

Liikkuva-sovellusprojekti

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

Vaatimusmäärittely

**Julkinen
Versio 0.5.0
20.3.2014**

**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.koskenkorva@student.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, Vaatusmääritys

Sivumäärä: 17

Tiivistelmä: Dokumentissa kuvataan Liikkuva-projektissa toteutettavan sovelluksen toiminnalliset ja tekniset vaatimukset sekä rajoitteet. Lisäksi kuvataan vaatimusten prioriteetit ja tilat.

Avainsanat: Aktiivisuustaso, CAVAPA, käyttöliittymä, liikemittari, liikemäärä, rajoitteet, sovelluksen kokonaisrakenne, tekniset vaatimukset, toiminnalliset vaatimukset.

Muutoshistoria

Versio	Päivämäärä	Muutokset	Tekijät
0.0.1	6.2.2014	Vaatusmäärityksen laatiminen on aloitettu.	ML
0.0.2	7.2.2014	Aloitettu luku <i>Vaatusmääritys</i> ja sen alaluvut.	ML
0.0.3	10.2.2014	Lisätty prioriteettien ja tilojen pikkukuvat. Lisätty muutoshistoriasivu ja alustavat vaatusmääritykset.	ML
0.0.4	11.2.2014	Lisätty alustavat tekniset vaatusmääritykset. Muokattu listojen sisennyksiä.	ML
0.0.5	12.2.2014	Muunnettu toiminnalliset vaatusmääritykset kokonaisiksi virkkeiksi. Lisätty vaatusmääritys liikemääräkäytön tallentamisesta.	ML
0.1.0	13.2.2014	Lisätty johdanto.	ML
0.1.1	17.2.2014	Päivitetty määritelmiä. Jaettu videon käsittely kahdeksi kategoriaksi. Lisätty kolme vaatusmääritystä tulosten näyttämiseen liittyen.	ML
0.2.0	18.2.2014	Päivitetty määritelmiä. Lisätty määritelmät <i>CAVAPA-algoritmi</i> ja <i>Tynnyrivääritys</i> . Korjailtu kieliasua.	ML
0.2.1	24.2.2014	Päivitetty vaatusmäärityskokonaisuuksia. Lisätty teknisiä vaatusmäärityksiä formaattien osalta. Parannettu kieliasua.	ML
0.2.2	25.2.2014	Vaatusmääritykset numeroitu. Jaettu luku <i>Vaatusmääritys</i> kolmeen lukuun.	ML
0.2.3	26.2.2014	Aloitettu luku <i>Sovelluksen kokonaisrakenne</i> .	ML
0.2.4	27.2.2014	Tarkennettu sovelluksen rakennetta.	ML
0.3.0	28.2.2014	Tarkennettu vaatusmäärityksiä metatietojen syöttämiseen liittyen. Lisätty alaluku <i>Tekniset termit</i> .	ML
0.3.1	4.3.2014	Kieliasua korjailtu. Tietojärjestelmää kuvaavia ja teknisiä termejä lisätty.	ML
0.3.2	7.3.2014	Lisätty CAVAPA-ohjelman rajapinnan kuvaus. Päivitetty vaatusmäärityksiä.	ML
0.4.0	10.3.2014	Lisätty yhteenveto ja alaluku <i>Rajoitteet</i> . Päivitetty vaatusmääritysten tiloja.	ML

Versio	Päivämäärä	Muutokset	Tekijät
0.4.1	12.3.2014	Lisätty termit <i>Aktiivisuusdata</i> ja <i>CSV</i> . Korjattu kieliasua. Lisätty kaksi rajoitetta. Jaettu vaatimuskokonaisuus <i>Mittauksen aloitus ja seuranta</i> kahtia.	ML
0.4.2	17.3.2014	Tarkennettu havainnon ja kohteen määritelmiä. Lisätty tekniset termit <i>FPS</i> ja <i>XML</i> . Lisätty rajoite <i>FPS</i> :ään liittyen. Lisätty <i>OpenGL</i> käytettyihin kirjastoihin. Lisätty vaatimuskokonaisuudet <i>Yleiset asetukset</i> ja <i>Aiemman mittauksen käsittely</i> . Päivitetty vaatimuksia.	ML
0.5.0	20.3.2014	Tarkennettu rajoitteita. Tarkennettu konenäkövaatimusta. Lisätty vaatimus viimeksi käytetyistä videolähteistä. Päivitetty vaatimusten prioriteetteja ja tiloja.	ML

Tietoa projektista

Liikkuva-sovellusprojekti kehittää Jyväskylän yliopiston liikuntakasvatuksen laitoselle käyttöliittymän konenäköpohjaiselle liikemittarille, jolla voidaan visualisoida videokuvasta mitattua ihmisjoukon aktiivisuustasoa sekä suorittaa mittauksessa tarvittava kalibrointi.

Tekijät:

- Joel Kivelä (JK) joel.a.kivela@student.jyu.fi
- Erkki Koskenkorva (EK) erkki.koskenkorva@student.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

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: <https://korppi.jyu.fi/list-archive/liikkuva/>,
https://korppi.jyu.fi/list-archive/liikkuva_opetus/

Sisältö

1	Johdanto	1
2	Termit	2
2.1	Aihealueen termit	2
2.2	Tietojärjestelmän termit	3
2.3	Tekniset termit	4
3	Sovelluksen kokonaisrakenne	5
3.1	Käytetyt kirjastot	5
3.2	CAVAPA-ohjelma	5
3.3	Kokonaisrakenne	6
4	Vaatimusten prioriteetit ja tilat	8
5	Toiminnalliset vaatimukset	9
5.1	Yleiset asetukset	9
5.2	Aiemman mittauksen käsittely	9
5.3	Videolähteiden valinta	9
5.4	Mittauksen tietojen määrittäminen	9
5.5	Kalibrointi	10
5.6	Mittauksen aloitus	10
5.7	Mittauksen seuranta	11
5.8	Tulosten näyttäminen ja käsittely	12
5.9	Konenäkö	12
5.10	WWW-käyttöliittymä	13
6	Tekniset vaatimukset ja rajoitteet	14
6.1	Käyttöjärjestelmät	14
6.2	Videon tallennusformaatit	14
6.3	Videon lukuformaatit	14
6.4	Liikemääräkäyrän tallennusformaatit	14
6.5	Metatiedon tallennusformaatit	15
6.6	Rajoitteet	15
7	Yhteenveto	16

8 Lähteet

17

1 Johdanto

Ryhmäliikunnan tutkimuksessa liikemäärän selvittämiseksi käytetään tavallisesti henkilöihin kiinnitettäviä laitteita tai tutkijoiden silmämääräistä arviointia. Nämä menetelmät voivat kuitenkin olla kalliita ja/tai aikaavieviä, sekä etenkin pidemmän aikavälin liikemäärän seuranta on hankalaa.

Liikuntakasvatuksen laitos ja tietotekniikan laitos ovat yhteistyössä kehittäneet ohjelmiston, joka mahdollistaa ihmisjoukon liikemäärän mittaamisen videokuvasta. Kyseisen ohjelmiston käyttäminen on kuitenkin vaativaa, koska sillä ei ole kunnollista käyttöliittymää.

Liikkuva-projekti suunnittelee ja toteuttaa käyttöliittymän kyseiselle konenäköpohjaiselle liikemittarille, jolla voidaan visualisoida videokuvasta mitattua ihmisjoukon aktiivisuustasoa sekä suorittaa mittauksessa tarvittava kalibrointi. Sovellusprojektin kokonaistavoitteena on siis mahdollistaa liikemittarin helppo käyttö liikuntakasvatuksen laitoksen tutkijoille ja opiskelijoille. Projektin taustoja, tarpeita ja tavoitteita sekä kehitettävällä tietojärjestelmällä tuettavaa prosessia kuvataan tarkemmin projektisuunnitelmassa [1].

Vaatusmääritys kuvaa Liikkuva-projektissa toteutettavan sovelluksen toiminnalliset ja tekniset vaatimukset sekä kokonaisuuden. Dokumenttiin päivitetään myös vaatimusten prioriteetit ja niiden toteutumisen tila.

2 Termit

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

2.1 Aihealueen termit

Projektin aihealueen termejä ovat seuraavat:

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	on Jyväskylän yliopiston liikuntakasvatuksen ja tietotekniikan laitoksen kehittämä ryhmätason fyysistä aktiivisuutta mittaava tietokoneavusteinen menetelmä. Se on lyhenne sanoista <i>Computer Assisted Video Analysis of Physical Activity on group level</i> .
CAVAPA-algoritmi	on menetelmä, joka laskee yhden tai useamman videokuvan ja tarvittavien parametrien perusteella aktiivisuusdatan.
Kalibrointi	sisältää ne toimenpiteet, joilla varmistetaan, että aktiivisuusdata mitataan videokuvasta yhdenmukaisesti.
Liikemäärä	on fysikaalinen suure, joka on suoraan verrannollinen aktiivisuustasoon.
Linssivääristymä	on optiikan ilmiö, jossa todellisen maailman suorat viivat näyttävät kameran kuvassa vinoutuneilta.
Perspektiivin korjaus	tarkoittaa CAVAPA-algoritmin alustamista sellaisilla parametreilla, että se saa käsityksen oikeasta kuva-

kulmasta. Käytännössä tämä tarkoittaa kaksiulotteisen ruudun sovittamista videokuvan päälle.

Tynnyrivääristymä on erikoistapaus linssivääristymästä. Se aiheuttaa suorien viivojen kaartumisen ulospäin.

2.2 Tietojärjestelmän termit

Tietojärjestelmän osia ja ominaisuuksia kuvaavia käsitteitä ovat luvussa 2.1 kuvattujen ohella seuraavat:

Analyysi	tarkoittaa aktiivisuusdatan muodostamista videokuvan perusteella.
CAVAPA-ohjelma	(engl. <i>Cavapa program</i>) on toteutus CAVAPA-algoritmista.
CAVAPA-GUI	on Liikkuva-projektissa toteutettavan sovelluksen työnimi.
Havainto	(engl. <i>sighting</i>) on CAVAPA-algoritmin havaitsema kohde.
Kohde	on videokuvassa esiintyvä liikkuva hahmo.
Kohteen korostus	tarkoittaa suorakulmion piirtämistä videokuvaan kyseisen kohteen reunoille.
Liikemääräkäyrä	on kaksiulotteinen kuvaaja, joka kuvaa liikemäärän ajan funktiona.
Lämpökuva	on kaksiulotteinen kuva, joka kuvaa, millä alueilla videokuvassa aktiivisuutta esiintyi määritellyllä aikavälillä.
Mittaus	on prosessi, jossa CAVAPA-algoritmille syötetään videolähteistä saatavaa videokuvaa ja algoritmin vasteita otetaan talteen.
Ryhmä	on ihmisjoukko, jonka aktiivisuutta videokuvasta mitataan.
Videolähde	on joko videotiedosto tai videokamera.

2.3 Tekniset termit

Vaatusmäärittelyssä esiintyviä tiedostoformaatteihin ja toteutustekniikoihin liittyviä termejä ovat seuraavat:

CSV	on tiedostomuoto, jolla tallennetaan taulukkomuotoista tietoa tekstitiedostoon.
FPS	on lukuarvo, joka kertoo, montako kuvapäivitystä videolähteestä otetaan yhden sekunnin aikana.
JPEG	eli JPG on häviöllinen kuvatiedostoformaatti.
MJPEG	eli MJPG on videokuvan pakkaustekniikka, jossa jokainen videon ruutu pakataan JPG-kuvana.
MPEG-4	on MJPG:tä edistyneempi videokuvan pakkaustekniikka.
PNG	on häviötön kuvatiedostoformaatti.
SVG	on vektorikuvaformaatti.
XML	on tekstimuotoinen merkintäkieli, jolla tiedon merkitys voidaan kuvata tiedon yhteyteen.

3 Sovelluksen kokonaisrakenne

Luvussa kuvataan Liikkuva-projektissa kehitettävän CAVAPA-GUI-sovelluksen komponentit ja niiden suhteet toisiinsa sekä sovelluksen käyttämät aliohjelmakirjastot. Koko tietojärjestelmä koostuu CAVAPA-GUI:sta, CAVAPA-ohjelmasta, videokameroista ja mahdollisista muista laitteista.

3.1 Käytetyt kirjastot

CAVAPA-GUI käyttää seuraavia kirjastoja:

- OpenCV** (*Open Source Computer Vision Library*) on avoimen lähdekoodin kone-näkökirjasto. CAVAPA-GUI käyttää sitä videotiedostojen ja kameran lukemiseen sekä kameralta tulevan kuvan tallentamiseen.
- OpenGL** (*Open Graphics Library*) on laitteistoriippumaton grafiikkakirjasto. CAVAPA-GUI käyttää sitä videokuvan ja siihen liittyvän kalibrointi-grafiikan esittämiseen.
- Qt** on avoimen lähdekoodin käyttöliittymä- ja sovelluskehys. CAVAPA-GUI käyttää sitä graafisen käyttöliittymän esittämisessä.

3.2 CAVAPA-ohjelma

Luvussa 3.1 mainittujen kirjastojen lisäksi CAVAPA-GUI käyttää tilaajan kehittämää analyysiohjelmaa, joka laskee annetun videokuvan perusteella siihen liittyvän kokonaisliikemäärän ajan funktiona sekä havaittujen hahmojen sijainnit ajan funktiona. Liikkuva-projekti ei tee muutoksia CAVAPA-ohjelmaan, vaan tarvittaessa ohjelman kehittänyt Jarkko Vilhunen.

CAVAPA-ohjelman tarjoama rajapinta muodostuu yksittäisestä luokasta `Detector`, joka tarvitsee tiedot kameroista ja vapaaehtoisesti listan kalibraatiopisteistä, joiden avulla useamman kameran sijainnin määrittäminen tarkentuu.

Yksittäisen kameran tietoja ovat seuraavat:

- kamerakuvan resoluutio,
- kameran sijainti,
- kameran kuvaussuunta,
- näkökentän leveys,
- tynnyriväärityksen voimakkuus.

Videokuvan analysointia varten `Detector`-luokassa on metodi `compute`, joka ottaa vastaan yksittäisen kuvaruudun kustakin videolähteestä ja palauttaa jokaista videolähdettä kohti joukon havaintoja (`Sighting`).

Yksittäinen havainto muodostuu seuraavista tiedoista:

- kohteen tunniste,
- kohteen vasemman yläkulman koordinaatti videokuvassa,
- kohteen oikean alanurkan koordinaatti videokuvassa,
- kalibroinnin pohjalta tehty arvio kohteen sijainnista maailmassa,
- arvioitu havainnon luotettavuus.

3.3 Kokonaisrakenne

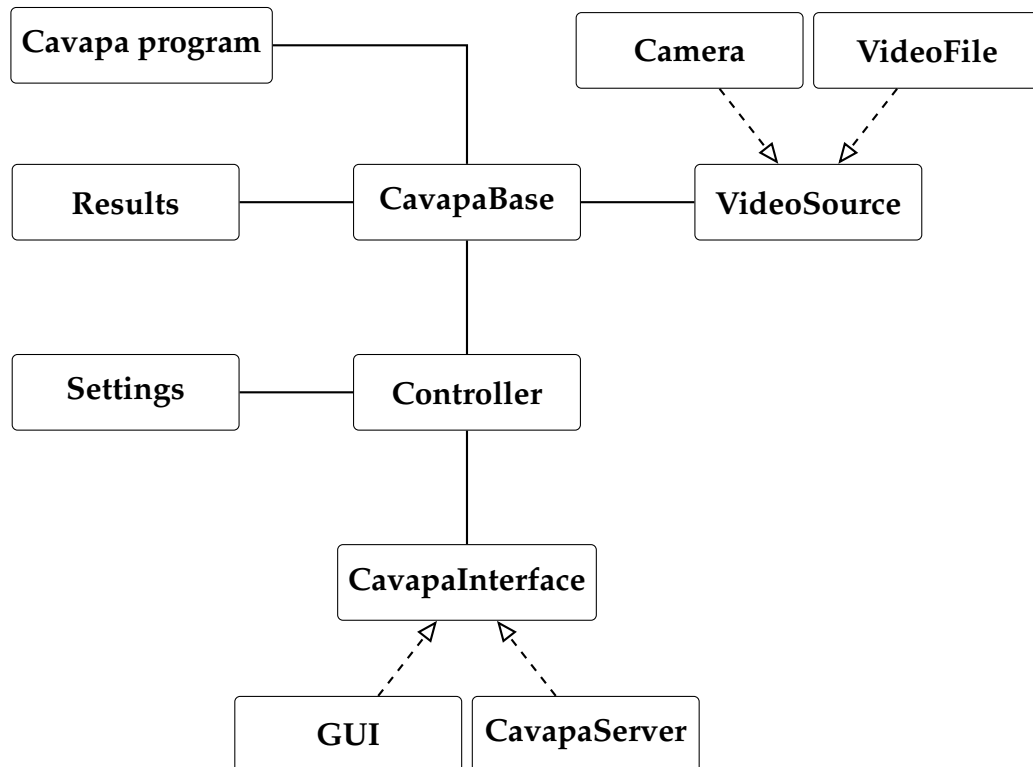
Sovelluksen kokonaisrakenne on esitetty kuvassa 3.1. Komponentti `CavapaBase` vastaanottaa videokuvaa yhdestä tai useammasta videolähteestä ja lähettää sitä eteenpäin analyysiohjelmalle *Cavapa program*, joka puolestaan laskee videokuvan perusteella aktiivisuusdatan.

Ohjainkomponentti `Controller`

- tallentaa, lukee ja välittää sovelluksen tarvitsemia asetuksia `Settings`-komponentin avustuksella,
- vastaanottaa komentoja käyttöliittymärajapinnasta `CavapaInterface` ja lähettää niitä eteenpäin `CavapaBase`-komponentille sekä
- ottaa vastaan laskentatietoa `CavapaBase`-komponentilta ja välittää sitä käyttöliittymälle.

Käyttöliittymärajapinnan toteuttava komponentti `GUI` on työpöytäkäyttöliittymä, joka huolehtii videoiden ja laskentatiedon esityksestä käyttäjälle sekä ottaa vastaan käyttäjän syötteitä ja komentoja.

Käyttöliittymärajapinta mahdollistaa myös muunlaisten käyttöliittymien liittämisen sovellukseen. Yksi mahdollinen tällainen on palvelinkomponentti `CavapaServer`, jonka kautta sovellusta voisi käyttää WWW-käyttöliittymän kautta. Käyttöliittymistä GUI on ensisijainen, ja `CavapaServer` toteutetaan ajan salliessa.



Kuva 3.1: CAVAPA-GUI-sovelluksen kokonaisrakenne.







4 Vaatimusten prioriteetit ja tilat

Luvussa kuvataan toteutettavan sovelluksen vaatimusten mahdolliset prioriteetit ja tilat.

Vaatimukset on priorisoitu asteikolla 1–5 seuraavasti:

- ① Pakollinen
- ② Tärkeä
- ③ Mahdollinen
- ④ Idea
- ⑤ Ei toteuteta

Vaatimusten tilat esitetään seuraavasti:

-  Ulkopuolinen moduuli
-  Hyväksytty
-  Testattu
-  Toteutettu
-  Osittain toteutettu
-  Toteuttamatta

5 Toiminnalliset vaatimukset

5.1 Yleiset asetukset

- ② ✘ 5.1.1 Käyttäjä voi valita liikemääräkäyrän värit.
- ② ✘ 5.1.2 Käyttäjä voi valita oletushakemiston, jonne mittausten tulokset tallennetaan.
- ② ✘ 5.1.3 Käyttäjä voi valita kohteen korostuksen värin.
- ② ✘ 5.1.4 Käyttäjä voi määrittää videolähteiden FPS:n.
- ② ✘ 5.1.5 Käyttäjä voi määrittää, miten pitkiin osiin video pilkotaan kamerakuvaa tallennettaessa.

5.2 Aiemman mittauksen käsittely











- ② ✘ 5.2.1 Käyttäjä voi avata aiemman mittauksen tiedot tarkasteltaviksi.

5.3 Videolähteiden valinta








- ① ⚠ 5.3.1 Mittauksen videolähteeksi voi valita työasemalla olevan videotiedoston.
- ① ⚠ 5.3.2 Mittauksen videolähteeksi voi valita työasemaan yhteydessä olevan videokameran.
- ① 🚩 5.3.3 Mittaukseen liittyviä videolähteitä voi valita useamman kuin yhden.
- ② ⚠ 5.3.4 Käyttäjä voi valita videolähteen viimeksi käytettyjen videolähteiden listalta.

5.4 Mittauksen tietojen määrittäminen


- ① ✘ 5.4.1 Sovellus asettaa automaattisesti mittauksen ajankohdan (päivämäärän ja kellonajan) metatiedostoon.
- ② ✘ 5.4.2 Sovellus asettaa automaattisesti mittauksen tekijän nimen työaseman käyttäjänimen perusteella.

- ①  5.4.3 Käyttäjä voi muokata mittauksen tekijän nimeä.
- ④  5.4.4 Käyttäjä voi muokata mittauksen ajankohtaa.
- ①  5.4.5 Käyttäjä voi syöttää mittauksen nimen.
- ①  5.4.6 Käyttäjä voi syöttää mitattavan ryhmän nimen.
- ①  5.4.7 Käyttäjä voi syöttää mittauksen paikan.
- ①  5.4.8 Käyttäjä voi syöttää mittaukseen liittyviä lisätietoja.
- ①  5.4.9 Käyttäjä voi määrittää mittauksen tulosten tallennuspaikan.
- ③  5.4.10 Käyttäjä voi määrittää ne aikavälit päivästä, jolloin mittaus on käynnissä.
- ③  5.4.11 Käyttäjä voi määrittää ne viikonpäivät, joina mittaus on käynnissä.
- ①  5.4.12 Valitut videolähteet ja syötetyt tiedot tallennetaan metatiedostoon.

5.5 Kalibrointi

- ①  5.5.1 Kameran katsomissuunnan ja korkeuden voi määrittää sovittamalla kaksiulotteisen ruudukon videokuvan päälle.
- ①  5.5.2 Kameran tynnyrivääristymän voi korjata. (Tarkennetaan myöhemmin, miten.)
- ①  5.5.3 Näkökentän leveyden voi määrittää. (Tarkennetaan myöhemmin, miten.)
- ①  5.5.4 Useamman kameran suhteellisen sijainnin voi määrittää asettamalla videokuvien päälle yhteisiä pisteitä.
- ②  5.5.5 Sovellus asettaa automaattisesti videolähteen nimen perusteella videolähteen kalibrointitiedot, jos ne on aiemmin tallennettu.
- ②  5.5.6 Käyttäjä voi halutessaan valita videolähteen kalibrointitiedot aiemmin tallennetuista tiedoista.
- ②  5.5.7 Sovellus tallentaa videolähteen kalibrointitiedot automaattisesti asetustiedostoon.

5.6 Mittauksen aloitus

- ①  5.6.1 Käyttäjä voi käynnistää mittauksen.

- ② ✘ 5.6.2 Käyttäjä voi valita, tallennetaanko aktiivisuusdataa levyille.
- ② ✘ 5.6.3 Videokameran ollessa videolähteenä käyttäjä voi valita, tallennetaanko videokuva levyille.
- ② ✘ 5.6.4 Videokameran ollessa videolähteenä käyttäjä voi asettaa mittauksen enimmäispituuden.
- ① ✘ 5.6.5 Videotiedoston ollessa videolähteenä käyttäjä voi valita tarkasteltavan aikavälin.
- ③ ✘ 5.6.6 Useamman videotiedoston ollessa lähteenä käyttäjä voi synkronoida ne ajan suhteen etsimällä videoista yhteinen ajanhetki.

5.7 Mittauksen seuranta

- ① ✘ 5.7.1 Käyttäjä voi peruuttaa mittauksen, jolloin videokuvaa ja analyysituloksia ei tallenneta.
- ① ⚠ 5.7.2 Kustakin videolähteestä tuleva kuva näytetään mittauksen aikana samassa ikkunassa.
- ① ✘ 5.7.3 Käyttäjä voi päättää mittauksen, jolloin kaikki mittaukseen liittyvät tulokset tallennetaan.
- ③ ✘ 5.7.4 Käyttäjä voi pysäyttää mittauksen ja jatkaa sitä myöhemmin.
- ③ ✘ 5.7.5 Sovellus pysäyttää käyttäjän halutessa videon tallennuksen automaattisesti, jos kamerakuvasta ei havaita aktiivisuutta käyttäjän määrittämän ajan kuluessa.
- ② ✘ 5.7.6 Liikemääräkäyrästä voi valita aiemman aikavälin tarkasteltavaksi mittauksen aikana.
- ① ⚠ 5.7.7 Käyttäjä voi määrittää, minkä pituinen aikaväli näytetään nykyhetkestä taaksepäin.
- ① 🚩 5.7.8 Videokameran ollessa videolähteenä videokuva voidaan tallentaa levyille.
- ③ ✘ 5.7.9 Videokameran ollessa videolähteenä videokuvan voi tallentaa suoraan tilatallennusjärjestelmään.
- ② ✘ 5.7.10 Jos levytila on loppumassa, sovellus ilmoittaa tästä ja pysäyttää mittauksen.

- ② ✗ 5.7.11 Sovellus pilkkoo tallennettavan videon automaattisesti useammaksi tiedostoksi käyttäjän määrittelemän pituisiksi osiksi.

5.8 Tulosten näyttäminen ja käsittely

- ① ⚠ 5.8.1 Videokuvan päällä voidaan käyttäjän halutessa näyttää havaitut liikkuvat kohteet laatikoituina.
- ① ⚠ 5.8.2 Liikemääräkäyrä voidaan näyttää käyttäjän määrittämältä aikaväliltä.
- ② ✗ 5.8.3 Liikemääräkäyrään voi nimetä ajanhetkiä.
- ② ✗ 5.8.4 Liikemääräkäyrään nimetyt ajanhetket tallennetaan metatiedostoon.
- ① ⚠ 5.8.5 Liikemääräkäyrän datan voi tallentaa levyille.
- ② ⚠ 5.8.6 Liikemääräkäyrän voi tallentaa levyille kuvatiedostona.
- ② ✗ 5.8.7 Havaituista kohteista voidaan näyttää 2D-lämpökuva käyttäjän valitsemalla aikavälillä.
- ② ✗ 5.8.8 Käyttäjä voi määrittää, mistä kamerasta katsottuna lämpökuva näytetään.
- ③ ✗ 5.8.9 Lämpökuva voidaan näyttää todellisesta maailmasta ylhäältä päin katsottuna.
- ③ 🚩 5.8.10 Liikkuvien hahmojen lukumäärää kuvaava käyrä voidaan näyttää liikemääräkäyrän kanssa samassa graafissa.
- ④ ✗ 5.8.11 Videokuvasta voi valita ne kohteet, joita ei oteta huomioon aktiivisuustasoa laskettaessa.

5.9 Konenäkö






- ④ ✗ 5.9.1 Sovellus osaa yksilöidä useamman kameran kuvaamat todellisen ympäristön havainnot yksittäisiksi havainnoiksi.

5.10 WWW-käyttöliittymä



- ③ ✗ 5.10.1 WWW-käyttöliittymässä näytetään meneillään olevan mittauksen liikemääräkäyrä.
- ③ ✗ 5.10.2 WWW-käyttöliittymässä näytetään mittauksen metatiedot.

6 Tekniset vaatimukset ja rajoitteet



6.1 Käyttöjärjestelmät

- ①  6.1.1 Sovellus toimii Windows 7 -käyttöjärjestelmässä.
- ③  6.1.2 Sovellus toimii Windows 8 -käyttöjärjestelmässä.
- ③  6.1.3 Sovellus toimii Windows Vista -käyttöjärjestelmässä.
- ⑤  6.1.4 Sovellus toimii Windows XP -käyttöjärjestelmässä.
- ①  6.1.5 Sovellus toimii Linux-pohjaisissa käyttöjärjestelmissä.
- ③  6.1.6 Sovellus toimii OS X -käyttöjärjestelmässä.






6.2 Videon tallennusformaatit

- ①  6.2.1 Sovellus tukee MPEG-4-videomuotoa tallennuksessa.
- ⑤  6.2.2 Sovellus tukee MJPEG-videomuotoa tallennuksessa.

6.3 Videon lukuformaatit

- ①  6.3.1 Sovellus voi lukea MJPEG-muodossa olevia videotiedostoja.
- ①  6.3.2 Sovellus voi lukea MPEG-4-muodossa olevia videotiedostoja.

6.4 Liikemääräkäyrän tallennusformaatit

- ①  6.4.1 Liikemääräkäyrän voi tallentaa PNG-kuvana.
- ④  6.4.2 Liikemääräkäyrän voi tallentaa SVG-kuvana.
- ⑤  6.4.3 Liikemääräkäyrän voi tallentaa JPG-kuvana.
- ①  6.4.4 Liikemääräkäyrän datan voi tallentaa CSV-formaatissa.
- ②  6.4.5 Liikemääräkäyrän datan voi tallentaa raakaformaattissa.

6.5 Metatiedon tallennusformaatit

- ①  6.5.1 Mittauksen metatiedot tallennetaan XML-formaatissa.

6.6 Rajoitteet

- 6.6.1 Videokameran lukemista varten mahdolliset siihen liittyvät ajurit on oltava asennettuina työasemalla.
- 6.6.2 Videokameran lukeminen ei ole mahdollista, jos OpenCV-kirjasto ei tue kameran rajapintaa.
- 6.6.3 Videoita voi tallentaa työasemalle korkeintaan niin paljon kuin siinä on levytilaa.
- 6.6.4 Sovelluksessa voi esiintyä hitautta riippuen työaseman tehoista, videolähteiden määrästä ja niiden kuvan tarkkuudesta.
- 6.6.5 Videokameran tyyppi määrää videokuvan enimmäistarkkuuden.
- 6.6.6 Sovellus toimii Linux-pohjaisista käyttöjärjestelmistä vain Fedoralla ja Ubuntulla (versio 14.04).
- 6.6.7 Sovellus voi hidastua, jos käytettävä FPS asetetaan liian suureksi.
- 6.6.8 Mittaustulokset voivat olla epätarkkoja, jos käytettävä FPS on liian pieni tai liian suuri.

7 Yhteenveto

Liikkuva-sovellusprojekti suunnittelee ja toteuttaa Jyväskylän yliopiston liikunta-kasvatuksen laitokselle käyttöliittymän konenäköpohjaiselle liikemittarille. Toteuttavalla sovelluksella voidaan visualisoida videokuvasta mitattua ihmisjoukon aktiivisuustasoa sekä suorittaa mittauksessa tarvittava kalibrointi.

8 Lähteet

- [1] Kivelä Joel, Koskenkorva Erkki, Lehtinen Mika, Leppäaho Oskari ja Partanen Petri, "Liikkuva-projekti, Projektisuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, 2014.