

Janne Suurkoivu

Reaaliaikaisen 3D-animaation toteuttaminen Cave-ympäristössä

Opinnäytetyö

Kevät 2011

Tekniikan yksikkö

Tietojenkäsittely

Digitaalinen mediatuotanto



SEINÄJOEN AMMATTIKORKEAKOULU

Opinnäytetyön tiivistelmä

Koulutusyksikkö: Tekniikka

Koulutusohjelma: Tradenomi

Suuntautumisvaihtoehto: Digitaalinen mediatuotanto

Tekijä: Janne Suurkoivu

Työn nimi: Reaaliaikaisen 3D-animaation toteuttaminen Cave-ympäristössä

Ohjaaja: Kimmo Salmenjoki

Vuosi: 2010

Sivumäärä: 43

Liitteiden lukumäärä: 5

Tämän opinnäytetyön tarkoituksena oli tutkia miten VR-laboratorion Cave-järjestelmään voidaan rakentaa reaaliaikainen virtuaalihahmon liikkeentunnistus.

Virtuaalihahmon osalta työ keskittyi valmiin virtuaalihahmon luurangon ja geometrian yhteen liittämiseen. Miten saadaan luuranko toimimaan hahmon geometrian kanssa ja miten valmis virtuaalihahmo viedään liikkeentunnistussovellukseen. Virtuaalihahmon rakennussovelluksena käytetään 3Ds max –ohjelmistoa.

Paikannuslaitteen osalta työ keskittyy siihen kuinka paikannussensoreita lisätään olemassa olevaan järjestelmään.

Lopuksi tutkittiin miten virtuaalihahmo ja paikannussensorit liitetään yhteen Virtools-ohjelmassa.

Tutkimuksessa ilmeni, että helpoin osuus oli Virtuaalihahmon rakennus. Virtuaalihahmon luiden rakentamisessa piti ottaa huomioon liikkeentunnistussovelluksena toimiva Virtools-ohjelmisto ja sen asetukset.

Paikannussensorien lisäämisessä oli ongelmia vanhojen laitteiden ja Virtools-ohjelmiston teknisen tuen puutteen takia. Myös ympäristön magneettisuus aiheutti häiriöitä herkille laitteille.

Sensorien mittaustiedon lisääminen Virtools-ohjelmistoon oli ongelmaton ja virtuaalihahmon lisääminen samaan kokoonpanotiedostoon oli suhteellisen helppoa. Virtuaalihahmon rakenteena käytettiin Virtools-ohjelmiston Building Block -rakennuspalikoita.

Avainsanat: 3Ds Max, virtuaalihahmo, Virtools, animaatio, liikkeentunnistus.

SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

Thesis abstract

Faculty: School of Technology
Degree programme: Business Information Technology
Specialisation: Digital Media Production

Author: Janne Suurkoivu

Title of the thesis: Real time 3D animation in cave environment

Supervisor: Kimmo Salmenjoki

Year: 2010

Number of pages:38

Number of appendices:4

The purpose of this final year project was a study how virtual character's motion capture system can be built in a CAVE environment of the VR laboratory.

For the virtual character project focused on how to attach the bone system and the geometry of the finalized virtual character together and how finalised virtual character may be exported to the Virtools composition. In this project 3ds max was used in finalising the virtual character.

For the motion capture sensors the project focused on how motion sensors may be added to the motion capture system.

Finally it was studied how to add together the virtual character and motion sensors in the Virtools composition.

The easiest part of the project was to build a virtual character. When building a virtual character bone system Virtools as a motion detection software and its settings had to be considered.

When adding the motion sensors there were problems because of old devices and lack of Virtools software support. Also environmental magneticity caused problems to the delicate devices.

Adding the motion sensor data to the Virtools composition was not a problem and adding the virtual character in the same composition was relatively easy Virtools Building Blocks were used as the virtual character structure.

Keywords: 3Ds Max, Virtual puppet, Virtools, Animation, motion capture.

SISÄLTÖ

OPINNÄYTETYÖN TIIVISTELMÄ	2
THESIS ABSTRACT	3
SISÄLTÖ	4
KÄYTETYT TERMIT JA LYHENTEET	6
KUVIO- JA TAULUKKOLUETTELO	9
1 JOHDANTO	15
1.1 Työn tausta	15
1.2 Työn tavoite	15
1.3 Työn rakenne	15
1.4 SeAMK VR-laboratorio	16
1.5 Virtuaalihahmon liikkeentunnistuksen tutkimusongelmat	16
2 VIRTUAALIIHAHMO	18
2.1 Virtuaalihahmon kehitystyökalu	18
2.1.1 Hahmon geometrian rakennus	19
2.1.2 Mesh-verkko mallinnustekniikka	19
2.1.3 Subdivision Surfaces mallinnustekniikka	20
2.1.4 Virtuaalihahmon luuranko	21
2.1.5 Bone-luuranko	22
2.1.6 Luurangon kinematiikka	23
2.1.7 Biped-luuranko	24
2.1.8 Virtuaalihahmon ihon luonti	25
2.1.9 Virtuaalihahmon siirtäminen Virtools-ohjelmistoon	26
2.2 Virtuaalihahmon kokoaminen	27
3 LIIKKEENTUNNISTUS JA LIIKKEENTUNNISTUSSOVELLUS	31
3.1 Magneettinen tunnistus	31
3.2 VR-laboratorion liikkeentunnistusjärjestelmä	32
3.2.1 Magneettisen paikannussensorin lisääminen	33

3.3 Paikannussensorien lisääminen Cave-järjestelmään	33
3.4 Liikkeentunnistussovellus.....	34
3.5 Liikkeentunnistussovelluksen rakennustyökalu	34
3.5.1 IkPosition rakennuspalikka.....	35
3.6 Liikkeentunnistussovelluksen rakentaminen	36
3.6.1 Virtuaalihahmon lisääminen	36
4 JOHTOPÄÄTÖKSET	38
LÄHTEET	41

Käytetyt termit ja lyhenteet

3D	Kolmiulotteisuus.
3ds Max export	Mallin vientityökalu.
3ds Max Scene	3ds Maxin virtuaalinen malli(nnustila).(3D-Malli / malli-hierarkia/ ”näyttämö”/ Virtuaalimalli).
3ds Max	Kolmiulotteiseen mallintamiseen tarkoitettu ohjelma.
Biped-luuranko	Ihmisluurankoa muistuttava objektiketju.
Bone-työkalu	3Ds Maxin työkalu, jolla voidaan rakentaa luurangon yksittäisistä kappaleista.
Building block	Virtoolsin rakennuspalikka (toiminto, funktio).
Cave	Virtuaalitila, jossa käyttäjän ympärille heijastetaan 3D-malli.
Character studio plugin	3Ds Max -ohjelmiston lisäosa.
Childbone	luuketjuhierarkiaan kuuluva luu (hierarkiassa lapsi vanhempaansa nähden)
Config-tiedosto	tiedosto, johon määritellään eli konfiguroidaan laite- tai ohjelmistoasetuksia
Edge	3D-geometriassa polygonin äärireuna (”särmä”).

Edit mesh modifier Mesh verkko muunnin

Export as A Character Vienti 3ds Maxista hahmona Virtools-ohjelmaan.

Extended Range transmitter

Face 3D-geometriassa kolmioverkon (mesh) pinalappu.

Flock of Birds Ascension Technologyn magneettisen paikannuslaitteiston tuotemerkki.

Force Skin to Morph Pakottaa geometrian ihoksi.

Forward Kinematics suora kinematiikka.

Geometria 3D-mallin geometrinen rakenne.

Ik Position bb Virtools-ohjelman rakennuspalikka,

Inverse Kinematics käänteinen kinematiikka.

Kinematics kinematiikka.

Laajan kantomatkan lähetin

Mesh-verkko mallinnus 3D mallin rakennustapa.

Offline - sovellus Ei reaaliaikainen järjestelmä

Online - sovellus Reaaliaikainen järjestelmä.

Polygon 3D-geometriassa monikulmion pinta

Primitiiviobjektit	Valmiita yksinkertaisia 3D malleja.
Rootbone	luuketjuhierarkiaan vaikuttava luu.
Subdivision surface mallinnus	3D-mallin rakennustapa.
TrackerPc	Cavessa käytössä oleva Pc-laitteisto, johon on asennettu erilaisia paikannusohjelmistoja.
Transmitter unit	Magneettisen paikannuslaitteen lähetyksikkö.
Wand	Cavessa käytössä oleva ohjainsauva.
Vertex	3D-geometriassa verkon kärkipiste
Virtools	Dassault Systèmesin valmistama Virtuaalitodellisuus esitykseen käytettävä visualisointi-ohjelmisto.
VRPack	Virtools ohjelman virtuaalitodellisuuspaketti.
Vrpn	Ilmainen virtuaalitodellisuustyökalu (Virtual Reality Peripheral Network)

Kuvio- ja taulukkoluetelo

Kuva 1. 3Ds Max.....	18
Kuva 2. Mesh-verkko- ja Subdivision surface -mallinnustekniikka.	20
Kuva 3. Biped- ja bone-luuranko. (Animation importer 2010)	21
Kuva 4. Luuketjun hierarkia.	22
Kuva 5. Suora kinematiikka.	23
Kuva 6. Käänteinen kinematiikka.....	24
Kuva 7. Vasemman käden vaikutusalue skin-muokkaimella.....	25
Kuva 8. Physique-muokkain.....	26
Kuva 9. Virtools exporter -työkalu.	27
Kuva 10. Valmis virtuaalihahmo.....	29
Kuva 11. Liikkeentunnistus. (Täydellisen animaation jäljillä 2007.).....	31
Kuva 12. Magneettisen järjestelmän sensoreita (Täydellisen animaation jäljillä 2007.).....	32
Kuva 13. Virtools-ohjelma.	35
Kuva 14. IkPosition building block.....	35

Kuva 15. Biped Structure.	47
Kuva 16. 3Ds Max -Align työkalu.	48
Kuva 17. Virtuaalihahmo + biped luuranko.	50
Kuva 18. Virtuaalihahmon pään envelopit.....	51
Kuva 19. Virtuaalihahmon kaulan Envelopit.....	52
Kuva 20. Virtuaalihahmon rintakehän envelopit.	52
Kuva 21. Virtuaalihahmon selkärangan envelopit.	53
Kuva 22. Virtuaalihahmon selkärangan envelopit.	53
Kuva 23. Selkärangan envelopia.	53
Kuva 24. Lantion envelopit.....	54
Kuva 25. Vasemman jalan envelopet.....	54
Kuva 26. Vasemman säären envelopit.	55
Kuva 27. Vasemman jalkapöydän envelopit.	55
Kuva 28. Vasemman käden envelopit.....	56
Kuva 29. Virtools exporter -ikkuna.	58
Kuva 30.Seitsemän sensorilaitetta ja PaikannusPc.	62

Kuva 31. Extended Range Controller.....	63
Kuva 32. VRPN Server_6_Sensors.	66
Kuva 33. Muutettu VRPN Server_7_sensors.cfg	68
Kuva 34. VRPack.cfg tiedoston muuttaminen.....	70
Kuva 35. VRBaseSample.cfg tiedosto.	73
Kuva 36. VRBaseSample.cfg tiedosto.	73
Kuva 37. VRNR_Navigation_HeadAndWand.cfg.....	74
Kuva 38. VRNR_Tracking_VRNR.cfg.....	76
Kuva 39. VRNR_tracking_vrnr.cfg - Muutettu käyttämään seitsemän sensoria.	76
Kuva 40. Koordinaatisto arvot jotka toimi.....	78
Kuva 41. Virtools:n 3D Frames -valikko.....	85
Kuva 42. Virtoolsin 3D Objects -valikko.....	87
Kuva 43. VRNR_Global_Header_VRNR Variables skriptiin	89
Kuva 44. Tracking skriptin muokkaaminen.	91
Kuva 45. String Behavior Graph -ketju.....	92
Kuva 46. Avoinna oleva string BG.	93

Kuva 47. Distribution Skripti.....	93
Kuva 48. VRNR_Tracking_VRPN_manager muokkaaminen.....	94
Kuva 49. VRNR_Tracking_VRPN_Master Muokkaaminen.....	96
Kuva 50. SAS Vrpn modif muokkaaminen.	98
Kuva 51. VRNR_Navigation_HeadAndWand Skripti.....	99
Kuva 52. Remind Starting position Skripti.....	101
Kuva 53. VRNR_Navigation_HeadAndWand_Master_Transform skripti.	103
Kuva 54. Koko virtuaalihahmon liikuttaminen.	114
Kuva 55. Virtuaalihahmon pään koodaaminen.....	114
Kuva 56. Virtuaalihahmon raajan koodaaminen.....	115
Kuva 57. Selkärangan koodaaminen.	115
Taulukko 1. 3Ds Max virtuaalihahmon tuonti ja ohjelman esivalmistelu.....	46
Taulukko 2. Luurangon luonti.....	47
Taulukko 3. Luurangon vienti Hahmon sisälle.	49
Taulukko 4. Virtuaalihahmon ihon liittäminen.....	50

Taulukko 5. Sensorin asennus.....	66
Taulukko 6. Sensorin lisääminen paikannusPc:lle	68
Taulukko 7. Sensorin lisääminen työkoneelle.	69
Taulukko 8. VrPack.cfg tiedoston muokkaus.	71
Taulukko 9. VRBaseSample.cfg muokkaaminen.	73
Taulukko 10. VRNR_navigation_HeadAndWand.cfg tiedoston muokkaus.	75
Taulukko 11. VRNR_Tracking_VRNR.cfg muuttaminen.	75
Taulukko 12. Sensorin koordinaatisto arvojen muuttaminen.....	77
Taulukko 13. 3D Framen luonti	86
Taulukko 14. 3D objektin luonti ja hierarkiaan siirtäminen.	88
Taulukko 15. Parametrien position ja quaternion luonti tracker BG:hen.	89
Taulukko 16. Master Only Values Behavior BG:n lisääminen.	90
Taulukko 17. Parametrien luonti Navigation Enties BG:hen.	90
Taulukko 18. Tracking skriptin koodaaminen.	91
Taulukko 19. String BG:n muokkaaminen.....	92
Taulukko 20. Vr distrib koodaaminen.....	94

Taulukko 21. VRNR_Tracking_VRPN_Manager Skriptin koodaaminen.	96
Taulukko 22. VRNR_Tracking_VRPN_Master koodaaminen.	97
Taulukko 23. SAS Vrpn modif muokkaaminen.....	99
Taulukko 24. VrNR_Navigation_HeadAndWanD skriptin koodaaminen.	100
Taulukko 25.Remind Starting position koodaaminen.	102
Taulukko 26 VRNR_Navigation_HeadAndWand_Master_Transform koodaaminen.	104
Taulukko 27. Virtuaalihahmon tuonti Virtools-kokoonpanotiedostoon.....	109
Taulukko 28. kokoonpanotiedoston viimeistely.	110
Taulukko 29. Virtuaalihahmon liikkeen koodaaminen.	113

1 JOHDANTO

1.1 Työn tausta

Vr-laboratoriossa oli tarve tutkia tarkemmin virtuaalihahmon liikuttamista käyttäen Inverse Kinematic -tekniikkaa ja paikannussensoreita Cave-järjestelmässä. Virtuaalihahmon liikkeentunnistusta ei ole aiemmin käytetty ja sen ominaisuuksiin haluttiin tutustua lähemmin. Aihe kiinnosti opinnäytetyöntekijää, joten aihetta alettiin tutkia.

1.2 Työn tavoite

Tämä työn tavoitteena on tutkia virtuaalihahmoa ja selvittää minkälaisia asetuksia virtuaalihahmo tarvitsee, että se toimii liikkeentunnistuksessa. Työssä selvitetään kuinka rakennetaan valmiille hahmolle luuranko ja iho 3Ds Max -ohjelmistolla. Tästä käy ilmi kuinka paikannuslaitteita lisätään olemassa olevaan järjestelmään ja kuinka paikannuslaitteisto ja hahmo liitetään yhteen Virtools-kokoonpanotiedostoon. Opinnäytetyöntekijän kannalta tavoitteena oli oppia liikkeentunnistustekniikoista ja virtuaalihahmon animoimisesta mahdollisimman paljon uutta.

1.3 Työn rakenne

Toisessa luvussa käsitellään tarkemmin 3Ds Max ohjelmistoa, jolla tehtiin virtuaalihahmo. Kolmannessa luvussa käsitellään liikkeentunnistusta. Minkälaisia liikkeentunnistustapoja on olemassa ja minkälainen järjestelmä on käytössä. Neljännessä luvussa käsitellään Virtools-ohjelmaa, jolla tehdään reaaliaikainen liikkeentunnistussovellus. Siihen liitetään paikannusjärjestelmä ja hahmo. Viidennessä luvussa käsitellään työstä saatuja erilaisia johtopäätöksiä.

1.4 SeAMK VR-laboratorio

Virtuaalitodellisuuslaboratorio tarjoaa visualisointipalveluita erilaisille yrityksille. Virtuaalilaboratoriossa sijaitsee myös Cave-tila, sitä käytetään erilaisten tietokoneella tehtyjen 3D-mallien visualisoimiseen. Cave on viisiseinäinen keinotodellisuus-tila, jossa voi seikkailla. Cave-tilan Keinotodellisuus luodaan harhauttamalla näkö - ja kuuloaisteja 3d-stereografiikalla ja tilääniijärjestelmällä. (Seamk VR-laboratorio 2010.)

Cavessa tietokonejärjestelmänä käytetään kahta laitteistoa: Silicon Graphics (SGI) Onyx 4 UltimateVision grafiikkatietokone sekä Fujitsu-Siemens Celsius R630 PC-visualisointiklusteri. Paikannuslaitteistona on Ascension Technology Corporationin magneettinen Flock of Birds. Kuvan tuottamiseen käytetään Christie Digital Mirage 4000 DLP-projektoreita. (Seamk VR-laboratorio 2010.)

Cavessa käytetään myös seuraavia ohjelmistoja:VRCO VRscape (IRIX/Windows/Linux) TreeC VR4MAX (Windows), ja Virtools (Windows) (Seamk VRlaboratorio 2010).

1.5 Virtuaalihahmon liikkeentunnistuksen tutkimusongelmat

Tarkoituksena on tutkia kuinka 3Ds Maxissa rakennettu virtuaalihahmo toimii Virtools-ohjelmistossa. Virtuaalihahmoa ei tarvitse rakentaa itse, vaan se voidaan etsiä internetistä. Virtuaalihahmolle luodaan luuranko ja kuinka virtuaalihahmon geometria muutetaan ihoksi.

Paikannuslaitteistoon asennetaan lisää paikannuslaitteita ja tutkitaan miten seitsemän paikannuslaitetta toimii yhdessä.

Virtools on ohjelmisto, jossa paikannuslaitteisto ja virtuaalihahmo liitetään yhteen. Tutkitaan miten virtuaalihahmo ja paikannuslaitteisto toimivat yhdessä, kun liikutaan paikannuslaitteiston sensoreita.

Haastetta työssä on totisesti paljon, mutta opinnäytetyöntekijää auttoi ainakin 3Ds Max -ohjelmiston tunteminen sekä animoitavan hahmon rakentamisen kokemus. Virtools-ohjelmistosta ei ollut kokemusta.

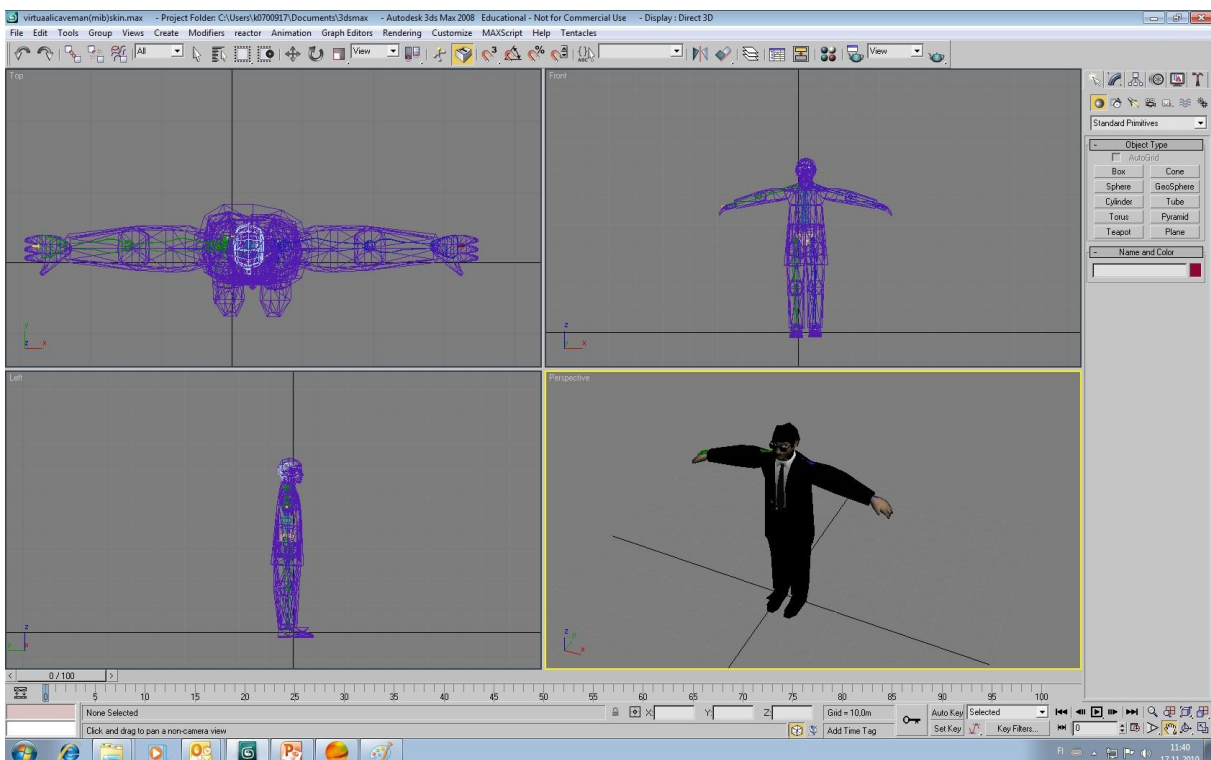
2 VIRTUAALIIHAHMO

Ideana oli käyttää valmista 3Ds Maxissa valmistettua hahmoa, jolle luotaisiin joko bone- tai biped-luuranko ja tämän jälkeen liittää hahmon geometria luihin sopivalla iho-muokkaimella.

Tehtävänä tässä oli selvittää myös minkälaisia asetuksia. 3Ds Maxin Virtools Exporter työkalu tarvitaan virtuaalihahmon siirrossa Virtools-ohjelmistoon.

2.1 Virtuaalihahmon kehitystyökalu

Virtuaalihahmon kehitystyökaluna ohjeistettiin käyttämään Autodesk 3Ds Max 2008 -ohjelmistoa. Työssä käytettiin tätä ohjelmistoversiota Virtools exporter työkalun takia. Virtools exporter vientityökalua ei ole virallisesti muille 3Ds Max versioille.



Kuva 1. 3Ds Max

Autodesk 3Ds Max -ohjelmisto on kolmiulotteisten tietokonemallien rakennusohjelma. Ohjelmisto tarjoaa erilaisia työkaluja kolmiulotteisten mallien ja niiden liikkeen rakentamiseen. Sitä käytetään viihdetuotannossa, teollisuuden ja arkkitehtuurin suunnittelussa. (Autodesk 2010.)

3Ds Maxin työkaluilla pystyy simuloimaan erilaisia esineitä, fysiikkaa, ihmisluurangan rakennetta, vettä, tulta ja kaikkea mitä mielikuvitus keksiikin. (Autodesk 2010).

2.1.1 Hahmon geometrian rakennus

Mallin rakentaminen aloitetaan yleensä 3Ds Maxin mukana tulevilla valmiilla objekteilla. Näitä objekteja kutsutaan primitiiviojekteiksi. Objekteihin kuuluu mm. laatikko, pallo, sylinteri, kartio ja pyramidi jne. Tämän jälkeen primitiiviobjektia muokataan käyttäen erilaisia mallinnustekniikoita muokkaimia (modifier). (Fox 2000,113.)

Primitiiviobjektia muutetaan tarpeen mukaan minkälainen käyttötarkoitus sillä on ja miten tarkan mallista haluaa. 3Ds maxissa on muutama erilainen tapa muokata primitiiviojekteja. Alla niistä kaksi perustyyppiä.

2.1.2 Mesh-verkko mallinnustekniikka

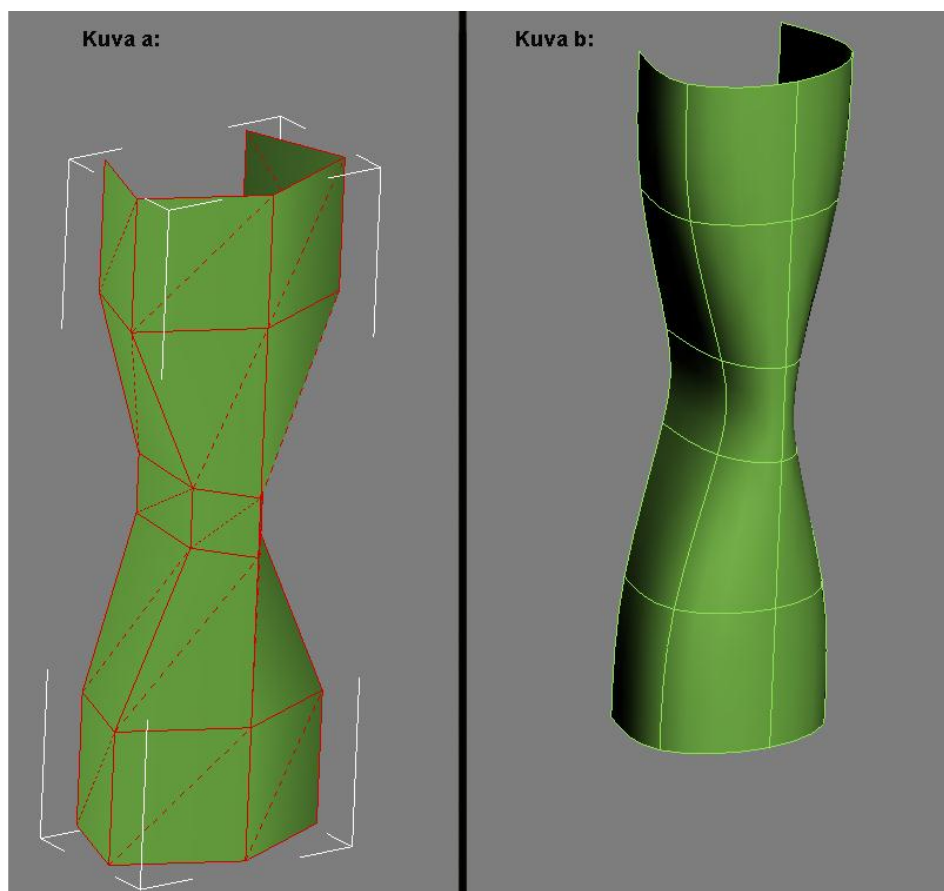
Mesh-verkko malli koostuu pinnoista (face), jotka ovat kolmikulmaisia kappaleita. Mesh-verkossa, pinnassa on kolme kärkipistettä (vertex) ja sivua (edge) (Fox 2000,113).

Mesh-verkon sivut ovat aina suorassa linjassa ja monimutkainen pinta vaatii tiheämmän verkon. Tämä yksinkertaisuus tekee mallin käytöstä helpompaa reaaliaikaisissa videopelimoottoreissa. (Fox 2000,113.)

Mesh-Verkkotekniikan (kuva 2 a) saa käyttöön 3Ds Maxissa Edit mesh -muokkaimella. Tällä työkalulla voidaan muokata mesh-verkon pintaa (polygon), sen kärkipisteitä (vertex) ja kärkipisteiden välissä olevia sivuja (edge). (Fox 2000,113.)

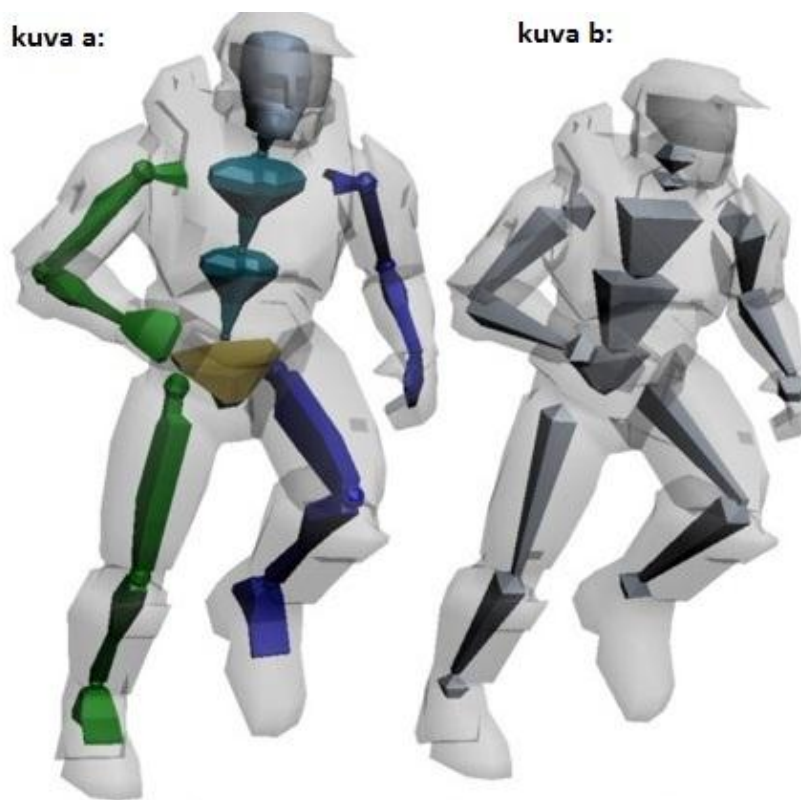
2.1.3 Subdivision Surfaces mallinnustekniikka

Subdivision surface (kuva 2 b) on monikulmiopintaverkko. Se koostuu pinnasta, kärkipisteistä ja sivuista, kuten mesh-verkko. Toistuva prosessi pehmentään verkkoa sitä mukaan, miten tiheäksi se kasvaa. 3Ds Maxissa tämä tunnetaan Edit Poly -mallinnuksena. Edit Polygon -muokkaimen työkalut mahdollistavat nopean työskentelyn. (Holmes [viitattu 10.11.2010].)



Kuva 2. Mesh-verkko- ja Subdivision surface -mallinnustekniikka.

2.1.4 Virtuaalihahmon luuranko



Kuva 3. Biped- ja bone-luuranko. (Animation importer 2010)

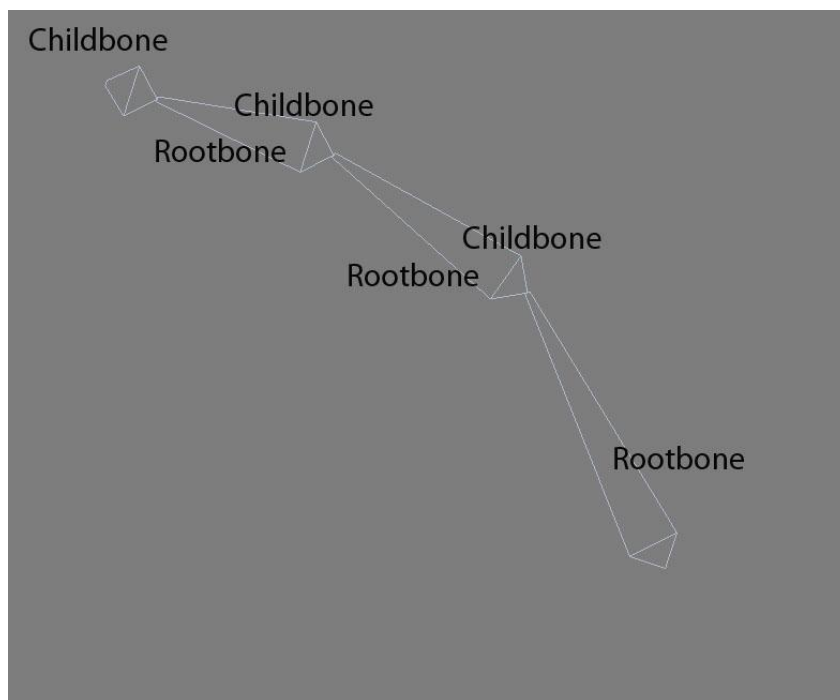
Ihmismallin animaation luontiin tarvitaan luuranko. Ilman luurankoa on vaikea tehdä virtuaalihahmolle minkäänlaista liikettä. 3Ds Max -ohjelmistolla pystyy mallintamaan hahmolle kaksi erityyppistä luurankoa. (Maestri, 2008,119.)

2.1.5 Bone-luuranko

Bone-luuranko luodaan yksittäisistä luunkappaleista ja siihen muodostuu (kuva 3 b) rakenne, joka on hierarkkisesti kiinni toisissaan. Luuranko voidaan liittää mihin tahansa geometriaan ihomuokkaimen avulla, jolloin sitä voidaan käyttää virtuaalihahmossa tai missä tahansa missä tarvitaan luurankoa. (Maestri, 2008,120.)

Bone-luurangon hierarkia toimii mallinnusjärjestyksen mukaan. Esimerkiksi luurangossa on neljä luuta (Kuva 4). Ensimmäinen valittu luu on Äiti-luu (Root bone) toinen, kolmas ja neljäs ovat lapsi-luita (child bone). Esimerkki ketjun toinen ja kolmas luu ovat valittuna myös Äiti-luita (Root bone), jolloin hierarkiassa seuraavana tulevat luut ovat lapsiluita. (child bone).

Bone-luuranko ei ole valmis suoraan animoitavaksi, siihen täytyy ennen animointia määrittää kinematiikka-asetukset. Kinematiikka valitaan luun tarkoituksen mukaan, joko suora tai käännteinen kinematiikka. (Maestri, 2008,122).



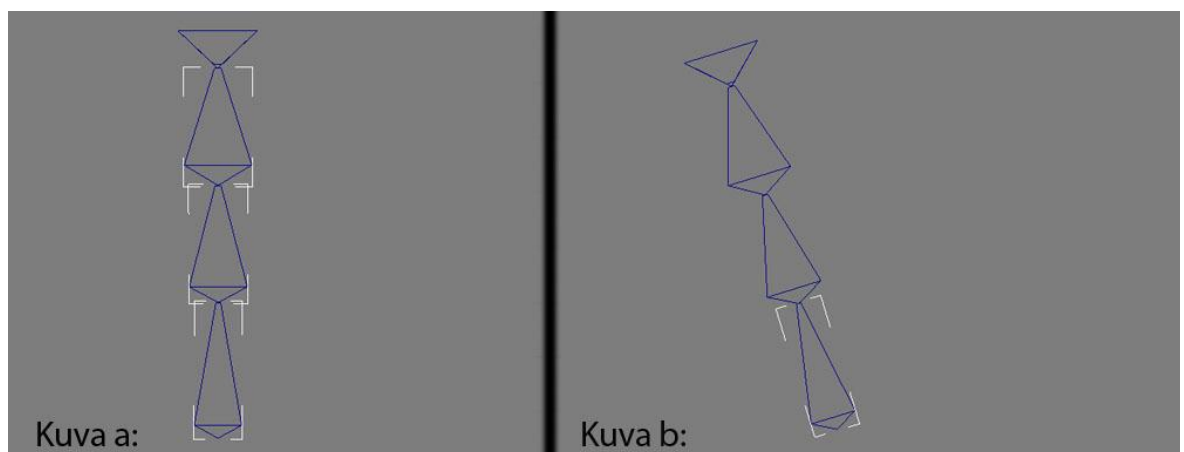
Kuva 4. Luuketjun hierarkia.

2.1.6 Luurangon kinematiikka

Virtuaalihahmon animointi vaatii myös kinematiikkajärjestelmän luonnin, mikä mahdollistaa kolmiulotteisen luurangon liikuttelun animaattorin haluamalla tavalla. Erilaisia kinematiikkoja voidaan käyttää luurangossa kahta erilaista. (Halme 2006, 25.)

Suora kinematiikka (forward kinematics). Suora kinematiikka toimii suoraa luuta kääntämällä, jolloin valittu Äitiluu (Root bone) kääntää luuketjun Lapsiluuta (child bone). (Maestri, 2008,122.)

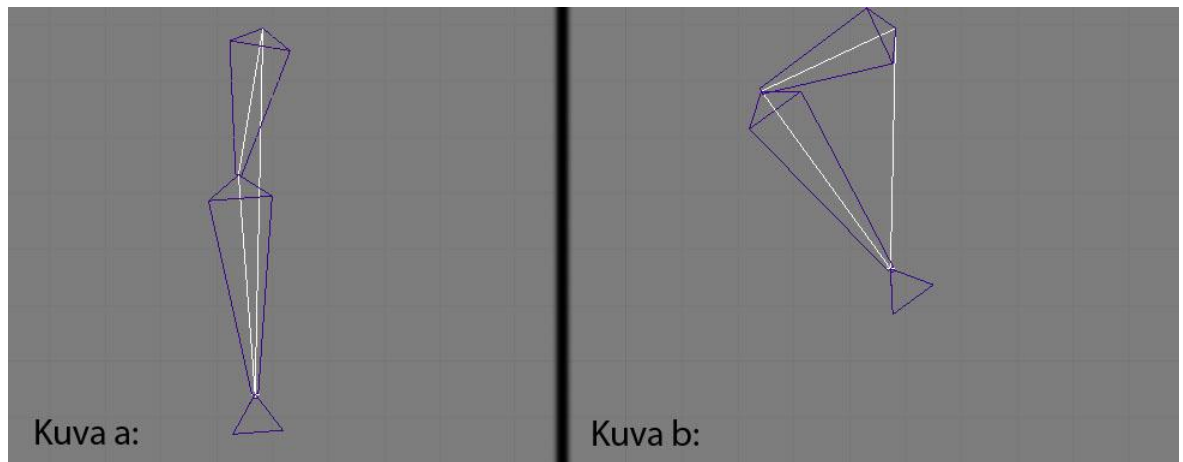
(Kuvassa 5 a) valitaan ensimmäinen luu ja käännetään sitä, loput ketjussa olevat luut seuraavat sitä (kuva 5 b). Tilanne toimii samalla tavalla kun valitaan ketjusta toinen luu ja käännetään, loput seuraavat. (Maestri, 2008,122.)



Kuva 5. Suora kinematiikka.

Käänteinen kinematiikka (inverse kinematics) Käänteisessä kinematiikassa (kuva 6) lapsiluu (Child bone) ohjaa Äiti-luuta (Rootbone). Käänteinen kinematiikka kääntää automaattisesti luurankoa, joten liike voidaan luoda siirtämällä luuta eikä

kaikkia luita tarvitse kääntää. Kinematiikan avulla voidaan virtuaalihahmolle luoda realistinen jalan tai käden liike. (Maestri, 2008,122.)



Kuva 6. Käänteinen kinematiikka.

2.1.7 Biped-luuranko

Biped-luuranko on 3Ds Max ohjelmiston Character Studio -pluginin mukana tuleva luurankotyyppi (kuva 3 a). Luuranko on standardi ihmismallille, mutta sitä pystyy muokkaamaan myös monenlaisiin tapauksiin. (Maestri, 2008,150.)

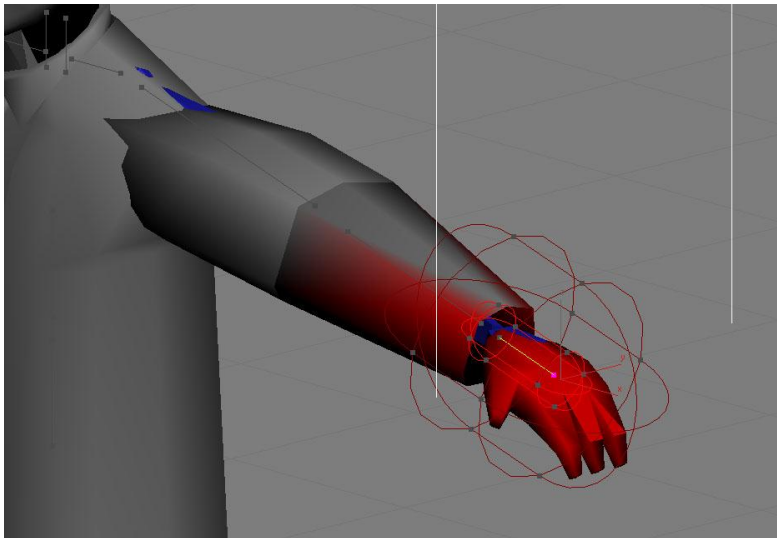
Biped-luurangon hierarkiassa Root-bonelle on oma objekti nimeltä "bip01" ja se sijaitsee luurangon lantiossa.

Biped-luuranko on valmis animoitavaksi, koska siinä on valmiiksi rakennettu käänteinen ja suora kinematiikka. Biped-luurangosta ei saa näitä kinematiikka-asetuksia myöskään pois. Luurangon käsivarren kääntämisessä vaikuttaa suora kinematiikka ja käden ja jalan koukistamisessa käänteinen kinematiikka. (Maestri, 2008,150.)

2.1.8 Virtuaalihahmon ihon luonti

Hahmon ihon luomisella tarkoitetaan hahmon luiden ja geometrian yhteen liittämistä. Valmiille mallille on luotu luuranko, ja luuranko on aseteltu oikealle paikalle hahmon sisällä. Luuranko ja geometria voidaan liittää toisiinsa 3Ds Maxissa kahden muokkaimen avulla. Nämä muokkaimet ovat nk. ihomuokkaimet Skin- ja Physique-muokkaimet. (Pardow 2006, 357.)

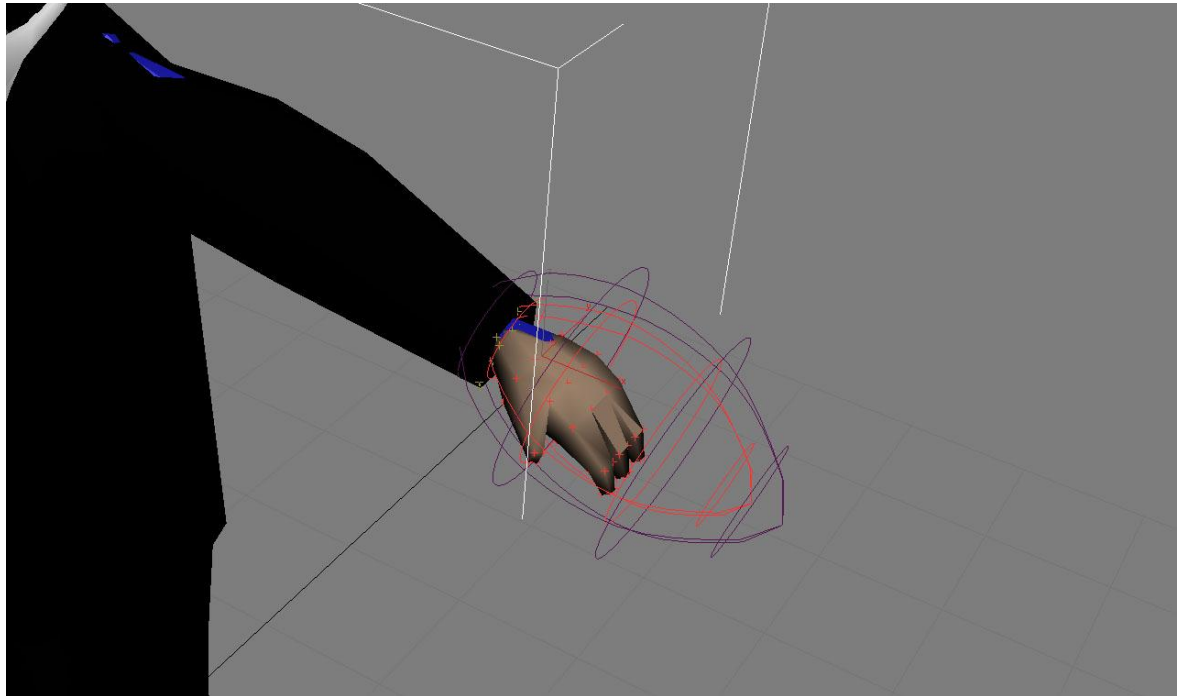
- Skin-muokkain; (kuva 7) toimii bone-luurangon ja biped-luurangon kanssa. Jokaiselle luulle määritetään hahmon geometriassa vaikutusalue skin-muokkaimen kärkipisteiden(envelopien) avulla. Luun reagointi ihoon määritellään vaikutusalueen voimakkuussäätimellä. (Maestri, 2008,126.)



Kuva 7. Vasemman käden vaikutusalue skin-muokkaimella.

- Physique-muokkain; (kuva 8) samankaltainen kun Skin-muokkain. Luun vaikutusalue määritellään hahmon geometriaan ja ihon reagoinnin voimakkuutta säädetään. Ominaisuuksia physique-muokkaimessa on erilaisia kun skin-muokkaimessa esim. geometrian muodon kohoamisessa, ihon liukumis- ja kiertymisominaisuuksissa. Physique-muokkaimen vaiku-

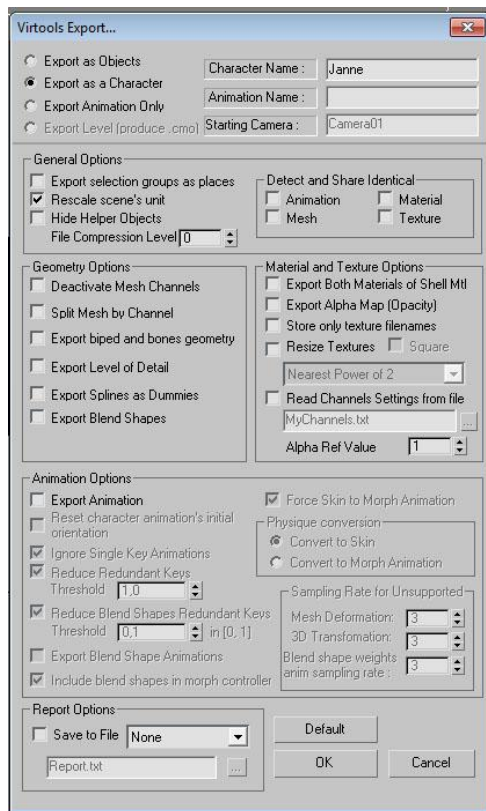
tusaluetyökalu on paljon kehittyneempi kuin skin-muokkaimen. (Maestri, 2008,166.)



Kuva 8. Physique-muokkain.

2.1.9 Virtuaalihahmon siirtäminen Virtools-ohjelmistoon

3Ds Maxin export -työkalulla (kuva 9) voidaan malleja siirtää eri ohjelmiin. Erilaisia malleja pystyy siirtämään myös Virtoolsiin 3Ds Max Virtools Exporterin kautta.



Kuva 9. Virtools exporter -työkalu.

Exporterin kautta pystyy viemään Virtoolsin puolelle hahmoja, erilaisia objekteja niiden materiaaleja ja liikettä. (Virtools 2006. [Viitattu15.12.2010].)

2.2 Virtuaalihahmon kokoaminen

Toimivan virtuaalihahmon etsiminen, jolla oli ainoastaan geometria valmiina, oli työn ensimmäinen ongelma. Tässä vaiheessa testattiin Virtuaalihahmon geometri-an toimivuutta.

Opinnäytetyöntekijää ohjeistettiin käyttämään virtuaalihahmoa, jolla geometria oli mallinnettu valmiiksi. Tämän takia luovuttiin oman virtuaalihahmon rakentamisesta.

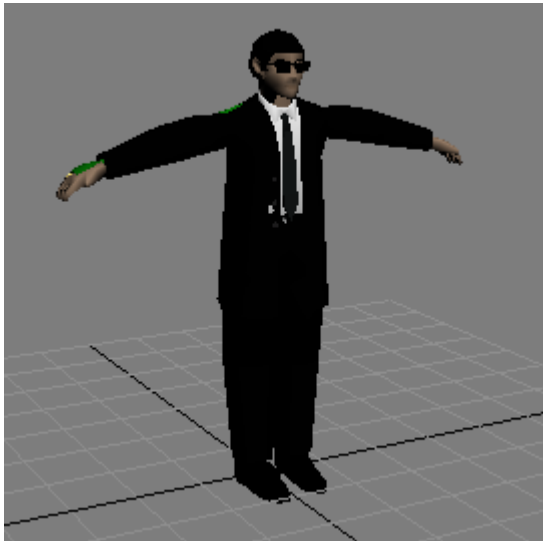
Valmiin virtuaalihahmon löytämiseksi testattiin kahta eri tilannetta.

Ensimmäinen tilanne: 3Ds Max – ohjelmistolla valmistetun hahmon tuodaan internetistä. 3Ds Maxin puolella sille lisätään luuranko ja ihomuokkain.

Virtuaalihahmon geometriaksi soveltuivat 3Ds Maxin ja Virtoolsin tuetut mallinnustekniikat, Polygon- ja Mesh-tekniikka. Hahmot toimivat hienosti 3Ds Maxissa, mutta myös Virtools-ohjelmistossa.

Toinen tilanne: Toisesta mallinnusohjelmistosta otetaan valmis hahmo ja se tuodaan 3Ds Maxiin, joko objektitiedostona tai muuna 3Ds Maxin yhteensopivana tiedostona. 3Ds Maxin puolella sille lisätään luuranko ja ihomuokkain.

Hahmo toimii 3Ds Maxissa, mutta ei Virtools-ohjelmistossa. Virtools-ohjelma ei käsittänyt hahmon geometriaa ja testaustilanteessa Virtoolsin puolella vain hahmon luuranko liikkuu niin kuin piti. Muutamassa tapauksessa myös koko hahmon geometria ei näkynyt. Virtuaalihahmon geometrian mallinnustapa saattaa selittää tämän ongelman.



Kuva 10. Valmis virtuaalihahmo.

Seuraava ongelma oli valita Virtuaalihahmolle luuranko. Virtuaalihahmon luurankoksi käy Biped-luuranko. Opinnäytetyöntekijän mielestä paras luuranko hahmolle syntyy Biped-luurangolla. Se on nopea valmistaa ja sitä käytetään yleensä liikkeen tunnistussovelluksissa sen valmiin kinematiikka-asetuksien takia. Bone-luurangon ongelmana on sen rakenne. Se on vaikea saada toimimaan oikein. Ongelmana bone-luurangossa on, että se koostuu yksittäisistä luista ja sen rakentaminen vaatii paljon työtä ja myös omat ongelmansa on kinematiikka-asetuksien lisäämisessä.

Virtuaalihahmon geometrian muuttamiseksi ihoksi testattiin Skin- ja physique-muokkainta, jolla hahmoon liitetään yleensä luuranko.

Huomattiin, että ainoastaan physique-muokkain toimi Virtools-ohjelmassa oikein virtuaalihahmon ihona.

Neljäs ongelmakohta oli viedä Virtuaalihahmo Virtools-ohjelmaan 3Ds Maxin Virtools exporter -työkalulla.

Työtä tehdessä huomattiin, että monimutkainen malli siirtyy Virtools-ohjelmaan suhteellisen helposti. Virtools Exporter -työkalulla ei tarvitse tehdä hahmolle kun

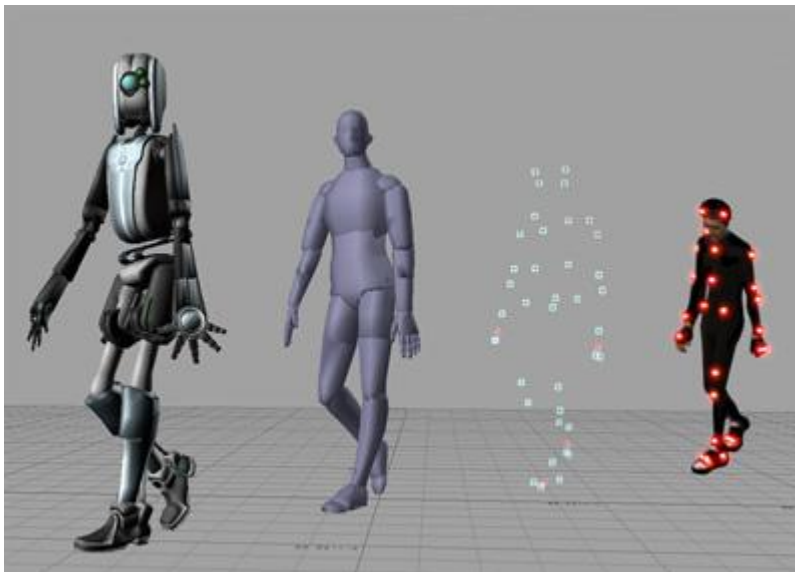
kaksi asetusta: Export as A Character, joka vie 3Ds Maxin -objektin virtuaalihahmona. Lisäksi pitää valita Force Skin to Morph animation, näin Virtools käyttää hahmon geometriaa ihona.

Liitteessä 1 on selvennetty tarkemmin kuinka virtuaalihahmolle lisätään luuranko ja ihomuokkain ja kuinka se tallennetaan export-työkalulla valmiiksi Virtools-ohjelmistoon vietäväksi.

3 LIIKKEENTUNNISTUS JA LIIKKEENTUNNISTUSSOVELLUS

Liikkeentunnistusjärjestelmä ottaa käyttäjän päältä olevista paikannussensoreista mittaustietoa, jonka avulla määritetään virtuaalimallin asema, nopeus ja ulottuvuus. Käyttötarkoituksia voi olla monia esim. virtuaalimaailmassa navigointia, sekä virtuaalihahmon realistisen liikkeen aikaansaaminen esim. kasvojen ja vartalon liikkeet. (Majuri, 2007,1.)

Liiketunnistusjärjestelmiä on erilaisia. Liikkeentunnistuksessa käytetään optisia järjestelmiä, mekaanisia tai magneettisia. (Majuri, 2007,5-6.)



Kuva 11. Liikkeentunnistus. (Täydellisen animaation jäljillä 2007.)

3.1 Magneettinen tunnistus

Magneettikenttiin pohjautuva liikkeentunnistus hyödyntää ympäristöstä saatavaa dc-virtaa. Magneettista järjestelmää käyttävä näyttelijä ohjaa järjestelmää sensoreilla, jotka osaavat kertoa sijaintinsa, kulmansa suhteessa magneettikenttään.

Hankaluuksia järjestelmän käyttäjälle tulee muista metalleista, jotka tuottavat pyöriä magneettikenttään. (Majuri, 2007,6.)



Kuva 12. Magneettisen järjestelmän sensoreita (Täydellisen animaation jäljillä 2007.)

3.2 VR-laboratorion liikkeentunnistusjärjestelmä

VR-laboratorion Cavessa on käytössä Flock of Bird magneettinen paikannusjärjestelmä. Sitä ohjataan Pc-laitteella, jossa on kiinni Extended Range transmitter sekä käytettävissä olevat sensorit. (Seamk VRlaboratorio 2010.)

Flock Of Bird laitteisto on kuuden asteen ja vapauden (six degree of freedom) mittauslaitteisto, joka pystyy paikantamaan yli 30 Flock Of Bird -sensoria samanaikaisesti. (User manual revisions, 2004,1.)

Paikannuslaitteen sensori pystyy tekemään 20 – 144 mittausta sekunnissa, silloin kun sensori sijaitsee \pm neljä jalkaa eli 1,2192 m laitteiston lähetyksyksiköstä (transmitter unit). Laajennetulla kantaman lähettimellä (Extended Range transmitter) sensoreilla on suurempi kantama. (User manual revisions.2004.1).

3.2.1 Magneettisen paikannussensorin lisääminen

Uuden sensorin lisääminen liikkeentunnistusjärjestelmään tapahtuu Vrpn konfiguraatio tiedoston kautta.

Vrpn:n asetuksia muokataan konfiguraatiodostoista. Näihin konfiguraatio tiedostoihin voidaan lisätä, poistaa ja määrittää erilaisia paikannuslaitteita ja sensoreita. (Virtools [viitattu 12.10.2010].)

3.3 Paikannussensorien lisääminen Cave-järjestelmään

Opinnäytetyön tavoitteena oli lisätä olemassa olevaan laitteistoon Virtuaalihahmolle tarvittava määrä sensoreita, yhteensä seitsemän kappaletta. Paikannussensoreita VR-laboratoriossa oli käytössä kaksi ja ylimääräisiä viisi kappaletta.

Opiskelija työskenteli Vrpn serverin kanssa. Virtools osaa tulkita sen kautta ainoastaan magneettisia Flock Of Bird -laitteita.

Työssä lisättiin laitteita yksitellen ja samalla määriteltiin Vrn server konfiguraatio tiedostoihin tarvittavat asetukset. Näihin kuuluivat seuraavat asennukset

- Paikannusjärjestelmän pc:lle uusi vrpn_server laitetedosto.

Opinnäytetyöntekijä testasi järjestelmää ja määritti ensiksi VRPN_Tracking_VRNR konfiguraatiodostot Paikannusjärjestelmän pc-koneelle, mutta tämä jumitti Virtools-kokoonpanotiedoston. Tähän ei löytynyt mitään järkevää syytä. Eikä testauksen avulla saatu asiaa korjaantumaan.

Tämän jälkeen VRPN_Tracking_VRNR konfiguraatiotiedostot määritettiin niin, että niitä ei lueta paikannusjärjestelmän Pc:ltä. Tiedostot määriteltiin omaan työkoneen kansioon. Näin opinnäytetyöntekijä sai testatuksi virtuaalihahmon liikettä.

Paikannuslaitteiston lisääminen oli vaikea operaatio sillä erilaisia virhetilanteita tuli paljon. Virhetilanteita tuli paikannussensorin magneettikenttähäiriöistä, laitteiston vääristä asetuksista, erilaisissa konfiguraatio määrytyksissä sekä paikannusjärjestelmän pc koneen vanhojen osien rikkoutumisesta.

Liitteessä 2 on tarkemmin kerrottu kuinka lisätään laitteita ja kuinka ne saatiin toimimaan.

3.4 Liikkeentunnistussovellus

Liikkeentunnistusjärjestelmällä voidaan tehdä kahdenlaista sovellusta.

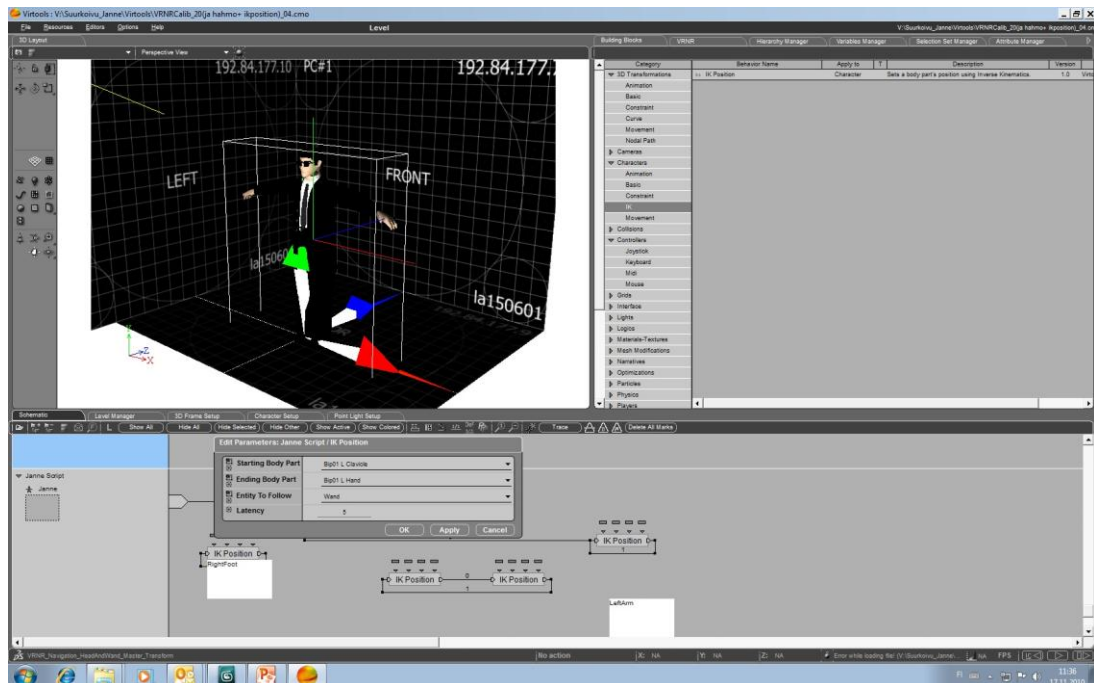
- Offline sovellus: Käyttäjä ei näe liikettä reaaliaikaisesti, mutta liikkeitä tehdään liikekirjasto myöhempää käyttöä varten.
- Online sovellus: Käyttäjä näkee liikkeen ja liike on reaaliaikaista, sitä ei tallenneta liikekirjastoon. (Majuri 2007, 4.)

3.5 Liikkeentunnistussovelluksen rakennustyökalu

Virtools on Dassault Systèmes -yhtiön valmistama visualisointiohjelma. Sitä käytetään esittämään reaaliajassa erilaisia kolmiulotteisia esityksiä. Ohjelmistoa käytetään grafiikan esittämiseen, animointiin ja konsolipelien valmistukseen.

Erilaisissa 3d-mallinnusohjelmistoissa rakennettuihin malleihin saa Virtoolsin avulla linkitettyä erilaisia koodipätkiä. Näitä koodipätkiä tai rakennuspalikoita Virtoolsissa kutsutaan Building blockeiksi. Ne antavat objekteille uusia ominaisuuksia

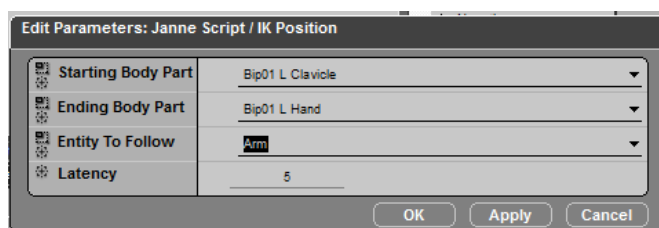
esim. värin, liikkeen tai kontrollin, jotka ohjaavat animaatiota. Erilaisia rakennuspalikoita on Virtoolsissa n. 500 kpl. (Virtools user's guide,2006,45.)



Kuva 13. Virtools-ohjelma.

Virtoolsin avulla voidaan tehdä myös erilaisia virtuaaliodellisuusesityksiä. Tätä varten täytyy olla olemassa lisenssi Virtoolsin mukana tulevaan VRPack-pakettiin. (Dassault systemes,2010.)

3.5.1 IkPosition rakennuspalikka



Kuva 14. IkPosition building block

IkPosition building block on valmis koodinpätkä käänteisen kinematiikan luontiin Virtools-ohjelmistossa. Tämä buildingblock koodataan avaamalla se ja lisäämällä Starting Body Part -kohtaan luu, jonka halutaan olevaan käsiteltävän luuketjun ensimmäinen luu. Ending Body part on luu, joka on käsiteltävän ketjun viimeinen luu. Entity To Follow -kohtaan voidaan määrittää objekti, mitä ending Body Part osiossa oleva luu seuraa. Latency arvo määrittää miten sujuvaa kinematiikka ketjun liike on. (Gauthier 2005. 193 - 196.)

3.6 Liikkeentunnistussovelluksen rakentaminen

Tässä vaiheessa tutkittiin ja dokumentoitiin se miten Virtools-kokoonpanotiedostoon lisätään yksi sensori, mitä skriptejä, buiding blockkeja ja parametrejä kokoonpanotiedostoon koodataan.

Liitteessä 3 on selvitetty kuinka työssä lisättiin Virtools-kokoonpanotiedostoon paikannuslaitteita.

Liitettä 3 hyväksikäyttäen opinnäytetyön tekijä lisäsi Virtools-kokoonpanotiedostoon yhteensä neljä paikannussensoria. Näin Virtools-kokoonpanotiedoston käytettäväksi tuli yhteensä seitsemän paikannussensoria.

3.6.1 Virtuaalihahmon lisääminen

Tässä vaiheessa ideana oli liittää paikannuslaitteen sensori hahmon luurakenteen tiettyyn luuhun esim. käteen ja liikuttaa luuta paikannussensorin liikkeen avulla.

Tavoitteeseen pääsemiseksi tutkittiin ja dokumentoitiin se kuinka hahmo tuodaan Virtools-kokoonpanotiedostoon ja minkälainen rakenne sille luodaan.

Työssä ohjeistettiin käyttämään Virtools building blockkeja, sekä saatavilla olevia koodinpätkiä. Virtoolsin omaa VSL-ohjelmointikieltä ohjeistettiin välttämään työssä. Koodi, joka määrittä virtuaalihahmon liikkeen. Kirjoitettiin Virtools-ohjelmistoon tuodun virtuaalihahmon koodialueelle, jossa opiskelija käytti sellaisia rakennuspalikoita, jolla virtuaalihahmo saatiin liikkeelle. Työssä käytettiin mm. Ikposition- ja mimic rakennuspalikoita runsaasti.

Tarkemmin Virtuaalihahmon liikkeenrakentamisesta löytyy liitteestä 4

4 JOHTOPÄÄTÖKSET

Opinnäytetyön tavoitteena oli luoda VR-laboratorion caveen virtuaalihahmon liikkeen tunnistus. Työssä tutkittiin aluksi minkälainen, virtuaalihahmo pitää olla että se toimii Virtools-ohjelmistossa. Työssä tutkittiin myös miten paikannussensoreita lisätään olemassa olevaan paikannusjärjestelmään. Miten sensorit ja virtuaalihahmon liike saadaan toimimaan yhdessä.

Virtuaalihahmon osalta työssä käytettiin 3Ds Maxia ja Virtoolsia. Tutkittiin minkälainen geometria, luuranko ja iho toimivat Virtools-ohjelmassa.

Virtuaalihahmon osalta työssä oli vaikeuksia saada hahmo toimimaan oikein Virtoolsissa, mutta testauksien jälkeen hahmon vienti onnistui ja hahmo toimi kuten pitikin.

Seuraavaksi asennettiin olemassa olevaan paikannusjärjestelmään lisää laitteita. Ongelmia työssä oli paikannussensorien asennuksessa ja käytössä. Ongelmista suurin osa ilmeni mm. paikannusjärjestelmää ohjaavan tietokoneen komponenttien peittämisestä. Paikannussensoreiden osalta ongelmia oli kahden sensorin rikkoutuminen.

Lopulta sensorit saatiin toimimaan ja pienellä laitteiston koordinaatiosäädöillä saatiin sensorit toimimaan oikein ja paikantamaan Vr-laboratorion Cave-tilaa.

Lopulta kun Virtuaalihahmo oli rakennettu 3Ds Max -ohjelmistossa ja hahmo toimi moitteettomasti Virtools-kokoonpanotiedostossa sekä paikannusjärjestelmän sensorit olivat oikein asennettu. Sen jälkeen lisättiin paikannussensoreiden datan lisääminen Virtools-kokoonpanotiedostoon.

Sensoreiden datan lisäämiseksi Virtools-kokoonpanotiedostoon käytettiin esimerkiksi jossa oli paikannus lisätty päälle ja wand-ohjaimelle. Tätä tutkimalla selvitin miten sensoreita lisätään virtools-kokoonpanotiedostoon.

Työssä saatiin onnistuneesti lisättyä sensoreiden data tietoa virtools-kokoonpanotiedostoon. Työssä lisättiin virtools-kokoonpanotiedostoon mahdollisuus seitsemään sensoriin, mutta sitä käytettiin lopulta vain viidellä sensorilla.

Kuitenkin yksi vika järjestelmään jäi. Useamman sensorin yhteiskäyttö jumitti Virtools Vr-playerin. Syytä tälle ei lukuisista yrityksistä huolimatta saatu selville. Virtuaalihahmoa päästiin onnistuneesti testaamaan työ-tietokoneella virtools-ohjelmiston develop (kehitys-tilassa).

Paikannussensoreiden lukumäärän pudottua viiteen työssä ei pystytty testaamaan kaikkia raajoja.. Työssä keskityttiin koko hahmon liikuttamiseen, pään ja käsien ja selkärangan liikkeen paikantamiseen.

Magneettikenttä aiheutti häiriötä hahmon testaamisessa ja liikettä ei saatu sujuvaksi.

Virtuaalihahmon liikkeen luomiseen työssä käytettiin seuraavia koodikokonaisuuksia.

Työssä testattiin koko virtuaalihahmon liikkumista Cave-tilassa ja se onnistui hyvin Keep At constant distance bb:n avulla.

Päänliike ei toiminut niin kuin sen olisi pitänyt. Työssä etsittiin ratkaisua 3Ds Maxin puolelta biped-luiden hierarchy manager -osiosta, Virtools exporter -työkalusta ja Virtoolsin koodista.

Päänliike saatiin toimimaan lopulta LookAt-rakennuspalikan avulla. jolloin virtuaalihahmon pää pysyi hartioilla ja liikkui sinne mihin liikkeentunnistuksen käyttäjä katsoi.

Työssä testattiin virtuaalihahmon käsiin IkPosition-rakennuspalikkaa, ja se toimi niin kuin piti.

Selkärangan taitos saatiin työssä toimimaan KeepAtConstantDistance-rakennuspalikan sekä mimic-rakennuspalikan avulla niin, että virtuaalihahmon selkä liikkuu sinne mihin liikkeentunnistusanteri menee.

Testaus vaiheessa otettiin myös videokuvaa testauksesta, virtuaalihahmon liikkeestä videot löytyvät liitteenä olevasta cd-levystä.

Työ oli vaikea, mielenkiintoinen ja työssä todella oppi virtuaalihahmon luujärjestelmän asennuksesta sekä virtuaalihahmon geometrian muuttamisesta ihoksi. Myös Virtools-ohjelmistosta työn aikana oppi koodaamisen logiikan. Paikannussensoreiden asennuksesta sekä niiden määrittämisestä Virtools-kokoonpanotiedostoon oppi todella paljon. Kaikista teknisistä ja käytännön veikeuksista huolimatta.

LÄHTEET

- 3DVIA Virtools 5.[WWW-lähde]. [Viitattu: 12.10]. Saatavissa:
<http://www.3ds.com/products/3dvia/3dvia-virttools/portfolio/authoring/3dvia-virttools/>
- Autodesk, 2010. Autodesk 3Ds Max Products. [WWW-lähde]. [Viitattu: 1.10.2010]. Saatavissa:
<http://www.autodesk.co.uk/adsk/servlet/pc/index?siteID=452932&iid=14596387>
- Flock of Birds. 2007. Position and Orientation measurement system. [WWW-lähde]. Installation and operation guide. [Viitattu: 12.11.2010]. Saatavissa: ftp://ftp.ascension-tech.com/MANUALS/Flock_of_Birds_Manual-RevC.pdf
- Fox, B. 2004. 3Ds Max 6 Animation: CG Filmmaking from concept to Completion. California: McGraw-Hill/Osborne
- Gauthier, J-M. 2005. Building Interactive Worlds in 3D: visual sets and Pre-visualization for games, Burlington: Focal press/Elsevier
- Halme, T.2006. Kolmiulotteisen animaation hahmon kinematiikka. [WWW-lähde]. Lahden Ammattikorkeakoulu. [Viitattu: 17.11.2010]. Saatavissa:
<http://publications.theseus.fi/bitstream/handle/10024/11529/2006-08-17-07.pdf?sequence=1>
<http://publications.theseus.fi/bitstream/handle/10024/11529/2006-08-17-07.pdf?sequence=1>
- Lehtovirta, P. & Nuuttinen, K. 2000. 3D sisällöntuotannon peruskirja. Jyväskylä: Dosendo
- Maestri, G. 2008. 3Ds Max at a Glance. [WWW-lähde], [Viitattu: 12.10], Saatavissa:
<http://site.ebrary.com/lib/seamkebrary/docDetail.action?docID=10296680&p00=character%20rigging>
- Majuri, M.2007. Täydellisen animation jäljillä. [WWW-lähde]. Informaatioverkostot: studio4.[Viitattu: 28.10.2010]. Saatavissa:
http://users.tkk.fi/~mmmajuri/studio4/taydellisen_animaation_jaljilla.pdf
- Pardow, L. 2006. Game Character Animation All in One. [WWW-lähde]. [Viitattu: 5.10]. Saatavissa:

<http://site.ebrary.com/lib/seamkebrary/docDetail.action?docID=10228204&p00=3ds%20max%20rigging>

Seamk. 2010. Virtuaalilaboratorio. [WWW-lähde]. [Viitattu: 5.10.2010]. Saatavissa: <http://www.seamk.fi/contentparser.aspx?deptid=1374&abc=2>

Tammi, M. 2004. Liikkeentunnistus - Input. [WWW-lähde]. TKK Informaatioverkostot: studio4. [Viitattu: 28.9.2010]. Saatavissa: <http://users.tkk.fi/~mtammi/studio4/Liikkeentunnistus.pdf>

Virtools. 2006. Virtools User's guide. [WWW-lähde]. [Viitattu: 10.12]. Saatavissa: http://student-theater.hku.nl/~sil/Geen%20Winteracademie/Virtools%20Les1/Virtools_User_Guide.pdf

Virtools. Virtools solution for totally immersive pc cluster based vr. [WWW-lähde]. [Viitattu: 12.10]. Saatavissa: shttp://www.virttools.co.kr/solutions/products/pdfs/Virtools_VR_Pack.pdf

Virtools.2006. Virtools Exporter for 3DS Max Online Reference. [WWW-lähde]. [Viitattu: 15.12.2010]. Saatavissa: <http://www.virttools.co.kr/dc/onlineref/Virtools/import/3dsmax/readmeMax.htm>

Kuvat

Animation importer. 2010. For 3Ds max v5.0+ and gmax 1.2. Kuva 3. Saatavissa: <http://ghost.halomaps.org/bluestreak/animation/>

Täydellisen animaation jäljillä. 2007. Kuva 11 – Kuva12. Saatavissa: http://users.tkk.fi/~mnmajuri/studio4/taydellisen_animaation_jaljilla.pdf

LIITTEET

Sisällys

LIITE 1 VIRTUAALIHAHMON VALMISTUS	446
LIITE 2: SENSORIEN KÄYTTÖÖNOTTO	62
LIITE 3: SENSORIEN KÄYTTÖÖNOTTO VIRTOOLS- KOKOONPANO TIEDOSTOSSA	84
LIITE 4: VIRTUAALIHAHMON LIIKKEEN KOODAAMINEN	109

LIITE 1 VIRTUAALIAHAHMON VALMISTUS

Virtuaalihahmon Valmistus

Virtuaalihahmon mallinnustavaksi käy 3Ds Maxin tukemat mallinnustekniikat. Hahmon voi rakentaa Mesh, Polygon mallinnusta käyttämällä.

Hahmon mittakaava

Hahmon mallinnuksessa olisi hyvä käyttää oikeaa mittakaavaan ja metrijärjestelmää, sillä hahmo vietään Virtools-ohjelmistoon, joka käyttää ainoastaan metrijärjestelmää.

Työssä virtuaalihahmossa käytettiin 3Ds Maxin -oletus mittayksikköä (generic unit) mallin saa oikeaan mittakaavaan Virtools exporter -valikossa.

Opinnäytetyöntekijä latasi valmiin hahmon netistä osoitteesta:

<http://www.3dm3.com/modelsbank/model118.htm>

Hahmo mallin tuonti 3ds Maxiin muista ohjelmista

Hahmomallin tuonnissa (importoinnissa) käy 3Ds Maxilla luotu hahmo.

Valmiin hahmon importointi esimerkiksi toisesta mallinnusohjelmasta ja luurangon rakentaminen 3Ds Maxin avulla ei tule kysymykseen. Virtoolsissa hahmon geometria ei enää toimi.

Lujärjestelmän valinta

Lujärjestelmäksi voisi valita joko Bone- tai Biped- lujärjestelmän. Tekijä valitsi vaihtoehtoista Biped luurangon. Biped luuranko on nopea valmistaa ja sitä käytetään yleensä liikkeentunnistus-sovelluksissa sen symmetrisen lujärjestyksen (hierarkian) takia.

Ihomuokkaimen valinta

Hahmon geometrian liittäminen luihin onnistuu 3Ds Maxissa Physique-muokkaimella. Virtuaalihahmon loppusijoituspaikka Virtools-ohjelma tukee ainoastaan Physique-muokkainta.

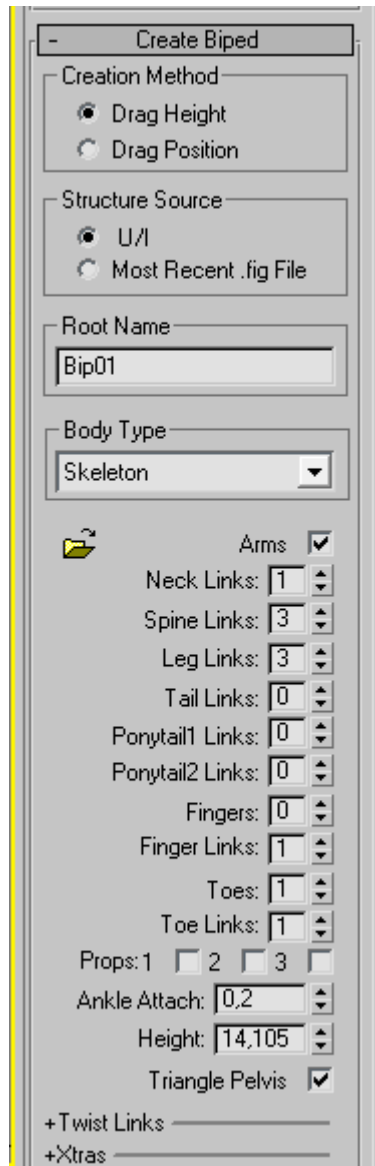
Tekijä kokeili rakentaa kummallakin ihomuokkaimella hahmolle ihon. Homma hoi-
tui helposti physique-muokkaimella, mutta ongelmaksi muodostui nivelten taitos-
kohtien realistinen liike. Virtoolsissa toimii kunnolla ainoastaan physique-
muokkain. Tässä työssä tekijä ei aseta Physique-muokkainta silmille, suulle, sor-
mille ja varpaille.

Esimerkki Virtuaalihahmon rakentamisesta

3Ds Max
Virtuaalihahmon tuonti ja ohjelman esivalmistelu
<ol style="list-style-type: none"> 1. Valmis hahmo avataan 3ds Max 2008 -ohjelmaan. 2. Tarkistetaan että metri järjestelmä on päällä. (customize – Units setup) ja sieltä valitaan metric – meters ja OK. 3. Napautetaan wireframe-ominaisuus päälle. 4. Käyttäjä suurentaa front(etu)-näkökuvan Maximize Viewport Toggle- toiminnolla.
Luurangon asetukset
<ol style="list-style-type: none"> 1. Valitaan työkaluriviltä Create – systems - biped 2. Tässä vaiheessa voi nyt määrittää biped luurangon luurankenteen avautuvasta ominaisuus-valikosta.

Taulukko 1. 3Ds Max virtuaalihahmon tuonti ja ohjelman esivalmistelu.

Biped-muokkaimen structure-ominaisuus paneelista (Kuva 1.) löytyy luurangon erilaisia ominaisuuksia mm. Root name (Äiti luun nimi), Skeleton type (luurangon tyyppi) sekä luurangon luut.



Kuva 15. Biped Structure.

Luurangon asetuksissa löytyy hahmon luiden nimet ja niiden lukumäärä. Tästä vain käyttäjä lisää tai vähentää luiden lukumäärää tarkoitukseen sopivaksi.

Opinnäytetyön tekijä suosittelee selkärangan (Spine links) lukumääräksi kolmea, realistisuuden takia. Keskivartalo taittuu normaalisti kolmesta kohtaa.

Tärkeää: Vaikka sormia ja varpaiden liikettä ei animoisi. Tarvitsee hahmo ainakin yhden sormen ja varpaan jotta ihon liittäminen onnistuu kämmeneen ja jalkaan. Niinpä Fingers ja toes kohdassa täytyy olla ainakin ykkönen.

Luiden luonti

Biped luuranko luodaan vetämällä se vain 3ds maxin scenen näkymään. Tämän jälkeen luurankoa pystyy siirtämään ainoastaan luurangon "bip01" root (juuri/äiti) luusta. Luu on helppointa luoda 3Ds Maxin front(etu)- näkymään.

1. Työssä valitaan biped-luujärjestelmä ja tarkistetaan luiden lukumäärä.
2. Piirrettiin virtuaalihahmon luurangon pituus hiirellä vasennappi pohjassa hahmon geometrian viereen.

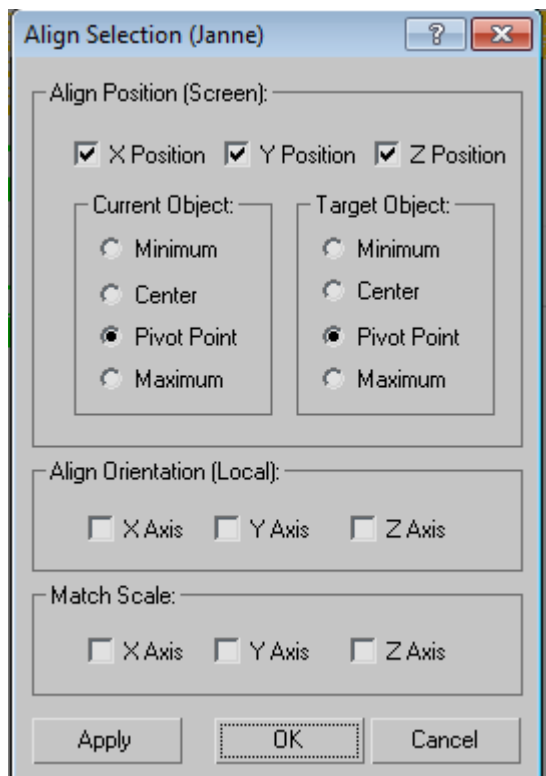
Taulukko 2. Luurangon luonti.

Biped luuranko vedetään esiin 3Ds Maxin sceneen ja työssä paras tapa on vetää luuranko virtuaalihahmon geometrian viereen käyttäen geometriaa mittatikkuna luurangolle.

Luiden vienti virtuaalihahmon sisälle

Luuranko on nyt luotu, mutta se ei ole oikeassa kohdassa virtuaalihahmoon nähdessä. Luuranko täytyy asettaa hahmon sisälle niin että se "istuu" siellä mahdollisimman tarkasti oikealla paikalla.

Käyttäjät valitsee luurangon (Tärkeää: Root bonen "bip01" (juuriluun)) ja ominaisuuksista figure moodin.



Kuva 16. 3Ds Max -Align työkalu.

Figure mode on luurangon muokkauksen kannalta tärkeä ominaisuus. Tämä täytyy olla aktiivisena kun luurangon sijaintia tai yksittäisten luiden mittakaavaa muuttaa.

Luiden vienti virtuaalihahmon sisälle
1. Työssä valittiin Main Toolbar (päätyökaluvalikosta) Align-työkalu ja tämän jälkeen virtuaalihahmo.
2. Avautuu align-työkalun ikkuna ja siitä määritellään kuvassa 2 näkyvät asetukset.

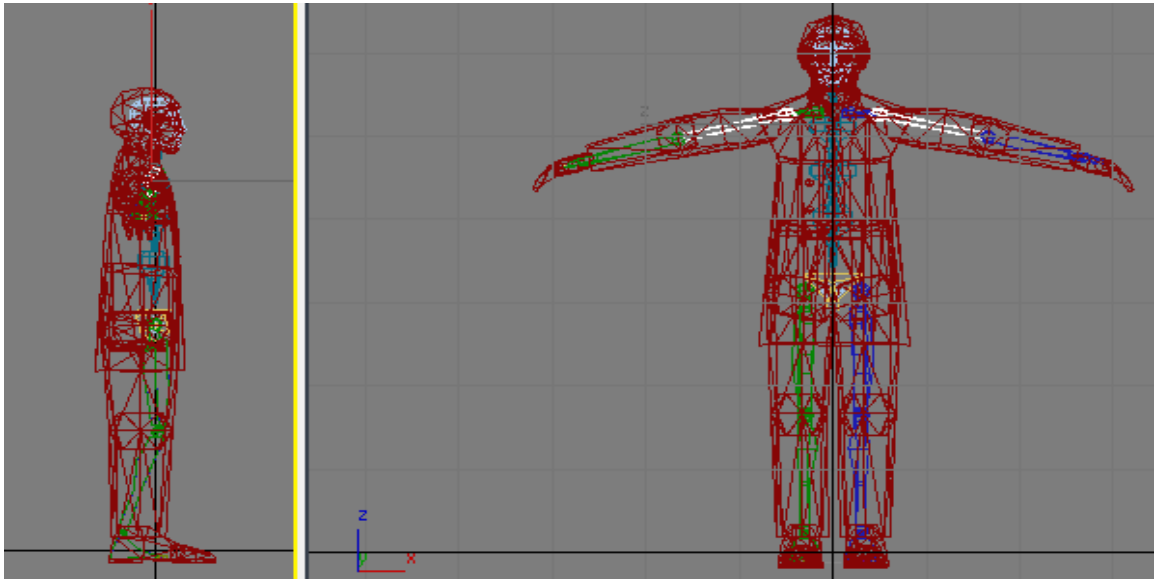
Taulukko 3. Luurangon vienti Hahmon sisälle.

Luiden tarkempaa asettelua

Luuranko pomppaa hahmon sisälle. Käyttäjä tarkastaa virtuaalihahmon **perspective**-näkyvällä, joka suunnasta.

Luut näyttävät liian suurilta ja ne eivät ole oikealla paikoilla. Luita täytyy asetella hieman tarkemmin virtuaalihahmon sisälle, niin että luut ovat mahdollisimman realistisesti oikeilla paikoillaan.

Työssä käytettiin Select and Uniform Scale -työkalua luiden mittakaavan muuttamiseen 3d hahmon hartioiden ja raajojen asettelussa. opinnäytetyöntekijä myös katsoi, että luurangon ja hahmon geometrian nivelet ovat tarkasti siinä missä pitääkin.



Kuva 17. Virtuaalihahmo + biped luuranko.

Virtuaalihahmon Ihon liittäminen

Virtuaalihahmon ihon liittäminen

1. Lisätään hahmon geometriaan Physique-muokkain aktiiviseksi.
2. Napautetaan Physique-muokkaimessa olevaa Attach To Node -näppäintä.

Valitaan "Bip01 luu"

3. Physique intialization -ikkunassa valitaan Intialize
4. Testataan luuta

Taulukko 4. Virtuaalihahmon ihon liittäminen.

Luun testaaminen

Testaus tapahtuu luuta liikuttamalla, joten valitaan haluttu luu.

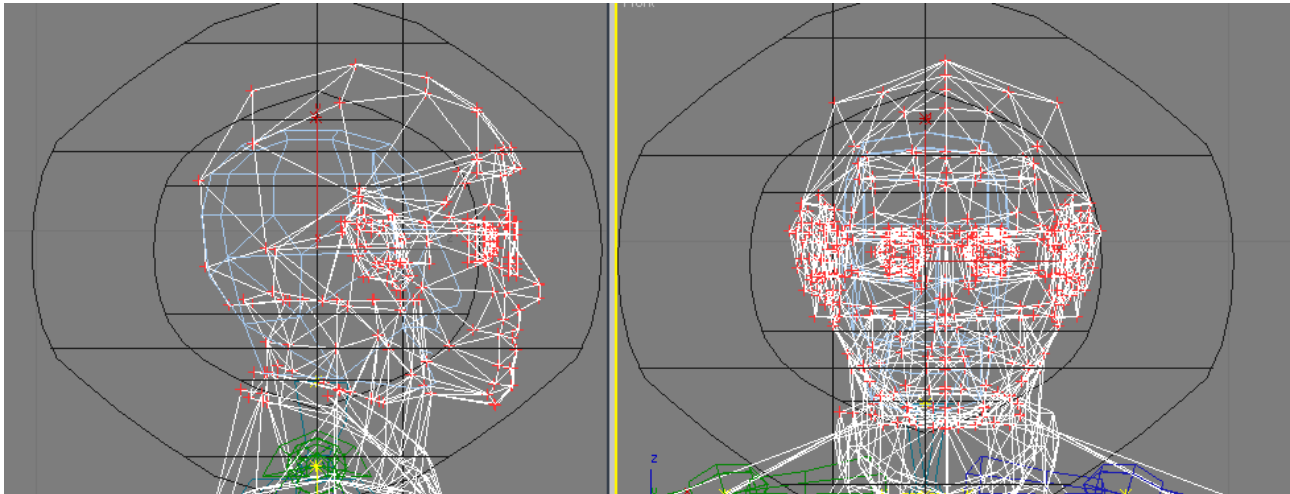
1. valitaan testattava luu tässä tapauksessa bip01 L Hand

3. Valitaan luu ja Select And Move ja Select And Rotate -työkalulla testataan luun liikettä. Kun luun liikkuu realistisesti, joka suuntaan. Siirrytään seuraavaan luuhun ja testataan luun liikettä.

Luun envelopien säätö

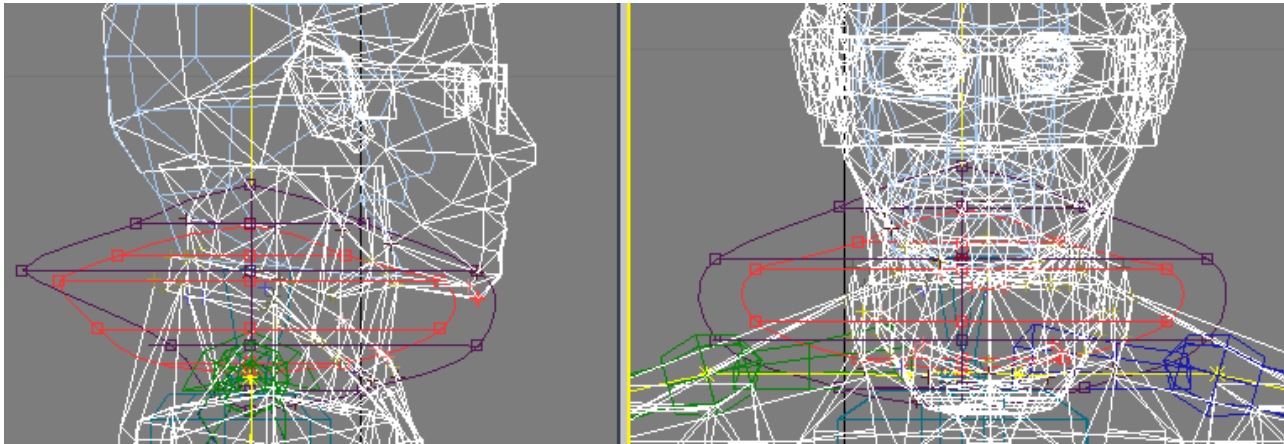
Jos Virtuaalihahmon geometria ei reagoi realistisesti luu testauksessa. Sitä säädetään yleensä physique-muokkaimen Radial Scale -säädintä ja Parent -ja Child Overlapping säädintä käyttämällä.

1. Valitaan Physique-muokkaimen Envelope ala kategoria.
2. Valitaan 3ds Maxin scenestä virtuaalihahmon pää(Kuva 1), joka näkyy nyt punaisena
3. Säädetään Radial Scale säädintä ja Parent ja Child Overlapping säädintä käyttämällä vaikutusaluetta niin kauan kunnes liike on realistinen.



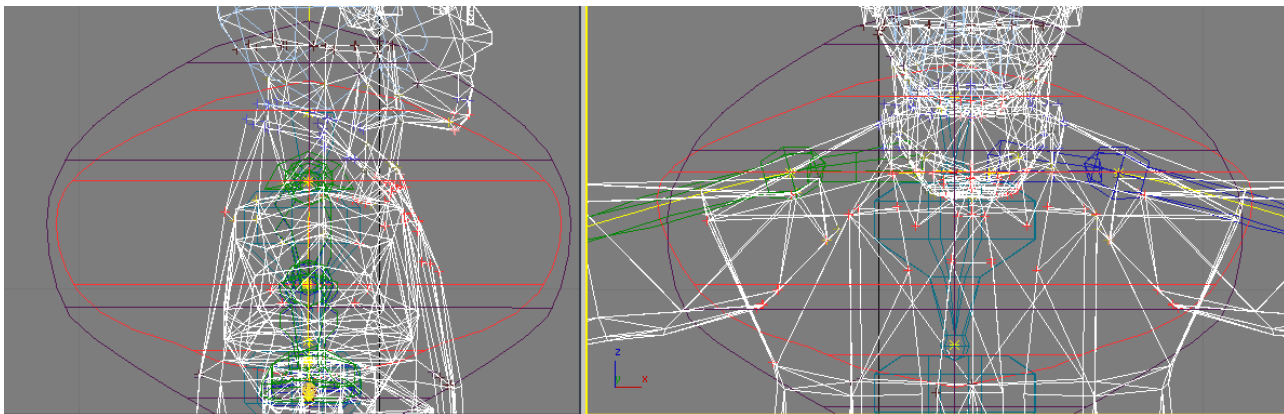
Kuva 18. Virtuaalihahmon pään envelopit.

4. Testataan ja kokeillaan. Kun luu kääntyy realistisesti, siirrytään seuraavaan luuhun. Tässä kuvassa (Bip01 Head)



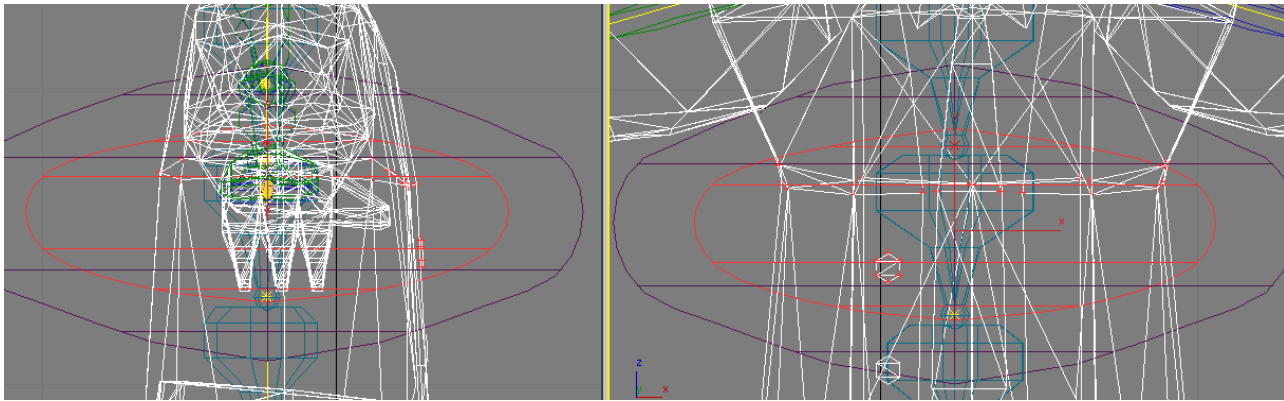
Kuva 19. Virtuaalihahmon kaulan Envelopit.

5. Niskan luu kääntyy aina pään kanssa. kuvan asettelu on riittävä realistisen niskan liikkeen aikaansaamiseksi. Tässä kuvassa (Bip01 Neck)



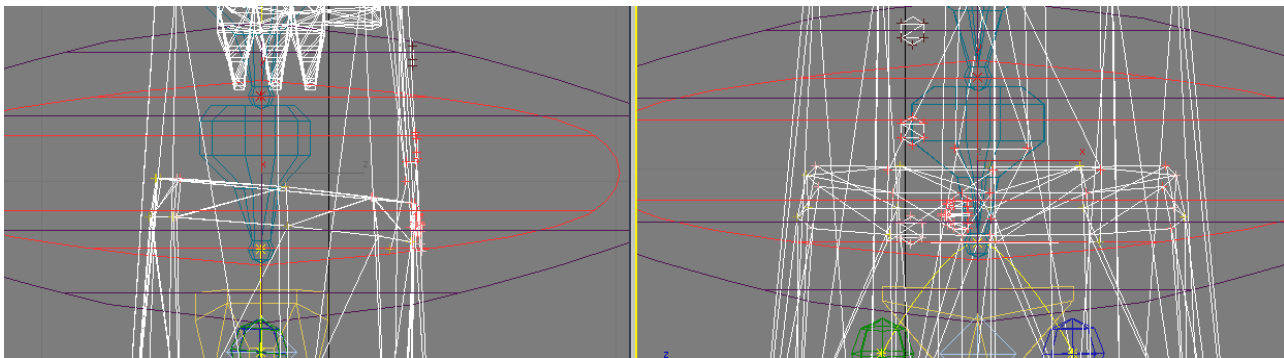
Kuva 20. Virtuaalihahmon rintakehän envelopit.

6. Tämän jälkeen siirrytään selkärangan luihin. Ensimmäinen selkärangan "nikama" asetetaan kuvan osoittamalla tavalla, niin se toimii realistisesti. Testauksessa pitää vielä tarkistaa luun liike. (Bip01 Spine2 -luu)



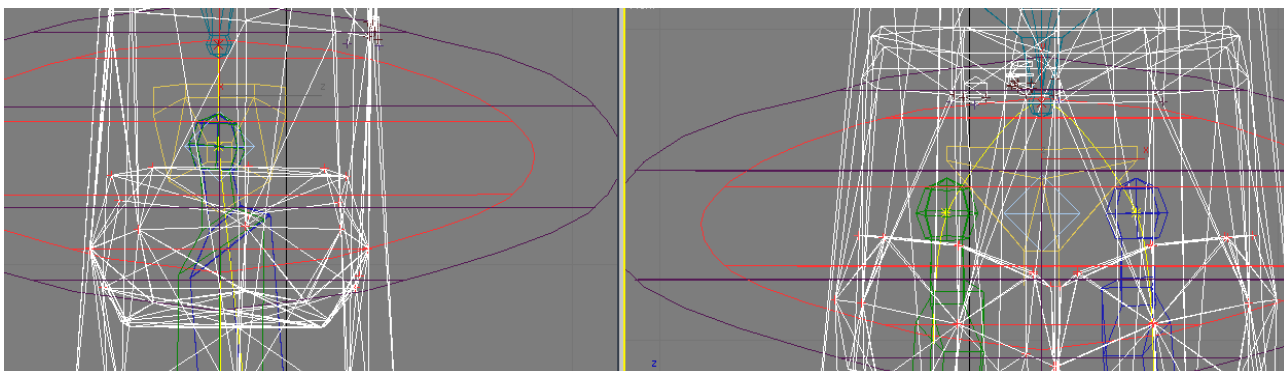
Kuva 21. Virtuaalihahmon selkärangan envelopit.

7. Jatketaan samalla kaavalla kaikki selkärangan luut. Valitaan seuraava luu ja sen vaikutusalue. (Bip01 Spine1)



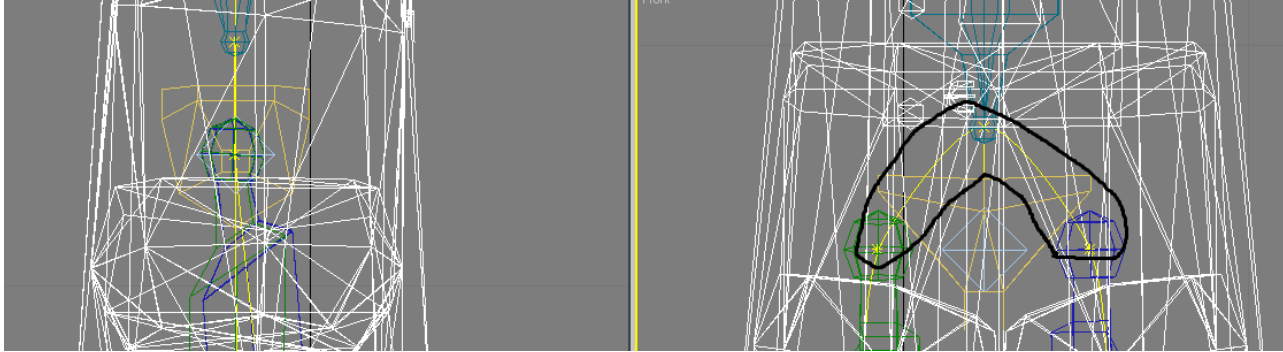
Kuva 22. Virtuaalihahmon selkärangan envelopit.

8. Jatketaan samalla kaavalla kaikki selkärangan luut. Valitaan seuraava luu ja sen vaikutusalue. (Bip01 Spine1)



Kuva 23. Selkärangan envelopia.

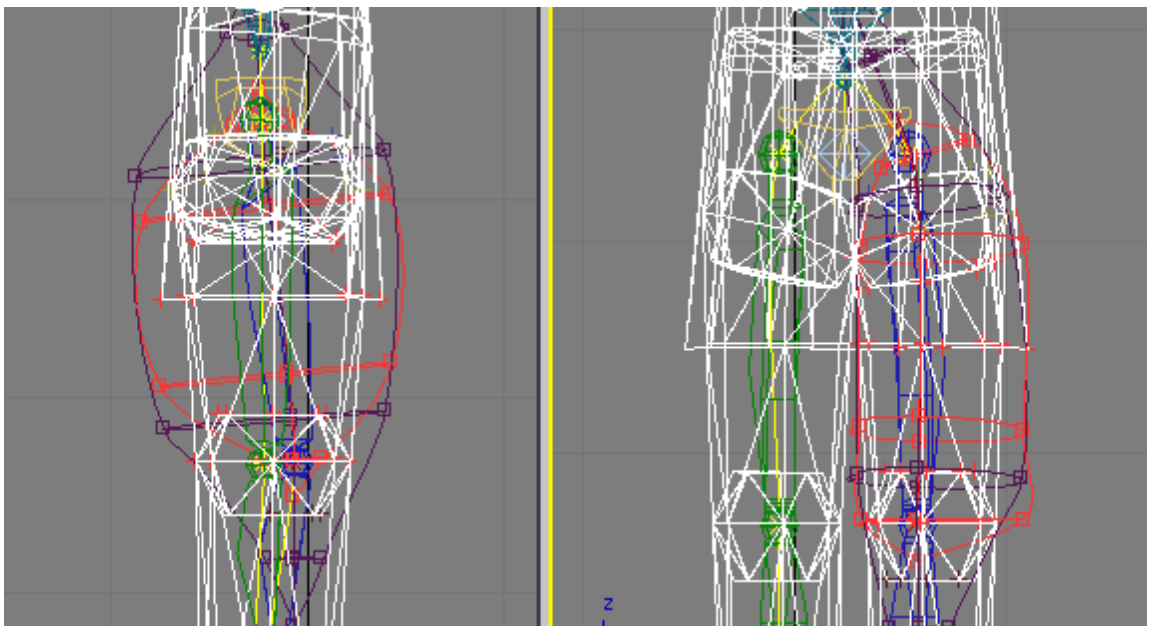
9. Työssä on saatu selkäranka valmiiksi. Siirrytään "Bip01 Pelvis" luuhun. Asetetaan sen vaikutusalueeksi hahmon lantion seutu.



Kuva 24. Lantion envelopit.

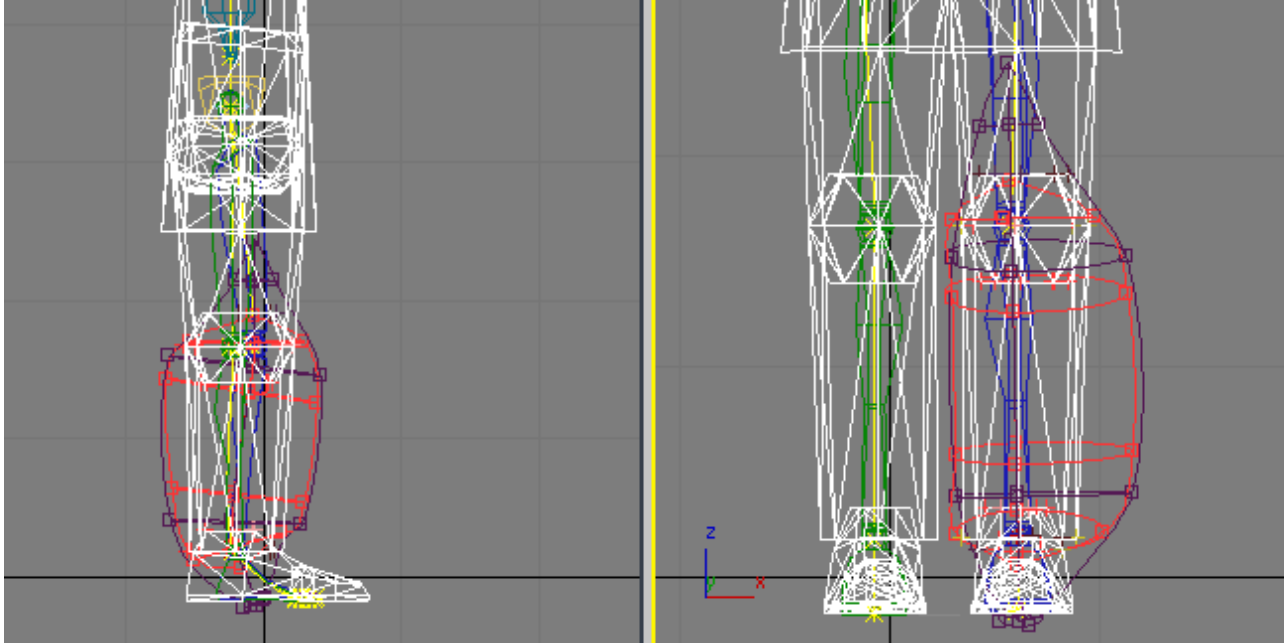
10. Luu "Bip01 Pelvis" on lantion seudulla mustan alueen sisällä oleva envelope vaikutusalue. Tällä vaikutusalueella simuloidaan "lantionseudun liikettä", mutta tässä työssä ei tarvitse "lantiorutistusta" joten kumpaankin käyttäjä määrittä Radial Scale arvoksi nollan.

11. Siirrytään vasempaan jalkaan. (Bip01 L Thigh) ja määritellään envelopit kuvan osoittamalla tavalla.



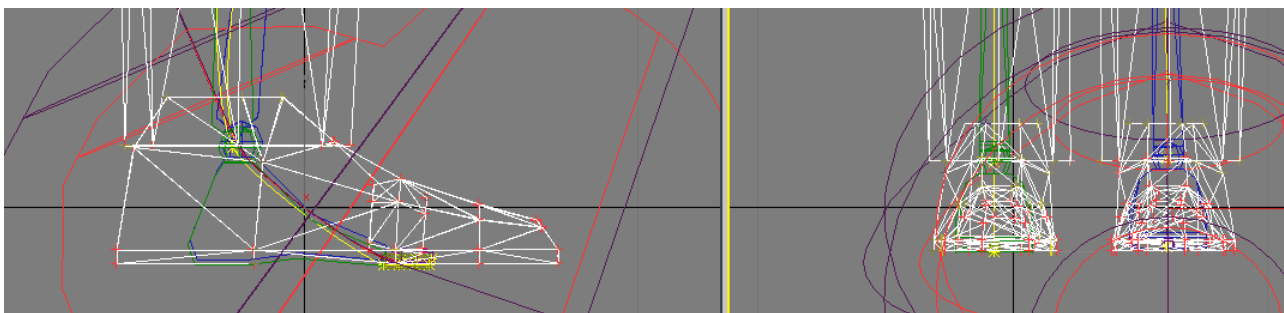
Kuva 25. Vasemman jalan envelopet.

12. Siirrytään virtuaalihahmon reisiluuhun (Bip01 L Calf). Määritetään envelopit kuvan osoittamalla tavalla.



Kuva 26. Vasemman säären envelopit.

13. Siirrytään (Bip01 L Foot). Määritetään envelopit kuvan osoittamalla tavalla.

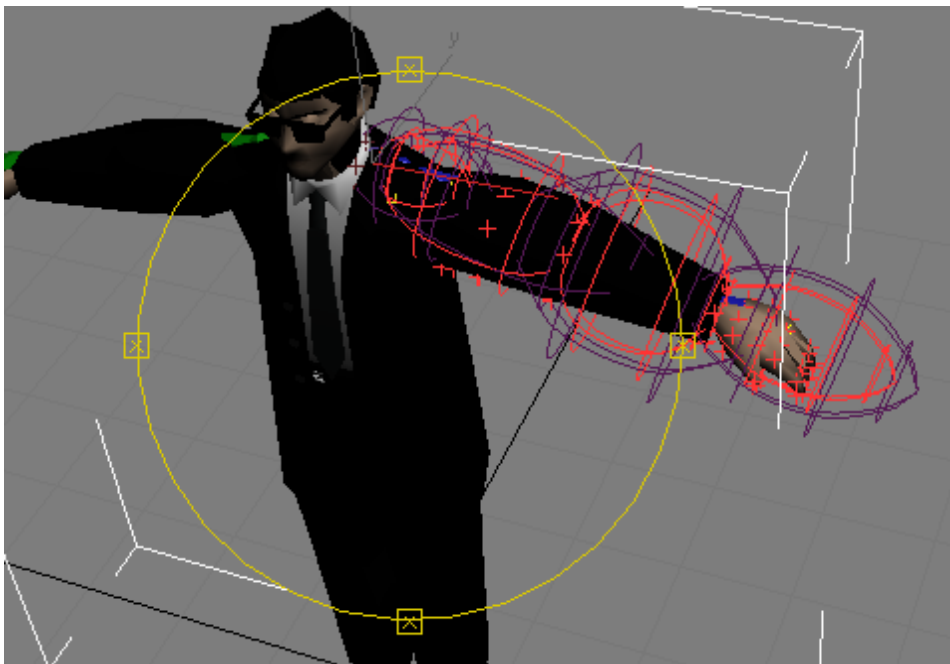


Kuva 27. Vasemman jalkapöydän envelopit.

14. Kun jalka on saatu valmiiksi. Työssä siirrytään virtuaalihahmon vasempaan käteen ja sen olkapäähän(Bip01 L Clavicle) ja valitaan virtuaalihahmon olkapäästä pieni vaikutusalue.

15. (Bip01 L Upperarm) olkavarsi luu kuvassa näkyvillä envelopeilla.

16. (Bip01 L Lower arm) kyynärpää luu kuvassa näkyvillä envelopeilla.



Kuva 28. Vasemman käden envelopit.

Envelopien kopiointi toiselle puolelle

Physique-muokkaimessa on Edit Commands -ryhmä, josta löytyy Copy/paste-toiminto.

1. Valitaan vasemman käden vaikutusalue(bip01 L Hand)
2. Napautetaan Copy
3. Valitaan oikealta puolelta(bip01 R Hand) ja napautetaan Paste
4. Testataan käden liikettä.
5. Näin kopioimalla siirtyy vasemman käden envelopit oikeaan käteen. Selkärangalle kopiointia ei tarvitse.

Kun Virtuaali- hahmo on valmis. Työn seuraava vaihe on vienti Virtoolsiin. Hahmon vienti onnistuu 3Ds Maxissa Export työkalun kautta.

Virtools Exporter –ikkuna

1. Valitaan hahmo näpäytetään ctrl + A näppäimistöltä, joka valitsee kaikki objektit 3Ds Maxin scenestä.
2. 3Ds Maxin Työkaluvalikosta (File – Export)
3. Valitaan työkansio ja nimi hahmolle
4. Tallennusformaatiksi Virtools Export(*.NMO,*CMO.*VMO) ja napautetaan tallenna.
5. Valitaan Export as A Character ja annetaan hahmolle nimi.

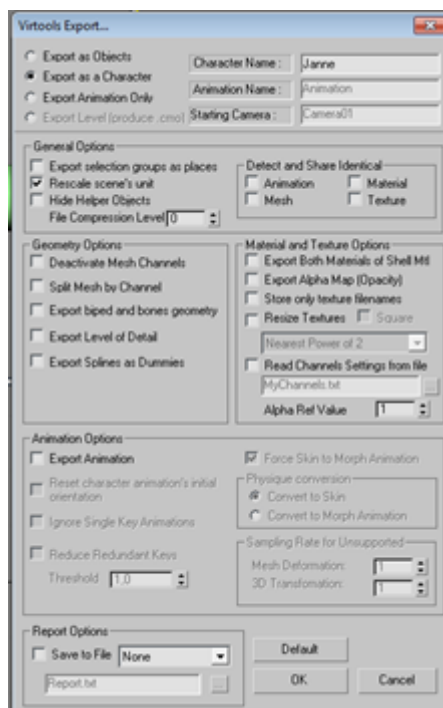
Kaikki monimutkaiset 3ds Max -mallit kannattaa tuoda hahmona virtoolsiin. Objektina tuomisessa on se hankaluus että Virtools ei tunnista kaikkia 3ds Maxissa luotuja ominaisuuksia. Export as a Character -ominaisuus tuo luuranko hierarkian Virtoolsiin. (Virtoolsin Character kansion alla oleva BodyParts Kansio).

ReScale Scene -units ”ruksi” muuttaa virtuaalihahmon mittakaavan Virtools sopivaksi, jos hahmo on mallinnettu käyttäen 3Ds Maxin generic unitsia,

6. Tarkistetaan että Force Skin to Morph animation ”ruksi” on päällä.

7. Force Skin to Morph animation ”ruksin” ansiosta Virtools käyttää virtuaalihahmon geometriaa ihona.

Virtuaalihahmossa käytettiin (Kuvassa 29.) näkyviä Virtools Exporterin -asetuksia.



Kuva 29. Virtools exporter -ikkuna.

LIITE 2: SENSORIEN KÄYTTÖNOTTO

Sensorien käyttöönotto

Virtuaalilaboratorion Cavessa on käytössä ennestään PaikannusPc:hen kytketty Extended Range Controller ja kaksi paikannussensoria päälle ja wand-ohjaimelle. Työssä ohjeistettiin lisäämään sensoreita olemassa olevaan järjestelmään.



Kuva 30. Seitsemän sensorilaitetta ja PaikannusPc.

Extended Range Controller

Sensoriketjun ensimmäinen laitteesta Extended Range Controller on laite, jossa on kiinni paikannuksen lähettäjä (Extended Range Transmitter). Näiden avulla paikannuslaitteiston mittauskapasiteetti kasvaa.

Järjestelmän ensimmäiseen laitteeseen on valmiiksi määritelty tarvittava osoite ja jumpperointi. Laitteeseen tulee virta ja tieto kulkee paikannusPc:n välillä. Tätä ei tarvitse siis muuttaa ollenkaan, Tässä esimerkkinä toimivan Extended range Controllerin osoite asetukset.



Kuva 31. Extended Range Controller.

Osoite asetukset

Osoite määritetään koneen kyljessä olevasta kahdeksan kytkimisestä osoite paneelista. Osoitteen määrittämiseen käytetään binäärilukuja. Laitteen Binääri- luvuissa 1 on alas (päällä) asennossa ja 0 on ylhäällä (pois käytöstä).

Osoitekytkimen kahdeksasta kytkimestä ensimmäiset kolme kytkintä täytyy olla ala asennossa. Nämä kolme kytkintä määrittävät baudin rate (baudi) tässä työssä baudina käytetään 1152000 hertsiä.

Seuraavat neljä kytkimillä (4,5,6 ja 7) määrittellään laitteen osoitteen. Extended Range Controller on ensimmäinen laite. Osoitteeksi tulee siis yksi elikkä osoite on binääri-lukuina 0001 ja kytkimen asennot ovat tällöin ylä,ylä,ylä,ala

Viimeinen elikkä kahdeksas kytkin joka osoite kytkimistöissä on tarkoittaa onko laite testitilassa vai ei. Jotta paikannus toimisi, täytyy kahdeksas kytkin olla ylä asennossa eli fly tilassa.

Jumpperointi

Ensimmäisessä ja viimeisessä paikannuslaitteessa täytyy tarkastaa piirilevyn jumpperointi. Tämä sen takia, että paikannusdatan vastaanotto ja lähetys toimisi laitteessa FBB kaapelin kautta. Jumppereista jpr10, jpr14 ja jpr16 paikat täytyy olla extended Range Controllerissa jumpperoitu.

Extenden Range Controllerin kaapelointi

Ensimmäinen paikannuslaite on liitetty paikannusPc:n com1 ja com3 sarjaliikenneporttiin Paikannuslaitteessa on oma virtajohto, joka on liitetty tavalliseen virtapistokkeeseen. Extended Range Controlleriin tulee paikannuksen mittaava laite Extended Range transmitter, joka on kiinni paikannuslaitteen etupuolella.

Ensimmäisestä paikannuslaitteesta lähtee FBB kaapeli, joka menee seuraavaan paikannuslaitteeseen, tällä kaapelilla ketjutetaan laitteet.

Esimerkki seitsemännen sensorin asennuksesta

Seitsemännen sensorin asennus

1. Sammutetaan ensimmäiseksi Extended Range Controllerin sen virtakytkimestä.
2. Seuraavaksi yhdistetään kuudes ja seitsemäs laite FBB kaapelilla ja lisätään seitsemänteen laitteeseen virtajohto.
3. Lisätään paikannussensori laitteen receiver- liittimeen. Sensoripää laitetaan paikkaan jossa ei ole metallia.
4. Avataan paikannussensorin kotelon ja tarkistetaan jumpperointi. Tässä täytyy muistaa, että Ensimmäisessä ja viimeisessä paikannus-sensorilaitteessa on oltava jumpperoitu laitteen piirilevyn jpr10, jpr14 ja jpr16 kohdassa.
5. Seitsemänteen paikannuslaitteeseen määritetään baud rate nopeus, osoite sekä laitteen tila. Paikannuslaitteessa käytetään 1152000 baud rate nopeutta, osoitteeksi määritellään 0007 ja laitteen tila on fly-asennossa.
6. Seitsemännen paikannuslaitteen testauksessa aloittamiseksi tarkastetaan, että kaikki johdot ovat kiinni sekä käynnistää Extended Range Controllerin sen virtakytkimestä.
7. Odotetaan 15 sekuntia ennen kahdeksanteen vaiheeseen siirtymistä.
8. Myös laitteen tila muutetaan standby -tilasta fly -asentoon. Tämän jälkeen laitteen ledi - valon täytyy vilkkua viidesti, jolloin laitteen asetukset ovat kunnossa.
9. Seuraava sensori liitetään. Ohje toistetaan ja seitsemännestä laitteesta otetaan jumpperointi irti ja siirretään kahdeksanteen laitteeseen.

Taulukko 5. Sensorin asennus.

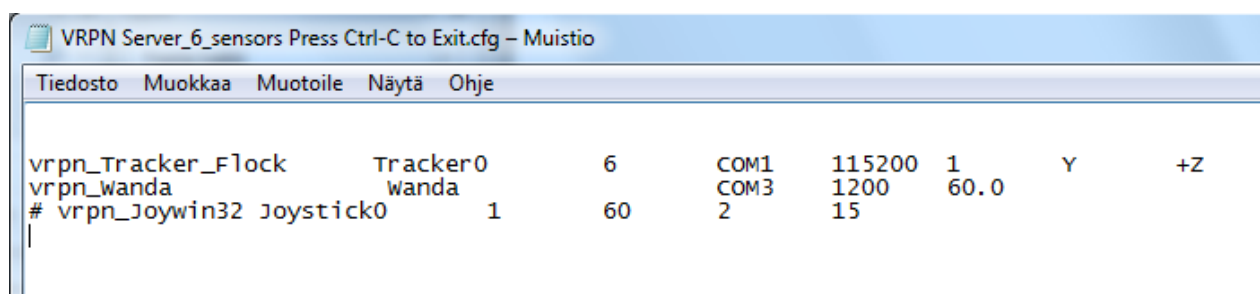
Paikannussensorin asetukset trackerpc koneella

Vrpn on ohjelmisto, jolla hallitaan ja käytetään erilaisia paikannuslaitteistoja. Sillä voidaan määrittää mikä kone käyttää ja kuinka monta paikannusjärjestelmän laitetta on toiminnassa tällä hetkellä.

Vrpn-ohjelmiston muokkaamiseen tarvitaan notepad-tekstinkäsittelytyökalua ja kokemusta config-tiedostojen muokkaamisesta. Paikannuslaitteiston sensori kapasiteetin lisäämisessä on tarkasteltava kahta erilaista Config- tiedostoa.

vrpn.config muuttaminen

vrpn.config tiedoston asetuksista löytyy tiedot mikä paikannusjärjestelmä on valittu, paikannussensorien määrä löytyy täältä, tietokoneen com portti johon tietoa paikannuslaitteilta lähetetään, baudit hertsitaajuus, miten tarkkaa paikannusta näillä laitteilla käytetään, myös koordinaatiston tietoja löytyy täältä.



```

vrpn_Tracker_Flock Tracker0 6 COM1 115200 1 Y +Z
vrpn_wanda wanda
# vrpn_Joywin32 Joystick0 1 60 2 15
|

```

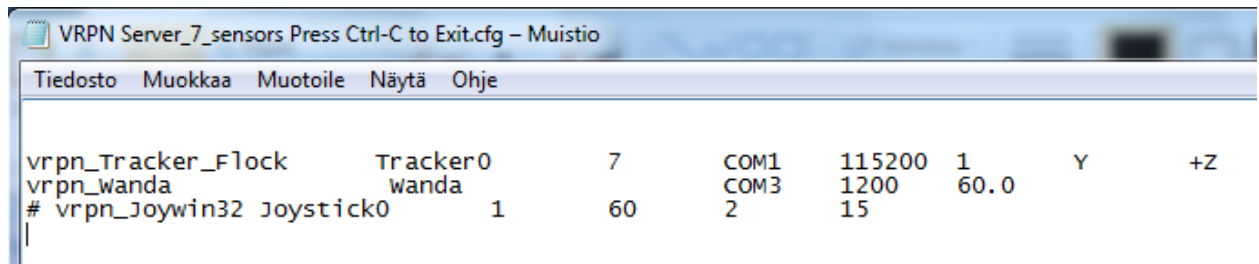
Kuva 32. VRPN Server_6_Sensors.

TrackerPc:n työpöydältä löytyy "VRPN Server_6_sensors - Press Ctrl-C to Exit" pikakuvake, jota ohjeistettiin käyttämään, tämän avulla paikannuslaitteiden lisääminen on yksinkertaista.

Esimerkki sensorin asennuksesta paikannuspc koneelle.

1. Kopioidaan trackerPc:n työpöydällä olevan pikakuvakkeen "VRPN Server_6_sensors Press Ctrl-C to Exit" työpöydältä.
2. Liitetään tiedosto työpöydälle uudella nimellä "VRPN Server_7_sensors PressCtrl-C to Exit".
3. Avataan "VRPN Server_7_sensors - Press Ctrl-C to Exit" pikakuvakkeen ominaisuudet.
4. Muutetaan pikakuvakkeen hakemistopolusta -f "C:\Program Files\VRPN\vrpn_6_sensors.cfg" muotoon vrpn_7_sensors.cfg.
5. Avataan uusi ikkuna C:\Program Files\VRPN\.
6. Kopioidaan vrpn_6_sensors.cfg ja liitetään uudella nimellä vrpn_7_sensors.cfg samaan hakemistoon.
7. Avataan Vrpn_7_sensors.cfg tekstieditorilla.
8. Muutetaan Tracker0 tekstin jälkeen numero kuutonen seiskaksi.
9. Tallennetaan tiedosto.
10. Avataan Vrpn_7_sensors.cfg tekstieditorilla.
11. Muutetaan Tracker0 tekstin jälkeen numero kuutonen seiskaksi.
12. Avataan Vrpn_7_sensors.cfg tekstieditorilla.
13. Muutetaan Tracker0 tekstin jälkeen numero kuutonen seiskaksi.
14. Tallennetaan tiedosto.

Taulukko 6. Sensorin lisääminen paikannusPc:lle



```
VRPN Server_7_sensors Press Ctrl-C to Exit.cfg - Muistio
Tiedosto Muokkaa Muotoile Näytä Ohje
vrpn_Tracker_Flock Tracker0 7 COM1 115200 1 Y +Z
vrpn_wanda wanda COM3 1200 60.0
# vrpn_Joywin32 Joystick0 1 60 2 15
|
```

Kuva 33. Muutettu VRPN Server_7_sensors.cfg

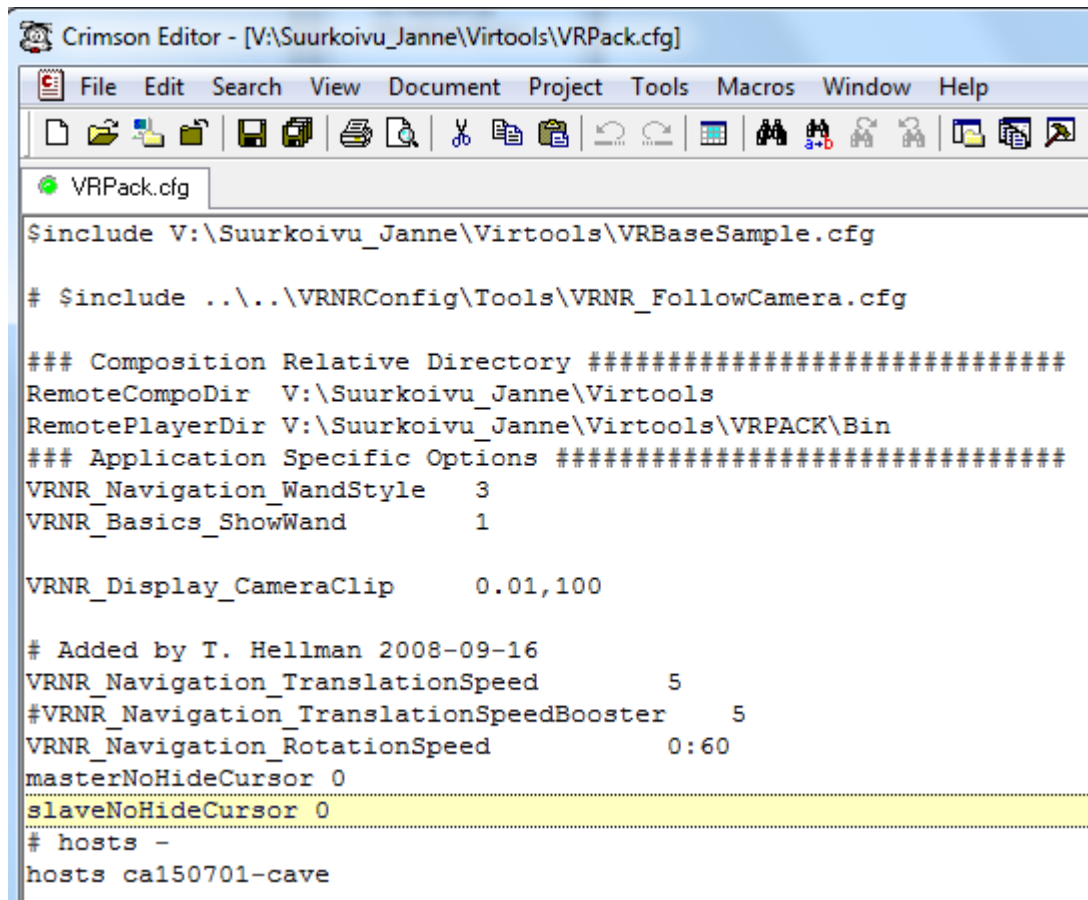
Paikannuksen Lisääminen työkoneelle

Työssä oli vaikeuksia saada järjestelmää toimimaan Virtools esitysohjelman eli Vr-playerin kanssa cave-ympäristössä. Ohjeistettiin selvittämään kuinka VR-player saadaan toimimaan seitsemän paikannussensorin kanssa. Tämä vaihe tehtiin virtuaalihahmon testauksen kannalta. Tämä osio työstä mahdollisti Cave-järjestelmän muun käytön.

Esimerkki paikannuksen lisäämisestä työkoneelle

1. Luodaan työkansio V:\Oma_nimi\Virtools\
2. Virtools-kansion sisälle \VRPack kansion
3. Kopioidaan kohdissa 1 ja 2 luotuun VRPack kansioon Y:\Virtools\VRPack kansios-
ta Bin,Calibration,DeviceServer,Logs,System Con-
fig,ThirdParty,VrAddons,Vrnr,VrnrConfig,VrnrRemote kansiot ja tiedostot niistä
4. Kopioidaan työkansioon VRNR.rsc resurssitiedosto.
5. Virtools-työkansioon kopioidaan VRPack.cfg tiedosto
6. VRPack.cfg tiedosto täytyy olla V:\Oma_nimi\Virtools\ kansiossa (työssä käytetty
hakemistopolkua V:\Suurkoivu janne\virtools)
7. Avataan VRPack.cfg tekstieditorilla.

Taulukko 7. Sensorin lisääminen työkoneelle.



```
Crimson Editor - [V:\Suurkoivu_Janne\Virtools\VRPack.cfg]
File Edit Search View Document Project Tools Macros Window Help
VRPack.cfg
$include V:\Suurkoivu_Janne\Virtools\VRBaseSample.cfg
# $include ../../VRNRConfig/Tools/VRNR_FollowCamera.cfg
### Composition Relative Directory ###
RemoteCompoDir V:\Suurkoivu_Janne\Virtools
RemotePlayerDir V:\Suurkoivu_Janne\Virtools\VRPACK\Bin
### Application Specific Options ###
VRNR_Navigation_WandStyle 3
VRNR_Basics_ShowWand 1
VRNR_Display_CameraClip 0.01,100
# Added by T. Hellman 2008-09-16
VRNR_Navigation_TranslationSpeed 5
#VRNR_Navigation_TranslationSpeedBooster 5
VRNR_Navigation_RotationSpeed 0:60
masterNoHideCursor 0
slaveNoHideCursor 0
# hosts -
hosts cal50701-cave
```

Kuva 34. VRPack.cfg tiedoston muuttaminen.

VRPack Tiedoston muokkaus

1. Muutetaan RemoteCompodir hakemistopoluksi V:\Oma_nimi \Virtools (työssä käytettiin V:\Suurkoivu janne\virtools)
2. Osoite määrittää paikan mistä Vnr hakee virtoolsin paikannuskokoonpanotiedoston. Tämä on tärkeää määrittää oikein, sillä jos tallettaa väärään paikkaan kokoonpanotiedostosi, ei paikannus toimi niissä.
3. Muutetaan RemotePlayerDir hakemistopoluksi v:\Oma_nimi\Virtools\VRPack\Bin (työssä käytettiin V:\Suurkoivu janne\virtools\VRPACK\Bin)
4. Kopioidaan hakemistopolku V:\Oma_nimi\Virtools\VRBaseSample.cfg
5. Avataan VRBaseSample.cfg uuteen tekstinkäsittely ikkunaan.

Taulukko 8. VrPack.cfg tiedoston muokkaus.

VRBaseSample:n muokkaaminen

11/17

1. Ensimmäinen huomioitava asia on BaseDirector. Siihen täytyy muuttaa hakemistopolku V:\Oma_nimi\Virtools (työssä käytettiin V:Janne Suurkoivu\virtools)
2. Etsitään VRBaseSample.cfg tiedostosta `### VRNR Display settings: choose one and edit the file to configure ###`

Varmistetaan, että VRNR_Display_Cube.cfg on seuraavanlaisessa muodossa. `$include V:\Oma_nimi \Virtools\VRPack\VRNRConfig\Display\VRNR_Display_Cube.cfg`

3. Etsitään tiedostosta VRBaseSample.cfg `### VRNR Navigation: choose one and edit the file to configure ###`

Varmistetaan, että VRNR_Navigation_HeadAndWand.cfg on seuraavanlaisessa muodossa. `$include V:\Oma_nimi \Virtools\VRPack\VRNRConfig\Navigation\VRNR_Navigation_HeadAndWand.cfg`

4. Etsitään VRBaseSample.cfg tiedostosta `### VRNR Tools: choose one and edit the file to configure ###`

Varmistetaan, että VRNR_StereoMng.cfg on seuraavanlaisessa muodossa. `$include V:\Oma_nimi \Virtools\VRPack\VRNRConfig\Tools\VRNR_StereoMng.cfg`

5. Etsitään `##### VR Library settings: should not be changed, unless you know what you do`

VrResources kohtaan lisätään hakemistopolku tälle VRNR.resurssitiedostolle. Resurssitiedoston VRNR.RSC paikka on kuvassa selvitetty se tulee VRPack kansioon.

RemotePlayerDir kohtaan tulee hakemistopolku joka johtaa VRPlayer.exe tiedostoon.

6. Etsitään .cfg tiedostosta `### VRNR Navigation: choose one and edit the file to configure ###`

kopioidaan hakemistopolku V:\Oma_nimi

`\Virtools\VRPack\VRNRConfig\Navigation\VRNR_Navigation_HeadAndWand.cfg`

7. Avataan se uuteen tekstieditori-ikkunaan.

Taulukko 9. VRBaseSample.cfg muokkaaminen.

```

### VRNR Display settings: choose one and edit the file to configure ###
#include ..\VRNRConfig\Display\VRNR_Display_Wall.cfg
#include V:\Suurkoivu_Janne\Virtools\VRPack\VRNRConfig\Display\VRNR_Display_Cube.cfg
#include ..\VRNRConfig\Display\VRNR_Display_Cylinder.cfg
#include ..\VRNRConfig\Display\VRNR_Display_Sensics.cfg
#include ..\VRNRConfig\Display\VRNR_Display_Holostage.cfg
#include ..\VRNRConfig\Display\VRNR_Display_Geode_Stereo.cfg
#include ..\VRNRConfig\Display\VRNR_Display_HMD_eMagin_2800.cfg
#include ..\VRNRConfig\Display\VRNR_Display_Generic_HMD_1_Input.cfg
#include ..\VRNRConfig\Display\VRNR_Display_Generic_HMD_2_Inputs.cfg
#include ..\VRNRConfig\Display\VRNR_Display_CRVM.cfg

### VRNR Navigation: choose one and edit the file to configure ###
#include V:\Suurkoivu_Janne\Virtools\VRPack\VRNRConfig\Navigation\VRNR_Navigation_HeadAndWand.cfg
#include ..\VRNRConfig\Navigation\VRNR_Navigation_FirstPersonCamera.cfg
#include ..\VRNRConfig\Navigation\VRNR_Navigation_JoystickBasic.cfg
#include ..\VRNRConfig\Navigation\VRNR_Navigation_EasyJoystick.cfg
#include ..\VRNRConfig\Navigation\VRNR_Navigation_EasyJoypad.cfg
#include ..\VRNRConfig\Navigation\VRNR_Navigation_SpaceMouse.cfg

### VRNR TOOLS: choose one and edit the file to configure ###
#include ..\VRNRConfig\Tools\VRNR_AntiAliasingMng.cfg
#include V:\Suurkoivu_Janne\Virtools\VRPack\VRNRConfig\Tools\VRNR_StereoMng.cfg
#include ..\VRNRConfig\Tools\VRNR_LoadingSprite.cfg
#include ..\VRNRConfig\Tools\VRNR_RenderingMng.cfg
#include ..\VRNRConfig\Tools\VRNR_Credits.cfg
#include ..\VRNRConfig\Tools\VRNR_ShowcaseMode.cfg
#include ..\VRNRConfig\Tools\VRNR_ShorcutKeysRemapping.cfg
#include ..\VRNRConfig\Tools\VRNR_Stats.cfg
#include ..\VRNRConfig\Tools\VRNR_FollowCamera.cfg
#include ..\VRNRConfig\Tools\VRNR_VideoSprite.cfg
#include ..\VRNRConfig\Tools\VRNR_PreloadTexture.cfg
#include ..\VRNRConfig\Tools\VRNR_eMagin2800_KeepAlive.cfg

```

Kuva 35. VRBaseSample.cfg tiedosto.

```

##### VR Library settings: should not be changed, unless you know what you do

VRResources V:\Suurkoivu_Janne\Virtools\VRPack\VRNR.rsc
RemotePlayerDir V:\Suurkoivu_Janne\Virtools\VRPACK\Bin\VRPlayer.exe
SoftSwapLock 0
# Set to 1 to enable NVidia Swaplock. Be sure to disable SoftSwaplock
NVSwaplock 1
CloseConsole 0
ForceIgnoreAspectRatio 1

```

Kuva 36. VRBaseSample.cfg tiedosto.

```

VRNR_Navigation      VRNR_Navigation_HeadAndWand

# Choose tracking system
#include      V:\Suurkoivu_Janne\Virtools\VRPACK\VRNRConfig\Navigation\TrackingSystem\VRNR_Tracking_VRPN.cfg
#include      TrackingSystem\VRNR_Tracking_Vicon.cfg
#include      TrackingSystem\VRNR_Tracking_BirdTracker.cfg
#include      TrackingSystem\VRNR_Tracking_BirdMotionStar.cfg
#include      TrackingSystem\VRNR_Tracking_MotionAnalysis.cfg
#include      TrackingSystem\VRNR_Tracking_eMagin_2800.cfg
#include      TrackingSystem\VRNR_Tracking_XSens.cfg

# Choose controller system
#include      V:\Suurkoivu_Janne\Virtools\VRPACK\VRNRConfig\Navigation\ControllerSystem\VRNR_Controller_VRPN.cfg
#include      ControllerSystem\VRNR_Controller_Win32Joystick.cfg
#include      ControllerSystem\VRNR_Controller_XInputController.cfg

```

Kuva 37. VRNR_Navigation_HeadAndWand.cfg.

VRNR_Navigation_HeadAndWand.cfg tiedoston muokkaaminen

Choose Tracking system

1. Pidetään huoli että VRNR_Tracking_VRPN.cfg tiedoston hakupolku on esitetty kuvan osoittamalla tavalla. esim.

V:\Oma_nimi\Virtools\VRPack\VRNRconfig\Navigation\trackingsystem\vrnr_tracking_VRPN.cfg.

Tähän VRNR_Tracking_VRPN.cfg tiedostoon lisätään käytössä olevat paikannussensorit.

Choose Controller system

Pidetään huoli että VRNR_controller_VRPN.cfg tiedoston hakupolku on esitetty kuvan 37 osoittamalla tavalla.

Taulukko 10. VRNR_navigation_HeadAndWand.cfg tiedoston muokkaus.

VRNR_Tracking_VRPN.cfg Tiedoston muokkaaminen

Tässä cfg-tiedostossa sijaitsee paikannussensorin tietoja ja täältä niitä voidaan myös muokata notepad-tekstinkäsittely ohjelman avulla. Tähän tiedostoon paikannuslaitteen asentajan tarvitsee vain lisätä muutama rivi, jolloin uusi sensori saadaan käyttöön.

VRNR_Tracking_VRNR.cfg muuttaminen

1. Sensorin lisäämiseksi Avataan uuteen tekstieditoriin tiedosto.VRNR_Tracking_VRPN.cfg.
2. ”Rullataan” .cfg tiedoston loppuun, jolloin tulee vastaan viimeisen paikannussensorin tietoja (tässä tapauksessa kuudennen sensorin tietoja).
3. Kopioidaan yllä olevat kuudennen sensorin tiedot.
4. Liitetään teksti uudelleen ja nimetään seitsemännen sensorin tiedot Kuva 8 mukaan.

Taulukko 11. VRNR_Tracking_VRNR.cfg muuttaminen.


```

VRNR_Tracking_VRPN.cfg - Muistio
Tiedosto Muokkaa Muotoile Näytä Ohje
vrpnTracker_5 Tracker0@1a150705:9500
neutralPosition_5 -0.0268 -2.222 1.246
neutralQuaternion_5 -0.996 0.022 0.012 0.051
axisPermute_5 1 2 0
axisSign_5 -1 -1 -1
trackerScale_5 1

vrpnTracker_6 Tracker0@1a150705:9500
neutralPosition_6 -0.0268 -2.222 1.246
neutralQuaternion_6 -0.996 0.022 0.012 0.051
axisPermute_6 1 2 0
axisSign_6 -1 -1 -1
trackerScale_6 1

vrpnAnalog wanda@1a150705:9500
vrpnButton wanda@1a150705:9500

TrackerGroup_0 T0:0:6 T1:0:6 T2:0:6 T3:0:6 T4:0:6 T5:0:6 T6:0:6

```

Kuva 38. VRNR_Tracking_VRNR.cfg.

```

VRNR_tracking_vrnr.cfg - Muistio
Tiedosto Muokkaa Muotoile Näytä Ohje
trackerScale_5 1

vrpnTracker_6 Tracker0@1a150705:9500
neutralPosition_6 -0.0268 -2.222 1.246
neutralQuaternion_6 -0.996 0.022 0.012 0.051
axisPermute_6 1 2 0
axisSign_6 -1 -1 -1
trackerScale_6 1

vrpnTracker_7 Tracker0@1a150705:9500
neutralPosition_7 -0.0268 -2.222 1.246
neutralQuaternion_7 -0.996 0.022 0.012 0.051
axisPermute_7 1 2 0
axisSign_7 -1 -1 -1
trackerScale_7 1

vrpnAnalog wanda@1a150705:9500
vrpnButton wanda@1a150705:9500

TrackerGroup_0 T0:0:6 T1:0:6 T2:0:6 T3:0:6 T4:0:6 T5:0:6 T6:0:6 T7:0:6

```

Kuva 39. VRNR_tracking_vrnr.cfg - Muutettu käyttämään seitsemän sensoria.

1. VRNR_Tracking_VRPN.Cfg tiedoston lopusta löytyy rivi TrackerGroup_0 T0:0:6 T1:0:6 T2:0:6 T3:0:6 T4:0:6 T5:0:6 T6:0:6 lisätään yllä olevan kuvan mukaan seitsemäs laite. T7:0:6

Paikannuksen koordinaatiston muuttaminen

Työssä ilmeni, että sensorit eivät liikkuneet oikein. Työssä saatiin koordinaatisto toimimaan seuraavalla toimenpiteellä.

Koordinaatiston muuttaminen	
1. Kopioidaan tiedot neutralPosition_0 perästä koordinaatisto arvot	<ul style="list-style-type: none"> • Liitetään tämä koordinaatistoarvo jokaisen sensorin neutralPosition_# koordinaatisto arvon perään.
2. Kopioidaan tiedot neutralQuaternion_0 perästä koordinaatisto arvot	<ul style="list-style-type: none"> • Liitetään tämä koordinaatistoarvo jokaisen sensorin neutralQuaternion_# koordinaatisto arvon perään.
3. Kopioidaan tiedot AxisPermute_0 perästä koordinaatisto arvot	<ul style="list-style-type: none"> • Liitetään tämä koordinaatistoarvo jokaisen sensorin AxisPermute_# koordinaatisto arvon perään.
4. Kopioidaan tiedot Axissign_0 perästä koordinaatisto arvot	<ul style="list-style-type: none"> • Liitetään tämä koordinaatistoarvo jokaisen sensorin Axissign_# koordinaatisto arvon perään.

Taulukko 12. Sensorin koordinaatisto arvojen muuttaminen.

```

VRNR_Navigation_HeadTrackerID      0

#WandTrackerID integer ID from VRPN server. Use -1
VRNR_Navigation_WandTrackerID      1

## Calibration
VRNR_Navigation_Calibration_UseOffset      1
VRNR_Navigation_Calibration_UseAxisPermute 1
VRNR_Navigation_Calibration_UseScale      1

#-0.707 0.707 0.0 -0.1
#neutralPosition_0  -0.127 -0.56 -0.27
vrpnTracker_0      Tracker0@1a150705:9500
neutralPosition_0  -0.122 -2.218 1.572
neutralQuaternion_0 0.707 -0.707 0.0 0.0
axisPermute_0      1 2 0
axisSign_0         -1 -1 -1
trackerScale_0     1

vrpnTracker_1      Tracker0@1a150705:9500
neutralPosition_1  -0.122 -2.218 1.572
neutralQuaternion_1 0.707 -0.707 0.0 0.0
axisPermute_1      1 2 0
axisSign_1         -1 -1 -1
trackerScale_1     1

vrpnTracker_2      Tracker0@1a150705:9500
neutralPosition_2  -0.122 -2.218 1.572
neutralQuaternion_2 0.707 -0.707 0.0 0.0
axisPermute_2      1 2 0
axisSign_2         -1 -1 -1
trackerScale_2     1

vrpnTracker_3      Tracker0@1a150705:9500
neutralPosition_3  -0.122 -2.218 1.572
neutralQuaternion_3 0.707 -0.707 0.0 0.0
axisPermute_3      1 2 0
axisSign_3         -1 -1 -1
trackerScale_3     1

vrpnTracker_4      Tracker0@1a150705:9500
neutralPosition_4  -0.122 -2.218 1.572
neutralQuaternion_4 0.707 -0.707 0.0 0.0
axisPermute_4      1 2 0
axisSign_4         -1 -1 -1
trackerScale_4     1

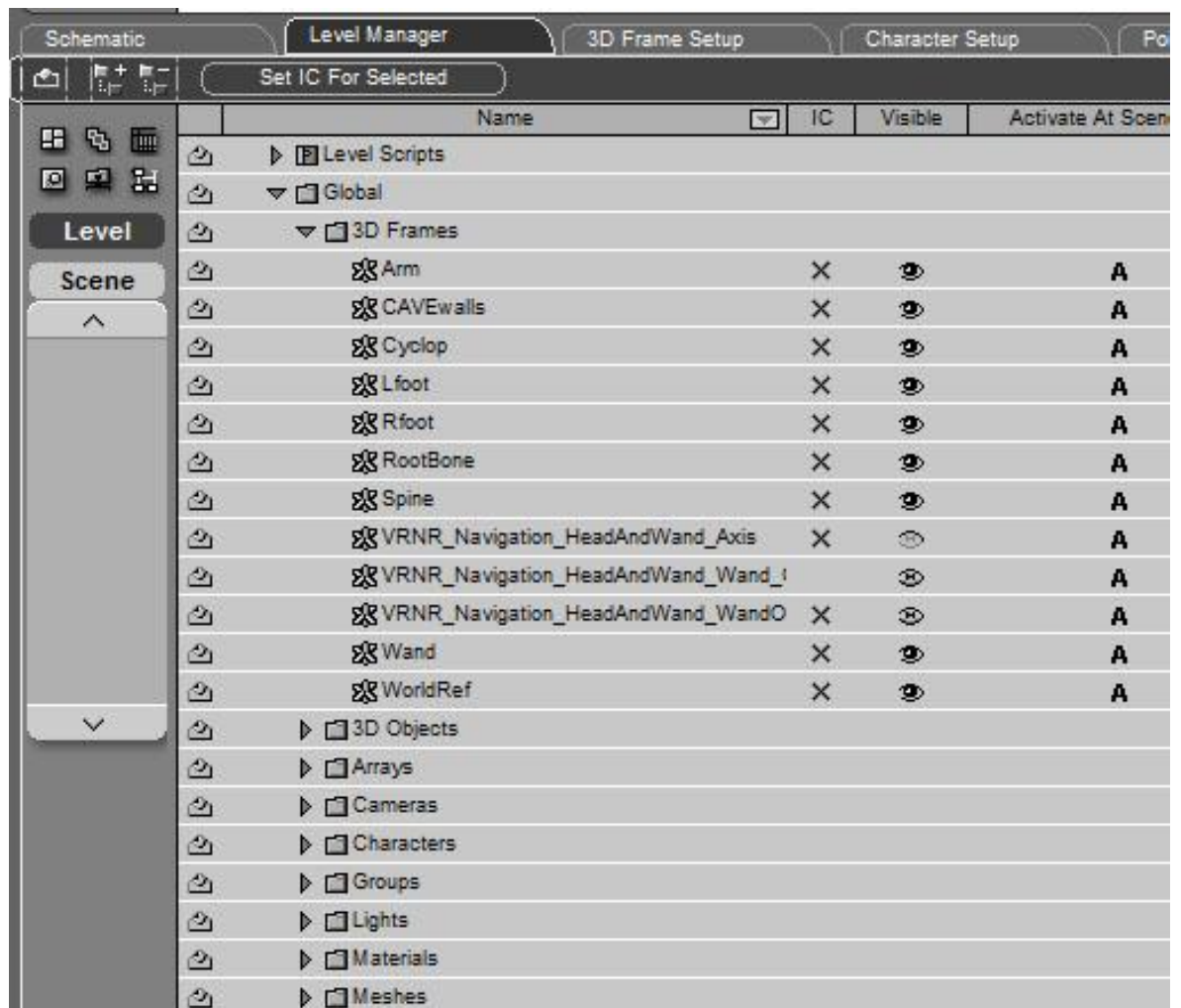
```

Kuva 40. Koordinaatisto arvot jotka toimi.

LIITE 3: SENSORIEN KÄYTTÖÖNOTTO VIRTOOLS- KOKOONPANO-TIEDOSTOSSA

Johdanto

Paikannuslaitteen lisääminen tapahtui malli tiedostona käytetyn head and wand kokoonpanotiedoston sisälle. Tässä osiossa selvitetään mitä vaaditaan yhden sensorin lisäämiseksi Virtools-kokoonpanotiedostoon.



Kuva 41. Virtools:n 3D Frames -valikko.

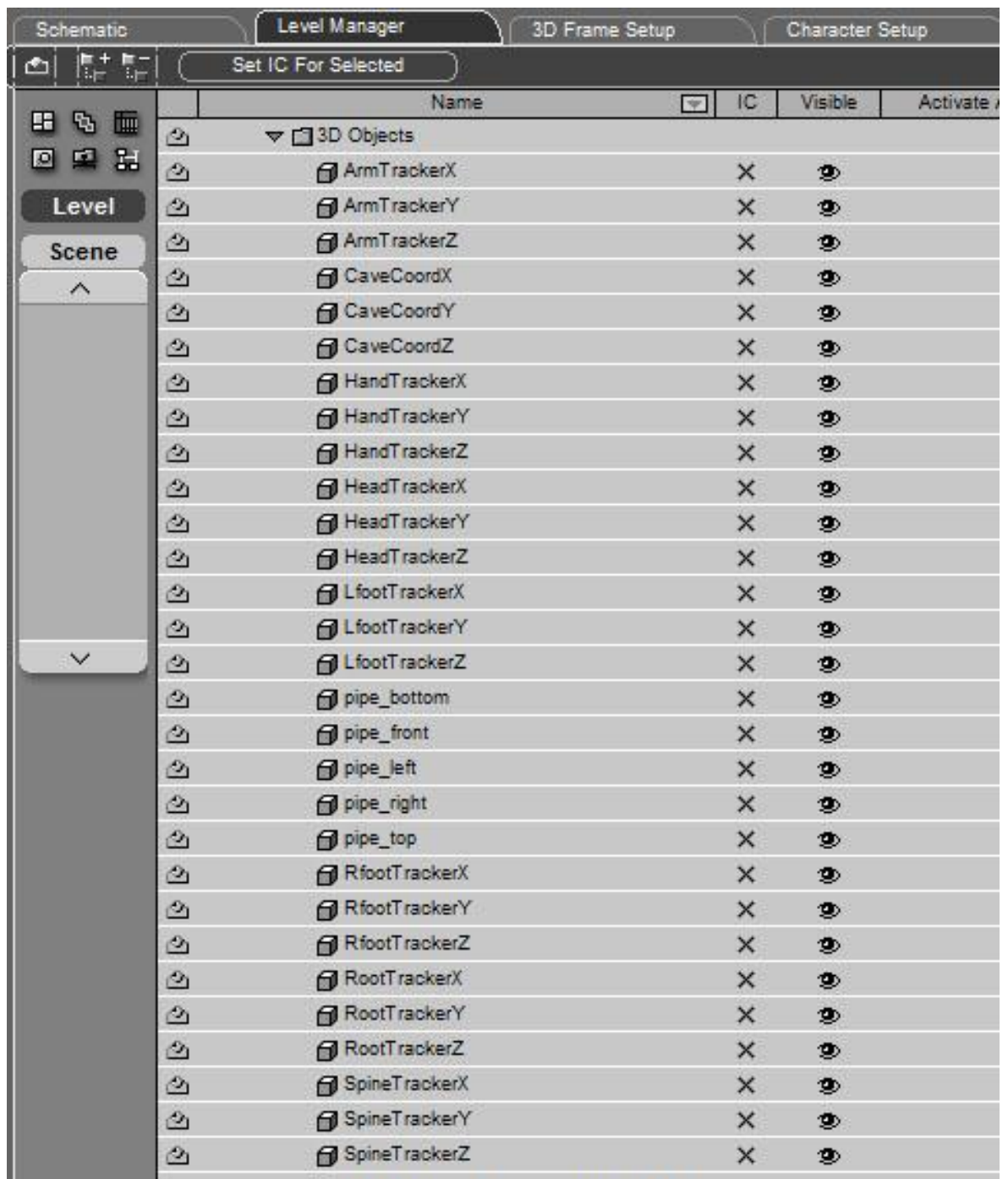
Tässä työvaiheessa tehtiin sensorille visuaalinen objekti. Objekti on virtoolsin 3dframe objekti. Tämän työvaiheen ansiosta sensorin liikettä pystytään havainnoimaan virtools-kokoonpanotiedostossa.

Level manager/3Dframes

- uusi 3dframe
- 3Dframe asetettiin uusi nimi "sensoria kuvaava nimi" esim. spine
- "Initial Condition -ominaisuus"

Esimerkki x,y,z objektien luonnista

Uuden sensorin asennuksen yhteydessä täytyy tehdä sensorille myös x,y,z objektit ja tämä onnistuu level manager/3dobjects valikossa. Työvaiheessa voi käyttää myös olemassa olevia x,y,z arvoja.



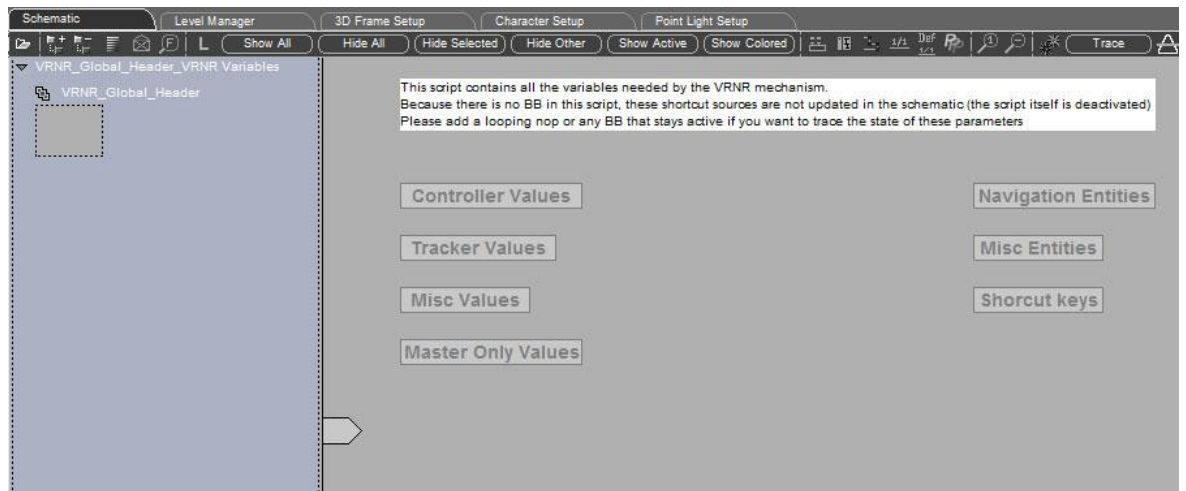
Kuva 42. Virtoolsin 3D Objects -valikko.

<p>Level manager/3DObjects</p>
<ol style="list-style-type: none"> 1. ArmTrackerx,y,z (kolme objektia Level Managerin 3D Objects osiosta) 2. Kopioidaan objektit 3. Liitetään objektit uudestaan. "ilmoitusikkunaan" "No dependencies". 4. Muutetaan nimi kuvaavaksi esim. <ul style="list-style-type: none"> • SpineTrackerx • SpineTrackery • SpineTrackerZ
<p>Hierarchy manager/Root objects</p>
<ol style="list-style-type: none"> 1. Hierarchy Managerista valitaan tässä työvaiheessa spineTrackerx,y,z objektit ja siirretään ne Spine 3Dframe objektin hierarkiaan. 2. Objekteille Set intial Condition -ominaisuus

Taulukko 14. 3D objektin luonti ja hierarkiaan siirtäminen.

Linkkiobjektin lisääminen VRNR_Global_Header_VRNR Variables skriptiin

Linkkiparametri välittää tietoa eteenpäin VRNR_Global_Header_VRNR Variables skriptin kautta. Tässä Skriptissä löytyy paljon erilaisia linkkiparametrejä, joita muuttamalla asetukset muuttuvat. Paikannussensorin lisäämiseksi tänne linkkikokoelmaan täytyy lisätä erilaisia tietoja ainakin kolmeen eri osastoon. Näitä osastoja ovat: Tracker values; Master Only Values; Navigation enties



Kuva 43. VRNR_Global_Header_VRNR Variables skriptiin

Tracker values BG
Position parametri
<ol style="list-style-type: none"> 1. Kuvaava nimi "esim. Esim. VRNR_SpinePos" 2. Parametrin tyypiksi "Vector" 3. Laajennetaan Parametri näyttämään nimen lisäksi arvot
Quaternion Parametri
<ol style="list-style-type: none"> 1. Kuvaava nimi "esim. Esim. VRNR_SpineQuat" 2. Parametrin tyypiksi "Quaternion" 3. Laajennetaan Parametri näyttämään nimen lisäksi arvot

Taulukko 15. Parametrien position ja quaternion luonti tracker BG:hen.

Master Only Values BG

1. Kuvaava nimi "esim. Esim. VRNR_UseSpineTracker"
2. Parametrin tyypiksi "Boolean"
3. Lisätään "booleaniin" ruksi
4. Laajennetaan Parametri näyttämään nimen lisäksi arvot

Taulukko 16. Master Only Values Behavior BG:n lisääminen.

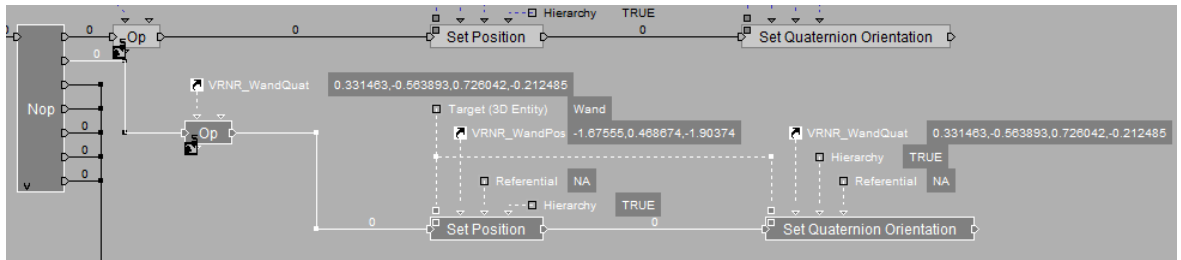
Navigation Entities BG

1. Kuvaava nimi "esim. Esim. VRNR_Navigation_NavSpine"
2. Parametrin tyypiksi "3D Entity"
3. Laajennetaan Parametri näyttämään nimen lisäksi arvot

Taulukko 17. Parametrien luonti Navigation Entities BG:hen.

Tracking skriptissä

Paikannussensorin asema(position) ja Quaternion(kvaternion) arvo lasketaan tracking skriptissä.



Kuva 44. Tracking skriptin muokkaaminen.

Op, Set Position ja Set Quaternion Rakennuspalikan luonti tracking skriptiin.

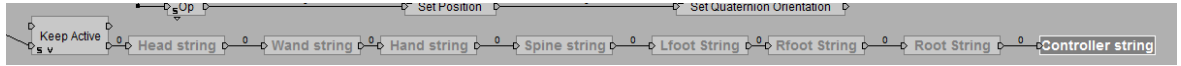
1. Lisätään Nop rakennuspalikkaan uusi ulostulo.
2. Kopioidaan olemassa olevasta linkitetystä ketjusta Op, Set Position ja Set Quaternion Orientation bb:t + parametrit jotka on liitetty siihen.
3. Liitetään(Paste) ne takaisin tyhjälle alueelle.
4. Linkitetään Nop rakennuspalikan kanssa.
5. 3D target parametriin liitetään 3d frame joka on tehty kyseiselle sensorille. Esimerkkinä spine 3Dframe
6. Poistetaan muut uudet parametrit.

Position ja Quaternion parametrien tuonti tracker skriptiin

1. Liitetään ja linkitetään Tracker values Bgstä Position parametri tracker values skriptin set Position ulostuloon.
2. Liitetään ja linkitetään Tracker values Bgstä Quaternion parametri tracker values skriptin set Quaternion ulostuloon.

Taulukko 18. Tracking skriptin koodaaminen.

Uuden String Bg:n luonti



Kuva 45. String Behavior Graph -ketju.

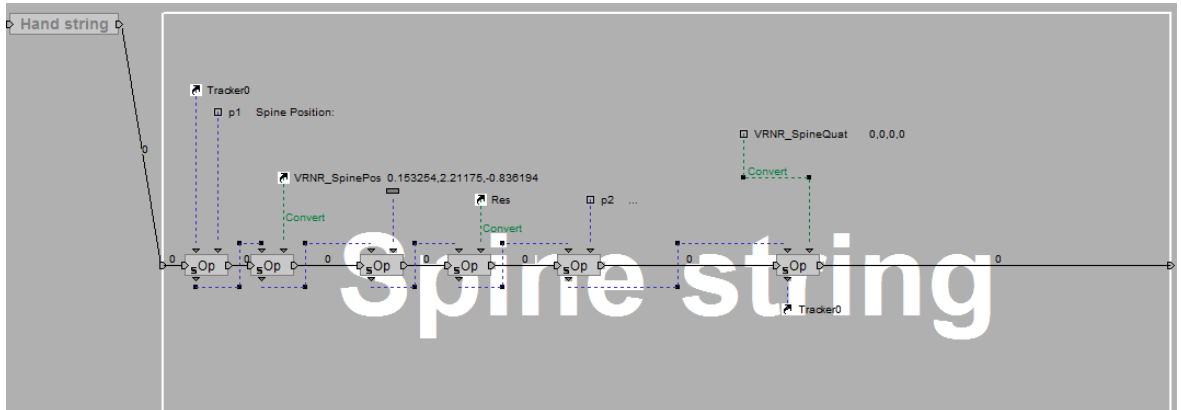
String BG luonti tracking skriptiin

1. Valitaan olemassa oleva esim. "Hand String BG" kopioidaan ja liitetään se tyhjiin tilaan. Nimetään uudestaan.
2. Linkitetään se kuvan mukaisesti.

String Bg muokkaaminen

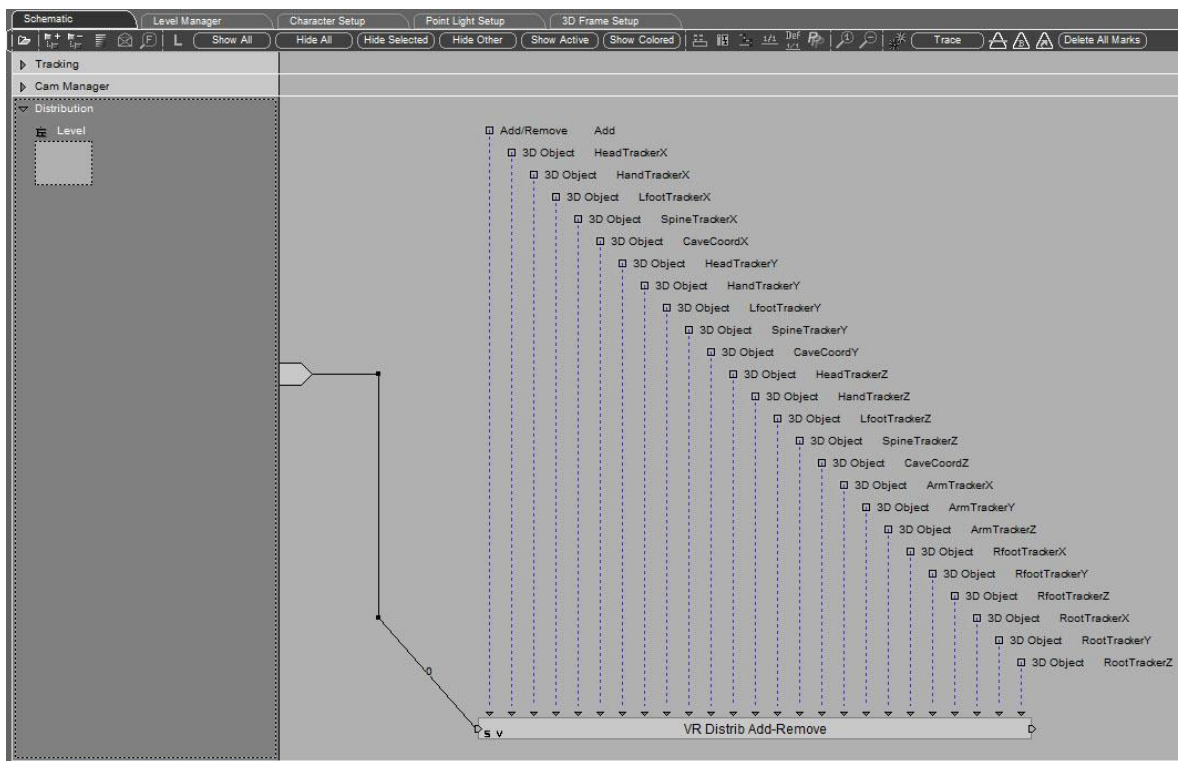
1. Liitetään Tracker values BG position arvo ja liitetään(Paste As Shortcut) se vanhan position parametrin tilalle.
2. Liitetään Tracker values BG Quaternion arvo ja liitetään(Paste As Shortcut) se vanhan Quaternion parametrin tilalle.

Taulukko 19. String BG:n muokkaaminen.



Kuva 46. Avoinna oleva string BG.

Distribution Skripti



Kuva 47. Distribution Skripti.
Tänne määritellään kaikki objektit jotka halutaan näyttää Caven seinillä.

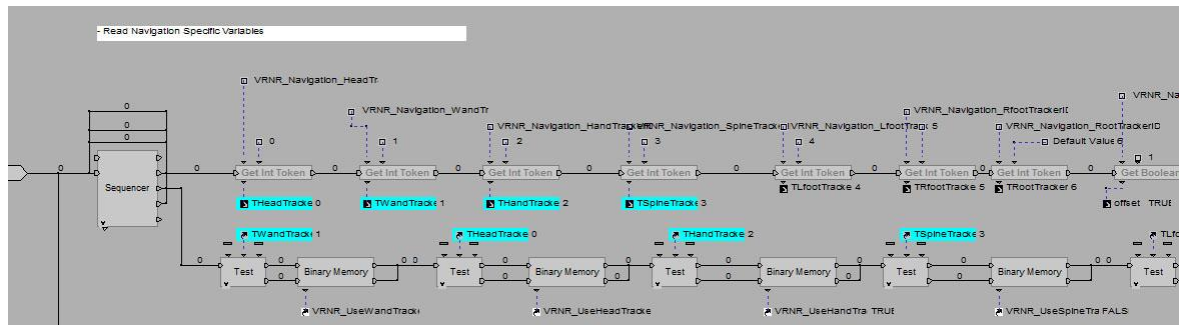
VR Distrib Add-Remove Lisääminen

Ulostuloväylien luonti VR Distrib Add-Remove rakennuspalikkaan

1. Tehdään kolme sisäänmenoväyliä
2. Ulostuloväylän tyypiksi: 3D objects
3. VR Distrib Add-Remove rakennuspalikasta valitaan kolme olemassa olevaa parametria esim. headx,y,z
4. kopioidaan ne tyhjälle alueelle ja annetaan niille uusi 3D objekti esim. spinex,y,z (Uudet 3D objektit Level Managerin 3DObjects osiossa).
5. Linkitetään parametrit VR Distrib Add-Remove rakennuspalikkaan.

Taulukko 20. Vr distrib koodaaminen.

VRNR_Tracking_VRPN_Manager:n muokkaaminen



Kuva 48. VRNR_Tracking_VRPN_manager muokkaaminen.

Get in Token BG

1. Etsitään Skriptistä Get Int Token, jossa on Parametrinä VRNR_Navigation_HandTrackerID
2. Kopiodaan Get Int Token ja liitetään(Paste) se tyhjään skriptin kohtaan
3. Linkitetään uusi Get Int Token olemassa olevan ketjuun ennen Get Boolean Tokenia

Get Int Tokenin Muokkaaminen**Parametrin "VRNR_Navigation_SpineTrackerID"**

1. Parametrin nimeksi: esim. "VRNR_Navigation_SpineTrackerID"
2. Parametrin tyyppi String
3. linkitetään se Get Int Token:n (String) sisäänmenoväylään

Default value parametri

1. Parametrin nimeksi default value
2. Parametrin tyyppi integer
3. Nollaan sensorin seuraava numero
4. Linkitetään se Get Int Token:n (String) ulostuloväylään

Test Ja Binary Memory rakennuspalikka

1. Kopioidaan ja liitetään(Paste) tyhjään tilaan
2. Otetaan mallia aiemmista ja linkitetään ne yhteen

Test Rakennuspalikan muokkaaminen

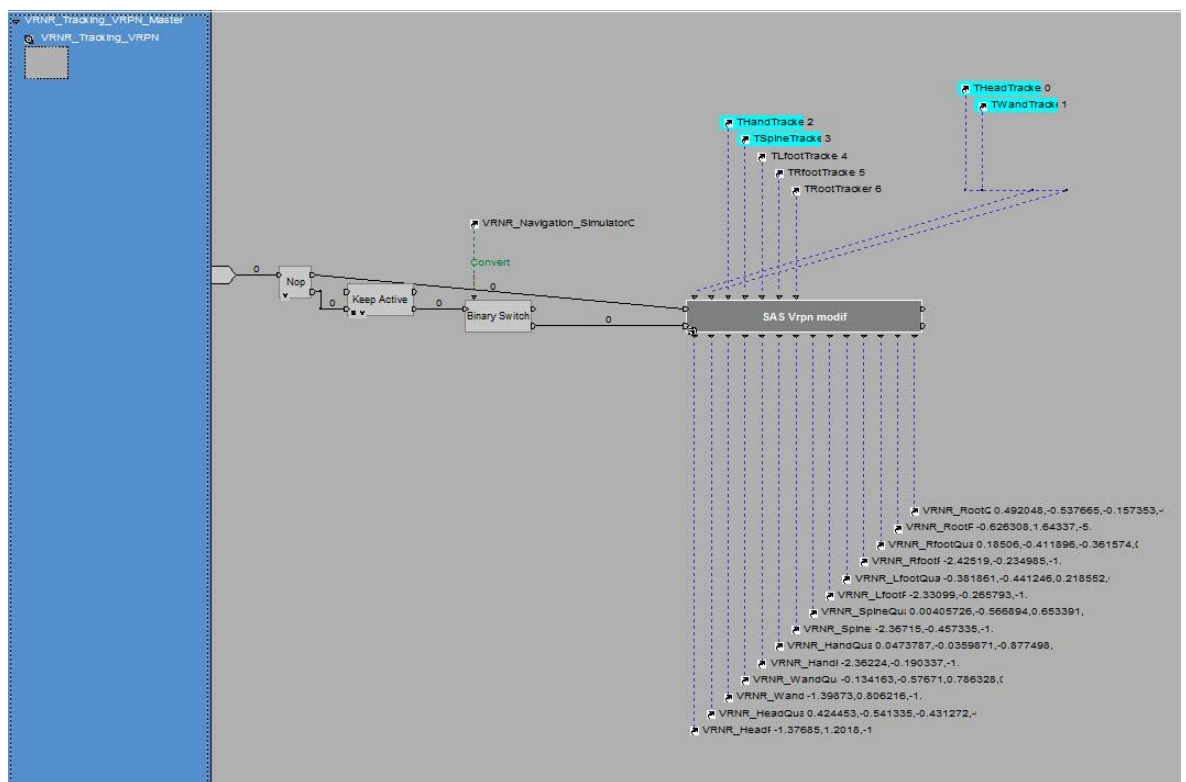
13/21

1. Kopioidaan Get Int Tokenista THandTrackerID 3 Parametri
2. Liitetään se tyhjäan kohtaan (Paste As Shortcut) ja linkitetään Test building Block-kiin A sisäänmenoväylään

Binary Memory Rakennuspalikan muokkaaminen

1. Linkitetään ulostuloväylään VRNR_UseSpineTracker parametri

Taulukko 21. VRNR_Tracking_VRPN_Manager Skriptin koodaaminen.

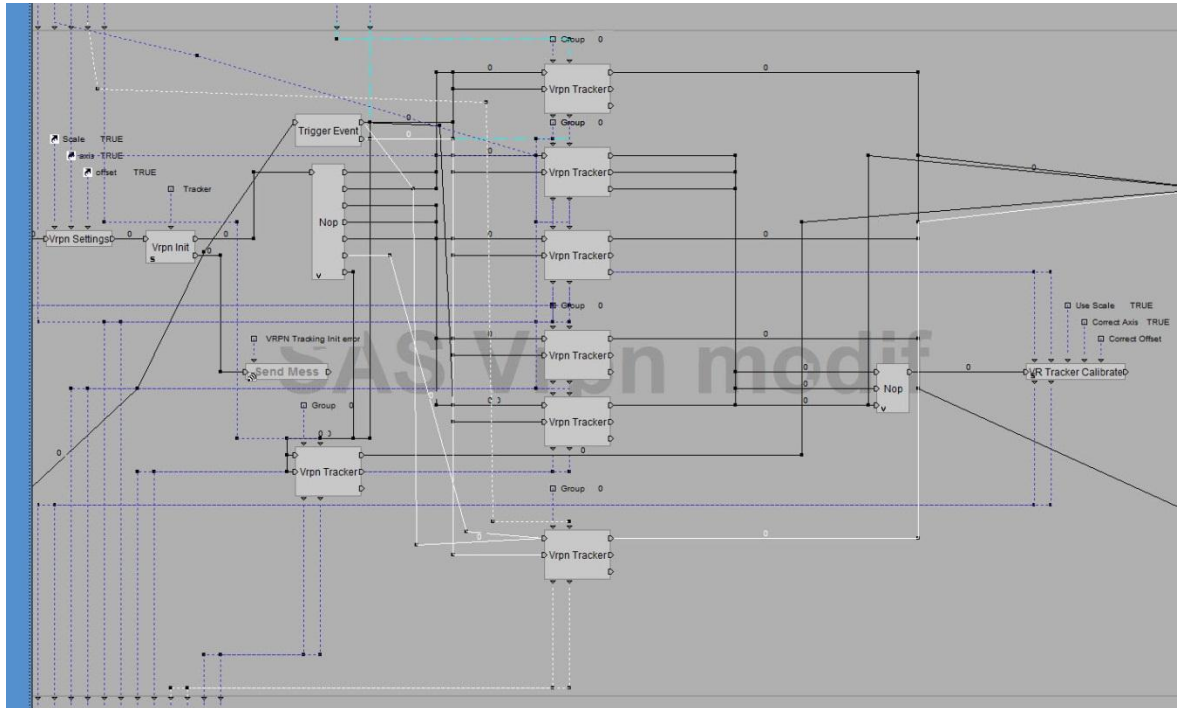
VRNR_Tracking_VRPN_Master Muokkaaminen

Kuva 49. VRNR_Tracking_VRPN_Master Muokkaaminen.

SAS Vrpn modif behaivor Graph
Sisäänmenoväylän parametri
<ol style="list-style-type: none"> 1. Parametrin nimeksi:ID 2. Tyypiksi: Integer 3. Sisäänmenoväylän lisätään TSpineTrackerID(Paste As Shortcut)
Position ulostuloväylä parametri
<ol style="list-style-type: none"> 1. Parametrin nimeksi: VRNR_SpinePos 2. Parametrin tyypiksi: Vector 3. Liitetään/linkitetään(Paste As Shortcut) sensorin position parametrin kanssa ”esim. VRNR_SpinePos”
Quaternion ulostuloväylä parametri
<ol style="list-style-type: none"> 1. Parametrin nimeksi: VRNR_SpineQuat 2. Parametrin tyypiksi: Quaternion 3. Liitetään/linkitetään(Paste As Shortcut) sensorin position parametrin kanssa ”esim. VRNR_SpineQuat”

Taulukko 22. VRNR_Tracking_VRPN_Master koodaaminen.

SAS Vrpn modif muokkaaminen



Kuva 50. SAS Vrpn modif muokkaaminen.

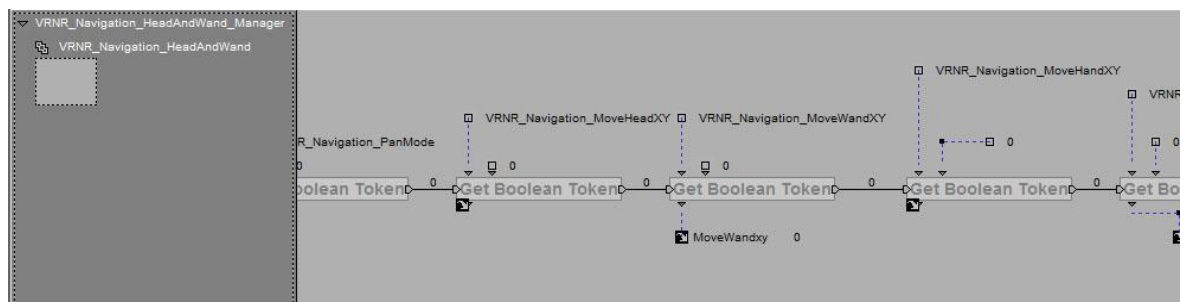
SAS Vrpn modif behavior Graph Muokkaaminen

1. Lisätään uusi Vrpn Tracker rakennuspalikka listasta(VR/Devices)
2. Etsitään Vrpn Init ja siitä seuraava Nop Rakennuspalikka
3. Valitaan Nop rakennuspalikka
4. Linkitetään Trigger Event:n bb:n Active ulostuloväylästä Vrpn Tracker bb:n On sisäänmenoväylään
5. Linkitetään Trigger Event:n bb:n Deactive ulostuloväylästä Vrpn Tracker bb:n Off

sisäänmenoväylään	16/21
6. Linkitetään Vrpn Tracker bb:n Out On ulostuloväylästä SAS Vrpn modif Ok ulostuloväylään.	
Group 0 Parametri	
1. Kopioidaan Group 0 Parametri jostakin Vrpn Tracker bb:stä	
2. Liitetään se (Paste) tyhjäan tilaan ja linkitetään parametri Vrpn Tracker bb:n Group sisäänmenoväylään	
Nop Rakennuspalikan muokkaaminen	
1. Lisätään uusi ulostuloväylä	
2. Linkitetään Nop rakennuspalikan (Out ulostuloväylästä) Vrpn Tracker bb:n On sisäänmenoväylään	

Taulukko 23. SAS Vrpn modif muokkaaminen.

VRNR_Navigation_HeadAndWand_Manager



Kuva 51. VRNR_Navigation_HeadAndWand Skripti.

Get Boolean token muokkaaminen

1. Etsitään Get Boolean Token jossa on Kiinni Parametri VRNR_Navigation_MoveHandXY
2. Kopiodaan olemassa oleva Get Boolean Token ja Default Value 0 Parametri
3. Liitetään ne (Paste) tyhjään tilaan
4. Linkitetään default value tähä Get Boolean tokenin Default value sisään menoväylään.
5. linkitetään Get boolean token olemassa olevaan Boolean Tokenin ketjuun, ennen Get Float Tokenia

Token 0 Parametri

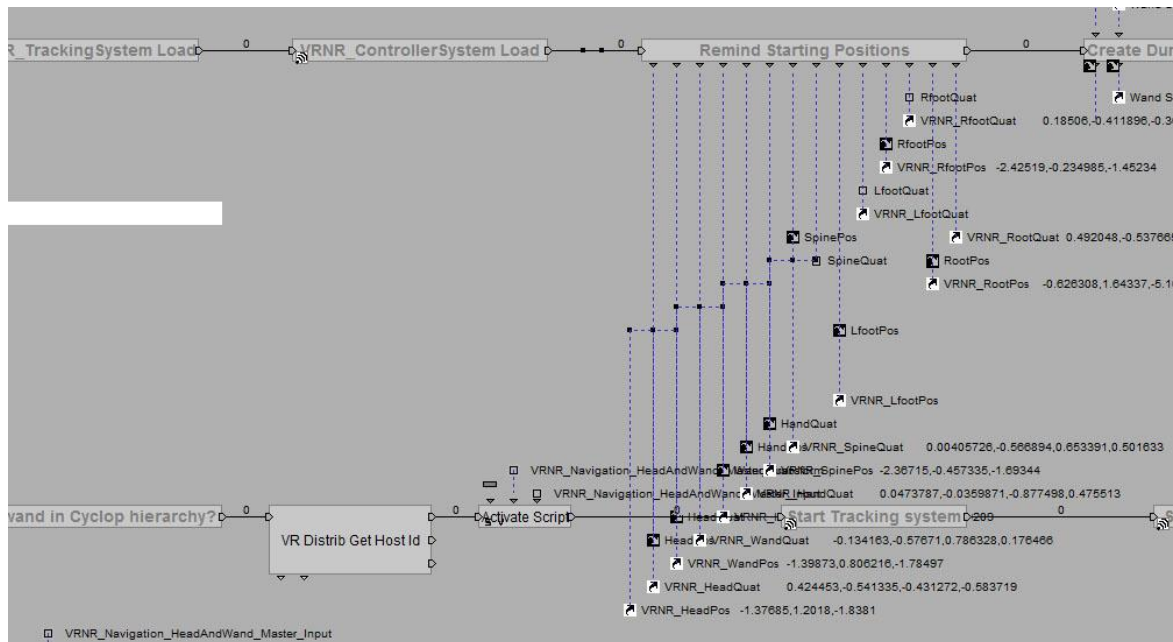
1. Parametrin Nimeksi Token 0
2. Parametrin tyyppi String
3. Viivalle teksti esim. VRNR_Navigation_Move”sensorin nimi”XY
4. Linkitetään Get Boolean Token sisääntuloväylästä Parametri Token 0

MoveSpineXY parametrin luonti

1. Parametri Nameksi tulee MoveSpineXY
2. Parametri Typeksi Integer
3. Linkitetään Get Boolean Token:n Interger ulostuloväylästä Parametri MoveSpineXY

Taulukko 24. VrNR_Navigation_HeadAndWanD skriptin koodaaminen.

Remind starting position

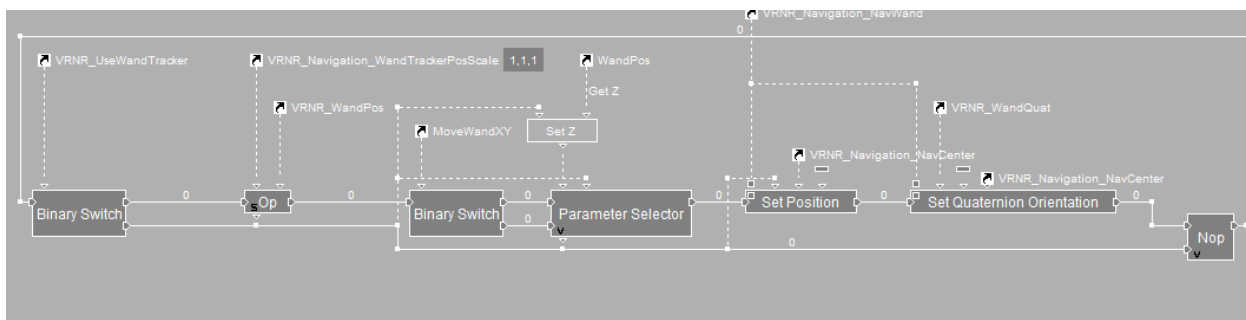


Kuva 52. Remind Starting position Skripti.

Remind Starting position BG
Vector ja Quaternion Ulostulon luonti
<ol style="list-style-type: none"> 1. Luodaan kaksi ulostuloa remind starting position BG:stä 2. Toiselle parametri tyypiksi vector 3. toiselle parametri tyypiksi Quaternion 4. Vector ulostuloparametriin sensorin Position parametrilinkki esim. VRNR_SpinePos (Tracker Values osiosta) 5. Quaternion ulostuloparametriin sensorin Quaternion parametrilinkki esim. VRNR_SpineQuat (Tracker Values osiosta)
Remaind Starting Position Behaivor Graphin muokkaaminen
<ol style="list-style-type: none"> 1. Kopioidaan Behaivor graphista Identity, Set Position, Convert ja Get local matrix 2. Liitetään (Paste) ne uudestaan samaan behaivor graphiin 3. Linkitetään se olemassa olevien rakennuspalikoiden kanssa 4. Haetaan VRNR_Navigation_NavSpine (Navigation Enties osiosta) 5. Liitetään (Paste As Shortcut) ja linkitetään se Get Local Matrix parametri objektin kanssa. 6. Linkitetään Identity bb:n Vector ulostuloväylästä Remind Starting position Vector ulostuloväylään. 7. Linkitetään Linkitetään Identity bb:n Quaternion ulostuloväylästä Remind Starting position Quaternion ulostuloväylään.

Taulukko 25.Remind Starting position koodaaminen.

VRNR_Navigation_HeadAndWand_Master_Transform



Kuva 53. VRNR_Navigation_HeadAndWand_Master_Transform skripti.

Position Head And Wand Behavior Graph

1. Valitaan kuvassa olevat kyseiset building blockit ja parametrit
2. liitetään ne tyhjälle alueelle
3. Poistetaan linkki vanhasta Nop bb:stä ja behavior graphin väliltä
4. Linkitetään olemassa oleva Nop bb ja uusi Binary Switch bb kiinni toisiinsa
5. Linkitetään Nop bb Head And Wand Behavior Graphin kanssa

Binary Switch rakennuspalikka

1. Mennään VRNR_Global_Header_VRNR Variables skript:n (Master Only Values osioon)
2. Kopioidaan sieltä VRNR_UseSpineTracker linkki
3. Liitetään(Paste As Shortcut) ja linkitetään VRNR_UseSpineTracker parametri binary Switch bb:n Sisäänmenöväylän Condition kanssa

Op rakennuspalikka

21/21
<ol style="list-style-type: none"> 1. Mennään VRNR_Global_Header_VRNR Variables skript:n (Tracker Values osioon) 2. Kopioidaan Position Parametri linkki (VRNR_SpinePos) 3. Liitetään(Paste As Shortcut) ja linkitetään se Op bb:n Vector Sisäänmenoväylästä
Op rakennuspalikan jälkeinen Binary Switch rakennuspalikka
<ol style="list-style-type: none"> 1. Haetaan VRNR_Navigation_HeadAndWand_Manager Skriptin Get Boolean tokenista MoveSpineXY linkki tähän 2. Liitetään (Paste As Shortcut) linkki lähelle binary Switch bb:tä 3. linkitetään MoveSpineXY ja binary Switch
Set Z Parametri Objekti
<ol style="list-style-type: none"> 1. Haetaan VRNR_Navigation_HeadAndWand_Manager skriptistä Remind Starting positionista SpinePos linkki 2. Liitetään se (Paste As ShortCut) ja linkitetään Set Z parametriin
Set Position rakennuspalikka
<ol style="list-style-type: none"> 1. Haetaan VRNR_Navigation_NavSpine (Navigation Enties osiosta) 2. Liitetään se (Paste As ShortCut) ja linkitetään Set Position rakennuspalikkaan
Set Quaternion orientation rakennuspalikka
<ol style="list-style-type: none"> 1. Haetaan ja linkitetään siihen VRNR_SpineQuat linkki (Tracking Values osioista)

Taulukko 26 VRNR_Navigation_HeadAndWand_Master_Transform koodaaminen.

LIITE 4: VIRTUAALIIHAHMON LIIKKEEN KOODAAMINEN

Virtuaalihahmon asetukset

Virtuaalihahmon tuonti Virtools-kokoonpanotiedostoon.
2. Virtools-valikkoriviltä Resources
3. Import File As
4. Character
5. Virtuaalihahmo

Taulukko 27. Virtuaalihahmon tuonti Virtools-kokoonpanotiedostoon.

Virtools-kokoonpanotiedoston viimeistely

Valolähteen luonti Virtools -kokoonpanoon
1. Valolähde napautetaan Create light -kuvakkeesta Virtools työkaluriviltä.
2. siirretään valo niin, että ei ole virtuaalihahmon tiellä.
Virtuaalihahmon asetukset level manager välilehdellä
1. Mennään LevelManager välilehdelle
2. Global ja Character kansioon
<ul style="list-style-type: none">• Character kansiossa näkyy ensimmäisenä virtuaalihahmo ja sen nimi. Nimi tulee siitä mitä on 3ds maxin puolella virtools exporter -ikkunassa virtuaalihahmolle antanut.

<p>3. Annetaan Virtuaalihahmolle ominaisuus Set intial Conditions</p> <ul style="list-style-type: none"> • Set Intial Condition on tärkeä ominaisuus, tällä hahmo saadaan palautettua samaan paikkaan kun virtools-kokoonpanotiedostoa on testattu. <p>4. Ilman tätä ominaisuutta virtuaalihahmo ei palautuisi siihen paikkaa mihin se alunperin tuotiin.</p> <p>5. Valitaan kaikki luut Body Parts -kansioista.</p> <p>6. Annetaan Set Intial Conditions ominaisuus myös luille.</p>	2/9
<p>Virtuaalihahmon Skriptin luonti</p>	
<ol style="list-style-type: none"> 1. Character kansiossa virtuaalihahmon nimenpäällä hiiren pikavalikko 2. Create Script 	

Taulukko 28. kokoonpanotiedoston viimeistely.

Virtuaalihahmon liikkeen koodaaminen

Virtuaalihahmon liike koodataan ”virtuaalihahmon nimi skripti alueelle”. Työssä käytettyyn Janne skripti alueelle. Työssä tehdään virtuaalihahmon liikuttaminen vaihe kerrallaan ilman muita sensoreita. Lopuksi kun kaikki sensorit ovat saatu päälle kokeillaan niiden yhteiskäyttöä.

<p>Virtuaalihahmon liikuttaminen</p>
<p>Tämän skriptin tarkoitus on liikuttaa hahmoa virtuaalitodellisuustilan sisällä. Tässä vaiheessa ei puututa raajojen liikkeeseen. Virtuaalihahmon liikuttaminen hoidettiin työssä sensorilla numero 7 Työvaiheesta on myös videomateriaalia. Video löytyy levyltä nimeltä: Virtual_character_move.avi</p>
<p>Keep At Constant Distance -rakennuspalikka</p>

Edit Parameters

Position: X: 0 Y: 0.30 Z: 0

Referential: Rootbone(Sensori 7)

Virtuaalihahmon Pään liikuttaminen

Tässä vaiheessa lisätään Virtuaalihahmon pään liike. Lukuisista yrityksistä huolimatta työssä ei saatu toimimaan pään liikettä oikein. Työssä rakennettiin pään liike toimimaan sensori1 "cyclop" mukaan. Video testauksesta. Pään_liikkeen_ testaaminen.avi

Look At -rakennuspalikka loopataan (Out ulostulosta In sisäänmenoon)**Insert Target Parameters**

Target(3D Entity): Bip01 Head

Edit Parameters

Referential: Cyclop(sensori1) lasit

Edit Settings

Direction: Z

Roll: turn:0 degree: 270

Virtuaalihahmon raajan lisääminen

Tässä käytettiin IkPosition rakennuspalikkaa. Sensoreiden vähyyden vuoksi työssä tehtiin ainoastaan käden raajalloille liike. Tässä käytössä oli sensori 2(wand) ja sensori 3(arm).

IkPosition -rakennuspalikka loopataan (Out ulostulosta In sisäänmenoon)

Oikeakäsi(sensori2 wand)

Edit Parameters

Starting Bodypart: Bip01 R Clavecle

Ending Bodypart: bip01 R Hand

Entity to Follow: Wand

Latency: 5

Vasenkäsi(sensori3 Arm)

Edit Parameters

Starting Bodypart: Bip01 L Clavecle

Ending Bodypart: bip01 L Hand

Entity to Follow: Arm

Latency: 5

Virtuaalihahmon selkärangan liikuttaminen

Koodaaminen tapahtuu kahdella rakennuspalikalla. Ensimmäinen rakennuspalikka "mimic" käskää selkärangan liikkua sinne mihin sensori liikkuu. Toinen rakennuspalikka "KeepAtConstantDistance" pitää selkärangan tietyllä etäisyydellä selkärankaan liikuttavasta sensorista. Lopulta linkitetään skriptit yhteen. Sensori 4 on käytössä tässä.

**Mimic Rakennuspalikka linkitetään Keep at Constant Distance rakennuspalikkaan.
Mimic loopataan (Out ulostulosta In sisäänmenoon)**

Insert Target Parameters

Target(3D Entity): Bip01 Spine2

Edit Parameters

Object: Spine

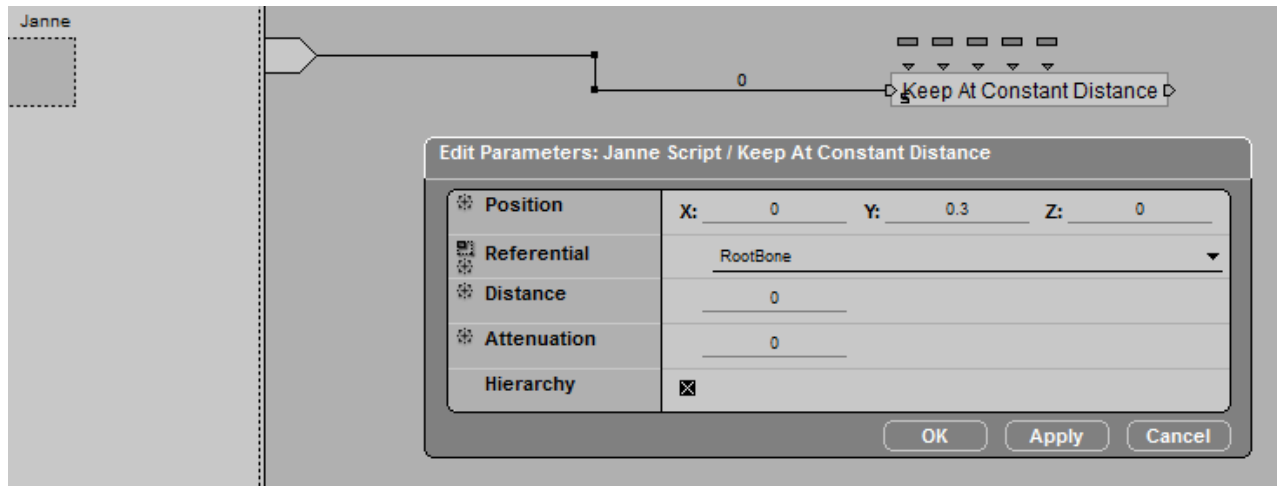
Keep At Constant Distance -rakennuspalikka loopataan (Out ulostulosta In sisäänmenoon)

Edit Parameters

Position: X:-0.3 Y: 0 Z:0.1

Referential: Bip01 Spine2

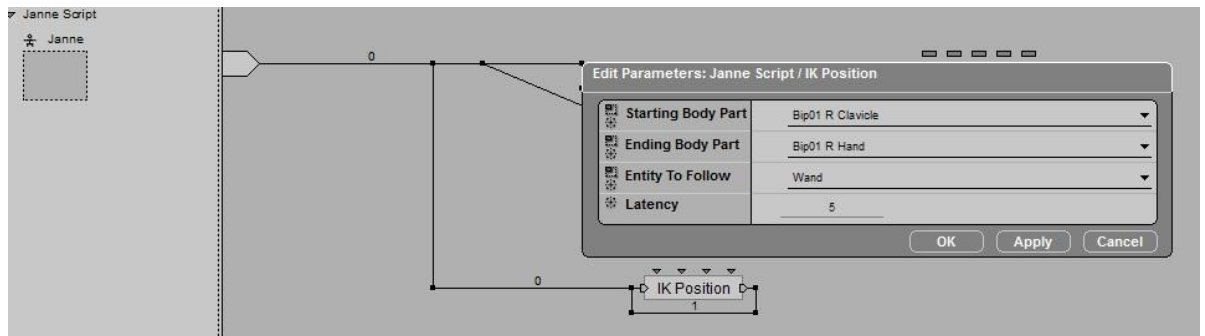
Taulukko 29. Virtuaalihahmon liikkeen koodaaminen.



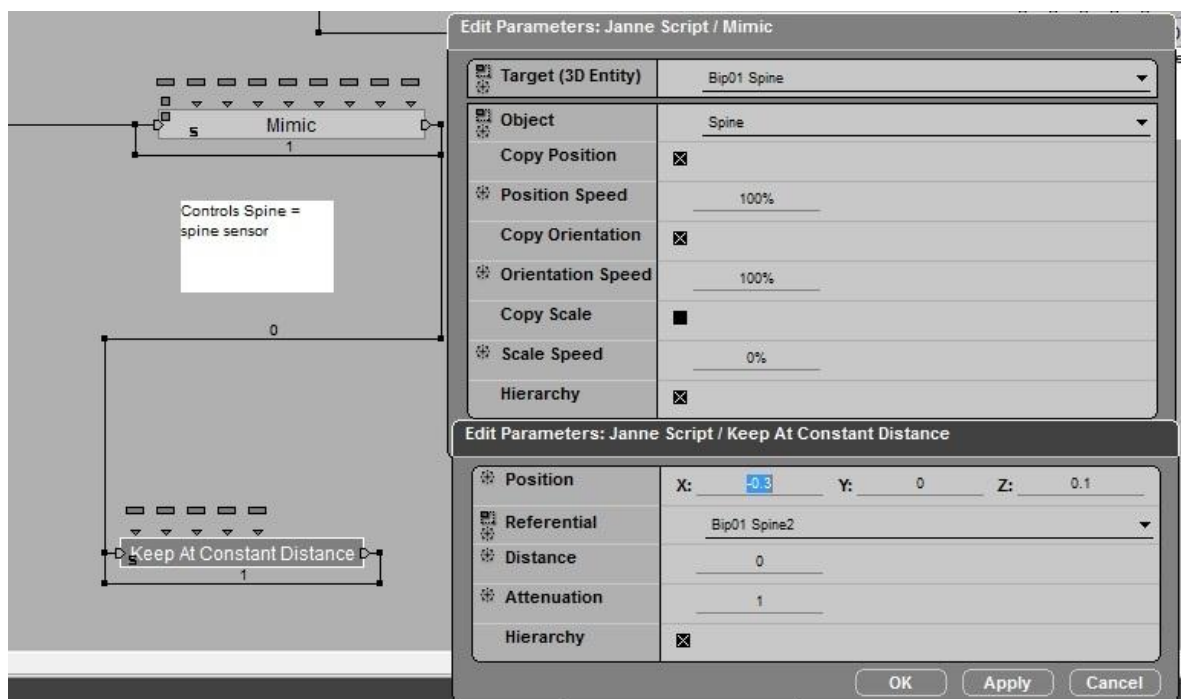
Kuva 54. Koko virtuaalihahmon liikuttaminen.



Kuva 55. Virtuaalihahmon pään koodaaminen.



Kuva 56. Virtuaalihahmon raajan koodaaminen.



Kuva 57. Selkärangan koodaaminen.

