantunnistimen leveyttä y -akselin suuntaisesti paranee reunantunnistuksessa saavutettava tarkkuus.

Kakaralan ja Heron työ rajoittuu kynnyksireunan tapaukseen. Työn yleistys löytyy Rohrin työstä [19], jossa hän tutkii kohinaisen aineiston satunnaisjäännöksiä ja johtaa lausekkeen pienimmän epävarmuuden sijainnin estimaattorille annettujen kaksiulotteisten epälineaaristen reunojen ja kulmien tapauksissa.

Devernay [6], sekä myöhemmin Ramdani, Bouchara ja Djemal [17], tarkastelevat Canny'n ja Marr-Hildreth'in algoritmiin perustuvaa non-maxima suppression -menetelmää. Töissä päästään alipikselin tarkkuuteen käyttämällä lineaarista tai kvadraattista interpolointia. Devernayn teoreettiset tarkastelut rajoittuvat kuitenkin vain yksiulotteisen tapauksen. Hän etsii reunan sijainnin digitaalisesta kuvasta etsintäsuunnasta non-maxima suppression (NMS) -metodia käyttäen. Tämän jälkeen hän interpoloi sovitteen kolmen vierekkäisen pikselin avulla. Reuna havaitaan interpoloidusta sovitteesta alipikselin tarkkuudella. Empiirisesti Devernay tarkasteli havaitun reunan sijainnin keskiarvoista erotusta todellisen reunan sijainnista ja vierekkäisten pikselien muodostaman viivan eroa todellisesta viivasta. Huomioon otettavaa on, että Devernayn menetelmä on kehitetty reunakartan luomiseen, ei niinkään reunan sijainnin mittaamiseen.

Ramdani, Bouchara ja Djemal [17] analysoivat Devernayn menetelmän teoreettisesti ja vertailevat teoreettisia tuloksia empiirisesti havaittuihin. Interpolointimenetelmänä he tutkivat ainoastaan kvadraattista interpolointia. Jatkuvaan kuvamalliin he olettavat additii-

visen kohinan, jolloin malli on muotoa

$$F(x, y) = f(x, y) + \varepsilon(x, y), \quad (18)$$

jossa ε on nollakeskeisesti jakautunut autokorreloitu virhetermi. Mallin pohjalta he johtavat interpoloinnin avulla havaitun reunan sijainnin jakauman kaksi ensimmäistä momenttia. Empiiristen ja teoreettisten tulosten vertailujen yhteydessä teoreettisen mallin ongelmaksi muodostuu heidän olettamansa gaussisuus, mikä ei pidä paikkaansa, sillä kahden gaussisen muuttujan suhde ei ole gaussinen. Tästä syntyvä virhe voidaan kuitenkin olettaa olemattomaksi pienillä kohinan määrillä. Edellisen lisäksi tarkasteluissa paljastuu deterministisen estimaattorin harha, joka on kuitenkin mahdollista poistaa harhan estimoinnin avulla.

Åström ja Heyden [34] tutkivat omassa työssään derivaatan approksimointia käyttäviä reunantunnistimia hieman eri lähtökohdista. Heidän työnsä on saanut suuresti huomiota myös muissa alan julkaisuissa ([26], [17]). Verrattuna vastaaviin töihin heidän kuvamallinsa on monimutkaisempi ja se ottaakin huomioon kuvan sumentamisen, diskretisoinnin ja kohinan. Tutkimuksen he suorittavat spatiaalisten satunnaisprosessien teoriaa käyttäen. Kohina sisältyy malliin diskretisointivaiheen jälkeen ollen additiivinen autokorreloitu kenttä, mutta pikseleissä kohinatermien voidaan ajatella olevan toisistaan riippumattomia ja samalla tavalla jakautuneita (iid). Diskreetti aineisto interpoloidaan takaisin jatkuvaan ympäristöön low-pass -interpolointia käyttäen.

Åströmin ja Heydenin kohteena on kynnsyreuna. Tunnistuksessa he määrittelevät reunan sijainnin etsintäsuunnan suurimman gradientin sijainniksi. Reunantunnistus suoritetaan DoG- ja LoG-suotimia vastaavilla gaussisen osan sisältävillä tunnistimilla. Näillä määrityksillä he saavat havaittavan reunan sijainnin varianssin kaavaksi

$$\mathbb{E}(\hat{X}_0^2) = \frac{3\sigma_n^2(\sigma_b^2 + \sigma_e^2)^3}{8h^2\sigma_e^2 \cos^6 \theta},$$

jossa σ_b on gaussisen sumentamisytimen keskihajonta, σ_n kohinan keskihajonta, σ_e reunantunnistimen gaussisen osan keskihajonta, h kynnsyreunan korkeus ja θ kulma tunnistuksen suhteessa reunaan. Empiiriset tutkimukset tukivat teoreettisia tuloksia, mikä tukee tutkimuksen onnistuneisuutta. Tästä huolimatta muunmuassa Steger [26] arvosteli tekijöitä siitä, että he eivät ole esittäneet kaavan johtamisen kaikkia vaiheita, mikä aiheuttaa epävarmuuden sen oikeellisuudesta.

Omassa työssään Steger [26] keskittyy reunojen ja viivojen alipikselitason paikallistamisen

tarkkuuteen sen jälkeen, kun ne on ensin eristetty kuvasta. Tällaiset menetelmät ovatkin tarpeellisia esimerkiksi teollisuuden laaduntarkastustehtävissä. Tämä vastaakin suurelta osin työssämme tarkasteltavaa ongelmaa. Steger tutkii itse kehittämänsä algoritmia, joka löytyy tarkemmin hänen aikaisemmista töistään. Taustalla oleva kohinatön viiva on muotoa

$$f_a(x) = \begin{cases} 0, & x < -w \\ 1, & |x| \leq w \\ a, & x > w, \end{cases} \quad (19)$$

jossa $0 \leq a < 1$. Viivan leveys on siis $2w$. Kuvansaantimalli vastaa Ramdanin ym. [17] mallia, joka löytyy yhtälöstä 18. Näistä lähtökohdista hän johtaa varianssin viivan sijainnille ja leveydelle, sillä mitkään edeltävistä tutkimuksista eivät anna ratkaisua tähän. Tulosten tarkastelut suoritettiin simulointikokein sekä synteettisille, että reaali maailman kuville. Tarkasteluista huomattiin, että menetelmä oli harhaton ja että reaali maailman kuvissa päästiin 99,9% :n varmuudella kymmenesosapikselin tarkkuuteen. Tästä johtuen voidaan sanoa, että reunojen ja viivojen paikallistamisella kyseistä menetelmää käyttäen päästään kymmenesosapikseliä tarkempaan tulokseen ($\alpha < 0.1$).

Peter Rockett [20] tutkii työssään Cannyn reunantunnistusoperaattorin tarkkuutta alipikselintarkkuudella. Kuva on suoraan diskreetissä ympäristössä, jota prosessoidaan Cannyn algoritmilla. Algoritmi vastaa Devernayn tarkastelemaa menetelmää ja käyttää kvadraattista interpolointia alipikselin tarkkuuteen pääsemiseksi. Tarkastelussa kynnysreuna on lähes vastaava kuin Lyversin ja Mitchellin [12] tarkastelema reunafunktio, joka koostuu parametreista b , h , ρ ja θ , missä ρ on reunan etäisyys lähimmän pikselin keskipisteestä. Yksittäisen pikselin intensiteetit laskettiin alhaisen b ja korkean intensiteetin h osuuksien summina painotettuna niiden osuuksilla pikselin alassa.

Työn tuloksina Rockett osoittaa, että tyypillisen kokoisille silotusytimille kvadraattinen interpolointi synnyttää systemaattista virhettä, jota on mahdollista korjata korjaustaulukon avulla. Kuvauksen taustalla olevan ilmiön ollessa normaalisti jakautunutta Kakarala ja Hero [9] osoittivat, että gaussisen funktion sovittaminen kvadraattisen sijaan saavuttaisi Cramér-Rao -rajan. Tämän lisäksi se tuottaa harhattomia tuloksia riippumatta ρ :sta. Gaussisen funktion sovittaminen ei kuitenkaan onnistunut kolmella pisteellä. Puutteena Rockettin työssä on sen rajoittuminen empiirisiin kokeisiin, joiden teoreettisia perusteita ei ole esitelty.

Kuten edellisistä tarkasteluista huomaamme on gradienttipohjaisten reunantunnistimien ominaisuuksia tarkasteltu useissa eri yhteyksissä. Lukuun 2 viitaten ei gradienttipohjaisten menetelmien käyttö ole ainoa vaihtoehto. Myös momentti- ja pienimmän neliösumman menetelmiin liittyvää analysointia ja vertailua on tehty. Seuraavaksi paneudummekin lyhyesti näihin menetelmiin liittyviin julkaisuihin ([8], [12], [27], [32])

Tabatabai ja Mitchell ovat työssään [27] tarkastelleet kehittämänsä momenttipohjaisen menetelmän toimivuutta teoreettisesti yksiulotteisessa tapauksessa ja vertailleet sen toimivuutta sekä Hueckelin menetelmään, että Sobelin gradienttipohjaiseen menetelmään (luku 2.4.2) kaksiulotteisessa tapauksessa. Teoreettisissa tarkasteluissa on keskitytty kohinan vaikutukseen sekä siihen, kuinka mediaani- tai keskiarvosuodatus, eli kohinan tsaaminen monotonisuuden saavuttamiseksi, vaikuttaa reunantunnistukseen. Empiirisissä kaksiulotteisissa tarkasteluissa on vertailtu reunan paikallistamiseen liittyvää virhettä, jossa Tabatabain ja Mitchellin menetelmä näyttäisi toimivan Hueckelin operaattoria paremmin. Toisaalta reunan suunnan havaitsemisessa asetelma on päinvastainen. Yllättävää on, että Sobelin maskia on vertailtu ainoastaan visuaalisesti.

Lyvers, Mitchell, Akey ja Reeves ovat työssään [12] tarkastelleet momenttipohjaisia menetelmiä laaja-alaisemmin. Yksiulotteisessa tapauksessa he ovat tutkineet teoreettisesti diskretisoinnin aiheuttamaa harhaa ja kohinan vaikutusta. Näistä vaikutuksista on tehty kattavat tarkastelut kyseisessä työssä esitellyn spatiaalisen momenttioperaattorin tapauksessa. Tämän lisäksi on tehty vertailu Tabatabain ja Mitchellin harmaasävymomenttioperaattoriin vaihtuvilla kohinan arvoilla.

Kaksiulotteisessa tapauksessa on tehty vastaavat tarkastelut kuin yksiulotteisessa tapauksessa sekä näiden lisäksi on tutkittu harmaasävytasojen kvantitasoinnin, erityyppisten reunojen ja ei-neliönmuotoisten pikselien vaikutusta sijainnin tunnistukseen. Empiirisiin vertailuihin on lisäksi lisätty Hueckelin operaattori ja LoG-suodin, jossa alipikselintarkkuuteen on päästy käyttämällä kaksiulotteista toisen asteen polynomin facet-mallia suodatetun kuvan 3×3 kokoisille alueille.

Momenttipohjaisten menetelmien kanssa samalla tavoin suoraan diskreetistä aineistosta alipikselin tarkkuuteen pääsevä menetelmä on Yen, Fun ja Poudelin esittelemä [32] pienimmän neliösumman menetelmä. Tutkittavana reunana kyseisessä työssä on Shanin ja Boonin [23] inspiroima gaussisesti sekoittunut kynnsyreuna, johon sisältyy korkeamman

asteen polynomi. Pikselin saama harmaasävyarvo havaitaan integraalina pikselin alueelta. Olosuhdemuuttujat vastaavat lähes täysin tässä työssä tarkasteltavia, ollen pikselien lukumäärä ja signaali-kohinasuhde. Julkaisuun [32], josta pienimmän neliösumman menetelmä löytyy, sisältyy laajat vertailut Shanin ja Boonin [23] momenttipohjaiseen ja Stegerin [25] interpolaatioon perustuvaan menetelmään.

Tähän mennessä olemme tarkastelleet käytännössä toteutettuja alipikselitason reunan sijainnin tunnistusmenetelmiä. Seuraavaksi siirrymme tarkastelemaan Cannyn kehittämää gradienttiin perustuvan reunantunnistuksen teoriaa tarkemmin ja johdamme sen pohjalta tässä työssä käyttämämme menetelmän teoreettisen virhearvion.

3.3 Cannyn analyysi löydetyn reunan sijaintivirheelle

Kuten aiemmin on todettu, poikkeaa kohteen todellinen sijainti x_0^* reunantunnistimien avulla löydetystä sijainnista \hat{x}_0 , kun mittausdataan sisältyy kohinaa. Lisäksi löydetty sijainti \hat{x}_0 voidaan tulkita satunnaismuuttujan \hat{X}_0 realisaatioksi, missä \hat{x}_0 vaihtelee mitauskertojen välillä. Voimmekin arvioida mittauksissa syntyvää sijaintivirhettä satunnaismuuttuja \hat{X}_0 :n ominaisuuksien kautta.

Julkaisussa [3] Canny muodostaa erään arvion reunan sijaintivirheelle käyttäen Taylor-aproksimaatiota. Cannyn virheaproksimaation kohteena on hänen kehittämänsä gradienttipohjainen reunantunnistin, jota sovelletaan kohinaiseen kynnyöreunaan

$$F(x) = f(x) + \varepsilon_x,$$

missä $f(x)$ on ideaalinen ilmiö ja ε_x on mittauskohinaa.

Palautetaan mieleen, että Cannyn reunantunnistin perustuu suotimen $g(x)$ antaman kokonaisvasteen eli -responsin

$$m_T(x) = \int g(z)F(x-z)dz$$

maksimointiin. Canny olettaa, että $g(z)$ on rajoitettu välille $[-a, a]$, jolloin kokonaisresponsi voidaan jakaa ideaalisen reunanresponsiin $m_f(x)$ ja kohinan responsiin $m_N(x)$ muodossa

$$m_T(x) = m_f(x) + m_N(x), \tag{20}$$

missä

$$m_N(x) = \int_{-a}^a g(z)N(x-z)dz \quad (21)$$

$$m_f(x) = \int_{-a}^a g(z)f(x-z)dz. \quad (22)$$

Cannyn määritelmän mukaisesti reunan havaittu sijainti \hat{x}_0 on kokonaisresponsin maksimipisteessä. Tällöin, valitsemalla koordinaatisto siten, että reunan todellinen sijainti osuu pisteeseen $x_0^* = 0$, on ideaalisen reunan responsin maksimiarvo tässä pisteessä $m_f(0)$ ja derivaatta sijainnin suhteen on $m'_f(0) = 0$. Toisaalta, havaitussa pisteessä on kokonaisresponsin derivaatta muotoa

$$m'_T(\hat{X}_0) = m'_f(\hat{X}_0) + m'_N(\hat{X}_0) = 0. \quad (23)$$

Canny tekee yhteyden responsin maksimien ja samalla oikean ja löydetyn sijainnin välille käyttäen Taylorin approksimaatiota termille $m'_f(\hat{X}_0)$ pisteessä $x_0^* = 0$:

$$m'_f(\hat{X}_0) = m'_f(0) + \hat{X}_0 m''_f(0) + Err(\hat{X}_0^2). \quad (24)$$

Koska $m'_f(0) = 0$, saa Canny pisteiden \hat{X}_0 ja $x_0^* = 0$ välille yhtälön

$$m'_N(\hat{X}_0) + \hat{X}_0 m''_f(0) \approx 0. \quad (25)$$

Tästä ratkaisemalla \hat{X}_0 saadaan

$$\hat{X}_0 \approx -\frac{m'_N(\hat{X}_0)}{m''_f(0)}. \quad (26)$$

Cannyn mukaan $m'_N(\hat{X}_0)$ on gaussinen satunnaismuuttuja, jonka varianssi on

$$\mathbb{E}[m'_N(\hat{X}_0)^2] = \sigma^2 \int_{-a}^a g'^2(x)dx. \quad (27)$$

Tagare ja deFigueiredo [29] ovat osoittaneet tämän virheelliseksi oletukseksi. Näiden tulosten avulla Canny saa reunan sijainnin keskineliövirheeksi

$$\mathbb{E}[\hat{X}_0^2] \approx \frac{\sigma^2 \int_{-a}^a g'^2(x)dx}{[\int_{-a}^a g'(x)f'(-x)dx]^2}, \quad (28)$$

joka on \hat{X}_0 :n varianssin approksimaatio. Tulos on merkittävä myös sen vuoksi, että Cannyn lokalisaatio- eli paikallistamismitta määräytyy varianssin kautta ollen

$$Localization = \frac{|\int_{-a}^a g'(x)f'(-x)dx|}{\sigma \sqrt{\int_{-a}^a g'^2(x)dx}}. \quad (29)$$

Sarkar ja Boyer [21] tutkivat reunan sijaintia ja sen varianssia myös toista derivaattaa approksimoivien suodattimien tapauksessa. Näissä reunan sijainti löytyy suoraan konvoluution responsin nollakohdasta. Tästä johtuen reunan sijainnin approksimaatioksi saadaan

$$\hat{X}_0 \approx -\frac{m_N(\hat{X}_0)}{m'_f(0)}. \quad (30)$$

Käyttäen Cannyn evaluaatiota $m'_N(\hat{X}_0)$:n varianssista yhtälössä 27 ja saivat toisen derivaatan suodattimen tapauksessa varianssiksi

$$\mathbb{E}[m'_N(\hat{X}_0)^2] = \sigma^2 \int_{-\infty}^{\infty} g^2(x) dx, \quad (31)$$

josta reunan sijainnin varianssiksi saadaan

$$\mathbb{E}[\hat{X}_0^2] \approx \frac{\sigma^2 \int_{-\infty}^{\infty} g^2(x) dx}{[\int_{-\infty}^{\infty} g'(x) f(-x) dx]^2}. \quad (32)$$

Tagare ja de Figueiredo [29] kyseenalaistavat Cannyn johtaman yhtälön 27, sillä kyseinen yhtälö pitää paikkansa ainoastaan silloin, kun $m'_N(x)$ on otettu samassa pisteessä kaikille stokastisen prosessin $m_N(x)$ realisaatioille. Tämä ei pidä paikkaansa, koska \hat{X}_0 vaihtelee jokaisella satunnaisella prosessilla $m'_N(x)$ [11]. Koska näiden varienssien laskeminen näyttää kompleksiselta, Koplowitz ja Greco [11] johtavat reunan sijainnille \hat{X}_0 ilmaisut, jotka eivät sisällä \hat{X}_0 :aa sisältäviä funktioita. Tästä johtuen he tekevät Taylorin approksimaation kokonaisresponsille pelkän kohinattoman ilmiön responsin sijaan. Tällöin derivaatan Taylor-approksimaatio pisteessä $x_0^* = 0$ on muotoa

$$m'_T(x) = m'_T(0) + x m''_T(0) + Err(x^2). \quad (33)$$

Jakamalla totaaliresponsi osiin huomataan, että $m'_T(0) = m'_f(0) + m'_N(0)$, jossa määritelmän mukaisesti $m'_f(0) = 0$. Tällöin totaaliresponsin derivaataksi pisteessä \hat{X}_0 saadaan $m'_T(\hat{X}_0) = m'_N(0) + x_0[m''_f(0) + m''_N(0)]$, josta reunan sijainniksi saadaan piste

$$\hat{X}_0 = -\frac{m'_N(0)}{m''_f(0) + m''_N(0)}. \quad (34)$$

Osoittajan termi $m'_N(0)$ on normaalisti jakautunut varianssilla, joka löytyy yhtälöstä 27. Nimittäjässä on osana termi $m''_N(0)$, joka ei ole merkitsevä kohinan ollessa pientä, mutta kohinan kasvaessa tätä termiä ei voida ohittaa. Toisen derivaatan suodattimille reunan sijainniksi saadaan yhtälöä 35 vastaavasti

$$\hat{X}_0 = -\frac{m_N(0)}{m'_f(0) + m'_N(0)}. \quad (35)$$

Koplowitz ja Greco [11] johtavat näiden yhtälöiden perusteella reunan sijainnin todennäköisyysjakauman suljetussa muodossa ja osoittavat, että jakauman varianssia ei ole olemassa ja huomauttavat, että tästä johtuen Cannyn mittoja reunantunnistukselle tulisi muokata.

3.3.1 Diskretisoinnin ja lineaarisen interpoloinnin vaikutus kohteen paikannusvirheisiin

Edellä esitetty Canny analyysi kohteen paikannusvirheille ei ota huomioon kuvan diskretisointia. Canny on kyllä tarkastellut diskretisoinnin vaikutusta erikseen [3], mutta tällöin lähtökohtana ei ole ollut alipikselitason kohteen paikannus. Kuten luvussa 3 on kerrottu, löytyy kirjallisuudesta myös joitain esimerkkejä alipikselin tarkkuuden menetelmien analysoinnista. Eräät näistä ovat lähellä seuraavaksi esitettävää menetelmää.

Esitämme seuraavaksi omatekoisen analyysin alipikselin tarkkuuden menetelmälle, jossa sijainti paikallistetaan lineaarisesti interpoloidusta diskreetistä aineistosta. Lähtökohtana on siis diskreetti kuva $I = \{F_i\}_{i=1}^{np}$, josta muodostetaan jatkuva-arvoinen sovite

$$\hat{F}(x) = \frac{F_i - F_{i-1}}{\Delta p}(x - x_{i-1}) + F_{i-1}, \quad \text{kun } x_{i-1} < x \leq x_i.$$

Kuten edellä, olemme kiinnostuneita selittämään, mikä on paikallistamisessa syntyvä virhe, kun virhe esitetään sijainnin harhan ja varianssin muodossa.

Tarkastelemme seuraavassa tilannetta, jossa kuvapikselit ovat kohinaisia havaintoja pisteissä x_i , $i = 1, \dots, n$, siten, että

$$F_i = f(x_i) + \varepsilon_i,$$

missä kohinatermit ε_i ovat toisistaan riippumattomia ja odotusarvoltaan nollia ($\mathbb{E}[\varepsilon_i] = 0$) ja niiden varianssi on $\text{Var}[\varepsilon] = \sigma^2$. Olettaen, että diskretisoinnin alkupiste x_a ei vaihdu, voisimme arvioida sovitteen $\hat{F}(x)$ ominaisuuksia sen odotusarvon $\mathbb{E}[\hat{F}(x)]$ ja varianssin $\text{Var}[\hat{F}(x)]$ avulla.

Lineaarisen interpoloinnin tuloksena saadun sovitteen odotusarvo on muotoa

$$\begin{aligned} \mathbb{E}\hat{F}(x) &= \mathbb{E}\left(\frac{F_i - F_{i-1}}{\Delta p}(x - x_{i-1}) + F_{i-1}\right) & (36) \\ &= \mathbb{E}\left(\frac{f(x_i) + \varepsilon_i - f(x_{i-1}) + \varepsilon_{i-1}}{\Delta p}(x - x_{i-1}) + f(x_{i-1}) + \varepsilon_{i-1}\right) \\ &= \frac{f(x_i) + \mathbb{E}\varepsilon_i - f(x_{i-1}) + \mathbb{E}\varepsilon_{i-1}}{\Delta p}(x - x_{i-1}) + f(x_{i-1}) + \mathbb{E}\varepsilon_{i-1} \\ &= \frac{f(x_i) - f(x_{i-1})}{\Delta p}(x - x_{i-1}) + f(x_{i-1}). \end{aligned}$$

Varianssi diskretisointipisteiden x_{i-1} ja x_i välillä on olettaen, että jäännöstermit $\{\varepsilon\}$ ovat

riippumattomia ja samoinjakautuneita, on muotoa

$$\begin{aligned}
\text{Var}(\hat{F}(x)) &= \mathbb{E}(\hat{F}(x) - \mathbb{E}\hat{F}(x))^2 & (37) \\
&= \mathbb{E}\left(\frac{f(x_i) + \varepsilon_i - f(x_{i-1}) - \varepsilon_{i-1}}{\Delta p}(x - x_{i-1}) + f(x_{i-1}) + \varepsilon_{i-1}\right. \\
&\quad \left. - \frac{f(x_i) - f(x_{i-1})}{\Delta p}(x - x_{i-1}) + f(x_{i-1})\right)^2 \\
&= \mathbb{E}\left(\frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta x}(x - x_{i-1}) + \varepsilon_{i-1}\right)^2 \\
&= \mathbb{E}\left(\frac{(\varepsilon_i - \varepsilon_{i-1})^2}{\Delta p^2}(x - x_{i-1})^2 + 2\varepsilon_{i-1}\frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p}(x - x_{i-1}) + \varepsilon_{i-1}^2\right) \\
&= \frac{\mathbb{E}(\varepsilon_i^2) - 2\mathbb{E}(\varepsilon_i\varepsilon_{i-1}) + \mathbb{E}(\varepsilon_{i-1}^2)}{\Delta p^2}(x - x_{i-1})^2 + 2\frac{\mathbb{E}(\varepsilon_{i-1}\varepsilon_i) - \mathbb{E}(\varepsilon_{i-1}^2)}{\Delta p}(x - x_{i-1}) + \mathbb{E}(\varepsilon_{i-1}^2) \\
&= \frac{2\sigma^2}{\Delta p^2}(x - x_{i-1})^2 - \frac{2\sigma^2}{\Delta p}(x - x_{i-1}) + \sigma^2 \\
&= 2\sigma^2\left(\frac{(x - x_{i-1})^2}{\Delta p^2} - \frac{x - x_{i-1}}{\Delta p} + \frac{1}{2}\right).
\end{aligned}$$

Kaava 38 osoittaa, että sovituksen varianssin suuruus voidaan esittää kuvan kohinan varianssin σ^2 ja niin kutsutun skaalatekijän $S(x)$ avulla muodossa

$$\text{Var}[\hat{F}(x)] = \sigma^2 \cdot S(x),$$

missä skaalatermi välillä x_{i-1} ja x_i on muotoa

$$S(x) = 2\left(\frac{(x - x_i)^2}{\Delta p^2} - \frac{(x - x_i)}{\Delta p} + \frac{1}{2}\right). \quad (38)$$

Termi saa maksiminsa diskretisointipisteissä x_{i-1} ja x_i ja pienenee x :n lähestyessä välin $[x_{i-1}, x_i]$ puoliväliä $\frac{x_i + x_{i-1}}{2}$, jossa $S(x)$ saa minimiarvon.

Olettaen edelleen, että käytettävissä on useita toistokuvia samasta kohteesta samalla diskretisoinnilla, ilmiöstä saatu aineisto on muotoa $D = \{I_x(t)\}_{t=1}^s$, missä s on toistojen määrä. Kun diskretisointi on vakio, voidaan kuvajoukko D korvata keskiarvoitetulla kuvalla

$$\bar{I} = \{\bar{F}_i\}_{i=1}^{n_p} = f(x_i) + \{\bar{\varepsilon}_i\}_{i=1}^{n_p},$$

missä $\bar{\varepsilon}_i = \frac{\sum_{t=1}^s \varepsilon_i(t)}{s}$ on keskiarvoitetujen kuvien kohina pisteessä i . Jos nämä poimitaan samalla tavalla toisistaan riippumatta tiedämme, että $\mathbb{E}[\bar{\varepsilon}_i] = \mathbb{E}[\varepsilon_i]$ ja $\text{Var}[\bar{\varepsilon}_i] = \frac{\text{Var}[\varepsilon_i]}{s}$. Näin ollen, usean kuvan toistostrategialla, kun diskretisointi on kiinteä, on sovituksen varianssi muotoa

$$\text{Var}[\hat{F}(x)] = \frac{\sigma^2}{s} S(x).$$

Reunafunktion estimaatti $[\hat{F}(x)](t)$ on laskettu kuvanoton yhteydessä saadusta diskreetistä aineistosta interpoloimalla. Integraali saadulle sovitteelle ja valitulle suotimelle lasketaan tutkittavalla välillä $z \in [-a, a]$, joka kiinnitetään suodatinta $g(z)$ rakennettaessa.

Konvoluutio sovitteelle ja suodattimelle on

$$m_T(x) = \int_{-a}^a g(z)\hat{F}(x-z)dz. \quad (39)$$

Seuraavissa tarkasteluissa oletamme, että suodattimen $g(z)$ leveys on asetettu kahden pikselin levyiseksi, $T = \Delta p$. Tällöin $g(z)$ osuu liu'utettaessa kahden tai kolmen diskretisointipisteiden välin alueelle. Tästä johtuen integraali tulee jakaa osiin, jolloin kokonaisresponsi on muotoa

$$m_T(x) = \int_{x-a}^{x_{i-1}} \hat{F}(z)g(x-z)dz + \int_{x_{i-1}}^{x_i} \hat{F}(z)g(x-z)dz + \int_{x_i}^{x+a} \hat{F}(z)g(x-z)dz. \quad (40)$$

Kuten yhtälö 20, myös tämä yhtälö jaetaan kahteen osaan: ilmiöstä johtuvaan osaan ja kohinaosaan. Tällöin kokonaisresponsi $m_T(x)$ lineaarisesti interpoloidulle aineistolle, jossa suotimen $g(z)$ leveys on $2\Delta p$, on muotoa

$$\begin{aligned} m_T(x) &= \underbrace{f_{i-2} \int_{x-a}^{x_{i-1}} g(z-x)dz + \frac{f_{i-1} - f_{i-2}}{\Delta p} \int_{x-a}^{x_{i-1}} (x - x_{i-2})g(z-x)dz}_{\subset m_f(x)} \\ &+ \underbrace{f_{i-1} \int_{x_{i-1}}^{x_i} g(z-x)dz + \frac{f_i - f_{i-1}}{\Delta p} \int_{x_{i-1}}^{x_i} (x - x_{i-1})g(z-x)dz}_{\subset m_f(x)} \\ &+ \underbrace{f_i \int_{x_i}^{x+a} g(z-x)dz + \frac{f_{i+1} - f_i}{\Delta p} \int_{x_i}^{x+a} (x - x_i)g(z-x)dz}_{\subset m_f(x)} \\ &+ \underbrace{\varepsilon_{i-2} \int_{x-a}^{x_{i-1}} g(z-x)dz + \frac{\varepsilon_{i-1} - \varepsilon_{i-2}}{\Delta p} \int_{x-a}^{x_{i-1}} (x - x_{i-2})g(z-x)dz}_{\subset m_N(x)} \\ &+ \underbrace{\varepsilon_{i-1} \int_{x_{i-1}}^{x_i} g(z-x)dz + \frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p} \int_{x_{i-1}}^{x_i} (x - x_{i-1})g(z-x)dz}_{\subset m_N(x)} \\ &+ \underbrace{\varepsilon_i \int_{x_i}^{x+a} g(z-x)dz + \frac{\varepsilon_{i+1} - \varepsilon_i}{\Delta p} \int_{x_i}^{x+a} (x - x_i)g(z-x)dz}_{\subset m_N(x)}. \end{aligned}$$

Integraalin lineaarisuuden perusteella kokonaisresponsi saadaan derivoinnin kannalta sel-

keämpään esitysmuotoon

$$\begin{aligned}
m_T(x) = & \underbrace{f_{i-2} \int_{x-a}^{x_{i-1}} g(z-x) dz + \frac{f_{i-1} - f_{i-2}}{\Delta p} \int_{x-a}^{x_{i-1}} xg(z-x) dz}_{\subset m_f(x)} \\
& - \underbrace{\frac{f_{i-1} - f_{i-2}}{\Delta p} x_{i-2} \int_{x-a}^{x_{i-1}} g(z-x) dz}_{\subset m_f(x)} \\
& + \underbrace{f_{i-1} \int_{x_{i-1}}^{x_i} g(z-x) dz + \frac{f_i - f_{i-1}}{\Delta p} \int_{x_{i-1}}^{x_i} xg(z-x) dz}_{\subset m_f(x)} \\
& - \underbrace{\frac{f_i - f_{i-1}}{\Delta p} x_{i-1} \int_{x_{i-1}}^{x_i} g(z-x) dz}_{\subset m_f(x)} \\
& + \underbrace{f_i \int_{x_i}^{x+a} g(z-x) dz + \frac{f_{i+1} - f_i}{\Delta p} \int_{x_i}^{x+a} xg(z-x) dz}_{\subset m_f(x)} \\
& - \underbrace{\frac{f_{i+1} - f_i}{\Delta p} x_i \int_{x_i}^{x+a} g(z-x) dz}_{\subset m_f(x)} \\
& + \underbrace{\varepsilon_{i-2} \int_{x-a}^{x_{i-1}} g(z-x) dz + \frac{\varepsilon_{i-1} - \varepsilon_{i-2}}{\Delta p} \int_{x-a}^{x_{i-1}} xg(z-x) dz}_{\subset m_N(x)} \\
& - \underbrace{\frac{\varepsilon_{i-1} - \varepsilon_{i-2}}{\Delta p} x_{i-2} \int_{x-a}^{x_{i-1}} g(z-x) dz}_{\subset m_N(x)} \\
& + \underbrace{\varepsilon_{i-1} \int_{x_{i-1}}^{x_i} g(z-x) dz + \frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p} \int_{x_{i-1}}^{x_i} xg(z-x) dz}_{\subset m_N(x)} \\
& - \underbrace{\frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p} x_{i-1} \int_{x_{i-1}}^{x_i} g(z-x) dz}_{\subset m_N(x)} \\
& + \underbrace{\varepsilon_i \int_{x_i}^{x+a} g(z-x) dz + \frac{\varepsilon_{i+1} - \varepsilon_i}{\Delta p} \int_{x_i}^{x+a} xg(z-x) dz}_{\subset m_N(x)} \\
& - \underbrace{\frac{\varepsilon_{i+1} - \varepsilon_i}{\Delta p} x_i \int_{x_i}^{x+a} g(z-x) dz}_{\subset m_N(x)}.
\end{aligned}$$

Seuraavissa tarkasteluissa oletamme, että diskretisointipisteet $\{x_i\}_{i=1}^{n_p}$ ovat tunnettuja jollain kiinteällä diskretisoinnin alkupisteellä x_a . Toinen oletus, joka helpottaa analyysin suorittamista, on reunan todellisen sijainnin asettaminen nolllaksi, $x_0^* = 0$. Reunan sijainnin saamme tekemällä Taylorin kehitelmän responssin ilmiöstä johtuvalle osalle reunan

todellisessa sijainnissa $x_0^* = 0$. Hajotelma on tällöin muotoa

$$m'_f(\hat{X}_0) \approx m'_f(0) + \hat{X}_0 m''_f(0),$$

jolloin reunan sijainnin approksimaatioksi saadaan

$$\hat{X}_0 \approx -\frac{m'_f(0) + m'_N(\hat{X}_0)}{m''_f(0)}.$$

Reunan sijainnin varianssi ratkaistaan laskemalla responssin kohinasta johtuvan osan odotusarvoinen neliö sekä ilmiöstä johtuvien osien neliölliset arvot. Oletetaan että

$$\mathbb{E}_{\hat{X}_0} \mathbb{E}[m'_N(\hat{X}_0)^2 | \hat{X}_0, x_a] \approx \mathbb{E}[m'_N(0)^2 | x_a]. \quad (41)$$

Yhtälöön 41 liittyy olennaisesti oletus reunan sijainnin \hat{X}_0 jakauman harhattomuudesta ja symmetrisyydestä sekä tieto siitä, että reunasuotimet ovat symmetrisiä. Tällöin varianssin approksimaatio on muotoa

$$\mathbb{E}[\hat{X}_0^2] \approx \frac{(m'_f(0)^2 + \mathbb{E}[m'_N(0)^2])}{m''_f(0)^2}. \quad (42)$$

Ilmiöön liittyvän responssin osan derivaatat $m'_f(0)$ ja $m''_f(0)$ ovat muotoa

$$\begin{aligned} m'_f(0) = & \frac{1}{\Delta p} \left[((\Delta p + x_{i-2})\bar{f}_{i-2} - x_{i-2}\bar{f}_{i-1}) \int_{-a}^{x_{i-1}} g'(z) dz \right. \\ & + (\bar{f}_{i-1} - \bar{f}_{i-2}) \int_{-a}^{x_{i-1}} g(z) dz \\ & + ((\Delta p + x_{i-1})\bar{f}_{i-1} - x_{i-1}\bar{f}_i) \int_{x_{i-1}}^{x_i} g'(z) dz \\ & + (\bar{f}_i - \bar{f}_{i-1}) \int_{x_{i-1}}^{x_i} g(z) dz \\ & + ((\Delta p + x_i)\bar{f}_i - x_i\bar{f}_{i+1}) \int_{x_i}^a g'(z) dz \\ & \left. + (\bar{f}_{i+1} - \bar{f}_i) \int_{x_i}^a g(z) dz \right] \end{aligned} \quad (43)$$

ja

$$\begin{aligned} m''_f(0) = & \frac{1}{\Delta p} \left[((\Delta p + x_{i-2})\bar{f}_{i-2} - x_{i-2}\bar{f}_{i-1}) \int_{-a}^{x_{i-1}} g''(z) dz \right. \\ & + (\bar{f}_{i-1} - \bar{f}_{i-2}) \int_{-a}^{x_{i-1}} g'(z) dz \\ & + ((\Delta p + x_{i-1})\bar{f}_{i-1} - x_{i-1}\bar{f}_i) \int_{x_{i-1}}^{x_i} g''(z) dz \\ & + (\bar{f}_i - \bar{f}_{i-1}) \int_{x_{i-1}}^{x_i} g'(z) dz \\ & + ((\Delta p + x_i)\bar{f}_i - x_i\bar{f}_{i+1}) \int_{x_i}^a g''(z) dz \\ & \left. + (\bar{f}_{i+1} - \bar{f}_i) \int_{x_i}^a g'(z) dz \right], \end{aligned} \quad (44)$$

jossa \bar{f}_k viittaa keskitettyyn funktion arvoon $\bar{f}_k = f_k - \bar{f}_k$, $k = i - 2, \dots, i + 1$, jolla poistetaan skaalauksen vaikutus tunnistukseen. Kohinasta johtuvan osan odotusarvoinen neliö on muotoa

$$\begin{aligned}
\mathbb{E}_{\hat{X}_0} \mathbb{E}[m'_N(\hat{X}_0)^2 | \hat{X}_0, x_a] &= \frac{2\sigma^2}{\Delta p^2 s} \left[\left(\frac{\Delta p^2}{2} + \Delta p x_{i-2} + x_{i-2}^2 \right) \left[\int_{-a}^{x_{i-1}} g'(z) dz \right]^2 \right. & (45) \\
&+ \left(\frac{\Delta p^2}{2} + \Delta p x_{i-1} + x_{i-1}^2 \right) \left[\int_{x_{i-1}}^{x_i} g'(z) dz \right]^2 \\
&+ \left(\frac{\Delta p^2}{2} + \Delta p x_i + x_i^2 \right) \left[\int_{x_i}^a g'(z) dz \right]^2 \\
&- (\Delta p + 2x_{i-2}) \int_{-a}^{x_{i-1}} g'(z) dz \int_{-a}^{x_{i-1}} g(z) dz \\
&- (\Delta p + 2x_{i-1}) \int_{x_{i-1}}^{x_i} g'(z) dz \int_{x_{i-1}}^{x_i} g(z) dz \\
&- (\Delta p + 2x_i) \int_{x_i}^a g'(z) dz \int_{x_i}^a g(z) dz \\
&- (\Delta p + x_{i-1}) x_{i-2} \int_{-a}^{x_{i-1}} g'(z) dz \int_{x_{i-1}}^{x_i} g'(z) dz \\
&- (\Delta p + x_i) x_{i-1} \int_{x_{i-1}}^{x_i} g'(z) dz \int_{x_i}^a g'(z) dz \\
&+ (\Delta p + x_{i-1}) \int_{-a}^{x_{i-1}} g(z) dz \int_{x_{i-1}}^{x_i} g'(z) dz \\
&+ (\Delta p + x_i) \int_{x_{i-1}}^{x_i} g(z) dz \int_{x_i}^a g'(z) dz \\
&+ x_{i-2} \int_{-a}^{x_{i-1}} g'(z) dz \int_{x_{i-1}}^{x_i} g(z) dz \\
&+ x_{i-1} \int_{x_{i-1}}^{x_i} g'(z) dz \int_{x_i}^a g(z) dz \\
&+ \int_{-a}^{x_{i-1}} g(z) dz \int_{x_{i-1}}^{x_i} g(z) dz \\
&+ \int_{x_{i-1}}^{x_i} g(z) dz \int_{x_i}^a g(z) dz \\
&+ \left[\int_{-a}^{x_{i-1}} g(z) dz \right]^2 \\
&+ \left[\int_{x_{i-1}}^{x_i} g(z) dz \right]^2 \\
&+ \left. \left[\int_{x_i}^a g(z) dz \right]^2 \right],
\end{aligned}$$

jossa s on kappaleesta otettujen toistokuvien lukumäärä. Ottaen huomioon tehdyt oletukset, yksiulotteisen reunan sijainnin varianssi on ratkaistavissa sijoittamalla yhtälöt (43), (44) ja (45) kaavaan (42). Kyseisten yhtälöiden johtaminen on esitetty tarkemmin liitteessä C.

Seuraavaksi tarkastelemme teoreettista varianssi sigmoidisen reunan tilanteessa. Oletam-

me että kohinaton reuna on muotoa

$$f(x) = \frac{1}{1 + e^{-(x+0.5)+0.5}}. \quad (46)$$

Oletetaan edelleen, että tunnustimena on templaattisuodin, joka on muotoa

$$g(z) = \frac{1}{1 + e^{(z-0.5)+0.5}} - 0.5, \quad z \in [-4, 4].$$

Reunasta on havaittu neljä toistokuvaa $\{I_x(t)\}_{t=1}^4$ kohinan varianssilla $\sigma^2 = 0.01$ ja pikselivälillä $\Delta p = 4$.

Koska diskretisointipisteet oletetaan tunnetuiksi, asetetaan kaavassa esiintyvät diskretisointipisteet pisteiksi $x_{i-2} = -6$, $x_{i-1} = -2$, $x_i = 2$ ja $x_{i+1} = 6$. Kaavan 42 ratkaisemiseen tarvittavat integraalit ovat tällöin seuraavat: Sijoittamalla saadut arvot yhtälöihin

$$\begin{array}{lll} \int_{-a}^{x_{i-1}} g(z)dz \approx 0.891 & \int_{x_{i-1}}^{x_i} g(z)dz \approx 0 & \int_{x_i}^a g(z)dz \approx -0.891 \\ \int_{-a}^{x_{i-1}} g'(z)dz \approx 0.101 & \int_{x_{i-1}}^{x_i} g'(z)dz \approx 0.764 & \int_{x_i}^a g'(z)dz \approx 0.101 \\ \int_{-a}^{x_{i-1}} g''(z)dz \approx -0.087 & \int_{x_{i-1}}^{x_i} g''(z)dz \approx 0 & \int_{x_i}^a g''(z)dz \approx 0.087 \end{array}$$

43, 44 ja 45 saadaan reunan sijainnin varianssi

$$Var(\hat{X}_0) \approx 0.075,$$

jolloin reunan sijainnin keskihajonta on

$$sd(\hat{X}_0) = \sqrt{Var(\hat{X}_0)} \approx 0.274,$$

mikä tarkoittaa sitä, että reunan paikannuksessa päästään $\frac{sd(\hat{X}_0)}{\Delta p} \approx \frac{1}{15}$ alipikselitason tarkkuuteen.

Diskretisointipisteet ja reunan sijainnin varianssi ovat riippuvia, mistä johtuen tarkkailemme varianssia myös tilanteessa, jossa reunan todellinen sijainti on yksi diskretisointipisteistä. Tällöin diskretisointipisteet ovat $x_{i-2} = -8$, $x_{i-1} = -4$, $x_i = 0$ ja $x_{i+1} = 4$. Erona edelliseen on se, että tässä tilanteessa suodin osuu vain kahdelle integrointivälille, joten reunan sijainnin varianssin ratkaisemiseksi tarvitaan vain seuraavat integraalit: Reunan

$$\begin{array}{ll} \int_{-4}^0 g(z)dz \approx 1.325 & \int_0^4 g(z)dz \approx -1.325 \\ \int_{-4}^0 g'(z)dz \approx 0.482 & \int_0^4 g'(z)dz \approx 0.482 \\ \int_{-4}^0 g''(z)dz \approx -0.232 & \int_0^4 g''(z)dz \approx 0.232. \end{array}$$

sijainnin varianssi saa tällöin arvon

$$\text{Var}(\hat{X}_0) \approx 0.449,$$

ja reunan sijainnin keskihajonta on

$$\text{sd}(\hat{X}_0) \approx 0.670,$$

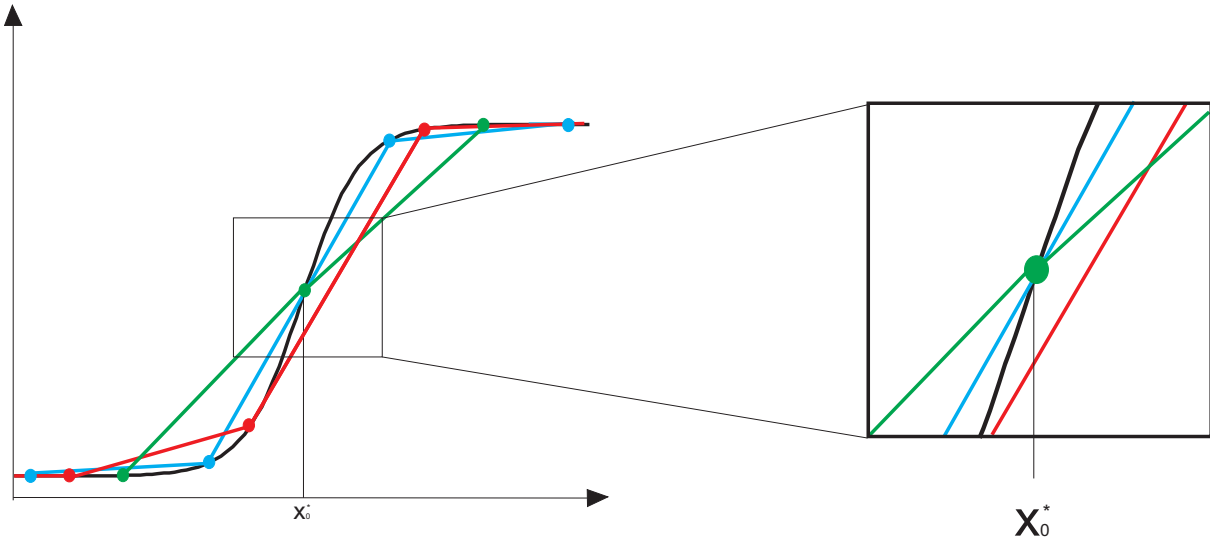
jolloin tunnistustarkkuudeksi jää noin $\frac{\text{sd}(\hat{X}_0)}{\Delta p} \approx \frac{1}{6}$ pikseliä.

Edellä olevista tarkasteluista huomaamme, että diskretisoinnin alkupisteellä on todella suuri vaikutus reunantunnistuksen onnistumisessa. Näyttäisi siltä, että tapauksessa, jossa reunan todellinen sijainti x_0^* on yksi diskretisointipisteistä x_i , menetelmä on harhaisempi ja silloin $m'_f(0)$ saa suurempia arvoja.

3.3.2 Harha lineaarisen interpoloinnin alipikselintarkkuudenmenetelmissä

Interpolointiin perustuviin alipikselintarkkuuden menetelmiin liittyy mahdollisesti harhatekijöitä, kuten edellä olevista lineaarista interpolointia käyttävien reunan sijainnin varianssin tarkasteluista voidaan huomata. Myös muut menetelmät kuten kvadraattinen interpolointi synnyttävät harhaa, mikä on todettu muun muassa Rockettin [20] ja Ramdanin, ym. [17] töissä. Pedersini, Sarti ja Tubaro [16] ehdottavat ja analysoivat menetelmän, joka parantaa reunan sijainnin havaitsemista alipikselintarkkuudella kompensoimalla reunan sijainnin harhan systemaattista osaa. Menetelmä on käytännöllinen erityisesti niissä sovelluksissa, joissa reunan sijainnin tarkkuus on tärkeää ja reunan lokalisatiiovirhe on suurimmillaan. Kyseinen kompensatiometodi ei riipu käytettävästä alipikselinestimointimenetelmästä. Pedersinin, Sartin ja Tubaron menetelmällä on mahdollista poistaa systemaattinen harha. On kuitenkin mahdollista, että harha ei ole selitettävissä, vaan se syntyy satunnaisesta diskretisoinnin alkupisteestä. Työssä tutkittavassa asetelmassa tilanne on juuri tällainen eikä harhan poistaminen ole kaikissa tilanteissa mahdollista millään menetelmällä.

Oletetaan, että käytössä on yksi kuva kappaleesta. Tällöin kappaleen diskretisointipisteen x_a osuessa paikkaan, jossa $\min \{x_i\}_{i=1}^{n_p} - x_0^*$ on tiettyä suuruusluokkaa, tulisi lineaarisella interpoloinnilla suoritetuista soviteista $\hat{I}(t)$ harhaisia reunan todellisessa sijainnissa x_0^* . Tämä on nähtävissä kuvassa 24. Kuvassa lineaarinen interpolointi on suoritettu kohinatto-



Kuva 24: Kohinattomaan sigmoidiseen reunaan on sovitettu kolme eri lineaarista sovitetta. Kun diskretisointi pisteet x_i osuvat tiettyyn kohtaan tulee sovitteesta harhainen reunan todellisen sijainnin x_0^* suhteen.

masta ilmiöstä havaittujen diskretisointipisteiden avulla. Diskretisointiväli Δp on kaikissa tapauksissa sama ollen sitä luokkaa, että reuna nousee noin kahden pikselin matkalla, $\frac{\Delta e}{\Delta p} \approx 2$.

Diskretisoinnin satunnaista alkupistettä X_a tarkkaillaan kolmella eri arvolla. Huomataan, että kun reunan todellista sijaintia lähinnä olevan diskretisointipisteen etäisyys reunan todellisesta sijainnista on nolla tai puolet pikselivälistä, $\min(\{X_i(t)\}_{i=1}^{n_p} - x_0^*) = 0 \vee \frac{\Delta p}{2}$, on lineaarinen interpolointi harhaton. Tämän etäisyyden ollessa neljäsosa pikselivälistä, $\min(\{X_i(t)\}_{i=1}^{n_p} - x_0^*) = \frac{\Delta p}{4}$, on lineaarinen interpolointi kuitenkin harhainen. Kyseistä harhaa on mahdoton poistaa, sillä emme voi tietää mihin kohtaan pikselit osuvat suhteessa reunan todelliseen sijaintiin x_0^* . Tällöin harha ei ole systemaattista, eikä siten poistettavissa.

Kyseisestä ilmiöstä johtuen on oletus diskretisoinnin alkupisteen X_a satunnaisuudesta mielenkiintoinen, kun käytössä on useita kuvia yhdestä kappaleesta. Voidaan ajatella, että jokainen kappaleesta havaittavien kuvien $I(t)$ pikselit $\{X_i(t)\}_{i=1}^{n_p}$ havaittaisiin samoissa paikoissa

$$\{X_i(t)\}_{i=1}^{n_p} = \{X_i(u)\}_{i=1}^{n_p}, \quad t \neq u,$$

jolloin kappaleesta otettavan kuvan diskretisoinnin alkupisteet $\{X_a(t)\}_{t=1}^s$ olisivat kiinteitä arvoja x_a , jolloin harhaa ei pystyttäisi poistamaan. Jos taas kappaleesta havaittavien

kuvien $I(t)$ pikselit $\{X_i(t)\}_{i=1}^{n_p}$ havaittaisiin satunnaisella diskretisoinnin alkupisteellä

$$\{X_i(t)\}_{i=1}^{n_p} \neq \{X_i(u)\}_{i=1}^{n_p}, \quad t \neq u,$$

saataisiin aineistoa kerättyä useammista pisteistä, jolloin systemaattista, satunnaisen alkupisteen luomaa harhaa ei syntyisi. Tämä osoitetaan empiirisesti kappaleessa 4.3.

4 Työssä tutkittavien menetelmien evaluointi empiirisin kein

Työn empiirisessä evaluoinnissa kohteen paikannuksen onnistumista tarkastellaan simuloitinkokeiden avulla. Tarkastelun kohteena ovat reunansuodatukseen ja templaattisovituksen perustuvat alipikselintarkkuuden menetelmät, joissa diskreetti kuva-aineisto interpoloidaan suurempaan tarkkuuteen. Tutkimusongelmaa varten on luotu simulaattori, joka jäljittelee reaali maailman tilannetta, jossa tehtävänä on havaita kohteen sijainti mahdollisimman tarkasti diskreetistä kohinaisesta kuvasta I . Simulaatiokokeet on toteutettu käyttämällä tilastollista laskentaohjelmaa R.

Tunnistettavina kohteina ovat yksiulotteinen ja kaksiulotteinen sigmoidinen reuna. Tämän lisäksi testaamme, mihin tarkkuuteen on mahdollista päästä, kun kaksiulotteinen sigmoidinen reuna havaitaan mittausstrategialla, jossa eräiden kohteiden sijainnit määräävät miten toisten kohteiden sijaintia haetaan. Luvussa pyrimme vastaamaan seuraaviin kysymyksiin:

- Mihin tarkkuuteen kohteen sijainnin havaitsemisessa on mahdollista päästä interpolointiin perustuvilla menetelmillä ja tarkasteltavilla toistostrategioilla?
- Mikä edellämainituista reunasuotimista havaitsee reunan sijainnin tarkimmin tarkasteltavissa olosuhteissa?
- Miten olosuhteiden muutokset vaikuttavat kohteiden sijainnin havaittavuuteen?
- Mikä vaikutus diskretisoinnin alkupisteen satunnaisuudella on kohteen sijainnin tunnistamisessa?
- Onko toistostrategialla merkitystä kohteen sijainnin havaittavuuteen tarkasteltavissa olosuhteissa?

Virhetekijöitä pyritään tarkastelemaan luvusta 3 esitettyjen tunnuslukujen avulla, sekä mediaanivirheellä yksiulotteisessa tapauksessa. Laskettavat tunnusluvut ovat pikselivirhe, keskihajonta, harhan itseisarvo ja mediaanivirhe. Tunnusluvuista kaikki, mediaanivirhettä lukuunottamatta, liittyvät jollain tavalla keskineliövirheeseen (MSE), jota voi käyttää paikallistamisvirheen kokonaismittana.

Tunnusluvut lasketaan satunnaistoistojen avulla. Tällä menetelmällä saadaan jonkin laskettavan tunnusluvun

$$\theta = \int \phi(x)f(x)dx$$

empiirinen estimaattori muodossa

$$\hat{\theta} = \frac{1}{n} \sum_{i=1}^n \phi(x),$$

missä n on toistokokeiden $\phi(x)$ lukumäärä. Havaittujen reunojen sijaintien \hat{X}_0 odotusarvon estimaattoriksi saadaan

$$\mathbb{E}\hat{X}_0 \approx \bar{x}_0 = \frac{\sum_{i=1}^N \hat{x}_0(i)}{N}.$$

Otoskeskihajonnan estimaattori \hat{sd} lasketaan puolestaan kaavasta

$$\hat{sd}(\hat{X}_0) \approx \sqrt{\frac{\sum_{i=1}^N (\hat{x}_0(i) - \bar{x}_0)^2}{N}},$$

jonka avulla myös pikselivirhe lasketaan jakamalla kohteen sijainnin keskihajonnan ja harhan estimaattorien neliöiden summa pikselivälillä Δp muodossa

$$\hat{\alpha} = \frac{\sqrt{\hat{sd}(\hat{X}_0)^2 + \widehat{Bias}^2}}{\Delta p}.$$

Sijaintien harha lasketaan havaittujen kohteiden odotusarvoisen sijainnin erotuksesta kohteiden todelliseen sijaintiin

$$\widehat{Bias} = \left(\frac{\sum_{i=1}^N \hat{x}_0(i)}{N} - x_0^* \right).$$

Mediaanivirhe eroaa muista laskettavista tunnusluvuista siinä, että se ei perustu harhavarianssi-hajotelmaan. Mediaanivirhe mittaa havaittujen reunojen sijainnin mediaanin eroa reunan todellisesta sijainnista

$$\widehat{ME} = Median\{\hat{x}_0(i)\} - x_0^*.$$

Jos havaittujen reunojen mediaani eroaa niiden keskiarvosta, viittaa se jakauman mahdolliseen vinouteen tai poikkeavien havaintojen mahdollisuuteen.

Kaksiulotteisessa tapauksessa jätämme yksiulotteisen tapauksen tunnusluvuista mediaanivirheen pois ja tutkimme ainoastaan harhaa ja piksetason keskihajontaa. Harha on odotusarvoisesti havaittavan sijainnin $\mathbb{E}(\hat{X}_0, \hat{Y}_0)$ etäisyys kohteen todellisesta sijainnista (x_0^*, y_0^*) , joka lasketaan kaavasta

$$|\widehat{Bias}| = \sqrt{\left(\frac{\sum_{i=1}^N \hat{x}_0(i)}{N} - x_0^* \right)^2 + \left(\frac{\sum_{i=1}^N \hat{y}_0(i)}{N} - y_0^* \right)^2}.$$

Hajontaa tarkastelemme siis ainoastaan pikselitasolla, sillä tällöin luvuista tulee helposti tulkittavia ja sisällöllisesti mielenkiintoisia alipikselintarkkuuden suhteen. Kuten luvussa 3 on todettu, ei pikselivirhe yleisty suoraan yksiulotteisesta tapauksesta kaksiulotteiseen. Tästä johtuen pikselihajonta $\hat{sd}(\alpha_{xy})$ lasketaan kaavan

$$\hat{sd}(\alpha_{xy}) = \frac{\sum_{i=1}^N \sqrt{\left(\hat{x}_0(i) - \frac{\sum_{i=1}^N \hat{x}_0(i)}{N}\right)^2 + \left(\hat{y}_0(i) - \frac{\sum_{i=1}^N \hat{y}_0(i)}{N}\right)^2}}{\Delta p \cdot N}$$

mukaisesti. x - ja y -akselien suuntaiset tunnusluvut $Bias_x$ ja $Bias_y$ lasketaan kuten yksiulotteisessa tapauksessa. Pikselihajonnat ovat puolestaan muotoa $\hat{sd}(\alpha_x) = \frac{\hat{sd}(\hat{X}_0)}{\Delta p}$ ja $\hat{sd}(\alpha_y) = \frac{\hat{sd}(\hat{Y}_0)}{\Delta p}$. Keskihajonta lasketaan yksinkertaisesti kertomalla pikselihajonta pikselivälillä, $\hat{sd}((\hat{X}_0, \hat{Y}_0)) = \hat{sd}(\alpha_{xy}) \cdot \Delta p$

Seuraavaksi tarkastelemme kohteiden, jotka oletamme tyypillisiksi reaali maailmassa esiintyviksi kohteiksi, sijaintien tunnistamista simulointien avulla. Alilukuihin liittyvät kohteet, olosuhteet ja niissä toteutettavat menetelmät on esitetty taulukossa 2.

Luku	Kohde	Tunnistin	Toistostrategia	Interpolointi	SNR	$\frac{\Delta \epsilon}{\Delta p}$
4.1	1D sigmoid	Templaatti	Keskiarvo reunatoistoista	Yksiulotteinen	10	2
		DoG	Keskiarvo pikselitoistoista	lineaarinen	20	4
		LoG	Mediaani reunatoistoista		50	8
			Mediaani pikselitoistoista	toistokuvia 4		12
4.2	2D sigmoid	Templaatti	Keskiarvo reunatoistoista	Yksi- tai	10	2
		LoG	Keskiarvo pikselitoistoista	kaksisuuntainen	20	4
				lineaarinen	50	8
						12
4.3	1D sigmoid	Templaatti	Keskiarvo reunatoistoista	lineaarinen	10	2
		LoG	toistokuvia 10	spline regressio spline		
4.4	Ruuvitorni 2D sigmoid	Templaatti	Ei toistoja	Yksi- tai	10	2
				kaksisuuntainen		4
				lineaarinen		8

Taulukko 2: Luvussa tarkasteltavat olosuhteet ja sijainnin paikallistamismenetelmät.

4.1 Yksiulotteisen sigmoidisen reunan sijainnin havaitseminen

Tyypillinen kohde, jonka sijainti kuvasta halutaan havaita, on reuna. Kuten edelläolevasta käy ilmi, ne saattavat olla kynnsreunan tyyppisiä, tai niihin saattaa sisältyä gaussisia komponentteja. Tutkimuksessamme oletamme, että luonnossa esiintyvät reunat eivät todellisuudessa ole muodoltaan kynnsreunoja, vaan niiden muodoissa esiintyy kaarevuutta. Tästä johtuen valitsimme tutkittavaksi ilmiöksi sigmoidisen reunan, jonka muotoa monet reaali maailman reunat mielestämme vastaavat. Tutkittava yksiulotteinen ideaalinen ilmiö on muotoa

$$f(x) = \frac{1}{1 + e^{-\alpha x + \beta}}, \quad \alpha = 1, \beta = 0.5, x \in [-12, 12]. \quad (47)$$

Ilmiöön sisältyy tällöin kohdefunktio $r(x) \subset f(x)$, joka on rajoitettu välille $x \in [-5.5, 6.5]$, ja jonka sijainti määrätään pisteeseen $x_0^* = 0.5$. Kohteen sijainnin piste on samalla funktion gradientin maksimi $\max(|\nabla_x r(x)|)$. Kohteen sijainti valittiin gradientin maksimiksi johtuen siitä, että se voidaan havaita yksikäsitteisesti myös ensimmäistä, tai toista derivaattaa approksimoivilla suodattimilla. Tämä mahdollistaa templaattisovituksen ja reuansuodattimien paremmuuden vertailun.

Ideaalisen ilmiön $f(x)$ parametrit, jotka auttavat määrittelemään reunan ominaisuudet, ovat reunan voimakkuus ja sen leveys. Ilmiön voimakkuus on

$$\max[f(x)] - \min[f(x)] = \Delta I = 1$$

ja reunan leveys on mitta, joka kertoo välin pituuden jolla ilmiö nousee minimistä maksimiin. Leveys mitataan suoralla, joka saa kulmakertoimensa reunan gradientista kohteen todellisessa sijainnissa x_0^* . Reunan leveys on tällöin muotoa

$$\Delta e = \frac{\max f(x) - \min f(x)}{\max |f'(x)|} = \frac{\Delta I}{\nabla_x f(x|_{x_0^*, \beta})} = \frac{(1 + e^0)^2}{0.5e^0} = 8.$$

Pikselin leveytenä Δp tarkastellaan neljää eri vaihtoehtoa:

$$\Delta p = 4, 2, 1, \text{ tai } \frac{2}{3}.$$

Tällöin reunan leveyden suhde pikselin leveyteen saa arvot

$$\frac{\Delta e}{\Delta p} = 2, 4, 8, \text{ tai } 12.$$

Toinen reunan sijainnin tunnistamisen tarkkuuteen vaikuttava muuttuja on signaali-kohinasuhde

$$SNR = \frac{\Delta I}{\sigma},$$

missä σ on kohinan ε keskihajonta. Signaali-kohinasuhde saa simuloinneissa kolme eri arvoa:

$$SNR = 10, 20, \text{ tai } 50.$$

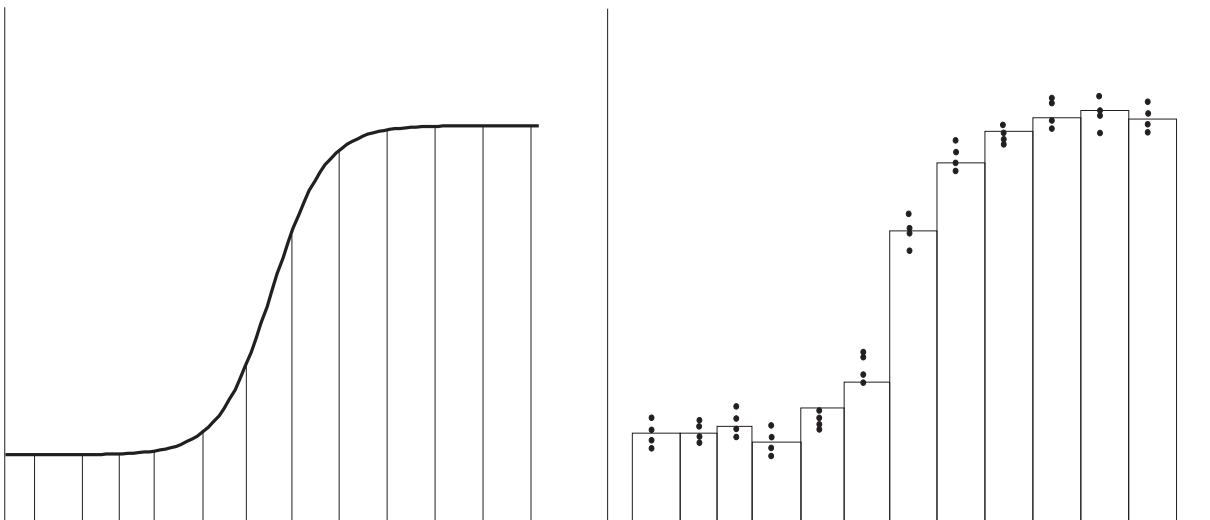
Kappaleesta havaittu t :s kuva on muotoa

$$I_x(t) = \{f(x_i) + \varepsilon_i(t)\}_{i=1}^{n_p},$$

jossa $\varepsilon_i \sim N(0, \sigma^2)$. Kuvanotossa yhdestä kappaleesta otetaan neljä kuvaa kiinteällä alkupisteellä $s = 4$, eli aineisto on muotoa

$$D = \{I_x(t)\}_{t=1}^4 = \{f(x_a + i \cdot \Delta p) + \varepsilon_i(t)\}_{i=1, t=1}^{n_p, 4}$$

jossa x_a on yhden kappaleen alkupiste. Interpolointimenetelmänä käytetään lineaarista interpolointia. Toistostrategia toteutetaan joko laskemalla mediaani tai keskiarvo reunatoistoista tai pikselitoistoista.



Kuva 25: Vasemmalla: Yksiulotteinen ilmiö $f(x) = \frac{1}{1+e^{-x+0.5}}$. Oikealla: Ilmiöstä havaittu aineisto $D = \{I_x(t)\}_{t=1}^4$.

Kuvassa 25 on pyritty kuvaamaan aineiston muodostumista, jossa kappaleesta on otettu neljä toistokuvaa kiinteällä diskretisoinnin alkupisteellä x_a . Huomaamme, että kaikki havainnot tehdään tällöin samoissa x -akselin pisteissä $x_i \quad i = 1, \dots, n_p$.

Reunantunnistusoperaattorina käytetään kolmea eri menetelmää:

1. Templaattisovitusta, eli templaattia, jolloin tunnistimena on kohdefunktio itsessään

$$g(z) = -\frac{1}{1+e^{-\alpha z+\beta}}$$

2. Gaussisen funktion derivaattaa (Derivative of Gaussian)

$$g(z) = ze^{\frac{-z^2}{2\sigma^2}},$$

jossa $\sigma^2 = 1.0$

3. Gaussisen funktion toista derivaattaa (Laplacian of Gaussian)

$$g(z) = -\frac{z^2 - \sigma^2}{\sigma^4} e^{\frac{-z^2}{2\sigma^2}},$$

jossa $\sigma = 1.6$

Simuloinnit tapahtuivat toistamalla reunantunnistusoperaatio kaksi kaksisataa kertaa, $N = 200$, kaikilla kombinaatioilla edellämainituista muuttujista. Yhden kappaleen, josta toistokuvat suoritetaan, alkupiste on satunnainen

$$X_a \sim U[(-12), (-12 + \Delta p)].$$

Pääasiallisena tarkastelun kohteena on eri suodattimien toiminta tarkasteltavissa olosuhteissa, sekä eri toistostrategioiden vaikutus tuloksiin. Tämä koe toistettiin neljälle erilaiselle mittausstrategialle ja kolmelle reunantunnistimelle, jolloin koetaulukoita syntyy kaikkiaan kaksitoista.

Käytetyt mittausstrategiat hyödyntävät useampaa samasta kohteesta otettua kuvaa, mitä voidaan hyödyntää kohinan vaikutusten eliminointiin. Luvussa 1.5 esitettyjen periaatteiden mukaisesti lopullinen sijainti \hat{X}_0 saadaan strategiasta riippuen joko

a) reunatoistojen mediaanina

$$\hat{x}_0 = Med(\{\hat{X}_0(t)\}_{t=1}^s),$$

jossa jokaisesta kuvasta $I(t)$ etsitään reuna $\hat{X}_0(t)$, ja lopullinen reunan sijainti on näiden mediaani;

b) reunatoistojen keskiarvona

$$\hat{X}_0 = \frac{1}{s} \sum_{t=1}^s \hat{X}_0(t),$$

jossa periaate on muuten sama kuin edellä, mutta lopullinen reunan sijainti lasketaan toistojen keskiarvona mediaanin sijaan.

- c) pikselitoistojen mediaanina, jossa reunan sijainti \hat{x}_0 etsitään vain yhdestä (mediaani) kuvasta,

$$I^{med} = \{F_i^{med} | F_i^{med} = Med(\{F_i(t)\}_{t=1}^s)\},$$

jonka kuvapisteen lasketaan toistokuvien $I(t)$, $t = 1, \dots, s$ mediaanista, tai

- d) pikselitoistojen keskiarvona, jossa edellisen kohdan tapaan käytetään vain yhtä kuvaa

$$\bar{I} = \{\bar{F}_i | \bar{F}_i = \frac{1}{s} \sum_{t=1}^s F_i(t)\},$$

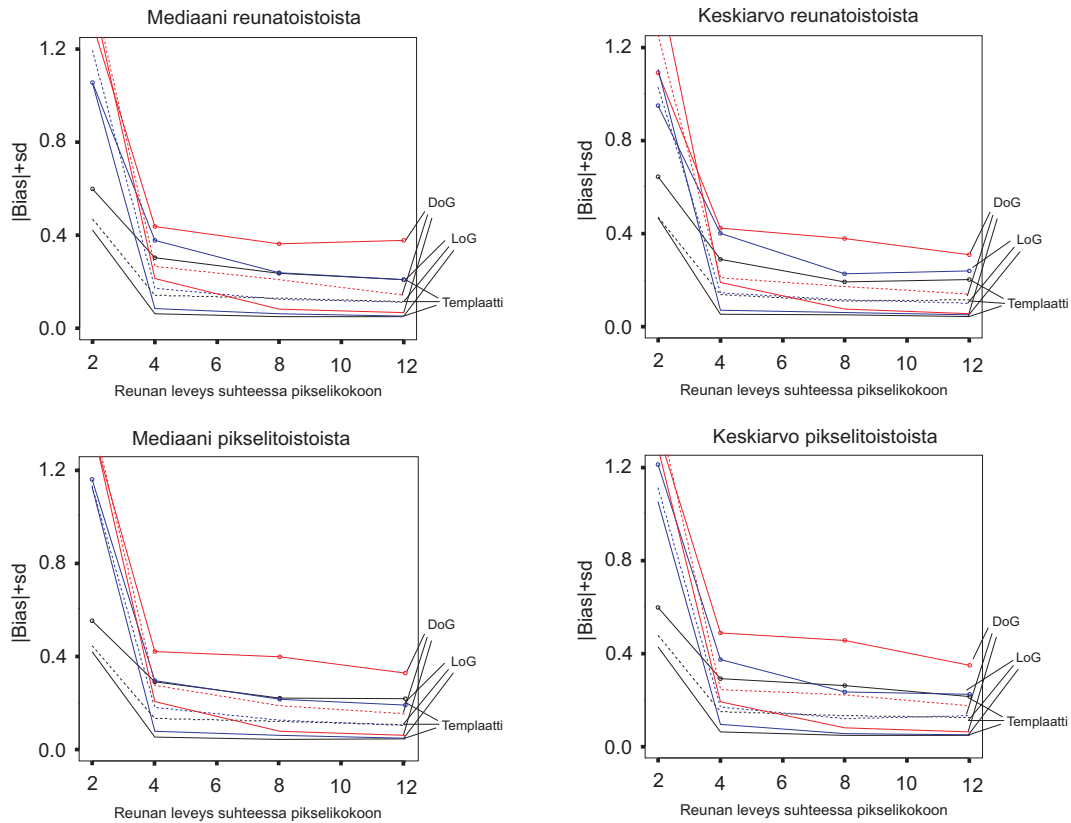
joka on toistokuvien $t = 1, \dots, s$ keskiarvo.

4.1.1 Yhteenveto yksiulotteisen sigmoidisen reunan havaitsemisesta

Seuraavassa on vertailtu toistostrategian merkitystä sekä templaattisovituksen, DoG-suotimen (Derivative of Gaussian) ja LoG-suotimen (Laplacian of Gaussian) toimivuutta reunantunnistimena yksiulotteisen sigmoidisen reunan tapauksessa. Seuraavissa testeissä parhaimmat tulokset (kaikilla mittareilla) saatiin käyttämällä templaattisovitusta. Muihin menetelmiin verrattuna templaatin etuna on pikselivirheen ja harhan vähäisyys. Ainoastaan LoG-suodin pystyi haastamaan templaattisovituksen tapauksessa, jossa reuna nousi kahdentoista pikselin mitalla, $\frac{\Delta e}{\Delta p} = 12$ ja ero korostui signaali-kohina-suhteen kasvaessa. Lisätutkimukset LoG-suotimen ja templaatin käyttäytymisen eroista ovat paikallaan.

Yksiulotteisen sigmoidisen funktion paikallistamisen tulokset on esitetty kokonaisuudessaan taulukkomuodossa liitteessä A. Koska taulukoiden tulkinta ja vertailu on vaikeaa, on samat tulokset esitetty graafisesti kuvassa 26. Kuvassa esitetään MSE :n arvot (pystyakselilla) suhdeluvun $\frac{\Delta e}{\Delta p}$:n (vaaka-akselilla) suhteen. Signaali-kohinasuhde SNR sai kolmea eri arvoa. Kullakin filterillä keskimääräisesti korkeimmalla oleva käyrä edustaa pienintä signaali-kohinasuhdetta $SNR = 10$ (yhtenäinen viiva), kun taas alin edustaa suurinta $SNR = 50$ (ympyröity viiva). Kuten kuvasta huomataan, näyttäisi templaatti toimivan parhaiten kaikilla menetelmillä. Erityisenä etuna on myös sen robustisuus verrattuna DoG-suotimeen ja LoG-suotimeen suurilla pikseliväleillä ($\frac{\Delta e}{\Delta p} = 2$). Huomattavaa on myös se, että toistostrategian yhteydessä mediaani näyttäisi käyttäytyvän keskiarvoa systemaattisemmin hajontojen kasvaessa.

Toisaalta, käytetyllä koestrategialla kohina ei aiheuta poikkeavia havaintoja (outlier).



Kuva 26: Pikselivälin ja signaali-kohinasuhteen ($SNR = 50$ ympyröity viiva, $SNR = 20$ katkoviiva, $SNR = 10$ yhtenäinen viiva) vaikutus harhan ja keskihajonnan arvoihin templaattisovitusta, DoG-suodinta ja LoG-suodinta käyttäen.

Tästä johtuen toistostrategioissa, joissa käytettiin keskiarvoa saatiin numeerisesti paremmat tulokset, kuin mediaania käytettäessä. Reunatoistojen ja pikselitoistojen välille ei saatu tuloksellisesti suuria eroja parhaiden tulosten suhteen suuria eroja. Tämä tukeekin luvussa 3.3.1 tehtyä huomiota, että nämä erilaiset toistostrategiat antavat odotusarvoisesti saman tuloksen.

Edellä esitetyt tulokset tukevat Canny'n [3] johtopäätöstä, jonka mukaan paras reunantunnistin olisi reunafunktio itsessään. LoG-suotimen ääriolosuhteissa saaman edun voisi olettaa johtuvan siitä, että toisen derivaatan filttäreillä reuna löytyy filterin ja havaitun aineiston konvoluution responssin nollakohdasta, mikä on kohinaisessa tapauksessa yksikäsitteisempi, kuin maksimi. DoG-suodinta käyttämällä saadut tulokset olivat järjestään heikommat, kuin muilla tarkastelluilla reunantunnistimilla.

Kuten jo työn alkupuolella on mainittu, todellisuudessa tarkkailtava ilmiö on kaksiulotteinen toteutus kolmiulotteisesta kappaleesta kuvaussuunnassa. Havaittuna aineistona on kaksiulotteinen digitaalinen kuva, josta kohteita pyritään paikantamaan mahdollisimman

suurella tarkkuudella. Seuraavaksi siirrymme tarkasteluissa edellä tarkasteltua yksiulotteista sigmoidista funktiota vastaavaan kaksiulotteiseen ilmiöön. Koska edellä olevista tarkasteluista huomaamme, että toistostragian avulla päästään parempiin tuloksiin keskiarvoa käyttämällä, joten jätämme mediaanien tarkastelut pois kaksiulotteisesta tapauksesta. Myöskään DoG-suodinta ei tarkastella sen yksiulotteisessa tapauksessa antamien heikkojen tulosten johdosta.

4.2 Kaksiulotteisen sigmoidisen reunan sijainnin havaitseminen

Konenäköön perustuvassa laadunvarmistuksen aineistona on kaksiulotteinen kuva. Kuten edellä, tarkastelun kohteena on reuna. Seuraavassa olemme valinneet tarkasteltavaksi kohdefunktioksi sigmoidisen reunan, koska oletamme niiden muodon vastaavan reaalimaailmassa esiintyvien reunojen muotoa.

Formaalisti kaksiulotteinen sigmoidinen reunafunktio on muotoa

$$f(x, y) = \frac{1}{1 + e^{-\beta_1 x + \beta_2 y + \beta_0}}, \quad x, y \in [-16, 16] \quad (48)$$

jossa uutena parametrina on suuntaparametri β_2 . Tutkittaessa ilmiötä oletamme, että parametrit ovat tunnettuja: $\beta_1 = 1$, $\beta_2 = 0.1$ ja $\beta_0 = 0.5$. Kyseessä on siis pääosin y -suuntainen reuna, jossa reunan oikea sijainti, $f(x, y) = \frac{1}{2}$, noudattaa yhtälöä $x = 0.1y + 0.5$. Kuvasta etsittävä kohdefunktio on muodoltaan sama kuin ideaalinen ilmiö ja sen sijainti reaalimaailmassa määritellään suhteessa pisteeseen $(z, q) = (0, 0)$, jossa $r(z, q) = \frac{1}{2}$.

Kohteesta havaittava kuva ajanottohetkellä t on muotoa

$$I_{xy}(t) = \{F_{ij}(t)\}_{i=1, j=1}^{n_p, n_p} = \{f(x_i, y_j) + \varepsilon_{ij}(t)\}_{i=1, j=1}^{n_p, n_p},$$

missä $\{\varepsilon_{ij}(t)\}_{i=1, j=1}^{n_p, n_p}$ ovat toisistaan riippumattomia noudattaen jakaumaa $N(0, \sigma^2)$.

Pikselikoon määrittävinä x - ja y -akselin diskretisointiväleinä Δp tarkastellaan neljää eri vaihtoehtoa:

$$\Delta p = 4, 2, 1, \text{ tai } \frac{2}{3}.$$

Diskretisointivälit ovat kaikissa tapauksissa yhtä suuret molemmilla akseleilla. Tällöin x -akselin suuntaisessa reunantunnistamisessa oleellinen reunan leveyden suhde pikselin

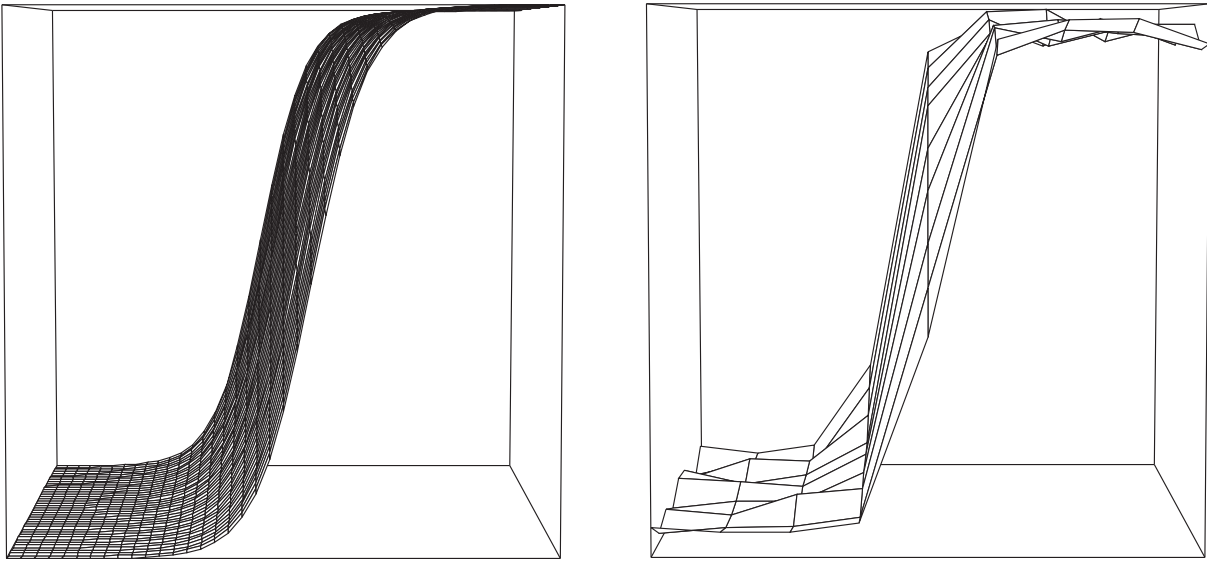
leveyteen saa samat arvot kuin yksiulotteisessa tapauksessa, eli

$$\frac{\Delta e}{\Delta p} = 2, 4, 8, \text{ tai } 12.$$

Signaali-kohina-suhde $SNR = \frac{\Delta I}{\sigma^2}$, jossa σ^2 on kohinan ε varianssi. Signaali-kohina-suhdetta testataan kolmella eri arvolla:

$$SNR = 10, 20, \text{ tai } 50.$$

Kuvassa 27 vasemmalla on kohinaton ilmiö $f(x, y)$. Oikealla samassa kuvassa on ilmiöstä



Kuva 27: Kuvassa esitettyinä tarkasteltava kaksiulotteinen reunafunktio $f(x, y) = \frac{1}{1+e^{-\beta_1 x + \beta_2 y + \beta_0}}$ (vasemmalla) ja siitä havaittu kuva $I_{xy}(t) = \{F_{ij}(t)\}_{i=1}^{n_p} = \{f(x_i, y_j) + \varepsilon_{ij}(t)\}_{i=1}^{n_p}$ (oikealla)

havaittu kohinainen kuva $I_{xy}(t)$.

Kuvanotossa yhdestä kappaleesta otetaan neljä kuvaa ($s = 4$) kiinteillä diskretisoinnin alkupisteillä $(X_a(t), Y_a(t)) = (x_a, y_a)$. Interpolointimenetelmänä käytetään lineaarista interpolointia. Toistostrategioita on yksiulotteiseen esimerkkiin 4.1 verrattuna vähemmän, sillä mediaania käyttävät strategiat on jätetty tarkasteluista pois. Jäljelle jäävät strategiat ovat keskiarvo reuna- tai pikselitoistoista.

Yksiulotteisen tarkastelun perusteella olemme jättäneet kaikissa tapauksissa heikoimmin toimineen reunantunnistimen, eli DoG-suotimen pois. Tästä johtuen ainoastaan templaattisovituksen ja LoG-suotimen antamat tulokset on esiteltyinä kaksiulotteisessa tapauksessa. Formaalisti reunantunnistimet ovat seuraavanlaiset

1. Templaattisovitus

$$g(z, q) = -\frac{1}{1 + e^{-(z+0.5)+0.1q+0.5}} - 0.5, \quad z, q \in [-10, 10],$$

2. Laplacian of Gaussian

$$g(z, q) = -\frac{1}{\pi\sigma^4} \frac{1 - (z^2 + q^2)}{2\sigma^2} e^{-\frac{q^2+z^2}{2\sigma^2}},$$

jossa $\sigma = 1.6$.

Simuloinneissa reunantunnistusoperaatio toistetaan kaksi sataa kertaa, $N = 200$, kaikilla kombinaatioilla edellämäinuituista muuttujista. Diskretisointien alkupisteet ovat simuloinneissa satunnaisia,

$$X_a \sim U[(-12), (-12 + \Delta p)],$$
$$Y_a \sim N(-12, \frac{\Delta p^2}{10}).$$

Merkittävä ero yksiulotteiseen tapaukseen on se, että kaksiulotteisessa tapauksessa reunan sijaintia voidaan etsiä kahden muuttujan suhteen. Tosin käytännön simulointien helpottamiseksi oletamme, että reunantunnistus tapahtuu vain x -akselin suuntaisesti. Koska kohteen sijainti y -akselilla on ennalta määritelty, olemme kiinnostuneita pääosin x -akselilla saavutettavaan alipikselintarkkuudesta, joka luonnollisesti riippuu Y_0 :n hajonnasta reunan ollessa vino.

Oletus perustuu ajatukseen, että reunan y -sijainti on etsitty jo aiemmin jollain sopivalta tavalla. Reaalimaailmassa tämä tehdään esimerkiksi etsimällä reunaa kappaleen puolivälistä, joka taas estimoidaan muiden kohteiden avulla. Tällöin y -sijaintiin liittyy myös epävarmuus, jota simuloinneissa mallinnetaan poimimalla y -sijainnit normaalijakaumasta varianssilla $\sigma^2 = \frac{\Delta p^2}{10}$.

Kuten luvussa 2.2.1 on esitetty, ei lineaarista interpolointia voida yksikäsitteisesti yleistää kaksiulotteiseen tapaukseen. Tästä johtuen sovellamme kahta yksiulotteiseen lineaariseen interpolointiin perustuvaa menetelmää:

- **Kaksisuuntainen lineaarinen interpolointi**, jossa interpolointi suoritetaan molempien akselien (rivit ja sarakkeet) suuntaisesti yhtälöiden 7 ja 8 mukaisesti.

- **Yksisuuntainen lineaarinen interpolointi**, jossa interpolaatio suoritetaan ai-noastaan kohteen etsintäsuunnassa yhtälön 6 mukaisesti kaikille kuvamatriisin I_{xy} riveille, tai sarakkeille riippuen kohteen etsintäsuunnasta.

Yksisuuntaisen interpoloinnin etuna kaksisuuntaiseen verrattuna on sen laskennallinen keveys. Tämä saattaa olla merkittävä etu, sillä on mahdollista, että syntyy tilanteita, joissa laskenta-aika tai -kapasiteetti ei ole riittävä kaksisuuntaisen interpoloinnin suorittamiseen.

Kaksisuuntaisessa lineaarisessa interpoloinnissa yksiulotteinen lineaarinen interpolointi suoritetaan kaksiulotteiselle kuvamatriisille I_{xy} ensin toisen akselin (rivi/sarake) suuntaisesti valitulla interpolointivälillä Δsim . Tuloksena saatu $n_{sim} \times n_p$ kokoinen matriisi interpoloidaan myös toisen akselin (sarake/rivi) suuntaisesti, jonka jälkeen interpoloitu aineisto on kokoa $n_{sim} \times n_{sim}$. Käytettäessä LoG-suodinta sekä templaattia reunantunnistukseen, tulee myös nämä pystyä esittämään samoilla diskretisointiväleillä Δsim .

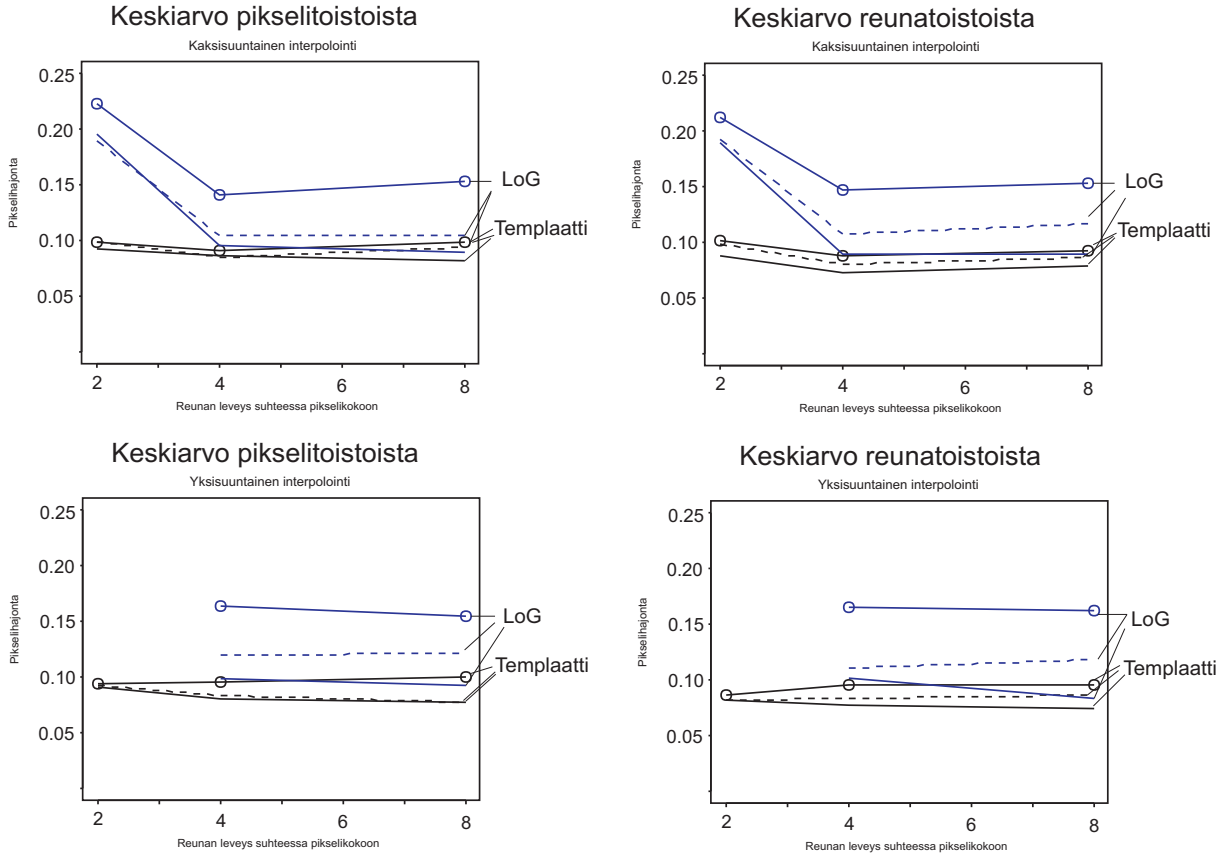
Tilanteessa, jossa laaduntarkastus tapahtuu on-line tilassa, mittauksen nopeus on oleellinen ominaisuus. On mahdollista, että konenäköjärjestelmän laskentakapasiteetti ei ole riittävä kaksisuuntaisen interpoloinnin suorittamiseen tarvittavassa ajassa. Jotta tarkastus olisi mahdollista, joudutaan interpoloinnin tiheyttä pienentämään. Koska oletamme, että suunta, jossa reunaa etsimme, on tiedossa pystymme suorittamaan lineaarisen interpoloinnin vain etsittävään suuntaan. Tällöin interpoloidun aineiston \hat{I}_{xy} kooksi muodostuu $n_p \times n_{sim}$ tai $n_{sim} \times n_p$, mikä aiheuttaa laskennan keventymisen.

4.2.1 Yhteenveto kaksiulotteisen sigmoidisen reunan havaitsemisesta

Simulointikokeiden tuloksia vertaamalla pystymme ottamaan kantaa siihen, millaista interpolointia, tunnistinta ja toistostrategiaa tulisi käyttää parhaaseen tarkkuuteen pääsemiseksi. Kun oletetaan, että laskenta-ajalla ja systemaattisella harhalla ei ole merkitystä, voidaan menetelmien paremmuusjärjestystä arvioida suoraan saaduilla pikselihajonnan arvoilla.

Kuten luvussa 4.1, myös kaksiulotteisessa tapauksessa liitteen B taulukkojen tulokset pyritään esittämään graafisesti yksinkertaisemmassa muodossa. Keskitymme tarkasteluissa

ainoastaan pikselihajonnan $\hat{sd}(\alpha_{xy})$ kuvaamiseen johtuen y -akselin suuntaisen pikselihajonnan kiinnittämisestä kymmenesosaan $\hat{sd}(\alpha_y) = 0.1$. Tämän vuoksi tulokset ovat paremmin verrannollisia muilla pikselivälin arvoilla saatuihin tuloksiin. Pikselihajonnan arvot, käyttäen erilaisia suotimia, reunan leveyksiä suhteessa pikselinleveyteen, sekä erilaisia signaali-kohinasuhteita, on esitetty kuvassa 28. Kuvasta on nähtävissä, että templaatin ta-



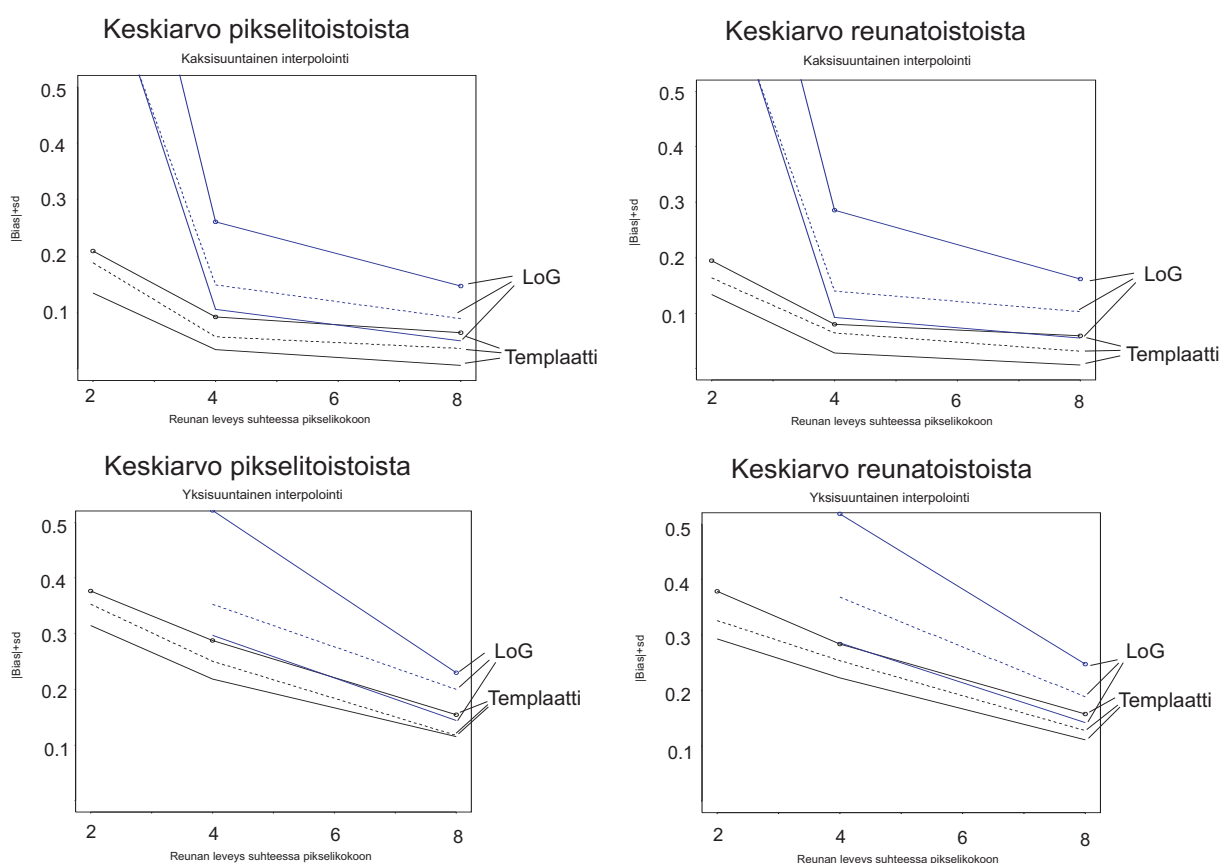
Kuva 28: Templaatilla ja Laplacian of Gaussianilla havaittavat kaksisuuntaisen sigmoidisen reunan sijainnin pikselihajonnat $\hat{sd}(\alpha_{xy})$ signaali-kohinasuhde ($SNR = 50$ ympyröity viiva; $SNR = 20$ katkoviiva; $SNR = 10$ yhtenäinen viiva) ja reunan leveyden suhde pikselikokoon $\frac{\Delta e}{\Delta p}$ huomioonottaen.

pauksessa pikselihajonnan arvot ovat luokkaa $\hat{sd}(\alpha_{xy}) \approx 0.1$ kaikilla signaali-kohinasuhteen arvoilla. Kuten aiemmin todettiin, ovat ne siis ennalta kiinnitetyn y -akselin suuntaisen pikselihajonnan ($sd(\alpha_y) = 0.1$) suuruusluokkaa. Mielenkiintoinen yksityiskohta on kohinan merkityksen vähäisyys, minkä voidaan ajatella johtuvan siitä, että templaatti käyttää hyväkseen suurta joukkoa havaintopisteitä, jotka silottavat kohinan vaikutusta huomattavasti.

LoG-suotimella kohinan vaikutus näyttäisi suuremmalta, mikä aiheuttaakin huomattavasti suuremmat pikselihajonnan arvot. Tämän voi olettaa johtuvan siitä, että Laplacian of

Gaussianissa suuri osa filteristä antaa responsseina nollaa. Tämä johtaa siihen, etteivät LoG-suotimen kaikki pisteet tasoita kohinaa toisin kuin templaatin tapauksessa. Toinen tärkeä huomio on, että yksisuuntaisen lineaarisen interpoloinnin tapauksessa Laplacian of Gaussian ei anna tulosta pikselivälin ollessa suuri ($\frac{\Delta e}{\Delta p} = 2$).

Koska reunan sijainnin haku tapahtui x -akselin suuntaisesti ja kokonaispikselihajonnan $\hat{sd}(\alpha_{xy})$ olivat y -akselin suuntaisen hajonnan dominoimia ei edellisestä kuvasta havaita kaikkea tarvitsemaamme informaatiota. Tästä johtuen tarkastelemmekin x -akselin suuntaisia tuloksia sen suuntaisten harhan $Bias_x$ ja hajonnan \hat{sd}_x avulla. x -akselin suuntaiset keskineliövirheet MSE_x ovatkin graafisesti esitettyinä kuvassa 29. Nämä arvot on esitetty reaaliakselin skaalalla, jotta vertaaminen yksiulotteiseen tapaukseen olisi luontevampaa.



Kuva 29: Templaattisovituksella ja LoG-suotimella havaittavat kaksiulotteisen sigmoidisen reunan sijainnin x -akselin suuntaiset keskineliövirheet $|Bias_x| + (\hat{sd}_x)$ signaalikohinasuhde ($SNR = 50$ viiva, jossa pisteet päällä; $SNR = 20$ katkoviiva; $SNR = 10$ yhtenäinen viiva) ja reunan leveyden suhde pikselikokoon $\frac{\Delta e}{\Delta p}$ huomioonottaen.

Pikselihajonnan arvoista huomataan siis, että templaattisovitus toimii kaikissa tutkituissa olosuhteissa Laplacian of Gaussiania paremmin. Laplacian of Gaussianin tuottamat

x -akselin suuntaiset pikselihajonnan arvot $\hat{sd}(\alpha_x)$ olivat useissa tapauksissa moninkertaisia verrattuna templaatin tuottamiin. Laplacian of Gaussiania ei tulisi kuitenkaan täysin jättää huomiotta, sillä osa saaduista tuloksista viittaa siihen, että se saavuttaisi templaattia tarkkuudessa pikselivälin Δp pienentyessä. Kaikesta huolimatta templaattisovitus oli vertailtavista menetelmistä sekä tarkkuudeltaan että kohinaherkkyydeltään parempi ja se toimi kaikilla diskretisointiväleillä LoG-suotimen romahtettua reunan leveyden ja pikselivälin suhteen ollessa $\frac{\Delta e}{\Delta p} = 2$ yksisuuntaisen lineaarisen interpoloinnin tapauksessa. Laplacian of Gaussianin toimimattomuus kyseisessä tapauksessa johtui sen muodon katoamisesta pikselivälin Δp käymisestä liian suureksi.

Vertailtaessa yksi- ja kaksisuuntaista lineaarista interpolointia huomattiin, että niiden tuottamissa pikselihajonnan arvoissa ei juurikaan ole eroa, jos käytettävä menetelmä oli muuten samanlainen. Yksisuuntaisen lineaarisen interpoloinnin heikkoutena kaksisuuntaiseen verrattuna oli sen harhaisuus. Toinen merkittävä ero oli Laplacian of Gaussianin romahtaminen, kun $\frac{\Delta e}{\Delta p} = 2$. Se että yksi- ja kaksisuuntaisen lineaarisen interpoloinnin välillä on ollut eroa, viittaa siihen, että jos etsittävä suunta on tunnettu, vain siihen suuntaan suoritettava interpolointi lisää sijainnin tunnistuksen tarkkuutta. Jos interpolointi suoritetaan myös toiseen suuntaan, heikentää se ainoastaan laskennan tehokkuutta. Interpolointi tulee siis näiden tarkastelujen perusteella suorittaa vain yhteen suuntaan, jos kohteen etsimissuunta on tunnettu. Kaksisuuntainen lineaarinen interpolointi on kuitenkin tarpeellinen menetelmä, sillä jos etsittävä suunta ei ole tiedossa, tarvitaan informaatiota molempiin suuntiin.

Keskiarvon sisältävien toistostrategioiden välille ei yksisuuntaista lineaarista interpolointia ja templaattia käyttämällä saatu eroja, vaan niiden paremmuus vaihteli olosuhteiden mukaan. Luvussa 3.3.1 esitetyt erot hajonnan väleillä jäivät niin pieniksi, että niitä ei ollut mahdollista havaita simulaatioiden määrällä $N = 200$. Tästä johtuen voidaan päätellä, että erot ovat niin pieniä, että toistostrategian valinnan ratkaisee laskentatehokkuus tai jokin muu ulkoinen tekijä. Jos etukäteen on tiedossa että kuvaan sisältyy poikkeavan suuria virhelähteitä, tulisi keskiarvon sijaan käyttää mediaania, sillä se toimii keskiarvoa paremmin poikkeavien havaintojen hallitsemiseen.

Parhaimpiin tuloksiin päästiin siis käyttämällä yksisuuntaista lineaarista interpolointia, templaattisovitusta ja keskiarvoa, joko pikseli-, tai reunatoistoista. Tarkasteltaessa liitteitä B.5 ja B.7 nähdään, että signaali-kohina-suhteen ollessa 10 päästiin x -akselinsuuntaisissa

pikselihajonnassa $\hat{sd}(\alpha_x)$ arvoihin, joka vaihtelivat 0.042:sta 0.059:ään y -akselin suuntaisen pikselihajonnan $\hat{sd}(\alpha_y)$ ollessa kiinnitettynä noin 0.1:een. Tämän perusteella todetaan kyseisellä signaali-kohina-suhteella päästävän noin $\frac{1}{20}$ pikselin tarkkuuteen x -akselin suuntaisesti.

Signaali-kohina-suhteen noustessa 20:een x -akselin suuntaisen pikselihajonnan arvot laskevat kohinan vaikutuksen pienentyttyä. Kyseisellä signaali-kohina-suhteen arvoilla y -akselin suuntaisen pikselihajonnan ollessa annettu ($\hat{sd}(\alpha_y) \approx 0.1$), x -akselin suuntaisessa pikselihajonnassa $\hat{sd}(\alpha_x)$ päästiin arvoihin, jotka vaihtelivat 0.017:stä 0.033:een. Samalla tavalla kuin edellä voidaan siis sanoa, että $\hat{sd}(\alpha_x) \approx 0.025$, mikä merkitsee noin $\frac{1}{40}$ pikselin tarkkuutta x -akselin suuntaisesti. Tämä tukee luvussa 3.3.1 esitettävää tulosta, jonka mukaan kohinan varianssin vaikutus skaalaa reunan sijainnin varianssin. Kaksinkertaistettaessa kohinan määrä myös hajonta kaksinkertaistuu.

Tarkasteluihin on sisällytetty myös hyvin suuri signaali-kohina-suhde $SNR = 50$. Tällöin kohina on hyvin pientä verrattuna intensiteetin minimin ja maksimin väliseen erotukseen ΔI . Tässä tapauksessa $\hat{sd}(\alpha_x)$ saa arvoja väliltä $[0.009, 0.028]$. Näistä arvoista huomataan, että pikselihajonta jatkaa pienenemistään kohinan vähentyessä, ja suurin osa näistä arvoista oli luokkaa 0.01. Tämä merkitsee sitä, että kun y -akselin suuntainen pikselihajonta $\hat{sd}(\alpha_y) \approx 0.1$ on kiinnitetty päästään x -akselin suunnassa lähes $\frac{1}{100}$ pikselin tarkkuuteen kyseisellä signaali-kohina-suhteen arvolla.

4.3 Kiinteään diskretisoinnin alkupisteeseen liittyvä harha

Kuten aiemmin luvuissa 2.2.1 ja 3.3.1 on todettu, lineaarinen interpolointi on menetelmänä mahdollisesti harhainen kohteen sijainnin x_0^* ympäristössä. Kuten luvun 3.3.1 kuvasta 24 on huomattavissa, riippuu harha lähimmän diskretisointipisteen etäisyydestä reunan todelliseen sijaintiin $\min(|X_i - x_0^*|)$.

Tunnistaaksemme harhan suuruusluokan tarkastelemme harhan ja hajonnan käyttäytymistä sadassa tasavälein asetetussa diskretisoinnin alkupisteessä x_a pikselivälin Δp sisällä. Näissä pisteissä kuvaus toistetaan kymmenen kertaa $s = 10$. Toistokuvien diskretisoinnin alkupistettä tarkastellaan sekä kiinteänä, että satunnaisena. Jos toistokuvien diskretisoin-

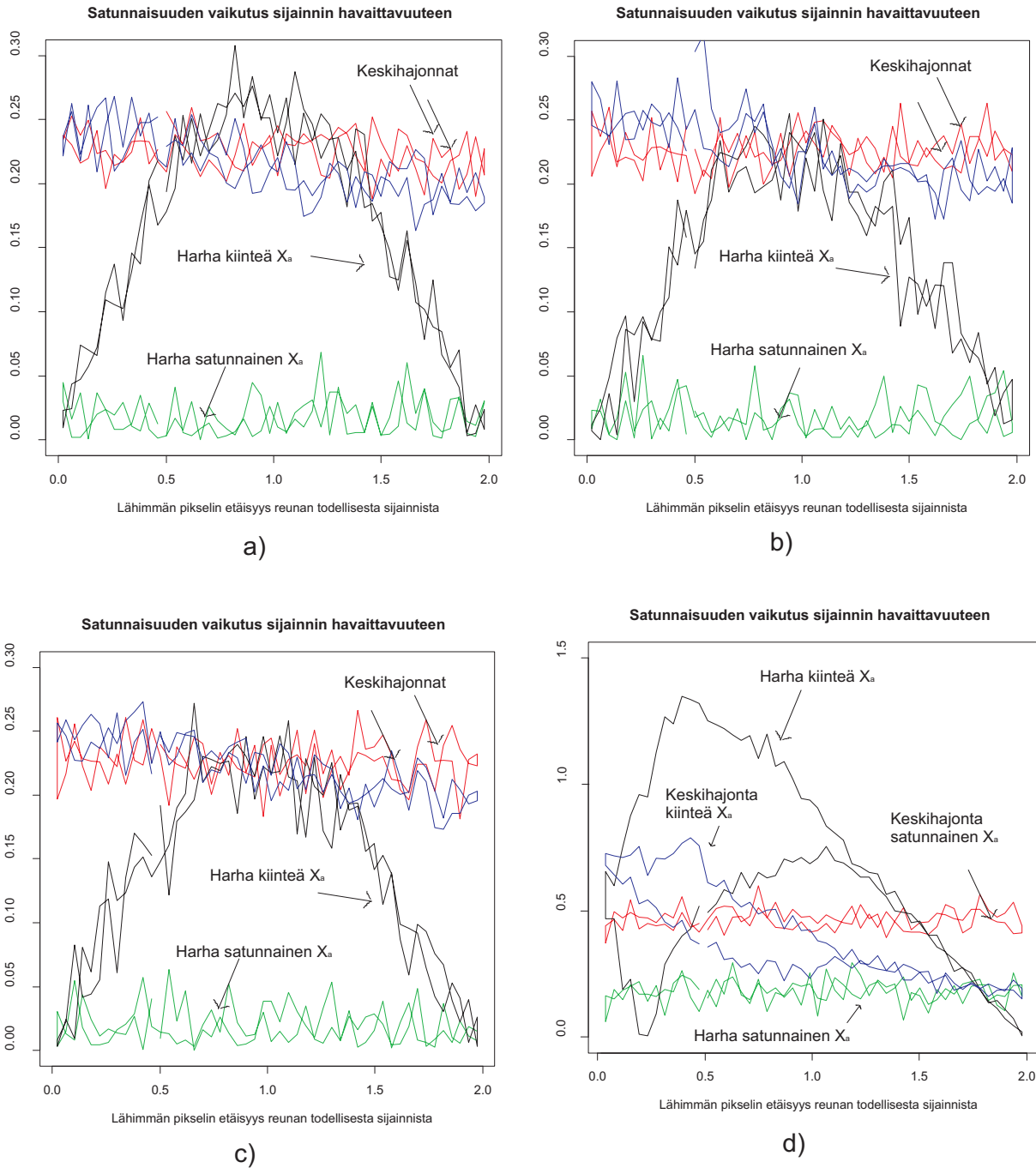
nin alkupiste on satunnainen, ei diskretisoinnin alkupisteen osumiskohdalla ole merkitystä, jos ne kaikki noudattavat samaa tasaista jakaumaa.

Toistostrategiana asetelmassa toimii keskiarvo reunatoistoista, sillä toistokuvien diskretisoinnin alkupisteen ollessa satunnainen ei pikselitoistoille ole yksikäsitteistä tulkintaa pikselien sijaintien osuessa eri kohtiin reaaliakselia. Koska templaattisovitus toimi tunnistimista parhaiten, tarkastelemme pääosin sen synnyttämää harhaa. Interpolointimenetelminä templaattisovituksen kanssa vertailemme lineaarista interpolointia, spline-interpolointia ja regressiospline -sovitusta. Vertailun vuoksi tutkimme myös miten LoG-suotimen ja lineaarisen interpoloinnin tapauksessa. Tulokset näistä asetelmista on esitetty kuvassa 30. Kuvissa on esitetty, miten paikannuksessa syntyvä harha $Bias(\hat{X}_0|X_a)$ riippuvat diskretisoinnin alkupisteistä, kun

- i) x_a on kiinteä
- ii) X_a on satunnainen.

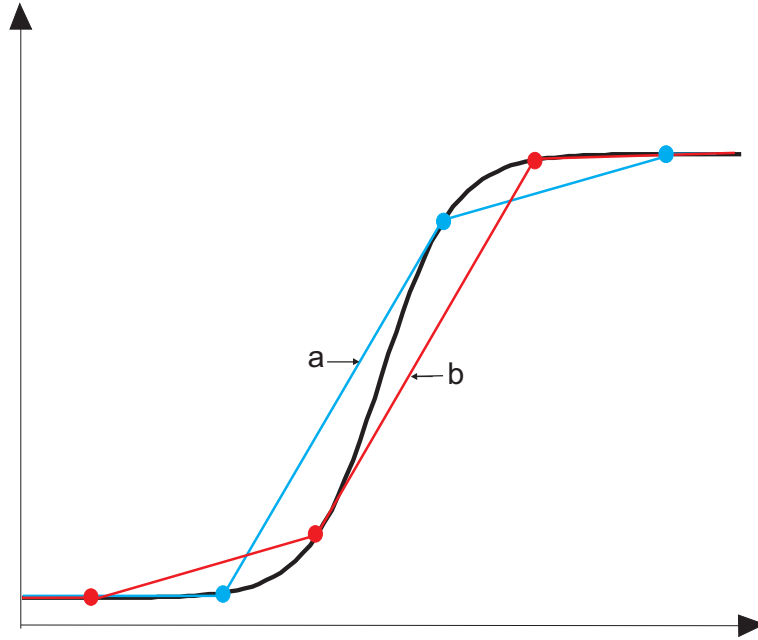
Reunan noustessa kahden pikselin matkalla, $\frac{\Delta e}{\Delta p} = 2$, syntyy keskineliövirheen arvoihin eroja riippuen siitä, mihin kohtaan kappaleen diskretisoinnin alkupiste x_a osuu suhteessa reunan todelliseen sijaintiin x_0^* . Käytettäessä templaattisovitusta ja kiinteää toistokuvan $I(t)$ diskretisoinnin alkupistettä on havaittu reunan sijainti harhainen (musta) kaikilla tarkasteltavilla interpolointimenetelmillä ääritapauksia $\min\{|x_i - x_0^*|\} \neq 0$ ja $\min\{|x_i - x_0^*|\} \neq \frac{\Delta p}{2}$ lukuunottamatta, mikä on nähtävissä kuvan 30 a, b ja c. Harha on suurimmillaan etäisyyden ollessa ääritapausten puolella välissä, $\min\{|x_i - x_0^*|\} = \frac{\Delta p}{4}$. Harha (musta) näyttäisi kasvavan nopeasti ääritapausten ympäristössä ja tasaantuvan lähestyttäessä pistettä, jossa $\min\{|x_i - x_0^*|\} = \frac{\Delta p}{4}$.

Toinen mielenkiintoinen huomio käytettäessä tunnistukseen templaattisovitusta on hajonnan (sininen) käyttäytyminen toistokuvan kiinteän diskretisoinnin alkupisteen tapauksessa. Keskihajonta näyttäisi pienentyvän lähimmän diskretisointipisteen etäisyyden kasvaessa. Tämän voidaan olettaa johtuvan siitä, että kun $\min\{|x_i - x_0^*|\} = \frac{\Delta p}{4}$ sovitteen \hat{I} muoto rakennetaan pienemmästä määrästä pikseleitä kuin muissa tapauksissa. Tällä tavoin kyseisen kohteen muotoa pystytään jäljittelemään tarkemmin, sillä reunan nousuun vaikuttaa yksi pikseli vähemmän, jolloin muoto on symmetrisempi.



Kuva 30: $\frac{\Delta e}{\Delta p} = 2$: a) Lineaarinen interpolointi ja templaatti, b) Spline-interpolointi ja templaatti, c) Regressiospline ja templaatti, d) Lineaarinen interpolointi ja LoG.

Toisen tunnistimen, eli LoG-suotimen ja lineaarisen interpoloinnin antamat tulokset (kuva 30d) olivat mielenkiintoiset. Kuvan 30 kohdan d skaala on huomattavasti suurempi. Harha ja hajonta ovat moninkertaiset, mikä kertoo sijainnin tunnistuksen heikkoudesta templaattisovitukseen verrattuna. Vain silloin kun $\min\{|x_i - x_0^*|\} = \frac{\Delta p}{2}$ päästään LoG-suotimella templaattisovitukseen tasolle. Koska LoG-suodin perustuu derivaatan havaitsemiseen, olivat tulokset epäsymmetrisiä riippuen siitä, kummalle puolelle reunan todellista sijaintia x_0^* lähin diskretisointipiste sijoittui. Tämä johtuu siitä, että muissa kuin ääritapauksissa $\min\{|x_i - x_0^*|\} \neq 0$ ja $\min\{|x_i - x_0^*|\} \neq \frac{\Delta p}{2}$ reuna koostuu hitaammasta ja nopeammasta



Kuva 31: Kuvassa on kaksi lineaarista sovitetta kohinattomalle sigmoidiselle reunalle. Molemmissa sovitteissa reunan sijainnin ja lähinnä olevan diskretisointipiste etäisyys $\min\{|x_i - x_0^*|\} = \frac{\Delta p}{2}$

noususta, jota on havainnollistettu kuvassa 31. Kuvassa sovite **a** nousee ensin nopeasti, minkä jälkeen nousu tasaantuu. Sovitteessa **b** tilanne on päinvastainen.

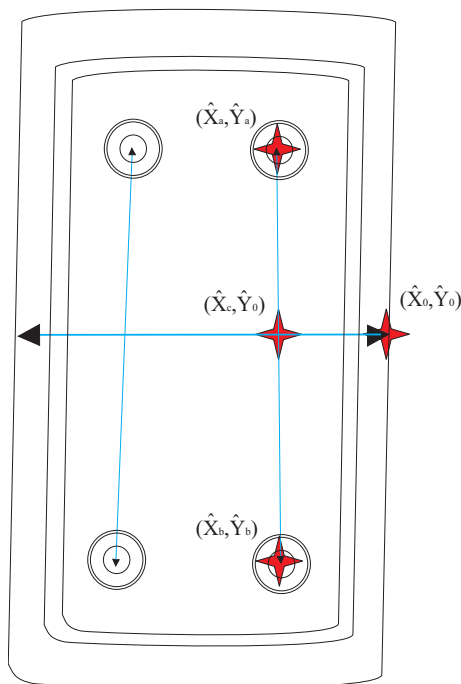
Toistokuvien diskretisoinnin alkupisteen ollessa tasaisesti jakautunut pikselivälin sisällä, $X_a \sim U[0, \Delta p]$, nähdään kuvasta 30, että tunnistus on harhatonta lukuunottamatta LoG-suotimen ja lineaarisen interpoloinnin yhdistelmää. Hajonta on samaa luokkaa kuin toistokuvien kiinteän diskretisoinnin alkupisteen tapauksessa, mikä viittaa siihen, että näissä olosuhteissa satunnainen tapaus toimisi paremmin, sillä se on harhaton toisin kuin kiinteä tapaus.

Yllättävää on, että kaikki tarkastellut interpolointimenetelmät tuottivat harhan, jonka suuruusluokassa ei ilmennyt merkittäviä eroja. Tosin lineaarinen interpolointi näytti harhan suhteen hieman heikommalta käytettäessä kiinteää diskretisointia. Myöskään keskihajonnan arvot eivät merkittävästi eronneet toisistaan. Mielenkiintoista on, että jos diskretisoinnin alkupisteet on mahdollista satunnaistaa tasajakautuneeksi, päästään harhasta eroon ja sitä kautta tarkempaan tunnistukseen keskihajonnan ollessa samaa luokkaa sekä kiinteässä että satunnaisessa tapauksessa. Tutkimukset siitä, ovatko reaali maailmassa ilmenevät kappaleen toistokuvausten diskretisoinnin alkupisteet satunnaisia ja tasaisesti jakautuneita pikselivälin sisällä, ovat tarpeellisia. Jos ne ilmenevät kiinteiksi, tulee mit-

tauslaitteistoa muokata esimerkiksi tärjäämällä mittausharhan poistamiseksi. Tällä tavoin menetelmästä saataisiin pois hallitsemattomia virhetekijöitä, jotka rajoittavat alipikselin-tarkkuuden parantamista toistomittausten avulla.

4.4 Mittausstrategia ruuvitorneja hyväksikäyttäen

Käytännössä monet mittauskohdet määritellään toisten mittausten kautta. Esimerkiksi reuna tullaan mittaamaan kappaleen puolivälistä, joka taas saadaan sopivasti valittujen pistemäisten mittauskohdeiden kautta. Esimerkissämme tätä edustavat (matkapuhelimen-kuoren) ruuvitornit. Tällöin kaksiulotteinen reunan haku muuttuu luonteeltaan yksiulotteiseksi. Kuvassa 32 on esitelty mahdollinen mittaussstrategia, jossa reunan sijainti \hat{Y}_0 y -akselilla on määritelty kahden ruuvitornin sijainnin perusteella.



Kuva 32: Reunan sijainti y -akselilla perustuu ruuvitornien avulla havaittavaan pisteeseen 2.

Jotta mittaussstrategian avulla suoritettava kohteenpaikannus onnistuisi, tulee myös ruuvitornin paikannuksesta olla tietoa. Testataan tätä yksinkertaisella kokeella. Templaattisovituksen idean mukaisesti sovitamme tunnettua ruuvitornin funktiota $r(z)$ kuvaan, joka on käsitelty kaksisuuntaisella lineaarisella interpoloinnilla. Kohdefunktio on muotoa

$$r(z, q) = \begin{cases} \frac{1}{1+e^{-8(z^2+q^2)+20}}, & \text{jos } (z^2 + q^2) < 3.5 \\ \frac{1}{1+e^{-8(z^2+q^2)+36}} & \text{muuten} \end{cases}, \quad z, q \in [-5, 5], \quad (49)$$

eli kohdan intensiteetti oli riippuvainen sen etäisyydestä pisteestä $(0, 0)$, johon kohteen sijainti on määritelty. Ideaalisen ilmiön oletettiin olevan samaa muotoa

$$f(x, y) = \begin{cases} \frac{1}{1+e^{-8(x^2+y^2)+20}}, & \text{jos } (x^2 + y^2) < 3.5 \\ \frac{1}{1+e^{-8(y^2+y^2)+36}} & \text{muuten} \end{cases}, \quad x, y \in [-6, 6]. \quad (50)$$

Ilmiöstä kerättiin diskreetti aineisto kiinteällä pikselivälillä $\Delta p = 0.16$ ja signaali-kohina-suhteella $SNR = 10$. Pikselikoko oli siis 0.16×0.16 . Näillä arvoilla pyrittiin jäljittelemään mahdollisimman hyvin reaali maailmassa ilmenevien ruuvitornien muotoja ja havaittavuuksia mukailleen. Toistokuvauksia ei suoritettu, vaan yhdestä kappaleesta havaittiin vain yksi kuva. Näillä olosuhteilla aineisto oli muotoa

$$I_{xy} = \{f(x_i, y_j) + \varepsilon_{ij}\}_{i,j=1}^{76}, \quad \varepsilon \sim N(0, 0.1). \quad (51)$$

Aineisto interpoloitiin kaksisuuntaisella lineaarisella interpoloinnilla diskretisointivälillä $\Delta sim = 0.01$. Tästä johtuen myös kohdefunktio diskretisoitiin samalla diskretisointivälillä $\Delta sim = 0.01$ ja saadut funktiot $\hat{I}_{xy}(i)$ ja r_{sim} konvoloiitiin Fourier-muunnosta avuksi käyttäen.

Kohde, eli ruuvitorni pystyttiin näissä olosuhteissa havaitsemaan harhalla, jonka suuruus oli $|Bias| \leq 0.001$. Myös suuntaiset harhat suuntiin x ja y olivat luokkaa $Bias_x \leq 0.001$ ja $Bias_y \leq 0.001$. Näiden tarkastelujen perusteella voidaan siis sanoa, että menetelmä on odotusarvoisesti harhaton. Harhan lisäksi myös tarkkuus, johon päästiin, oli erittäin hyvä. Pikselihajonnassa päästiin tarkkuuteen $\hat{\alpha} = 0.012$. Tarkkuus oli siis erittäin suuri, kun otetaan huomioon, että signaali-kohina-suhde oli suurin mitä tämän luvun muissa tarkasteluissa on käytetty, eli $SNR = 10$. Verrataessa edellisiin tarkasteluihin tulee ottaa huomioon, että toistokuvauksia ei suoritettu, joten kohinan vaikutusta ei ole pienennetty tällä tavoin. Ruuvitornin tarkasteluiden perusteella tarkkuus, johon päästään on huomattavan suuri.

Käytettävän mittausstrategian mukaisesti reunan sijainti y -akselilla siis määriteltiin ruuvitornien sijaintien (\hat{X}_a, \hat{Y}_a) ja (\hat{X}_b, \hat{Y}_b) avulla kohtaan

$$(\hat{X}_c, \hat{Y}_0) = \left(\frac{\hat{X}_a + \hat{X}_b}{2}, \frac{\hat{Y}_a + \hat{Y}_b}{2} \right).$$

Laskemalla tämä sijainti kahden riippumattoman ruuvitornin sijainnin avulla saatiin sen y -akselin suuntaiseksi pikselihajonnaksi $\hat{sd}(\alpha_y) \approx 0.015$.

Seuraavaksi tutkimme luvussa 4.2 esitellyn reunan havaitsemista mittausstrategiassa saadun y -akselin suuntaisen pikselihajonnan arvolla. Kohinan vaikutusta testattiin vain tapauksessa, jossa $SNR = 10$, koska muilla signaali-kohina-suhteen arvoilla ei ruuvitorneissa juurikaan havaittu vaihtelua. Kyseessä voidaan ajatella olevan heikoimman mahdollisen tilanteen arviointi. Tulokset mittausstrategian mukaisesti havaittavasta reunasta on esitet-

$\frac{\Delta e}{\Delta p}$	SNR	$ Bias $	$Bias_x$	$Bias_y$		$\hat{sd}(\alpha_x)$	$\hat{sd}(\alpha_y)$	$\hat{sd}(\alpha_{xy})$
2	10	0.196	-0.196	-0.001		0.044	0.013	0.038
4	10	0.195	-0.195	-0.002		0.040	0.015	0.036
8	10	0.096	-0.096	0.000		0.050	0.014	0.040

Taulukko 3: Mittausstrategian avulla saadut reunan sijainnit yksisuuntaista lineaarista interpolointia ja templaattia käyttäen.

ty taulukossa 3. Huomioonotettavaa on, etteivät tulokset juuri eroa liitteessä B.5 saaduista arvoista. Tämä johtunee siitä, ettei reunan suunta on lähes y -akselin suuntainen, jolloin y :n vaihtelu ei siihen juuri vaikuta. Kokonaispikselihajonnassa $\hat{sd}(\alpha_{xy})$ päästään huonoimman tapauksen skenaariossa luokkaa 0.035 - 0.040 oleviin arvoihin, eli alipikselintarkkuuden menetelmästä on huomattavaa hyötyä. Mittausstrategiaa käyttämällä päästään siis vähintään $\frac{1}{25}$ pikselin tarkkuuteen. Koska kohina on tyypillisissä olosuhteissa edellisistä pienempää, on syytä olettaa, että lopullinen tarkkuus on tätäkin pienempi.

5 Johtopäätökset

Työssä esiteltiin konenäköön perustuvan helppokäyttöisen kohteen paikallistamismenetelmän. Menetelmä mahdollistaa kohteen sijainnin mittaamisen alipikselintarkkuudella, mistä johtuen sen tarkkuus ei rajoitu kameran pikselikokoon. Tämä mahdollistaa vaadittavaan tulokseen pääsemisen heikomman resoluution kameralla, mikä omalta osaltaan lisää menetelmän kilpailukykyä. Muut kilpailukykyyn vaikuttavat tekijät ovat menetelmän helppokäyttöisyys ja laskennallinen tehokkuus, mitkä mahdollistavat sen soveltamisen online mittauksissa. Helppo toteutus alentaa menetelmän käytön kynnyksiä.

Menetelmä perustuu toistokuvauksiin, interpolointiin ja templaattisovitukseen. Templaattisovituksen mahdollistaa ajatus etsittävän kohteen muodon tuntemisesta. Kyseinen menetelmä on monipuolisempi kuin perinteiset reunantunnistusmenetelmät, jotka mahdollistavat ainoastaan reunojen havaitsemisen. Työssä tarkasteltava menetelmä mahdollistaa minkä tahansa kohteen yksikäsitteisen havaitsemisen, kunhan sen muoto on ennalta määriteltävissä.

Kohteen paikallistamiselle on tehty sekä teoreettiset että empiiriset tarkastelut. Näiden tarkastelujen perusteella menetelmä toimii odotetulla tavalla. Tarkasteluissa ilmeni, että mittaustapahtumaan sisältyy ilmiöitä, jotka aiheuttavat erityisvaatimuksia kuvanottotapahtumaa ajatellen. Teoreettisissa tarkasteluissa pystyimme havaitsemaan harhan, jonka vaikutusta ei ole mahdollista poistaa perinteisillä menetelmillä. Toistokuvausten suorittaminen tietyin oletuksin kuitenkin poistaa tämän harhan, mikä on osoitettu empiiristen tarkastelujen yhteydessä.

Suoritetuissa empiirisissä tarkasteluissa on vertailtu perinteisten reunantunnistimien toimintaa tarkasteltavan menetelmän osalta vaihtelevissa tilanteissa. Nämä koostuivat signaali-kohinasuhteesta (SNR) ja reunan leveyden suhteesta pikselikokoon ($\frac{\Delta c}{\Delta p}$). Tarkasteluissa havaittiin, että templaattisovitus näyttäisi toimivan paremmin verrattuna gradientin approksimointiin perustuviin suotimiin. Toinen empiiristen tarkastelujen kohde oli toistostrategian valinta. Toistostrategiaa suunniteltaessa tulee erityistä huomiota kiinnittää sen harhan poistavaan ominaisuuteen.

Tarkkuus, johon työssä käytetyllä menetelmällä päästään, on todella hyvä. Saatujen tu-

lostien perusteella voidaan sanoa, että tarkastellun menetelmän tarkkuus on vähintään $\frac{1}{25}$ pikseliä. Todellisuudessa kyseinen tarkkuus voi olla vielä suurempi, riippuen kuvausolosuhteista ja tutkittavien kohteiden muodoista, sekä toistostrategian suorittamisesta työssä määritellyllä tavalla.

Viitteet

- [1] R.N. Bracewell: *The Fourier Transform and Its Applications*, Second Edition, Revised, McGraw-Hill Book Company, 1986
- [2] J.F. Canny: *Finding Edges and Lines in Images*, Master's thesis, MIT Artificial Intell. Lab. 1983
- [3] J.F. Canny: *Computational Approach to Edge Detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. PAMI-8, No. 6, November 1986, pp. 679-698
- [4] S. Chaudhuri: *Super-resolution Imaging*, Kluwer Academic Publishers 2001
- [5] J. W. Cooley, J. W. Tukey: *An algorithm for the machine calculation of complex Fourier series* Mathematics of Computation, Vol. 19, No. 90, April 1965, pp. 297-301
- [6] F. Devernay: *A Non-Maxima Suppression Method for Edge Detection with Sub-Pixel Accuracy*, INRIA Sophia Antipolis Research Report No. 2724, 1995
- [7] R.C. Gonzalez, R.E. Woods: *Digital Image Processing*, Second Edition, Prentice-Hall, Inc. 2002
- [8] M.H. Hueckel: *An Operator Which Locates Edges in Digitized Pictures*, Journal of the Association for Computing Machinery Vol. 18, No. 1, January 1971, pp. 113-125
- [9] R. Kakarala, A.O. Hero: *On Achievable Accuracy in Edge Localization*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 14, No. 7, November 1992, pp. 777-781
- [10] M. Kisworo, S. Venkatesh, G. West: *Modelling Edges at Subpixel Accuracy Using the Local Energy Approach*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 16, No. 4, April 1994, pp. 405-410
- [11] J. Koplowitz, V. Greco: *On the edge location error for local maximum and zero-crossing edge detectors*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 16, No. 12, December 1994, pp. 1207-1212
- [12] E.P. Lyvers, O.R. Mitchell, M.L. Akey, A.P. Reeves: *Subpixel Measurements Using a Moment-Based Edge Operator*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 11, No. 12, November 1989, pp. 1293-1309

- [13] V.S. Nalwa, T.O. Binford: *On Detecting Edges*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 8, No. 6, November 1986, pp. 699-714
- [14] D. Marr: *Vision a Computational Investigation into the Human Representation and Processing of Visual Information*, W.H. Freeman and Company 1982
- [15] M.S. Nixon, A.S. Aguado: *Feature Extraction and Image Processing*, First Edition, Newnes 2002
- [16] F. Pedersini, A. Sarti, S. Tubaro: *Estimation and Compensation of Subpixel Edge Localization Error*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 19, No. 11, November 1997, pp. 1278-1284
- [17] S. Ramdani, F. Bouchara, K. Djemal: *Probability Distributin of Sub-pixel Edge Position*, Signal Processing Vol. 84, No. 3, March 2004, pp. 445-452
- [18] G.X. Ritter, J.N. Wilson: *Handbook of Computer Vision Algorithms in Image Algebra*, Second Edition, CRC Press LLC 2001
- [19] K. Rohr: *On the Precision in Estimating the Location of Edges and Corners*, Journal of Mathematical Imaging and Vision 7, 1997, pp. 7-22
- [20] P. Rockett: *The Accuracy of Sub-Pixel Localisation in the Canny Edge Detector* In Proc. of the British Machine Vision Conference (BMVC), 1999
- [21] S. Sarkar, K.L. Boyer: *On Optimal Infinite Impulse Response Edge Detection Filters*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 13, No. 11, November 1991, pp. 1154-1171
- [22] S. Sarkar, K.L. Boyer: *"On the Localization Performance Measure and Optimal Edge Detection"*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 16, No. 1, January 1994, pp. 106-108
- [23] Y. Shan, G.W. Boon: *Sub-pixel location of edges with non-uniform blurring: a finite closed-form approach*, Image and Vision Computing, Vol. 18, 2000, 1015-1023
- [24] L.G. Shapiro, G.C. Stockman: *Computer Vision*, Prentice-Hall, Inc. 2001
- [25] C. Steger: *Unbiased extraction of curvilinear structures from 2D and 3D images*, Dissertation of PhD, Technischen Universität München, Germany, 1998.
- [26] C. Steger: *Evaluation of Subpixel Line and Edge Detection Precision and Accuracy*, International Archives of Photogrammetry and Remote Sensing (1998) Volume XXXII, Part 3/1, 256-264

- [27] A.J. Tabatabai, O.R. Mitchell: *Edge Location to Subpixel Values in Digital Imagery*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. PAMI-6, No. 2, March 1984, pp. 188-201
- [28] S.A. Tabbone, L. Alonso, D. Ziou: *Behavior of the Laplacian of Gaussian Extrema*, Journal of Mathematical Imaging and Vision 23, 2005, pp. 107-128
- [29] H.D. Tagare, R.J.P. deFigueiredo: *On the Localization Performance Measure and Optimal Edge Detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 12, No. 12, December 1990, pp. 1186-1190
- [30] H.D. Tagare, R.J.P. deFigueiredo: *Reply to "On the Localization Performance Measure and Optimal Edge Detection"*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 16, No. 1, January 1994, pp. 108-110
- [31] V. Torre, T. Poggio: *On Edge Detection*, Massachusetts Institute of Technology 1984
- [32] J. Ye, G. Fu, U.P. Poudel: *High-accuracy Edge Detection with Blurred Edge Model*, Image and Vision Computing Vol. 23, Issue 5, May 2005, Pages 453-467
- [33] D. Ziou, S. Tabbone: *Edge Detection Techniques - An Overview*, Technical Report, No. 195, Dept Math & Informatique. Universit de Sherbrooke, 1997
- [34] K. Åström, A. Heyden: *Stochastic Modelling an Analysis of Sub-pixel Edge Detection*, In Proc. International Conference on Pattern Recognition, Vienna, Austria, 1996

A Yksiulotteisen sigmoidisen reunan empiiristen tarkastelujen tulokset

A.1 Mediaani reunatoistoista templaattia käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.131	0.081	0.071	0.518
	20	0.097	0.091	0.083	0.378
	50	0.087	0.084	0.109	0.338
4	10	0.149	0.004	0.019	0.299
	20	0.070	0.002	0.008	0.140
	50	0.028	0.006	0.002	0.057
8	10	0.231	0.003	0.001	0.231
	20	0.120	0.010	0.004	0.119
	50	0.047	0.003	0.002	0.047
12	10	0.310	0.012	0.025	0.198
	20	0.159	0.008	0.023	0.106
	50	0.070	0.005	0.004	0.046

A.2 Mediaani reunatoistoista DoG-suodinta käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.305	0.091	0.092	1.218
	20	0.303	0.289	0.344	1.177
	50	0.300	0.249	0.335	1.175
4	10	0.204	0.032	0.051	0.406
	20	0.123	0.023	0.027	0.244
	50	0.099	0.016	0.016	0.197
8	10	0.335	0.028	0.049	0.334
	20	0.194	0.017	0.017	0.193
	50	0.076	0.008	0.014	0.075
12	10	0.489	0.057	0.068	0.321
	20	0.212	0.001	0.002	0.141
	50	0.095	0.004	0.002	0.063

A.3 Mediaani reunatoistoista LoG-suodinta käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.253	0.046	0.165	1.009
	20	0.244	0.248	0.473	0.946
	50	0.239	0.105	0.250	0.948
4	10	0.167	0.047	0.031	0.332
	20	0.078	0.018	0.025	0.154
	50	0.038	0.009	0.014	0.075
8	10	0.235	0.004	0.017	0.235
	20	0.121	0.003	0.005	0.121
	50	0.049	0.018	0.020	0.046
12	10	0.298	0.011	0.002	0.199
	20	0.149	0.014	0.011	0.098
	50	0.062	0.015	0.020	0.038

A.4 Keskiarvo reunatoistoista templaattia käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.127	0.165	0.177	0.479
	20	0.098	0.088	0.089	0.382
	50	0.089	0.137	0.213	0.330
4	10	0.137	0.014	0.017	0.274
	20	0.066	0.007	0.011	0.131
	50	0.025	0.003	0.008	0.050
8	10	0.190	0.001	0.007	0.190
	20	0.106	0.004	0.013	0.106
	50	0.045	0.006	0.008	0.044
12	10	0.278	0.019	0.008	0.184
	20	0.163	0.007	0.009	0.109
	50	0.061	0.002	0.002	0.041

A.5 Keskiarvo reunatoistoista DoG-suodinta käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.228	0.202	0.243	0.889
	20	0.282	0.133	0.124	1.120
	50	0.321	0.234	0.509	1.263
4	10	0.204	0.015	0.036	0.407
	20	0.095	0.020	0.014	0.189
	50	0.092	0.005	0.020	0.184
8	10	0.363	0.015	0.020	0.362
	20	0.166	0.007	0.001	0.166
	50	0.073	0.004	0.002	0.072
12	10	0.440	0.016	0.006	0.293
	20	0.207	0.002	0.007	0.138
	50	0.082	0.000	0.007	0.055

A.6 Keskiarvo reunatoistoista LoG-suodinta käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.207	0.131	0.183	0.818
	20	0.234	0.097	0.162	0.930
	50	0.246	0.130	0.210	0.974
4	10	0.189	0.022	0.002	0.378
	20	0.069	0.006	0.001	0.138
	50	0.032	0.007	0.005	0.064
8	10	0.205	0.022	0.030	0.204
	20	0.099	0.016	0.023	0.098
	50	0.048	0.015	0.014	0.046
12	10	0.312	0.034	0.033	0.205
	20	0.132	0.013	0.011	0.087
	50	0.056	0.016	0.017	0.034

A.7 Mediaani pikselitoistoista templaattia käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.124	0.115	0.162	0.481
	20	0.099	0.093	0.145	0.383
	50	0.085	0.105	0.127	0.322
4	10	0.142	0.007	0.004	0.284
	20	0.074	0.001	0.004	0.148
	50	0.029	0.005	0.016	0.057
8	10	0.243	0.019	0.043	0.242
	20	0.124	0.008	0.020	0.124
	50	0.048	0.001	0.004	0.048
12	10	0.305	0.012	0.016	0.203
	20	0.182	0.003	0.008	0.121
	50	0.068	0.003	0.002	0.045

A.8 Mediaani pikselitoistoista DoG-suodinta käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.309	0.119	0.227	1.230
	20	0.318	0.191	0.397	1.259
	50	0.317	0.014	0.139	1.267
4	10	0.233	0.023	0.0508	0.465
	20	0.115	0.012	0.031	0.231
	50	0.096	0.001	0.008	0.192
8	10	0.433	0.024	0.016	0.432
	20	0.194	0.031	0.031	0.191
	50	0.077	0.004	0.004	0.076
12	10	0.458	0.046	0.043	0.302
	20	0.245	0.011	0.014	0.163
	50	0.090	0.003	0.008	0.060

A.9 Mediaani pikselitoistoista LoG-suodinta käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.277	0.106	0.402	1.102
	20	0.260	0.073	0.121	1.036
	50	0.248	0.060	0.332	0.990
4	10	0.171	0.032	0.008	0.341
	20	0.080	0.010	0.008	0.160
	50	0.039	0.020	0.020	0.075
8	10	0.234	0.001	0.004	0.234
	20	0.106	0.016	0.020	0.105
	50	0.049	0.007	0.008	0.049
12	10	0.299	0.025	0.020	0.198
	20	0.163	0.027	0.043	0.105
	50	0.062	0.010	0.008	0.040

A.10 Keskiarvo pikselitoistoista templaattia käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.124	0.053	0.045	0.495
	20	0.092	0.084	0.098	0.359
	50	0.087	0.079	0.104	0.339
4	10	0.120	0.008	0.020	0.280
	20	0.065	0.002	0.008	0.131
	50	0.025	0.003	0.004	0.050
8	10	0.218	0.002	0.008	0.218
	20	0.108	0.011	0.016	0.108
	50	0.040	0.004	0.008	0.040
12	10	0.298	0.018	0.010	0.198
	20	0.148	0.005	0.004	0.099
	50	0.063	0.003	0.008	0.042

A.11 Keskiarvo pikselitoistoista DoG-suodinta käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.326	0.065	0.145	1.303
	20	0.312	0.206	0.490	1.232
	50	0.317	0.124	0.338	1.260
4	10	0.203	0.013	0.004	0.405
	20	0.125	0.027	0.016	0.248
	50	0.093	0.020	0.027	0.185
8	10	0.365	0.032	0.045	0.363
	20	0.168	0.019	0.020	0.167
	50	0.070	0.009	0.008	0.070
12	10	0.446	0.031	0.033	0.296
	20	0.227	0.002	0.008	0.151
	50	0.090	0.001	0.004	0.060

A.12 Keskiarvo pikselitoistoista LoG-suodinta käyttäen

$\frac{\Delta e}{\Delta p}$	SNR	$\hat{\alpha}$	\hat{Bias}	\hat{ME}	\hat{sd}
2	10	0.269	0.079	0.191	1.073
	20	0.268	0.052	0.326	1.072
	50	0.261	0.078	0.320	1.042
4	10	0.143	0.008	0.004	0.285
	20	0.080	0.020	0.020	0.159
	50	0.033	0.011	0.008	0.066
8	10	0.209	0.006	0.004	0.209
	20	0.108	0.018	0.020	0.107
	50	0.049	0.014	0.008	0.047
12	10	0.278	0.004	0.004	0.185
	20	0.131	0.017	0.014	0.086
	50	0.059	0.011	0.008	0.037

B Kaksiulotteisen sigmoidisen reunan empiiristen tarkastelujen tulokset

B.1 Keskiarvo pikselitoistoista käyttäen reunantunnistusmenetelmänä kaksisuuntaista lineaarista interpolointia ja templaattisovitusta.

$\frac{\Delta e}{\Delta p}$	SNR	$ Bias $	$Bias_x$	$Bias_y$		$\hat{s}d(\alpha_x)$	$\hat{s}d(\alpha_y)$	$\hat{s}d(\alpha_{xy})$
2	10	0.015	-0.013	0.008		0.049	0.103	0.098
	20	0.089	-0.029	-0.085		0.040	0.111	0.098
	50	0.035	-0.011	-0.033		0.031	0.105	0.092
4	10	0.017	-0.008	0.015		0.042	0.098	0.091
	20	0.007	-0.007	0.001		0.025	0.099	0.085
	50	0.008	-0.008	-0.004		0.013	0.107	0.086
8	10	0.011	-0.010	0.004		0.054	0.101	0.099
	20	0.008	-0.008	0.001		0.029	0.110	0.094
	50	0.004	0.000	-0.004		0.007	0.104	0.082

B.2 Keskiarvo pikselitoistoista käyttäen reunantunnistusmenetelmänä kaksisuuntaista lineaarista interpolointia ja Laplacian of Gaussiania.

$\frac{\Delta e}{\Delta p}$	SNR	$ Bias $	$Bias_x$	$Bias_y$		$\hat{s}d(\alpha_x)$	$\hat{s}d(\alpha_y)$	$\hat{s}d(\alpha_{xy})$
2	10	0.071	-0.070	0.013		0.272	0.101	0.225
	20	0.028	0.019	0.020		0.183	0.093	0.190
	50	0.047	-0.044	-0.018		0.182	0.106	0.196
4	10	0.013	-0.011	-0.008		0.125	0.102	0.141
	20	0.014	-0.005	0.013		0.072	0.095	0.104
	50	0.040	-0.020	-0.035		0.043	0.101	0.096
8	10	0.006	-0.003	0.006		0.144	0.098	0.153
	20	0.013	-0.012	-0.004		0.077	0.092	0.105
	50	0.015	-0.010	-0.011		0.040	0.102	0.089

B.3 Keskiarvo reunatoistoista käyttäen reunantunnistusmenetelmänä kaksisuuntaista lineaarista interpolointia ja templaattisovitusta.

$\frac{\Delta e}{\Delta p}$	<i>SNR</i>	$ Bias $	$Bias_x$	$Bias_y$		$\hat{sd}(\alpha_x)$	$\hat{sd}(\alpha_y)$	$\hat{sd}(\alpha_{xy})$
2	10	0.021	-0.007	-0.020		0.047	0.105	0.101
	20	0.021	-0.020	-0.008		0.036	0.110	0.098
	50	0.050	-0.014	-0.047		0.030	0.099	0.088
4	10	0.017	0.002	-0.017		0.039	0.101	0.088
	20	0.028	-0.018	-0.021		0.023	0.095	0.080
	50	0.005	-0.004	0.004		0.012	0.095	0.073
8	10	0.007	-0.005	0.005		0.054	0.096	0.092
	20	0.008	-0.004	0.007		0.027	0.104	0.086
	50	0.005	0.000	0.005		0.007	0.097	0.079

B.4 Keskiarvo reunatoistoista käyttäen reunantunnistusmenetelmänä kaksisuuntaista lineaarista interpolointia ja LoG-suodinta.

$\frac{\Delta e}{\Delta p}$	<i>SNR</i>	$ Bias $	$Bias_x$	$Bias_y$		$\hat{sd}(\alpha_x)$	$\hat{sd}(\alpha_y)$	$\hat{sd}(\alpha_{xy})$
2	10	0.170	-0.161	0.058		0.249	0.101	0.212
	20	0.034	-0.029	-0.019		0.182	0.104	0.193
	50	0.077	0.067	-0.038		0.179	0.095	0.190
4	10	0.016	0.012	-0.010		0.137	0.104	0.147
	20	0.018	-0.006	-0.017		0.067	0.101	0.107
	50	0.015	-0.002	0.015		0.045	0.094	0.090
8	10	0.014	-0.014	-0.003		0.148	0.099	0.153
	20	0.016	-0.016	0.005		0.087	0.099	0.117
	50	0.011	-0.009	0.006		0.046	0.097	0.089

$\frac{\Delta e}{\Delta p}$	SNR	$ Bias $	$Bias_x$	$Bias_y$		$\hat{sd}(\alpha_x)$	$\hat{sd}(\alpha_y)$	$\hat{sd}(\alpha_{xy})$
2	10	0.212	-0.205	-0.057		0.043	0.103	0.094
	20	0.223	-0.221	-0.032		0.033	0.105	0.093
	50	0.203	-0.203	-0.003		0.028	0.105	0.091
4	10	0.202	-0.202	-0.005		0.043	0.103	0.095
	20	0.198	-0.198	0.006		0.026	0.097	0.083
	50	0.200	-0.2	-0.007		0.009	0.098	0.080
8	10	0.099	-0.099	-0.001		0.055	0.106	0.100
	20	0.101	-0.100	0.014		0.017	0.095	0.077
	50	0.100	-0.100	0.004		0.015	0.099	0.078

B.5 Keskiarvo pikselitoistoista käyttäen reunantunnistusmenetelmänä yksisuuntaista lineaarista interpolointia ja templaattisovitusta.

B.6 Keskiarvo pikselitoistoista käyttäen reunantunnistusmenetelmänä yksisuuntaista lineaarista interpolointia ja LoG-suodinta.

$\frac{\Delta e}{\Delta p}$	SNR	$ Bias $	$Bias_x$	$Bias_y$		$\hat{sd}(\alpha_x)$	$\hat{sd}(\alpha_y)$	$\hat{sd}(\alpha_{xy})$
2	10	6.498	-6.498	-0.012		0.006	0.103	0.082
	20	6.500	-6.5	-0.065		0	0.105	0.081
	50	6.500	-6.5	-0.071		0	0.094	0.075
4	10	0.210	-0.209	0.015		0.156	0.099	0.163
	20	0.191	-0.189	0.030		0.082	0.106	0.120
	50	0.207	-0.207	0.010		0.045	0.104	0.098
8	10	0.088	-0.088	-0.000		0.142	0.103	0.154
	20	0.106	-0.106	-0.003		0.094	0.100	0.121
	50	0.104	-0.103	-0.015		0.041	0.103	0.093

B.7 Keskiarvo reunatoistoista käyttäen reunantunnistusmenetelmänä yksisuuntaista lineaarista interpolointia ja templaattisovitusta.

$\frac{\Delta e}{\Delta p}$	<i>SNR</i>	$ Bias $	$Bias_x$	$Bias_y$		$\hat{sd}(\alpha_x)$	$\hat{sd}(\alpha_y)$	$\hat{sd}(\alpha_{xy})$
2	10	0.199	-0.199	0.005		0.045	0.095	0.087
	20	0.202	-0.202	0.006		0.031	0.096	0.082
	50	0.189	-0.185	0.038		0.027	0.095	0.082
4	10	0.200	-0.200	0.002		0.042	0.108	0.095
	20	0.205	-0.204	0.015		0.025	0.099	0.084
	50	0.201	-0.201	-0.016		0.011	0.097	0.077
8	10	0.099	-0.099	0.002		0.059	0.094	0.096
	20	0.101	-0.101	-0.002		0.026	0.104	0.086
	50	0.100	-0.099	0.013		0.012	0.091	0.074

B.8 Keskiarvo reunatoistoista käyttäen reunantunnistusmenetelmänä yksisuuntaista lineaarista interpolointia ja LoG-suodinta.

$\frac{\Delta e}{\Delta p}$	<i>SNR</i>	$ Bias $	$Bias_x$	$Bias_y$		$\hat{sd}(\alpha_x)$	$\hat{sd}(\alpha_y)$	$\hat{sd}(\alpha_{xy})$
2	10	6.500	-6.5	-0.019		0	0.100	0.080
	20	6.500	-6.5	-0.013		0	0.102	0.084
	50	6.500	-6.5	-0.007		0	0.097	0.077
4	10	0.211	-0.21	0.015		0.154	0.105	0.165
	20	0.210	-0.210	-0.015		0.079	0.097	0.110
	50	0.200	-0.198	0.030		0.044	0.108	0.101
8	10	0.102	-0.102	-0.002		0.154	0.103	0.162
	20	0.100	-0.1	0.010		0.088	0.101	0.118
	50	0.103	-0.103	-0.004		0.039	0.091	0.082

C Reunan sijainnin varianssin osien johtaminen

C.1 Kokonaisresponsi

$$\begin{aligned}
m_T(x) &= \int_{x-a}^{x_{i-1}} \hat{F}(z)g(x-z)dz + \int_{x_{i-1}}^{x_i} \hat{F}(z)g(x-z)dz + \int_{x_i}^{x+a} \hat{F}(z)g(x-z)dz \\
&= \underbrace{f_{i-2} \int_{x-a}^{x_{i-1}} g(z-x)dz + \frac{f_{i-1} - f_{i-2}}{\Delta p} \int_{x-a}^{x_{i-1}} (x - x_{i-2})g(z-x)dz}_{\subset m_f(x)} \\
&\quad + \underbrace{f_{i-1} \int_{x_{i-1}}^{x_i} g(z-x)dz + \frac{f_i - f_{i-1}}{\Delta p} \int_{x_{i-1}}^{x_i} (x - x_{i-1})g(z-x)dz}_{\subset m_f(x)} \\
&\quad + \underbrace{f_i \int_{x_i}^{x+a} g(z-x)dz + \frac{f_{i+1} - f_i}{\Delta p} \int_{x_i}^{x+a} (x - x_i)g(z-x)dz}_{\subset m_f(x)} \\
&\quad + \underbrace{\varepsilon_{i-2} \int_{x-a}^{x_{i-1}} g(z-x)dz + \frac{\varepsilon_{i-1} - \varepsilon_{i-2}}{\Delta p} \int_{x-a}^{x_{i-1}} (x - x_{i-2})g(z-x)dz}_{\subset m_N(x)} \\
&\quad + \underbrace{\varepsilon_{i-1} \int_{x_{i-1}}^{x_i} g(z-x)dz + \frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p} \int_{x_{i-1}}^{x_i} (x - x_{i-1})g(z-x)dz}_{\subset m_N(x)} \\
&\quad + \underbrace{\varepsilon_i \int_{x_i}^{x+a} g(z-x)dz + \frac{\varepsilon_{i+1} - \varepsilon_i}{\Delta p} \int_{x_i}^{x+a} (x - x_i)g(z-x)dz}_{\subset m_N(x)}
\end{aligned}$$

$$\begin{aligned}
m_T(x) = & \underbrace{f_{i-2} \int_{x-a}^{x_{i-1}} g(z-x) dz + \frac{f_{i-1} - f_{i-2}}{\Delta p} \int_{x-a}^{x_{i-1}} xg(z-x) dz}_{\subset m_f(x)} \\
& - \underbrace{\frac{f_{i-1} - f_{i-2}}{\Delta p} x_{i-2} \int_{x-a}^{x_{i-1}} g(z-x) dz}_{\subset m_f(x)} \\
& + \underbrace{f_{i-1} \int_{x_{i-1}}^{x_i} g(z-x) dz + \frac{f_i - f_{i-1}}{\Delta p} \int_{x_{i-1}}^{x_i} xg(z-x) dz}_{\subset m_f(x)} \\
& - \underbrace{\frac{f_i - f_{i-1}}{\Delta p} x_{i-1} \int_{x_{i-1}}^{x_i} g(z-x) dz}_{\subset m_f(x)} \\
& + \underbrace{f_i \int_{x_i}^{x+a} g(z-x) dz + \frac{f_{i+1} - f_i}{\Delta p} \int_{x_i}^{x+a} xg(z-x) dz}_{\subset m_f(x)} \\
& - \underbrace{\frac{f_{i+1} - f_i}{\Delta p} x_i \int_{x_i}^{x+a} g(z-x) dz}_{\subset m_f(x)} \\
& + \underbrace{\varepsilon_{i-2} \int_{x-a}^{x_{i-1}} g(z-x) dz + \frac{\varepsilon_{i-1} - \varepsilon_{i-2}}{\Delta p} \int_{x-a}^{x_{i-1}} xg(z-x) dz}_{\subset m_N(x)} \\
& - \underbrace{\frac{\varepsilon_{i-1} - \varepsilon_{i-2}}{\Delta p} x_{i-2} \int_{x-a}^{x_{i-1}} g(z-x) dz}_{\subset m_N(x)} \\
& + \underbrace{\varepsilon_{i-1} \int_{x_{i-1}}^{x_i} g(z-x) dz + \frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p} \int_{x_{i-1}}^{x_i} xg(z-x) dz}_{\subset m_N(x)} \\
& - \underbrace{\frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p} x_{i-1} \int_{x_{i-1}}^{x_i} g(z-x) dz}_{\subset m_N(x)} \\
& + \underbrace{\varepsilon_i \int_{x_i}^{x+a} g(z-x) dz + \frac{\varepsilon_{i+1} - \varepsilon_i}{\Delta p} \int_{x_i}^{x+a} xg(z-x) dz}_{\subset m_N(x)} \\
& - \underbrace{\frac{\varepsilon_{i+1} - \varepsilon_i}{\Delta p} x_i \int_{x_i}^{x+a} g(z-x) dz}_{\subset m_N(x)}
\end{aligned}$$

C.2 Ilmiöstä johtuva kokonaisresponsin osa

$$\begin{aligned}
m'_f(x) = & \frac{1}{\Delta p} \left[((\Delta p + x_{i-2})f_{i-2} - x_{i-2}f_{i-1}) \int_{x-a}^{x_{i-1}} g'(z-x)dz + (f_{i-1} - f_{i-2}) \int_{x-a}^{x_{i-1}} g(z-x)dz \right. \\
& + ((\Delta p + x_{i-1})f_{i-1} - x_{i-1}f_i) \int_{x_{i-1}}^{x_i} g'(z-x)dz + (f_i - f_{i-1}) \int_{x_{i-1}}^{x_i} g(z-x)dz \\
& \left. + ((\Delta p + x_i)f_i - x_i f_{i+1}) \int_{x_i}^{x+a} g'(z-x)dz + (f_{i+1} - f_i) \int_{x_i}^{x+a} g(z-x)dz \right]
\end{aligned}$$

$$\begin{aligned}
m''_f(x) = & \frac{1}{\Delta p} \left[((\Delta p + x_{i-2})f_{i-2} - x_{i-2}f_{i-1}) \int_{x-a}^{x_{i-1}} g''(z-x)dz + (f_{i-1} - f_{i-2}) \int_{x-a}^{x_{i-1}} g'(z-x)dz \right. \\
& + ((\Delta p + x_{i-1})f_{i-1} - x_{i-1}f_i) \int_{x_{i-1}}^{x_i} g''(z-x)dz + (f_i - f_{i-1}) \int_{x_{i-1}}^{x_i} g'(z-x)dz \\
& \left. + ((\Delta p + x_i)f_i - x_i f_{i+1}) \int_{x_i}^{x+a} g''(z-x)dz + (f_{i+1} - f_i) \int_{x_i}^{x+a} g'(z-x)dz \right]
\end{aligned}$$

C.3 Kohinasta johtuva kokonaisresponsin osa

$$\begin{aligned}
m'_N(x) &= \varepsilon_{i-2} \int_{x-a}^{x_{i-1}} g'(z-x) dz + \frac{\varepsilon_{i-1} - \varepsilon_{i-2}}{\Delta p} \int_{x-a}^{x_{i-1}} x' g(z-x) dz \\
&\quad - \frac{\varepsilon_{i-1} - \varepsilon_{i-2}}{\Delta p} x_{i-2} \int_{x-a}^{x_{i-1}} g'(z-x) dz \\
&\quad + \varepsilon_{i-1} \int_{x_{i-1}}^{x_i} g'(z-x) dz + \frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p} \int_{x_{i-1}}^{x_i} x' g(z-x) dz \\
&\quad - \frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p} x_{i-1} \int_{x_{i-1}}^{x_i} g'(z-x) dz \\
&\quad + \varepsilon_i \int_{x_i}^{x+a} g'(z-x) dz + \frac{\varepsilon_{i+1} - \varepsilon_i}{\Delta p} \int_{x_i}^{x+a} x' g(z-x) dz \\
&\quad - \frac{\varepsilon_{i+1} - \varepsilon_i}{\Delta p} x_i \int_{x_i}^{x+a} g'(z-x) dz \\
m'_N(x) &= \underbrace{\left(\varepsilon_{i-2} - \frac{\varepsilon_{i-1} - \varepsilon_{i-2}}{\Delta p} x_{i-2} \right) \int_{x-a}^{x_{i-1}} g'(z-x) dz}_A \\
&\quad + \underbrace{\frac{\varepsilon_{i-1} - \varepsilon_{i-2}}{\Delta p} \int_{x-a}^{x_{i-1}} g(z-x) dz}_B \\
&\quad + \underbrace{\left(\varepsilon_{i-1} - \frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p} x_{i-1} \right) \int_{x_{i-1}}^{x_i} g'(z-x) dz}_C \\
&\quad + \underbrace{\frac{\varepsilon_i - \varepsilon_{i-1}}{\Delta p} \int_{x_{i-1}}^{x_i} g(z-x) dz}_D \\
&\quad + \underbrace{\left(\varepsilon_i - \frac{\varepsilon_{i+1} - \varepsilon_i}{\Delta p} x_i \right) \int_{x_i}^{x+a} g'(z-x) dz}_E + \\
&\quad \underbrace{\frac{\varepsilon_{i+1} - \varepsilon_i}{\Delta p} \int_{x_i}^{x+a} g(z-x) dz}_F
\end{aligned}$$

$$\begin{aligned}
m'_N(x) = & \frac{1}{\Delta p} \left[\underbrace{((\Delta p + x_{i-2})\varepsilon_{i-2} - x_{i-2}\varepsilon_{i-1}) \int_{x-a}^{x_{i-1}} g'(z-x) dz}_A \right. \\
& + \underbrace{(\varepsilon_{i-1} - \varepsilon_{i-2}) \int_{x-a}^{x_{i-1}} g(z-x) dz}_B \\
& + \underbrace{((\Delta p + x_{i-1})\varepsilon_{i-1} - x_{i-1}\varepsilon_i) \int_{x_{i-1}}^{x_i} g'(z-x) dz}_C + \\
& \underbrace{(\varepsilon_i - \varepsilon_{i-1}) \int_{x_{i-1}}^{x_i} g(z-x) dz}_D \\
& + \underbrace{((\Delta p + x_i)\varepsilon_i - x_i\varepsilon_{i+1}) \int_{x_i}^{x+a} g'(z-x) dz}_E + \\
& \left. \underbrace{(\varepsilon_{i+1} - \varepsilon_i) \int_{x_i}^{x+a} g(z-x) dz}_F \right]
\end{aligned}$$

C.4 Kokonaisresponsin kohinasta johtuvan osan nelion odotusarvo

$$\begin{aligned}
\mathbb{E}[m'_N(x)]^2 = & \frac{1}{\Delta p^2} \mathbb{E}[A^2 + 2AB + 2AC + 2AD + \underbrace{2AE + 2AF}_0 \\
& + B^2 + 2BC + 2BD + \underbrace{2BE + 2BF}_0 \\
& + C^2 + 2CD + 2CE + 2CF \\
& + D^2 + 2DE + 2DF \\
& + E^2 + 2EF \\
& + F^2] \\
\mathbb{E}[m'_N(x)]^2 = & \frac{1}{\Delta p^2} [\mathbb{E}A^2 + \mathbb{E}2AB + \mathbb{E}2AC + \mathbb{E}2AD \\
& + \mathbb{E}B^2 + \mathbb{E}2BC + \mathbb{E}2BD \\
& + \mathbb{E}C^2 + \mathbb{E}2CD + \mathbb{E}2CE + \mathbb{E}2CF \\
& + \mathbb{E}D^2 + \mathbb{E}2DE + \mathbb{E}2DF \\
& + \mathbb{E}E^2 + \mathbb{E}2EF \\
& + \mathbb{E}F^2]
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}[m'_N(x)]^2 &= \frac{\sigma^2}{\Delta p^2} \left[(\Delta p^2 + 2\Delta p x_{i-2} + 2x_{i-2}^2) \left[\int_{x-a}^{x_{i-1}} g'(z-x) dz \right]^2 \right. \\
&\quad - 2(\Delta p + 2x_{i-2}) \int_{x-a}^{x_{i-1}} g'(z-x) dz \int_{x-a}^{x_{i-1}} g(z-x) dz \\
&\quad - 2((\Delta p + x_{i-1})x_{i-2}) \int_{x-a}^{x_{i-1}} g'(z-x) dz \int_{x_{i-1}}^{x_i} g'(z-x) dz \\
&\quad + 2x_{i-2} \int_{x-a}^{x_{i-1}} g'(z-x) dz \int_{x_{i-1}}^{x_i} g(z-x) dz \\
&\quad + 2 \left[\int_{x-a}^{x_{i-1}} g(z-x) dz \right]^2 \\
&\quad + 2(\Delta p + x_{i-1}) \int_{x-a}^{x_{i-1}} g(z-x) dz \int_{x_{i-1}}^{x_i} g'(z-x) dz \\
&\quad + 2 \int_{x-a}^{x_{i-1}} g(z-x) dz \int_{x_{i-1}}^{x_i} g(z-x) dz \\
&\quad + (\Delta p^2 + 2\Delta p x_{i-1} + 2x_{i-1}^2) \left[\int_{x_{i-1}}^{x_i} g'(z-x) dz \right]^2 \\
&\quad - 2(\Delta p + 2x_{i-1}) \int_{x_{i-1}}^{x_i} g'(z-x) dz \int_{x_{i-1}}^{x_i} g(z-x) dz \\
&\quad - 2((\Delta p + x_i)x_{i-1}) \int_{x_{i-1}}^{x_i} g'(z-x) dz \int_{x_i}^{x+a} g'(z-x) dz \\
&\quad + 2x_{i-1} \int_{x_{i-1}}^{x_i} g'(z-x) dz \int_{x_i}^{x+a} g(z-x) dz \\
&\quad + 2 \left[\int_{x_{i-1}}^{x_i} g(z-x) dz \right]^2 \\
&\quad + 2(\Delta p + x_i) \int_{x_{i-1}}^{x_i} g(z-x) dz \int_{x_i}^{x+a} g'(z-x) dz \\
&\quad + 2 \int_{x_{i-1}}^{x_i} g(z-x) dz \int_{x_i}^{x+a} g(z-x) dz \\
&\quad + (\Delta p^2 + 2\Delta p x_i + 2x_i^2) \left[\int_{x_i}^{x+a} g'(z-x) dz \right]^2 \\
&\quad - 2(\Delta p + 2x_i) \int_{x_i}^{x+a} g'(z-x) dz \int_{x_i}^{x+a} g(z-x) dz \\
&\quad \left. + 2 \left[\int_{x_i}^{x+a} g(z-x) dz \right]^2 \right]
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}[m'_N(x)]^2 &= \frac{2\sigma^2}{\Delta p^2} \left[\left(\frac{\Delta p^2}{2} + \Delta p x_{i-2} + x_{i-2}^2 \right) \left[\int_{x-a}^{x_{i-1}} g'(z-x) dz \right]^2 \right. \\
&+ \left(\frac{\Delta p^2}{2} + \Delta p x_{i-1} + x_{i-1}^2 \right) \left[\int_{x_{i-1}}^{x_i} g'(z-x) dz \right]^2 \\
&+ \left(\frac{\Delta p^2}{2} + \Delta p x_i + x_i^2 \right) \left[\int_{x_i}^{x+a} g'(z-x) dz \right]^2 \\
&- (\Delta p + 2x_{i-2}) \int_{x-a}^{x_{i-1}} g'(z-x) dz \int_{x-a}^{x_{i-1}} g(z-x) dz \\
&- (\Delta p + 2x_{i-1}) \int_{x_{i-1}}^{x_i} g'(z-x) dz \int_{x_{i-1}}^{x_i} g(z-x) dz \\
&- (\Delta p + 2x_i) \int_{x_i}^{x+a} g'(z-x) dz \int_{x_i}^{x+a} g(z-x) dz \\
&- (\Delta p + x_{i-1}) x_{i-2} \int_{x-a}^{x_{i-1}} g'(z-x) dz \int_{x_{i-1}}^{x_i} g'(z-x) dz \\
&- (\Delta p + x_i) x_{i-1} \int_{x_{i-1}}^{x_i} g'(z-x) dz \int_{x_i}^{x+a} g'(z-x) dz \\
&+ (\Delta p + x_{i-1}) \int_{x-a}^{x_{i-1}} g(z-x) dz \int_{x_{i-1}}^{x_i} g'(z-x) dz \\
&+ (\Delta p + x_i) \int_{x_{i-1}}^{x_i} g(z-x) dz \int_{x_i}^{x+a} g'(z-x) dz \\
&+ x_{i-2} \int_{x-a}^{x_{i-1}} g'(z-x) dz \int_{x_{i-1}}^{x_i} g(z-x) dz \\
&+ x_{i-1} \int_{x_{i-1}}^{x_i} g'(z-x) dz \int_{x_i}^{x+a} g(z-x) dz \\
&+ \int_{x-a}^{x_{i-1}} g(z-x) dz \int_{x_{i-1}}^{x_i} g(z-x) dz \\
&+ \int_{x_{i-1}}^{x_i} g(z-x) dz \int_{x_i}^{x+a} g(z-x) dz \\
&+ \left[\int_{x-a}^{x_{i-1}} g(z-x) dz \right]^2 \\
&+ \left[\int_{x_{i-1}}^{x_i} g(z-x) dz \right]^2 \\
&+ \left. \left[\int_{x_i}^{x+a} g(z-x) dz \right]^2 \right]
\end{aligned}$$