Tampereen yliopisto

Jukka Heikintalo

KOHTEENSEURANNAN MENETELMIÄ KONENÄÖN AVULLA

Kandidaattitutkielma

Informaatioteknologian ja viestinnän tiedekunta
Kandidaattitutkielma
Joulukuu 2019

TIIVISTELMÄ

Jukka Heikintalo: Kohteenseurannan menetelmiä konenäön avulla
Kandidaattitutkielma
Tampereen yliopisto
Tietojenkäsittelytieteiden tutkinto-ohjelma
Joulukuu 2019

Tämä tutkielma koostuu kirjallisuuskatsauksesta kohteenseurantaan konenäön avulla. Tarkoituksena on selvittää kohteenseurantamenetelmien toimintaperiaatteita ja verrata eri kohteenseurannan ratkaisuja keskenään. Tutkielman lopussa esitellään kohteenseurannan hyvä tarkkuus ideaaliolosuhteissa, hybridiratkaisujen hyödyt ja ratkaisujen heikentynyt toiminta peitettäessä.

Avainsanat: Kohteenseuranta, konenäkö

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck –ohjelmalla.

1 Johdanto 1

2 Menetelmät 2

2.1 Blob Tracking eli alueen seuranta 2

2.2 Model Based eli malliin perustuva seuranta 3

2.3 Contour Tracking eli reunojen seuranta 4

2.4 Optical Flow eli liike kahden kuvan välillä 5

3 Tulokset 6

3.1 Alueen seuranta 7

3.2 Malliin perustuva seuranta 8

3.3 Reunojen avulla seuranta 8

3.4 Näennäinen liike 9

4 Pohdinta 9

4.1 Käyttötarkoitukset 10

5 Viiteluettelo 12

1 Johdanto

Kuvissa olevien piirteiden tunnistamisen kysyntä on ollut kasvussa uusien pienempien ja parempien kameroiden myötä. Erityisesti puhelimissa kamerat ovat tarpeeksi hyviä ja itse puhelimet tarpeeksi nopeita mahdollistaakseen kuvantunnistuksen ja kuvien piirteiden seurantaa. Näistä seikoista johtuen kohteenseurannan menetelmille alkaa olla kysyntää. Kohteenseurantaan ei kuitenkaan ole vielä yhtä parasta ratkaisua, joten tässä tutkielmassa käsitellään erityisesti eri kohteenseurantamenetelmiä ja näitä hyödyntävien algoritmien välisiä eroja hyötyineen ja haittoineen. Tulokset on kerätty viitatuista tutkielmista osana systemaattista kirjallisuuskatsausta kohteenseurantaan.

Kohteenseurannasta voidaan tunnistaa viisi eri seurantatapaa [Lychkov *et al.*, 2018]. Näistä tavoista tässä dokumentissa tullaan käsittelemään neljää: *Alueen seurantaa* (Blob tracking myöhemmin BT) hyödyntävä menetelmä, *malliin perustuva* (Model based myöhemmin MB) menetelmä, *reunan muotoja seuraava* (Contour tracking myöhemmin CT) menetelmä ja *kuvan liikkeen lukeva* (Optical Flow myöhemmin OF) menetelmä. Näiden lisäksi löytyy vielä syväoppimisen tapa, jossa käytetään *neuroverkkoja* (Neural Network), mutta tämä jätettiin tutkielmasta pois hyvin erilaisen toimintatapansa vuoksi.

Tutkielman tavoitteena on koota näihin seurantatapoihin kuuluvia algoritmeja ja verrata näitä keskenään. Tämän avulla yritetään määritellä algoritmien ideaaliset käyttökohteet ja kohteet, joissa algoritmit toimivat heikosti. Tässä tullaan huomaamaan kolme näkökohtaa. Ensimmäiseksi uudemmat algoritmit sisältävät yleensä kattavammin ominaisuuksia. Toiseksi sopiviin käyttötarkoituksiin kuuluu: BT soveltuu tuntemattomien kohteiden seurantaan, MB-menetelmä tarjoaa erinomaisen tarkkuuden kohteen ennakkotiedon kanssa, CT-menetelmän nopeus ja OF tuo tilan ymmärtämisen tukemaan muita menetelmiä. Kolmanneksi huomataan hybridialgoritmien tarjoama hyöty, jolloin kaksi algoritmia yhdessä saavuttaa paremman tuloksen kuin kumpikaan algoritmeista erikseen.

Tutkielma on jaettu neljään päälukuun. Ensimmäinen eli tämä luku Johdanto antaa tietoa tutkielman sisällöstä ja käytetyistä termeistä. Luvussa 2 määritellään mitä eri kohteenseurannan menetelmiä on olemassa ja lisäksi kerrotaan näiden toimintatavasta. Tämä luku on jaettu alilukuihin, joissa jokaisessa käsitellään yhtä kohteenseurannan menetelmää. Luvussa 3 kerrotaan valittujen algoritmien tuloksista tutkimuksissaan ja mitkä ovat kyseisen algoritmin hyödyt tai heikkoudet. Luvussa 4 analysoidaan tuloksia.

Käytettyjä termejä: *konenäkö* on kameroiden ja niiden avulla muodostetun kuvan kautta tapahtuva kappaleen tai ympäristön hahmon tunnistus [MOT]. *Kohteenseuranta* on videokuvassa olevan tietyn piirteen sijainnin löytäminen kuvasta toiseen. Kohteenseurantamenetelmien nimet ja näiden hyödyntämät muut tavat esitellään omissa luvuissaan.

2 Menetelmät

Alussa on yksinkertaistettu esimerkki kohteenseurannan yleisestä toiminnasta. Videosta halutaan eristää jokin tietty piirre esim. auto. Tämä piirre on haluttu kohteemme. Kohteen löytämiseksi käytetään kuvantunnistamisen menetelmiä eristämään itse kohde ja tätä ympäröivä tausta. Tähän eri menetelmillä on omat tapansa. Tämän jälkeen voidaan kohdetta alkaa seurata samalla merkaten kohde tätä *ympäröivillä rajoilla* (Bounding box). Jokaisen uuden kuvan kohdalla määritellään kohteen sijainti edellisiä kuvia hyödyntäen. Näin liikkuvassakin kuvassa kohteen rajojen avulla nähdään kohteen liike kuvasta toiseen.

2.1 Blob Tracking eli alueen seuranta

BT:n ideana on erottaa selvästi muista kuvan piirteistä erottuva alue kohteeksi. Tämä voidaan toteuttaa värin perustella, koon mukaan tai muuten muusta sisällöstä selvästi erottavan tekijän avulla.

Yksinkertaisimmillaan BT on paikoillaan pysyvässä kamerassa, joka mahdollistaa samana pysyvän taustan poistamisen. Tällöin liikkuva osa määritellä halutuksi kohteeksi epämääräisenä alueena. Tätä kuitenkin haittaa kolme tekijää. Ensimmäiseksi, taustan liike voidaan ymmärtää uudeksi kohteeksi. Toiseksi, kohteen peittyessä tarkkuus kärsii. Viimeiseksi, kahden seuratun alueen tavatessa näiden määrittely erilleen on vaikeaa [Bedruz *et al*., 2017].

Seuraavaksi esitellään kahden BT menetelmää hyödyntävän algoritmin toimintatapa Bedruz *et al*. [2017] ja Lu & Feng [2010] mukaan. Molemmissa on huomattavissa yhtäläisyyksiä, sillä molemmat käyttävät ”Mean-Shift”-algoritmin (MS) variaatioita seurantaan. MS-algoritmi kokoaa todennäköisyydet samanlaisuudesta yhteen ja etsii näiden muodostaman huipun. Kuitenkaan algoritmeja ei voi pitää identtisinäkään Bedruz *et al.* [2017] keskittyessä enemmän taustan mallintamiseen.

Bedruz *et al*. [2017] autojen seuranta-algoritmi vaatii esikäsittelyn, joka koostuu neljästä osasta: Ensiksi mallinnetaan tausta ”Mixture of Gaussian” (MOG) ja ”K-Nearest Neighbor” (K-NN) menetelmin. MOG-mallinnus määrittelee tuntemattomien osajoukkojen osuuden kokonaisuudesta ja K-NN-mallinnus määrittelee tuntemattoman pisteen tämän vieressä olevien tunnettujen pisteiden perusteella. Mallinnuksen jälkeen poistetaan kuvan selvä tausta jättäen jäljelle vain mahdolliset kohteet. Kolmanneksi tunnistetaan jäljelle jääneestä mustavalkokuvasta autot laskemalla pikselirykelmien keskipiste. Lopuksi kuvaa käsitellään, kunnes autot ovat yhtenäisiä alueita.

Itse seurantavaihe Bedruz *et al*. [2017] algoritmissa koostuu kolmesta osasta: MS (Mean-Shift) avulla kohteenseuranta, samanlaisuuden testaus ja lopuksi taustan vaikutuksen vähennys. MS:n etuna on lähes automaattinen kohteenseuranta, algoritmille tarvitsee vain

kertoa, minkä rykelmän keskipistettä seurataan. Seuraavassa osassa Bhattacharyya-laskua käytetään löytämään samanlaisuus kohteen ja ehdokkaan välillä. Lopuksi osa taustaa saattaa sekoittua kohdealueeseen, joten tämän vaikutuksen vähentämiseksi poistetaan taustaksi määriteltyä aluetta. Taustan poistamiseksi tarvitaan tieto taustasta, jota säilytetään ”Background-Weighted histogram” (BWH) avulla.

Samaa ongelmaa kohteen ympärillä olevan taustan sisällyttämisestä käsittelee Lu & Feng [2010]. Samanlaisuuden säilyttämiseksi he kehittivät *taustan painon perusteella samanlaisuuden mittauksen* (Background-Weighted similarity measure). Tämän uuden samanlaisuuden laskutavan avulla taustan vaikutusta mallissa voidaan vähentää, mikä johtaa parempaan tarkkuuteen seurannassa. Heidän algoritminsa sisältää samanlaisuuksia Bedruzin käyttämään MS (Mean-Shift) algoritmiin, kuitenkin erot yleisemmän algoritmin toimintaan tekevät heidän ”Centroid iteration”-algoritmistansa yksinkertaisemman, mutta samalla tarkkuus säilyy.

2.2 Model Based eli malliin perustuva seuranta

Vaativampaan seurantaan voidaan hyödyntää 3D mallia halutusta kohteesta. Tässä tunnistetaan kohde vertaamalla kohdetta ennalta määriteltyyn malliin. Tämä voi olla 3D CAD malli tai 2D esimerkki [Pressigout ja Marchand 2006]. Mallin etuna on tarkkuus. Vielä paremman tarkkuuden saamiseksi omassa tutkimuksessaan he yhdistävät kaksi malliin perustuvaa menetelmää ja seuraavat kohdetta ulkonäön ja reunojen avulla saatujen mallien avulla.

Suuri osa malliin perustuvista algoritmeista hyödyntää Kalman suodattimia (Linear Kalman Filter tai Extended Kalman Filter) arvioimaan kohteen oikeaa sijaintia, johon hyödynnetään ajan kanssa saatuja monen laskukerran tuloksia [Pressigout ja Marchand 2006]. Kalman suodattimet koostuvat kahdesta vaiheesta: kohteen sijainnin arvio ennakkoon ja sijainnin päivitys jälkikäteen. Tuloksen perustuminen pidemmältä ajalta saatuun tietoon tarjoaa suodattimelle myös edun tarkkuudessa verrattaessa vain yhden laskukerran tarjoamiin tuloksiin.

Malliin perustuvalla algoritmilla on myös etu muihin algoritmeihin, joissa liike määritellään yhden pisteen avulla, sillä malliin perustuva algoritmi tarjoaa tiedon muodosta, ulottuvuuksista ja koosta [Aftab *et al*., 2019]. Tämä mahdollistaa monta liikkeen laskutapahtumaa samanaikaisesti.

Seuraavaksi esitellään kaksi MB-menetelmään perustuvaa algoritmia Aftab *et al*. [2019] ja Pauwels *et al*. [2016]. Aftab *et al.* [2019] kehittämä seuranta hyödyntää yleistä Kalman suodatin-menetelmää ja lisää oman mallinnuksensa tämän tueksi, mutta Pauwels *et al.* [2016] toteavat Kalman suodattimen liian rajalliseksi ja valitsee suodattimensa OF testin tulosten mukaan.

Laajennettu kohteenseuranta (Extended object tracking myöhemmin EOT) algoritmi [Aftab *et al*., 2019] hyödyntää kahta eri menetelmää seurantaan: kohteen keskipisteen ja muodon avulla. Kohteen muutokset seurantavaiheen aikana eivät haittaa, sillä algoritmi ymmärtää myös erot muodossa. Tutkimuksessa jatketaan rekursiivista ”spatio-temporal Gaussian process” STGP mallinnusta tehden tästä reaaliajassa toimivan mallinnuksen muuttuville kohteille. Mallin luotuaan tämä syötetään *laajennetulle Kalman suodattimelle* (EKF), joka hoitaa seurannan. STGP-EKF mahdollistaa myös mallin muodostamisen kohdetta seurattaessa ilman tarkkaa ennakkotietoa.

Malliin perustuva algoritmin tehtävät voivat olla myös erittäin haastavia. Tutkimuksessaan Pauwels *et al*. [2016] esittelevät algoritmin, joka pystyy seuraamaan 150 eri kohteen *asentoa* (Six degrees of freedom) samanaikaisesti reaaliajassa. Algoritmi toimii hieman erilailla kuin yleisimmät malliin perustuvat algoritmit. Yleisesti käytetään vain itsestään selviä mallin piirteitä, kuten avainpisteet, ääriviivat, reunat ja muoto. Sen sijaan Pauwels *et al*. [2016] algoritmi yrittää säilyttää kaiken tiedon mallin ulkonäöstä ja tämän avulla yhdistää kohde ja malli keskenään. Toimiakseen algoritmi tarvitsee videokuvan kahdesta kamerasta saadakseen syvyystietoa mallille tai syvyystiedon (RGB-D) antavan erikoiskameran. Algoritmi toimii monen tavan hybridinä hyödyntäen myös muita menetelmiä, kuten seurantapisteitä ja OF menetelmää. Pisteet asetetaan mallin perusteella kohteen reunoihin tukemaan seurantatarkkuutta. Tämän lisäksi *riittävän liikkeen* (dense motion) saamiseksi käytetään OF:ta. Mallinnettua virtualisoitua liikerataa ja OF:ää käytetään rinnakkain. Näiden erotessa toisistaan liikaa joudutaan kohde etsimään uudestaan. Algoritmi käyttää myös ”M-Estimation”-laskelmaa, joka vähentää ääritapausten vaikutusta seurannassa.

2.3 Contour Tracking eli reunojen seuranta

Tämä menetelmä mahdollistaa seurannan etsimällä kohteen reunat, jonka jälkeen kohteen reunat merkitään. Tähän yleinen tapa on määritellä kohteen ympärille *rajat* (bounding box). Myös rajojen käytön ongelmana on taustan sisällytys rajojen sisälle ja näin vaikutus kohteeseen. Kuitenkin CT:llä on myös omat etunsa. CT tarjoaa tiedon sekä sijainnista että muodosta [Heo *et al*. 2017]. Tämän lisäksi reunojen avulla seuranta on verrattain kevyt toimenpide.

Seuraavaksi tarkastellaan tarkemmin kahden CT menetelmään perustuvan algoritmin toimintaa Panin *et al*. [2006] ja Heo *et al*. [2017]. Panin *et al.* [2006] määrittelevät kohteen asennon reunojen perusteella ja käyttävät tämän jälkeen Kalman suodatinta seurantaan kun taas Heo *et al.* [2017] kehittävät oman uuden Bayes-tyyppisen todennäköisyyden määrittelyn, johon myös Kalman suodattimet kuuluvat. Uusi määrittelytapa koostuu kolmesta seurattavasta osasta: kontrasti, sijainti ajan perusteella ja värit.

Panin *et al*. [2006] kehittävät reunanseuranta-algoritmin, joka parantaa olemassa olevaa reaaliaikaista algoritmia (Contracting Curve Density eli CCD) tehden tästä vielä nopeamman. Kohteen kadotessa algoritmi pystyy aloittamaan alusta ja etsimään kohteen esimerkkikuvan perusteella. Tämä on kuitenkin raskasta, joten seurantaan hyödynnetään mieluummin kohteen rajoja kokonaan uudelleen etsimisen sijaan. Kohteen sijainti varmistetaan myös Kalman suodattimella. Suodattimen antaman sijainnin erotessa liikaa reunoilla seuratun kohteen sijainnista algoritmi aloittaa alusta ja hakee kohteen kuvasta uudestaan esimerkin avulla. Algoritmi on erittäin kevyt verrattuna muihin tässä tutkielmassa mainittuihin algoritmeihin. Kyseinen algoritmi mahdollisti tarkan kohteenseurannan jo vuonna 2006 aikansa peruskoneilla.

Reunantunnistuksessa yleensä oletetaan rajojen pysyvän samoina jolloin kohde ei voi muuttua seurannan aikana. Tähän avointa ratkaisua ehdottaa [Heo *et al*. 2017]. Heidän ratkaisunsa perustuu kahteen ominaisuuteen. Ensimmäiseksi he hyödyntävät *tilan tallennus* (state-space) mallia, joka sisältää sekä kohteen liikkeen jäykkänä ilman muutoksia että liikkeestä seuraavien muutoksien kanssa. Näiden erottaminen toisistaan mahdollistaa erikseen harjoittamisen. Tämän lisäksi he hyödyntävät rekursiivista Bayes-arviota, joka mahdollistaa epävarman todennäköisyyden päivittämisen uudella tiedolla. Täten algoritmi pystyy tunnistamaan kohteen muutokset ja arvioimaan virheitään.

2.4 Optical Flow eli liike kahden kuvan välillä

OF kertoo näennäisen liikkeen kahden kuvan välillä. Tämä toimii erityisen hyvin ympäristön liikkeiden erottamiseen antaen selvän eron eri kohteiden välillä. Nämä voivat olla esim. puu, taivas jne. Tämä kaikkien kohteiden erittely mahdollistaa OF:n toiminnan myös liikkuvassa kamerassa.

Kahdessa tutkimuksessa tätä käytettiin ainoastaan tukemaan MB algoritmia tarkistamalla mahdollisia virheitä [Pauwels *et al*., 2016 ja Harmanci *et al*., 2016]. Molemmissa hyödynnettiin *tilaan perustuvaa liikettä* (Phase-based OF), jossa kaksi tai useampia peräkkäisiä kuvia määritellään verrattavaksi kohteeksi. Tämä vähentää virheellistä liikkeen arviointia verrattuna yleisempään Lucas-Kanade-algoritmiin, joka vertaa vain peräkkäisiä kuvia.

Seuraavaksi esitellään kaksi OF-menetelmään perustuvaa algoritmia: Lychkov *et al*. [2018] ja Sajid *et al*. [2019]. Molemmat ovat verrattain uusia lähestymistapjoa, johtuen OF-laskennan vaatimasta prosessointitehosta reaaliaikaiseen toimintaan. Lychkov *et al.* [2018] käyttää yleistä Lucas-Kanade algoritmia seurantaan. Sajid *et al.* [2019] taas keskittyy vain etualan ja taustan erottamiseen ja hyödyntää tähän aikaisemmin mainittua MOG-mallinnusta.

Ensimmäisessä menetelmässä hyödynnetään *osittaista liikkeen tunnistusta* (Sparse optical flow), joka on nopeampi, mutta samalla epätarkempi versio *täydestä liikkeen tunnistuksesta* (Dense optical flow). Se koostuu selvien piirteiden tunnistamisesta pisteiden avulla rajatulla alueella ja näiden piirteiden seurannasta tulevissa kuvissa [Lychkov *et al*. 2018]. Kuitenkin tässä on omat heikkoutensa. Ensimmäiseksi, kohteen peittyessä seurantapisteet katoavat. Toiseksi, pisteet liikkuvat hieman kuvasta toiseen, jolloin nämä seurantapisteet voivat seurata virheellisesti taustaa kohteen sijaan. Näiden ongelmien korjaamiseksi Lychkov *et al*. [2018] ehdottavat seurantapisteiden palautusta. Kohteen havaittuaan algoritmi piirtää tämän ympärille neliön ja neliön sisälle käyttäjän valitseman määrän seurantapisteitä. Pisteiden joutuessa neliön ulkopuolelle nämä tuhotaan ja luodaan uudelleen. Toisin sanoen algoritmi pystyy korjaamaan omia virheitään parantaen tarkkuutta.

Toinen menetelmä keskittyy OF:n etuun erottaa tausta ja etuala toisistaan. Tätä käsittelevät Sajid *et al*. [2019] tutkimuksessaan seurannasta vapaasti liikkuvista kameroista kuten kännykät. Algoritmissa hyödynnetään sekä liikettä, että mallia. Toiminta ei myöskään vaadi ennakkotietoa ympäristöstä. Tämä onnistuu määrittelemällä *suuret liikkeet* (dense motion field) kahden kuvan välille, jonka avulla saadaan todennäköisyys etualasta jokaisen pikselin kohdalla. Suuret liikkeet kameran kuvassa häiritsevät yleistä *taustan poisto* (Background Subtraction)-menetelmää. Tätä varten luodaan malli taustalle ja etualalle. Mallien avulla voidaan erottaa kameran ja kohteen liikkeen erilaisuus. Toinen uusi ominaisuus on ”Mega-Pixel” tässä samanväriset vierekkäiset pikselit otetaan mukaan ja käytetään keskiarvoa näiden välillä laskettaessa.

3 Tulokset

Tuloksissa tarkastellaan aikaisemmin mainittujen tutkimuksien algoritmeja keskenään ja näiden sisältämiä ominaisuuksia. Taulukossa 1 esitetään neljä esille noussutta tärkeää ominaisuutta. Nämä ovat: reaaliaikainen toiminta, sietääkö algoritmi kohteen peittoa, voidaanko kameraa liikuttaa seurannan aikana. Näiden lisäksi viidentenä annetaan mahdollinen prosenttiosuus jolloin kohteenseuranta onnistui seuraamaan kohdetta onnistuneesti.

Taulukosta nousee esille neljä algoritmia, jotka sisältävät mainitut kriteerit parhaiten, yksi jokaiselle menetelmätyypille. Nämä ovat alueen seuranta MS-algoritmin avulla [Bedruz et al., 2017], malliin ja liikkeeseen perustuva hybridialgoritmi [Pauwels et al., 2016], reunan seurannan muuttuville kohteille mahdollistava algoritmi [Heo et al., 2017] ja taustan poisto liikkeen perusteella vapaassa kamerassa [Sajid et al., 2019]. Edellä mainituilla algoritmeilla puuttuu seuratuista ominaisuuksista vain yksi, joko puuttuva kriteeri tai oikein seurattujen tilanteiden osuus. Näistä vahvimmiksi voisi olettaa algoritmit joilla kaikki kriteerit täyttyvät, mutta tämän tekee vaikeaksi tiedon puute epäonnistuneiden seurantatilanteiden suhteesta onnistuneisiin. Tieto algoritmien kattavuudesta, tarkkuudesta ja

heikkouksista tekevät Pauwels et al. [2016] ja Bedruz et al. [2017] algoritmeista vakuuttavammat muihin algoritmeihin verrattuna.

	Reaaliaikainen toiminta	**Sietää kohteen peittoa**	**Liikkuva kamera**	**Kohteen muutos**	**Tarkkuus seurattaessa**
BT Centroid Iteration Lu ja Feng, 2010	X		X	X	
BT Bedruz *et al.*, 2017	X	X		X	91%
MB Pressigout ja Marchand, 2006	X	X			
MB+OF Pauwels *et al.*, 2016	X	X	X		64-100%*
MB Aftab *et al.*, 2019	X			X	
CT Panin *et al.*, 2006	X	X			
CT Heo *et al.*, 2017	X	X	X	X	
OF Lychkov *et al.*, 2018	X	X			43-71%
OF Sajid *et al.*, 2019**	X	X	X	X	

* Peitteestä riippuen hybridi MB+OF ratkaisulla. Pelkkä MB 42-77%. Pelkkä OF 57-93%
** Erottaa taustan ja etualan kohteet mahdollistaen helpon seurannan esim. BT menetelmillä.

Taulukko 1. kohteenseurannan algoritmien ominaisuudet

3.1 Alueen seuranta

BT:hen perustuva ”Centroid Iteration”-algoritmi [Lu ja Feng, 2010] onnistui seuraamaan kohdetta epätarkoissakin kuvissa erinomaisesti. Testeissä verrattiin MS algoritmia eri samanlaisuuden laskutavoilla ja uutta algoritmia *taustan painotetulla laskulla* (Background Weighted Similarity Measure) keskenään. Saatu tarkkuus oli parempi kuin kilpailevien Bhattacharyya- ja Kullback-Leibler-laskutapojen. Uusi algoritmi on hyvin samanlainen kuin MS, mutta algoritmi onnistuu olemaan myös hieman nopeampi. Kuitenkaan algoritmi ei ole täydellinen. Heidän testituloksissaan kohde ei ollut koskaan osittain tai täysin peitettynä.

Toinen BT algoritmi [Bedruz *et al*. 2017] taas hyödyntää MS algoritmia ja Bhattacharyya samanlaisuuden laskelmaa löytämään erot kuvien välillä. Tutkimuksessa todetaan MOG (0.0035 s) taustan mallinnuksen toimivan nopeammin kuin K-NN (0.0293 s). Rykelmien käsittely vie yhtä lyhyen ajan (0.0037 s). Algoritmi seurasi kohdetta oikein 91.209% tapauksista. Uutuutena algoritmi tuo alueen seurantaan myös osittain peitetyn kohteen seurannan MS-algoritmin avulla.

3.2 Malliin perustuva seuranta

Kahteen malliin perustuvan algoritmin hybridi [Pressigout ja Marchand, 2006] seuraa sekä ulkonäön että reunojen avulla saatuja malleja. Yhdistämällä nämä saatiin parempi tarkkuus, kuin algoritmien toimiessa erikseen. Poikkeuksia saatiin vähennettyä lisäämällä seurantaan ääritapausten poisto (M-Estimation). Tutkimuksessa huomioitiin myös varjot, kohteen kääntyminen ja osittainen peittyminen.

Malliin ja OF perustuva hybridialgoritmi [Pauwels *et al*., 2016] saavuttaa erinomaisen tarkkuuden seurannassa. Ilman esteitä 100%, kuvan artifaktit 96% ja peitettynä 64%. Erikseen algoritmit toimivat hieman heikommin MB 77%, 47%, 42% ja OF 93%, 81%, 57%. Algoritmi kestää myös kameran liikettä, kuvan artifakteja ja osittaista kohteiden peittymistä.

Viimeinen malliin perustuvaa EOT [Aftab *et al*., 2019] algoritmin STGP-EKF mallinnusta verrattiin GP-EKF mallinnukseen yli 100:ssa Monte Carlo kohteenseuranta-ajossa. Tuloksissa STGP-EKF johti parempaan tarkkuuteen kaikissa testeissä paitsi yhdessä, jossa tulos oli sama. Algoritmi oli myös tarpeeksi nopea reaaliaikaiseen seurantaan. Tulos parani erityisesti kohteissa, joissa kohteen muoto muuttuu seurannan aikana; esim. kävelijän käden liikkeet.

Kaikki valitut malliin perustuvat algoritmit suoriutuivat seurannasta reaaliajassa. Kuitenkin tämä aikarajoite on ollut ennestään esteenä malliin perustuvien kohteenseuranta-algoritmien käytölle reaaliajassa [Lychkov *et al*., 2018].

3.3 Reunojen avulla seuranta

CT-pohjainen algoritmi [Panin *et al*., 2006] onnistuu seuraamaan kohdetta paikallaan olevassa kamerassa. Saatu tarkkuus on hyvä. Kyseinen algoritmi sietää osittaista peittoa, valon muutoksia ja muutoksia kohteen asennossa. Kohteen muodon on kuitenkin pysyttävä samana ja seurattavan puolen oltava kohti kameraa.

Toinen CT:hen perustuva algoritmi [Heo *et al*. 2017] tarjoaa parannetun tarkkuuden verrattuna tavanomaisiin rajoja hyödyntäviin algoritmeihin (Condensation, Snapcut, Hough Tracking) häviten vain kerran Snapcut-algoritmille etäisyydessä oletettuun sijaintiin (Chamfer Distance). Tämä johtuu uuden algoritmin kyvystä sietää koon muutoksia ja

360° kuvan kiertoa. Menetelmä selviää myös kohteen osittaisesta peittymisestä ja algoritmi pärjää myös määritettäessä isompi alue seurattavaksi, joka sisältää myös taustaa. Tällaisessa rajauksessa algoritmi voittaa neljä tavanomaista jäykän seurannan algoritmia, jotka sisältävät myös suositun Lucas-Kanade-algoritmin (ELK). Tämä pärjää myös hyvin edeltävään tapaan rajattuja muuttuvaa kuvaa seuraavia (Hough Tracking, Superpixel tracking, Dynamic Graph Tracking) algoritmeja vastaan. Algoritmi kestää myös kohteen osittaista peittoa ja kameran liikettä

3.4 Näennäinen liike

OF-menetelmään perustuva algoritmi [Lychkov *et al*., 2018] onnistui säilyttämään tarkkuutensa myös osittain peitettynä. Muutokset alkuperäiseen ”Lucas-Kanade sparse optical flow” algoritmiin lisäsi parhaimmillaan 7% tarkkuutta ja heikoimmillaan johti samaan tulokseen alkuperäisen kanssa. Kuitenkin alkuperäisen algoritmin painotus nopeuteen ja kohteen peittyminen johti heikompaan tarkkuuteen (43-71%) kohteenseurannassa. Algoritmi kestää osittaista peittymistä, kohteen kietoa ja koon muutoksia.

Vapaasti liikkuvalle kameralle suunniteltu OF-menetelmän algoritmi [Sajid *et al*.,2019] onnistui seuraamaan kohdetta reaaliajassa. Algoritmin uusi taustan määrittely päihitti aikaisemman RANSAC-menetelmän. Heidän MOG-mallinnuksensa saavutti yleensä paremman tarkkuuden kuin kuusi (Sheikh, Narayana, Zamalieve, Kwak, Lim, Elqursh) muuta taustanpoistomenetelmää. Mahdollinen ongelma ilmenee kohteen ollessa yli 70% kuvasta, joka rikkoo heidän asettamaansa oletusta taustan enemmistöstä. Samanvärinen auto taustassa kuin seurattava aiheuttaa myös virheen. Algoritmi tarjoaa erinomaisen kohteiden erottelun liikkuvassa kamerassa, sietää hyvin kohteiden peittoa ja seuraa tennispelaajan liikkeitä kohteen muutoksista riippumatta.

4 Pohdinta

Tuloksista voidaan huomata kohteenseurannan hyvä tarkkuus, mutta muutokset ympäristössä ja erityisesti kohteen peittyminen heikentää tuloksia rajusti. Käsitellyt algoritmit ovat edistystä kyseisissä asianomaisissa tutkimuksissa verrattuihin algoritmeihin. Ajan myötä kehitys on oletettavaakin, mutta vanhemmatkin algoritmit onnistuivat seurannassa yksinkertaisissa tilanteissa. Joissain tutkimuksissa olisi ollut hyvä olla lisää testausta, kuten ”Centroid Iteration”-algoritmi [Lu ja Feng, 2010] vaikutti erinomaiselta omissa testeissään. Kuitenkin tuloksista puuttui kohteen peittymisen testaus, joka todennäköisesti heikentäisi tulosta.

Tutkimuksen aikana esille nousi neljä yleistynyttä algoritmia, jotka mainittiin monen tutkimuksen kohdalla. Näihin kuuluu Kalman suodattimet (LKF, EKF), Mean Shift, Lucas-

Kanade ja Kullback-Leibler. Monet tässäkin tutkimuksessa tarkastelluista algoritmeista hyödynsivät joko suoraan tai variaatiota näistä algoritmeista.

Tuloksien perusteella hybridialgoritmit vaikuttavat erittäin lupaavilta. Monessa varsinkin uudemmassa tutkimuksessa ei ollutkaan enää käytössä pelkästään yhtä hyvää menetelmää vaan näissä hyödynnettiin monen menetelmän yhdistelmiä. Syynä tuntuivat olevan menetelmien eri toimintatapojen vahvuudet ja heikkoudet, jolloin yleisesti heikompi menetelmä voi kuitenkin toimia paremmin erikoistilanteissa tukemassa toisen menetelmän toimintaa.

Kohteenseurannalle ideaalisissa olosuhteissa (hyvä kuvanlaatu, valaistus ja suora näköyhteys kohteeseen) kaikki käsitellyt algoritmit onnistuisivat työssä helposti. Tämän takia kohteenseurannan kehitys onkin edennyt kohti virhetilanteiden ratkaisua. Erityisesti kohteen peittyminen ja taustan vaikutus kohteeseen ilmeni ongelmaksi monen tutkimuksen kohdalla. Näiden lisäksi kaikkein uusimmat algoritmit eivät enää tyydy oletukseen kohteen pysymisestä samana vaan osaavat mukautua tämän muutoksiin ajon aikana.

Kaikilla algoritmeilla oli hieman erilaiset tavat ratkaista ongelmiaan. Kuitenkin osa myös päätyi hyvin samankaltaisiin ratkaisuihin, kuten ”Mega-Pixel”, ”Background-Weighted similarity measure” ja ”Background-Weighted histogram” taustan vaikutuksen vähentämiseksi. Näissä taustan vaikutusta pienennetään ottamalla suurempi osa kohdealueesta tarkasteluun ja käytetään alueen keskiarvoa laskelmiin. Näin annetaan kohteen keskellä olevalle tiedolle etu reunoilla olevaan verrattuna.

4.1 Käyttötarkoitukset

BT sopii yksinkertaisen toimintatapansa ansiosta tilanteisiin, joissa seurattavasta kohteesta ei tiedetä ennakkoon. Hyvä esimerkkitilanne on valvontakamerassa. Tällöin BT-algoritmin etu monen kohteen nopeasta seurannasta taustan poistamisen avulla pääsee loistamaan ja vältetään ongelma kameran liikkeen suhteen. Tämä voi seurata esim. autoja tai ihmisten silhuetteja.

MB on raskaampi vaihtoehto, joka vaatii ennakkotietoa kohteesta. Kuitenkin tutkimuksissa EOT:ssa tämä malli voitiin muodostaa kohdetta seurattaessa STGP:n avulla. MB-seurantaa pystytään hyödyntämään tehtaissa tuotteiden valvontaan. Tuotteen täsmäys malliin vähentää virheitä. Toinen mahdollisuus voisi olla kännykälle tehty virtuaalitodellisuuden sovellus, jonka avulla annetaan kaupan tuotteista lisätietoa. Erityisesti Pauwelsin algoritmi mahdollistaa edellisen seuraten monia kohteita liikkuvassakin kamerassa.

CT on nopea seuranta kohteelle, jolla on selvät rajat. Kuitenkin uudemmassa tutkimuksessa tämä vaatimus oli pienempi. Samalla uudemman tutkimuksen perusteella CT on myös erinomainen vaihtoehto ihmisten seurantaan. Kohteen muuttuminen seurattaessa otettiin huomioon ja tehtäessä rajaus olkapäistä ylöspäin algoritmi mukautui seuraamaan

pään piirteitä taustan vaikutuksesta huolimatta. Monen kohteen seurantaa tässä ei mainittu, mutta mahdollinen käyttökohde voisi olla urheilutapahtumissa urheilijoiden seuranta videolähetyksessä. Tämä antaisi lisätietoa kentän pelaajista näiden vieressä reaaliajassa.

Myös OF mahdollistaa liikkuvan kameran käytön reaaliajassa, mutta tutkimuksissa tätä käytettiin yleensä vain tukemaan toisen algoritmin toimintaa tarkistamalla näiden virheitä. Näin ollen hybridiratkaisut joissa OF oli mukana, olivat vakuuttavia tarkkuudellaan, mutta erikseen tulokset heikkenivät. Uusin OF-ratkaisu kuitenkin vaikutti toiminnallaan auton katolla ja ihmisiä seurattaessa. Tämä mahdollistaa monenlaisen seurannan ja vaikutti kaikkein yleishyödyllisimmältä käsitellyistä algoritmeista, joskin verrattain monimutkainen.

Kohteenseurannan tulevaisuus näyttää hyvältä uusine algoritmeineen aina paremmin tuloksin. Kiinnostus kehitykselle on varmasti kasvussa myös itseajavien autojen myötä. Näitä seurantamenetelmiä voidaan hyödyntää juuri autojen tutkissa. Ainakin OF-algoritmi on erinomainen tukemaan auton ympäristön tunnistamista. Tämän lisäksi EOT-menetelmät ovat jo nyt käytössä tutkien tukena [Aftab *et al*., 2019].

5 Viiteluettelo

I. I. Lychkov et al., ‘Tracking of Moving Objects With Regeneration of Object Feature Points’. IEEE, 2018 Global Smart Industry Conference (GloSIC), pp. 1–6, Nov. 2018 [Online]. Available: 10.1109/GloSIC.2018.8570061.

G. Panin et al., ‘An Efficient and Robust Real-Time Contour Tracking System’. IEEE, Fourth IEEE International Conference on Computer Vision Systems (ICVS’06), p. 44, 2006 [Online]. Available: 10.1109/ICVS.2006.13.

R. A. Bedruz et al., ‘Real-time vehicle detection and tracking using a mean-shift based blob analysis and tracking approach’. IEEE, 2017 IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), pp. 1–5, Dec. 2017 [Online]. Available: 10.1109/HNICEM.2017.8269528.

W. Aftab et al., ‘Spatio-Temporal Gaussian Process Models for Extended and Group Object Tracking With Irregular Shapes’, vol. 68, no. 3. IEEE, IEEE Transactions on Vehicular Technology, pp. 2137–2151, 2019 [Online]. Available: 10.1109/TVT.2019.2891006.

Pressigout and Marchand, ‘Real-time 3D model-based tracking: combining edge and texture information’. IEEE, Proceedings 2006 IEEE International Conference on Robotics and Automation, pp. 2726–2731, 2006 [Online]. Available: 10.1109/ROBOT.2006.1642113.

K. Pauwels et al., ‘Real-Time Pose Detection and Tracking of Hundreds of Objects’, vol. 26, no. 12. IEEE, IEEE Transactions on Circuits and Systems for Video Technology, pp. 2200–2214, 2016 [Online]. Available: 10.1109/TCSVT.2015.2430652.

S. Heo et al., ‘Open-Contour Tracking Using a New State-Space Model and Nonrigid Motion Training’, vol. 27, no. 11. IEEE, IEEE Transactions on Circuits and Systems for Video Technology, pp. 2355–2366, 2017 [Online]. Available: 10.1109/TCSVT.2016.2592321.

MOT. [Online]. Available: https://mot.kielikone.fi/finelib/netmot.shtml. [Accessed: 2019].

N. Lu and Z. Feng, ‘Centroid Iteration algorithm for image tracking’, vol. 15, no. 2. Springer-Verlag, Pattern Analysis and Applications, pp. 163–174, May. 2012 [Online]. Available: 10.1007/s10044-011-0252-8.

H. Sajid et al., ‘Motion and appearance based background subtraction for freely moving cameras’, vol. 75. Elsevier, Signal Processing: Image Communication, pp. 11–21, 2019 [Online]. Available: 10.1016/j.image.2019.03.003.