

Liikkuva-sovellusprojekti

**Joel Kivelä
Erkki Koskenkorva
Mika Lehtinen
Oskari Leppäaho
Petri Partanen**

Sovellusraportti

Julkinen
Versio 1.0.0
9.6.2014

**Jyväskylän yliopisto
Tietotekniikan laitos
Jyväskylä**

Hyväksyjä	Päivämäärä	Allekirjoitus	Nimenselvennys
Projektipäällikkö	__.__.2014		
Tilaaja	__.__.2014		
Ohjaaja	__.__.2014		

Tietoa dokumentista

Tekijät:

- Joel Kivelä (JK) joel.a.kivela@student.jyu.fi
- Erkki Koskenkorva (EK) erkki.a.koskenkorva@jyu.fi
- Mika Lehtinen (ML) mika.k.lehtinen@student.jyu.fi
- Oskari Leppäaho (OL) oskari.h.leppaaho@student.jyu.fi
- Petri Partanen (PP) petri.m.partanen@student.jyu.fi

Dokumentin nimi: Liikkuva-projekti, Sovellusraportti

Sivumäärä: 50

Tiivistelmä: Liikkuva-projekti kehitti Jyväskylän yliopiston liikuntakasvatuksen laitokselle käyttöliittymän konenäköpohjaiselle liikemittarille. Sovellusraportissa kuvataan projektissa kehitetyn tietojärjestelmän prototyyppi käyttöliittymän, sovelluksen kokonaisu rakenteen, syöte- ja vastetiedostojen, tavoitteiden toteutumisen sekä puutteellisten ja heikkojen toteutusratkaisujen osalta.

Avainsanat: Jatkokehitys, kokonaisu rakenne, komponentit, konenäkö, käyttöliittymä, käytänteet, liikemittari, rajapinnat, tavoitteet, testaus, tiedostot, tietojärjestelmä.

Muutoshistoria

Versio	Päivämäärä	Muutokset	Tekijät
0.0.1	12.5.2014	Muokattiin lukua "Johdanto" projektisuunnitelman [7] pohjalta.	OL
0.0.2	13.5.2014	Aloitettiin luku "Käyttöliittymä" (3).	OL
0.1.0	14.5.2014	Kirjoitettiin luku "Käyttöliittymä" (3) loppuun. Muokattiin lukua "Sovelluksen rakenne" projektisuunnitelman [7] pohjalta.	OL
0.1.1	19.5.2014	Korjattiin lukua "Käyttöliittymä" (3) vastaavan ohjaajan palautteen perusteella.	OL
0.2.0	20.5.2014	Korjattiin lukua "Käyttöliittymä" (3) vastaavan ohjaajan palautteen perusteella. Lisättiin lukuun alaluvut "Pääikkunan komentovalikko" (3.13) ja "Tilarivi" (3.17). Päivitettiin ja korjattiin muotoiluja myös lukuun "Sovelluksen kokonaisrakenne ja rajapinnat" (4). Lisättiin otsikot puuttuville pääluvuille.	OL
0.2.1	21.5.2014	Kirjoitettiin luvut "Ulkopuoliset komponentit" (4.2), "Olennaisimmat muutokset toteutusratkaisuissa" (4.3), "Syöte- ja vastetiedostojen kuvaus" (nykyisessä versiossa jaettu lukuihin 4.4 – 4.6), "Heikot ja puutteelliset toteutusratkaisut" (5.3) sekä "Jatkokehitykseen suositellut toiminnallisuudet" (siirretty nykyisessä versiossa lukuun 5.3)	OL
0.3.0	22.5.2014	Täydennettiin lukuja "Syöte- ja vastetiedostojen kuvaus" (nykyisessä versiossa jaettu lukuihin 4.4 – 4.6), "Heikot ja puutteelliset toteutusratkaisut" (5.3) sekä "Jatkokehitykseen suositellut toiminnallisuudet" (siirretty nykyisessä versiossa lukuun 5.3). Kirjoitettiin luvut "Käytänteet" (poistettu nykyisestä versiosta), "Vaatimusten toteutuminen" (5.1) ja "Yhteenveto" (7).	OL
0.4.0	27.5.2014	Muokattu dokumentin kirjoitusasua kauttaaltaan. Täydennetty lukuja 5, 6 ja 7.	OL

Versio	Päivämäärä	Muutokset	Tekijät
0.4.1	27.5.2014	Muokattu lukuun "Termit" (2) korjaukset, jotka oli tehty projektiraporttiin. Korjattu muutamia automaattisen oikoluvun havaitsemia kirjoitusvirheitä.	OL
0.5.0	3.6.2014	Kirjoitusvirheitä ja sanavalintoja sekä dokumentin asettelua korjattu kauttaaltaan. Muokattu avainsanalistaa. Päivitetty regressiotestauksen tulokset lukuun "Suoritetut testaukset ja niiden tulokset" (5.2). Lisätty lukuun "Heikot ja puutteelliset toteutusratkaisut" (5.3) maininta helpommista mahdollisuuksista toteuttaa yhden videokameran kalibrointi sekä kameran kierron viemisestä CAVAPA-ohjelmalle väärällä tavalla. Poistettu luvussa ollut maininta Windowsissa kaatuilusta. Jaettu luvussa olleet listat omiksi luvuikseen. Poistettu luvusta "Aktiivisuusdatan kuvaajien välilehteä koskevat kehitysiedat" (5.6) maininta merkittyjen ajanhetkien tallentumisesta ainoastaan mittauksen aikana. Lisätty lukuun idea kuvaajien skaalautumistapaan tarjottavista vaihtoehdoista. Lisätty lukuun "Kameran asetusten välilehteä koskevat kehitysiedat" (5.9) maininta ongelmasta kalibroitiasetuksia ladatessa samannimisiä USB-kameroita käytettäessä. Lisätty lukuun "Videonäkymää koskevat kehitysiedat" (5.12) idea mahdollisuudesta kelaata videotiedostoja ruutu kerrallaan sekä maininta Linuxissa sutuksi menevistä kalibroitipisteiden numeroista. Lisätty lukuun "Ohjeita jatkokehittäjälle" (6.1) maininta Doxygenillä generoituun dokumentaatioon tulevista epätoivotuista sivunvaihtoista. Poistettu luvusta "Yhteenveto" (7) luettelo sovelluksen tärkeimmistä näkymistä.	OL
0.6.0	5.6.2014	Korjattu kauttaaltaan kirjoitusasua koskevia virheitä sekä pieniä asiavirheitä.	OL

Versio	Päivämäärä	Muutokset	Tekijät
1.0.0	9.6.2014	Korjattu muutama kirjoitusvirhe ja paranneltu kuvien 3.8 ja 3.10 asettelua. Muutettu luvussa "Aktiivisuusdatan kuvaajien välilehden kehitysideat" (5.6) olevaa, kuvaajien skaalautumista Y-suunnassa koskevan kehitysidean sisältöä vastaamaan paremmin nykyistä toteutusta, ja jaettu se kahdeksi eri ideaksi. Lisätty luku "Kokemuksia käytetyistä kameroista" (6.3).	OL

Tietoa projektista

Liikkuva-projekti kehitti Jyväskylän yliopiston liikuntakasvatuksen laitokselle käyttöliittymän konenäköpohjaiselle liikemittarille.

Tekijät:

- Joel Kivelä (JK) joel.a.kivela@student.jyu.fi
- Erkki Koskenkorva (EK) erkki.a.koskenkorva@jyu.fi
- Mika Lehtinen (ML) mika.k.lehtinen@student.jyu.fi
- Oskari Leppäaho (OL) oskari.h.leppaaho@student.jyu.fi
- Petri Partanen (PP) petri.m.partanen@student.jyu.fi

Tilaaaja:

- Taru Lintunen taru.lintunen@jyu.fi
- Heidi Pasi heidi.pasi@jyu.fi
- Kimmo Suomi kimmo.suomi@jyu.fi
- Ville Tirronen ville.e.t.tirronen@jyu.fi
- Hanna Toivonen toivonen.hanna@yahoo.com

Ohjaajat:

- Jukka-Pekka Santanen santanen@mit.jyu.fi
- Jarkko Vilhunen jarkko.s.vilhunen@student.jyu.fi

Yhteystiedot:

- Sähköpostilistat: liikkuva@korppi.jyu.fi,
liikkuva_opetus@korppi.jyu.fi
- Sähköpostiarkistot: <http://korppi.jyu.fi/kotka/servlet/list-archive/liikkuva/>,
http://korppi.jyu.fi/kotka/servlet/list-archive/liikkuva_opetus/
- Työhuone: Agora C222.2, puh. 040-8053308

Sisältö

1	Johdanto	1
2	Termit	2
2.1	Kohdealueen termejä	2
2.2	Analyysin termejä	3
2.3	Ohjelmistoja ja teknisiä termejä	4
3	Käyttöliittymä	6
3.1	Sovelluksen näkymät	6
3.2	Pääikkuna	7
3.3	Videonäkymä	8
3.4	Videolähteiden hallinta	9
3.5	Mittauksen metatiedot	10
3.6	Kameran asetukset	10
3.7	Aktiivisuusdatan kuvaajat	12
3.8	Loki	15
3.9	Mittauksen hallinta	16
3.10	Uuden merkin lisäys	16
3.11	CSV-tiedostoon tallennus	17
3.12	Kuvaajien asetukset	18
3.13	Pääikkunan komentovalikot	19
3.14	Videovirran avaus	20
3.15	Ohjelman asetukset	21
3.16	Tietoa ohjelmasta	22
3.17	Tilarivi	22
4	Sovelluksen kokonaisrakenne ja rajapinnat	23
4.1	Sovelluksen kokonaisrakenne	23
4.2	Ulkopuoliset komponentit	24
4.3	Olellisimmat muutokset toteutusratkaisuihin	25
4.4	Asetustiedosto	25
4.5	Mittauksen metadatatiedosto	28
4.6	Analyysidatatiedosto	32

5	Tavoitteiden toteutuminen	34
5.1	Vaatimusten toteutuminen	34
5.2	Suoritetut testaukset ja niiden tulokset	35
5.3	Heikot ja puutteelliset toteutusratkaisut	35
5.4	Sovelluksen yleiset kehitysideat	37
5.5	Mittauksen metatietojen välilehden kehitysideat	38
5.6	Aktiivisuusdatan kuvaajien välilehden kehitysideat	38
5.7	Uuden merkin lisäysikkunan kehitysideat	40
5.8	CSV-tiedostoon tallennusikkunan kehitysideat	40
5.9	Kameran asetusten välilehden kehitysideat	41
5.10	Videolähteiden hallintänäkymän kehitysideat	42
5.11	Lokivälilehden kehitysideat	43
5.12	Videonäkymän kehitysideat	43
5.13	Pääikkunan komentovalikoiden kehitysideat	43
6	Ohjeita ylläpitäjälle ja jatkokehittäjälle	45
6.1	Ohjeita jatkokehittäjälle	45
6.2	Ohjeita ylläpitäjälle	45
6.3	Kokemuksia käytetyistä kameroista	46
7	Yhteenveto	47
8	Lähteet	48

1 Johdanto

Liikkuva-projekti kehitti Sovellusprojekti-kurssilla keväällä 2014 Jyväskylän yliopiston liikuntakasvatuksen laitokselle käyttöliittymän konenäköpohjaiselle liikemittarille. Liikemittaria tullaan käyttämään pääasiassa tutkimustarkoituksiin.

Sovellusprojektissa kehitetty käyttöliittymä on osa tietojärjestelmää, johon kuuluu käyttöliittymän lisäksi videon liikemäärää analysoiva ohjelma sekä niiden ja videokameroiden väliset ohjelmakerrokset. Käyttöliittymä toimii videosoittimena näyttäen nauhoitetun materiaalin ja siitä mitatun liikemääräkäyrän. Lisäksi käyttöliittymällä pystytään suorittamaan liikemäärämittauksessa tarvittava kameroiden kalibrointi. Sillä voidaan myös suorittaa algoritmin parametrien säätö. Käyttöliittymällä on myös mahdollista valita mielenkiintoisia aikavälejä ja irrottaa niistä analyysin antamat mittaustulokset käsiteltäväksi muissa ohjelmissa.

Sovellusraportin laatimisessa on hyödynnetty Paatti-projektin sovellusraporttia [1] sekä Liikkuva-projektin projektiraporttia [6], projektisuunnitelmaa [7], vaatimusmäärittelyä [10] sekä testausraportteja [2], [3], [8] ja [9]. Muita liikkuva-projektin dokumentteja ovat luokkadokumentaatio [5] ja testausuunnitelma [4].

Sovellusraportti muodostuu kahdeksasta luvusta. Luvussa 2 kuvataan olennaisia termejä. Luvussa 3 kuvataan sovelluksen käyttöliittymää näkymittäin. Luvussa 4 kuvataan sovelluksen kokonaisrakenne ja rajapinnat. Luvussa 5 kuvataan tavoitteiden toteutumista. Luvussa 6 annetaan ohjeita ylläpitäjälle ja jatkokehittäjälle.

2 Termit

Luvussa kuvataan projektissa käytettäviä aihealueen, tietojärjestelmän ja toteutus-tekniikoiden termejä.

2.1 Kohdealueen termejä

Projektin kohdealueen termejä ovat seuraavat:

CAVAPA	on Jyväskylän yliopiston liikuntakasvatuksen ja tietotekniikan laitoksen kehittämä ryhmätason fyysistä aktiivisuutta mittaava tietokoneavusteinen menetelmä. Se on lyhenne sanoista Computer Assisted Video Analysis of Physical Activity on group level.
Konenäkö	on tietojärjestelmä, joka analysoi tarkasteltavasta liikkuvasta kuvasta dataa. Konenäköllä voidaan korvata ihmiselle rasittavia rutiinitehtäviä esimerkiksi liukuhihnalla tai suorittaa ihmisen näkökyvylle mahdottomia tehtäviä käyttämällä aallonpituuksia, joita ihmisen silmä ei pysty havaitsemaan.
Käyttöliittymä	on ohjelmiston osa, jonka kautta käyttäjä käyttää ohjelmistoa.
Liikemittari	on mittari, joka mittaa liikemäärää tapahtuneen mittauksen aikana.
Liikemäärä	on fysikaalinen suure, joka on suoraan verrannollinen aktiivisuustasoon (katso luku 2.2).
Lämpökuva	on kaksiulotteinen kuva esittäen värein alueet, joilla videokuvassa aktiivisuutta esiintyi määritellyllä aikavälillä.
Ryhmä	on ihmisjoukko, jonka aktiivisuutta videokuvasta mitataan.
Sovellus	on tietojärjestelmän osa, joka sisältää analyysiohjelman ja käyttöliittymän sekä niiden väliset välityskerrokset.
Tietojärjestelmä	on ihmisistä, tietojenkäsittelylaitteista, tiedonsiirtolaitteista ja ohjelmistoista koostuva järjestelmä, jonka tarkoituksena on

tietojen käsittelyn avulla tehostaa tai helpottaa jotain toimintaa tai tehdä se ylipäättään mahdolliseksi.

Videolähde on joko videotiedosto tai videokamera.

2.2 Analyysin termejä

Analyysin termejä ovat seuraavat:

Analyysi	tarkoittaa aktiivisuusdatan muodostamista videokuvan perusteella.
Analyysiohjelma	muodostaa videotiedostoista aktiivisuusdatan, joka esitetään käyttöliittymän graafissa ja tallennetaan tiedostoon jatkokäsittelyä varten.
Aktiivisuusdata	tarkoittaa CAVAPA-algoritmin vasteita. Näitä ovat liikemäärä ja havaittujen kohteiden tiedot videokuvassa.
Aktiivisuustaso	on videokuvan tiettyyn aikaväliin liittyvä numeerinen arvo välillä $[0, 1]$, joka on kyseisellä aikavälillä havaitun liikemäärän ja sillä hetkellä tunnetun maksimiliikemäärän suhde.
CAVAPA-algoritmi	on menetelmä, joka laskee yhden tai useamman videolähteen videokuvan ja tarvittavien parametrien perusteella aktiivisuusdatan. CAVAPA-algoritmi suorittaa tietojärjestelmässä tarvittavan analyysin.
CAVAPA-ohjelma	(engl. <i>Cavapa program</i>) on toteutus CAVAPA-algoritmista.
CAVAPA-GUI	on Liikkuva-projektissa toteutettavan ohjelmiston työnimi.
Havainto	(engl. <i>sighting</i>) on CAVAPA-algoritmin havaitsema kohde.
Kalibrointi	sisältää ne toimenpiteet, joilla varmistetaan aktiivisuusdata mitattavan videokuvasta yhdenmukaisesti.

Kohde	on videokuvassa esiintyvä liikkuva hahmo.
Kohteen korostus	tarkoittaa suorakulmion piirtämistä videokuvaan kyseisen kohteen reunoille.
Liikemääräkäyrä	on kaksiulotteinen kuvaaja, joka kuvaa liikemäärän ajan funktiona.
Linssivääristymä	on optiikan ilmiö, jossa todellisen maailman suorat viivat näyttävät kameran kuvassa vinoutuneilta.
Mittaus	on prosessi, jossa CAVAPA-algoritmille syötetään videolähteistä saatavaa videokuvaa ja algoritmin vasteita otetaan talteen.
Perspektiivin korjaus	tarkoittaa CAVAPA-algoritmin alustamista sellaisilla parametreilla, että se saa käsityksen oikeasta kuvakulmasta. Käytännössä tämä tarkoittaa kaksiulotteisen ruudukon sovittamista videokuvan päälle.
Tynnyrivääristymä	on erikoistapaus linssivääristymästä. Se aiheuttaa suorien viivojen kaartumisen ulospäin.

2.3 Ohjelmistoja ja teknisiä termejä

Ohjelmistoja ja teknisiä termejä ovat seuraavat:

CSV	eli Comma Separated Values on tiedostomuoto, jolla tallennetaan taulukkomuotoista tietoa tekstitiedostoon.
Doxygen	on ohjelma luokkadokumentaation generointiin ohjelmakoodista.
Excel	on taulukkolaskentaohjelma, jonka toiminta perustuu taulukon soluihin.
FPS	eli Frames Per Second on lukuarvo, joka kertoo, montako kuvapäivitystä videolähteestä otetaan yhden sekunnin aikana.
JavaDoc	on ohjelma, jonka avulla Javan lähdekoodista voidaan generoida luokkadokumentaatio.

JPEG	eli JPG on häviöllinen kuvatiedostoformaatti.
Ohjelmakirjasto	on yleiskäyttöinen kokoelma ohjelmakoodia, joka on tarkoitettu hyödynnettäväksi ohjelmistoja kirjoitettaessa. Tällöin ohjelmoijien ei tarvitse ratkoa sellaisia ongelmia, jotka joku muu on jo ratkaissut.
Lähdekoodi	on tietokoneohjelman tekstimuotoinen ohjelmointikielinen listaus. Ennen varsinaista suorituskelpoista ohjelmaa lähdekoodi käännetään objektimuotoiseksi ohjelmaksi.
MJPEG	eli MJPG on videokuvan pakkaustekniikka, jossa jokainen videon ruutu pakataan JPG-kuvana.
MPEG-4	on MJPG:tä edistyneempi videokuvan pakkaustekniikka.
OpenGL	on grafiikkakirjasto ennen kaikkea 3D-grafiikan piirtoon.
OpenCV	on konenäkökirjasto, joka mahdollistaa muun muassa videokameroiden ja videotiedostojen käsittelyn.
PDF	eli Portable Document Format on PostScript-kieleen pohjautuva ohjelmistoriippumaton, siirrettävä tiedostomuoto.
PNG	on häviötön kuvatiedostoformaatti.
Qt	on käyttöliittymäkirjasto, jolla on mahdollista toteuttaa C++-kielellä ohjelmia, joista voidaan helposti tehdä toimivat versiot usealle eri käyttöjärjestelmälle.
SVG	eli Scalable Vector Graphic on vektorikuvaformaatti.
XML	eli Extensible Markup Language on tekstimuotoinen merkin-täkieli, jolla tiedon merkitys voidaan kuvata tiedon yhteyteen.

3 Käyttöliittymä

Luvussa esitellään sovelluksen käyttöliittymän eri näkymät sekä niihin liittyvät toiminnot ja niiden väliset suhteet.

3.1 Sovelluksen näkymät

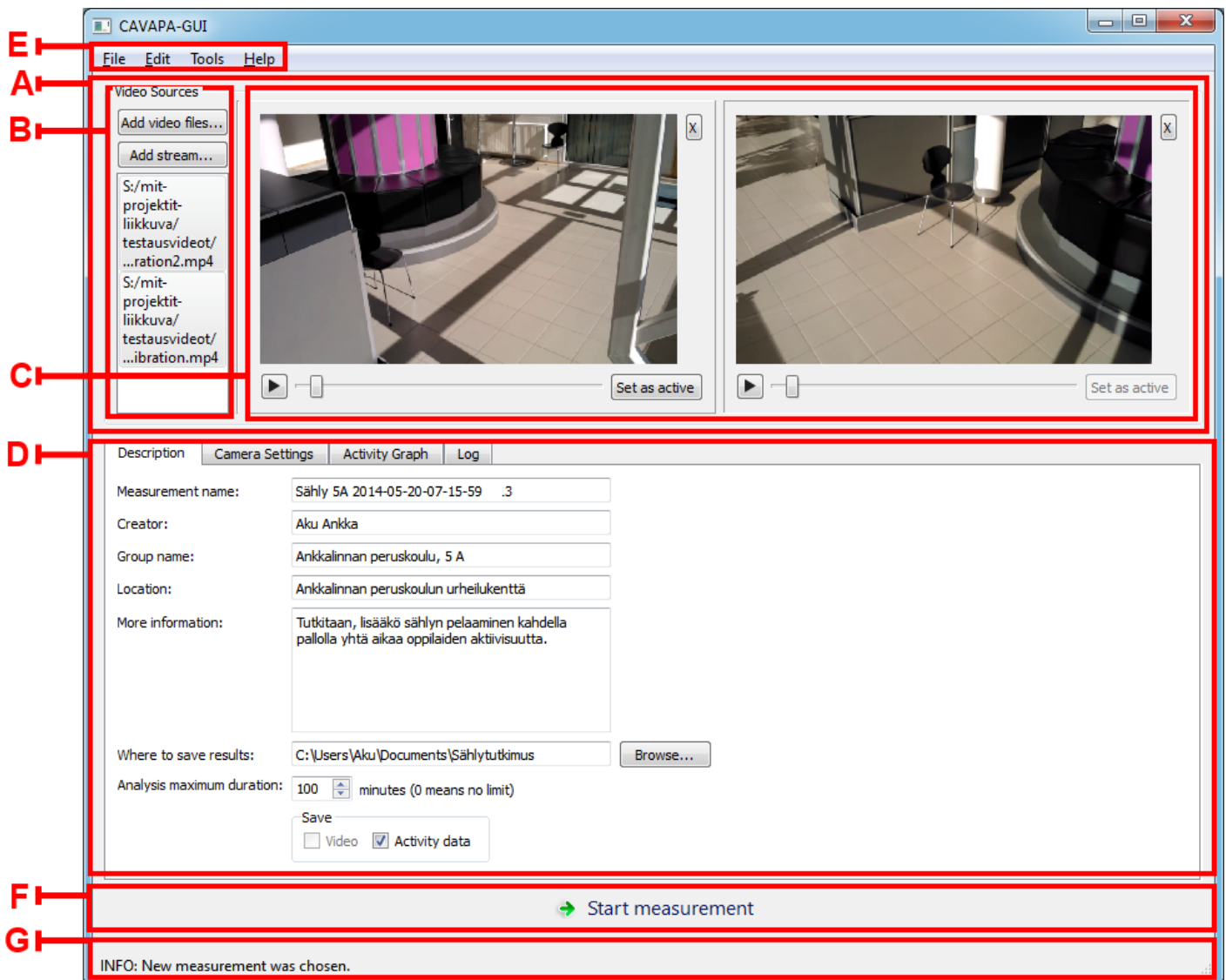
Kuvassa 3.1 havainnollistetaan sovelluksen eri näkymiä. Pääikkunanäkymä (luku 3.2) sisältää

- videonäkymät (osoitettu kirjaimella C, luku 3.3),
- videolähteiden hallinnan (B, luku 3.4),
- välilehtien näkymät (D, luvut 3.5 – 3.8) sisältäen
 - mittausta kuvaavat metatiedot (*Description*),
 - videolähteiden kalibroinnin (*Camera Settings*),
 - analyysin tulokset (*Activity Graph*) ja
 - lokin (*Log*),
- mittauksen hallinnan painikkeet (F, luku 3.9) sekä
- tilarivin (G, luku 3.17).

Muut näkymät ovat erillisiä ikkunoita. *Activity Graph* -välilehteen liittyvät näkymät *Add new marker*, *Export to CSV* ja *Graph Settings* esitellään luvuissa 3.10 – 3.12. Pääikkunan komentovalikoista (E, luku 3.13) ovat avattavissa ikkunat *Open stream*, *Settings* ja *About*, jotka esitellään luvuissa 3.14 – 3.16. Sovelluksessa on lisäksi dialogeja, joissa valitaan tiedostoja levyltä tai määritetään, mihin tiedostoon jotain tallennetaan. Kyseisiä tiedostojen valinta- ja tallennusikkunoita ei kuvata tarkemmin, sillä ne vastaavat käyttöjärjestelmän konventioita.

3.2 Pääikkuna

Kuvassa 3.1 esitetty pääikkuna on sovelluksen päänäkymä, joka aukeaa sovellus käynnistettäessä. Ikkunan yläosassa (A) näkyvät käytettävät videolähteet (B), jotka voivat esittää videokameroita tai -tiedostoja. Yläosan vasemmassa reunassa (C) hallitaan mittaukseen valittuja videolähteitä.



Kuva 3.1: Pääikkuna.

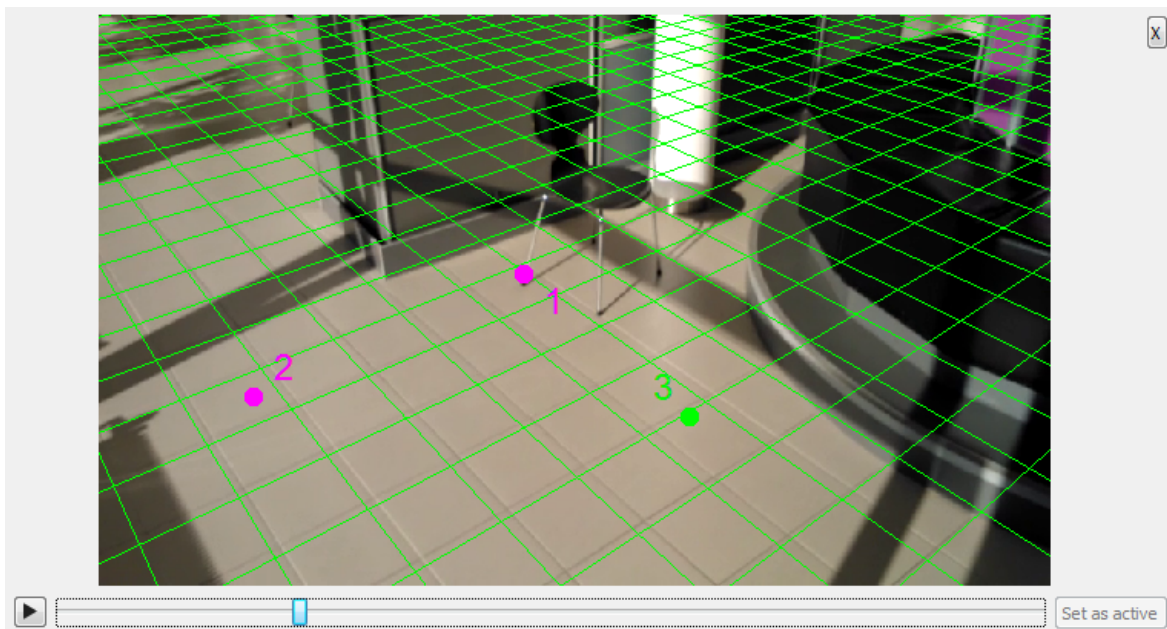
Päänäkymän alaosassa (D) ovat mittauksen asetuksiin ja tulosten esittämiseen liittyvät välilehdet. Näiden alapuolella on painike mittauksen aloittamiseen, kun mittauksen asetukset on ensin valittu (F). Kun mittaus on käynnissä, painikkeen tilalle tulevat mittauksen peruuttamisen ja lopettamisen mahdollistavat painikkeet sekä

videotiedostoja analysoitaessa myös taukopainike.

Pääikkunan yläreunassa ovat komentovalikot (E) ja alareunassa on tilarivi (G), jossa esitetään informaatio-, varoitus- ja virheviestejä ohjelman toimintaan liittyen.

3.3 Videonäkymä

Jokaiselle mittaukseen valitulle videolähteelle on pääikkunan yläosassa oma videonäkymänsä (kuva 3.2). Videonäkymässä esitetään kameroilta tulevat kuvat tai valittujen videotiedostojen kuvat. Kunkin videolähteen voi poistaa mittauksesta painamalla videonäkymän oikeassa yläkulmassa olevaa X-painiketta.



Kuva 3.2: Videonäkymä, jossa on näkyvillä kalibroitiruudukko ja -pisteet.

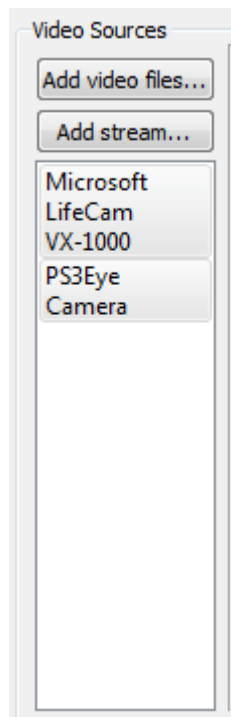
Videotiedostojen ollessa kyseessä voi videoita soittaa ja kelata ennen mittauksen aloittamista. Mittaus aloitetaan videotiedostojen tapauksessa kohdasta, johon video on kelattu aloituspainiketta painettaessa. Jos videot alkavat eri ajanhetkinä, voidaan ne synkronoida kelaamalla ne samaan kohtaan ennen aloituspainikkeen painamista. Mittauksen ollessa käynnissä videonäkymässä esitetään havaitut liikkuvat kohteet korostettuina värillisillä suorakulmioilla.

Kalibroitaessa analyysialgoritmia voidaan videonäkymässä tarkastella kalibroitiruudukkoa tai kalibroitipisteitä. Ruudukon tai pisteet voi tuoda näkyviin tai pois-

taa näkyvistä *Camera Settings* -välilehdellä (katso luku 3.6). *Set as active* -painikkeella valitaan, mitä videolähdettä *Camera Settings* -välilehden kalibrointisäätimillä kalibroidaan. Kalibrointipisteitä käytettäessä kyseistä painiketta ei tarvitse painaa. Kalibrointia kuvataan tarkemmin luvussa 3.6.

3.4 Videolähteiden hallinta

Pääikkunan (katso kuva 3.1) vasemmassa yläkulmassa on *Video Sources* -näkyvä (kuva 3.3). Kyseisessä näkymässä voi ennen mittauksen aloittamista lisätä videotiedostoja tai internetin kautta lähetettäviä videovirtoja mittaukseen.



Kuva 3.3: Videolähteiden hallinta.

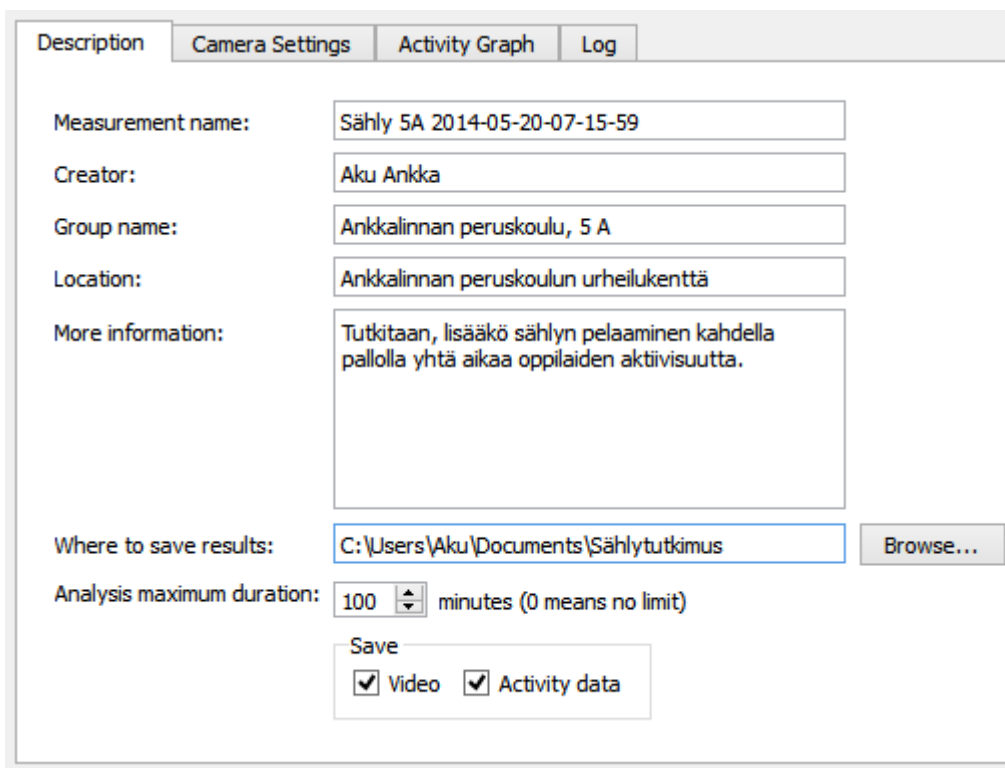
Tietokoneeseen liitetyt videokamerat tulevat automaattisesti videolähteiksi, kun ohjelma käynnistetään. Jos videokameran on poistanut videolähteiden joukosta tai tietokoneeseen on kiinnittänyt uuden videokameran ohjelman käynnistämisen jälkeen, voi kaikki tietokoneeseen liitetyt videokamerat tuoda jälleen näkyviin *Edit*-valikon (katso luku 3.13) komennolla *Refresh cameras*.

Video sources -näkymässä voi myös valita näkyvillä olevat videolähteet. Klikkaamalla videolähteen nimeä sen voi poistaa näkyvistä, ja piilotetun videolähteen nimeä

klikkaamalla sen voi tuoda näkyviin.

3.5 Mittauksen metatiedot

Kuvassa 3.4 esitetyllä *Description*-välilehdelle täytetään mittausta kuvaavia ja yksilöiviä metatietoja, kuten mittauksen nimi, suorittaja ja liikuntaryhmän nimi. Välilehdellä valitaan myös mittauksen tallennuskansio ja analyysin maksimipituus. Käyttäjä voi myös valita, haluaako hän tallentaa kiintolevyille videokuvaa ja aktiivisuusdataa.



The screenshot shows a software interface with four tabs: 'Description', 'Camera Settings', 'Activity Graph', and 'Log'. The 'Description' tab is active. It contains several input fields and a 'Save' button. The fields are: 'Measurement name' (Sähly 5A 2014-05-20-07-15-59), 'Creator' (Aku Anka), 'Group name' (Ankkalinnan peruskoulu, 5 A), 'Location' (Ankkalinnan peruskoulun urheilukenttä), 'More information' (a text area containing 'Tutkitaan, lisääkö sählyn pelaaminen kahdella pallolla yhtä aikaa oppilaiden aktiivisuutta.'), 'Where to save results' (C:\Users\Aku\Documents\Sählytutkimus) with a 'Browse...' button, and 'Analysis maximum duration' (100 minutes (0 means no limit)). Below these is a 'Save' button with two checked checkboxes: 'Video' and 'Activity data'.

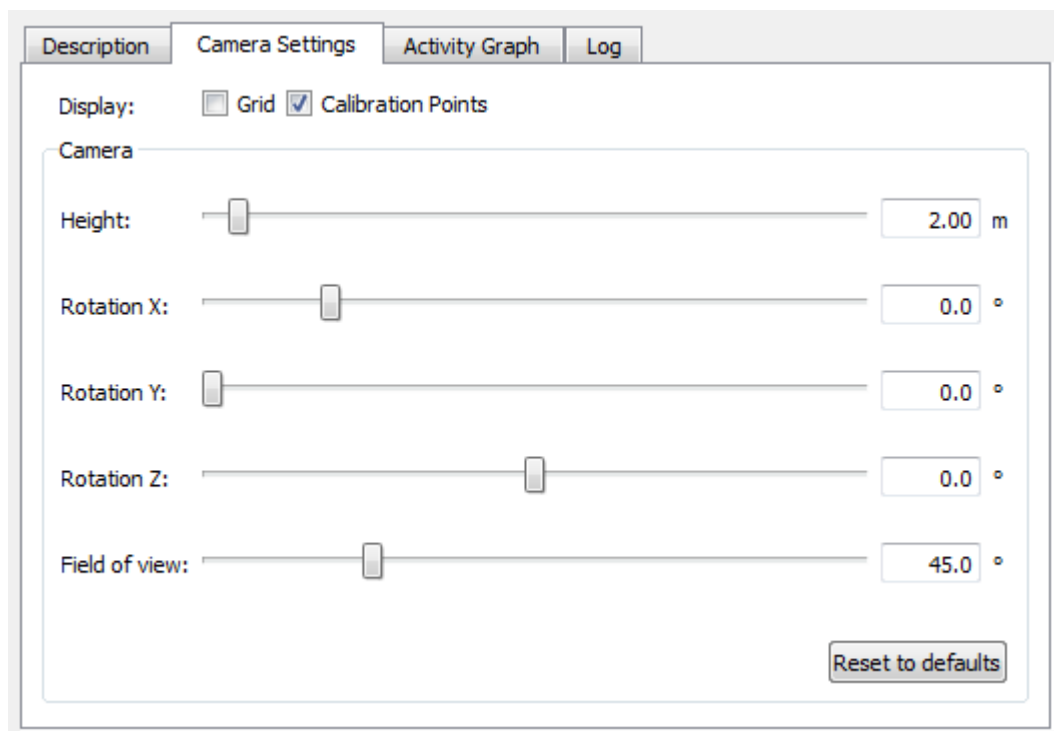
Kuva 3.4: Mittauksen metatiedot.

3.6 Kameran asetukset

Kuvassa 3.5 esitetyllä *Camera Settings* -välilehdellä käyttäjä voi valita, näytetäänkö kuvassa 3.2 esitetyn videonäkymän päällä kalibroitiruudukkoa tai kalibroitipisteitä. Välilehdellä käyttäjä voi kalibroida analyysialgoritmin arvioimalla kameran korkeutta ja kulmaa eri akseleiden suhteen sekä katselukulmaa. Arvot voi syöttää

joko liikusäätimillä tai syöttökenttiin. Arvon syöttökenttään syöttämisen jälkeen on painettava *enter*-näppäintä syötetyn arvon vahvistamiseksi.

Videonäkymässä (katso kuva 3.2) näkyvä ruudukko havainnollistaa, missä maan pinta kulkisi kulloisillakin kameran parametrien arvoilla. Videonäkymässä on napautettava ensin *Set as active* -painiketta, jolla valitaan kalibroitava videonäkymä. Videonäkymässä on myös mahdollista kelata videota, jotta löydettäisiin kohta, joka on mahdollisimman helppo kalibroida.



Kuva 3.5: Kameran asetukset.

Analyysin kannalta oleellimmat kalibroitiruudukon avulla säädettävät kameran parametrit ovat kameran korkeus ja sen kulma X-akselin suhteen, eli miten ylös tai alas kamera osoittaa. Jos maanpinnan tason arviointi ruudukon avulla tuottaa vaikeuksia, on kameran asetukset mahdollista määrittää myös arvioimalla sen sijaintia kuvaustilanteessa todellisessa maailmassa silmämääräisesti näiden kahden parametrin osalta. Kameran korkeudeksi voi asettaa suoraan kameran arvioidun korkeuden metreissä.

Kameran kulma X-akselin suhteen määräytyy seuraavasti:

- Kameran osoittaessa yläviistoon kulma on välillä $[-90^\circ, 0^\circ]$.
- Kameran osoittaessa vaakasuoraan kulma on 0° .

- Kameran osoittaessa alaviistoon kulma on välillä $[0^\circ, 90^\circ]$.
- Kameran osoittaessa suoraan maata kohden kulma on 90° .

Jos samaa aluetta kuvaavia videokameroita on useita, on suositeltavaa käyttää kalibrointiin **kalibrointipisteitä**. Näitä voi lisätä valikosta, joka aukeaa napsauttamalla videonäkymää (katso kuva 3.2) hiiren kakkospainikkeella. Kalibrointipisteillä merkitään jokaiseen videonäkymään jokin sama maailman piste, esimerkiksi liikuntasalin rajaviivan kulma. Pisteitä on lisättävä vähintään kolme kuhunkin videolähteeseen, jotta kalibrointi olisi mahdollista. Pisteitä on myös oltava sama määrä jokaisessa videolähteessä. Kalibrointipisteitä käytettäessä *Camera Settings* -välilehden liukusäätimien arvoja ei oteta huomioon.

Kalibrointipisteitä voi poistaa videonäkymän hiirivalikosta. Komento *Remove selected calibration point* poistaa valittuna olevan kalibrointipisteen (pisteitä voi valita hiirellä napsauttamalla). Komento *Remove all calibration points* poistaa kaikki kalibrointipisteet kyseisestä videonäkymästä.

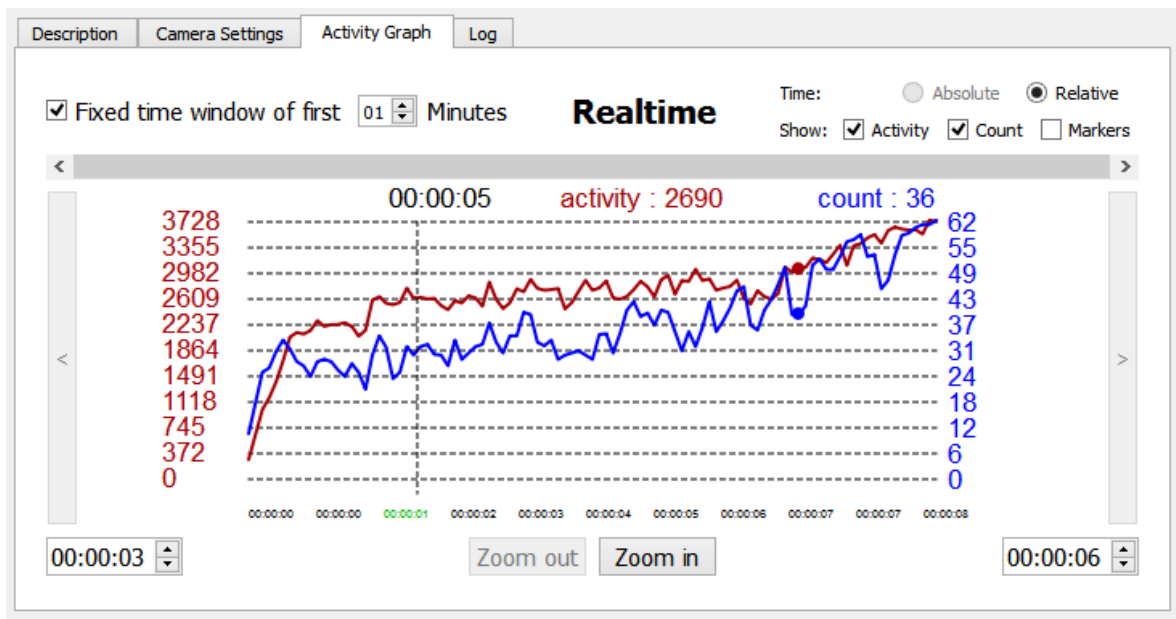
3.7 Aktiivisuusdatan kuvaajat

Kuvassa 3.6 esitetty *Activity Graph* -välilehti havainnollistaa mittausdataa kuvaajien avulla mittauksen aikana ja sen jälkeen. Välilehdellä esitetään korkeintaan kaksi kuvaajaa, joista toinen esittää havaitun aktiivisuuden määrää ja toinen havaittujen liikkuvien kohteiden määrää. Kuvaajien vasemmassa reunassa näkyy aktiivisuuskuvaajan asteikko ja oikeassa reunassa liikkuvien kohteiden määrän kuvaajan asteikko.

Viemällä hiiren osoittimen kuvaajien ylle, niiden yläpuolella esitetään kyseisellä ajanhetkellä mitattu aktiivisuuden määrä ja havaittujen liikkuvien kohteiden lukumäärä kuvaajia vastaavilla väreillä, sekä kyseisen ajanhetken aikaleima. Kyseinen ajanhetki korostetaan myös kuvaajissa näkyvillä pisteillä.

Activity Graph -välilehden toiminnot ovat seuraavat:

Hiirellä napsauttamalla voi valita kuvaajista tietyn kohdan. Jos kuvaajiin lisää hiirivalikosta merkin (katso luku 3.10), kun jokin kohta kuvaajasta on valittuna, tulee merkki valittuun kohtaan.



Kuva 3.6: Aktiivisuusdatan kuvaajat.

Hiirellä maalaamalla

voi valita kuvaajista aikavälin. Tämä vaikuttaa *Zoom in* -painikkeen toimintaan siten, että painiketta painamalla tarkasteltavaa aikaväliä rajataan valitun aikavälin mukaiseksi. Alueen valitseminen vaikuttaa myös hiirivalikon *Export to CSV File* -komentoon (katso luku 3.11) siten, että CSV-tiedostoon vietään ainoastaan valittu aikaväli.

Aikakentät

Activity Graph -välilehden vasemmassa ja oikeassa alakulmassa ovat toinen tapa valita osa kuvaajasta.

Hiirellä kaksoisnapsauttamalla kuvaajia voidaan nimetä valittu ajanhetki (katso luku 3.10).

Fixed time window of first (n) minutes -valintaruutu määrittää, mitä kuvaajista näytetään niiden ollessa zoomattuna mahdollisimman kauaksi. Voidaan näyttää joko lukukenttään syötetty määrä minutteja kuvaajien lopusta tai koko kuvaajat.

Realtime Analysis

-teksti kuvaajien yllä kertoo, että tarkasteltavaa aikaväliä ei ole rajattu. Jos aikaväliä on rajattu, tekstinä on *Data Analysis*.

Time	-valintapainikkeilla voidaan videokameralta tulevaa kuvaa analysoitaessa vaihtaa esitystapaa kuvaajien alapuolella näkyvien aikaleimojen sekä hiiren kuvaajan päälle vietäessä näkyvän aikaleiman osalta. <i>Absolute</i> -valinnalla aikaleimat kuvaavat todellisia kellonaikoja, kun taas <i>Relative</i> -valinnalla aikaleimat kuvaavat mittauksen aloituksesta kulunutta aikaa.
Show	-valintaruuduilla on mahdollista piilottaa ja tuoda näkyviin aktiivisuuskuvaaja (<i>Activity</i> -valintaruutu), havaittujen kohteiden lukumäärän kuvaaja (<i>Count</i> -valintaruutu) tai nimettyjen ajanhetkien nimet (<i>Markers</i> -valintaruutu).
Zoom in	-painiketta painamalla kuvaajista voi tarkastella pienempää aikaväliä kerralla. Jos hiirellä maalaamalla tai aikaikkunoilla on valittu kuvaajista jokin aikaväli, rajataan näkyvä osa kuvaajista tälle välille.
Zoom out	-painiketta painamalla kuvaajista voi tarkastella suurempaa aikaväliä kerralla.
Nuolipainikkeet	kuvaajien molemmilla puolilla mahdollistavat tarkasteltavan aikavälin siirron. Nuolipainikkeita voi myös pitää pohjassa, jolloin kuvaajia kelataan, kunnes painike vapautetaan.
Vierityspalkki	kuvaajien yläpuolella mahdollistaa tarkasteltavan aikavälin siirron.

Napsauttamalla kuvaajia hiiren kakkospainikkeella aukeaa hiirivalikko, jossa ovat seuraavat komennot:

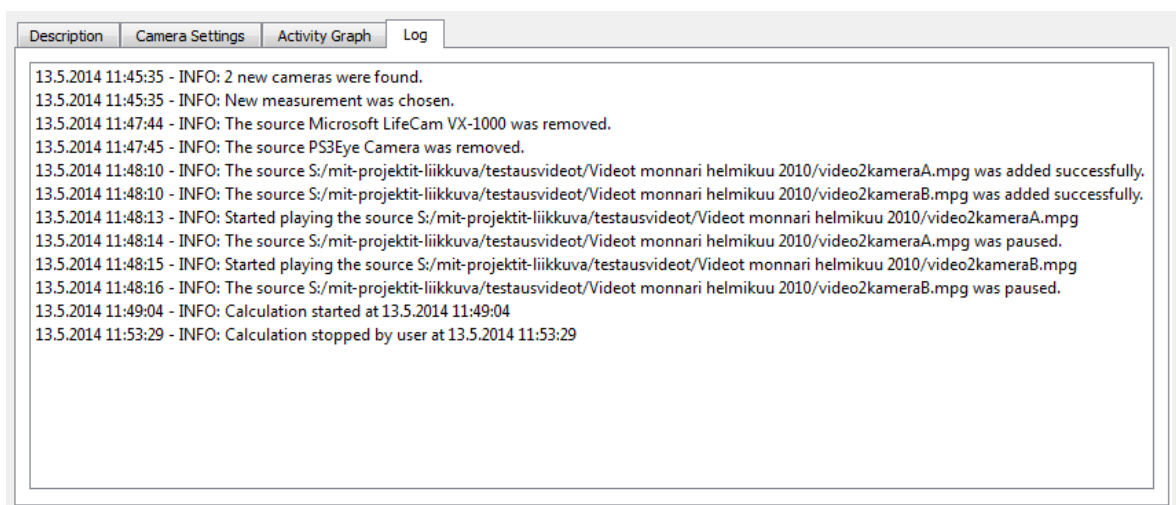
Create Marker	nimeää ajanhetken, jonka kohdalla hiirivalikko avattiin (katso luku 3.10).
Delete Marker	poistaa nimetyn ajanhetken, jonka kohdalla hiirivalikko avattiin.

- Save to SVG Image** tallentaa kuvaajat SVG-kuvaksi. Kuvatiedostoksi tallennetaan kuvaajanäkymä sellaisena kuin se tallennushetkellä esitetään, ts. rajausta, valinta ja merkityt ajanhetket vastaavat tallennushetken tilannetta.
- Export to CSV File** vie valitun alueen tulosdatan CSV-tiedostoon. Jos mitään aluetta ei ole valittu, eikä tarkasteluväliä ole rajattu, vietään kaikki analysoitu data (katso luku 3.11).
- Graph Settings** avaa *Graph Settings* -ikkunan, jossa voidaan säätää kuvaajaan liittyviä asetuksia (katso luku 3.12).

Lukujen 3.10 – 3.12 näkymät liittyvät aktiivisuuskuvaajiin.

3.8 Loki

Kuvassa 3.7 esitetyllä *Log*-välilehdellä näkyvät käyttäjälle tilarivillä (katso luku 3.17) lähetetyt viestit aikaleimoinen. Nämä voivat olla informaatio-, varoitus- tai virheviestejä.



Kuva 3.7: Loki.

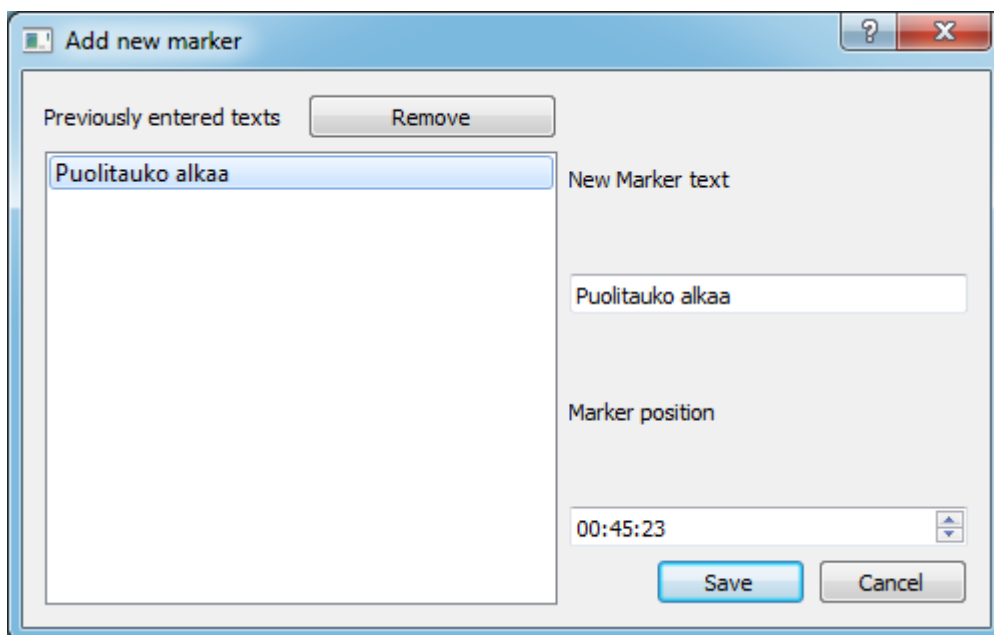
3.9 Mittauksen hallinta

Mittaus käynnistetään painamalla pääikkunan (katso kuva 3.1) *Start Measurement* -painiketta. Tämän jälkeen mittauksen metatietoja (katso luku 3.5) ei enää voi muokata. Mittauksen ollessa käynnissä käytössä ovat seuraavat painikkeet:

- Cancel** peruuttaa mittauksen, jolloin mitään dataa ei tallenneta.
- Stop** lopettaa mittauksen.
- Pause** laittaa mittauksen tauolle, jolloin sitä voidaan vielä jatkaa. Painike on käytettävissä ainoastaan videotiedostoja analysoitaessa.

3.10 Uuden merkin lisäys

Kuvassa 3.8 esitetty *Add new marker* -ikkuna avataan *Acitivity Graph* -välilehdellä (katso kuva 3.6) olevien kuvaajien hiirivalikosta valinnalla *Create Marker* tai kaksoisnapsauttamalla kuvaajia.

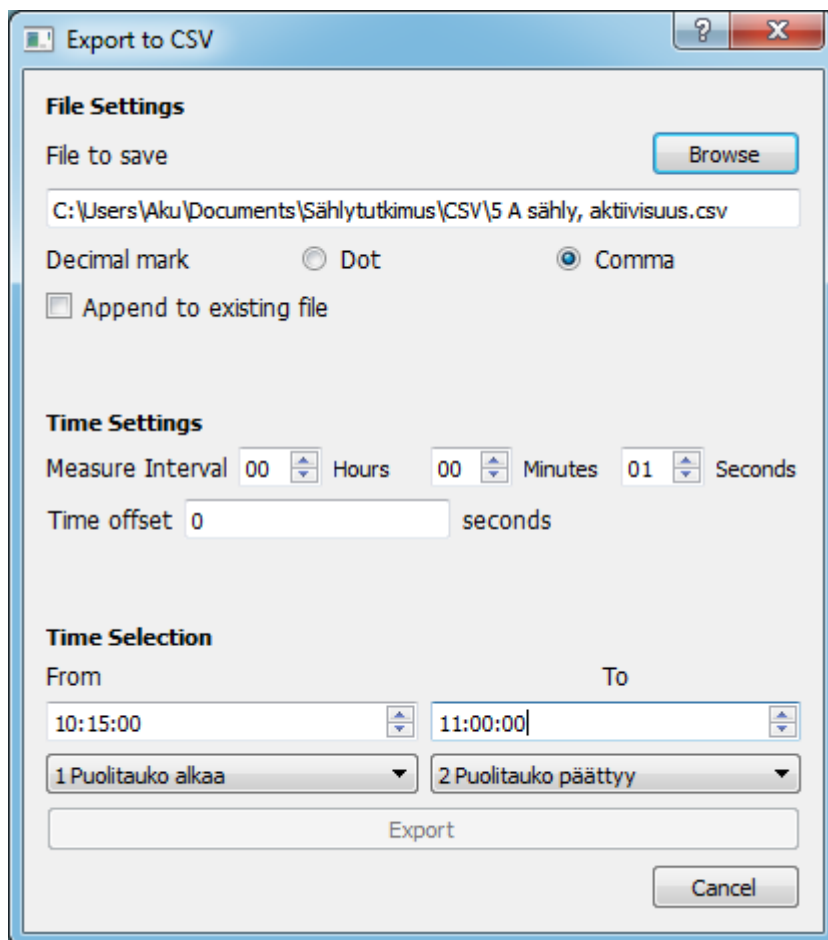


Kuva 3.8: Uuden merkin lisäys.

Ikkunassa nimetään kuvaajiin kyseinen ajanhetki. Ikkunassa näkyvät aiemmin käytetyt ajanhetkien nimet, ja merkinnän aikaleimaa voi muuttaa. Nimetyt ajanhetket tulevat näkyviin myös viettäessä dataa CSV-tiedostoon.

3.11 CSV-tiedostoon tallennus

Kuvassa 3.9 esitetty *Export to CSV* -ikkuna avataan *Activity Graph* -välilehdellä (katso kuva 3.6) olevien kuvaajien hiirivalikosta valinnalla *Export to CSV File* tai pääikkunasta *File*-valikon (katso luku 3.13) komennolla *Export activity graph data....* Ikkunan avulla tallennetaan valittuna olevan alueen tai koko mittauksen data Excel-taulukkolaskentaohjelmalla luettavissa olevaan CSV-tiedostoon.



Kuva 3.9: Tallennus CSV-tiedostoon.

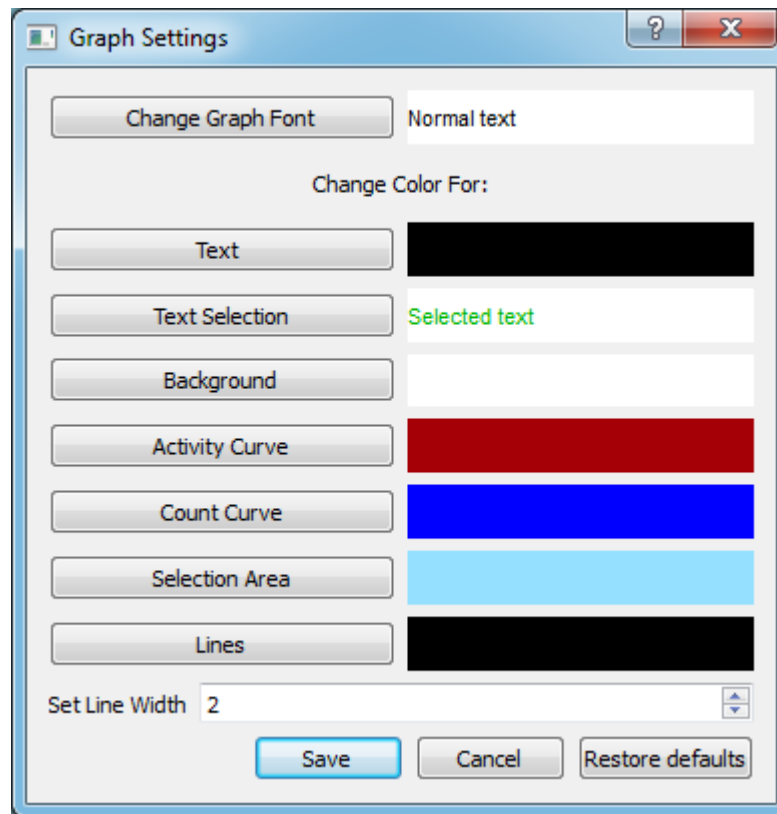
Ikkunassa on mahdollista määrittää seuraavat asetukset CSV-vientiä varten:

- | | |
|--------------------------------|--|
| File to save | määrittää polun, johon CSV-tiedosto tallennetaan. |
| Decimal mark | määrittää, onko desimaalimerkkinä piste vai pilkku. |
| Append to existing file | määrittää, lisätäänkö tiedot vanhan tiedoston perään vai kirjoitetaanko vanhan tiedoston päälle. <i>File to save</i> |

	-kentässä tulee olla valittuna jokin olemassa oleva tiedosto.
Measure Interval	määrittää aikavälin, jonka mittaisista mittausjaksoista viedyt tiedot ovat keskiarvoja.
Time offset	määrittää mittausdatan aikaleimoihin lisättävän määrän sekunteja. Arvo voi olla myös negatiivinen.
Time Selection	määrittää aikavälin, jonka data halutaan viedä CSV-tiedostoon. Aikaväli voidaan määrittää antamalla aloitus- ja lopetusaika tai valitsemalla videoon nimetyt ajanhetket aloitus- ja lopetusajaksi.

3.12 Kuvaajien asetukset

Kuvassa 3.10 esitetty *Graph Settings* -ikkuna avataan *Activity Graph* -välilehdellä (katso kuva 3.6) olevien kuvaajien hiirivalikosta valinnalla *Graph Settings*. Ikkunassa voi muokata kuvaajien ulkoasua vaihtamalla kuvaajissa käytettyä kirjasinta, värejä ja kuvaajien viivojen paksuutta.



Kuva 3.10: Kuvaajien asetukset.

3.13 Pääikkunan komentovalikot

Pääikkunan *File*-valikko sisältää seuraavat komennot:

New measurement	luo uuden mittauksen. Käytössä olevat videolähteet säilyvät.
Open existing measurement...	avaa aiemmin suoritetun mittauksen. Mittaus avataan valitsemalla siitä tallennettu XML-tiedosto.
Redo current measurement	suorittaa nykyisen mittauksen uudestaan.
Open video files...	avaa videotiedostoja. Tiedostot lisätään mittaukseen videolähteiksi.
Open stream...	avaa videovirran URL-osoitteen perusteella. Videovirta lisätään mittaukseen videolähteeksi (katso luku 3.14).

Export activity graph data...	tallentaa aktiivisuuskuvaajien datan CSV-tiedostoon (katso luku 3.11).
Export activity graph image...	tallentaa aktiivisuuskuvaajat SVG-tiedostoksi. Tiedostoon tallennetaan kuvaajanäkymä sellaisena kuin se tallennushetkellä näyttää, ts. rajaus, valinta ja merkityt ajanhetket vastaavat tallennushetken tilannetta.
Exit	sulkee ohjelman.

Pääikkunan *Edit*-valikko sisältää seuraavan komennon:

Refresh cameras	tuo näkyviin kaikki tietokoneeseen liitetyt kamerat.
------------------------	--

Pääikkunan *Tools*-valikko sisältää seuraavan komennon:

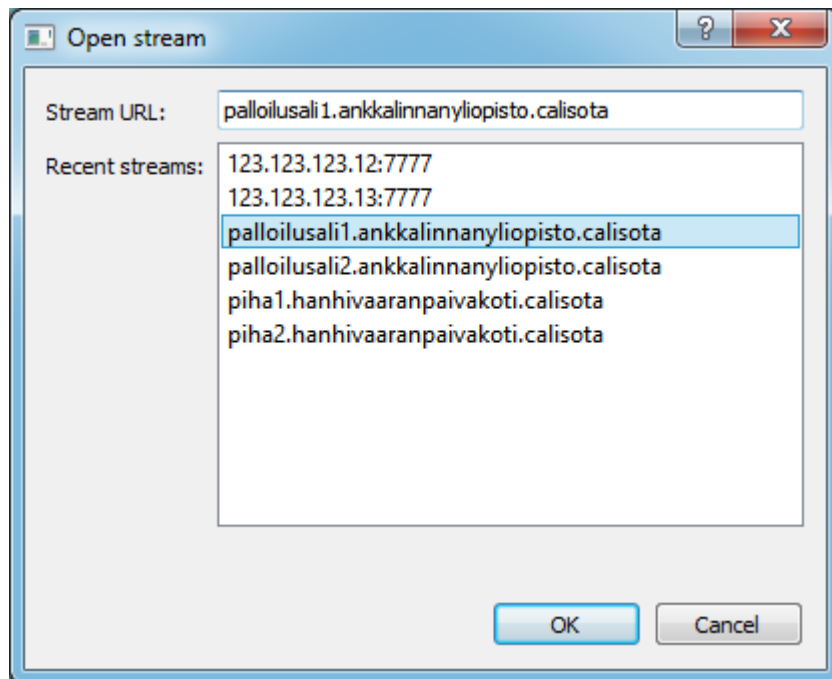
Settings... avaa *Settings*-ikkunan (katso luku 3.15).

Pääikkunan *Help*-valikko sisältää seuraavan komennon:

About... avaa *About*-ikkunan (katso luku 3.16).

3.14 Videovirran avaus

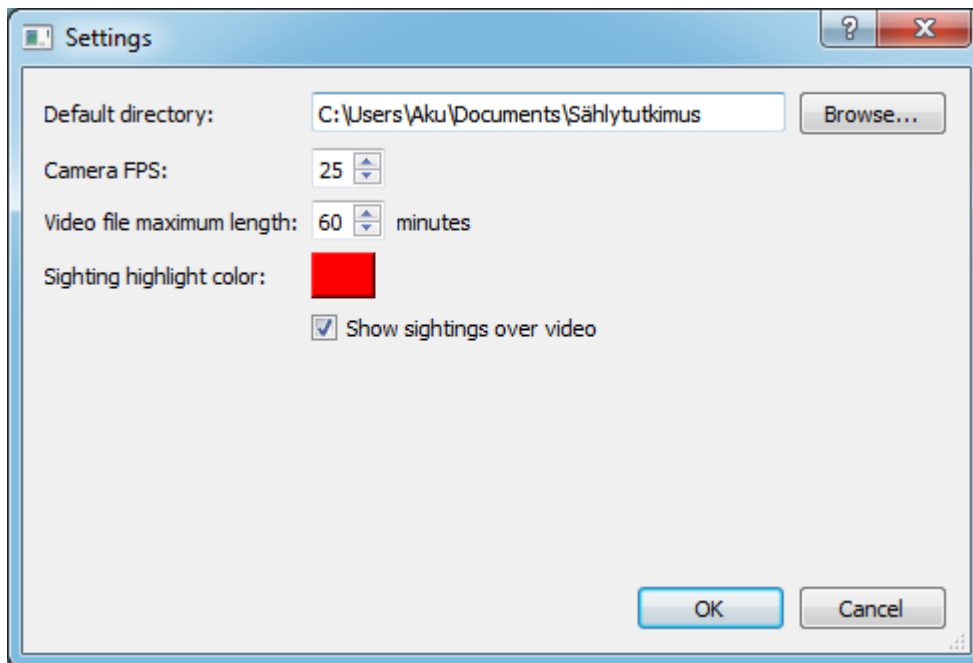
Kuvassa 3.11 esitetyn *Open stream* -ikkunan voi avata joko *File*-valikon (katso luku 3.13) komennolla *Open stream...* tai pääikkunan (katso kuva 3.1) painikkeella *Add stream....* Ikkunassa voidaan avata videovirta URL-osoitteen perusteella tai valita avattava videovirta viimeisimpien käytettyjen videovirtojen joukosta.



Kuva 3.11: Videovirran avausikkuna.

3.15 Ohjelman asetukset

Kuvassa 3.12 esitetty *Settings*-ikkuna aukeaa pääikkunan (katso kuva 3.1) *Tools*-valikon komennolla *Settings....* Ikkunassa voidaan vaihtaa oletustallennuskansiota (jota sovellus tarjoaa tallennussijainniksi uutta mittausta suoritettaessa), käytettävää FPS-arvoa ja videotiedostojen maksimipituutta. Lisäksi voidaan vaihtaa liikkuvien kohteiden korostusväriä näytettävässä videokuvassa sekä sitä, näytetäänkö liikkuvien kohteiden korostusta.



Kuva 3.12: Ohjelman asetukset.

3.16 Tietoa ohjelmasta

Help-valikon (katso luku 3.13) komennolla *About...* avattavassa ikkunassa esitetään sovelluksen nimi, tekijöiden nimet ja sovelluksen lisenssi.

3.17 Tilarivi

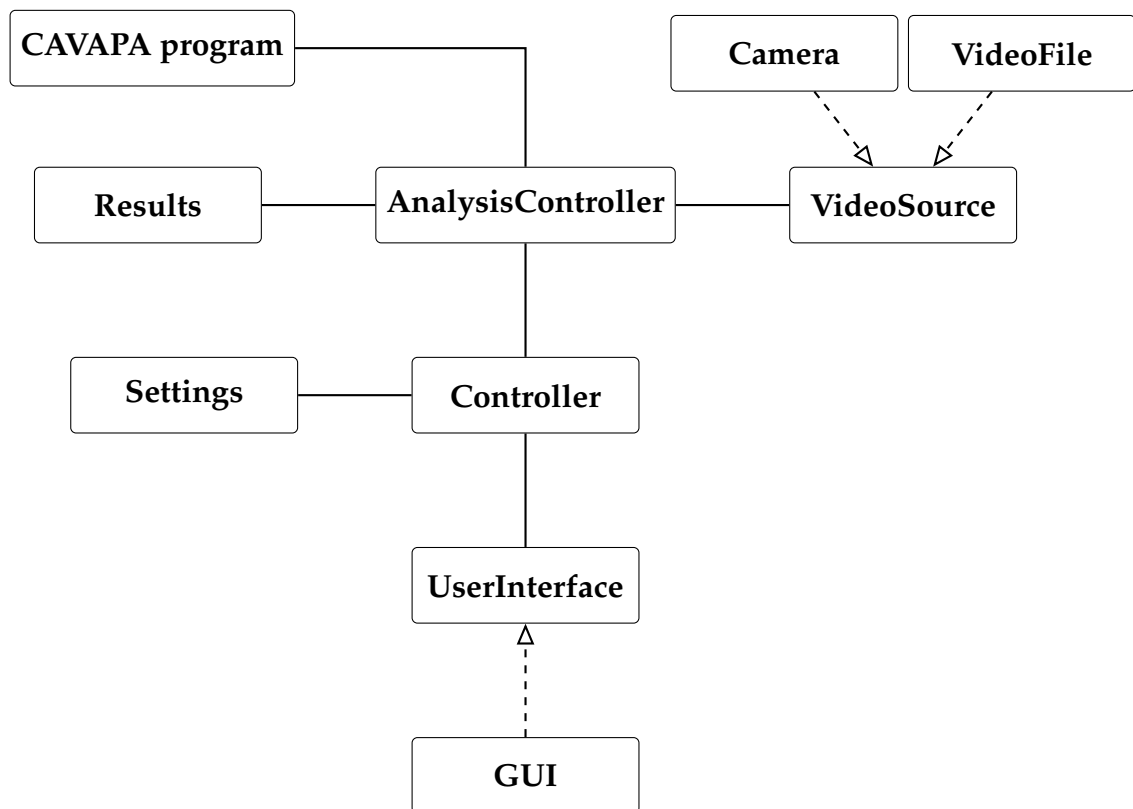
Pääikkunan (katso kuva 3.1) alareunassa on tilarivi, jossa esitetään informaatio-, varoitus- ja virheviestejä ohjelman toimintaan liittyen.

4 Sovelluksen kokonaisrakenne ja rajapinnat

Luvussa kuvataan tietojärjestelmän kokonaisrakennetta, käytettyjä ulkopuolisia komponentteja, toteutusratkaisuihin tehtyjä olennaisimpia muutoksia, sekä syöte- ja vastetiedostoja. Liikkuva-projektissa kehitetyn käyttöliittymän käyttäjille tarjoamat tiedot ja toiminnot on kuvattu tarkemmin vaatimusmäärittelyssä [10]. Sovelluksen luokat ja niiden rajapinnat kuvataan yksityiskohtaisesti luokkadokumentaatiossa [5].

4.1 Sovelluksen kokonaisrakenne

Sovelluksen kokonaisrakenne on kuvattu kuvassa 4.1.



Kuva 4.1: Sovelluksen kokonaisrakenne.

Sovellus jakautuu seuraaviin kokonaisuuksiin:

CAVAPA program suorittaa varsinaisen videokuvan analysoinnin. CAVAPA-ohjelma ei ole Liikkuva-projektin toteuttama.

Camera	sisältää kamerakuvan käsittelyn.
VideoFile	sisältää videotiedoston käsittelyn.
VideoSource	sisältää yhteisen rajapinnan kamerakuvalle ja videotiedostoille.
Results	sisältää analyysitulosten hallinnan. Se kattaa sovelluksessa näytettävän liikemäärän ja aktiivisuustason kuvaajassa, sekä näiden tietojen siirtämisen sovelluksesta ulos jatkoanalyysiä varten.
Settings	sisältää asetusten hallinnan. Se kattaa haluttujen asetusten hallinnan kameroille ja tarkasteltavalle ajanjaksolle.
AnalysisController	välittää videokuvaa videolähteistä CAVAPA-ohjelmalle ja tallentaa CAVAPA-ohjelman palauttamaa analyysidataa.
Controller	vastaa ohjainkontrollista välittäen tietoa käyttöliittymän ja analyysikontrollin välillä ja huolehtien asetusten hallinnasta.
UserInterface	toimii rajapintana ohjainkontrollin ja käyttöliittymän välillä mahdollistaen myös toisenlaisten käyttöliittymien toteuttamisen ilman muutoksia sovelluksen muihin osiin.
GUI	sisältää käyttöliittymän ulkoasun. Se sisältää sovelluksen ulkonäköön liittyvät osuudet kamerakuvan sijoittelusta kuvaajien sijoitteluun.

4.2 Ulkopuoliset komponentit

Tietojärjestelmä hyödyntää ulkopuolisia komponentteja seuraavista kirjastoista ja ohjelmista:

- CAVAPA** (kuvassa 4.1) on Jarkko Vilhusen toteuttama analyysiohjelma, joka analysoi videokuvassa esiintyvää liikettä.
- OpenCV** on konenäkökirjasto, jota sovelluksessa hyödynnetään videokameroiden ja -tiedostojen käsittelyyn.

OpenGL on grafiikkakirjasto, jota käytetään videokuvan ja sen päällä esitettävän kalibroitiruudukon piirtämiseen.

Qt on käyttöliittymäkirjasto, jolla graafinen käyttöliittymä on toteutettu.

4.3 Olennaisimmat muutokset toteutusratkaisuisa

Sovelluksen kokonaisrakenteeseen ei tehty muutoksia projektin aikana.

Projektin alussa aiottiin ensin toteuttaa kameroiden ja videotiedostojen käsittely ilman OpenCV:tä Qt:n omilla kamerakirjastoilla. Nämä eivät kuitenkaan toimineet kovin hyvin, joten sovelluksen rakenteen selkeydyttyä päätettiin toteuttaa kameroiden käsittely OpenCV-kirjastolla.

Aktiivisuuskuvaajat toteutettiin aluksi OpenGL:ää käyttäen. Myöhemmin siirryttiin käyttämään niissä Qt:n vektorigrafiikkakirjastoja, koska ne tarjosivat paremman tuen tekstin piirtämiselle.

Videonäkymän piirroksessa käytetty OpenGL-koodi vaati muutoksia, ennen kuin näkymän päälle oli mahdollista piirtää Qt:n vektorigrafiikkakirjastoilla kalibroitipisteet ja niiden numerointi. OpenGL-piirto oli toteutettava ilman verteksipuskuriolioita.

4.4 Asetustiedosto

Ohjelman asetukset sekä pääikkunan koko ja sijainti tallennetaan ohjelman asennuskansiossa olevaan tiedostoon nimeltä `cavapa-gui.ini`. Tiedoston sisältö voi olla esimerkiksi seuraava:

```
[General]
camera_fps=25
default_directory=C:\Users\Aku\Documents\Sählytutkimus
graph_activity_curve_color="165;0;5"
graph_background_color="255;255;255"
graph_count_curve_color="0;0;255"
graph_half_line_color="0;0;0"
graph_line_width=2
```

```
graph_selection_color="149;224;255"  
graph_text_color="0;0;0"  
graph_text_font=Arial  
graph_text_selection_color="0;179;0"  
highlight_color="255;0;0"  
is_highlight_enabled=true  
is_window_maximized=true  
max_video_file_length=60  
win_location="0;23"  
win_size="1920;1017"
```

```
[recentsources]
```

```
size=6
```

```
1\value=123.123.123.12:7777
```

```
2\value=123.123.123.13:7777
```

```
3\value=palloilusali1.ankkalinnanyliopisto.calisota
```

```
4\value=palloilusali2.ankkalinnanyliopisto.calisota
```

```
5\value=piha1.hanhivaaranpaivakoti.calisota
```

```
6\value=piha2.hanhivaaranpaivakoti.calisota
```

```
[markerhistory]
```

```
size=2
```

```
1\value=Puolitauko alkaa
```

```
2\value=Puolitauko päättyy
```

```
[Source 1 name]
```

```
barrel_distortion=0.000000
```

```
direction="0.000000;0.000000;1.000000"
```

```
fov=45.000000
```

```
position="0.000000;2.000000;0.000000"
```

```
resolution="640;480"
```

General-listan kentillä on seuraavat merkitykset:

camera_fps	määrittää, kuinka monta kertaa sekunnissa kame-roilta pyydetään kuvaa.
default_directory	on oletuskansio, jota ehdotetaan uuden mittauk-sen tallennussijainniksi.
graph_background_color	määrittää kuvaajien taustaväriin.
graph_count_curve_color	määrittää havaittujen kohteiden lukumäärän esittävän kuvaajan väriin.
graph_half_line_color	määrittää kuvaajan taustalla olevien X-akselin suuntaisten katkoviivojen väriin.
graph_line_width	määrittää kuvaajien viivojen paksuuden.
graph_selection_color	määrittää kuvaajista valitun alueen taustaväriin.
graph_text_color	määrittää kuvaajassa esiintyvien tekstien väriin.
graph_text_font	määrittää kuvaajassa esiintyvien tekstien kirjasi-men.
graph_text_selection_color	määrittää valittuna olevan ajanhetken ai-kaleiman väriin.
highlight_color	määrittää videonäkymässä korostettujen havaittu-jen kohteiden korostusväriin.
is_highlight_enabled	määrittää, korostetaanko videonäkymässä havait-tuja kohteita.
is_window_maximized	määrittää, onko pääikkuna koko ruudun kokoi-nen avattaessa ohjelma.
max_video_file_length	määrittää videotiedostojen maksimipituuden. Pi-demmat videot pilkotaan useampaan osaan.
win_location	määrittää pääikkunan sijainnin avattaessa ohjel-ma.
win_size	määrittää pääikkunan koon avattaessa ohjelma.

Tiedoston kolmen viimeisen listan tarkoitukset ovat seuraavat:

- recentsources** sisältää viimeksi käytettyjen videovirtojen osoitteet.
- markerhistory** sisältää viimeksi käytetyt kuvaajiin merkittyjen ajanhetkien nimet.
- Source 1 name** sisältää name-nimisen videolähteen viimeksi käytetyt kalibrointitiedot.

4.5 Mittauksen metadatatiedosto

Mittausta kuvailevat ja yksilöivät tiedot sekä mittaukseen liittyvät videolähteet kalibrointitietoineen tallennetaan mittauksen tallennuskansioon XML-tiedostoon, jonka nimi on sama kuin mittauksen nimi. Tiedosto voi näyttää esimerkiksi tältä:

```
<?xml version="1.0" encoding="UTF-8"?>
<measurement>
  <creator>Aku Ankka</creator>
  <enddate>21.5.2014 13:15:38</enddate>
  <group>Ankkalinnan peruskoulu, 5 A</group>
  <info>Tutkitaan, lisääkö sählyn pelaaminen kahdella
  pallolla yhtä aikaa oppilaiden aktiivisuutta.</info>
  <location>Ankkalinnan peruskoulun urheilukenttä
</location>
  <max_duration>100</max_duration>
  <name>Sähly 5A 2014-05-21-13-14-59</name>
  <rawdata>activity.cpa</rawdata>
  <save_activity>true</save_activity>
  <save_video>true</save_video>
  <startdate>21.5.2014 14:45:37</startdate>
  <videosources>
    <source type="video">
      <id>11</id>
      <name>sähly 5 A 21.5.2014.wmv</name>
      <startoffset>0</startoffset>
      <barrel_distortion>0.000000
```

```
</barrel_distortion>
<direction>0.000000;0.000000;1.000000
</direction>
<fov>45.000000</fov>
<position>0.000000;2.000000;0.000000
</position>
<resolution>854;480</resolution>
<video>C:/Users/Aku/Documents/Sählytutkimus/
videot/sähly 5 A 21.5.2014.wmv</video>
<calibrationpoints>
  <calibrationpoint>
    <id>0</id>
    <position>169;186</position>
  </calibrationpoint>
  <calibrationpoint>
    <id>1</id>
    <position>549;115</position>
  </calibrationpoint>
  <calibrationpoint>
    <id>2</id>
    <position>449;306</position>
  </calibrationpoint>
</calibrationpoints>
</source>
</videosources>
<markers>
  <marker>
    <active>true</active>
    <frametime>0</frametime>
    <text>Puoliaika alkaa</text>
  </marker>
  <marker>
    <active>true</active>
    <frametime>1028</frametime>
    <text>Puoliaika alkaa</text>
  </marker>
  <marker>
```

```
<active>true</active>
<frametime>7444</frametime>
<text>Puoliaika päättyy</text>
</marker>
</markers>
</measurement>
```

measurement-elementti on XML-tiedoston juurielementti. measurement-elementin lapsielementtien merkitykset ovat seuraavat:

creator	sisältää mittauksen tekijän nimen.
enddate	sisältää mittauksen lopetuksen päivämäärän ja kellonajan.
group	sisältää mitattavan liikuntaryhmän nimen.
info	sisältää mittauksen pidemmän kuvauksen.
location	sisältää mittauspaikan nimen.
max_duration	sisältää mittauksen maksimikeston.
name	sisältää mittauksen nimen.
rawdata	sisältää mittausdatan tallennustiedoston nimen.
save_activity	kertoo, onko aktiivisuusdataa tallennettu kiintolevyille.
save_video	kertoo, onko videokuvaa tallennettu kiintolevyille videokameraa videolähteenä käytettäessä.
startdate	sisältää mittauksen aloituksen päivämäärän ja kellonajan.
videosources	on videolähteiden vanhempielementti.
markers	on merkittyjen ajanhetkien vanhempielementti.

videosources-elementin lapsielementtien merkitykset ovat seuraavat:

source	on yhden videolähteen tiedot sisältävä elementti.
type	on attribuutti, joka sisältää videolähteen tyyppin. Vaihtoehdot ovat camera, stream ja video.

id	sisältää videolähteen tunnistenumeron.
name	sisältää videolähteen nimen.
startoffset	kertoo, mistä kohdasta videota mittaus on aloitettu.
barrel_distortion	sisältää tynnyrivääristymän voimakkuuden.
direction	sisältää suunnan, johon kamera on kalibroitaessa arvioitu osoittavan.
fov	sisältää katselukentän laajuuden asteissa.
position	sisältää kalibroitaessa arvioidun kameran sijainnin.
resolution	sisältää videolähteen resoluution.
video	sisältää videolähteeseen liittyvän videotiedoston polun.
calibrationpoints	on kalibroitipisteiden vanhempielementti.

calibrationpoints-elementin lapsielementtien merkitykset ovat seuraavat:

calibrationpoint	on yhden kalibroitipisteen tiedot sisältävä elementti.
id	sisältää kalibroitipisteen tunnistenumeron.
position	sisältää kalibroitipisteen sijainnin videokuvassa.

markers-elementin lapsielementtien merkitykset ovat seuraavat:

marker	on yhden merkityn ajanhetken tiedot sisältävä elementti.
active	kertoo, onko merkitty ajanhetki vielä käytössä. Se mahdollistaa merkintöjen poistamisen peruuttamisen, jota ei ole kuitenkaan vielä toteutettu sovellukseen.
frametime	sisältää merkityn ajanhetken aikaleiman videon alusta millisekunteina.
text	sisältää merkitylle ajanhetkelle annetun nimen.

4.6 Analyysidatatieosto

Analyysidata tallennetaan binääritiedostoon `activity.cpa`, jonka muoto on seuraava (tämä ei ole esimerkkidataa, vaan hahmotelma tiedoston rakenteesta):

```
frame_count (i) : uint32_t
  [FRAME 1]
    frame_stats   : FrameStats
    cameras (n)   : uint32_t
    sightings (m) : uint32_t

    [CAMERA 1]
      sighting_count (j) : uint32_t
    [CAMERA 2]
      sighting_count      : uint32_t
    [CAMERA 3]
    ...
    [CAMERA n]

    [CAMERA 1 SIGHTINGS]
      [SIGHTING 1]
        sighting          : SightingFixed
      [SIGHTING 2]
        sighting          : SightingFixed
      [SIGHTING 3]
      ...
      [SIGHTING j]
    [CAMERA 2 SIGHTINGS]
    ...
    [SIGHTINGS m]
  [FRAME 2]
  ...
  [FRAME i]
```

Rakennehahmotelman muuttujien merkitykset ovat seuraavat:

- frame_count** sisältää tallennettujen videoruutujen määrän.
- frame_stats** sisältää yhden ruudun tiedot, jotka koostuvat videon alusta kuluneesta ajasta, aktiivisuuden määrästä ja havaittujen liikkuvien kohteiden määrästä.
- cameras** sisältää videoruutuun liittyvien kameroiden määrän.
- sightings** sisältää videoruutuun liittyvien havaittujen liikkuvien kohteiden määrän.
- sighting_count** sisältää yhdestä kamerasta havaittujen liikkuvien kohteiden määrän.
- sighting** sisältää yhden havaitun liikkuvan kohteen tiedot, jotka koostuvat tunnistenumeroista, maailman koordinaateista, havainnon varmuusasteesta ja havainnon videokuvassa rajaavan suorakulmion tiedoista.

5 Tavoitteiden toteutuminen

Luvussa kuvataan vaatimusten toteutumista sekä heikkoja ja puutteellisia toteutusratkaisuja. Sovelluksen vaatimukset on kuvattu vaatimusmäärittelyssä [10]. Tunnetut virheet on lueteltu järjestelmätestausraporteissa [8] ja [9].

Vaatimusmäärittelyn ensimmäisen prioriteetin 33:sta vaatimuksesta toteutui 32. Rajallisen aikataulun vuoksi sovellukseen jäi myös heikkoja ja puutteellisia toteutusratkaisuja. Ohjelmointikäytänteet ja niiden toteutuminen on kuvattu projektiraportin [6] luvussa 5.5.

5.1 Vaatimusten toteutuminen

Vaatimusten toteutuminen vaatimuskohtaisesti on kuvattu vaatimusmäärittelyssä [10]. Pakollisista 33 vaatimuksesta toteutui kokonaan 32 vaatimusta ja yksi toteutettiin osittain. Osittain toteutettu pakollinen vaatimus 6.1.5 liittyy toimintaan Linux-käyttäjärjestelmässä, sillä sovelluksen virheet hankaloittavat käyttöä Linuxissa.

Tärkeistä 26 vaatimuksesta toteutettiin 23. Osittain toteutettiin seuraavat vaatimukset:

5.3.4 *Käyttäjä voi valita videolähteen viimeksi käytettyjen videolähteiden listalta.*

5.5.7 *Sovellus tallentaa videolähteen kalibrointitiedot automaattisesti asetustiedostoon.*

Vaatimusmäärittelyssä [10] on kuvattu, miltä osin vaatimukset jäivät toteuttamatta.

Kokonaan toteuttamatta jäivät seuraavat vaatimukset:

5.8.11, *Käyttäjä voi avata näkymän, jossa havaituista kohteista näytetään 2D-lämpökuva käyttäjän määrittämältä aikaväliltä.*

5.8.12, *Käyttäjä voi määrittää, mistä kamerasta katsottuna lämpökuva näytetään.*

Mahdollisista 16 vaatimuksesta toteutettiin 2 ja osittain toteutettiin 2 vaatimusta. Toteuttamatta jäi 12 mahdollista vaatimusta. Viisi ideatason vaatimusta suljettiin jo alussa projektin ulkopuolelle. Vaatimusmäärittelyyn kirjattiin myös prioriteetilla *Ei toteuteta* 4 vaatimusta, joita sovellukseen ei kannata jatkossakaan toteuttaa.

Tavoitteena oli myös toteuttaa mahdollisuus käyttää sovelluksessa internetkame- roita. Tällaiset kamerat saatiin kuitenkin onnistuneesti asennettua projektiryhmän

käyttöön vasta aivan projektin loppuvaiheilla, joten ominaisuutta ei ehditty toteuttaa.

5.2 Suoritetut testaukset ja niiden tulokset

Sovelluksen toteutusvaiheessa jokainen ryhmän jäsen suoritti omatomaisesti manuaalista yksikkötestausta testaten tekemiään muutoksia.

Projektin lopussa Joel Kivelä laati järjestelmätestaussuunnitelman [4] ja suoritti sitä noudattaen järjestelmätestaukset Windowsille ja Linuxille. Testauksia ei suoritettu todellisessa käyttötilanteessa. Testausten tulokset on raportoitu järjestelmätestausraporteissa [2] ja [3]. Kullakin testauskerralla testattiin yhteensä 55 vaatimusmäärittelyn vaatimusta. Windowsissa suoritettussa järjestelmätestauksessa [2] havaittiin 6 virhettä ja tehtiin 14 muuta huomiota. Linuxissa suoritettussa järjestelmätestauksessa [3] havaittiin 6 virhettä ja tehtiin 9 muuta huomiota.

Kaikki kriittiset virheet korjattiin, minkä jälkeen suoritettiin regressiotestaus, jonka tulokset on raportoitu regressiotestausraporteissa [8] ja [9]. Windowsissa suoritettussa regressiotestauksessa [9] havaittiin 2 virhettä ja tehtiin 8 muuta huomiota. Linuxissa suoritettussa regressiotestauksessa [8] havaittiin 4 virhettä ja tehtiin 6 muuta huomiota.

Projektin aikana järjestettiin käytettävyydestaustapäivä, jonka aikana käytettävyyssiantuntija Johanna Silvennoinen antoi palautetta sovelluksen käytettävyydestä. Varsinaista käytettävyydestausta ei ehditty valmiilla sovelluksella tehdä ajanpuutteen vuoksi.

Projektin aikana sovelluksen eri versioita koekäyttivät myös vastaava ohjaaja Jukka-Pekka Santanen, tekninen ohjaaja Jarkko Vilhunen, sekä tilaajan edustajista Heidi Pasi ja Hanna Toivonen. Erityisesti Jukka-Pekka Santanen ja Heidi Pasi antoivat sovelluksesta palautetta.

5.3 Heikot ja puutteelliset toteutusratkaisut

Videotiedostoja analysoitaessa käyttäjältä tulisi kysyä, halutaanko kopioida analysoitava videotiedosto mittauksen tallennuskansioon. Tällöin sovelluksen kannattaa tallentaa mittauksen XML-tiedostoon kopioidun videon suhteellinen polku. Näin

käytetty video pysyy tallessa ja mittaus voidaan aina avata, kunhan videota ei ole poistettu mittauksen tallennuskansiosta. Tällä hetkellä XML-tiedostoon tallennetaan ainoastaan käytetyn videon absoluuttinen polku, jolloin mittausta ei voi enää käsin XML-tiedostoa muokkaamatta avata, jos video on siirretty toiseen paikkaan.

Käyttäjän on vaikea yhdistää videolähteiden hallintaa, videonäkymiä (erityisesti videonäkymässä olevaa *Set as active* -painiketta) ja *Camera Settings* -välilehteä toisiinsa. Vastaava ohjaaja ehdotti tähän ratkaisuksi mallia, jossa videolähteiden hallintanäkymässä voisi valintaruuduilla osoittaa näkyvissä olevat videolähteet, ja lisäksi napsauttamalla videolähdettä valita, mihin videolähteeseen *Camera Settings* -välilehden kalibrointisäätimet vaikuttavat. Yksi vaihtoehto on myös, että kalibrointisäätimet olisivat käytettävissä ainoastaan, kun videolähteitä on vain yksi. Useamman videolähteen tapauksessa kalibrointi pitäisi periaatteessa olla mahdollista aina tehdä kalibrointipisteiden avulla. Kalibrointi on todennäköisesti sovelluksen käytön vaativin osuus, ja sitä olisi hyvä tukea esimerkiksi käyttöohjeella.

Kalibrointi kameran asetusten välilehden liukusäätimellä on nykyisellään vaikeaa. Kalibrointi on mahdollista toteuttaa myös automaattisesti. Tapoja automaattiseen kalibrointiin on kuvattu muun muassa artikkeleissa [22], [23] ja [24]. Nykyistä yksinkertaisempi tapa olisi myös käyttäjän kuvaan piirtämät X- ja Z-akselien suuntaiset viivat, joiden perusteella voitaisiin laskea kohteiden suhteelliset koordinaatit. Kalibrointipisteitä käytettäessä kalibrointi on yksinkertaista toteuttaa OpenCV:n `cvFindHomography`-metodilla.

Sovelluksen kehityksen aikana pohdittiin, mitä käyttäjälle olisi hyödyllisintä näyttää sovellus käynnistettäessä. Sovellukseen suunniteltiin aloitusnäkömää, jossa olisi ollut painikkeet yleisimpien sovelluksella suoritettavien toimintojen käynnistämiseksi sekä lista viimeisistä sovelluksella suoritetuista ja avatuista mittauksista. Tällaista aloitusnäkömää ei kuitenkaan ehditty toteuttaa.

Kalibrointitiedot viedään tällä hetkellä kameran kierron osalta CAVAPA-ohjelmalle eri tavalla, kuin CAVAPA:n rajapinnan kuvaus määrittelee. Rajapinnan kuvauksen mukaan kierto pitäisi ilmaista kameran kuvaussuuntaan osoittavalla vektorilla. Sen sijaan kierto ilmaistaan nyt X-, Y- ja Z-akselien ympäri tehtyjä kiertoja kuvaavina astelukuina. Tämä ei tällä hetkellä vaikuta sovelluksen toimintaan, koska CAVAPA-ohjelma ei vielä hyödynnä kyseistä tietoa.

5.4 Sovelluksen yleiset kehitysideat

Projektin kuluessa esitettiin seuraavat kehitysideat:

- Videokameroiden kautta tapahtuvaa mittauksia ei voida keskeyttää ja jatkaa myöhemmin (videotiedostojen tapauksessa tämä onnistuu).
- Kaikissa sovelluksen tiedoston tallennus- ja avausikkunoissa olisi hyvä muistaa viimeksi käytetty sijainti ikkunoita uudelleen avattaessa [16].
- Joissakin sovelluksen ikkunoissa on sulkemispainikkeen vieressä kysymysmerkkipainike, jota ei hyödynnetä mitenkään.
- Mittauksen nimi voitaisiin esittää otsikkorivillä, kun se on mahdollista [13].
- *Start measurement* -painikkeen ei tulisi olla nähtävillä tai aktiivisena ennen videolähteiden valintaa [13].
- Mittauksen hallintapainikkeet eivät välttämättä liity kaikkiin välilehtiin, vaan ne kannattaisi ehkä sijoittaa jollekin välilehdistä [19].
- Käyttäjälle voisi sovellus avattaessa ja *Edit*-valikon *Refresh cameras* -komentoa käyttäessä ilmoittaa jotenkin, jos videokameroita ei ole saatavilla [14].
- Kun mittaus pysäytetään, tulisi tilarivillä mainita mittauksen tietojen tallentamisesta [17].
- Ohjelman asetusten (katso luku 3.15), CSV-tiedostoon tallennuksen (katso luku 3.11), uuden merkin lisäyksen (katso luku 3.10) ja videovirran avauksen (katso luku 3.14) ikkunoiden kokoa ei pitäisi pystyä muuttamaan, sillä niiden asettelua ei ole suunniteltu toimimaan muussa kuin oletuskoossa, eikä koon muuttamiselle ole muutenkaan tarvetta.
- Edellisessä kohdassa mainittujen ikkunoiden tulisi olla aina päällimmäisenä, kun sovellus on aktiivisena. Vaihtoehtoisesti aikaisemmin avattu ikkuna tulisi tuoda päällimmäiseksi, jos käyttäjä yrittää avata ikkunan uudelleen sen ollessa jo avattuna. Nykyisellään ikkuna saattaa olla taustalla, jolloin mitään ei tapahdu käyttäjän yrittäessä avata ikkunan uudelleen.

5.5 Mittauksen metatietojen välilehden kehitysideat

Mittauksen metatietojen välilehdelle (katso luku 3.5) esitettiin projektin kuluessa seuraavat kehitysideat:

- *Save*- ja *Cancel*-painikkeilla käyttäjä voisi varmistua syöttämiensä tietojen tulevan tallennetuksi tai hylätyiksi [14].
- *Analysis maximum duration* -kentän nimi ei ehkä kuvaa riittävän selvästi kentän merkitystä [16].
- *Where to save results* -kentässä saattaisi olla hyvä esittää myös mittauksen tiedostoille luotavan kansion nimi. Nykyisessä versiossa kansion nimeksi tulee sama kuin mittauksen nimi, eikä sitä näytetä kuin *Measurement name* -kentässä [17].
- Ohjelman asetusikkunan (katso luku 3.15) asetus *Video file maximum length* saattaisi kuulua tälle välilehdelle [18].
- Kentälle *Where to save results* kannattaisi varata enemmän tilaa [19].
- *Save*-otsikko voisi olla valintaruutujen edessä.

5.6 Aktiivisuusdatan kuvaajien välilehden kehitysideat

Aktiivisuusdatan kuvaajien välilehdelle (katso luku 3.7) esitettiin projektin kuluessa seuraavat kehitysideat:

- Aktiivisuuskuvaajien muoto muuttuu hieman mittauksen aikana johtuen siitä, että kuvaajan pisteet ovat keskiarvoja useammasta mittausarvosta. Kuvaajaa tulisi vierittää siten, että tietty keskiarvo laskettaisiin aina samasta joukosta pisteitä. Näin voitaisiin tehdä ainakin silloin, kun *Fixed Time Window* on päällä. Tällöin kuvaajan muoto pysyisi samana [12].
- Lyhyt käyttöohje voisi kuvata, miten kuvaajat on mahdollista saada tarkasteltavaksi, jos mittausta ei ole aloitettu tai vanhaa mittausta avattu [14].
- Useiden kirjasinkokojen ja tyylien vuoksi välilehti näyttää hieman epäyhtenäiseltä [14].

- Asteikkojen kirjasinkoko sekä hiiri kuvaajien päälle vietäessä kuvaajien yläpuolelle ilmestyvän tekstin kirjasinkoko pienenee sovellusikkunaa kavennettaessa, vaikka isommillekin teksteille olisi tilaa.
- Molemmiin puolin olevien nuolipainikkeiden pitäisi olla harmaana, jos graafia ei pysty siirtämään enää enemmän vasemmalle tai oikealle [14].
- Alaosassa olevat aikakentät tarvitsisivat ehkä vihjetekstin lisäksi otsikot.
- Alaosassa olevien aikakenttien arvo on mittauksen aikana ja sen jälkeen 00:00:00, jos näkymää ei ole zoomattu kertaakaan. Aikakentissä saattaisi olla hyvä näkyä esimerkiksi kuvaajilla esitettävän aikavälin alku- ja loppuaika [17].
- Hiirivalikon toiminnoista *Save to SVG Image* ja *Export to CSV File* avautuvissa tallennusikkunoissa voitaisiin tarjota jotain oletusnimeä tiedostolle [16].
- Y-akselien asteikot tarvitsisivat niitä kuvaavat otsikot [16].
- *Zoom out* -painikkeen kannattaisi zoomata vain tietyn verran, ei aina koko mittauksen aikavälille [16].
- *Realtime*-tekstin tarkoitus ei ole kovin selvä. Teksti näkyy esimerkiksi videotiedostoja analysoitaessa, mikä ei voi olla reaaliaikaista [17].
- Y-akselin asteikoiden olisi hyvä olla sadalla, kymmenellä tai viidellä jaollisia, jotta niiden kokoluokkaa olisi helpompi tulkita [17].
- Kun hiiri viedään pois välilehden kuvaajien yltä, voisi olla hyvä esittää kuvaajien yllä hiiren edellisen sijainnin arvojen sijaan sen ajanhetken arvot, jonka videokuva esitetään videonäkymässä [17].
- Hiiren kohdan arvoissa voisi olla hyvä lisätä ajankohdan eteen sana *time* [17].
- Hiirivalikossa toiminnon *Create Marker* ei tulisi olla käytössä, jos mitään ajanhetkeä ei ole valittu. Samoin toiminnon *Delete marker* ei tulisi olla käytössä, jos mitään merkittyä ajanhetkeä ei ole valittu [17].
- Asetusikkunassa (katso luku 3.12) olisi hyvä olla mahdollisuus tallentaa asetukset tiedostoon, jotta voitaisiin tallentaa eri asetukset eri tilanteita varten [17].

- *Show*-valintaruudut voisivat olla alleviivattuja värillä, jolla kyseiset asiat esitetään kuvaajissa [19].
- *Zoom In* -painikkeen tulisi toimia, vaikka tiettyä aluetta ei olisikaan valittu. Tällöin zoomattaisiin tietty määrä näkyvän aikavälin keskikohdasta [19].
- Nykyisessä toteutuksessa aktiivisuuskuvaajat skaalautuvat mittauksen aikana viimeisimpien havaittujen maksimiarvojen mukaan. Voisi olla hyvä tarjota mahdollisuus asettaa asteikkojen maksimiarvot myös käsin [20].
- Jos mittaus on käynnissä, mutta mittauksesta tarkastellaan aiempaa aikajännettä, olisi hyvä päivittää tarkasteltavan aikavälin kuvaajat, jos mittauksessa havaitaan uusi maksimiarvo [21].

5.7 Uuden merkin lisäysikkunan kehitysideat

Uuden merkin lisäysikkunalle (katso luku 3.10) esitettiin projektin kuluessa seuraavat kehitysideat:

- Syöttökenttien tekstien ja kenttien välissä on liian paljon tilaa. Hahmolakien mukaan yhteen liittyvien asioiden tulisi olla lähempänä toisiaan kuin erillisten [16].
- Lisättävän merkin nimeä kirjoitettaessa nimen olisi hyvä näkyä kuvaajissa jo ennen *Save*-painikkeen painamista [16].
- Jos kuvaajista ei ole valittu mitään kohtaa, olisi loogista merkitä ajanhetki, jonka kohdalla hiirivalikko on avattu *Create Marker* -toimintoa valitessa [19].

5.8 CSV-tiedostoon tallennusikkunan kehitysideat

CSV-tiedostoon tallennusikkunalle (katso luku 3.11) esitettiin projektin kuluessa seuraavat kehitysideat:

- *Time offset* -kentän merkitystä olisi hyvä tarkentaa esimerkiksi vihjetekstin avulla [16].

- *From-* ja *To-*otsikoilla tarjottaviin aikakenttiin ja merkittyjen ajanhetkien valintaan liittyen ikkunassa olisi hyvä olla valintapainike sille, käytetäänkö aikakenttiä vai merkittyjä ajanhetkiä [16].
- Ikkunan asettelu on hieman muusta sovelluksesta poikkeavaa. Lihavoitujen otsikoiden sijaan kannattaisi käyttää ryhmälaatikoita, kuten esimerkiksi kameran asetusten välilehdellä (katso luku 3.6).
- CSV-tiedostoon tulisi olla mahdollista viedä myös kaikki kerätty data ilman keskiarvojen ottamista. Tällä hetkellä tarkin mahdollinen datan vienti vie keskiarvot sekunnin välein.
- Ikkunassa kannattaisi olla valintaruutu koko mittauksen datan viennille, jolloin valitulla aikavälillä ei olisi merkitystä.
- Jos valitun aikajänteen pituus on nolla, pitäisi käyttäjää varoittaa.
- Jos CSV-tiedostoon tallennus on jo kerran tehty, on *File to Save* -kentässä oletuksena aiemmin tallennetun tiedoston polku. *Export*-painiketta ei voi silti painaa, ennen kuin on käynyt tiedostopolun valintaikkunassa tai muokannut tiedostopolkua.

5.9 Kameran asetusten välilehden kehitysideat

Kameran asetusten välilehdelle 3.6 esitettiin projektin kuluessa seuraavat kehitysideat:

- Liukusäätimien arvoja voi muuttaa, vaikka yhtään videolähdettä ei olisi valittuna. Tämän ei tulisi olla mahdollista [15].
- Syöttökenttiin kirjaimia sijoittaessa arvoksi tulee 0. Tällöin tulisi säilyttää edellinen arvo ja mahdollisesti huomauttaa virheellisestä syötteestä. Myös käytävissä olevan arvoalueen ulkopuolella olevasta syötteestä tulisi ilmoittaa [15].
- Kalibrointitiedoille tulisi tarjota mahdollisuus tallentaa ne omaan tiedostoonsa [15].
- Liukusäätimien mitta-asteikot tai ainakin minimi- ja maksimiarvojen esittäminen selkeyttäisi käyttöä [15].

- Kun mittaus pysäytetään, kameran asetuksista nollaantuvat kierto X-akselin ja Y-akselin suhteen.
- Ohjelman asetusikkunan (katso luku 3.15) asetus *Camera FPS* saattaisi kuulua tälle välilehdelle [18].
- Saman nimisiä USB-kameroita käyttäessä aiemmin käytetyt kalibrointitiedot eivät lataudu oikein. Kamerat tulisi pyrkiä yksilöimään nimen lisäksi esimerkiksi käytetyn USB-portin perusteella.

5.10 Videolähteiden hallintanäkymän kehitysideat

Videolähteiden hallintanäkymälle (katso luku 3.4) esitettiin projektin kuluessa seuraavat kehitysideat:

- Painikkeiden tulisi sijaita videolähdelistan alapuolella [17].
- Videolähteen nimelle varattu tila on liian pieni, sillä siihen mahtuu vain kymmenkunta merkkiä. Näkymän käyttämää tilaa tulisi kyetä säätämään sen ja videonäkymien välissä olevaa erotinta raahaamalla.
- Näkymässä olevan listan videolähteitä on hankala yhdistää videonäkymiin. Videolähteen nimi olisi hyvä olla näkyvissä videonäkymän yhteydessä esimerkiksi videonäkymän yllä tai vihjetekstinä [17].
- Videolähteinä voi olla samanaikaisesti videokameroita, -tiedostoja ja -virtoja. Tämä ei välttämättä ole kovin järkevää. Videokameroiden ja -virtojen yhtäaikainen käyttö saattaisi jossain melko harvinaisessa tilanteessa olla tarpeen.
- Videolähteen sulkeminen kuuluisi loogisemmin videonäkymän (katso luku 3.3) ruksipainikkeen sijaan tähän näkymään [17].

5.11 Lokivälilehden kehitysideat

Lokivälilehdelle (katso luku 3.8) esitettiin projektin kuluessa seuraavat kehitysideat:

- Viestit tulisi tallentaa tiedostoon.
- Virheviestit voisi tuoda käyttäjälle selvemmin näkyviin esimerkiksi punaisella huutomerkillä välilehden otsikossa [17].

5.12 Videonäkymän kehitysideat

Videonäkymälle (katso luku 3.3) esitettiin projektin kuluessa seuraavat kehitysideat:

- Aikakenttä voisi ilmaista, missä kohdassa video on menossa [12].
- Näkymä jää sovellusta pienellä resoluutiolla käytettäessä liian pieneksi, kuten myös kalibrointipisteiden numeroinnin kirjasinkoko. Tähän auttaisi näkymä, jossa olisi mahdollisuus tarkastella yhtä videonäkymää kalibrointikontrolleineen koko ruudun kokoisena [13].
- Videotiedostoja tulisi voida kelata myös esimerkiksi ruutu kerrallaan, jotta tarkka synkronointi olisi helpompaa.
- Linuxilla kalibrointipisteiden numerot näkyvät videonäkymässä (katso luku 3.3) pikselisotkuna. Ongelma vaikuttaa olevan `QPainter`-luokan `drawText`-metodissa. Myös Qt:n omat värinvalintaikkunat näkyvät ajoittain Linuxilla vain tyhjinä ikkunoina.

5.13 Pääikkunan komentovalikoiden kehitysideat

Pääikkunan komentovalikoille (katso luku 3.13) esitettiin projektin kuluessa seuraavat kehitysideat:

- Osassa valikoita on ainoastaan yksi komento, joten valikoita kannattaisi yhdistellä.

- *Edit*-valikon komento *Refresh cameras* ei pitäisi olla käytettävissä, kun mittaus on käynnissä [18].
- *Help*-valikon *About*-toiminnolla aukeavassa ikkunassa (katso luku 3.16) voisi lyhyesti mainita myös ohjelman käyttötarkoituksesta [18].
- *Tools*-valikon *Settings*-toiminnolla aukeavalla ohjelman asetusikkunalla (katso luku 3.15) voisi olla kuvaavampi otsikko [18].
- *Tools*-valikon *Settings*-toiminnolla aukeavassa ohjelman asetusikkunassa (katso luku 3.15) kentällä *Default directory* voisi olla kuvaavampi nimi. Kenttä määrittää mittauksen tietojen tallennukseen mittauksen metatietojen välilehdellä (katso luku 3.5) tarjottavan oletuskansion [18].
- *Tools*-valikon *Settings*-toiminnolla aukeavan ohjelman asetusikkunan (katso luku 3.15) ohjaimia kannattaisi ryhmitellä [18].

6 Ohjeita ylläpitäjälle ja jatkokehittäjälle

Luvussa esitetään ohjeita sovelluksen ylläpitäjälle ja jatkokehittäjälle.

6.1 Ohjeita jatkokehittäjälle

Käännettäessä Qt-projektia Windowsilla kehityskoneella on oltava asetettuna ympäristömuuttuja `OPENCV_DIR`, joka osoittaa OpenCV-kirjaston asennuskansion. Ympäristömuuttujaan asetetaan kansio, joka sisältää OpenCV:n `bin`-, `include`- ja `lib`-kansiot.

Toteutusratkaisuihin kannattaisi tehdä seuraavat muutokset:

- `AnalysisController`-luokka tulisi jakaa ainakin kahteen eri luokkaan. Nykyisestä luokasta tulisi eriyttää eräänlainen `VideoPlayer`-luokka, joka ajaisi erillään laskelman ja tulosten pyörittämistä.
- `VideoFileSet`-luokka pitäisi kirjoittaa siltä osin uusiksi, että se tekisi kaiken yhdessä säikeessä. Nykyisellään säikeitä on yhtä monta kuin pätkittyjä videotiedostoja.
- CAVAPA-ohjelmalle viedään tällä hetkellä ohjelman vaatiman kameran osoitussuunnan sijaan kameran rotaatiot eri akseleiden suhteen. Ohjelma ei tällä hetkellä hyödynnä kameran osoitussuuntaa. Rotaatioista eri akseleiden suhteen olisi laskettava vektori, jonka suuntaan kamera osoittaa.
- Doxygenillä luodun luokkadokumentation LaTeX-versiossa muutaman metodin kuvaus katkeaa tuntemattomasta syystä sivunvaihdolla.

6.2 Ohjeita ylläpitäjälle

Sovellus toimii sovelluspaketin purkamisen jälkeen ajamalla paketissa oleva `cavapa-gui`-ohjelma. Sovellusta asennettaessa on otettava huomioon seuraavat rajoitteet:

- Sovellus toimii testatusti Windows 7 -käyttöjärjestelmässä.

- Sovellus toimii testatusti Linux-pohjaisissa käyttöjärjestelmissä Fedora (versio 18) ja Ubuntu (versio 14.04) vaatimusmäärittelyn [10] luvussa 6.1 kuvattuja virheitä lukuunottamatta.
- Videokameran lukemista varten mahdolliset siihen liittyvät ajurit on oltava asennettuina työasemalla.
- Videokameran lukeminen ei ole mahdollista, jos OpenCV-kirjasto ei tue kameran rajapintaa.

6.3 Kokemuksia käytetyistä kameroista

Projektiryhmä käytti kehitystyössä kolmentyyppisiä kameroita:

- PlayStation Eye
- Microsoftin webkamera, jonka merkkiä projektiryhmä ei muistanut ottaa ylös.
- Firewire-kamera, jonka merkkiä projektiryhmä ei muistanut ottaa ylös.

Seuraavat kokemukset kameroista pohjautuvat Erkki Koskenkorvan koostamaan yhteenvetoon [11].

PlayStation Eye -kamerat ovat edullisia ja yleisiä. Niissä oli nopea kuvanpäivitys ja melko hyvä kuvanlaatu. Ainoa havaittu haittapuoli oli puuttuva mahdollisuus resoluution vaihtoon.

Microsoftin webkameroista jäi yleisesti huono vaikutelma. Yksi ei toiminut, ja muissa kuvan värisävy oli välillä outo. Lisäksi kameroiden toiminta hidastui, jos niiltä pyydettiin kuvaa liian usein.

Firewire-kameroista vain yksi toimi. Ne vaikuttivat olevan tarkoitettuja johonkin automaatiokäyttöön. Saatu kuva oli mustavalkoista.

7 Yhteenveto

Liikkuva-projekti toteutti Sovellusprojekti-kurssilla keväällä 2014 Jyväskylän yliopiston liikuntakasvatuksen laitokselle sovelluksen, jolla voidaan mitata ryhmätason fyysistä aktiivisuutta. Sovelluksella käyttäjä analysoi videokuvaa liikuntatilanteesta. Videolähteenä voi olla tietokoneeseen kytketty videokamera tai aikaisemmin kuvattu videotiedosto. Tulokset ovat tarkempia, jos tilanne on kuvattu samaan aikaan useammasta kuvakulmasta.

Sovelluksessa käyttäjä täyttää ensin mittaukseen liittyvät oheistiedot ja kalibroi sitten analyysiohjelman joko arvioimalla kameran tai kameroiden sijaintia ja suuntaa tai merkitsemällä useampaa kameraa käytettäessä videokuvista yhteisiä tilan pisteitä. Tämän jälkeen mittaus voidaan käynnistää. Mittauksen aikana ja sen jälkeen käyttäjä voi tarkastella aktiivisuusdataa esittäviä kuvaajia ja halutessaan nimetä niihin tiettyjä ajanhetkiä. Mittauksen jälkeen käyttäjä voi viedä mittausdataa taulukkolaskentaohjelmalla avattavissa olevaan CSV-tiedostoon tai tallentaa kuvaajat tai osan niistä SVG-vektorigrafiikkakuvana.

Toteutettu järjestelmäkokonaisuus toimii pääosin asetettujen tavoitteiden mukaisesti, mutta joitain yksittäisiä virheitä ei ollut aikaa korjata projektin puitteissa. Ensimmäisen prioriteetin 33:sta vaatimuksista saatiin toteutettua 32. Projektiryhmästä riippumattomista syistä sovellus ei kuitenkaan ole vielä tuotantokäyttökelpoinen, sillä internetkameroiden käyttöä ei voitu toteuttaa laitteiston puuttumisen vuoksi. Lisäksi CAVAPA-analyysiohjelma toimii vielä puutteellisesti, eikä tuota luotettavaa analyysidataa. Jatkokehitysideoita esitettiin myös projektin aikana lukuisia.

Sovellus toteutettiin Qt-käyttöliittymäkirjastoa ja -sovelluskehitysympäristöä hyödyntäen. Sovellus hyödyntää myös Jarkko Vilhusen kehittämää CAVAPA-analyysiohjelmaa, OpenGL-grafiikkakirjastoa ja OpenCV-konenäkökirjastoa. Sovelluksen kokonaisrakenteen merkittävimmät komponentit ovat analyysiohjain, ohjain ja graafinen käyttöliittymä.

8 Lähteet

- [1] Tapio Keränen, Toni Salminen, Jari Salokangas ja Lauri Satokangas, "Paatti-sovellusprojekti, Sovellusraportti", saatavilla PDF-muodossa <URL: http://sovellusprojektit.it.jyu.fi/paatti/dokumentit/sovellusraportti/paatti_sovellusraportti_1.0.0.pdf>, Jyväskylän yliopisto, tietotekniikan laitos, 7.6.2012.
- [2] Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Järjestelmätestausraportti #1", Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [3] Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Järjestelmätestausraportti #2", Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [4] Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Järjestelmätestaussuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [5] Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Luokkadokumentaatio", Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [6] Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Projektiraportti", Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [7] Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Projektisuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [8] Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Regressiotestausraportti #1", Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [9] Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Regressiotestausraportti #2", Jyväskylän yliopisto, tietotekniikan laitos, 2014.

- [10] Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Vaatimusmäärittely", Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [11] Erkki Koskenkorva, "Liikkuva-projektin kamerat", sähköposti, lähetetty 6.6.2014.
- [12] Heidi Pasi, "VS: Sovelluksen uusin prototyyppi (29.4.2014)", sähköpostipalaute, Jyväskylän yliopisto, liikuntakasvatuksen laitos, lähetetty 1.5.2014.
- [13] Jukka-Pekka Santanen, "Re: Sovelluksen uusin prototyyppi", sähköpostipalaute, Jyväskylän yliopisto, tietotekniikan laitos, lähetetty 9.4.2014.
- [14] Jukka-Pekka Santanen, "Re: Sovelluksen uusin prototyyppi", sähköpostipalaute, Jyväskylän yliopisto, tietotekniikan laitos, lähetetty 15.4.2014.
- [15] Jukka-Pekka Santanen, "Re: Sovelluksen uusin prototyyppi", sähköpostipalaute, Jyväskylän yliopisto, tietotekniikan laitos, lähetetty 16.4.2014 klo 16.25.
- [16] Jukka-Pekka Santanen, "Re: Sovelluksen uusin prototyyppi", sähköpostipalaute, Jyväskylän yliopisto, tietotekniikan laitos, lähetetty 16.4.2014 klo 19.02.
- [17] Jukka-Pekka Santanen, "Re: Sovelluksen uusin prototyyppi (29.4.2014)", sähköpostipalaute, Jyväskylän yliopisto, tietotekniikan laitos, lähetetty 29.4.2014.
- [18] Jukka-Pekka Santanen, "Re: Sovelluksen uusin prototyyppi (29.4.2014)", sähköpostipalaute, Jyväskylän yliopisto, tietotekniikan laitos, lähetetty 5.5.2014.
- [19] Jukka-Pekka Santanen, "Re: Viimeisin prototyyppi", sähköpostipalaute, Jyväskylän yliopisto, tietotekniikan laitos, lähetetty 13.5.2014.
- [20] Jukka-Pekka Santanen, "Re: Sovellusraportti v.0.4.0", sähköpostipalaute, Jyväskylän yliopisto, tietotekniikan laitos, lähetetty 2.6.2014.
- [21] Jukka-Pekka Santanen, "Re: Sovellusraportti v.0.6.0", sähköpostipalaute, Jyväskylän yliopisto, tietotekniikan laitos, lähetetty 6.6.2014.
- [22] Se, Stephen ja Michael Brady. "Ground Plane Estimation, Error Analysis and Applications." *Robotics and Autonomous Systems* 39.2, s. 59–71, 2002.
- [23] Cherian, Anoop, Vassilios Morellas ja Nikolaos Papanikolopoulos. "Accurate 3D Ground Plane Estimation from a Single Image." *Robotics and Automation, 2009, ICRA'09, IEEE International Conference on, IEEE*, s. 2243–2249, 2009.

- [24] Zhao, Jun, Jayantha Katupitiya ja James Ward. "Global Correlation Based Ground Plane Estimation Using V-Disparity Image." Robotics and Automation, 2007 IEEE International Conference on, IEEE, s. 529–534, 2007.