Suorakäyttöinen kaavioeditorisovellus interaktiivisille tussitauluille

Jussi Rantala

Tampereen yliopisto
Tietojenkäsittelytieteiden laitos
Tietojenkäsittelyoppi
Pro gradu -tutkielma
Ohjaaja: Roope Raisamo
Kesäkuu 2007

Tampereen yliopisto
Tietojenkäsittelytieteiden laitos
Tietojenkäsittelyoppi
Jussi Rantala: Suorakäyttöinen kaavioeditorisovellus interaktiivisille tussitauluille
Pro gradu -tutkielma, 87 sivua, 5 liitesivua
Kesäkuu 2007

Interaktiiviset tussitaulut ovat nykyteknologian vastineita perinteisille seinälle kiinnitettäville tauluille. Interaktiivisissa tussitauluissa yhdistyvät suuri fyysinen piirtoala, suorakäyttöisyys ja tietokoneen avulla tapahtuva tietojenkäsittely. Interaktiivisten tussitaulujen avulla voidaan muokata joustavasti taululla näytettävää tietoa. Erilaisten editorisovellusten, kuten kaavioeditorien, käyttö tiedon mallintamiseen on ollut yksi interaktiivisten tussitaulujen yleisimmistä käyttökohteista. Tässä tutkielmassa olen kartoittanut, millä käyttöliittymäratkaisuilla ja vuorovaikutuskeinoilla interaktiivisten tussitaulujen sovelluksia on aikaisemmin pyritty kehittämään. Olen yhdistänyt interaktiivisten tussitaulujen asettamat vaatimukset kaavioeditoreita koskevaan teoriatietoon.

Olen jatkokehittänyt normaaliin työpöytäympäristöön tarkoitettua kaavio-editorisovellusta tukemaan interaktiivisten tussitaulujen käyttöä. Tutkimusongelmana tässä työssä on ollut tutkia, kuinka komentojen antaminen ja tilanhallinta tulisi toteuttaa interaktiivisia tussitauluja käytettäessä. Yleisemmällä tasolla olen kerännyt tietoa siitä, kuinka olemassa olevan sovelluksen muuttaminen tussitaulukäyttöä tukevaksi onnistuu. Teoriakartoituksen pohjalta kaavioeditorisovellukseen toteuttamistani ominaisuuksista keskeisimpiä ovat eleiden avulla annettavat komennot sekä työskentelytilan jakaminen yleis- ja muokkausnäkymään. Uusien ominaisuuksien käyttöä kokeiltiin käytännössä koehenkilöiden avulla toteutetussa testauksessa. Testitulosten perusteella sovelluksen käyttö oli luonnollista ja helppoa, mutta uusien ominaisuuksien tehokkuudessa ja käytettävyydessä oli vielä kehittämisen varaa.

Avainsanat ja -sanonnat: Interaktiiviset tussitaulut, kaavioeditorit, suorakäyttöisyys, kontekstivalikko, eleet, tasaus, skaalautuvuus.

Sisällys

1. Johdanto ... 1
2. Interaktiiviset tussitaulut ... 5
   2.1. Suorakäyttöisyyden käsite ... 5
   2.2. Tussitauluista käytetty terminologia ... 6
   2.3. Tekniset toteutustavat ... 8
   2.4. Kosketusnäytöt interaktiivisiin tussitauluihin verrattuna ... 10
   2.5. Määritelmä interaktiiviselle tussitaululle ... 12
   2.6. Erimalliset interaktiiviset tussitaulut ... 13
      2.6.1. Taulua käyttävät laitteet ... 14
      2.6.2. Siirrettävät vastaanotinlaitteet ... 15
   2.7. Interaktiivisten tussitaulujen käyttökohteita ... 17
      2.7.1. Suunnittelutyö ... 17
      2.7.2. Opetus ... 18
   2.8. Yhteenveto ... 19
3. Interaktiivisten tussitaulujen keskeiset ominaispiirteet ... 21
   3.1. Syötteiden antaminen ... 21
      3.1.1. Painikkeet ja eleet ... 21
      3.1.2. Eleet ja piirakkavalikko ... 22
      3.1.3. Eleet ilman kynää ... 24
   3.2. Syötteiden käsittely ja tunnistaminen ... 25
      3.2.1. Tekstintunnistus ... 26
      3.2.2. Yleisten geometristen kuvioiden tunnistaminen ... 27
      3.2.3. Tunnistus CASE-työkaluissa ... 28
   3.3. Tiedon järjestäminen ... 29
      3.3.1. Käyttäjän määrittelemät rajat ... 30
      3.3.2. Automaattiset lohkot ... 31
   3.4. Virtuaalisen tilan hallinta ... 33
      3.4.1. Zoomaaminen ... 33
      3.4.2. Kerroksisuus ... 35
      3.4.3. Käännettävä taulu -metafora ... 36
   3.5. Yhteenveto ... 37
4. Kaavioeditorit ... 39
   4.1. Graafiset editorit ... 39
   4.2. Kaaviot visuaalisten kielten osana ... 40
   4.3. Kaavioeditoreista yleisesti ... 41
   4.4. Graafiteorian hyödyntäminen kaavioeditoreissa ... 42

4.5. Kaavioesityksen ulkoasukriteerit ja muokkaaminen 43
4.5.1. Esteettiset kriteerit 44
4.5.2. Asettelumenetelmiä ja -työkaluja 45
4.6. Vapaakäyttöiset, syntaktiset ja älykkäät kaavioeditorit 46
4.7. Yhteenveto 47
5. Kaavioeditori interaktiivisille tussitauluille 49
5.1. Toteutusvaiheen lähtökohdat 49
5.2. Yksinkertainen kaavioeditorisovellus 50
5.3. Suunnittelu interaktiivisille tussitauluille 51
5.4. Komentojen ja syötteiden antaminen 53
5.4.1. Kontekstivalikko 53
5.4.2. Eleiden käyttö 54
5.4.3. Tekstinsyöttö ja objektien siirtäminen 56
5.5. Tilanhallinta 57
5.5.1. Yleiskuva- ja muokkausnäkymä 57
5.5.2. Muokkausnäkymän hallitseminen 60
5.6. Yhteenveto 61
6. Sovelluksen testaus 63
6.1. Asetetut tavoitteet 63
6.2. Tehtävät 64
6.3. Testauksen läpivienti 68
6.4. Tulokset 69
6.4.1. Testidatan analysointi 70
6.4.2. Subjektiiviset arviot ja palautteet 73
6.5. Yhteenveto 75
7. Lopuksi 78

Viiteluettelo 80
Liitteet

1. Johdanto

Interaktiiviset tussitaulut (interactive whiteboards) ovat laitteita, jotka korvaavat perinteiset, liidulla tai tussilla käytettävät taulut. Interaktiivisten tussitaulujen avulla pystytään yhdistämään taulun tarjoama suuri fyysinen piirtoala sekä tietokoneen avulla tapahtuva tehokas ja monipuolinen tietojenkäsittely. Laitteiden ohjaaminen perustuu pääsääntöisesti kosketustunnistukseen. Koskettamalla annettu syöte välitetään tietokoneelle käsiteltäväksi, jonka jälkeen syötteen vaikutus voidaan nähdä välittömästi taululla. Käyttäjä voi esimerkiksi piirtää tai kirjoittaa interaktiiviselle tussitaululle kuljettamalla sormea taulun pinnalla. Tämä suora vuorovaikutustekniikka on interaktiivisten tussitaulujen käytössä keskeistä. Erillisiä syötelaitteita ei interaktiivisten tussitaulujen kanssa välttämättä tarvita. Kosketustunnistus ei kuitenkaan ole ainoa tapa syötteiden antamiseen, sillä esimerkiksi kämmentietokoneita [Rekimoto, 1998] voidaan myös käyttää interaktiivisten tussitaulujen ohjauksessa.

Interaktiivisten tussitaulujen hyödyntämismahdollisuudet ovat moninaiset. Yleisiä käyttöskenaarioita ovat erilaiset ryhmämuotoiset tapaamiset. Taulua käytetään jonkin asiakokonaisuuden esittelyyn tai uudelleenjäsentämiseen esitystä muokkaamalla. Interaktiivisia tussitauluja voidaan tämän lisäksi hyödyntää muun muassa opetuksessa [Chong and Sakauchi, 2000; Ovaska *et al.,* 2003], suunnittelutyössä [Chen *et al.,* 2003; Damm *et al.,* 2000] tai henkilökohtaisten tietojen käsittelyssä [Mynatt *et al.,* 1999]. Yhteistä näille kaikille käyttötavoille on se, että ne perustuvat interaktiivisten tussitaulujen mahdollistamaan tiedon joustavaan muokkaamiseen. Interaktiivisia tussitauluja käytettäessä voidaan välittömästi nähdä muutosten vaikutus taululla. Lisäksi koska erillisiä syötelaitteita ei välttämättä tarvita, tuntuu käyttäjästä, kuin muokattava tieto olisi todellisuudessa digitaalisen muodon sijaan taulun pinnalla.

Interaktiivisten tussitaulujen kehitystyö käynnistyi 1990-luvun alkupuolella. Ominaispiirteidensä vuoksi interaktiiviset tussitaulut vaativat perinteisiin pöytätietokoneisiin verrattuna täysin erilaisen näkökulman niin fyysiseen kuin ohjelmalliseenkin toteutukseen. Ensimmäisille laitteille tarkoitettujen sovellusten suunnittelun yhteydessä tuli ilmi perustavia kysymyksiä interaktiivisten tussitaulujen käyttötavoista. Perinteisen graafisen käyttöliittymän suora siirtäminen suurelle tussitaululle ei osoittautunut onnistuneeksi ideaksi [Pedersen *et al.,* 1993]. Ryhmäkokouksien tukemiseen tarkoitetun Tivoli-sovelluksen käyttäminen vaati käyttäjältä kurottelua kohti taulun kulmia, sillä työpöytäkäyttöön tarkoitetuissa sovelluksissa komentoja kuvaavat ikonit on sijoitettu näyttöalan reunoille.

Ensimmäisten interaktiivisten tussitaulujen käyttöön liittyvän tutkimuksen avulla on saatu paljon arvokasta tietoa, jonka perusteella laitteita ja niiden ohjelmistoja on kehitetty käyttäjäystävällisempään suuntaan. Komentojen antamiseksi käyttäjältä ei tule edellyttää merkittävää liikkumista tai kurottelua. Syötteitä voidaan antaa esimerkiksi

eleiden avulla, jolloin erillisiä kuvakkeita komentojen käynnistämiseksi ei tarvita [Mynatt *et al.,* 1999]. Yleisesti tutkimusta on tehty esimerkiksi syötteiden antamisen [Igarashi *et al.,* 2000], virtuaalisen tiedonhallinnan [Moran *et al.,* 1997] ja tussitaulujen käyttötapojen [Mynatt, 1999] parissa. Kuitenkaan interaktiiviset tussitaulut eivät kaikilta osin ole vastanneet niille asetettuihin odotuksiin. Laitteiden hinnat ovat laskeneet järkevälle tasolle, mutta niiden käyttö on edelleen melko rajattua.

Samanaikaisesti interaktiivisten tussitaulujen kehityksen kanssa on tiedon muokkaamista tutkittu työpöytäympäristössä. Suorakäyttöisyyttä tukevien käyttöliittymien kehittyessä on yhä useammilla käyttäjillä ollut mahdollisuus muokata tietoa visuaalisessa muodossa tekstin sijaan. Erilaisten graafisten editorisovellusten yleistyminen onkin ollut tiukasti yhteydessä graafisten käyttöliittymien kehitykseen. Graafiset editorit ovat sovelluksia, jotka esittävät tutut objektit visuaalisessa muodossa ja mahdollistavat esityksen suoran muokkaamisen [Vlissides and Linton, 1990]. Graafisilla editoreilla voidaan esimerkiksi kirjoittaa tekstiä tai muokata kuvia. Useat graafiset editorit eivät kuitenkaan ymmärrä muokattavan esityksen tarkoitusta, vaan ainoastaan tarjoavat keinon ulkoasun muuttamiseen.

Kaavioeditorit (diagram editors) kuuluvat graafisiin editoreihin. Yleisistä graafisista editoreista kaavioeditorit eroavat siinä, että niiden käyttöä on rajoitettu sallimalla ja määrittelemällä tarkoin vain tietynlaisia informaation esitys- ja muokkaustapoja. Kaavioeditoria käyttämällä ei pysty laatimaan mielivaltaisia kuvioita. Lisäksi kaavioeditorit ymmärtävät käsiteltävän informaation merkityksen esitysmuotoa syvemmällä tasolla. Hierarkkisia suhteita kuvaavan kaavion tapauksessa kaavioeditorisovellus ymmärtää, että alemmalla tasolla olevat objektit ovat ylempänä olevan objektin lapsia. Erilaisia kaaviomalleja on olemassa lukematon määrä riippuen siitä, mitä tietoa esitetään. Kaikkia kaavioeditorisovelluksia yhdistävä tekijä on merkitysten antaminen graafisille objekteille, joita käyttäjä voi ennalta määritellyin vuorovaikutustavoin muokata.

Interaktiivisten tussitaulujen ja kaavioeditorien yhtymäkohta voidaan ymmärtää pohtimalla niiden perimmäisiä käyttötarkoituksia. Molempien avulla pyritään yleensä esittämään tietoa visuaalisesti mahdollisimman ymmärrettävässä muodossa. Interaktiivisten tussitaulujen avulla toistetaan usein graafisten esitysten avulla mallinnettua tietoa. Kaavioeditoreilla pyritään vastaavasti kuvaamaan tietyn tyyppistä tietoa helposti ymmärrettävässä ja jäsennettävässä muodossa.

Yhtenevien tiedonesittämistapojen lisäksi niin interaktiivisten tussitaulujen kuin kaavioeditorienkin avulla pyritään antamaan käyttäjälle keinot esitettävän tiedon joustavaan muokkaamiseen. Interaktiivisia tussitauluja koskevassa tutkimuksessa on käsitelty esimerkiksi paljon sitä, kuinka esitettäviä objekteja voitaisiin ryhmitellä havainnollisesti [Moran *et al.,* 1997]. Kaavioeditorien puolella on tutkittu kaavioiden esteettisten kriteerien merkitystä esityksen ymmärrettävyyden kannalta [Böhringer and

Newbery Paulisch, 1990; Di Battista *et al.*, 1999] ja kehitetty erilaisia menetelmiä esteettisten kriteerien noudattamiseksi [Masui *et al.*, 1994]. Samoin yhteneviä tutkimusintressejä voidaan löytää pohdittaessa käyttäjän vapauden astetta esityksen muokkaamisessa. Igarashi *et al.* [2000] esittelevät interaktiivisten tussitaulujen yhteydessä automaattisesti luotavia lohkoja, joihin objektit sijoitetaan. Minas [2002] vertailee kaavioeditoreille tarkoitettuja ratkaisuja, jotka määräävät käyttäjälle objektien asettelussa annettavan vapauden.

Tässä työssä tutkin, mitä asioita on huomioitava, kun kaavioeditorisovellusta käytetään interaktiivisella tussitaululla. Aikaisempaa teoriatietoa käyttäen olen tehnyt jäsennyksen interaktiivisten tussitaulujen keskeisimmistä vaatimuksista vuorovaikutustekniikoiden suhteen. Perinteisesti kaavioeditorisovellukset on suunnattu normaaliin työpöytäympäristöön, jossa hiireen ja näppäimistöön pohjautuva syötteiden antaminen eroaa merkittävästi tussitauluympäristöstä. Aikaisempien tutkimusten pohjalta olen asettanut kaksi tavoitetta, joista muodostuu myös tämän tutkielman keskeinen tutkimusongelma. Kaavioeditorisovelluksen tulisi sisältää:

- helppo tapa antaa komentoja interaktiivisella tussitaululla, ja
- tehokas keino hallita taulun tarjoamaa fyysistä ja virtuaalista tilaa.

Tutkimusongelman selvittämistä varten olen jatkokehittänyt normaaliin työpöytäympäristöön tarkoitettua kaavioeditorisovellusta interaktiivisia tussitauluja varten. Tämän toteutustyön avulla pyrin paitsi konkreettisesti testaamaan työn teoriapohjaa käytännössä, myös saamaan tietoa siitä, kuinka olemassa olevan sovelluksen muokkaaminen interaktiivisia tussitauluja varten onnistuu. Interaktiivisten tussitaulujen laajemman käytön suurimpana ongelmakohtana näen sen, että yleiset ohjelmat eivät tue tussitaulujen vaatimia vuorovaikutustapoja. Toisaalta sovellusten kehittäminen varta vasten interaktiivisille tussitauluille on harvinaista. Käyttäjien suosimien ohjelmien soveltuminen interaktiivisten tussitaulujen vaatimuksiin olisi tekijä, joka monessa mielessä helpottaisi laitteiden käytön omaksumista laajemminkin.

Luvussa 2 pohdin interaktiivisen tussitaulun käsitettä. Vertailen aikaisemmin laitteista käytettyjä termejä, joiden pohjalta määrittelen tämän tutkielman puitteissa käytettävän termin ”interaktiivinen tussitaulu”. Teen myös eron interaktiivisten tussitaulujen ja kosketusnäyttöjen välille sekä esittelen erityyppisiä tussitaululaitteita. Luvussa 3 kerron tarkemmin interaktiivisia tussitauluja koskevasta tutkimuksesta. Luku jakaantuu neljään alakohtaan, jotka kattavat aikaisemman tutkimuksen pääsuunnat; painikkeet ja eleet, syötteiden käsittely ja tunnistaminen, tiedon järjestäminen sekä virtuaalisen tilan hallinta. Luvussa 4 tutustutaan kaavioeditoreihin. Suhteutan kaavioeditorit yleisiin graafisiin editoreihin ja esittelen keskeiset kaavioeditoritutkimuksen aiheet, kuten graafit, ulkoasukriteerit ja erilaiset asettelumenetelmät.

Luvussa 5 esittelen konstruktiivista työtäni. Aluksi pohdin suunnitteluvaiheessa tekemiäni ratkaisuja ja esittelen toteutustyön lähtökohtana olleen kaavioeditori-

sovelluksen. Luvun lopuksi kuvaan toteuttamani interaktiivisia tussitauluja tukevat ominaisuudet tutkimusongelman pohjalta jäsennettynä. Luku 6 sisältää kuvauksen sovelluksen testauksesta. Esittelen testitehtävät sekä testaustapahtuman läpiviennin. Lopuksi käyn läpi saadut tulokset ja suhteutan ne testauksen suunnittelussa asetettuihin oletuksiin. Luvussa 7 teen yhteenvedon ja pohdin mahdollisia jatkotutkimussuuntia.

2. Interaktiiviset tussitaulut

Tässä luvussa annan yleiskuvan interaktiivisista tussitaululaitteista ja niiden hyödyntämismahdollisuuksista. Aluksi käyn läpi suorakäyttöisyyden käsitettä, joka on tärkeässä osassa interaktiivisten tussitaulujen kanssa työskenneltäessä. Tutkielman myöhemmissä luvuissa keskitytään interaktiivisten tussitaulujen suorakäyttöisyyden tutkimiseen. Erittelen, mitä eri termejä uuden teknologian avulla toteutetuista tussitauluista käytetään, ja mitä laitteita voidaan käytännössä pitää interaktiivisina tussitauluina. Määrittelen tässä tutkielmassa käytettävän termin, interaktiivisen tussitaulun. Vertailen lisäksi interaktiivisia tussitauluja ja kosketusnäyttöjä keskenään, sillä niitä käytetään samoihin tarkoituksiin. Lisäksi ne perustuvat yllättävänkin pitkälti samoihin teknisiin ratkaisuihin. Teen yhteenvedon erilaisista interaktiivisista tussitaululaitteista. Lopuksi havainnollistan interaktiivisten tussitaulujen käyttöä esittelemällä kaksi käyttökohdetta, suunnittelutyön ja opetuksen, joissa laitteita on hyödynnetty laajalti.

2.1. Suorakäyttöisyyden käsite

Suorakäyttöisyys (direct manipulation) on Ben Shneidermanin [1982; 1983] kehittämä termi käyttöliittymille, joiden avulla käyttäjä tuntee olevansa välittömässä vuoro-vaikutuksessa esimerkiksi näytöllä näkyvien objektien kanssa. Shneiderman analysoi sovelluksia, joiden vuorovaikutustavasta oli käyttäjien keskuudessa tullut kiehtova ja kokeilemiseen innostava. Shneiderman havaitsi kolme näitä ohjelmia yhdistävää seikkaa. Ensiksikin, ohjelmalla käsiteltävät objektit ovat jatkuvasti esillä ja nähtävissä. Toiseksi, kompleksisen syntaksin sijasta sovellusten ohjaamiseen käytetään ruudulla näkyviä komentoja ja painikkeita. Kolmanneksi, tehdyt operaatiot voidaan kumota tarvittaessa ja vaikutus nähdään välittömästi.

Jo vuonna 1963 luotiin ensimmäinen suorakäyttöisyyden periaatteita hyödyntänyt järjestelmä. Ivan Sutherlandin kehittämä Sketchpad sisälsi kynän, jonka avulla pystyttiin piirtämään ja muokkaamaan viivoihin perustuvia graafisia kuvioita. Kuitenkin vasta 1980-luvun alussa suorakäyttöiset ohjelmat alkoivat yleistyä toden teolla. Tekstieditorit olivat ensimmäisiä suorakäyttöisiä sovelluksia. Ne esittivät kirjoitetun tekstin muodossa, jossa se myös tulostettaisiin paperille. Käyttäjä pystyi kursoria ohjaamalla tekemään lisäyksiä haluamaansa dokumentin kohtaan ja havaitsemaan muutokset välittömästi.

Hutchins *et al.* [1986] näkevät suorakäyttöisyyden potentiaalin siinä, että sen avulla tekstimuotoiset ohjelmatoiminnot voidaan suorittaa graafisesti tavalla, joka vastaa käyttäjän omaa mielikuvaa ongelmasta tai käyttötilanteesta. Heidän mukaansa suorakäyttöisyyttä ei kuitenkaan voida määritellä kattavasti ainoastaan niiden ominaisuuksien perusteella, jotka näkyvät ohjelmasta ulospäin. Hutchins *et al.*

perustavat suorakäyttöisyyden arvioinnin siihen, kuinka käyttäjän aikeet voidaan muuttaa järjestelmän ymmärtämiksi syötteiksi (gulf of execution) ja kuinka järjestelmä tukee tiedon esittämistä käyttäjän tarpeiden mukaisesti (gulf of evaluation). Mitä pienempi näiden kahden käsitteen välinen etäisyys on, sitä paremmat ovat mahdollisuudet suorakäyttöisyyden hyödyntämiselle. Ideaalitilanteessa järjestelmän esittämää tietoa käytetään sellaisenaan komentojen antamiseen eli syöte ja palaute yhdistyvät.

Suoravaikutteisuus (direct engagement) on termi, jonka avulla voidaan arvioida vuorovaikutusta ohjelman kanssa. Jos käyttäjä on mielestään tietokoneen tai sovelluksen sijasta suoraan kosketuksissa käsittelemiensä objektien kanssa, voidaan puhua suoravaikutteisuuden tunteesta. Yleisesti suorakäyttöisyyttä käsiteltäessä on tärkeä huomata, että käytön suoruutta ei automaattisesti voida rinnastaa sen helppouteen. Lisäksi suora vuorovaikutustapa ei välttämättä ole paras ratkaisu kaikissa käyttötilanteissa. Esimerkiksi saman operaation toistamiseen tai mekaaniseen työskentelyyn ennalta määritellyt skriptit voivat olla tehokkaampia vaihtoehtoja. [Hutchins *et al.,* 1986]

Suorakäyttöisyydestä puhuttaessa täytyy huomioida syötteiden antamiseen käytettävät apuvälineet. Vaikka nykyiset graafiset käyttöliittymät perustuvat pitkälti suorakäyttöisyyteen, syöttölaitteet eivät osaltaan aina tue parhaalla mahdollisella tavalla todellista suoraa vuorovaikutusta. Työpöytämetaforaa hyödyntävissä järjestelmissä käyttäjät ovat tottuneet vuorovaikutukseen, jossa kohteita raahataan ja nostellaan ikään kuin tasopinnalla. Kuitenkaan tämä vuorovaikutus ei ole täysin suoraa, sillä käyttäjältä vaaditaan usein hiiren käyttöä, eikä graafisia kohteita voi käsitellä vain näyttöä koskettamalla. Hiiri ja näppäimistö ovatkin epäsuoria syöttölaitteita, sillä ne toimivat ainoastaan välikappaleina näytöllä esitettävien kohteiden kanssa käytävässä vuorovaikutuksessa.

Ziegler ja Fähnrich [1988] ennustivat litteiden näyttöjen yleistymisen lisäävän sormen käyttöä sovellusten ohjaamisessa. Kosketustunnistusta hyödyntävien laitteiden määrä onkin nykyään jo huomattavan suuri. Tunnistustekniikasta riippuen laitteille voidaan antaa syötteitä ilman erillistä syöttölaitetta, kuten kynää. Tällöin suorakäyttöisyyden aste kasvaa merkittävästi. Käyttäjä voi sormea siirtämällä liikuttaa näyttöpinnalla esitettäviä objekteja, jotka reagoivat välittömästi annettuihin syötteisiin. Tämän lisäksi objektien käsittelystä saatava palaute nähdään välittömästi sormen kosketuksen tapahduttua. Tunne siitä, että sormella ohjattavat objektit olisivat todellisia, voi olla hyvin vahva. Tässä tutkielmassa käsiteltävät interaktiiviset tussitaulut ovat malliesimerkki suorakäyttöisyydestä aidoimmillaan.

2.2. Tussitauluista käytetty terminologia

Interaktiivinen tussitaulu on termi laitteelle, joka toimii näyttönä tietokoneelta välitettävälle kuvalle ja joka ottaa samalla vastaan käyttäjän antamat syötteet.

Interaktiivinen tussitaulu -termiä määrittävä tekninen piirre on ollut se, että laitteen avulla voidaan suorittaa samat toiminnot kuin normaalinkin PC-tietokoneen avulla. Käyttäjä voi esimerkiksi piirtää piirto-ohjelmalla, selailla verkkosivuja tai näyttää diaesityksiä. Interaktiivinen tussitaulu mahdollistaa nimensä mukaisesti monipuolisen vuorovaikutuksen käyttäjän ja laitteen välillä. Vuorovaikutus tapahtuu suoraan taulun pintaa koskettamalla. Interaktiivinen tussitaulu ei ole staattinen laite, joka pystyisi vain tallentamaan taululle tehdyt digitaaliset muistiinpanot.

Elektroninen tussitaulu (electronic whiteboard) on SMART Technologies -yhtiön määrittelyn [Smarttech, 2007] mukaan tietokoneeseen yhdistetty laite, jolla pystyy kirjoittamaan ja tallentamaan taululle tehtyjä merkintöjä, mutta ei näyttämään tietokoneen työpöytää tai avaamaan sovelluksia. Käytännössä sillä voidaan siis tallentaa esimerkiksi taululle kirjoitettuja muistiinpanoja myöhempää käyttöä varten. Elektroninen tussitaulu on liitettynä tietokoneeseen, jolle taululle tehdyt merkinnät voidaan halutulla hetkellä siirtää. Elektroninen tussitaulu luo ikään kuin ruudunkaappauskuvan taulun sisällöstä. Tietokoneella näitä kuvia voidaan käsitellä edelleen ja lähettää esimerkiksi sähköpostilla eteenpäin.

Elektroninen tussitaulu -termiä käytetään kuitenkin useissa yhteyksissä [Pedersen *et al.,* 1993; Moran *et al.,* 1997; Damm *et al.,* 2000] kuvaamaan tussitaulua, jonka avulla pystytään tekemään pelkkää muistiinpanoa huomattavasti monipuolisempia tehtäviä. Esimerkiksi Damm *et al.* [2002] hyödynsivät elektroninen tussitaulu -termin käytöstä huolimatta tutkimuksessaan SMART Boardia, joka toimintansa ja tekniikkansa puolesta on interaktiivinen tussitaulu. Elektroninen tussitaulu -termin käyttö ei siis ole ollut yhtenevää eri tutkimuksissa. Useimmiten elektronisella tussitaululla tarkoitetaan pelkkään muistiinpanoon ja esityksiin tarkoitettua laitetta, mutta tapauskohtaisesti on syytä varmistua tussitaulun todellisesta toiminnallisuudesta.

Digitaalinen tussitaulu (digital whiteboard) on termi, jota Rekimoto [1998] on käyttänyt kuvaamaan elektronisen tussitaulun tavoin toimivaa laitetta. Rekimoton tutkimuksessa vuorovaikutuksen välineenä ovat kädessä pidettävät *PDA-laitteet* (Personal Digital Assistant), jotka toimivat tukena tussitaulun kanssa kommunikoitaessa. Rekimoton tutkimista apuvälineistä huolimatta tussitaululaite on käytännössä hyvin lähellä interaktiivista tussitaulua.

Termistö uuteen teknologiaan perustuvien tussitaulujen osalta on melko kirjavaa ja samoista laitteista puhutaan eri yhteyksissä eri termein. Kuitenkin peruslähtökohtana voidaan pitää sitä, että interaktiivinen tussitaulu mahdollistaa elektronista ja digitaalista tussitaulua tehokkaamman ja monipuolisemman työskentelyn. Elektroniset ja digitaaliset tussitaulut on pääsääntöisesti tarkoitettu taululle kirjoitettavien tekstien tai piirrosten siirtämiseen tietokoneelle.

2.3. Tekniset toteutustavat

Interaktiivisia tussitauluja yhdistävä tekninen piirre on kuvan projisoiminen tasaiselle pinnalle eli useimmiten seinälle asennetulle taululle. Interaktiivinen tussitaulu ei itsessään ole näyttö eli se ei pysty toistamaan tietokoneen kuvaa. Vaikka interaktiiviset tussitaulut liitetään tietokoneeseen, tulee kuva siirtää taululle erikseen. Tähän tarkoitukseen voidaan käyttää projektoria. Kuvan projisoiminen interaktiiviselle tussitaululle voidaan suorittaa joko taulun etu- tai takapuolelta.

Edestä tapahtuvaan projisointiin voidaan käyttää kattoon kiinnitettyä projektoria. Suurimpana ongelmana edestä tapahtuvassa projisoinnissa ovat varjot. Mikäli kuva välitetään suoraan taulun edestä, on käyttäjän ja taulun väliin muodostuvien varjojen välttäminen vaikeaa. Ongelman helpottamiseksi voidaan käyttää lähempää tapahtuvaa projisointia. Esimerkiksi SMART Board 600i -malli [Smarttech, 2007] käyttää lyhyen varren päähän sijoitettua projektoria, joka välittää kuvan läheltä taulun yläreunaa. Lisäksi voidaan pyrkiä suoraan edestä projisoimisen sijaan sijoittamaan projektori hieman sivulle, jolloin varjojen aiheuttamat haitat vähenevät [Summet *et al.*, 2005].

Toinen tapa kuvan välittämiseen interaktiiviselle tussitaululle on taulun takaa tapahtuva projisointi. Käytännössä projektori on usein erillisen kaapin sisällä. Kaapin etureunassa on läpinäkyvä pinta, jolle kuva projisoidaan. Työläämpi vaihtoehto on tussitaululaitteen asentaminen seinän sisään, jolloin laite ei vie ylimääräistä tilaa. Takaa projisoitavia tussitaululaitteita käytettäessä vältytään varjoilta. Lisäksi takaa tapahtuvan projisoinnin avulla puhujan on helpompi esiintyä yleisölle, sillä projektori ei häikäise yleisön suuntaan puhuttaessa. Takaa projisoitavat tussitaulumallit ovat kuitenkin usein edestä projisoitavia malleja kalliimpia ja niiden käyttöön vaikuttavat huomattavasti taulun rajoitetut sijoittelumahdollisuudet.

Interaktiivisten tussitaulujen keskeisin elementti suoran vuorovaikutuksen hyödyntämisen kannalta on niiden kyky tunnistaa käyttäjän antamat kosketussyötteet. Kosketustunnistuksen toteuttamiseen voidaan käyttää useaa eri tekniikkaa. Esittelen nyt lyhyesti neljä yleisintä tapaa, joita käytetään interaktiivisten tussitaulujen syötteiden tunnistamisessa [Presentation Technology Overview, 2007]:

- Resistiivistä tekniikkaa käytettäessä interaktiivisen tussitaulun näkyvän pinnan alla on kaksi sähköä johtavaa kalvoa, joiden välissä on lepotilanteessa ilmaa (kuva 2.1a). Kun sormella tai kynällä kosketetaan taulua, kalvot koskettavat toisiaan ja resistanssi muuttuu. Muutoskohdan koordinaatit voidaan nyt paikantaa. Resistiivisten interaktiivisten tussitaulujen pinta on usein pehmeä.
- Sähkömagneettisessa tekniikassa taulupinnan alla kulkee ristikkäin johtimia (kuva 2.1b), jotka reagoivat stylus-kynän päässä olevan käämin kanssa. Kosketuksesta pystytään laskemaan kynän paikka taululla. Sähkömagneettiset tussitaulut vaativat erillisen kynän, sillä pelkkää sormea ei voida käyttää. Sähkömagneettista tekniikkaa käyttävien taulujen pinta on usein kova.

- Laser-säteitä hyödynnettäessä taulun jokaisesta kulmasta lähetetään laser-sädettä pinnan myötäisesti. Kun erillinen kynälaite osuu säteisiin, ne heijastetaan takaisin lähtöpisteisiin. Heijastetuista säteistä voidaan laskea kynälaitteen paikka taulun pinnalla. Sormea ei voida käyttää syötteiden antamiseen.
- Ultraäänen ja infrapunan yhdistelmää käytettäessä taulun pinnan kosketus saa kynälaitteen lähettämään ultraääni- ja infrapunasignaalin. Taulupinnan kulmaan kiinnitetty vastaanotin laskee eroavaisuudet signaalien saapumis-ajoista ja paikantaa kynälaitteen. Sormea ei voida käyttää, mutta taulupintana voi toimia käytännössä mikä tahansa tasainen pinta, kuten normaali seinä.

(a) (b)

Kuva 2.1. Resistiivinen (a) ja sähkömagneettinen (b) tekniikka. [MKConnect, 2007]

Perinteisessä mielessä interaktiivinen tussitaulu -termin mukaisiksi voitaisiin käsittää laitteet, jotka todella käyttävät erillistä taulua. Ultraääneen ja infrapunaan perustuvan tekniikan avulla on kuitenkin mahdollista luoda tussitaulutoiminnallisuus myös ilman varsinaista taulua. Pienikokoiset vastaanotinlaitteet tuovat uusia ulottuvuuksia interaktiivisten tussitaulujen käyttöön, sillä asennettavuus- ja siirtely-mahdollisuudet kasvavat huomattavasti. Vastaanotinlaitetta voidaan periaatteessa siirrellä täysin vapaasti. Vaatimus tietokoneen ja projektorin liittämisestä ennen käytön aloittamista rajaa kuitenkin mahdollisuuksia jonkin verran.

Monien interaktiivisten tussitaulujen mukana toimitetaan kynälaite taulun käyttöä varten. Kynälaitteet voidaan karkeasti jaotella kahteen eri tyyppiin. Kynät voivat olla aktiivisia, jolloin ne vaativat toimiakseen patterin tai suoran yhteyden tietokoneeseen esimerkiksi sarjaportin kautta. Passiiviset kynälaitteet taas eivät tarvitse virtaa, vaan ne voivat esimerkiksi ainoastaan heijastaa tai muuttaa niihin osuvia säteitä. Lisäksi joidenkin interaktiivisten tussitaulujen mukana toimitetaan kyniä, jotka eivät ole pakollisia syötteiden antamiseksi. Näiden kynien tarkoituksena voi olla eriväristen syötteiden antaminen, jolloin tussitaululaite esimerkiksi tunnistaa käytettävän tussin värin sen perusteella, mikä tussi telineestä on nostettu. Kyniä voidaan käyttää syötteiden antamiseen sormien rinnalla, mikäli käyttäjä mieltää ne hyödyllisiksi.

2.4. Kosketusnäytöt interaktiivisiin tussitauluihin verrattuna

Kosketusnäytöt (touch displays, touch screens) ovat pitkään olleet varsin kalliita laitteita. Erityisesti suurikokoisten kosketusnäyttöjen käyttö on niiden kalleudesta johtuen ollut varsin rajoitettua. Viime aikoina hintakehitys on kuitenkin mennyt suurten kosketusnäyttöjen hankkimisen osalta suotuisampaan suuntaan. Kosketusnäyttöjä ja interaktiivisia tussitauluja voidaan hyödyntää hyvin pitkälti samoihin käyttötarkoituksiin. Kosketusnäyttöjen ja interaktiivisten tussitaulujen välillä on kuitenkin joitakin eroavaisuuksia, jotka on hyvä tietää esimerkiksi laitteiden hankintaa suunniteltaessa.

Kosketusnäytöt toimivat käyttäjän näkökulmasta samalla periaatteella useiden interaktiivisten tussitaulujen kanssa, sillä syötteet annetaan suoraan näyttölaitetta koskettamalla. Kosketusnäytöt eroavat kuitenkin yhdellä perustavalla tavalla interaktiivisista tussitauluista; kosketusnäytöt ovat itsessään näyttöjä, eikä kuvaa tarvitse heijastaa niiden pinnalle projektorilla. Normaaleista tietokonenäytöistä kosketusnäytöt eivät välttämättä eroa muuten, kuin niiden syötteidentunnistusmahdollisuuden osalta. Kosketusnäytöissä kuvan tuottaminen voi perustua yhtä hyvin LCD- kuin CRT-tekniikkaan.

Vaikka kosketusnäytöissä ja interaktiivisissa tussitauluissa itse kuva välitetään eri periaatteella, ovat eroavaisuudet pienempiä kosketustunnistuksen toteuttamisessa. Taulukossa 2.1 on vertailu kosketusnäytöissä ja interaktiivisissa tussitauluissa käytettävistä tekniikoista. Kosketusnäyttöjen tunnistustekniikoista on otettu mukaan neljä yleisintä ratkaisua, kuten edellisessä kohdassa interaktiivisten tussitaulujen kohdallakin. Taulukon perusteella voidaan huomata, että kosketusnäyttöjen ja interaktiivisten tussitaulujen tunnistustekniikat perustuvat samoihin yleistason tekniikkaratkaisuihin. Kuitenkin lähemmässä tarkastelussa saman tekniikan osalta löytyy eroavaisuuksia näiden kahden laitetyypin välillä.

Tekniikka	Kosketusnäytöt	Interaktiiviset tussitaulut
Resistiivisyys	X	X
Kapasitiivisuus	X	
Sähkömagneettisuus		X
Laser		X
Ultraääni	X	X
Infrapuna	X	X

Taulukko 2.1. Kosketusnäyttöjen ja interaktiivisten tussitaulujen syötteentunnistustekniikoita.

Kuten interaktiivisten tussitaulujen kohdalla, kosketusnäyttöjenkin resistiivinen tekniikka pohjautuu kahteen kerrokseen, jotka eivät ole toistensa kanssa kosketuksissa.

Kun sormella painetaan näytön pintaa, nämä läpinäkyvät kalvot osuvat toisiinsa ja aiheuttavat muutoksen sähkövirrassa. Resistiiviset kosketusnäytöt ovat verrattain edullisia valmistaa, mutta niiden elinikä on näyttöpinnan pehmeydestä johtuen melko lyhyt. Kosketusten määrän kasvaessa pinta on altis kulumille ja naarmuille. Lisäksi näytön pinnan päälle asetettavat kosketustunnistuskerrokset voivat huonontaa kuvanlaatua [NextWindow, 2007]. Käytännössä tämä riippuu kerrosten määrästä ja materiaalista.

Kosketusnäytöissä käytettävät infrapuna- ja ultraäänimenetelmät rakentuvat näytön reunoilta lähetettävään valonsäde- tai ääniaaltoverkkoon, joka kattaa koko näyttöalan. Kun käyttäjä vie sormensa säteen tai aallon tielle, havaitaan muutos näytön reunoille sijoitettujen tunnistimien avulla. Puuttuvan säteen tai aallon perusteella lasketaan kosketukselle x- ja y-koordinaatit. Painamisvoimakkuuden laskeminen on mahdollista kosketuspinta-alan perusteella. Lisäksi infrapunaa ja ultraääntä käytettäessä voidaan syötteitä antaa ilman taulun fyysistä koskettamista, sillä säteet ja aallot kulkevat hieman näyttöpinnan yläpuolella. Tämä voi toisaalta johtaa myös tarpeettomiin syötteisiin, ellei käyttäjä ole tietoinen tunnistustekniikan tarkasta toimintaperiaatteesta. Infrapuna- ja ultraäänitekniikoita hyödyntävien kosketusnäyttöjen kuvapinnan tulee olla puhdas ja tasainen, jotta tunnistusverkko ei vääristy ja säteet pääsevät kulkemaan ilman häiriöitä. [NextWindow, 2007]

Kosketusnäyttöjen kapasitiivinen tekniikka eroaa selvimmin interaktiivisissa tussitauluissa käytettävistä tekniikoista. Näytön pinta on peitetty läpinäkyvällä sähköä johtavalla kerroksella, jonka pinnassa kulkee jännite. Näytön jokaiseen kulmaan on kiinnitetty johdin. Kun taulua kosketetaan sormella, jännite vaihtuu, ja muutos voidaan havaita vastakkaisten nurkkapisteiden välillä. Kapasitiiviset kosketusnäytöt ovat pinnaltaan usein kovia ja kestäviä. Syötteitä ei voi antaa kuin sormea käyttäen, sillä tekniikka perustuu ihmisen kosketuksen aiheuttamaan jännitteen maahan kytkemiseen. [NextWindow, 2007]

Kosketusnäyttöjen kohdalla tärkeä huomioitava tekijä on kosketustunnistuksen vaikutus näytön kuvan tarkkuuteen. Jos näyttöpinnan päälle rakennetaan useista eri kerroksista koostuva tunnistusmekanismi, saattavat nämä kerrokset aiheuttaa alla näkyvän kuvan sumenemista tai vääristymistä. Esimerkiksi infrapunaan tai ultraääneen perustuvan, näyttöpinnasta riippumattoman tekniikan käyttäminen ratkaisee tämän ongelman. Niin ikään projisoitavien interaktiivisten tussitaulujen yhteydessä tältä ongelmalta vältytään.

Kosketusnäyttöjen ja interaktiivisten tussitaulujen toinen peruseroavaisuus niiden erilaisten kuvanvälitystekniikoiden lisäksi on laitteiden kokoero. Vaikka kosketusnäyttöjä löytyy yhä suuremmista kokoluokista, ovat ne silti yleisesti ottaen pinta-alaltaan interaktiivisia tussitauluja pienempiä. Siinä missä interaktiivisten tussitaulujen käyttö rajoittuu ainoastaan suurikokoisiin laitteisiin, käytetään kosketusnäyttöjä paljon

pienikokoisissa ja kannettavissa laitteissa. Taulumikroissa (Tablet PC) ja PDA-laitteissa on usein kosketusnäyttöjä, sillä kuten suurten tussitaulujenkin yhteydessä, hiiri ja näppäimistö eivät ole optimaalisia syöttölaitteita näissä tilanteissa.

2.5. Määritelmä interaktiiviselle tussitaululle

Olen edellä esiteltyjen eri tussitaulujen määritelmien ja omien näkemyksieni pohjalta luonut tämän tutkimuksen puitteissa käytettävän määritelmän interaktiivisille tussitauluille. Työn myöhemmissä vaiheissa tulen käyttämään termiä ”interaktiivinen tussitaulu” kuvaamaan tussitaululaitteita, joissa:

1) tietokoneen kuva välitetään projisoimalla erilliselle tussitaululle tai muulle tasaiselle pinnalle,
2) syötteet tunnistetaan taulun pintaan rakennetun tai erilliseen vastaanottimeen perustuvan tekniikan avulla, ja
3) toiminnallisuus on monipuolisempaa kuin pelkkä staattinen muistiinpanojen tallentaminen ja siirtäminen.

Määritelmä on tarkoitettu selkiyttämään termistöä, jota uuden teknologian mukaisten tussitaulujen yhteydessä on yleisesti käytetty. Olen kiinnittänyt termin kolmen pääominaisuuden mukaan, jotka interaktiivisille tussitauluille ovat mielestäni keskeisimpiä ja kuvaavimpia. Yllä esitetty määritelmä kuvaa niitä laitteita, joiden parissa tehty aikaisempi tutkimus on työni tärkeimpänä pohjana. Vastaavasti pyrin määritelmällä muotoilemaan sen, millaisilla laitteilla tutkimuksen tuloksia näkisin hyödynnettävän käytännössä.

Määritelmä interaktiivisille tussitauluille on tarkoitettu rajaamaan laitteita niiden teknisten ominaisuuksien perusteella. On huomattava, että tämän tutkielman myöhemmissä luvuissa en keskity interaktiivisten tussitaululaitteiden teknisiin piirteisiin. Pääpaino on interaktiivisilla tussitauluilla käytettävien sovellusten vuoro-vaikutustekniikoiden suunnittelussa ja toteuttamisessa. Kuitenkin esimerkiksi interaktiivisten tussitaulujen ja kosketusnäyttöjen välillä on monia yhtymäkohtia. Tämän vuoksi ei samankaltaisten laitteiden parissa tehtyä tutkimusta tule jättää täysin huomioimatta.

Kosketusnäyttöihin perustuvia vuorovaikutustekniikoita on tutkittu runsaasti. Tämän työn luvussa 3 käsiteltävistä kokonaisuuksista yksi koskee sitä, kuinka interaktiivisten tussitaulujen avulla voidaan manipuloida taululla näkyviä objekteja. Vastaavasti kosketusnäyttöjen puolella on tutkittu, kuinka näytöltä tapahtuvasta objektien valitsemisesta [Albinsson and Zhai, 2003; Raisamo, 1998] ja toisaalta siirtämisestä [Collomb *et al.,* 2005] saataisiin tehokasta ja helppoa. Tämän lisäksi on tutkittu esimerkiksi kahdella sormella tapahtuvaa kohteiden valintaa kosketusnäytöltä [Benko *et al.,* 2006].

Interactive Mural [Guimbretière *et al.,* 2001] on korkearesoluutioinen interaktiivinen seinä, jonka kokonaiskuva koostuu useiden yksittäisten projektorien heijastamista kuvan osista. Vaikka Interactive Muralissa kuva välitetään projisoimalla, on se lähempänä kosketusnäyttöä kuin interaktiivista tussitaulua. Silti käytettävyytensä ja vuorovaikutustekniikoidensa osalta Interactive Mural on hyvin lähellä interaktiivisia tussitaululaitteita. Tutkimuksessaan Guimbretière *et al.* ovat kartoittaneet esimerkiksi seinämetaforan hyödyntämiskeinoja, komentojen antamista ja tilanhallintaa.

ContextWall [Baudisch and Tandler, 2002] on interaktiivista tussitaulua muistuttava laite, jonka tarkoituksena on tukea usean käyttäjän samanaikaista työskentelyä. ContextWall perustuu ratkaisuun, jossa projisoitavaan kokonaiskuvaan yhdistetään tarkempiresoluutioinen näyttö [Baudisch *et al.,* 2001]. Samanaikaisen työskentelyn tukea varten ContextWall-laitteessa on kaksi näyttöä. Näytöt on sijoitettu ohuen projisointipinnan taakse, mistä niiden kuva näkyy pinnan toisella puolella olevalle käyttäjälle. Taulukokoisessa kuvassa on näin kaksi tarkempaa aluetta, joita voidaan käyttää yksityiskohtien hahmottamista vaativassa työskentelyssä. Baudisch *et al.* [2002] ovat tutkineet, miten tätä kahteen eri tarkkuuteen perustuvaa ratkaisua voidaan hyödyntää esimerkiksi työskentelytilan yleisnäkymän käytössä tai zoomaamisessa.

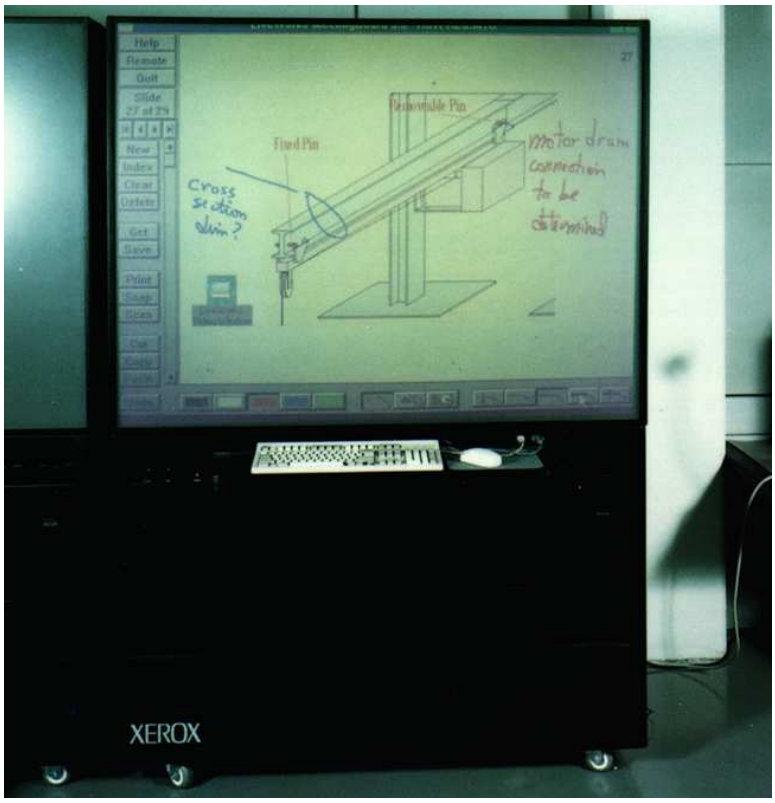
Kuten voimme huomata, laitteiden erilaisuudesta huolimatta suoran kosketuksen avulla tapahtuvassa ohjaamisessa on käytössä huomattavan paljon samoja ratkaisuja. Myös monet tässä tutkimuksessa esiintyvät interaktiivisten tussitaulujen piirteet ovat relevantteja muiden laitteiden kohdalla. Tämä kertoo suorakäyttöisyyttä hyödyntävien laitteiden samankaltaisista tai jopa täysin yhtenevistä tutkimusintresseistä. Tutkimusongelman mielekkään rajauksen mahdollistamiseksi en kuitenkaan ole ottanut käsittelyyn kosketustunnistuksella toimivia laitteita yleisellä tasolla, vaan pääpaino on tämän kohdan määrittelyn mukaisissa interaktiivisissa tussitauluissa.

2.6. Erimalliset interaktiiviset tussitaulut

Tässä kohdassa esittelen neljä interaktiivista tussitaululaitetta, jotka toimivat esimerkkeinä erilaisista teknisistä ratkaisuista. Laitteet on jaettu kahteen ryhmään sen perusteella, tarvitsevatko ne erillisen taulun käytön mahdollistamiseksi. Pyrin aikaisempien kohtien teoriapainotteisen käsittelyn jälkeen antamaan käsityksen todellisista interaktiivisista tussitaululaitteista. Tarkoituksena on ensimmäisessä alakohdassa käsiteltävän, uuden teknologian mukaisten tussitaulujen pioneerina toimineen laitteen kautta edetä tällä hetkellä käytössä oleviin tussitauluihin. Esimerkkilaitteet kattavat niin erilaisten projisointimahdollisuuksien käytön kuin myös vaihtoehtoiset tavat tunnistaa käyttäjän taululle antamat syötteet.

2.6.1. Taulua käyttävät laitteet

Liveboard [Elrod *et al.,* 1992] on ensimmäinen yleisesti tunnettu interaktiivinen tussitaululaite (kuva 2.2). Se suunniteltiin alun perin ryhmätapaamisiin, esitysten pitämiseen ja etäyhteistyöhön työympäristöissä. Xerox Palo Alto Research Center -keskuksessa (Xerox PARC) kehitetty laite perustuu läpinäkyvälle LCD-taululle takaa projisoitavaan kuvaan. Kuvan projisoimisessa käytettiin hyväksi laitteen sisälle sijoitettuja peilejä, joiden avulla säteet saatiin taitettua sopivasti taulupinnalle. Fyysiseltä kooltaan Liveboard eroaa edestä projisoitavista ja ohuista tussitaululaitteista, sillä se on rakennettu suuren, suorakaiteen muotoisen kaapin sisälle. Kynälaitteen paikannus tapahtuu optisesti. Tämä mahdollistaa kynän käyttämisen niin, ettei sillä tarvitse välttämättä koskettaa taulun pintaa syötteen antamiseksi.



Kuva 2.2. Liveboard-tussitaulu. [Elrod *et al.,* 1992]

Liveboardille on toteutettu useita ohjelmia, jotka tukevat esimerkiksi taululle tehtyjen muistiinpanojen tallentamista sekä ennalta laadittujen esitysten näyttämistä nykyisten PowerPoint-esitysten tapaan. Myöhemmin Liveboardin ympärille rakennettiin kehittyneempiä sovelluksia, kuten Tivoli [Pedersen *et al.,* 1993]. Liveboardin parissa tehty tutkimus on ollut merkittävä ensiaskel interaktiivisten tussitaulujen kehityksessä. Lähes kaikki Liveboardin jälkeen tehty alan tutkimus pohjaa jollain tavalla Xerox PARC:ssa tehtyyn pioneerityöhön.

SMART Board [Smarttech, 2007] on 1990-luvun lopulla esitelty kaupallinen interaktiivinen tussitaululaite, jonka kehitys on edelleen varsin aktiivista. SMART Board -laitteista on kolme eri mallistoa. Ensimmäisen vaihtoehdon mukaisissa laitteissa kuva välitetään tavallisella projektorilla taulun etupuolelta erilliselle kosketusherkälle taulupinnalle (kuvat 2.3a ja 2.3b). Edestä projisoitavat mallit perustuvat resistiiviseen tekniikkaan kosketusten tunnistamiseksi.

(a) (b)

Kuva 2.3. SMART Boardin 600- (a) ja 600i-mallit (b). [Smarttech, 2007]

Toinen vaihtoehto on kaappimallinen laite, jossa projektori on sijoitettu kaapin sisään ja kuva heijastetaan näyttöpinnan takaa. Tällöin kosketus tunnistetaan optisesti taulun etupuolelta. Kolmas malli perustuu olemassa olevien suurten LCD-taulujen päälle sijoitettavaan kehykseen, joka tunnistaa optisesti käyttäjän sormenliikkeet. Tämän mallin avulla tavallisesta LCD-näytöstä saadaan siis luotua kosketusnäyttö.

Kaikissa SMART Board -malleissa on mukana tusseja, joiden avulla käyttäjä voi tehdä merkintöjä taululle esimerkiksi tavallisten ohjelmien päälle. Vaikka tusseja on useimmiten neljä, laitteet eivät pääsääntöisesti kuitenkaan tue usean käyttäjän samanaikaista työskentelyä. Lisäksi SMART Boardin mukana toimitetaan useita erillisiä ohjelmia, joiden avulla suoran vuorovaikutuksen hyödyntämisestä pyritään tekemään mahdollisimman helppoa.

SMART Technologiesin interaktiiviset tussitaululaitteet ovat olleet melko suosittuja kaupallisessa mielessä ja niitä on hyödynnetty runsaasti myös tutkimuskäytössä [Mynatt *et al.,* 1999; Damm *et al.,* 2000; Ringel *et al.,* 2001]. Tämän tutkielman toteutus- sekä testausvaiheessa on käytetty SMART Board 500 -mallista interaktiivista tussitaulua.

2.6.2. Siirrettävät vastaanotinlaitteet

eBeam [eBeam, 2007] on pienikokoinen interaktiivinen tussitaululaite, joka tarvitsee projektorin lisäksi ainoastaan tasaisen pinnan, jolle kuva voidaan heijastaa. eBeamin kanssa voidaan esimerkiksi hyödyntää vanhaa, perinteisillä tusseilla käytettävää taulua. Tällöin taulupinta on tasaisempi ja kynää on helpompi liikuttaa verrattuna esimerkiksi maalattuun seinään. Projisointipinnan kulmaan kiinnitettävän eBeam-vastaanottimen

(kuva 2.4b) avulla paikannetaan erityinen stylus-kynä (kuva 2.4a), jota tulee käyttää syötteiden antamiseen. Kosketustunnistus perustuu infrapunan ja ultraäänen yhdistelmään. Tunnistetut painallukset muutetaan syötteiksi, jotka lähetetään käsiteltäviksi tietokoneelle.

(a) (b)

Kuva 2.4. eBeamin stylus-kynä (a) ja vastaanotin (b). [eBeam, 2007]

eBeamin kynässä on kaksi painiketta, joista toinen toimii kuten hiiren oikea painike ja toisella voi tuoda esiin eBeamin oman toimintovalikon. Valikossa on komentoja normaalien sovellusten hallitsemiseen sekä eBeamin omien toimintojen käyttämiseen. eBeamin etuna erillisen taulun vaativiin interaktiivisiin tussitaululaitteisiin verrattuna on luonnollisesti helppo asennettavuus erilaisiin käyttöympäristöihin.

eBeamia on käytännössä hyödynnetty esimerkiksi lastentarhassa, jossa laite on asennettu lattiapinnalle [Ovaska *et al.,* 2003]. Ovaska *et al.* tutkivat lasten yhteistoiminnallisuutta interaktiivista tussitaulua hyödyntäen. Lisäksi eBeamia on joustavuutensa vuoksi käytetty osana suurempaa järjestelmää, jonka avulla on esitetty 3D-taidetta [Rakkolainen *et al.,* 2006]. Tässä käyttötapauksessa eBeamin avulla heijastettua kuvaa ei suunnattu seinälle, vaan projisointipintana toimi sumunäyttö, jonka läpi on mahdollista kävellä. Koska näyttöpinta ei ole konkreettisesti kosketeltavissa, ei siihen voida rakentaa myöskään kosketustunnistusta. Tässä tapauksessa eBeamin kynälaitteen ja vastaanottimen avulla tapahtuva jäljitys on toimiva yhdistelmä.

Mimio [Sanford, 2007] on pienikokoinen ja helposti siirreltävä laite, joka vaatii ainoastaan normaalin projektorin interaktiivisen tussitaulutoiminnallisuuden luomiseksi. Kuten eBeam, Mimion kynäntunnistuslaite kiinnitetään halutun seinäpinnan reunaan tussitaulutoiminnallisuuden luomiseksi. Vastaanotin tunnistaa käyttäjän piirtämät liikkeet sekä painallukset, jonka jälkeen tiedot lähetetään tietokoneelle. Erillistä kynälaitetta käytetään kuten hiirtä, ja kynän kaksi painiketta voidaan määritellä tekemään haluttuja toimintoja eri tilanteissa. Mimiosta on useita erilaisia versioita riippuen siitä, haluaako käyttäjä ainoastaan tallentaa tekemiään muistiinpanoja vai

esimerkiksi yhdistää kynäntunnistuslaitteen tietokoneeseen USB-yhteyden sijasta langattomasti.

Mimion ohjelmistoon kuuluu useita erilaisia sovelluksia, joiden avulla käyttäjä pystyy muun muassa luomaan piirroksia tai esityksiä, tunnistamaan käsinkirjoitettua tekstiä tai kohdentamaan yleisön huomion tiettyyn kohtaan taulua tummentamalla reunoille jäävät alueet.

Mimiota on käytetty useissa tutkimuksissa, joissa on tarvittu helposti asennettavaa kosketustunnistuslaitetta suorakäyttöisissä järjestelmissä. Plimmer ja Apperley [2002] käyttivät Mimiota käyttöliittymäelementtien hahmottelussa ja Nam [2005] laitteiden ja niiden ohjelmistojen yhteensovittamisessa. Interaktiivisten tussitaulujen hyödyntämistä suunnittelutyössä käsitellään seuraavassa kohdassa.

2.7. Interaktiivisten tussitaulujen käyttökohteita

Edellä olen esitellyt erilaisia interaktiivisia tussitauluja. Luvun tässä kohdassa keskityn siihen, miten interaktiivisia tussitauluja hyödynnetään käytännössä. Valitsemani kahden esimerkkitapauksen avulla pyrin antamaan kuvan laitteiden todellisista käyttötilanteista. Interaktiivisten tussitaulujen 1990-luvun alkupuolelta alkaneen kehityksen aikana on muotoutunut tiettyjä tehtäviä, joissa laitteita käytetään erityisen paljon työskentelyn apuna. Tämä kertoo siitä, että interaktiivisten tussitaululaitteiden ominaispiirteet tulevat parhaiten esiin määrätynlaisissa käyttökohteissa. Käyttötapauksissa, joissa työskentelytilan suuri koko, suora ja intuitiivinen vuorovaikutus sekä joustava tiedonmuokkaus ovat tärkeitä, voi interaktiivisten tussitaulujen käytöstä olla suurta hyötyä.

Seuraavissa alakohdissa käsiteltävät käyttökohteet eivät varmasti ole ainoita esimerkkejä, joissa interaktiiviset tussitaulut toimivat tärkeinä apuvälineinä. Suunnittelutyössä ja opetuksessa interaktiivisia tussitauluja on kuitenkin hyödynnetty jo useita vuosia. Lisäksi tutkimus näiden käyttötapausten kohdalla on ollut hyvinkin kattavaa.

2.7.1. Suunnittelutyö

Interaktiivisia tussitauluja voidaan hyödyntää tehokkaasti erilaisissa suunnittelutehtävissä. Perinteisesti suunnittelutyössä on käytetty esimerkiksi suuria paperiarkkeja, seinällä olevia tauluja tai yksittäisiä muistilappuja. Interaktiivisten tussitaululaitteiden avulla käytettävissä oleva tila on käytännössä rajaton, sillä näyttöalaa voidaan vierittää tai siirtää tarvittaessa. Samoin aikaisemmin tehtyjen suunnitelmien muokkaaminen on joustavaa, sillä edellisiä hahmotelmia voidaan poistaa tai siirtää sivummalle helposti. Ehkäpä tärkeimpänä tekijänä voidaan pitää interaktiivisten tussitaulujen mahdollistamaa suoraa vuorovaikutusta piirtoalan kanssa. Hahmotteleminen on nopeaa ja intuitiivista, sillä käyttäjä voi piirtää vapaalla kädellä ilman erillisiä apuvälineitä, kuten esimerkiksi hiirtä.

Interaktiivisille tussitauluille on toteutettu monia *UML-kaavioiden* (Unified Modeling Language) luomiseen tarkoitettuja sovelluksia [Chen *et al.,* 2003; Damm *et al.,* 2000; Lank *et al.*, 2000]. UML-kaavioiden suunnittelussa ensimmäisenä tehtyyn versioon tulee väistämättä muutoksia, jolloin ne voidaan tehdä joustavasti esimerkiksi kynän avulla. UML-kaaviosovellusten yhteydessä on myös hyödynnetty piirrettävien kuvioiden tunnistamista. Tällöin vapaalla kädellä hahmoteltu kuvio tai teksti voidaan korvata tarkemmalla, tietokoneella piirretyllä versiolla. Näin suunnitelmia voidaan helpommin käsitellä tietokoneella, sillä piirretty UML-kaavio voidaan tallentaa jossakin yleisesti käytetyssä tiedostoformaatissa.

Interaktiivista tussitaulua on hyödynnetty myös käyttöliittymäsuunnittelussa [Plimmer and Apperley, 2002]. Käyttäjä voi hahmotella vapaalla kädellä käyttöliittymäelementtejä, kuten pudotusvalikoita tai valinta- ja tekstinsyöttökenttiä. Sovellus tunnistaa hahmotelmat tiettyjen ennalta määriteltyjen piirteiden perusteella ja esittää tulokset piirretyssä muodossa. Tämä helpottaa esimerkiksi kokemattomien käyttäjien käyttöliittymäsuunnittelua, sillä ohjelmakoodin kirjoittamisen sijaan pelkkä hahmottelu riittää tulosten luomiseksi. Lisäksi interaktiivisille tussitauluille on kehitetty sovelluksia piirilevyn suunnitteluun [Liu *et al.*, 2003] sekä laitteiston ja sen ohjelmiston yhteensovittamiseen [Nam, 2005].

Interaktiivisille tussitauluille luodut suunnittelusovellukset ovat usein niiden suunnittelijoiden ja toteuttajien mielenkiinnon kohteiden mukaisesti pitkälti tekniikkaan painottuneita, kuten yllä olevista esimerkeistä käy ilmi. Kuitenkin muissakin suunnittelutilanteissa interaktiiviset tussitaulut ovat käyttökelpoisia. Music Notepad [Forsberg *et al.*, 1998] on vapaalla kädellä käytettävä sävellysohjelma, joka tunnistaa taululle piirretyt nuotit. Ohjelma tuntee suuren joukon eri nuotteja ja lukuisia komentoja niiden käsittelyyn.

2.7.2. Opetus

Perinteisessä luokkaopetuksessa liitutauluja ja myöhemmin tussitauluja on käytetty hyvin pitkään. Näiden taulujen hyödyntäminen on edelleen tärkeä osa nykyisiä opetusmenetelmiä, vaikka teknologia kehittyy ja valtaa alaa myös koulumaailmassa. Siirtymä perinteisistä tauluista interaktiivisiin tussitauluihin ei periaatteessa ole suuri. Tietokoneeseen yhdistetty, kosketusherkkä taulu toimii tarvittaessa kuten perinteinen taulu, jolle opettaja voi kirjoittaa merkintöjä oppilaiden nähtäväksi. Tämä on yksinkertaisin esimerkki interaktiivisten tussitaulujen käytöstä, joka ei sinänsä juurikaan hyödynnä laitteiden todellista potentiaalia. Toinen perustava käyttötapa interaktiivisille tussitauluille voi olla ennalta valmisteltujen PowerPoint-esitysten korostaminen opetuksen yhteydessä. Nykyisin luennoitsijat pystyvät hiirtä käyttäen piirtämään merkintöjä kalvojen päälle, mutta useimmiten tämä on hyvin vaivalloista epäsuoralla osoitinlaitteella. Vapaalla kädellä tapahtuva piirtäminen on miellyttävämpää ja

tarkempaa luennoitsijan kannalta sekä huomattavasti havainnollisempaa esitystä seuraavien kannalta.

Interaktiivisten tussitaulujen todellinen voimavara opetuksen yhteydessä on mahdollisuudessa lisätä vuorovaikutusta opettajan ja oppilaiden välille. Opettaja voi kirjoittaa taululle kysymyksen tai tehtävän, jonka oppilaat näkevät omilla päätteillään, ja johon he voivat laatia vastauksen itsenäisesti [Berque *et al.*, 2001]. Tämä edellyttää oppilaskohtaisilla laitteilla varustettua luokkahuonetta. Tietokoneeseen yhdistetyn tussitaulun avulla voidaan paitsi muokata esitettävää tietoa joustavasti, myös suorittaa taustalla kompleksisempia tehtäviä. E-Chalk [Friedland *et al.*, 2004] mahdollistaa muun muassa kuvien liittämisen taululle, matemaattisten kaavojen tunnistamisen ja laskemisen sekä opettajan puheen ja muistiinpanojen välittämisen verkon kautta. Etäopetus ja -luennointi ovat Internetin aikakaudella yhä yleistyvä kehityssuunta [Chong and Sakauchi, 2000].

Interaktiiviset tussitaulut voivat olla tehokas keino yhteistoiminnan tukemiseksi opetuksessa. Ovaska *et al.* [2003] ovat tutkineet, kuinka interaktiivista tussitaulua voidaan käyttää opetuksen apuna lastentarhassa. Tutkimuksessa lapset toimivat yhteisen tehtävän parissa samaa interaktiivista tussitaulua käyttäen. Lasten oppimisen tueksi on myös kehitetty järjestelmä, jossa niin opettajat kuin oppilaatkin voivat samanaikaisesti käyttää taulua [Otsuki *et al.*, 2001]. Ympäristöä testattiin oppimispelin avulla, jossa vertailtiin lasten keskittymistä peliin suhteutettuna siihen, pelasiko lapsi yksin vai toisen lapsen kanssa. Kaksinpelin todettiin saavan lapset suuntaamaan keskittymisensä peliin tehokkaammin.

Uusien teknologioiden, kuten interaktiivisten tussitaulujen, yleistyessä opetuksessa kehitystä ajavat usein laitteiden käytöstä innostuneet tahot, jotka eivät aina ole yhteistyössä pedagogiikan asiantuntijoiden kanssa. Interaktiivisten tussitaulujen käytöstä opetuksessa onkin tehty viime vuosina tutkimuksia, joissa laitteiden vaikutuksia oppimiseen on tutkittu sopivan kriittisellä otteella [Beauchamp and Parkinson, 2005; Virtual Learning, 2002]. Pääsääntöisesti tulokset ovat olleet rohkaisevia. Interaktiiviset tussitaulut tuovat opetukseen monia mahdollisuuksia, jotka ovat aikaisemmin olleet vaikeita toteuttaa. Uudet laitteet asettavat kuitenkin opettajille suuren haasteen, sillä opetuksen suunnittelun tulee muotoutua koulun tarpeiden mukaisesti. Samat käytännöt eivät välttämättä toimi jokaisessa koulussa [BECTA, 2003]. Opettajilla tulee myös olla tarpeeksi aikaa tutustua uusiin laitteisiin. Tämän jälkeen tulee varata riittävästi resursseja tuntien valmisteluun, jotta uuden teknologian mahdollisuuksista saadaan todellista hyötyä [Levy, 2002].

2.8. Yhteenveto

Interaktiivisten tussitaululaitteiden kirjo on suuri ja ne toimivat eri tekniikoiden avulla. Tässä tutkielmassa olen päättänyt olla tekemättä tiukkaa rajausta laitteiden välille. Käytännössä tutkimuksen piiriin kuuluvat laitteet, jotka täyttävät kohdassa 2.5

interaktiivisille tussitauluille antamani määritelmän. Koska tutkimukseni pääpaino on ohjelmistoteknisissä käyttöliittymäasioissa, teknisen toteutuksen vaihtelevuudesta huolimatta samat lainalaisuudet pätevät lähes kaikissa laitteissa. Esittelemäni erilaiset interaktiiviset tussitaululaitteet, Liveboard, SMART Board, eBeam ja Mimio, ovat esimerkkejä alustoista, joilla tutkimuksessa käsiteltäviä asioita voitaisiin käytännössä hyödyntää.

Interaktiivisia tussitauluja voidaan käyttää moniin tarkoituksiin. Muistiinpanojen teko, esitelmien pitäminen, etäneuvottelut ja piirtäminen ovat joitakin esimerkkejä monista käyttökohteista. Erityisen suosittuja sovellusalueita interaktiivisten tussitaulujen tutkimuksessa ovat olleet graafinen suunnittelutyö ja opetus. Omaa tutkimustani lähimpänä ovat juuri graafiset piirto- ja suunnittelusovellukset. Kaavioeditorit ovat tyyppiesimerkki sovelluksista, joita käytettäessä interaktiivisten tussitaulujen parhaita ominaispiirteitä voidaan hyödyntää tehokkaasti. Luvussa 5 kuvaamassani konstruktiivisessa vaiheessa olen toteuttanut interaktiivisille tussitauluille suunnatun kaavioeditorisovelluksen.

3. Interaktiivisten tussitaulujen keskeiset ominaispiirteet

Tässä luvussa keskitytään tarkastelemaan aikaisempien tutkimustulosten valossa sitä, mihin asioihin interaktiivisten tussitaulujen sovellusten käyttöliittymien ja vuorovaikutustapojen suunnittelussa ja toteuttamisessa tulee kiinnittää huomiota. Koska laitteet ovat fyysisiltä ominaisuuksiltaan täysin erilaisia työpöytäkoneisiin verrattuna, tulee myös sovellusten käytettävyydessä ottaa huomioon eri asioita. Olen pyrkinyt tarkastelemaan interaktiivisten tussitaulusovellusten ominaisuuksia riippumatta siitä, mihin tehtävään sovellus on tarkoitettu. Tällä yleisen tason tarkastelulla olen saanut koottua perustavan tason ominaispiirteitä, jotka ovat tärkeitä sovellustyypistä riippumatta.

Luku on jaettu neljään eri kohtaan, joista jokaisessa käsiteltävät asiat vaikuttavat olennaisesti interaktiivisille tussitauluille tarkoitettujen sovellusten toimivuuteen. Syötteiden antaminen, syötteiden käsittely ja tunnistaminen, tiedon järjestäminen sekä virtuaalisen tilan hallinta ovat olleet merkittävimpiä aihepiirejä interaktiivisia tussitauluja koskevassa tutkimuksessa.

3.1. Syötteiden antaminen

Interaktiivisia tussitauluja käytettäessä saatavilla ei aina ole perinteisiä syöttölaitteita, kuten hiirtä tai näppäimistöä. Tämä on todellisen suoran vuorovaikutustekniikan hyödyntämisen peruslähtökohta. Syötteet pitäisi pystyä antamaan suoraan haluttua kohdetta manipuloimalla. Joissakin tilanteissa tästä on kuitenkin jouduttu tinkimään, sillä tehokkaat tekstinsyöttömahdollisuudet pelkkää kynää tai sormea käyttäen ovat olleet varmatoimisten syötteentunnistusmekanismien puuttumisen vuoksi vähissä.

Interaktiivisten tussitaulujen kehityksen alkumetreillä huomattiin merkittäviä eroavaisuuksia ihmisten tavassa käyttää kynää hiireen verrattuna. Xerox PARC:ssa kehitettyä Liveboardia testattaessa ihmiset käyttivät mielellään kynää piirtämiseen, mutta pitivät siihen sijoitettuja, hiirestä tuttuja painikkeita epäkäytännöllisinä [Elrod *et al.,* 1992]. Lisäksi huomattiin, että hiirellä heikosti toimivat luonnollisuuteen pyrkivät syötteet, kuten pyyhkäisyt tai osoittamiset, saattaisivat olla kynää käytettäessä hyödyllisiä. Näitä syötteitä kutsutaan yleisesti termillä *eleet* (gestures). Vertailen seuraavaksi erilaisia lähestymistapoja syötteiden ja komentojen antamiseksi interaktiivisille tussitauluille.

3.1.1. Painikkeet ja eleet

Liveboardille suunnitellussa Tivoli-sovelluksessa Pedersen *et al.* [1993] ovat käyttäneet komentojen antamiseen perinteisen graafisen käyttöliittymän ja eleiden yhdistelmää. Eleillä tarkoitetaan kynällä tehtäviä piirtoja, jotka tulkitaan komennoiksi tai muuksi syötteeksi. Eleet eivät ole interaktiivisten tussitaulujen yhteydessä kehitetty vuorovaikutustapa, vaan jo 1980-luvun alkupuolella on tutkittu sormella tehtävien

liikkeiden käyttämistä graafisen esityksen muokkaamisessa. Margaret Minsky [1984] tutki sormen asennon, näyttöpintaan kohdistuvan paineen ja eleen suunnan käyttämistä vuorovaikutuksen pohjana. Lisäksi GRANDMA-työkaluun perustuvan tutkimuksen avulla on mahdollistettu käyttäjän omien eleiden luominen ja niiden yhdistäminen haluttuun toimintoon piirto-ohjelmassa [Rubine, 1991]. Ongelmana eleiden hyödyntämisessä on Dean Rubinen mukaan ollut luotettavan tunnistusmekanismin kehittäminen.

Tivolista on kehitetty useita eri versioita [Moran *et al.,* 1997; Moran *et al.,* 1998a; Moran *et al.,* 1998b], joihin on lisätty ominaisuuksia suoritettujen käyttäjäkokeiden perusteella. Tivolin ensimmäisessä versiossa eleitä ei vielä ollut käytettävissä. Seuraavassa esimerkkitapauksessa [Pedersen *et al.,* 1993] kuvaillaan Tivolin alkuperäistä vuorovaikutusmallia: ”Tivolin käyttäjä siirtyy interaktiivisen tussitaulun luo ja kirjoittaa haluamansa tekstin vapaasti omaa käsialaansa käyttäen. Tämän jälkeen hän huomaa tehneensä kirjoitusvirheen tai haluaa muusta syystä johtuen korjata tekstiä. Käyttäjä etsii taulun alareunaan sijoitetuista toimintokuvakkeista pyyhi-painikkeen, jonka jälkeen kursori muuttuu pyyhekumiksi. Liikuttamalla kynää taulun pinnalla haluttu osa tekstistä pyyhkiytyy pois. Tämän jälkeen käyttäjä valitsee jälleen piirrä-painikkeen ja jatkaa kirjoittamista.”

Ensimmäisten Tivolilla suoritettujen testien perusteella havaittiin tarve eleille. Tussitaulu on fyysiseltä kooltaan suuri laite, jolloin sen reunoille sijoitettujen painikkeiden toistuva käyttäminen on hankalaa. Käyttäjän tulee tällöin ensin siirtää katsettaan työskentelykohteesta, jonka jälkeen hänen tarvitsee paikantaa halutun toiminnon käynnistävä painike näyttöalan sivulta. Tämän jälkeen on vielä kuroteltava tai siirryttävä pystyäkseen painamaan komentopainiketta. Mahdollisimman suuri osa ohjelman toiminnallisuudesta tulisi olla käytettävissä suoraan kynällä ilman painikkeita, tai painikkeiden tapauksessa komentojen pitäisi vähintäänkin olla käyttäjän ulottuvilla jatkuvasti. Luonnolliseen vuorovaikutustapaan pyrittäessä ylimääräiset painikkeet kynässä eivät ole toivottava vaihtoehto.

Tivolin kehittyneemmissä versioissa on toteutettu joitakin eleitä. Niissä poistaminen tapahtuu piirtämällä ympyrä tai muu suljettu kuvio halutun objektin ympärille, valitsemalla tämä suljettu alue aktiiviseksi painamalla aktivointia kuvaavaa valitse-painiketta ja piirtämällä poistoa vastaava ele alueen sisällä. Heikkoutena tässäkin Tivolin versiossa on se, että painikkeita tulee yhä käyttää eleiden tukena. Eleiden käyttämisen hankaluuden todettuaan Pedersen *et al.* [1993] päätyivät käyttämään tavallisen piirtämisen ja eleellä annettavan komennon erottamiseen kynään sijoitettua painiketta, jota pohjassa pitämällä voitiin valita haluttu toiminto.

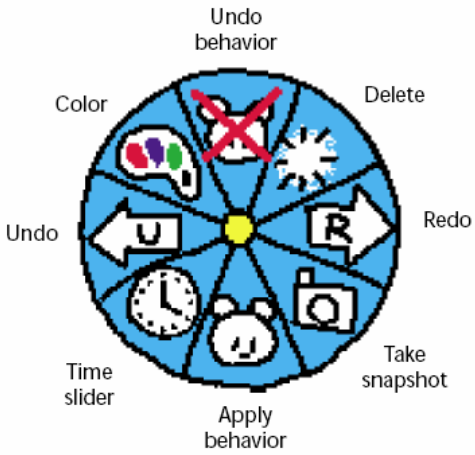
3.1.2. Eleet ja piirakkavalikko

Flatland [Mynatt *et al.,* 1999; Igarashi *et al.,* 1999] on tarkoitettu epämuodollisempaan ja yksinkertaisempaan työskentelyyn kuin Tivoli. Se on suunnattu tueksi käyttöön, jossa

yksi henkilö hyödyntää taulua lähinnä henkilökohtaisiin tarpeisiinsa. Käyttäjä voi esimerkiksi listata työhuoneessaan olevalle interaktiiviselle tussitaululle työtehtäviään tärkeysjärjestykseen. Flatland tukee myös yksinkertaisten karttojen piirtämistä sekä laskutoimitusten suorittamista. Flatland pyrkii toteuttamaan perinteisen tussitaulu-metaforan, jossa näkyvillä on ainoastaan valkoinen taulu. Tähän käyttötarkoitukseen sopii Flatlandin valitsema tapa syötteiden ja komentojen antamiseen, sillä kaikki tapahtuu vapaalla kädellä tehtävien piirtojen avulla. Jatkuvasti näkyvillä olevia käyttöliittymäelementtejä, kuten esimerkiksi menuvalikoita tai painikkeita, pyritään välttämään, eikä niitä olekaan Flatlandissa käytössä.

Flatlandia käytetään erityisellä kynällä, jossa on kaksi tilaa. Ensimmäisen ollessa päällä syötteet tulkitaan piirroiksi. Toista tilaa käytetään painamalla kynään sijoitettu painike pohjaan, jolloin piirrot tulkitaan eleiksi. Tämä muistuttaa pitkälti Tivolissa [Pedersen *et al.,* 1993] käytettyä tapaa. Taululla olevaa materiaalia voidaan esimerkiksi siirtää pitämällä kynän painiketta pohjassa, painamalla kynä taulun pintaan halutun objektin kohdalla, siirtämällä kynää ja lopulta nostamalla se irti halutussa kohdassa.

Komentojen antaminen Flatlandissa perustuu eleiden ja erityisen piirakkavalikon (kuva 3.1) yhdistelmään. Pitämällä kynän painiketta pohjassa ja napauttamalla kerran taulua, ponnahtaa esiin ympyrän muotoinen valikko. Käyttäjä pystyy pienellä kynän siirrolla valitsemaan haluamansa toiminnon valikosta.



Kuva 3.1. Flatlandissa käytettävä piirakkavalikko. [Mynatt *et al.,* 2000]

Flatlandin piirakkavalikosta löytyvät toiminnot muun muassa komennon kumoamiselle, uudelleensuorittamiselle, värin vaihdolle ja toiminnon valinnalle. Käyttäjän valitsemasta toiminnosta riippuen Flatland tunnistaa tehdyt piirrot joko normaaliin piirtotilaan, tehtävälistaan, 2D-piirtämiseen, kartan piirtämiseen tai laskutoimitukseen kuuluviksi. Flatlandissa pyritään lisäksi tekemään piirakkavalikon käytöstä entistä tehokkaampaa ajan myötä. Kun käyttäjä on opetellut sijainnit piirakkavalikon tarjoamille komennoille, hän voi suorittaa niitä ilman valikon näyttämistä pitämällä painiketta pohjassa ja liikuttamalla kynää kuviteltua piirakan

sektoria kohti. Tämä tekee komentojen antamisesta nopeampaa. Tavallisesti valikko tulisi näkyviin vasta, kun käyttäjä nostaa kynän taulun pinnasta.

Flatlandin vahvuutena syötteiden ja komentojen antamisessa on eleiden tehokas käyttö. Kaikki toiminnot voidaan suorittaa pelkkiä kynän piirtoja käyttäen. Eleiden käyttö komentojen antamisessa ei kuitenkaan ole täysin ongelmaton tapa. Kritiikkiä on osoitettu eleiden käytön vaikutuksista käyttäjään. Väärä tunnistus tai eleen unohtaminen voivat aiheuttaa keskeytyksiä käytössä ja lisätä pelkoa eleiden käyttöä kohtaan [Nakagawa *et al.,* 1999].

Näytöllä jatkuvasti esillä olevien kuvakkeiden käyttöön verrattuna eleet vaativat enemmän opettelua, sillä niiden joustava hyödyntäminen sovelluksen ohjaamisessa vaatii eri eleiden muistamista. Flatlandin heikkoudeksi voidaan laskea käytön riippuvuus erillisestä kynästä. Piirtojen ja eleiden erottamiseksi tarvitaan kynästä löytyvää painiketta, mikä rajoittaa sovelluksen käytettävyyttä. Pelkällä sormella annettavat syötteet ovat aina intuitiivisempia, sillä kynääkin voidaan periaatteessa pitää tarpeettomana välikappaleena suorassa vuorovaikutuksessa.

3.1.3. Eleet ilman kynää

Ringel *et al.* [2001] ovat kehittäneet kiehtovan Barehands-vuorovaikutustekniikan, joka perustuu pelkän käden käyttöön interaktiivisen tussitaulun ohjaamisessa. Barehands on luotu osana *jokapaikan tietotekniikkaa* (pervasive computing) tutkivaa projektia, johon kuuluu PDA-laitteiden ja kannettavien tietokoneiden lisäksi kaksi interaktiivista tussitaululaitetta. Tutkijat huomasivat, että käyttäjiä kiehtoo suora vuorovaikutus taulujen kanssa ilman kynälaitteita. Tämän havainnon pohjalta suunniteltiin tekniikka, jonka avulla käyttäjä pystyy käden asentoa muuttamalla antamaan useita erilaisia komentoja interaktiiviselle tussitaululle.

Käsien luonnollista käyttöä osana tietokoneen kanssa tapahtuvaa vuorovaikutusta on tutkittu jo 1980-luvun alussa [Krueger, 1983; Krueger *et al.,* 1985]. Kruegerin VIDEOPLACE-järjestelmä perustui videon avulla kuvatun käyttäjän hahmon yhdistämiseen tietokoneeseen. Käyttäjä näki oman hahmonsa heijastettuna seinälle, jossa oli samanaikaisesti näkyvissä tietokoneella luotuja graafisia objekteja. Käyttäjä pystyi esimerkiksi hallitsemaan ovaalin muotoista objektia etusormien ja peukaloiden avulla.

Barehandsin tunnistustekniikka perustuu SMART Boardiin yhdistettyyn, infrapunatunnistuksella toimivaan kameraan. SMART Board tunnistaa taulun kosketuksen ja kamera puolestaan käden asennon. Kamera tarkkailee jatkuvasti käyttäjän käden asentoa ja välittää tiettyjen ennalta määriteltyjen asentojen yhteydessä tiedon komennon antamisesta. Barehands pystyy tunnistamaan yhden tai kaksi sormea, pysty- tai vaaka-asentoisen käden ja kämmenen. Toimintojen määrittelemiseen on olemassa erillinen ohjelma, jonka avulla käden asennot voidaan kiinnittää haluttuihin komentoihin. Valittavissa olevat komennot ovat hiiren painikkeiden toiminnot, SMART

Boardin mukana toimitettavien ohjelmien toiminnot ja Windowsin standardit oikotiekomennot.

Barehands ei tunnista rajatonta määrää käden asentoja, mutta idealtaan tekniikka on varsin mielenkiintoinen. Käyttäjän ei tarvitse kurotella tai kävellä taulun reunaan painaakseen painiketta. Lisäksi erillistä fyysistä osoitinlaitetta ei tarvita, vaan pelkän käden asennon avulla voidaan suorittaa useita komentoja. Tämä on lupaava kehityssuunta, sillä mitä luonnollisempia vuorovaikutuskeinot interaktiivisten tussitaulujen kanssa ovat, sitä helpompaa ja nopeammin omaksuttavaa taulun käyttö teoriassa on. Tästä huolimatta on todettava, että löytyy myös käyttötilanteita, joissa fyysiset osoittimet tai erilliset apuvälineet ovat tarpeen.

Barehandsissa käytettävän tunnistustekniikan kaltaisten ratkaisuiden kehittyessä olisi mahdollista määritellä yhä intuitiivisempia käden asentoja komentojen antamiseksi. Käyttäjä voisi esimerkiksi siirtää haluttua objektia taululla yhdistämällä peukalon sekä etusormen ja näin ikään kuin poimia objektin sormien väliin. Sormien asentojen käytön yhteydessä optinen tunnistus saattaisi tosin osoittautua liian epätarkaksi. Käden asentojen tunnistus yhdistettynä eleisiin antaisi vuorovaikutuksen suunnitteluun jo huomattavan suuren määrän erilaisia vaihtoehtoja.

3.2. Syötteiden käsittely ja tunnistaminen

Syötteiden antaminen interaktiiviselle tussitaululle on periaatteessa varsin helppoa. Pieni lapsikin voi vaivatta piirrellä haluamiaan kuvioita taululle. Hiiren tai näppäimistön käyttöön verrattuna sormin tapahtuva ohjaus tuntuu nopeasti omaksuttavalta ja ongelmattomalta ratkaisulta. Vapaan käden avulla piirrettäessä tai kirjoitettaessa annettujen syötteiden jälki taululla ei kuitenkaan aina vastaa käyttäjän toiveita. Kirjaimet voivat näyttää suttuisilta tai piirretystä ympyrästä ei tulekaan pyöreää. Niin ikään projektorin kohdistus ei välttämättä ole kohdallaan, jolloin piirrokset tai kirjoitukset eivät ilmesty näkyviin samaan kohtaan, jossa kynä koskettaa taulua.

Kosketustunnistuksella varustettujen tussitaulujen tekniikka ei aina ole tarpeeksi herkkää tarkimpien syötteiden tunnistamiseksi. Optisten tunnistustekniikoiden yhteydessä laitteet eivät aina aukottomasti pysty havaitsemaan kättä tai kynää. Nämä ovat interaktiivisille tussitauluille valitettavasti kovin ominaisia piirteitä. Ehdotuksia näiden puutteiden ja ongelmakohtien ratkaisemiseksi on kuitenkin olemassa.

Käsittelen seuraavissa alakohdissa käyttäjän piirtämien kuvioiden tunnistamista interaktiivisiin tussitauluihin yhdistettynä. Tähän tarkoitukseen on kehitetty lukuisia ohjelmia, joiden avulla annettu tieto pyritään tunnistamaan ja muuttamaan tietokoneen ymmärtämään muotoon. Tunnistusta käyttäen syötettä voidaan digitaalisessa muodossa täydentää tai sen visuaalista esitystä parantaa. Lisäksi tunnistettua tietoa voidaan käsitellä bittimuodossa esimerkiksi laskutoimituksissa. Tätä käyttötapaa on tutkittu jo 1990-luvun alussa, jolloin Wellner [1991] esitteli digitaalisen ja perinteisen

työskentelyn yhdistävän DigitalDesk-työpöydän, jota hyödynnettiin muun muassa laskutoimitusten suorittamisessa. Tunnistus syötteiden käsittelyssä on laajempi tutkimusala, josta tässä yhteydessä otan esille interaktiivisten tussitaulujen kannalta relevantteja ideoita ja toteutuksia.

3.2.1. Tekstintunnistus

Tietokoneella tapahtuvassa tekstintunnistuksessa pyritään tulkitsemaan käyttäjän vapaalla kädellä kirjoittamat kirjaimet, numerot ja mahdollisesti muut merkit. Interaktiivisten tussitaulujen yhteydessä näppäimistön käyttö on paitsi ilman erillisiä syöttölaitteita tapahtuvan suorakäyttöisyyden periaatteiden vastaista, myös käytännössä hankalaa. Kuitenkin ihmiset ovat perinteisten työpöytätietokoneiden käytössä tottuneet syöttämään tekstin näppäimistöltä. Tämä asettaa interaktiivisten tussitaululaitteiden käytölle uuden vaatimuksen, sillä kynän tai sormen avulla on pystyttävä antamaan myös tekstimuotoista dataa tussitaulun kautta tietokoneelle.

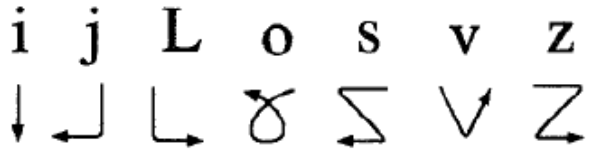
Tekstintunnistusta on käytetty niin kaupallisissa [eBeam, 2007] kuin myös tutkimuskäyttöön tarkoitetuissa [Guimbretière *et al.,* 2001] tussitaulu- tai kosketusnäyttölaitteissa. Calligrapher [PhatWare, 2007] on alun perin pienikokoisille PDA-laitteille tarkoitettu tekstintunnistusohjelma, jota Guimbretière *et al.* ovat hyödyntäneet Interactive Muralissa. Calligrapher yhdistää tekstin tunnistamisessa useita eri tekniikoita, kuten sumeaa logiikkaa ja neuroverkkoja. Lisäksi Calligrapher pyrkii opettelemaan käyttäjän usein tekemiä virheitä, ja se osaa joissakin tapauksissa tehdä korjaavat toimenpiteet automaattisesti. Interactive Muralissa tunnistus aktivoituu, kun käyttäjä lopettaa kirjoittamisen nostamalla sormen tai kynän taulun pinnalta. Tällöin kirjoitettu teksti ohjataan Calligrapherille käsiteltäväksi, jonka jälkeen tunnistettu merkkijono lähetetään takaisin pääohjelmalle.

Interaktiivisten tussitaulujen yhteydessä toimivasta tekstintunnistuksesta voi olla suurta apua käyttäjälle. Tunnistusta hyödynnettäessä tulee kuitenkin ottaa huomioon tiettyjä asioita. Vapaalla kädellä kirjoitetun tekstin automaattinen korvaaminen tunnistetulla versiolla voi olla vaarallista käytön joustavuuden kannalta. Interactive Muralissa syötteen tunnistettu versio näytetään käsinkirjoitetun tekstin rinnalla taulun alareunassa. Tämän jälkeen käyttäjä voi valita, korvataanko käsin kirjoitettu teksti tietokoneen esityksellä. Mikään tekstintunnistusteknologia ei pysty sataprosenttisen varmoihin tunnistustuloksiin, ja näin ollen väistämättömien virhetilanteiden johdonmukainen ja nopea käsittely on tärkeää. Jos alkuperäinen teksti vaihdetaan tunnistettuun versioon automaattisesti, joutuu käyttäjä virheellisen tunnistuksen sattuessa tekemään korjaustoimenpiteitä.

Tekstintunnistuksen suurimpana haittapuolena ovat väärät ja virheelliset tunnistukset [Mynatt *et al.,* 1999]. Tekstintunnistusohjelmissa voidaan välttää vääriä tunnistuksia tarkkailemalla piirtojen muotoja. Calligrapherissa voidaan ei-tekstimuotoisten syötteiden ohittamiseksi käyttää heuristiikoita, jotka asettavat

tunnistuksen pois päältä, mikäli syötteenä annetun piirron muoto ei näytä vastaavan mitään tunnettua kirjainta tai numeroa. Näin vältetään selvästi turhien tunnistuspyyntöjen lähettäminen ja käsittely.

Interaktiivisilla tussitauluilla tekstinsyöttö on poikkeuksetta tapahtunut normaalein kirjaimin ja merkein eli käytännössä länsimaisia aakkosia käyttäen. Tietokoneiden tekstinsyöttöä on tutkittu laajemmin, jonka seurauksena on kehitetty erilaisia tapoja ja merkistöjä syötteiden antamiseen. Yksi vaihtoehtoinen keino tekstin syöttämiseen ovat Unistroke-merkit [Goldberg and Richardson, 1993]. Unistroke-merkistö (kuva 3.2) on kehitetty tekemään tekstin syöttämisestä mahdollisimman nopeaa pienille laitteille, joissa ei voida käyttää stylus-kynää suurempaa syöttölaitetta.



Kuva 3.2. Esimerkki Unistroke-merkistön käytöstä. [Goldberg and Richardson, 1993]

Unistroke-merkistö koostuu joukosta piirtoja, jotka voidaan antaa kynää nostamatta. Merkistössä on viisi erimallista piirtoa, joiden asentoa ja piirtosuuntaa muutamalla saadaan muodostettua aakkosto. Unistroken kaltaisia vaihtoehtoisia tekstinsyöttötapoja ei juuri ole hyödynnetty interaktiivisilla tussitauluilla. Perinteisiin kirjaimiin verrattuna esimerkiksi Unistroke-merkistön käyttö helpottaisi tekstintunnistusta, sillä käyttäjien väliset erot käsialassa eivät olisi yhtä merkittäviä. Toisaalta uudenlaisen merkistön opettelemista ei tulisi pitää edellytyksenä tussitaulusovelluksen käytölle, vaan vaihtoehtoisia tekstinsyöttötapoja voitaisiin käyttää rinnakkain normaalin aakkoston rinnalla.

3.2.2. Yleisten geometristen kuvioiden tunnistaminen

Kun käyttäjä piirtää vapaalla kädellä interaktiiviselle tussitaululle, suorasta viivasta tulee harvoin todella suora. Tämä voi johtua esimerkiksi käyttäjän käden epätarkoista liikkeistä tai syötteidentunnistuksen laitetason epätarkkuudesta. Interaktiivisia tussitauluja käytetään laajalti kuvioiden tai piirustusten hahmotteluun. Tällaisessa ideoinnissa esimerkiksi viivojen visuaalisen esityksen tarkkuus ei välttämättä ole olennaista. Kuitenkin monissa tapauksissa interaktiivisten tussitaulujen käytössäkin pyritään esityksen tarkkuuteen. Tällöin olisi käytön joustavuuden kannalta olennaista, ettei viivan piirtämistä tarvitsisi toistaa monta kertaa halutun suoruuden saavuttamiseksi. Lisäksi, jos aikaisemmin hahmoteltu kuvio halutaan myöhemmässä vaiheessa muuntaa tarkkaan esitysmuotoon, on tällöin sen uudelleen piirtäminen turhauttavaa.

Pegasus [Igarashi *et al.,* 1997] on piirto-ohjelma, joka on tarkoitettu geometristen 2D-kuvioiden tarkan piirtämisen helpottamiseen. *Interaktiivinen kaunistaminen* (interactive beautification) on Igarashin ja kumppaneiden käyttämä menetelmä, jonka avulla vapaalla kädellä piirretystä suorasta saadaan todella luotua suora. Pegasus laskee annetun syötteen perusteella suunnan, johon tarkka suora piirretään. Kuvan 3.3a esimerkissä käyttäjän vapaalla kädellä hahmottelema punainen syöte ”kaunistetaan” niin, että piirrosta tulee suora. Pegasusta on käytetty geometristen kuvioiden tunnistamiseen Flatland-tussitaulusovelluksessa [Mynatt *et al.,* 2000]. Kuten tekstintunnistuksen yhteydessä, tunnistetulla graafisella kuviolla ei pidä automaattisesti korvata alkuperäistä esitystä. Pegasus esittää pinkin värin ja katkoviivan avulla ehdottamansa suorat, joista käyttäjä voi halutessaan valita haluamansa.

Pegasusta käyttävä Flatland huomioi käyttäjän aikaisemmin piirtämien kuvioiden muodot. Mikäli piirretyistä kuvioista tunnistetaan jokin toistuva tai symmetrinen piirre, antaa Pegasus käyttäjälle ehdotuksia tämän piirteen hyödyntämisestä uusien syötteiden kohdalla. Kuvassa 3.3a Pegasus tunnistaa punaisella näytettävän uuden suoran ja vanhan suoran välisen kulman. Tämän perusteella se ehdottaa myös kolmannen suoran lisäämistä, jolloin kuviosta saadaan muodostettua pyramidin mallinen. Kuvassa 3.3b käyttäjä piirtää punaisella värillä kuvatun uuden suoran aikaisemmin piirretyn kolmion viereen. Pegasus huomioi uuden suoran ja aikaisemman kolmion sivujen geometrisen suhteen. Käyttäjä voi nyt helposti luoda yhteneviä kuvioita olettaen, että Pegasuksen tarjoamat suoravaihtoehdot ovat hyödyllisiä.

(a) (b)

Kuva 3.3. Pegasus ”kaunistaa” (a) ja ehdottaa uusia suoria (b). [Igarashi *et al.,* 2000]

Lisäksi Flatlandissa on käytetty Teddy-nimistä hahmottelujärjestelmää [Igarashi *et al.,* 1999b] luomaan 3D-malleja piirrosten perusteella. Graafisten kuvioiden tunnistustekniikoiden toteuttaminen on erityisen haastavaa silloin, kun pyritään tukemaan tarkoin määrittelemätöntä käyttötapaa. Seuraavassa alakohdassa käsitellään tunnistusta tietyssä sovelluksessa, jolloin mahdollisten graafisten kuvioiden määrä on rajallinen ja tunnettu.

3.2.3. Tunnistus CASE-työkaluissa

Graafisten kuvioiden tunnistusta voidaan hyödyntää jossain tietyssä, tarkoin määritellyssä kontekstissa. Tällöin tunnistettavien kuvioiden määrä on mahdollisesti rajattu. Koska tunnistettavien vaihtoehtojen määrä on pienempi, voidaan tunnistusmekanismia käytettäessä rohkeammin valita mahdollisimman lähellä oleva

vaihtoehto. Väärän tunnistuksen todennäköisyys pienenee. Rajattua kontekstia käytettäessä saadaan tunnistuksesta teoriassa varmatoimisempaa ja tehokkaampaa. Interaktiivisten tussitaulujen sovelluksissa syötteentunnistusta on käytetty *CASE-työkaluissa* (Computer Aided Software Engineering) esimerkiksi UML-kaavioiden tuottamiseen.

CASE-työkalut Knight [Damm *et al.,* 2000] ja SUMLOW [Chen *et al.,* 2003] on tarkoitettu UML-kaavioiden piirtämiseen interaktiivisen tussitaulun avustuksella. Molemmat hyödyntävät syötteiden tunnistusta vertaamalla käyttäjän piirtämiä kuvioita ennalta tarkoin määriteltyihin elementteihin. Knight tunnistaa annetun syötteen heti, kun käyttäjä nostaa kynän taulun pinnasta. Esimerkiksi nelikulmion hahmottelu tulkitaan luokkakaavioita piirrettäessä uudeksi luokaksi, joka piirretään näkyviin käyttäjän hahmotteleman kuvion tilalle. Knightin käyttämällä automaattisella korvaamisella saadaan nopeasti tuotettua standardin mukaisia kaavioita, kunhan käyttäjän piirtämät hahmotelmat ovat tarpeeksi oikean kaltaisia tunnistuksen toimimiseksi.

SUMLOW käyttää erilaista periaatetta tunnistuksessaan. Sovelluksen käyttäjällä on mahdollisuus valita kahdesta rinnakkaisesta näkymästä, sillä hahmoteltu ja formalisoitu kaavio ovat samanaikaisesti käytettävissä. Käyttäjän piirtäessä esimerkiksi sekvenssikaaviota vapaalla kädellä annetut syötteet tunnistetaan ja muutetaan tarkempaan muotoon taustaprosessina. Toiseen näkymään tehty muutos heijastuu välittömästi toiseenkin, jolloin synkronisoinnin kanssa ei jälkikäteen tule ongelmia. Käyttäjä voi vapaasti vaihdella kahden näkymän välillä eri tilanteiden mukaan.

Knight ja SUMLOW molemmat tukevat kaavioiden XML-muotoista tallentamista ja esittämistä. Kun hahmotellut kaaviot on tunnistettu onnistuneesti, voidaan ne tallentaa yleisten standardien mukaisessa muodossa. Tämän jälkeen tiedostoja voidaan käsitellä ongelmitta eri ympäristöissä ja eri ohjelmilla. Tunnistuksen hyödyllisyys tulee esiin juuri tässä, sillä vapaalla kädellä piirrettyjen kaavioiden yhteensovittaminen muiden ohjelmien kanssa olisi liki mahdotonta.

3.3. Tiedon järjestäminen

Interaktiiviset tussitaulut mahdollistavat perinteisiin tauluihin verrattuna huomattavasti tehokkaamman tavan järjestää ja hallita taululla olevaa tietoa. Perinteisellä liitu- tai tussitaululla tiedon siirtäminen täytyy suorittaa pyyhkimällä aikaisempi kirjoitus tai piirros pois ja kirjoittamalla se uudelleen haluttuun paikkaan. Interaktiivisten tussitaulujen avulla tietoa voidaan käsitellä joustavammin. Materiaalin siirtämiseksi ei tarvita erillisiä poisto-operaatioita, vaan halutut objektit voidaan esimerkiksi vain raahata toiseen kohtaan. Jos interaktiivista tussitaulua käytetään muistiinpanojen tai havainnollistavien piirrosten tekemiseen, oletetaan, että taululla olevaa materiaalia voidaan vaikkapa ryhmitellä rajaamalla alueita. Ympyrän piirtäminen taululla olevan materiaalin ympärille tarkoittaa materiaalin kuulumista samaan kokonaisuuteen.

Interaktiivisilla tussitauluilla käytettävien sovellusten tulisikin osata tulkita käyttäjien luonnollisten piirtojen tarkoitus.

Ihmiset käyttävät asioiden ja esineiden hallitsemiseen erilaisia ja hyvin kompleksisiakin tekniikoita. Tiedon järjestämisen suunnittelu ja toteuttaminen interaktiiviselle tussitaululle tuntuu ensin ajateltuna helpolta tehtävältä, mutta intuitiivisen ja tehokkaan tekniikan kehittäminen tiedon järjestämiseen on varsin haastava tutkimusaihe. Tällaisen tekniikan olemassaolo on tussitaulusovelluksen käytettävyyden kannalta kuitenkin olennainen asia. Käyn seuraavaksi läpi kaksi aikaisemmissa tutkimuksissa esitettyä tapaa hallita tietoa interaktiivisella tussitaululla ja punnitsen niiden heikkouksia ja vahvuuksia.

3.3.1. Käyttäjän määrittelemät rajat

Liveboardille kehitetystä Tivoli-sovelluksesta on olemassa versio, jonka avulla on tutkittu erilaisten tiedonorganisointitapojen toimivuutta interaktiivisilla tussitauluilla [Moran *et al.,* 1997]. Tavoitteena suunnittelussa on ollut käyttää taulun tarjoamaa tilaa mahdollisimman tehokkaasti. Interaktiivisten tussitaulujen etuna perinteisiin tauluihin verrattuna on se, että virtuaalisen tilan määrä on ääretön. Tätä aspektia käsitellään kohdassa 3.4.

Interaktiivisten tussitaulujen materiaalin hallinnassa olennaista on taululle kerralla mahtuvan tiedon määrä ja sen esitysmuoto. Materiaalin ryhmittely on keino, jonka avulla nämä seikat voidaan huomioida käytössä. Tivoli antaa vastuun materiaalin järjestämisestä täydellisesti käyttäjälle. Käyttäjän tulee jakaa tussitaulun ala haluamiinsa visuaalisiin kokonaisuuksiin, joita voidaan tämän jälkeen käsitellä ryhminä. Rajattuun ryhmään voidaan kohdistaa tiedon järjestämiseen tarkoitettuja operaatioita. Tivolissa ei ole erillistä elettä tai komentoa rajojen määrittelyyn. Normaalisti luodut piirrot, kuten esimerkiksi suorat, nelikulmiot tai ympyrät, määrittelevät alueen. Nämä piirrot näkyvät taululla samalla tavalla muiden kirjoitusten ja piirrosten kanssa.

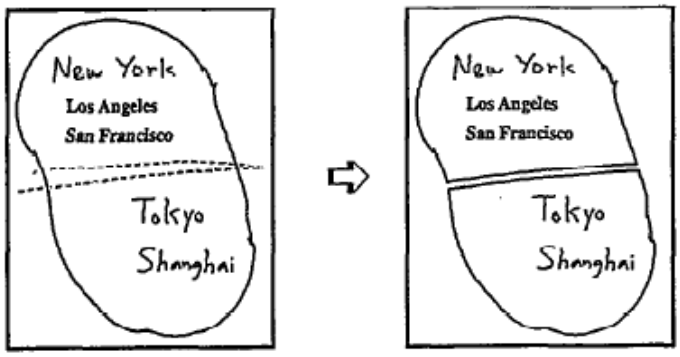
Käyttäjän piirtämät rajat voidaan jakaa kahteen kategoriaan; taulun reunasta reunaan ulottuviin pysty- tai vaakasuoriin viivoihin ja sulkeuman määritteleviin piirtoihin, kuten neliöihin tai ympyröihin. Koko taulun levyiset tai korkuiset viivat ovat tehokas keino jakaa materiaalia. Taulun vasempaan reunaan voidaan esimerkiksi rajata pystysuora kaistale, jonka sisälle on kirjoitettu listamuotoisesti tekstiä. Koska tämä alue on nyt oma rajattu kokonaisuutensa, voidaan siihen kohdistaa jokin operaatio. Tekstit pystytään esimerkiksi lajittelemaan aakkosjärjestyksen mukaan.

Sulkeumien käyttö on taulun pituisten viivojen piirtämiseen verrattuna huomattavasti hienovaraisempaa, sillä käyttäjä voi vapaalla kädellä rajata juuri haluamansa kokoisen alueen. Kun käyttäjä valitsee jonkin sulkeuman sisällä olevan objektin, valittuun aktiiviseen alueeseen kuuluvat kaikki tämän sulkeuman sisällä olevat objektit. Tämän jälkeen käyttäjä voi antaa esimerkiksi jonkin lajitteluoperaation, joka

kohdistuu kaikkiin sulkeuman rajaamiin objekteihin. Tivoli etsii objektia mahdollisesti ympäröivän suljetun kuvion automaattisesti, kun jokin piirretty kuvio valitaan.

Tivoli sisältää lukuisia eri operaatioita rajatun alueen sisältämien objektien ryhmittelyyn ja muokkaamiseen. Viivoin rajatulle alueelle voidaan siirtää sen ulkopuolelta materiaalia. Tällöin alueen koko suhteutetaan automaattisesti uuden materiaalin vaatiman tilan mukaan.

Alueita voidaan myös muokata komentojen avulla. Sulkeuma voidaan jakaa kahdeksi erilliseksi alueeksi piirtämällä edestakainen viiva alueen lävitse (kuva 3.4). Samoin kaksi aluetta voidaan yhdistää yhdeksi piirtämällä niiden välille sulkeuma, joka koskettaa kumpaakin aluetta.



Kuva 3.4. Esimerkkejä Tivolissa ryhmittelyyn käytettävistä alueista. [Moran *et al.,* 1997]

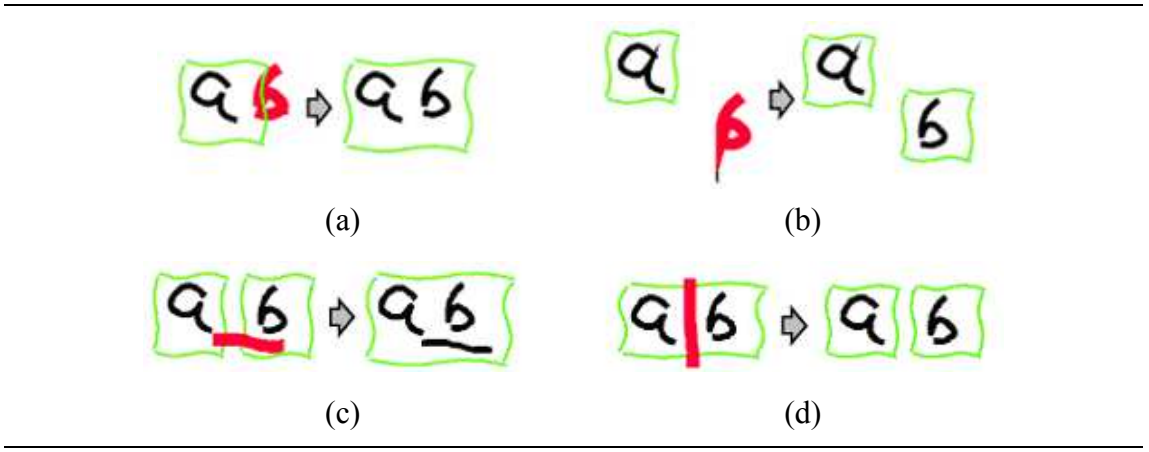
Kuten kynällä määriteltävien luonnollisten komentojen antamisessa yleensä, käyttäjältä vaaditaan erilaisten eleiden hyvää hallintaa. Tivolin suunnittelussa onkin tiedostettu, ettei sovelluksen tehokas käyttö onnistu ilman opettelua ensikertalaiselta. Lisäksi käyttäjältä vaaditaan tiedon järjestämiseksi Tivolissa aina toimenpiteitä. Vaikka ihminen mieltää vierekkäin kirjoitetut tekstit automaattisesti yhteen kuuluviksi, sovellukselle tämä pitää kertoa erikseen. Tähän lähestymistapaan liittyvät näkökohdat on tiedostettu. Eksplisiittisen rajojen osoittamisen sijaan voitaisiin pyrkiä selvästi yhteen kuuluvien tekstien tai kuvioiden automaattiseen ryhmittelyyn.

3.3.2. Automaattiset lohkot

Flatland käyttää materiaalin järjestämiseen automaattisesti luotavia *lohkoja* (segments) [Mynatt *et al.,* 1999; Igarashi *et al.,* 2000]. Pohjana on Mynattin [1999] tekemä tutkimus perinteisten tussitaulujen käytöstä yksittäisissä toimistohuoneissa. Kahden viikon seurantajakson aikana tehtyjen havaintojen pohjalta pyrittiin tunnistamaan tussitaulujen todellisia käyttötarkoituksia ja tapoja hallita materiaalia. Tutkimus-tuloksissa todettiin, että useat perinteisen tussitaulun käyttäjät jakavat piirtoalan lohkoihin, joista jotkut ovat pysyviä. Esimerkiksi tulevien työtehtävien listaamiselle monet rajaavat oman alueen taulusta. Toisessa reunassa voi taas olla vaikkapa sekalaisia

muistiinpanoja ja keskellä osio enemmän piirtotilaa vaativien suunnitelmien tekemiselle. Flatlandia suunniteltaessa on pyritty tukemaan tätä luonnollista jakoa myös interaktiivisten tussitaulujen yhteydessä.

Flatlandissa on periaatteena, ettei käyttäjän tarvitse aina eksplisiittisesti osoittaa materiaalin yhteenkuuluvuutta. Organisoinnin etukäteen tapahtuvaa suunnittelua ei tarvita, vaan tieto pyritään järjestämään automaattisesti käytön yhteydessä. Kun käyttäjä piirtää uuden kuvion, tutkitaan ohjelmallisesti, mihin lohkoon syöte tulisi sijoittaa. Lohkot ilmaistaan käsin piirretyn näköisillä vihreillä nelikulmioilla. Mikäli syötettä ei anneta jo olemassa olevan lohkon sisään tai tarpeeksi lähelle sitä (kuva 3.5a), luodaan uusi lohko (kuva 3.5b). Näin käyttäjän ei tarvitse ensin suurentaa olemassa olevaa lohkoa uuden materiaalin mahduttamiseksi samalle alueelle. Samoin vältytään tilanteelta, jossa uuden lohkon muodostamiseksi tarvitsisi kirjoittamisen jälkeen tehdä erikseen lohkon luomista tarkoittava ele, esimerkiksi nelikulmion piirto. Flatlandissa on lisäksi erilaisia eleitä, joiden avulla olemassa olevia lohkoja voidaan käsitellä. Kahden lohkon yhdistäminen tapahtuu piirtämällä lohkojen välille viiva (kuva 3.5c), ja erottaminen taasen piirtämällä lohkon halki pystyviiva (kuva 3.5d).



Kuva 3.5. Lohkoon lisääminen (a), uuden lohkon luominen (b), lohkojen yhdistäminen (c) ja lohkon jakaminen (d) Flatlandissa. [Igarashi *et al.,* 2000]

Flatlandissa viimeksi käsitelty lohko on aktiivinen. Jotta käyttäjä erottaisi tämän muista, suurennetaan aktiivisen lohkon kokoa hetkellisesti, ja sen reunojen väri on muita lohkoja kirkkaampi. Mynatt huomasi käyttäjätutkimuksessaan [1999], että monet pitävät perinteistä tussitaulua sijaismuistinaan, johon kirjataan ylös tärkeitä asioita. Tämä huomioiden Flatlandin suunnittelussa päädyttiin siihen, että lohkoja ei voi sijoittaa toistensa päälle. Kun yhtä lohkoa raahataan toista kohti, eteen jäävää lohkoa siirretään alta pois. Mikäli törmäyksen kohteena oleva lohko osuu taulun reunaan, pienennetään alle jäävän lohkon kokoa.

Flatlandin kehittämisen yhteydessä kokeiltiin visualisointivaihtoehtoa, jossa lohkojen reunoja ei näytetty lainkaan. Koska lohkojen reunat kuitenkin antavat tietoa taululla olevan materiaalin tilasta ja ohjaavat annettavia syötteitä, päätettiin reunat pitää

jatkuvasti näkyvissä. Flatland sisältää monia uusia interaktiivisille tussitauluille tarkoitettuja vuorovaikutustapoja, joita ei aikaisemmin oltu tutkittu juuri lainkaan. Vaikka Flatland on suunnattu yksittäisen ihmisen henkilökohtaiseen käyttöön, voidaan siinä esiteltyjä tekniikoita hyödyntää laajemminkin esimerkiksi ryhmätapaamisiin tai kokouksiin liittyvissä skenaarioissa.

Flatlandin vahvuutena, ja toisaalta myös mahdollisena heikkoutena, on materiaalin dynaaminen järjestäminen. Mikäli käyttäjä kokee automaattisesti luotavat lohkot hyödyllisiksi ja työskentelyä helpottaviksi, ovat ne mitä mainioin tapa tiedon hallintaan interaktiivisella tussitaululla. Toisaalta on käyttäjiä, joiden tottumukset tiedon käsittelystä ja visuaalisesta mieltämisestä ovat tavallisuudesta poikkeavia, jolloin automaattinen tekniikka materiaalin järjestämiseen saattaa olla häiritsevä ja työskentelyä rajoittava tekijä. Erilaisten tekniikoiden väliltä valittaessa on harvoin mahdollista täydellisesti miellyttää kaikkia osapuolia, ellei käyttäjällä ole mahdollisuutta mukauttaa vuorovaikutustapaa omien mieltymystensä perusteella. Tämän vuoksi erilaisten tiedonjäsentämismallien parissa tehtävä tutkimus on olennaista interaktiivisten tussitaulujenkin yhteydessä, jotta voidaan luoda mahdollisimman monen käyttäjän tarpeisiin sopivia työkaluja.

3.4. Virtuaalisen tilan hallinta

Interaktiiviset tussitaulut mahdollistavat periaatteessa äärettömän työskentelytilan käytön. Kun taululla näkyvä tila täyttyy syötteistä, on aina mahdollista saada lisää vapaata piirtotilaa. Olennainen kysymys onkin se, kuinka tätä vapautta tilan käytössä voidaan hyödyntää niin, että käyttäjä pystyy työskentelemään tehokkaasti hämmentymättä äärettömästä tilasta. Lisäksi mahdollinen kolmiulotteisuus tuo uusia aspekteja tilan käyttöön. Kolmiulotteisuutta käytettäessä voidaan toteuttaa syvyys-vaikutelmaan tai päällekkäisyyteen perustuvia sovelluksia. Tämän tutkielman puitteissa en kartoita tarkemmin kolmiulotteisuuteen perustuvia ratkaisuita, joiden avulla voitaisiin esimerkiksi käsitellä tai mallintaa erimuotoisia kappaleita.

Mahdollisuudet virtuaalisen tilan hallinnan osalta ovat suuret, mutta tilanne ei käytännössä ole kovin yksiselitteinen. Erilaisia lähestymistapoja virtuaalisen tilan hallintaan on olemassa lukuisia. Silti helppokäyttöisiä, tehokkaita ja interaktiivisten tussitaulujen vaatimuksiin sopivia tekniikoita on rajoitetusti. Seuraavaksi vertailen kolmea ehdotettua vaihtoehtoa piirtotilan hallintaan. Kaikki ratkaisuehdotukset eivät suoranaisesti ole interaktiivisille tussitauluille suunnattuja, mutta niiden hyödyntäminen olisi teoriassa mahdollista.

3.4.1. Zoomaaminen

Pad++ [Bederson and Hollan, 1994] on alun perin visualisointitekniikka, mutta idea sen käytöstä interaktiivisten tussitaulujen hallinnassa on tullut ilmi useissa yhteyksissä [Mynatt *et al.,* 1999; Moran *et al.,* 1997; Guimbretière *et al.,* 2001; Damm *et al.,* 2000].

Bedersonin ja Hollanin kehittämä tekniikka pohjautui Pad-ohjelmaan [Perlin and Fox, 1993]. Visualisointitekniikoiden tutkimukseen Pad++:lla on ollut suuri vaikutus. Se on luotu vaihtoehdoksi perinteisille ikkunoille ja ikoneille, joita käytetään työpöytä-metaforan mukaisissa järjestelmissä. Pad++ perustuu periaatteessa äärettömään zoomausmahdollisuuteen, jonka avulla käyttäjä pystyy hahmottamaan suuriakin graafisia informaatiokokonaisuuksia.

Vaihtaakseen tyhjään työskentelyalueeseen esimerkiksi piirto-ohjelmassa tarvitsee käyttäjän vain muuttaa näkyvää aluetta joko siirtymällä samalla tarkkuustasolla eri kohtaan tai zoomaamalla eri tarkkuustasolle. Nopean yleissilmäyksen sisällöstä saa zoomaamalla kauemmas. Pad++:ssa käytetään semanttisen zoomauksen periaatetta, jossa objektin yksityiskohdat lisääntyvät sitä mukaa, mitä lähempänä työskennellään. Jos taas zoomataan kauemmaksi, objekteja ei pelkästään skaalata pienemmiksi, vaan tieto näytetään eri esitysmuodossa. Tätä samaa periaatetta on käytetty myös MuSE-editorissa [Furnas and Zhang, 1998]. Pad++:ssa pyritään myös tuomaan esille objektien välisiä yhteyksiä. Hypertekstimuotoisten dokumenttien välillä on olemassa linkkien muodostama semanttinen vanhempi-lapsi-yhteys. Tämän havainnollistamiseksi linkkiä klikattaessa uusi sivu avautuu näkyviin pienempänä, ja se sijoitetaan näkyviin samanaikaisesti vanhemman sivun kanssa.

Flatlandissa ei ole käytetty Pad++:n kaltaista vieritys- tai zoomausmahdollisuutta, sillä se saattaisi aiheuttaa lohkojen hukkaamista virtuaaliseen tilaan [Mynatt *et al.,* 1999]. Myös Interactive Muralin suunnitteluvaiheessa päätettiin käyttää taulupinnan kokoista tilaa, jota ei voi vierittää tai liikuttaa [Guimbretière *et al.,* 2001]. Yksittäisiä objekteja pystyy skaalaamaan erikokoisiksi, mutta zoomaus ei ole mahdollista koko taulun laajuudella. Guimbretière *et al.* perustelevat ratkaisuaan sillä, että rajaton zoomausmahdollisuus voi vaikeuttaa työskentelyä ryhmässä. Erityisesti jos mukana on Pad++:n kaltainen semanttinen zoomaus, joka vaihtaa objektien esitysmuotoa etäisyyden perusteella, joutuvat kaikki käyttämään samaa työskentelyetäisyyttä. Navigointi usealla tasolla samanaikaisesti on tiedostettu ongelma [Moran *et al.,* 1997].

Vaikka Pad++:n esittelemää toiminnallisuutta ei sellaisenaan ole hyödynnetty interaktiivisten tussitaulujen tilanhallinnassa, olisi tämä käytännössä hyvinkin mahdollista. Työskentelyetäisyyden muuttaminen voidaan toteuttaa esimerkiksi eleiden avulla. Käyttäjän kannalta zoomaamisen mahdollistava hallintatekniikka saattaa aluksi tuntua vaativalta, mutta käytön yhteydessä tehokkuus luultavasti kasvaisi.

CASE-työkalu Knightissa [Damm *et al.,* 2000] käyttäjä pystyy vaihtamaan työskentelyetäisyyttä zoomauksen avulla rajattomasti. Lisäksi Knightissa on erillinen ikkuna, joka näyttää pienoiskoossa koko työskentelyalueen. Tämän *tutkaikkunan* (radar window) avulla pystyy saamaan kokonaiskuvan tussitaululla olevasta materiaalista. Ikkunan pienoiskuvan kautta voi vierittää tai zoomata itse työskentelyalaa. Yleiskuvan saamiseksi edellä mainitun kaltaiset ikkunat ovat käyttökelpoisia, sillä niiden avulla

voidaan hahmottaa objektien välisiä spatiaalisia suhteita, havainnoida, miten muutokset vaikuttavat näkymän ulkopuolella sekä hahmottaa oma näkymä suhteessa koko työskentelyalueeseen [Cox *et al.*, 1998].

Yleiskuvan tarjoavien ratkaisuiden käyttöä zoomaamiseen verrattuna on tutkittu laajemminkin tiedon visualisointiin liittyen. Hornbæk *et al.* [2002] ovat tutkineet, kuinka käyttäjät onnistuivat navigoimaan karttajärjestelmässä yleiskuvaikkunan kanssa ja ilman. Käyttäjistä 80 % piti yleiskuvaikkunan käyttöä hyödyllisenä, sillä se auttoi heitä navigoimaan ja ymmärtämään sijaintinsa kartalla. Tehtävien onnistumisprosenteissa ei eri tekniikoiden välillä ollut merkittävää eroa. Kuitenkin testit osoittivat, että käyttäjät suoriutuivat tehtävistä nopeammin ilman yleiskuvaikkunaa. Plumlee ja Ware [2006] ovat verranneet zoomaamisen ja useiden ikkunoiden käyttöä merenpohjaa mallintavan järjestelmän käytössä. Testissä käyttäjät tekivät enemmän virheitä käyttäessään zoomaamista. Plumlee ja Ware päättelevät, että järjestelmän kuvaaman tiedon kompleksisuudesta johtuen zoomaamista käytettäessä käyttäjät eivät kykene muistamaan tietoa tarpeeksi tarkasti. Sen sijaan useaa ikkunaa hyödynnettäessä käyttäjät voivat nopeasti tarkistaa yksityiskohtia eri tarkkuustasoa käyttävistä ikkunoista.

Suurimmat ongelmakohdat zoomaukseen perustuvien vuorovaikutustekniikoiden yhteydessä liittyvät informaatiokokonaisuuksien hallintaan. Käyttäjälle saattaa olla vaikeaa luoda yhteyksiä asioiden välille, mikäli zoomaustekniikka ei mahdollista helppoa tapaa saada yleissilmäystä työskentelyalasta. Jo lyhyen työskentelyn jälkeen käyttäjät voivat olla eksyksissä informaation keskellä eivätkä tiedä, miten löytää etsittävä tieto [Pook *et al.*, 2000]. Toisaalta taululla olevan materiaalin käsittely samanaikaisesti työskentelyetäisyyden hallinnan kanssa voi muodostua haastavaksi tehtäväksi esimerkiksi silloin, kun objekteja tulisi siirtää taulun reunasta toiseen.

3.4.2. Kerroksisuus

Perinteisen tussitaulun käytössä työskentelytilan rajoitukset tulevat esiin taulua rajaaviin reunoihin törmättäessä. Tämän lisäksi syvyysvaikutelman puuttuminen on luonnollinen asia. Interaktiivisten tussitaulujen virtuaalinen tila mahdollistaa työskentelyn useammalla kuin yhdellä tasolla, sillä käyttäjä pystyy käsittelemään päällekkäin olevia elementtejä. Tätä vuorovaikutusmahdollisuutta on tutkittu jo aikaisemmin työpöytätietokoneiden yhteydessä.

Perinteisen graafisen käyttöliittymän korvaajaksi Kramer [1994] on ehdottanut kerroksisuuteen perustuvaa Translucent Patches -ratkaisua. Kramerin kehittelemässä vuorovaikutustekniikassa tärkeimpiä elementtejä ovat vapaasti muotoiltavat läpinäkyvät *laput* (patches), jotka korvaavat perinteiset, läpinäkymättömät ikkunat. Kerrokset ovat läpikuultavia, ikään kuin ohuita papereita päällekkäin aseteltuina. Alimpana oleva materiaali häipyy lopulta näkyvistä, kun sen päälle asetetaan tarpeeksi monta kerrosta.

Kramerin idea pohjaa pitkälti arkkitehtien perinteiseen työskentelytapaan, jossa pöydälle asetetaan useita läpinäkyviä papereita. Ohjelmalliseen kerrosten hallintaan

Kramer esittelee useita vaihtoehtoja. Uuden kerroksen ei tarvitse kattaa koko näyttö- tai taulualaa, vaan uusi kerros voidaan sijoittaa tiettyyn, tarkasti haluttuun kohtaan. Kerrosten järjestystä ja muotoa voidaan vaihtaa tarpeen mukaan erilaisin eleiden avulla tapahtuvin komennoin.

Yleisemmällä tasolla Kramer pyrkii erottamaan tiukan yhteyden, joka käyttöliittymässä yleensä on olemassa ikkunoiden, niiden esittämän sisällön sekä taustalla olevan ohjelman välillä. Tähän ratkaisuna ovat siis kerroksittain toimivat laput, joita voidaan vapaasti sijoittaa haluttuihin piirtoalan kohtiin. Käyttäjä voi lappujen avulla rajata samaan ryhmään kuuluvia objekteja, joille voidaan suorittaa haluttuja operaatioita. Lappujen avulla tietoa voidaan käsitellä vapaammin kuin perinteisiä ikkunapohjaisia ohjelmia käyttäen. Perinteistä käyttöliittymärakennetta noudattavissa ohjelmissa tiedon käsittely perustuu siihen, että yhdessä ikkunassa muokataan esimerkiksi yhtä tekstitiedostoa. Kramerin ideana on sallia tiedon vapaampi muokkaaminen, jolloin jokaisen tiedoston käsittelyä varten ei tarvitse avata omaa ohjelmaa ja ikkunaa. Eri lapuilla tieto voidaan esittää eri tavoin. Toiselle voidaan piirtää kuvioita ja toisella tehdä laskutoimituksia. Tämä erilaisten operaatioiden suorittaminen samanaikaisesti yhdellä työskentelyalalla on etu verrattuna perinteisiin ohjelmiin, joita voidaan käyttää vain tiettyjen, tarkoin määriteltyjen työtehtävien suorittamiseen.

Kramerin kerroksisiin lappuihin perustuva idea on tullut esille useissa interaktiivisia tussitauluja käsittelevissä tutkimuksissa [Mynatt *et al.,* 1999; Moran *et al.,* 1997; Guimbretière *et al.,* 2001]. Kyseisen visualisointitekniikan kiehtovuudesta ja mahdollisesta potentiaalista huolimatta sitä ei käytännössä ole testattu interaktiivisten tussitaulujen käytössä. Tussitaulusovelluksiin mukautettuna kerroksinen työala olisi hyödyllisimmillään varmasti suunnittelutyössä, johon Kramer suunnitteluprosessinsa pohjasi. Perinteisten arkkitehtuurisovellusten lisäksi esimerkiksi UML-kaavioiden piirtämiseen kerroksittaisuus toisi lisämahdollisuuksia, sillä luokkakaavion päälle voitaisiin hahmotella erilaisia, tarkennuksia sisältäviä vaihtoehtoja. Kerroksisen tilanhallinnan käytettävyyden selvittäminen muissa käyttötarkoituksissa, kuten yksinkertaisen tiedon jäsentämisessä tai muistiinpanojen teossa, vaatisi todellisia sovelluksia ja käyttäjätestausta.

3.4.3. Käännettävä taulu -metafora

Edellä käsitellyt tilanhallintaratkaisut, zoomaaminen ja kerroksisuus, hyödyntävät interaktiivisten tussitaulujen mahdollisuuksia monimutkaisemman vuorovaikutustekniikan avulla. Toisaalta tilanhallintaratkaisuita pohdittaessa voidaan keskittyä myös siihen, miten ihmiset ovat tottuneet käyttämään perinteisiä tussi- ja liitutauluja. Monissa tauluissa on kaksi puolta, jotka saadaan esille kääntämällä taulu. On olemassa myös jaloillaan seisovia telineitä, joissa on suuri lehtiön tapainen paperinippu, josta voidaan kääntää tarvittaessa puhtaita sivuja esiin. Näiden tottumusten tukeminen myös

interaktiivisten tussitaulujen käytössä helpottaa ihmisten sopeutumista ja totuttautumista uusien laitteiden käyttöön.

Mynatt *et al.* [1999] ovat päätyneet käyttämään Flatlandissa virtuaalisen tilan hallintaan käännettävä taulu -metaforaa. Flatlandin yhteydessä kaksipuolisuus on viety kuitenkin elektronisesti pidemmälle, sillä käyttäjä voi ”kääntää” taulua loputtomasti uutta tilaa saadakseen. Näin voidaan tilapäisesti saada lisätilaa esimerkiksi silloin, kun käyttäjä ei halua tyhjentää taulua aikaisemmista merkinnöistä. Käytännössä kääntäminen tapahtuu tekemällä taulun vasempaan tai oikeaan reunaan kohdistuva raahaamisele. Puhtaan taulutilan saaminen on nopeaa, sillä yksi ele riittää.

Tutkimuksessa [Mynatt, 1999] potentiaaliset interaktiivisten tussitaulujen käyttäjät ilmaisivat pelkonsa siitä, että käytössä olevan tilan kasvaessa he hukkaavat kirjoittamansa tai piirtämänsä materiaalin, eivätkä enää saa paikannettua sitä taululta. Flatlandin suunnittelussa päädyttiin taulun kääntämiseen, koska materiaali on aina löydettävissä kääntämällä taulu tarpeeksi useasti takaisin päin. Mittakaavaa Flatlandissa ei pysty muuttamaan esimerkiksi zoomaamalla, sillä se saattaisi aiheuttaa vaikeuksia tiedon etsinnässä. Kun sivujen määrä kasvaa, tulee jatkuvasta kääntämisestä kuitenkin epäkäytännöllinen tapa hallita suurta tietomäärää. Flatlandissa pyritäänkin ensisijaisesti tukemaan yksittäisen taulusivun käyttöä.

Flatlandin lisäksi käännettävä taulu -metaforaa on käytetty Tivolissa [Moran *et al.*, 1997]. Käännettävä taulu -metafora on intuitiivinen ja näin käyttäjille helposti omaksuttava. Tämä tilanhallintatapa sopii käyttötarkoituksiin, joissa taululla kerralla esitettävä tietomäärä on rajallinen. Lisätilaa ei saada joustavasti taulun reunojen ulkopuolelta, vaan materiaalia on joko järjesteltävä tehokkaammin tai vaihtoehtoisesti skaalattava pienemmäksi. Kääntämällä taulu saadaan lisätilaa, mutta tällöin kahdella eri taulusivulla olevan materiaalin välille on vaikea luoda yhteyttä. Objekti voidaan sijoittaa kahden taulusivun taitekohtaan, jolloin siitä näkyy puolet kummallakin sivulla. Taulun kääntäminen ei kuitenkaan ole joustava tapa hallita tietoa tällaisissa tapauksissa, sillä käyttäjän tulisi jatkuvasti vaihdella taulusivujen välillä. Suurten tietomäärien esittämiseen ja hallitsemiseen esimerkiksi tilan vierittäminen tai zoomaaminen on tehokkaampi ja käyttäjäystävällisempi tekniikka.

3.5. Yhteenveto

Tässä luvussa olen käsitellyt keskeisimpiä asioita, jotka ovat aikaisemmin tulleet esille interaktiivisia tussitauluja koskevassa tutkimuksessa. Olen koonnut edellisissä kohdissa esiteltyjen ominaispiirteiden perusteella suuntaviivat, jotka tulisi interaktiivisten tussitaulujen käyttöliittymä- ja vuorovaikutusratkaisuja pohtiessa pitää mielessä:

1. *Syötteiden antaminen*. Perinteisistä GUI-sovelluksista tutut painikkeet soveltuvat huonosti interaktiivisilla tussitauluilla käytettäviksi. Eleet ja

esimerkiksi kynän painikkeen avulla tapahtuva komentojen antaminen ovat tehokkaampia tapoja.

2. *Syötteiden käsittely ja tunnistaminen*. Annettujen piirtojen tehokkaan hyödyntämisen mahdollistamiseksi syötteet on usein tunnistettava ja muutettava tietokoneen ymmärtämään muotoon. Tekstin ja graafisten kuvioiden tunnistuksen avulla syötteiden käyttömahdollisuudet kasvavat.
3. *Tiedon järjestäminen*. Tiedon syöttäminen interaktiiviselle tussitaululle on vapaamuotoista, mutta materiaalin käsittelemiseen ja järjestämiseen tarvitaan silti tehokas keino. Käyttäjän eksplisiittisesti määrittelemät rajat tai vaihtoehtoisesti implisiittisesti toimiva järjestäminen ovat käyttötarkoituksesta riippuen molemmat käyttökelpoisia vaihtoehtoja.
4. *Virtuaalisen tilan hallinta*. Ääretöntä virtuaalista tilaa voidaan hallita esimerkiksi zoomaamalla tai kerrosten avulla, jolloin työskentelyalueen laajuutta voidaan hyödyntää tehokkaasti. Toisaalta voidaan myös keskittyä käyttämään kerralla näkyvää tilaa ja korkeintaan antaa mahdollisuus taulun kääntämiseen lisätilan hankkimiseksi.

Interaktiivisten tussitaulujen käyttöliittymän tulee olla mukautettu laitteen fyysisten ominaisuuksien asettamiin vaatimuksiin. Tässä luvussa olen tiivistänyt tärkeimmät interaktiivisten tussitaulujen sovellusten käytettävyyteen vaikuttavat asiat. Olen pyrkinyt pitämään käsittelyn mahdollisimman yleisellä tasolla ja näin löytämään tärkeimmät eri sovellusalueita yhdistävät piirteet.

4. Kaavioeditorit

Tämän luvun tarkoituksena on antaa lukijalle käsitys *kaavioeditoreista* (diagram editors) ja niihin liittyvästä tutkimuksesta. Esittelen, mitä kaavioeditorit ovat ja missä yhteyksissä niitä käytetään. Aluksi teen rajanvedon yleisten graafisten editorien ja kaavioeditorien välille. Käsittelen kaavioeditorien historiaa ja sitä, miksi kaavioiden käyttö tiedon mallintamisessa on viime aikoina yleistynyt huomattavasti. Kerron lisäksi lyhyesti kaikkia kaavioeditoreja yhdistävästä piirteestä, visuaalisten kielten hyödyntämisestä.

Käsittelen tässä luvussa tarkemmin kahta kaavioeditoreihin keskeisesti liittyvää tekijää. Esittelen graafiteorian käyttöä kaavioiden visualisoinnin yhteydessä. Lopuksi jäsennän eri kaavioeditoreja sen perusteella, kuinka vapaasti ne sallivat käyttäjän muokata kaavioesitystä. Kaavioeditorit voidaan tämän piirteen perusteella jakaa syntaktisiin ja vapaaseen käyttöön pohjautuviin sovelluksiin.

4.1. Graafiset editorit

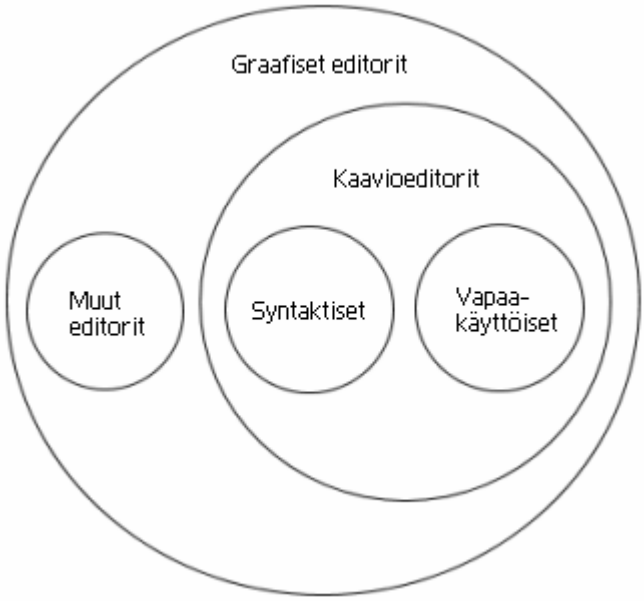
Graafiset editorit ovat sovelluksia, jotka esittävät tutut objektit visuaalisessa muodossa ja mahdollistavat esityksen suoran muokkaamisen [Vlissides and Linton, 1990]. Graafisilla editoreilla voidaan käsitellä esimerkiksi tekstiä, kuvia, 3D-malleja sekä taulukoita. Graafiset editorit ovat käyttökelpoisia hyvinkin erilaisilla sovellusalueilla ja tästä johtuen niiden hyödyntämismahdollisuudet ovat käytännössä rajattomat.

Ensimmäisiä editorisovelluksia käytettiin lähinnä tekstimuotoisen tiedon muokkaamiseen. Komentopohjaisiin sovelluksiin verrattuna tekstieditorit mahdollistivat joustavamman ja tehokkaamman tavan käsitellä tekstiä. Työpöytätietokoneiden kehittyessä tiedonkäsittelykapasiteetti kasvoi ja editorisovelluksilla pystyttiin käsittelemään yhä monimutkaisempia esityksiä. Graafisten käyttöliittymien yleistyttyä editorisovellukset, kuten Microsoftin Word ja Paint, tulivat osaksi käyttöjärjestelmien mukana toimitettavia ohjelmistopaketteja.

Graafisten editorien kehittyminen on ollut vahvasti sidoksissa suorakäyttöisten ohjelmien yleistymiseen. Suorakäyttöisyyden tunnusomaiset piirteet, kuten jatkuvasti käsiteltävissä olevat objektit ja välitön palaute, sopivat erinomaisesti graafisissa editoreissa käytettäviksi vuorovaikutustavoiksi. Graafisen esityksen muokkaamiseen voidaan käyttää esimerkiksi objektien osoittamista, raahaamista ja venyttämistä [Read and Marlin, 1996]. Lisäksi suorakäyttöisyys mahdollistaa elekomentojen käytön editorisovelluksissa.

Määritelmänä ”graafiset editorit” on varsin laaja. Siihen voidaan lukea kuuluvaksi sovelluksia, joita voidaan käyttää esimerkiksi teknisten kuvien tai taiteellisten esitysten piirtämiseen, musiikin säveltämiseen ja piirilevyjen suunnitteluun [Vlissides and Linton, 1990]. Graafisten editorien jaottelu niiden käyttötarkoituksen perusteella on

sovellusten suuresta määrästä johtuen liki mahdotonta. Kuitenkin graafisten editorien käsitettä voidaan pyrkiä tarkentamaan tarkastelemalla sovellusten tapaa käsitellä graafista esitystä muilla tasoilla visuaalisuuden lisäksi. Kuvassa 4.1 on karkeasti jäsennetty editorisovellukset sen perusteella, mahdollistavatko ne tiedon sisäisen esittämisen.



Kuva 4.1. Jäsennys graafisista editoreista ja kaavioeditoreista

Kuvassa 4.1 graafiset editorit on jaettu kaavioeditoreihin ja muihin editoreihin. Kaavioeditorit ovat graafisia editoreja, jotka on suunnattu ennalta määriteltyjen kuvioiden esittämiseen, ja jotka lisäksi ”ymmärtävät” esityksen merkityksen jollain tasolla [Minas, 2002]. Kuvassa olevista termeistä kaavioeditoreja käsitellään tarkemmin kohdassa 4.3. Syntaktisten ja vapaakäyttöisten kaavioeditorien eroavaisuuksia pohditaan kohdassa 4.6. Muihin editoreihin voidaan lukea kuuluviksi esimerkiksi yksinkertaiset piirto-ohjelmat. Nämä ohjelmat esittävät käyttäjän piirtämät kuviot, jotka periaatteessa koostuvat ainoastaan halutun värisistä pikseleistä tietyllä kohdalla näyttöä. Kuvioilla ei ole esimerkiksi keskinäisiä suhteita, eikä piirto-ohjelma rajoita käyttäjän määrittelemien kuvioiden muotoa millään tavoin.

4.2. Kaaviot visuaalisten kielten osana

Läpi historian ihmiset ovat käyttäneet kaavioita erityyppiseen kommunikointiin. Näitä *visuaalisia kieliä* (visual languages) on käytetty useilla oppialoilla ja useissa työtehtävissä. Esimerkkejä ovat arkkitehtien ensimmäiset hahmotelmat, joissa käytetty kieli ei ole kovin tarkoin kuvattua. Lopulliset tekniset määritelmät arkkitehdit muodostavat käyttäen tarkkaa ja ennalta määriteltyä merkintätapaa. Niin ikään omaa visuaalista kieltä voidaan käyttää musiikin, tilakaavioiden, matemaattisten kaavojen ja tiekarttojen esittämiseen. [Chok and Marriott, 1998]

Visuaalisten kielten lauseet, esimerkiksi kaaviot, voidaan usein mieltää graafisiksi objekteiksi, kuten ympyröiksi ja nuoliksi, joita yhdistää spatiaalinen suhde [Bardohl, 1997]. Spatiaalinen suhde voi määritellä toisen objektin toisen yläpuolella olevaksi tai vaikka toiseen objektiin kuuluvaksi. Ilman visuaalisen kielen määrittelyä erityyppisillä kaavioilla ei voi olla ilmaisuvoimaa. Jotta visuaalisella kielellä pystyttäisiin ilmaisemaan asioita, tulee ensin kartoittaa aihealueeseen kuuluvat käsitteet, joiden päälle visuaalinen kieli rakentuu. Käsitteellistämisen pohjalta erilaiset kaavioelementit ja niiden väliset suhteet voidaan määritellä ja merkityksellistää.

Visuaalisten kielten rakenteita käsiteltäessä ei olla tekemisissä pelkästään graafisten kuvioiden kanssa [Citrin and McWhirter, 1995]. Visuaalinen kieli mahdollistaa syntaksin ja semanttisten merkitysten lisäämisen kaavioiden käsittelyyn. Kaavio-editorisovellusten yhteydessä tämä avaa mahdollisuuksia monimutkaistenkin rakenteiden luomiseen. On täysin sovelluksen käyttötarkoituksesta kiinni, millaisia merkityksiä kaavioille luodaan ja miten kaavioelementtien kanssa käytävä vuorovaikutus toteutetaan.

Visuaalisen editorin suunnittelu riippuu siitä, kuinka pohjana oleva kieli on määritelty ja kuinka käyttäjä on vuorovaikutuksessa editorin kanssa [Minas, 2006]. Visuaalisten kielten elementtien avulla tulee siis määritellä se, millaista graafista esitystä kaavioeditorilla käsitellään ja miten tätä esitystä sillä voidaan muokata. Ilman tarkkaa määrittelyä kaavioeditori on hyvin lähellä normaalia piirto-ohjelmaa, jolla voidaan käsitellä rajatonta joukkoa ennalta määrittelemättömiä objekteja.

4.3. Kaavioeditoreista yleisesti

Kaavioeditorin määritelmää voidaan tarkentaa antamalla kolme päätekijää, jotka ovat erona kaavioita käsittelevien ja yleisten graafisten editorien välillä. Minasin [2002] määritelmän mukaan kaavioeditori on graafinen editori, joka on:

1) räätälöity tiettyä kaaviotyyppiä varten,
2) ymmärtää muokatun kaavion merkityksen jollain tasolla, ja
3) ei salli mielivaltaista piirtämistä, vaan rajoittaa käsittelyn kaaviomäärittelyn mukaisiin komponentteihin.

Kaikkia kaavioeditorisovelluksia yhdistävä tekijä on merkitysten antaminen graafisille objekteille, joita käyttäjä voi ennalta määritellyin vuorovaikutustavoin muokata. Kaavioita on käytetty historian aikana lukuisiin eri tarkoituksiin. Jokaista käyttötapaa varten on olemassa siihen parhaiten soveltuva kaaviotyyppi, jolloin yhden yleispätevän kaavion luominen on liki mahdotonta. Tämä pätee myös tietokoneella toteutettujen kaavioeditorien kohdalla. Erilaisia kaavioeditorisovelluksia on käytetty esimerkiksi ohjelmistosuunnittelussa ja ohjelmarakenteiden kuvaamisessa, proteiinien mallintamisessa, neuroverkkojen käsittelyssä, musiikin teossa ja elektronisten piirien suunnittelussa.

Kaavioeditorisovellusten graafinen ulkonäkö vaihtelee niiden käyttötarkoituksen mukaan suuresti. Objektien ja suhteiden kuvaamiseen voidaan käyttää käytännössä mitä graafista merkintätapaa tahansa. Kuitenkin eri sovellusalueista ja tutkimusaloista riippumatta kaavioeditoreissa käytetty tapa mallintaa tietoa sisältää samoja elementtejä. Objektit kuvataan usein nelikulmioiden, ympyröiden tai symbolien avulla. Objekteja yhdistävät suhteet taas esitetään yleensä viivojen tai kaarien avulla. Nämä perusominaisuudet ovat yllättävän lähellä toisiaan kaavioeditorien sovellusalueista riippumatta.

Seuraavissa kohdissa käsittelen kaavioeditorien rakentamisessa käytettäviä menetelmiä ja sovellusten toimintaperiaatteita. Graafiteorian käyttäminen kaavioiden mallintamisessa ja visualisoinnissa on ollut jo pitkään yksi keskeinen tutkimussuunta kaavioeditorien yhteydessä. Tähän perehdytään tarkemmin seuraavassa kohdassa. Kohdassa 4.5 käsittelen esteettisten kriteerien merkitystä ja soveltamista kaavio-editoreissa. Niin ikään on tutkittu sitä, miten käyttäjän vapauden ja toiminnaltaan rajoitetun kaavioeditorin suhde olisi kaikkein hyödyllisintä toteuttaa. Erilaisia lähestymistapoja tähän ovat vapaakäyttöiset, syntaktiset ja älykkäät kaavioeditori-sovellukset, joita vertailen kohdassa 4.6.

4.4. Graafiteorian hyödyntäminen kaavioeditoreissa

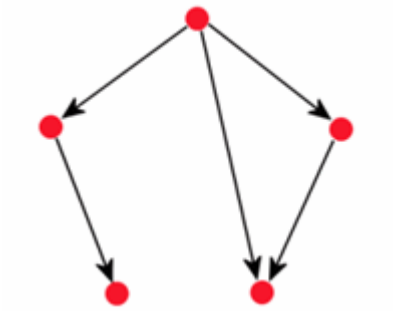
Graafi (graph) on tiedon mallintamiseen käytetty abstrakti sommitelma [Di Battista *et al.,* 1999]. Solmuista ja särmistä koostuvat graafit ovat yksi perustavimmista tavoista esittää suhteita objektien välillä [Böhringer and Newbery Paulisch, 1990]. Graafeihin perustuvien kaavioiden käyttö tiedon mallintamisessa on yleistynyt viime vuosikymmeninä, ja tämä sama kehitys näkyy myös tietokoneistetun kaavionkäsittelyn puolella.

Graafien käytön yleistymiseen tietokonesovelluksissa Böhringer ja Newbery Paulisch [1990] esittävät kaksi pääsyytä. Ensiksikin, ihmiset ymmärtävät tietoa yleensä paremmin, kun se on kuvallisessa muodossa tekstipohjaisen esityksen sijaan. Tämä johtuu osin siitä, että rakenteelliset ominaisuudet, kuten tasomaisuus, symmetria ja hierarkia, ovat heti havaittavissa hyvin piirretystä graafista ja näiden ominaisuuksien tunnistaminen helpottaa graafin ymmärtämistä. Toisekseen, korkealaatuisen grafiikan piirtoon kykenevien työasemien lisääntyminen ja niiden hintojen laskeminen ovat tehneet graafeja käyttävien graafisten käyttöliittymien hankkimisen mahdolliseksi yhä useammille ihmisille.

Kuten visuaalisten kielten ja kaavioiden yhteyttä tutkiessa on huomattu, kaavioiden sisältämien objektien välillä on poikkeuksetta suhteita. Nämä suhteet voidaan käsittää graafina, jossa objektit ovat graafien solmuja ja objekteja yhdistävät suhteet särmiä. Bardohlin [1997] mukaan kaavioiden rakenne voidaan siis esittää suunnattuna graafina. Suunnattu graafi eroaa suuntaamattomasta graafista siinä, että solmujen välillä kulkeville särmille tulee määritellä suunta. Käytännössä tätä voidaan siis soveltaa

esimerkiksi objektien välisten suhteiden esittämisessä kuvaamalla särmän suunnalla objektien välistä asemaa tai perintää.

Kuvassa 4.2 on esimerkki suunnatusta graafista. Ylimmäinen solmu kuvaa hierarkkisen kaavion ylintä objektia, jolla on särmien osoittamat suhteet kolmeen muuhun objektiin. Särmällä kuvataan tässä esimerkissä siis kahden objektin välistä lapsi-vanhempi-suhdetta.



Kuva 4.2. Esimerkki suunnatusta graafista, jossa on viisi solmua ja viisi särmää.

Graafiteoria on oma laaja tutkimusalansa, jossa pyritään esimerkiksi optimoimaan graafien käsittelyä ja havainnollistamaan niiden visuaalista esitystä. Kaavioiden käsittelyssä hyödyllisimpiä graafiteorian osa-alueita ovat *graafimuunnokset* (graph transformations) ja *asettelualgoritmit* (layout algorithms), joiden avulla graafien solmujen ja särmien asettelua voidaan muokata.

Graafimuunnokset kuvautuvat suoraan kaavioeditorisovellusten käyttöön, missä käyttäjän tulee pystyä muokkaamaan tekemäänsä graafista esitystä esimerkiksi siirtämällä objekti toiseen paikkaan. Graafimuunnosten avulla pystytään toimimaan johdonmukaisesti esimerkiksi tilanteissa, joissa lapsiobjekteja omaava objekti poistetaan. Asettelualgoritmien vahvuus tulee esiin tilanteissa, joissa on mahdollista käyttää tietokoneen laskentakapasiteettia graafien ulkoasun määrittelemisessä.

Asettelualgoritmit ovat sovellusten käyttämiä laskutoimenpiteitä, joiden avulla kaavion esityksestä pyritään luomaan ennalta määritellyn kaltainen. Käytännössä algoritmi laskee niin solmujen kuin särmienkin paikat ja pyrkii näin luomaan mahdollisimman ”hyvän” asettelun. Asettelun hyvyys on suhteellinen käsite, joka riippuu kaavion muodostavan graafin tyypistä ja sovelluksen käyttökohteesta.

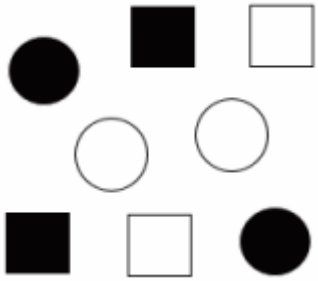
4.5. Kaavioesityksen ulkoasukriteerit ja muokkaaminen

Kaavioeditorien käytön kannalta on olennaista, että kaavioesityksen ulkoasu on mahdollisimman selkeä ja havainnollinen. Koska visuaalisena kielenä kaavioiden teho pohjautuu niiden ymmärrettävyyteen, tulee tämän ominaisuuden tärkeys ottaa huomioon kaavioeditorienkin yhteydessä. Ulkoasun merkitys heijastuu myös kaavioesityksen muokkaustapoihin, joita suunniteltaessa on ymmärrettävä esteettisen sommittelun tärkeys.

4.5.1. Esteettiset kriteerit

Graafien ja niihin pohjautuvien kaavioiden käytettävyys perustuu helppoon luettavuuteen eli kykyyn välittää kaavion merkitys nopeasti ja selkeästi. Huonosti piirretystä tai asetellusta kaavioesityksestä ei saa normaaliin tekstiesitykseen verrattuna vastaavaa hyötyä, vaan kaavio voi ennemminkin olla hämmentävä ja harhaanjohtava [Di Battista *et al.,* 1999]. Tämä havainto selittää sitä, miksi *esteettisten kriteerien* (aesthetic criteria) määritteleminen ja niihin pyrkiminen on kaavioita käytettäessä tärkeässä roolissa.

Useimmat esteettiset kriteerit ovat vakiintuneita asettelutapoja, jotka pohjautuvat psykologiseen tutkimustietoon siitä, miten ihmiset hahmottavat esimerkiksi erimuotoisia ja -värisiä kohteita. Näiden hahmolakien parissa tutkimusta on tehty jo vuosikymmeniä [Katz, 1948] ja niitä on yleisesti käytetty hyväksi graafisten käyttöliittymien yhteydessä. Kuvassa 4.3 on esimerkki, jolla havainnollistetaan värin vaikutusta samanlaisuuden hahmottamisessa. Ihminen mieltää samanväriset kappaleet poikkeuksetta yhdeksi ryhmäksi riippumatta siitä, minkä muotoisia kappaleet ovat.



Kuva 4.3. Väri samanlaisuuden määrittelijänä.

Soveltamalla hahmolakeja kaavioesitysten perinteisiin komponentteihin eli objekteihin ja niitä yhdistäviin suhteisiin, saadaan määriteltyä kaavioeditorien kannalta relevantteja esteettisiä kriteerejä. Näitä kriteerejä onkin kaavionkäsittelytutkimuksessa tullut esille lukuisia. Yleisiä kaavioesitysten esteettisiä kriteereitä ovat esimerkiksi särmien risteämien minimointi, symmetria sekä kaavion viemän pinta-alan minimointi [Böhringer and Newbery Paulisch, 1990]. Di Battista *et al.* [1999] puolestaan määrittelevät kaavion esteettisiksi kriteereiksi muun muassa tavoitteet särmien kulmien minimoinnista ja särmien pituuksien vakioinnista.

Vaikka hyviksi todettujen esteettisten seikkojen määritteleminen on melko suoraviivaista, niiden noudattaminen käytännössä ei ole ongelmatonta. Di Battista *et al.* [1999] korostavat, että kaavioeditorien yhteydessä esteettisten kriteerien mukaisiin esityksiin pyrittäessä on hyvä ottaa huomioon kaksi havaintoa:

- esteettisten kriteerien ristiriitaisuuden vuoksi kompromissit ovat usein väistämättömiä, ja
- vaikka esteettiset kriteerit eivät ole ristiriidassa, on niitä usein laskennallisesti vaikeaa käsitellä samanaikaisesti.

Lukuisten esteettisten kriteerien tiukka noudattaminen on siis useissa tilanteissa mahdotonta, sillä niihin pyrkiminen saattaa viedä tehokkaaltakin asettelualgoritmilta ja tietokoneelta huomattavan ajan. Käytännössä esteettiset kriteerit ovat ohjeistoja, joita noudattamalla kaavioesityksen sommittelusta saadaan havainnollisempi. On lopulta kaavioeditorisovelluksen toteutustavasta kiinni, kuinka tarkasti esteettisten kriteerien noudattamista pyritään tukemaan.

4.5.2. Asettelumenetelmiä ja -työkaluja

Kaavioeditoreita käytettäessä esteettisiin kriteereihin pyrkimiseen voidaan käyttää useaa lähestymistapaa, joista tässä yhteydessä esittelen yleisellä tasolla erityyppisiä ratkaisuita. Kohdassa 4.4 mainitut asettelualgoritmit ovat yksi keino esteettisesti miellyttävän esityksen luomiseen. Automaattisten asettelualgoritmien vahvuus on tilanteissa, joissa tavoiteltavat esteettiset kriteerit ovat yksiselitteisiä ja kaavion objekteilla ei ole monimutkaisia semanttisia merkityksiä. Tällöin kaavion esitysmuotoa voidaan muokata täysin vapaasti esteettisen sommittelun ehdoilla, ja asettelualgoritmin käyttö säästää huomattavasti aikaa.

Automaattisten asettelualgoritmien käyttö johtaa ongelmatilanteisiin silloin, kun asettelulta vaadittaisiin käyttäjä- tai kaaviokohtaisten kriteerien käyttöä. Toinen ongelma automaattisissa asettelualgoritmeissa on se, että ne harvoin ottavat huomioon kaavion aikaisempaa asettelua uutta asettelua laskiessaan. Tämä voi turhauttaa käyttäjää, joka menettää mieleisensä asettelun. [Böhringer and Newbery Paulisch, 1990]

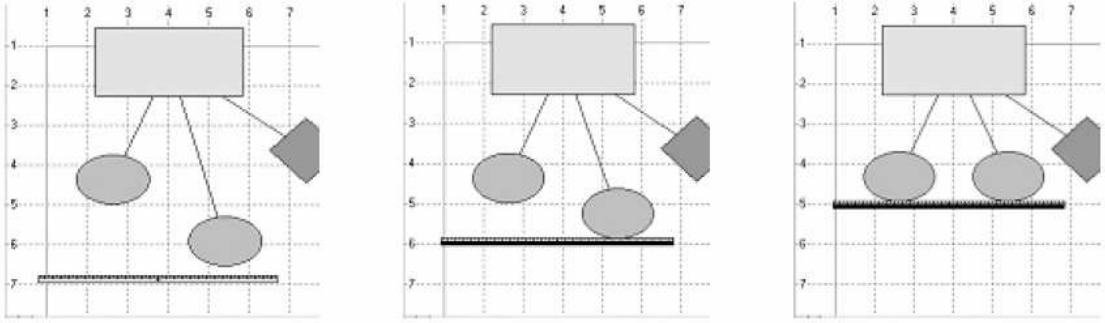
Käyttäjän tottumuksista ja kaaviotyypistä riippuen olisi asettelualgoritmin pystyttävä tapauskohtaisesti päättelemään tavoiteltavat esteettiset kriteerit. Tätä lähestymistapaan pohjautuvat dynaamiset asettelualgoritmit. Kehittyneimmät dynaamiset algoritmit pystyvät tarkkailemaan ja oppimaan käyttäjän työskentelytapoja. Masui *et al.* [1994] kuvaavat järjestelmää, jolle käyttäjä näyttää esimerkkejä hyvistä ja huonoista asetteluista. Tämän jälkeen järjestelmä pyrkii noudattamaan mahdollisimman hyvin käyttäjän määrittelemiä kriteerejä. Vaikka dynaamiset asettelualgoritmit tarjoavat käyttäjälle mahdollisuuden vuorovaikutteisuuteen ulkoasun laadinnassa, on saavutettu etu silti usein näennäinen [Raisamo, 2002]. Käyttäjällä on mahdollisuus vaikuttaa vain siihen, mitä ennalta vaihtoehtoina annettuja kriteerejä sovellus noudattaa.

Manuaalisesti tapahtuva kaavioiden asettelu on vastakohta automaattisille asettelualgoritmeille. Käsin tapahtuvasta asettelusta tunnetuin ja yksinkertaisin tapa on objektien raahaaminen, joka perinteisesti tapahtuu hiiren avulla. Käyttäjä on vastuussa objektien ja niiden välisten suhteiden hallinnasta, jolloin esityksen visuaalisesta ilmeestä saadaan varmasti käyttäjää miellyttävä. Käsin tapahtuvan asettelun ongelmana on luonnollisesti sen työläys esteettisten kriteerien saavuttamiseksi. Esimerkiksi kaavioesitykseen tehtävien suurempien muutosten yhteydessä jokainen objekti tulee valita, raahata ja asetella yksitellen.

Manuaalista asettelua kehittyneemmät asettelutavat perustuvat apuna toimivien työkalujen käyttöön. Perinteisesti näiden työkalujen avulla on tasattu objekteja, mistä johtuen tasaaminen on esimerkkinä tässäkin kohdassa. Tasaustyökalujen toiminta voi perustua komentopohjaisuuteen tai suorakäyttöisyyteen [Raisamo, 2002].

Komentopohjaiset tasaustyökalut käyttävät tasauskomennon valitsemiseen esimerkiksi valikkoa. Käytännössä käyttäjä valitsee valikosta kriteerin, jonka mukaan halutut objektit tasataan. Komentopohjaisten tasaustyökalujen suurimpana haittapuolena on suorakäyttöisyyden puute, sillä käyttäjä ei ennen kriteerin valitsemista näe, kuinka objektit todellisuudessa reagoivat komentoon.

Suorakäyttöiset tasaustyökalut perustuvat objektien vuorovaikutteiseen tasaamiseen. Kuvan 4.4 esimerkissä hiirikursorin korvaavaa tasauspuikkoa [Raisamo, 1995; Raisamo and Räihä, 1996] käytetään piirrettyjen objektien tasaamiseen. Tasauspuikon käyttö ei vaadi tasattavien objektien valitsemista, vaan kaikki työkalun eteen tulevat objektit tasataan samalla periaatteella. Käyttäjä voi tasausta suorittaessaan jatkuvasti tarkkailla objektien käyttäytymistä ja tarpeen tullen suorittaa korjaavia toimenpiteitä välittömästi.



Kuva 4.4. Objektien tasaustyökalu tasauspuikko. [Raisamo and Räihä, 1996]

4.6. Vapaakäyttöiset, syntaktiset ja älykkäät kaavioeditorit

Kaavioeditorit ovat normaaleihin graafisiin editoreihin verrattuna tarkemmin suunnattuja tiettyyn tehtävään. Sovellusten tulee rajoittaa kaavioiden käyttöä niin, että editorilla ei voida luoda tai muokata ennalta määrittelemättömiä objekteja. Lisäksi kaavioeditorien tulee ymmärtää graafinen esitys esimerkiksi sisäisen mallin, kuten graafien, avulla. Näiden vaatimusten toteuttamiseen voidaan käyttää kahta tapaa riippuen siitä, kuinka tarkoin käyttäjän toimia halutaan rajoittaa ja seurata.

Vapaassa muokkauksessa (free-hand editing) käyttäjälle annetaan paljon vapauksia kaavioiden hallitsemisessa [Minas, 2002]. Käyttäjä voi vapaasti luoda ja käsitellä editorin hyväksymiä kaavioita ilman, että sovellus tarkistaisi niiden oikeellisuutta välittömästi. Objektien muokkaamiseen käytettäviä tapoja ei ole erikseen rajoitettu, vaan käyttäjä voi käsitellä niitä parhaaksi katsomallaan tavalla. Lisäksi vapaaseen muokkaukseen perustuvissa kaavioeditoreissa ei tarvita sisäistä mallia, johon jokainen käyttäjän tekemä kaavion muutos heijastuisi. Sen sijaan kaavioeditorissa tulee olla

jäsennin (parser), joka tarkistaa käyttäjän tekemien muokkausten oikeellisuuden ja syntaktisen rakenteen.

Vapaa muokkaus jättää käyttäjälle enemmän vastuuta muokkausprosessista. Sovellus ei välittömästi korjaa virheitä, jolloin käyttäjällä tulee olla valmiudet työskennellä ilman jatkuvaa opastusta. Toisaalta vapaan muokkauksen etuna voidaan pitää sitä, että jäsentimen avulla tapahtuva kaavion rakenteen kontrollointi antaa enemmän vapauksia käyttäjän työskentelylle. Aika-ajoin suoritettavan jäsentimen käyttö sallii esimerkiksi väärien kaavioiden hetkellisen luomisen, mikä tukee ihmisten luonnollista tapaa hahmotella kuvioita [Minas, 1997].

Syntaktisessa muokkauksessa (syntax-direct editing) kaavioeditorisovellus kontrolloi tarkasti työskentelyä [Minas, 2002]. Jokainen käyttäjän suorittama operaatio muuttaa kaavion merkitystä. Syntaktisissa kaavioeditoreissa on käyttäjän tekemien muutosten mukaisesti päivittyvä sisäinen malli, joka voidaan kuvata kulloiseenkin kaaviomalliin sopivan graafin avulla. Käyttäjän tekemät muutokset, esimerkiksi objektin siirto toisen objektin alle, voidaan mallintaa graafimuunnoksia käyttäen.

Syntaktiset kaavioeditorit tarjoavat käyttäjälle jatkuvaa opastusta työskentelyssä [Chok and Marriott, 1998]. On käyttäjän osaamisesta ja editorisovelluksen aihealueen vaativuudesta kiinni, onko opastuksesta enemmän haittaa vai hyötyä. Ennalta määrittelemättömien muokkausoperaatioiden suorittaminen on mahdotonta, mikä saattaa joissakin tapauksissa aiheuttaa rajoitteita työskentelylle. Lisäksi suorakäyttöisyyden mahdollistamisen syntaksisissa kaavioeditoreissa on haastavaa, sillä tätä varten tulee määritellä objektien sallitut muokkausoperaatiot, kuten graafimuunnokset [Chok and Marriott, 1998].

Edellä esiteltyjen vapaakäyttöisten ja syntaktisten kaavioeditorien sisältämien haittapuolien pohjalta on ehdotettu kummankin lähestymistavan parhaat puolet sisältävää kaavioeditorimallia, *älykästä kaavioeditoria* (intelligent diagram editor). Älykäs kaavioeditori sallii kaavion luomisen käyttäjän haluamassa järjestyksessä [Chok and Marriott, 1998]. Kaavion sisäisen mallin luominen perustuu jäsentimeen, joka reaaliaikaisesti tarkkailee käyttäjän luomia kaavioita. Jäsennin kerää tietoa objektien välisistä etäisyyksistä ja muista suhteista. Tätä tietoa voidaan hyödyntää esimerkiksi objekteja siirrettäessä tai niiden kokoa muutettaessa, jolloin pystytään aikaisemmin kerätyn tiedon pohjalta säilyttämään objektien suhteet toisiinsa nähden.

Älykkäät kaavioeditorit soveltuvat paitsi perinteiseen hiirellä ja näppäimistöllä tapahtuvaan käyttöön, myös erityisen hyvin kynän ja kosketusnäytön hyödyntämiseen [Chok and Marriott, 1998].

4.7. Yhteenveto

Tässä luvussa olen käsitellyt kaavioeditorien sijoittumista yleisten graafisten editorien joukkoon, tarkastellut kaavioihin ja visuaalisiin kieliin liittyvää teoriapohjaa sekä tuonut esille käytännön ratkaisuja kaavioeditorien toteuttamiseen. Yhteistä kaikille

kaavioeditoreille on visuaalisen kielen käyttäminen tiedon mallintamisessa. Kaaviot ovat visuaalisen kielen osana löytäneet paikkansa ihmisten suosimana tapana esittää informaatiota visuaalisessa muodossa. Visuaalisten kielten lisäksi kaavioiden käsitteeseen liittyvät usein graafit. Graafit ovat tapa mallintaa tietoa kaavioiden visuaalista esitystä alemmalla tasolla.

Käyttäjän kannalta näkyvin asia editorien käytössä on kaavioiden muokkaamiseen annettu vapaus. Vapaakäyttöiset kaavioeditorisovellukset tukevat kaavioiden hahmottelua ja muokkausoperaatioiden kokeilua. Syntaksiin perustuvat editorit taas rajoittavat kaavion muokkaamista, mutta toisaalta myös tukevat työskentelyn etenemistä ja kaavion oikeellisuuden tarkkailua. Kompromissi näiden kahden lähestymistavan väliltä ovat älykkäät kaavioeditorit, jotka voidaan nähdä lupaavimpana tapana ohjata käyttäjän työskentelyä.

Omassa toteutustyössäni olen keskittynyt kaavion rakenteiden käsittelyä enemmän esitysten visuaaliseen puoleen. Kaavioeditorien yhteydessä graafiteoria liittyy asettelualgoritmien lisäksi paljon siihen, kuinka erilaisten matemaattisten ongelmien ratkaiseminen onnistuu graafien avulla. Tässä tutkielmassa pohjana käytettävä kaavioeditorisovellus on sisäisen mallinsa ja kaaviorakenteidensa puolesta melko yksinkertainen, eikä kaavion muokkaaminen perustu graafimuunnosten käyttöön. Pääpaino konstruktiivisessa vaiheessa onkin ollut kaavioeditorien tarjoamien graafisten esitysten muokkaamiseen käytettävissä vuorovaikutustavoissa.

5. Kaavioeditori interaktiivisille tussitauluille

Tässä luvussa kuvaan interaktiivisille tussitauluille toteuttamani graafisen kaavioeditorisovelluksen. Käyn läpi alkutilanteen, josta suunnittelu- ja toteutustyö käynnistyi. Esittelen konstruktiivisen vaiheen pohjana olleen graafisen kaavioeditorin, jota jatkokehitin interaktiivisten tussitaulujen asettamien vaatimusten mukaisesti. Tämän jälkeen pohdin suunnitteluvaiheeseen vaikuttaneita tekijöitä sekä esittelen toteutukselle asetetut tavoitteet. Lopuksi esittelen ja arvioin editorisovellukseen lisätyt ominaisuudet aikaisempien tutkimustulosten valossa.

5.1. Toteutusvaiheen lähtökohdat

Tämän tutkielman konstruktiivisessa vaiheessa on hyödynnetty lukujen 3 ja 4 tutkimustietoa käytännössä. Olen jatkokehittänyt olemassa olevaa, normaaliin työpöytätietokonekäyttöön tarkoitettua kaavioeditorisovellusta interaktiivisia tussitauluja varten.

Monissa aikaisemmissa tutkimuksissa interaktiivisille tussitauluille tarkoitetut sovellukset on toteutettu alusta asti tussitaulukäyttöä silmällä pitäen [Moran *et al.,* 1997; Mynatt *et al.,* 1999]. Tällöin sovellus voidaan sen sisältöä, esitystapaa ja vuorovaikutustapaa myöden räätälöidä sopivaksi juuri interaktiivisia tussitauluja varten. Interaktiivisille tussitauluille varta vasten suunniteltujen ohjelmien hyödyntäminen on kuitenkin melko rajattua; nämä sovellukset ovat joko tutkimuskäyttöön suunnattuja tai jonkin tussitaululaitteen mukana toimitettavia sovelluksia.

Tutkielman konstruktiivisen vaiheen yleisen tason tavoitteena on saada tietoa siitä, kuinka perinteisen *WIMP-sovelluksen* (Windows, Icons, Menus, Pointing Devices) muuntaminen interaktiivisille tussitauluille onnistuu käytännössä. Interaktiivisten tussitaulujen käytön yleistymiseksi on mielestäni olennaista pystyä käyttämään olemassa olevia ohjelmistoja tehokkaasti myös tauluympäristössä. Vain tussitaulukäyttöön suunnattujen ohjelmien vahvuutena on niiden täydellinen soveltuvuus ennalta tiedostettuun käyttötapaan, mutta tällaisten sovellusten saatavuus on käytännössä heikkoa. Olemassa olevien, käyttäjien hyviksi havaitsemien sovellusten ongelmaton käyttö myös interaktiivisilla tussitauluilla on aihealue, josta pyrin toteutustyön avulla saamaan lisätietoa.

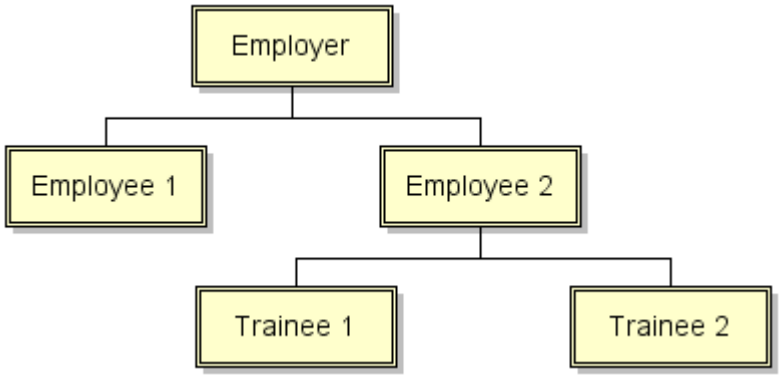
Työn konstruktiivisessa vaiheessa käsittelen kaavioeditoreja, jotka esittelin tarkemmin edellisessä luvussa. Olen jatkokehittänyt alun perin hiiren ja näppäimistön avulla käytettäväksi tarkoitettua kaavioeditorisovellusta interaktiivisten tussitaululaitteiden asettamien vaatimusten mukaisesti. Editorisovellukset ovat erityisen soveltuvia interaktiivisilla tussitauluilla käytettäviksi, sillä ne tukevat usein perinteisiltä tauluilta tuttua tiedon hahmottelua. Käyttäjä voi liitua tai tussia käyttäen piirtää taululle muistiinpanoja tai kuvioita. Ensisijainen päämääräni kaavioeditorisovelluksen muuttamisessa interaktiivisille tussitauluille on ollut mahdollistaa helposti omaksuttava,

luonnollinen ja intuitiivinen tapa käsitellä taululla esitettävää tietoa. Tarkemmat suunnitteluvaiheessa asetetut tavoitteet esitellään kohdassa 5.3.

5.2. Yksinkertainen kaavioeditorisovellus

Olen käyttänyt toteutusvaiheen pohjana Tampereen yliopiston tietojenkäsittelytieteiden laitoksen Programming Graphical User Interfaces -kurssin harjoitustyönä tekemääni kaavioeditorisovellusta. Java-kielinen sovellus on tarkoitettu yksinkertaisten graafisten kaavioiden käsittelyyn.

Sovelluksella voidaan käsitellä hierarkkisia kaavioita. Yksi käyttötapa on jonkin organisaation henkilöstörakenteen kuvaaminen (kuva 5.1). Jokaisella kaaviolla tulee aina olla yksi pääobjekti. Tämän pääobjektin alle voidaan luoda ääretön määrä lapsiobjekteja. Samoin lapsiobjektien alle voidaan luoda edelleen uusia lapsiobjekteja. Näin muodostuu alaspäin laajeneva kaaviorakenne tiedon mallintamiseen.



Kuva 5.1. Kaavion hierarkkinen rakenne.

Kaavioeditorissa on kohdassa 4.6 esitellyn jaottelun perusteella vaikutteita syntaktisista ja toisaalta myös älykkäistä kaavioeditoreista. Syntaktisen kaavioeditorin määritelmän mukaisesti sovellus tarkkailee käyttäjän tekemiä muokkaustoimenpiteitä ja tarvittaessa estää virheellisten muutosten suorittamisen. Lapsiobjektia ei voi raahata hierarkiassa ylempänä olevan objektin tasolle tai sitä ylemmäksi, vaan lapset esitetään graafisesti aina vanhemman alapuolella (kuva 5.1). Jos käyttäjä pyrkii siirtämään objektin kiellettyyn kohtaan, palautetaan se välittömästi takaisin alkuperäiseen paikkaansa. Sovellus päivittää lisäksi kaavioesityksen sisäistä mallia jokaisen käyttäjän tekemän toimenpiteen jälkeen. Jos käyttäjä poistaa objektin jolla on lapsiobjekteja, poistetaan samalla myös lapsiobjektit kaavion eheyden säilyttämiseksi. Jokaisella objektilla on pääobjektia lukuun ottamatta aina oltava yksi ja vain yksi vanhempi-objekti.

Kaavioeditorissa on myös piirteitä älykkäästä kaavioeditorista [Chok and Marriott, 1998]. Kaavioeditori huomioi samalla tasolla olevien eli sisarobjektien väliset etäisyydet. Kun käyttäjä siirtää näiden sisarobjektien vanhempaa, liikkuvat lapsiobjektit mukana säilyttäen aikaisemmin määritellyt suhteet toisiinsa nähden. Kaavioeditori ei

myöskään voimakkaasti kahlitse käyttäjän työskentelyä, vaan reagoi lähinnä havaitessaan epävalideja operaatioita. Monet tiukasti syntaktiset kaavioeditorit ovat aktiivisesti mukana käyttäjän ohjaamisessa.

Sovelluksen käyttö perustuu objektien luomiseen ja poistamiseen. Tämän lisäksi objektien paikkaa voidaan muuttaa raahausliikettä käyttämällä. Jos käyttäjä pudottaa objektin toisen objektin päälle, siirretään pudotettu objekti kohdeobjektin lapseksi. Kaavioeditori ei sisällä objektien automaattista asettelua. Käyttäjä on itse vastuussa luomiensa kaavioiden esteettisen asettelun toteuttamisesta. Ainoa poikkeus on hierarkkisen esityksen ylläpitämiseksi käytössä oleva automaattinen lapsiobjektin ja sen vanhemman suhteen ylläpitäminen.

Yksittäisen objektin esitystavan muuttamiseen on olemassa kolme toimintoa. Objektille voidaan määritellä teksti, joka näytetään objektia esittävän nelikulmion sisällä. Myös objektin väriä voidaan muuttaa valitsemalla neljän ennalta annetun vaihtoehdon väliltä. Lisäksi objektin kokoa voidaan muuttaa aktiivisen objektin oikeassa alakulmassa sijaitsevaan nelikulmioon kohdistuvan raahausliikkeen avulla. Kaavioesityksen tasolla editorissa on toiminnot uuden kaavion luomiseen sekä muokatun kaavion tallentamiseen ja avaamiseen.

5.3. Suunnittelu interaktiivisille tussitauluille

Alkuperäinen kaavioeditorisovellus pohjautui perinteiseen työpöytämetaforaan, jossa vuorovaikutus ohjelman kanssa tapahtuu hiiren ja näppäimistön avulla. Yksittäisen toiminnon käynnistämiseen oli kolme keinoa; valikkopalkin käyttäminen, ponnahdusvalikon esiin tuominen hiiren oikean painikkeen avulla ja näppäinoikoteiden hyödyntäminen. Nämä standardeiksi muodostuneet syötteidenantotavat ovat interaktiivisia tussitauluja käytettäessä kuitenkin lähes käyttökelvottomia. Tussitaulukäyttöistä kaavioeditoria suunnitellessa aikaisemmat vuorovaikutustavat tuli unohtaa.

Kaavioeditorisovelluksen muokkaamista suunnitellessani tein karkean jaon toteutukselle asettamistani vaatimuksista. Nämä kaksi tavoitetta pohjautuvat siihen, mitä interaktiivisten tussitaulujen vuorovaikutustapoja käsitelleissä aikaisemmissa tutkimuksissa on havaittu tärkeäksi. Suhteutin interaktiivisten tussitaulujen tärkeimmät vaatimukset kaavioeditoreista olemassa olevaan tietoon.

Tussitaululaitteille jatkokehittämäni kaavioeditorisovelluksen kaksi päätavoitetta ovat:

- helppo tapa antaa komentoja interaktiivisella tussitaululla, ja
- tehokas keino hallita taulun tarjoamaa fyysistä ja virtuaalista tilaa.

Ensimmäisen kohdan keskeisin aspekti on hiireen ja näppäimistöön pohjautuvan vuorovaikutustavan korvaaminen interaktiivisten tussitaulujen käyttöön paremmin soveltuvalla tekniikalla. Luvuissa 3 ja 4 käsiteltyjen aikaisempien ratkaisuiden pohjalta pyrin löytämään käyttötavan, jossa yhdistyisivät interaktiivisten tussitaulujen ja

kaavioeditorien asettamat vaatimukset komentojen ja syötteiden antamiselle. Sovelluksen käytöstä tulee tehdä mahdollisimman helppoa, jotta aloittelijat pystyvät omaksumaan uuden vuorovaikutustavan kohtuullisella vaivalla. Toisaalta kaavioeditorien käyttäjäkunta on usein teknisesti orientoitunutta, jolloin uudessakin ympäristössä tulee pystyä työskentelemään niin, ettei komentojen antaminen ole turhauttavan hidasta. Komentojen antamisesta pitäisi siis tehdä paitsi selkeää ja helppoa, myös tehokkaan työskentelyn mahdollistavaa.

Toinen tavoite sisältää vaatimuksen interaktiivisen tussitaulun tarjoaman työskentelytilan hallinnasta. Käyttäjälle ei saa tulla vaikutelmaa, että hän ei pysty hallitsemaan taululla muokattavaa kaaviota. Interaktiivisen tussitaulun tilanhallinta voidaan jakaa kahteen alueeseen: fyysiseen ja virtuaaliseen. Tilan fyysinen hallinta kattaa esimerkiksi sen, kuinka paljon käyttäjän tulee liikkua tai kurotella taulun edessä kaavioeditoria käyttäessään. Taulun fyysiset ulottuvuudet pitää pystyä näkemään hyödyllisenä asiana eikä käyttöä rajoittavana tekijänä. Tilan virtuaalinen hallinta tarkoittaa sitä, miten kaavioeditorin esitystä voidaan hallita taululla suhteessa käytettävissä olevaan tilaan, joka on periaatteessa ääretön. Käyttäjällä tulisi jatkuvasti olla kokonaiskäsitys kaavioon kuuluvista objekteista sekä järjestämis- ja hallitsemiskeinoista.

Pyrkimyksenäni on tukea sovelluksen yhteensopivuutta erimallisten interaktiivisten tussitaululaitteiden kanssa. Käytössäni olleessa SMART Board 500 -mallissa on mukana neljä tussia kirjoittamiseen sekä pyöreä kumi, jolla muistiinpanoja voi pyyhkiä pois. Useiden muidenkin tussitaululaitteiden mukana toimitetaan oheislaitteita, jotka ovat taulun käytön kannalta enemmän tai vähemmän tarpeellisia. Päätin olla hyödyntämättä näitä oheislaitteita ja keskittyä siihen, millä ratkaisuilla ohjelma olisi käyttökelpoinen mahdollisimman monessa eri laiteympäristössä. Käytännössä siis useissa aikaisemmissa toteutuksissa hyödynnettyjä kynälaitteiden painikkeita tai niiden mahdollisuuksia ei toteutuksessa ole käytetty. Pyrkimyksenä on tavoitella alakohdassa 3.1.3 esitellyn Barehands-tekniikan [Ringel *et al.*, 2001] kaltaista ratkaisua, jossa vuorovaikutus perustuu pelkän käden käyttöön.

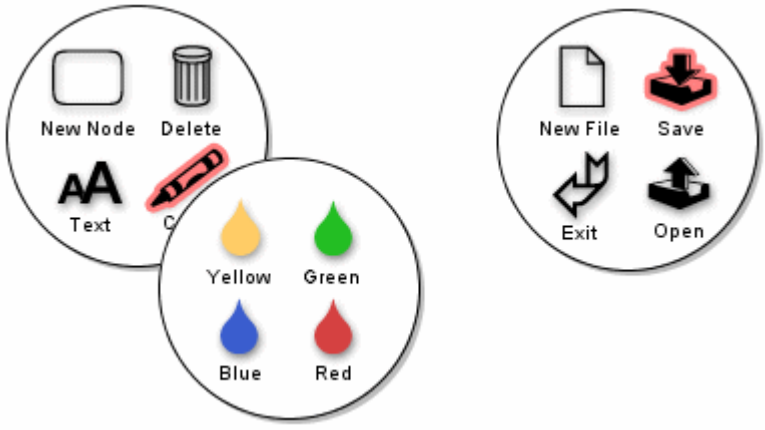
Kaavioeditorin muuntamisessa interaktiivisille tussitauluille olen asettanut pääpainon kaavionkäsittelyyn suunnattujen vuorovaikutustekniikoiden toteuttamiselle. Alkuperäinen kaavioeditori on toiminnaltaan varsin yksinkertainen, eikä tämän tutkielman puitteissa tarkoituksenani ole ollut juurikaan kehittää sitä monipuolisemmaksi kaavioteknisessä mielessä. Käytössä on esimerkiksi vain yhden tyyppisiä objekteja ja suhteita. Sen sijaan olen kaavionkäsittelyn tunnettujen lainalaisuuksien pohjalta pyrkinyt löytämään vuorovaikutustapoja, jotka voisivat olla käyttökelpoisia yleisellä tasolla. Liian spesifiä kaaviotyyppiä käsiteltäessä on olemassa vaara, että esimerkiksi komentojen antaminen tai tilanhallinta soveltuu vain tietyn tyyppiselle kaavioesitykselle.

5.4. Komentojen ja syötteiden antaminen

Toteutin useita tekniikoita, joiden avulla on mahdollista antaa komentoja interaktiivisille tussitauluille tarkoitetussa kaavioeditorisovelluksessa. Esittelen tässä kohdassa kontekstivalikon, eleiden käytön komentojen ja kirjoituksen antamisessa sekä click-and-drop-tekniikan kaltaisen ratkaisun objektien siirtämiseen.

5.4.1. Kontekstivalikko

Korvatakseni alkuperäisessä kaavioeditorissa olleet tavat antaa komentoja toteutin interaktiivisia tussitauluja varten kontekstivalikon (kuva 5.2). Kontekstivalikko on ulkoasultaan samankaltainen kuin esimerkiksi eBeamissa [eBeam, 2007] ja Flatlandissa [Mynatt *et al.,* 2000]. Käyttäjän valitsemaan kohtaan avautuvaa kontekstivalikkoa käytettäessä komennon antamiseksi ei tarvitse kurkotella kohti ruudun yläkulmassa sijaitsevia alasvetovalikoita.



(a) (b)

Kuva 5.2. Piirakkavalikot yksittäisen objektin (a) ja tiedoston (b) käsittelyyn.

Olen toteuttanut kaksi kontekstivalikkoa riippuen siitä, missä kohdassa käyttäjä tuo valikon esiin. Jos valikko tuodaan esiin jonkin objektin päällä, avautuu objektin muokkaamiseen käytettävä valikko (kuva 5.2a). Valikossa on neljä komentoa; uuden objektin luominen, objektin poistaminen, tekstin muuttaminen ja värin muuttaminen. Toinen kontekstivalikko avautuu silloin, kun käyttäjä ei ole yhdenkään objektin päällä (kuva 5.2b). Tämän valikon avulla käyttäjä voi luoda uuden kaavion, tallentaa ja ladata muokatun kaavion, tai sulkea sovelluksen. Kahden eri kontekstivalikon tarkoituksena on pitää kerralla näkyvissä olevien komentojen määrä pienenä ja näyttää vain komennot, jotka ovat käyttökelpoisia sen hetkisessä kontekstissa.

Käyttäjä voi kuljettaa sormeaan valikon vaihtoehtojen päällä, jolloin sormen alla olevan komennon kuvake muuttuu punaiseksi ilmoittaen mahdollisesta valinnasta (kuva 5.2b, kohta ”Save”). Lisäksi, kun käyttäjä siirtää sormensa objektin värin vaihtamista kuvaavan komennon päälle (kuva 5.2a, kohta ”Color”), avautuu alavalikko, josta haluttu väri voidaan valita samalla periaatteella.

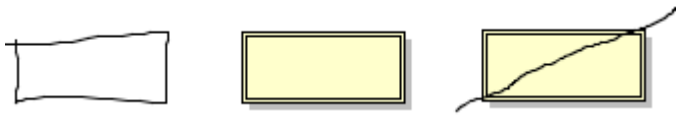
Kontekstivalikon esiin tuominen tapahtuu pitämällä sormi paikallaan taulun pinnalla puolen sekunnin ajan. Koska sormi voi taulua painettaessa liikahtaa hieman, sallitaan kontekstivalikon avaamisessa muutaman pikselin liike. Useissa aikaisemmissa ponnahdus- ja kontekstivalikkototeutuksissa [eBeam, 2006; Mynatt *et al.,* 2000] valikon avaamiseen on käytetty kynässä olevaa painiketta. Tämä tapa on riippuvainen tietystä interaktiivisesta tussitaululaitteesta ja sen stylus-kynästä. Avattaessa valikko pelkän sormen avulla ei erillisiä lisälaitteita tarvita.

Kontekstivalikkoa voidaan käyttää nostamatta sormea taulun pinnasta. Kun valikko tulee esiin, voi käyttäjä yhdellä liikkeellä valita haluamansa komennon. Valikko jää myös esille, vaikka käyttäjä nostaisi sormen taulun pinnasta. Tällöin käyttäjä voi sormen painalluksen avulla valita haluamansa komennon.

5.4.2. Eleiden käyttö

Olen toteuttanut interaktiivisia tussitauluja varten sovellukseen eleitä komentojen ja tekstin antamiseen. Eleet ovat aikaisempien tutkimusten valossa [Pedersen *et al.,* 1993; Igarashi *et al.,* 2000; Damm *et al.,* 2000; Chen *et al.,* 2003] käyttökelpoinen tapa tukea luonnollisen kaltaisia syötteitä interaktiivisilla tussitauluilla. Suora vuorovaikutus ilman syötelaitteita tukee parhaalla mahdollisella tavalla eleisiin pohjautuvia tekniikoita.

Valitun vanhempiobjektin alle voidaan luoda uusi lapsiobjekti hahmottelemalla nelikulmion muotoinen kuvio halutulle paikalle taululla (kuva 5.3a). Käyttäjän tulee aina ennen uuden objektin luontia valita objekti, jonka alle uusi lapsi halutaan lisätä. Eleen avulla tapahtuva objektin luominen tulee tehdä käyttäen yhtäjaksoista piirtoa, jossa sormea ei nosteta taulun pinnasta. Kun hahmoteltu nelikulmio on valmis ja sormi nostetaan ylös, kaavioeditori tunnistaa hahmotellun kuvion koon ja luo uuden objektin automaattisesti (kuva 5.3b). Uuden objektin vanhempiobjekti on lisäyseleen jälkeen edelleen valittuna, jotta käyttäjä pystyy joustavasti luomaan useampia uusia lapsiobjekteja.



(a) (b) (c)

Kuva 5.3. Uuden objektin luominen (a), luotu objekti (b) ja objektin poisto (c).

Uuden objektin luomiseksi eleen avulla tulee käyttäjän hahmotella nelikulmio, joka on vähintään 30 pikseliä leveä ja korkea. Lisäksi eleen alku- ja loppukohtien tulee olla 40 pikseliä leveän ja 40 pikseliä korkean neliön sisällä. Ele alkaa sormen koskettaessa taulun pintaa ja loppuu sormen nostamiseen. Kun nämä kaksi kriteeriä täyttyvät, luodaan uusi objekti. Jos ele ei ole vaatimusten mukainen, käyttäjälle ei erikseen ilmoiteta uuden objektin luomisen epäonnistumisesta. Tunnistusmekanismi on tehty

melko löyhäksi, jotta objektien luomiseen riittää kuvion hahmottelu. Nelikulmion hahmottelemiseen voi käyttää vapaasti mistä tahansa kohdasta alkavaa liikettä, eikä piirron suuntaa ole määritelty.

Kaavioeditorissa on lisäksi olemassa oma elekomento objektin poistamiseksi. Käyttäjän tulee piirtää poistettavan objektin yli ele, jonka alku- ja loppupisteet ovat objektin ala- ja yläreunojen ulkopuolella (kuva 5.3c). Lisäksi eleen alku- ja loppupisteiden välin tulee sivusuunnassa olla vähintään 15 pikseliä, jotta poistoele erottuu seuraavaksi esiteltävästä tasauseleestä. Jos ele on sallittu, poistetaan objekti ilman erillistä varmistusta.

Sovelluksessa on käytetty eleitä myös kaavionkäsittelystä tuttujen toimintojen toteuttamiseen. Käyttäjä voi tasata objekteja piirtämällä vaaka- tai pystysuoria eleitä objektien lähelle. Esimerkkinä olkoon tilanne, jossa kaksi objektia ovat eri korkeudella ja ne halutaan tasata alareunojen mukaan. Käyttäjän tulee piirtää vaakasuora ele, joka ulottuu leveyssuunnassa tasattavaksi haluttujen objektien alle (kuva 5.4a). Eleen ei siis tarvitse olla yhtä leveä kuin tasattavat objektit. Toinen vaatimus tasauseleelle on se, että piirretyn viivan ja tasattavien objektien välinen etäisyys saa olla korkeintaan 30 pikseliä. Näiden kriteerien täyttyessä objektit asetetaan samalle tasolle piirretyn eleen kanssa (kuva 5.4b). Samat kriteerit pätevät objektien tasaamiseen niiden yläreunan perusteella. Tällöin tasausele piirretään tasattaviksi haluttavien objektien yläpuolelle.

(a) (b)

Kuva 5.4. Tasauseleen piirto (a) ja tasatut objektit (b).

Pystysuora tasausele toimii vaakasuoran eleen kanssa samalla periaatteella, mutta tällöin objektit asetellaan joko niiden vasemman tai oikean reunan mukaan. Tasattavien objektien ja piirretyn pystysuoran eleen välinen etäisyys saa olla korkeintaan 30 pikseliä.

Tasauselettä käyttäen voidaan tasata objekteja myös silloin, kun kahden tai useamman objektin ylä- ja alareunat tai vastaavasti oikea ja vasen reuna ovat lähellä toisiaan. Esimerkiksi kuvassa 5.4b tällä tekniikalla voitaisiin objektien väliin piirtää pystysuora viiva, jonka vaikutuksesta objektit siirrettäisiin toisiinsa kiinni. Tasausele ikään kuin imee tasattavia objekteja itseään kohti. Toteutettu eleiden avulla tapahtuva tasaus eroaakin esimerkiksi puikkotyökaluista, joissa tasaus perustuu kursorin avulla tapahtuvaan objektien työntämiseen. Tällöin tasaus tapahtuu jokaisen objektin kohdalla saman reunan mukaan.

Elekomennot objektien lisäämiseen ja poistamiseen ovat toiminnoiltaan identtiset kontekstivalikosta löytyvien komentojen kanssa. Tällä ratkaisulla pyritään tukemaan eri

käyttäjien tarpeita ja mieltymyksiä. Oletuksena on, että kontekstivalikkokomentoihin verrattuna elekomentojen käyttöä suosivat henkilöt, jotka haluavat työskentelystä erityisen tehokasta.

Flatlandissa [Mynatt *et al.,* 1999; Igarashi *et al.,* 1999] elekomentojen ja piirtojen erottamiseen käytetään stylus-kynässä olevaa painiketta. Omassa toteutuksessani eleiksi tarkoitetut piirrot erotetaan muista komennoista objektin valinnan perusteella. Ennen elekomennon antamista tulee käyttäjän aina valita jokin objekti koskettamalla sitä sormella. Valittu objekti ilmoitetaan punaisin reunuksin. Nyt sormen avulla tehtävät piirrot tulkitaan eleinä, ja käyttäjä voi esimerkiksi luoda valitun objektin alle uuden lapsiobjektin.

5.4.3. Tekstinsyöttö ja objektien siirtäminen

Luonnollisia syötteitä hyödynnetään sovelluksessa myös objekteja kuvaavien tekstien syöttämisessä. Valitsemalla haluttu objekti ja suorittamalla kontekstivalikosta text-komento aktivoidaan tekstinsyöttötila. Vaihtoehtoisesti tekstinsyöttötila voidaan asettaa käynnistyväksi automaattisesti silloin, kun on luotu uusi objekti. Käyttäjä voi sormea käyttäen kirjoittaa objektin rajojen sisäpuolelle haluamansa tekstin (kuva 5.5b). Käytännössä syötteen ei tarvitse olla edes tekstiä, vaan objektin rajojen sisäpuolelle voidaan vapaasti piirtää mitä kuvioita tahansa. Tekstin ei tarvitse muodostua jatkuvasta eleestä, vaan käyttäjä voi nostaa sormea ja tehdä merkintöjä objektin eri kohtiin. Tekstinsyöttötila suljetaan, kun käyttäjä napauttaa sormella mitä tahansa objektin ulkopuolella. Kaavioeditorissa ei ole tekstintunnistusta vapaasti kirjoitetulle tekstille. Käyttäjän tulee valita joko vapaalla kädellä tai näppäimistön avulla kirjoitetun (kuva 5.5a) tekstin väliltä.

(a) (b)

Kuva 5.5. Näppäimistöltä annettu (a) ja käsin kirjoitettu teksti (b).

Objektien siirtämiseksi olen toteuttanut tekniikan, jonka avulla objektien siirtämistä taululla pyritään helpottamaan. Ratkaisu ei perinteisessä mielessä käytä eleitä, sillä sen käyttö ei vaadi varsinaisten näkyvien piirtojen tekemistä, mutta ideatasolla se pohjautuu luonnollisen kaltaiseen vuorovaikutukseen. Haluttu objekti tulee ensin merkata siirrettäväksi napauttamalla sitä kaksi kertaa sormella, ikään kuin tuplaklikkaamalla. Tämän jälkeen objekti muuttuu vaaleaksi ja ilmoittaa näin, että valinta onnistui. Nyt käyttäjä voi siirtää objektin koskettamalla kerran taulua kohdassa, joka halutaan objektin uudeksi sijaintipaikaksi.

Toteuttamani ratkaisu objektien siirtämiseen on lähellä Rekimoton [1998] pick-and-drop-tekniikkaa ja Geißlerin [1998] objektin ”kantamista”. Perinteisen, raahaamiseen perustuvan objektien siirtämisen haittapuolet tulevat esille silloin, kun kohdepaikka on fyysisesti pitkän matkan päässä. Sormen pitäminen kiinni taulun pinnassa voi osoittautua hankalaksi ja epämiellyttäväksi esimerkiksi silloin, kun sormi on nihkeä tai kosketustunnistus heikkolaatuinen.

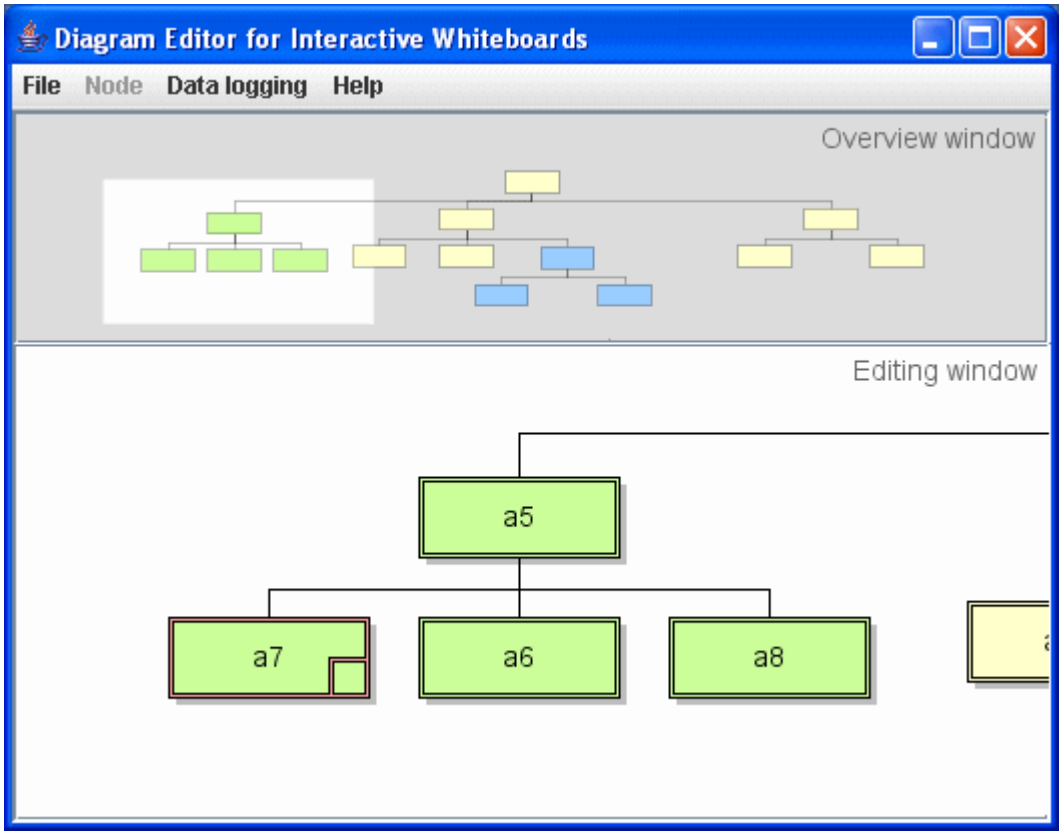
5.5. Tilanhallinta

Olen toteuttanut kaavioeditorisovellukseen eri ominaisuuksia, joiden on tarkoitus tukea tilanhallintaa interaktiivisella tussitaululla. Tässä kohdassa esittelen yleiskuva- ja muokkausnäkymiin pohjautuvan ikkunoinnin, muokkausnäkymän sijainnin muuttamisen raahaamalla, kiihtyvyyden käyttämisen muokkausnäkymän hallinnassa ja nuolten hyödyntämisen informaation paikantamisessa.

5.5.1. Yleiskuva- ja muokkausnäkymä

Kaavioeditorisovelluksen tilanhallinnassa olen päätynyt ratkaisuun, jonka avulla käyttäjä on jatkuvasti tietoinen kaikesta taululla olevasta informaatiosta ja voi suhteuttaa tekemänsä muokkaukset kaaviokokonaisuuteen. Kaavioita käsiteltäessä informaation määrä voi kaaviotyypistä riippuen kasvaa huomattavan suureksi, jolloin kokonaisuuksien hahmottaminen vaikeutuu. Kaavionkäsittelyn vaatima tila kasvaa nopeasti myös siitä syystä, että esteettisesti miellyttävän kaavion luomiseksi objekteja ei voida sijoittaa toisiinsa kiinni. Tyhjän tilan käyttö on ryhmittelykeinona tärkeä. Näiden tavoitteiden tukemiseksi olen interaktiivisia tussitauluja varten jakanut kaavioeditorisovelluksen pystysuunnassa kahteen ikkunaan: yleiskuva- ja muokkausnäkymään.

Kahdesta näkymästä ylempi, yleisnäkymä (kuva 5.6), on tarkoitettu auttamaan käyttäjää hahmottamaan kaavioon kuuluva informaatio ja näin helpottamaan sen hallitsemista. Yleisnäkymän avulla käyttäjä näkee jatkuvasti, missä kohdassa kaaviota kulloinkin ollaan. Tämä fokus kuvataan yleisnäkymässä vaalealla nelikulmiolla, joka siirtyy automaattisesti käyttäjän liikkeiden perusteella. Alempana oleva muokkausnäkymä (kuva 5.6) on tarkoitettu varsinaiseen kaavion muokkaamiseen, ja sitä käyttämällä tulee suorittaa kaavioon kohdistuvat toimenpiteet, kuten objektien lisääminen ja siirtäminen. Kaaviossa voi navigoida yleisnäkymää käyttäen, mutta muokkausoperaatioita sen avulla ei voi suorittaa.



Kuva 5.6. Ylempänä kaavioeditorin yleisnäkymä ja alempana muokkausnäkymä.

Oletustilassa yleisnäkymä on käytössä ja se on muokkausnäkymää pienempi. Yleisnäkymän ja muokkausnäkymän suhdetta voi muuttaa raahaamalla ikkunoiden välissä olevaa erotinpalkkia pystysuoraan ylös tai alas. Haluttaessa yleisnäkymä voidaan myös piilottaa kokonaan napauttamalla muokkausnäkymässä tyhjää kohtaa kahdesti tuplaklikkauksen tavoin. Samalla keinolla yleisnäkymä saadaan myös tuotua takaisin esiin. Näin käyttäjälle annetaan mahdollisuus mukauttaa sovelluksen käyttöliittymää parhaaksi katsomallaan tavalla.

Yleisnäkymän avulla voidaan valita haluttu työskentelykohta muokkausnäkymään. Navigointi tapahtuu koskettamalla haluttua kaavion kohtaa yleisnäkymästä. Tämän ratkaisun avulla voidaan muokkauskohtaa vaihtaa nopeasti suuremmissakin kaavioissa. Muokkausnäkymän siirtämiseen käytettävä toinen tekniikka, vierittäminen, esitellään alakohdassa 5.5.2.

Kaavioeditorin yleisnäkymässä on käytetty automaattisesti tapahtuvaa zoomausta eli skaalausta. Tällä lähestymistavalla pyritään estämään tilanteet, joissa käyttäjä zoomaamisen takia eksyy virtuaaliseen tilaan tai kadottaa etsimänsä informaation [Mynatt *et al.*, 1999; Pook *et al.*, 2000; Büring *et al.,* 2006].

Kun käyttäjän luoman kaavion leveys tai korkeus kasvaa niin suureksi, ettei se enää näy kokonaan yleisnäkymässä, skaalataan yleisnäkymää automaattisesti pienemmäksi, kunnes kaavio mahtuu jälleen näkymään. Vastaavasti jos käyttäjän suorittamien

objektien siirtojen tai poistojen seurauksena kaavion kokonaisleveys tai -korkeus pienenee, skaalataan yleisnäkymää jälleen suuremmaksi. Kuvassa 5.7 näkyy kaavion leveyden mukaan suoritettava yleisnäkymän skaalaus. Yleisnäkymässä ei kuitenkaan missään tilanteessa siirrytä muokkausnäkymää tarkemmalle tasolle.

Kuva 5.7. Yleisnäkymän skaalaaminen pienemmäksi kaavion leveyden kasvaessa.

Yleisnäkymässä kaavion ylin objekti pysyy aina sivuttaissuunnassa ikkunan keskellä riippumatta siitä, leveneekö kaavio oikealle tai vasemmalle. Tämän ratkaisun avulla yleiskuvaikkunan skaalauksessa pyritään johdonmukaisuuden avulla siihen, että käyttäjä voi nopeasti hahmottaa kaavion rakenteen suhteuttamalla muut objektit hierarkiassa ylimpänä olevaan objektiin.

Automaattisen skaalaamisen avulla pyritään paitsi minimoimaan informaation kadottaminen, myös vähentämään käyttäjän tarvitsemien komentojen ja toimenpiteiden määrää. Samanaikainen sivuttais-, pysty- ja syvyyssuunnan kontrollointi kaavion muokkaamisen ohella muodostuisi haastavaksi tehtäväksi.

Yleisnäkymä ei sisällä kaikkea sitä informaatiota, joka kaaviosta näkyy muokkausnäkymässä. Esimerkiksi objekteja kuvaavat tekstit jätetään yleisnäkymässä näyttämättä. Yksityiskohtien näyttäminen on yleisnäkymässä käytännössä turhaa, sillä niiden tulkitseminen on poikkeuksetta tarkkuustasosta johtuen mahdotonta. Lisäksi tällä pyritään selkiyttämään yleisnäkymän asemaa kaavion hahmottamisen ja navigoinnin apuna, ei muokkausten tekemisen välineenä.

Aikaisemmissa interaktiivisille tussitauluille suunnatuissa kaavioeditoreissa erillistä yleisnäkymäikkunaa on käytetty CASE-työkalu Knightissa [Damm *et al.,* 2000]. Knightin pienikokoisen apuikkunan avulla voi zoomata ja vierittää varsinaista työskentelynäkymää. Damm *et al.* eivät kuvaa kovin tarkoin yleisnäkymäikkunan toimintaa, eikä esimerkiksi yleisnäkymän skaalautuvuudesta ole tietoa. Omassa ratkaisussani pyrin siihen, että yleisnäkymäikkuna toimii kaavionmuokkauksessa

aktiivisena apuvälineenä. Sen avulla ei ainoastaan siirrytä eri kohtaan muokkausnäkymässä, vaan se auttaa käyttäjää hahmottamaan kaaviokokonaisuuden ja helpottamaan muokkaustoimenpiteiden suunnittelua.

5.5.2. Muokkausnäkymän hallitseminen

Muokkausnäkymässä näytettävää kaavion kohtaa käyttäjä voi vaihtaa myös muokkausnäkymästä vierittämistä käyttämällä. Vierittäminen tapahtuu raahauseleen avulla, joka kohdistetaan muokkausnäkymän tyhjään kohtaan, kun yhtään objektia ei ole valittu. Kaavio siirtyy nyt raahausliikkeen osoittamaan suuntaan ja uutta tilaa tai kaaviota paljastuu vastakkaiselta puolelta. Muokkausnäkymän vierittäminen tämän raahausliikkeen avulla on itse asiassa lähellä elettä, joka kuvaa perinteisen liitutaulun kääntämistä tai paperin siirtämistä pöydällä. Kaavioeditorin alkuperäisessä versiossa työskentelyalan vierittäminen tapahtui vierityspalkkien avulla, mutta interaktiivisten tussitaulujen fyysisten mittojen takia kurottelu näitä käyttöliittymäelementtejä kohti olisi erittäin hankalaa.

Vierittäminen on tarkoitettu tukemaan yleisnäkymän avulla tapahtuvaa muokkausnäkymäkohdan valitsemista. Yleisnäkymän käyttöön verrattuna vierittäminen on nopeampi tapa tehdä esimerkiksi lyhyitä siirtoja. Tällöin käyttäjän ei tarvitse siirtää kättään kohti yleisnäkymää, vaan muokkausnäkymää voidaan vierittää raahauseleen avulla lähellä käsiteltävää kohdetta. Lisäksi vierittämisen avulla tapahtuva muokkauskohdan siirtäminen on ainoa keino silloin, jos yleisnäkymä on piilotettu.

Jos käyttäjä haluaa hallita muokkausnäkymää pelkän vierittämisen avulla, muodostuvat pitkät siirrot ongelmallisiksi. Perinteisessä työpöytäympäristössä voidaan kursori siirtää ruudun laidalta toiselle varsin pienen hiiren liikkeen avulla. Sen sijaan interaktiivisten tussitaulujen kanssa vastaavan siirron suorittamiseksi käyttäjän tulee tehdä taulun fyysistä leveyttä vastaava liike.

Vierittämisen tukemiseksi otetaan muokkausnäkymän liikuttamisessa huomioon raahausliikkeen nopeus. Kaavioeditori tunnistaa sormen kosketusta vastaavat koordinaatit ja laskee ajan, joka niiden välillä siirtymiseen kuluu. Raahausliikkeen nopeudesta riippuen vieritettyä matkaa kasvatetaan. Rivakan liikkeen avulla muokkausnäkymä siirtyy siis vastaavaan pituista, mutta hidasta liikettä enemmän.

Kiihdytyksen hyvänä puolena tasaisen nopeaan liikkeeseen verrattuna on se, että hitaampia raahausliikkeitä käyttäen voidaan suorittaa tarkkuutta vaativia toimenpiteitä. Kiihdytystä hyödynnetään ainoastaan muokkausnäkymän vierittämisessä, eikä esimerkiksi objektien siirtämisessä.

Työpöytäympäristössä osoitinlaitteen liikkeen kiihdytystä on käytetty pitkään esimerkiksi hiirten ohjaamisessa. Interaktiivisiin tussitauluihin perustuvassa vuorovaikutuksessa osoitinlaitteen liikkeen kiihdyttämistä ei sen sijaan ole juurikaan käytetty, sillä usein on pyritty perinteistä liitu- tai tussitaulua vastaavaan metaforaan. Kaavioeditorisovellusten laajat informaatiomäärät yhdistettynä interaktiivisten

tussitaulujen edellyttämiin lyhyisiin piirtoihin avaavat kuitenkin mielestäni mahdollisuuden kiihdytyksen avulla tapahtuvaan navigointiin.

Mynattin [1999] interaktiivisia tussitauluja koskevassa tutkimuksessa käyttäjät pelkäsivät hukkaavansa materiaalin taulun reunojen ulkopuolelle. Kaavioeditoritoteutuksessani tätä pyritään ehkäisemään ensisijaisesti yleisnäkymän käytön avulla. Käyttäjä voi kuitenkin halutessaan piilottaa yleisnäkymäikkunan, jolloin kaavion hukkaamisen mahdollisuus kasvaa.

Eteen voi tulla tilanne, jossa käyttäjä vahingossa vierittää muokkausnäkymää enemmän kuin oli tarkoitus (kuva 5.8a), ja kaavio siirtyy sovelluksen reunan ”taakse” piiloon. Jotta käyttäjä ei tässä tilanteessa kadottaisi kaaviota kokonaan, ilmoitetaan kaavion sijainti muokkausnäkymän reunassa nuolen avulla (kuva 5.8b). Kaavio voidaan nuolen osoittaman suunnan perusteella tuoda takaisin esiin raahausliikkeen avulla. Nuoli piilotetaan, kun kaavion reuna yltää jälleen muokkausnäkymään.

(a) (b)

Kuva 5.8. Muokkausnäkymän reunalla oleva objekti (a) ja muokkausnäkymän ulkopuolella olevan objektin sijainnin näyttävä nuoli (b).

Käyttäjän tekemien tahattomien siirtojen vaikutuksia voitaisiin vähentää myös estämällä kokonaan objektien siirtäminen näkymän reunojen ulkopuolelle. Tämän lähestymistavan haittavaikutus on kuitenkin se, että mielikuva rajattomasta virtuaalisesta tilasta menetetään, kun objekti ikään kuin törmää näkymän reunaan. Olen omassa editorisovelluksessani pyrkinyt tukemaan rajattoman työskentelytilan käyttöä, jolloin nuolten käyttäminen informaation paikantamisessa on mielestäni soveltuvampi ratkaisu.

5.6. Yhteenveto

Tässä luvussa olen esitellyt aikaisempien lukujen teoriatiedon pohjalta jatkokehitellyn kaavioeditorisovelluksen interaktiivisille tussitauluille. Toteutustyössä olen pyrkinyt luomaan sovellukselle helpon tavan antaa komentoja ja hallita taulun fyysistä sekä virtuaalista tilaa. Tässä kappaleessa pohdin toteutettujen ominaisuuksien taustoja, tulevaa käyttöä ja suunnitteluvaiheessa punnittuja vaihtoehtoja.

Kontekstivalikon avulla on tarkoitus käyttää alkuperäisessä kaavioeditorissa perinteisten valikoiden avulla toimivia komentoja, kuten tiedostojen avaamista ja poistoa. Valikon suunnitteluvaiheessa päädyin jakamaan komennot kahteen eri valikkoon sen perusteella, pidetäänkö sormea objektin vai tyhjän tilan päällä. Riskinä tässä ratkaisussa on se, kuinka käyttäjät tottuvat valikon kontekstiperusteisuuteen.

Eleiden käyttöä on aikaisemminkin tutkittu interaktiivisten tussitaulujen yhteydessä. Päädyin toteuttamaan kaavioeditorisovellukseen eleitä komentojen antamiseen, koska niiden käyttö on osoittautunut lupaavaksi tavaksi luonnollisemman vuorovaikutuksen aikaansaamisessa. Objektien luomista ja poistamista eleiden avulla on tutkittu jo aikaisemmissa interaktiivisille tussitauluille tarkoitetuissa kaavioeditoreissa. Tasaus-eleiden käyttöä interaktiivisten tussitaulujen yhteydessä on sen sijaan tutkittu vähemmän. Suoraviivaisiin objektien luomisiin ja poistamisiin verrattuna tasaamisen toteuttaminen on haastavampaa. Elettä tulkittaessa pitäisi pystyä huomioimaan kaavion rakenne ja objektien sijoittelu. Jokaisella käyttäjällä on omat oletuksensa siitä, kuinka tasauseleen piirtämisen tulisi vaikuttaa tasattaviin objekteihin.

Käsin kirjoitetun tekstin syöttäminen objekteihin on toteutettu kaavioeditori-sovellukseen, koska näppäimistön käyttö tekstin antamiseen on tussitaululaitteiden yhteydessä kömpelöä. Kuitenkaan sovelluksessa ei ole tekstintunnistusta, jonka avulla käsin kirjoitetun tekstin muuttaminen tietokoneen ymmärtämään muotoon olisi mahdollista. Tämän tutkimuksen puitteissa tunnistustekniikkaa ei ole sovellukseen lisätty, vaikka se avaisi mielenkiintoisia tutkimuksellisia näkökulmia työhön.

Sovelluksen jatkokehittelyn alkuvaiheessa harkinnassa oli useita erilaisia tilan-hallintatekniikoita. Yksi vaihtoehto oli käyttää sovelluksessa ainoastaan yhtä näkymää, jota käyttäjä pystyisi eleiden avulla zoomaamaan lähemmäs ja kauas. Tämän tekniikan avulla lähemmäs zoomattaessa voitaisiin tarkentaa objektin tarkkuustasoa eli vaikkapa lisätä teksti objektin koordinaateista. Kuitenkin kahta erillistä näkymää ja automaattista skaalausta käytettäessä käyttäjällä on jatkuvasti kokonaiskuva käsiteltävästä kaaviosta, mikä auttaa häntä navigoimaan kaaviossa ja suhteuttamaan tehdyt muutokset koko kaavion tasolla. Lisäksi manuaalisen zoomaamisen riskinä on tarkkuustason hallitsemisen vaikutus muun työskentelyn tehokkuuteen.

Muokkausnäkymän hallitsemisessa on päädytty raahaamiseen perustuvaan vierittämiseen. Vierittäminen toimii vaihtoehtoisena tapana yleisnäkymään perustuvan liikuttamisen ohella. Yleisnäkymän ensisijaisena tarkoituksena on tarjota käyttäjälle jatkuvasti kokonaiskuva muokattavasta kaaviosta. Yleisnäkymän skaalaamisen asteen kasvaessa kaavion koko taululla pienenee, jolloin tarkkojen navigointikomentojen antaminen yleisnäkymän painalluksen avulla voi vaikeutua. Raahauksen avulla tapahtuva vierittäminen on monille käyttäjille tuttu tekniikka esimerkiksi Adobe Reader -sovelluksesta, jossa raahausliikkeiden avulla usein suoritetaan lyhyitä siirtymiä. Harva sen sijaan selaa suurempaa tekstiä pelkästään raahaamista käyttäen.

Kiihdytyksen hyödyntäminen muokkausnäkymän vierittämisessä on ominaisuus, jonka käytöstä tussitaulujen yhteydessä ei juurikaan ole aikaisempaa tietoa. Tässä suhteessa toteutuksen avulla on mahdollista tutkia, kuinka perinteistä raahaamista voidaan tehostaa yksinkertaisella tekniikalla.

6. Sovelluksen testaus

Tässä luvussa esittelen sovellukselle suorittamani testauksen ja raportoin siitä saadut tulokset. Testauksen tavoitteena on saada tietoa jatkokehitellyn kaavioeditori-sovelluksen käytöstä todellisen kaltaisissa tilanteissa. Aluksi esittelen testaukselle asetetut tarkemmat tavoitteet ja sen, kuinka ne näkyvät laadituissa tehtävissä. Testaustapahtuman läpivientiä käsittelevässä kohdassa esittelen testiympäristön ja testaustapahtuman kulun. Lopuksi raportoin testauksesta saadut tulokset sekä suhteutan ne suunnitteluvaiheessa asetettuihin oletuksiin.

6.1. Asetetut tavoitteet

Jatkokehittämääni kaavioeditorisovellukseen on toteutettu useita eri ominaisuuksia. Kaikkien ominaisuuksien kattavaa testausta ei tämän tutkimuksen puitteissa ole suoritettu. Tämä ei olisi ollut mielekästä, sillä esimerkiksi erilaisten valikoiden, kuten pyöreiden kontekstivalikoiden, käyttöä on testattu aikaisemmin laajalti. Sen sijaan testauksen painotuksia mietittäessä on otettu huomioon toteutukselle asetetut kaksi päätavoitetta. Komentojen antamiseen ja tilanhallintaan liittyvistä ominaisuuksista testattaviksi on valittu ne, joiden avulla parhaiten saataisiin uutta tietämystä kaavioeditorien käyttämisestä interaktiivisilla tussitauluilla.

Suunnittelussa asetettiin kolme erillistä tavoitetta. Testauksen avulla on tarkoitus saada tietoa:

1) eleiden toimivuudesta objektien luomisessa, poistamisessa ja tasaamisessa,
2) yleisnäkymään ja muokkausnäkymään perustuvan tilanhallintaratkaisun käytettävyydestä, ja
3) objektien siirtämisestä raahaus- ja click-and-drop-tekniikkaa käyttäen.

Ensimmäinen tavoite pohjautuu luonnollisuuteen pyrkivien eleiden tutkimiseen. Eleet ovat suorakäyttöisyytensä vuoksi mielenkiintoisia interaktiivisten tussitaulujen yhteydessä. Erillistä metaforaa, kuten esimerkiksi valikkoa, ei tarvita. Kontekstivalikon-kin avulla komentojen antaminen on mahdollista, mutta oletuksena on, että eleiden käyttö on vapaalla kädellä tapahtuvassa syötteidenannossa miellyttävä ja tehokas tapa. Lisäksi eleiden avulla tapahtuvaa objektien tasaamista ei ole aikaisemmissa kaavio-editoritutkimuksissa juurikaan käsitelty.

Toinen tavoite käsittelee sitä, kuinka käyttäjät mieltävät yleisnäkymän navigoinnin apuvälineenä ja käyttävät sitä todellisissa tilanteissa. Yleisnäkymästä valitsemisen lisäksi työskentelykohtaa voidaan muuttaa raahaamalla muokkausnäkymää. Oletuksena on, että suurikokoisia kaavioita käsiteltäessä käyttäjät navigoivat mieluummin yleis-näkymän avulla. Vastaavasti taas lyhyiden muokkausnäkymän siirtojen tekemiseen käytetään ennemmin raahaamista. Tämän työn puitteissa tilanhallinnan testauksessa ei

käytetä kiihdytystä vierittämisen apuna. Tarkoituksena on saada perustietoa siitä, kuinka normaali vierittäminen suhteutuu yleisnäkymän käyttöön. Kiihdytyksen ja mahdollisesti muidenkin tilanhallintatekniikoiden käytön testaaminen vaatisi kattavaa koeasetelmaa, joka olisi suunniteltu juuri eri tilanhallintaratkaisuiden testaamiseen.

Kolmannen tavoitteen avulla on tarkoituksena saada tietoa siitä, kuinka objektien käsittely onnistuu interaktiivisella tussitaululla. Suuren fyysisen kokonsa vuoksi pitkät yhtenäiset siirrot, kuten objektien raahaamiset, voivat osoittautua hankaliksi. Sormen siirtäminen ja samanaikainen pitäminen kiinni taulun pinnassa on oletuksen mukaan käytettävämpi tekniikka silloin, kun siirrettävä matka on melko lyhyt. Sen sijaan pitkiä siirtoja tehtäessä käyttäjät suosisivat ennemmin click-and-drop-tekniikkaa, jossa haluttu objekti merkataan ensin siirrettäväksi ja tämän jälkeen pudotetaan uuteen kohtaan.

Koska tehtävissä keskityttiin eleiden, tilanhallinnan ja objektien siirtämisen testaamiseen, käyttäjien ei tarvinnut kiinnittää huomiota objektien käsittelyn muihin mahdollisuuksiin. Tehtävänannoissa olevissa tavoitekaaviokuvissa objektit olivat kooltaan samoja, mutta tehtävien arvioinnissa tähän ei kiinnitetty huomiota. Koehenkilöille kuitenkin näytettiin, kuinka objektien kokoa voi haluttaessa muuttaa.

Testauksen ulkopuolelle jätettyjen ominaisuuksien, kuten kontekstivalikon, objektin vanhemman vaihtamisen ja muokkausnäkymän piilottamisen, käyttö oli testin aikana estetty. Lisäksi eleiden testaamiseksi sovelluksessa oletuksena käytössä oleva lapsiobjektien siirron estäminen vanhempia ylemmäksi oli ohitettu. Muutoin koehenkilöt voisivat tasata objekteja vaakasuunnassa nostamalla ne vanhempiaan ylemmäksi, jolloin objektit palautettaisiin automaattisesti samalle korkeudelle vanhemman alle. Koehenkilöille esiteltäviä ominaisuuksia ja komentoja oli asetettujen tavoitteiden tutkimiseksi jo useita, joten ylimääräisten seikkojen mukaan ottamista tuli rajoittaa. Tällä ratkaisulla pyrittiin estämään se, että testitilanteessa ylimääräiset ominaisuudet veisivät käyttäjän huomiota varsinaisista testauskohteista.

6.2. Tehtävät

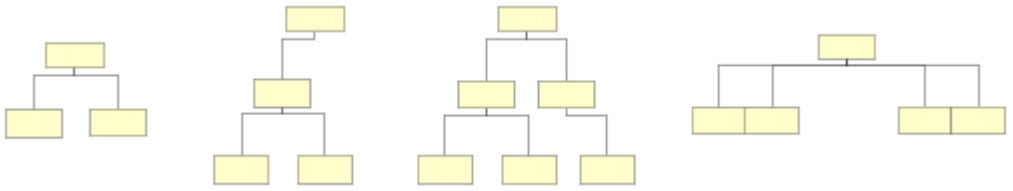
Testitehtävät suunniteltiin edellisessä kohdassa esiteltyjen tavoitteiden pohjalta. Tehtäviä oli yhteensä 12. Alla olevaan taulukkoon (taulukko 6.1) on koottu tehtävittäin niissä ensisijaisesti tutkitut ominaisuudet. Tehtävät ja niissä testatut ominaisuudet esitellään tarkemmin taulukon jälkeen.

Eleiden käyttöä testattiin periaatteessa kaikissa tehtävissä, sillä jokainen niistä pohjautui uusien objektien luomiseen. Tämän lisäksi objektien tasaaminen oli mukana 11 tehtävässä. Objektien poistoa testattiin kahdessa tehtävässä. Taulukkoon on tämän vuoksi merkitty eleet tutkittaviksi ominaisuuksiksi jokaiseen kohtaan. Kuitenkin ainoastaan neljässä ensimmäisessä tehtävässä eleiden käyttö oli pääasiallisena testauskohteena.

Tehtävä	1. Eleet	2. Objektien siirto	3. Tilanhallinta
1	X		
2	X		
3	X		
4	X		
5	(X)	X	
6	(X)	X	
7	(X)	X	
8	(X)		X
9	(X)		X
10	(X)		X
11	(X)		X
12	(X)		X

Taulukko 6.1. Testitehtävät ja niissä tutkitut ominaisuudet.

Kuvassa 6.1 on esitetty neljä ensimmäistä testitehtävää. Näissä tehtävissä koehenkilön tuli luoda yksinkertaisia kaavioita eleiden avulla. Tehtävissä 1-4 koehenkilö aloitti muokkaamisen kaaviosta, jossa oli ainoastaan pääobjekti valmiina. Pääobjektin alle tuli luoda uusia objekteja niin, että luotu kaavio vastasi tehtävänannossa määriteltyä kaaviota. Tehtävässä 1 (kuva 6.1a) totuttauduttiin vielä sovelluksen käyttöön, eikä luotuja kahta uutta objektia tarvinnut tasata. Tehtävässä 2 (kuva 6.1b) käyttäjän tuli tasata alimmalla tasolla olevat kaksi objektia alareunojen mukaan. Tehtävässä 3 (kuva 6.1c) oli tarkoituksena tasata kahdella alimmalla tasolla olevat objektit yläreunojen mukaan. Tehtävässä 4 (kuva 6.1d) ohjeistettiin käyttäjää tasaamaan objektit alareunojen ja pystysivujen mukaan.

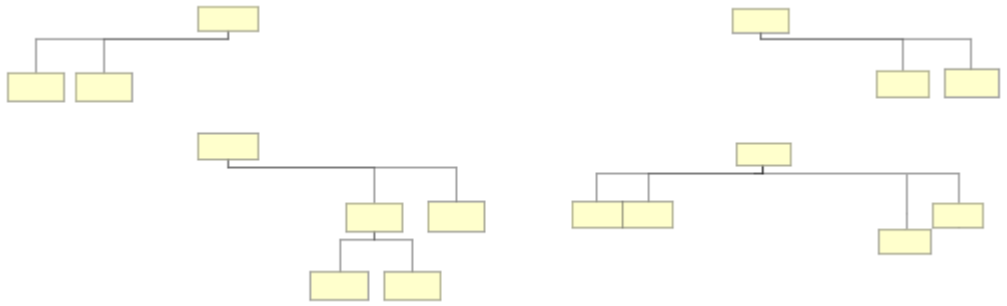


(a) (b) (c) (d)

Kuva 6.1. Tehtävät 1, (a) 2 (b), 3 (c) ja 4 (d) eleiden testaamiseen.

Objektien siirtämistä testattiin tehtävissä 5-7. Tehtävissä 5 ja 6 pyydettiin eksplisiittisesti koehenkilöä siirtämään objekteja taulun reunalta toiselle käyttäen joko raahaamista (tehtävä 5) tai click-and-drop-tekniikkaa (tehtävä 6). Tehtävien alussa annettiin koehenkilölle valmiiksi kaaviot, joista muokkaaminen aloitettiin objekteja siirtämällä. Tehtävän 5 lähtötilana toimiva kaavio on näkyvissä alla olevassa kuvassa

ylempänä (kuva 6.2a). Koehenkilöä ohjeistettiin siirtämään objekteja taulun reunalta toiselle sekä luomaan tämän jälkeen eleiden avulla kaksi uutta objektia. Tehtävän 5 tavoitekaavio on kuvassa alhaalla (kuva 6.2a). Kaksi uutta objektia tuli tasata alareunojen mukaan. Tehtävässä 6 tuli käyttäjän siirtää objekteja oikealta vasemmalle (kuva 6.2b). Siirretyt objektit tasattiin pystysivujen mukaan, jonka jälkeen oikealle luotiin ja tasattiin kaksi objektia alemman kuvan mukaisesti (kuva 6.2b).

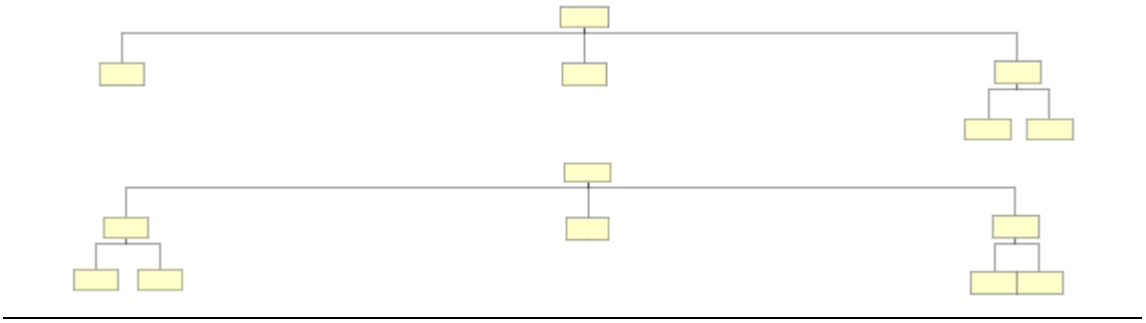


(a) (b)

Kuva 6.2. Tehtävät 5 (a) ja 6 (b) objektien siirtämisen testaamiseen. Ylempänä alkutilat ja alempana tavoitekaaviot.

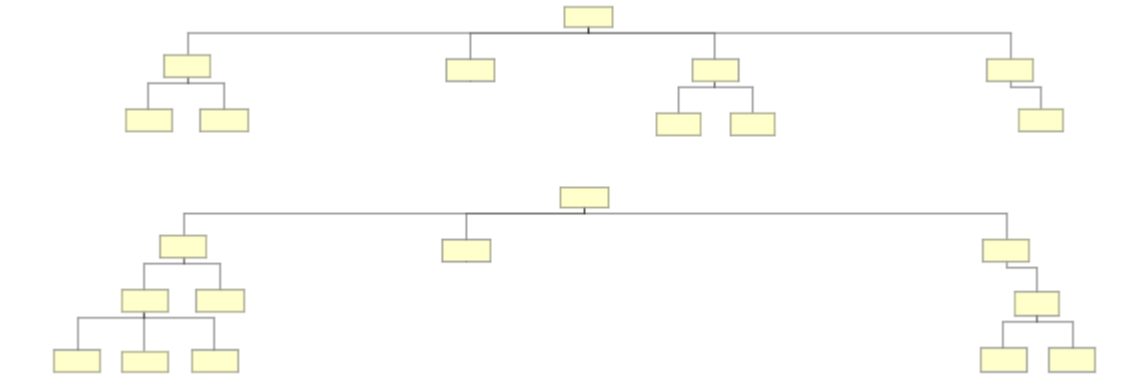
Tehtävien 5 ja 6 avulla tutkittiin kumpaakin objektien siirtämiseen käytettävää tekniikkaa. Tehtävässä 7 sen sijaan havainnoitiin, kumpaa tapaa koehenkilöt suosivat. Tässä tehtävässä koehenkilöitä ei tehtävänannossa kehotettu käyttämään tiettyä tekniikkaa. Koehenkilöiden tuli siirtää taulun oikealta laidalta vasemmalle laidalle kolme objektia, jonka jälkeen siirretyt objektit tasattiin vaakasuunnassa. Tehtävän tarkoituksena oli tutkia, kumpaa objektien siirtämiseen tarkoitettua tekniikkaa koehenkilöt käyttävät mieluummin.

Viidessä viimeisessä tehtävässä testattiin suurempien kaavioiden avulla tilanhallintatekniikoita. Aluksi koehenkilöä neuvottiin käyttämään joko yleisnäkymästä valitsemista (tehtävät 8 ja 9) tai muokkausnäkymän vierittämistä (tehtävät 10 ja 11). Tehtävässä 8 (kuva 6.3) koehenkilön tuli luoda vasemmalle kaksi uutta objektia ja tasata ne alareunojen mukaan. Tämän jälkeen oikealla olevat objektit tasattiin pystysivujen mukaan.



Kuva 6.3. Tehtävä 8 tilanhallinnan testaamiseen. Ylempänä alkutila ja alempana tavoitekaavio.

Tehtävässä 9 (kuva 6.4) koehenkilön tuli ensin luoda vasemmalle kolme uutta objektia, jonka jälkeen ne tuli tasata yläreunojen mukaan. Tämän jälkeen kaaviosta tuli poistaa kolme objektia. Lopuksi luotiin oikealle kaksi uutta objektia ja tasattiin ne yläreunojen mukaan.



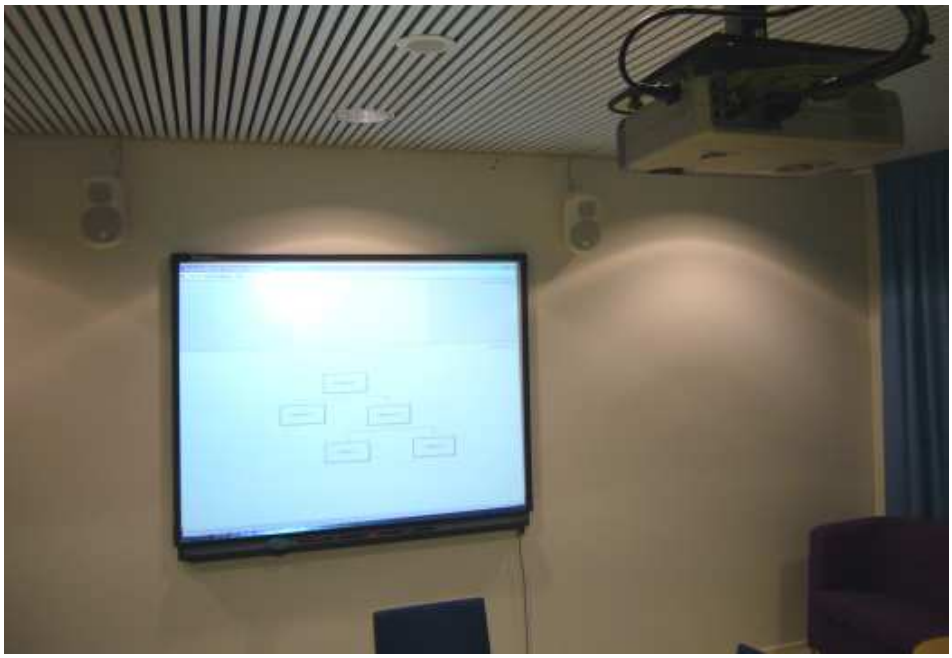
Kuva 6.4. Tehtävä 9 tilanhallinnan testaamiseen. Ylempänä alkutila ja alempana tavoitekaavio.

Tehtävät 8 ja 10 sekä 9 ja 11 ovat keskenään peilikuvia. Näin pyrittiin mahdollistamaan kahden eri tilanhallintatekniikan vertaileminen keskenään. Myös tilanhallintatehtävien lopuksi annettiin koehenkilölle mahdollisuus valita testaamistaan kahdesta tekniikasta mieleisempi. Viimeisessä tehtävässä (tehtävä 12) koehenkilö sai vapaasti navigoida valitsemallaan tekniikalla kaaviota käsitellessään. Tehtävässä tuli luoda neljä uutta objektia sekä tehdä vaaka- ja pystytasauksia.

Vaikka objektien siirtoa ja tilanhallintaa testattiin erityisesti jälkimmäisissä tehtävissä, neljässä ensimmäisessäkin tehtävässä oli hyvä käyttää mainittuja ominaisuuksia esimerkiksi lyhyisiin objektien raahauksiin ja näkymän vierittämisiin. Tähän ei koehenkilöä kuitenkaan erikseen ohjeistettu, vaan hän sai liikuttaa objekteja ja näkymää parhaaksi katsomallaan tavalla.

6.3. Testauksen läpivienti

Sovelluksen testaus tapahtui seinälle kiinnitetyn SMART Board 500 -tussitaululaitteen [Smarttech, 2007] avulla (kuva 6.5). Kuva taululle heijastettiin kattoon, taulun etupuolelle kiinnitetyn projektorin avulla. SMART Board -tussitaulun erillisiä stylus-kyniä tai pyyhekumia ei käytetty testissä, vaan kaikki syötteet annettiin pelkän sormen avulla. Interaktiivinen tussitaulu ja projektori olivat yhteydessä kannettavaan tietokoneeseen, jonka avulla käytettiin testisovelluksena toiminutta kaavio-editorisovellusta. Kokeen etenemistä kontrolloitiin kannettavalta tietokoneelta avaamalla eri tehtävissä tarvittavat alkutilakaaviot.



Kuva 6.5. Testauksessa käytetty SMART Board 500 -tussitaulu ja videotykki.

Kaavioeditorisovellus keräsi testin aikana automaattisesti tietoa kaavion muokkaustapahtumista. Jokaisen suoritetun komennon tai toimenpiteen yhteydessä tallennettiin kaavion tila, jotta tehtävän etenemistä voitiin myöhemmin tarkastella vaihe vaiheelta. Tämän lisäksi tallennettiin toiseen datatiedostoon muun muassa muokkaustapahtuman aika, kaavioon kuuluvien objektien koordinaatit ja suoritetun komennon nimi.

Testitapahtuman aluksi koehenkilöitä pyydettiin täyttämään taustatietolomake (liite 1). Lomakkeen avulla kerättiin tietoa siitä, kuinka kokeneita tietokoneiden käyttäjiä koehenkilöt olivat sekä olivatko he aikaisemmin käyttäneet esimerkiksi interaktiivisia tussitauluja tai kaavioeditorisovelluksia. Koehenkilöiltä ei edellytetty aikaisempaa kokemusta tutkimuksen aihepiiriin kuuluvista sovelluksista tai laitteista. Lomakkeen lopussa koehenkilöiltä pyydettiin lupa tiedon tallentamiseen analyysivaihetta varten. Lomakkeessa kerrottiin myös, ettei taustatietolomakkeeseen annettuja nimiä tuoda esille testauksen raportoinnin yhteydessä.

Esitietolomakkeen täyttämisen jälkeen koehenkilöille kerrottiin sanallisesti kaavio-editorisovelluksen käyttötarkoitus, joka on hierarkkisten rakenteiden mallintaminen objektien ja niitä yhdistävien suhteiden avulla. Samanaikaisesti sovelluksella näytettiin kuvan 5.1 mukainen esimerkkikaavio. Sanallisessa esittelyssä kerrottiin, kuinka kaavio rakentuu yhden pääobjektin alle ja miten jokaisella objektilla tulee pääobjektia lukuun ottamatta olla yksi vanhempiobjekti.

Tämän jälkeen koehenkilölle esiteltiin kuvan 5.1 kaavion avulla testitehtävissä tarvittavat komennot. Kaavion muokkaamiseen tarvittiin elekomennot objektien luomista, poistamista ja tasaamista varten. Objektien siirtäminen taas vaati raahaamisen ja click-and-drop-tekniikan käyttöä. Lisäksi tilanhallinta edellytti työskentelykohdan valitsemista yleisnäkymästä ja muokkausnäkymän vierittämistä raahaamisen avulla. Tarvittavien komentojen esittelyn jälkeen sai koehenkilö totuttautua sovelluksen käyttöön ja annettuihin komentoihin, kunnes tunsi olevansa valmis käyttämään niitä testissä. Totuttelun aikana koehenkilö sai kysyä apua, jos jonkin komennon käyttötapa unohtui. Käytännössä totutteluun kului yleensä muutama minuutti.

Varsinaisen testivaiheen alkaessa avattiin kaavioeditorisovellukseen näkyviin kaavio, josta tehtävän suorittaminen alkoi. Tehtävissä 1-4 lähtötilassa oli ainoastaan pääobjekti. Sen sijaan tehtävissä 5-12 lähtötilassa oli useampia objekteja. Tämän jälkeen koehenkilölle annettiin paperille piirretty tavoitekaavio, johon tehtävässä tuli pyrkiä. Tehtävänanto oli kaaviokuvan vieressä lisäksi sanallisessa muodossa. Tehtävänannon luettuaan käyttäjä sai aloittaa kaavion muokkaamisen. Kun koehenkilö oli suorittanut tehtävän ja pyysi seuraavaa tehtävää, siirryttiin eteenpäin.

Testin lopuksi koehenkilöä pyydettiin täyttämään arviointilomake (liite 2). Lomakkeen avulla kerättiin subjektiivista palautetta siitä, kuinka koehenkilö koki testissä käsitellyt ominaisuudet ja tekniikat. Koehenkilöltä pyydettiin arvioita esimerkiksi siitä, olivatko ominaisuudet vaikeita tai helppoja käyttää. Lomakkeen lopuksi koehenkilöt saivat kirjoittaa vapaasti kommentteja tai parannusehdotuksia sovelluksen käyttöön liittyen.

6.4. Tulokset

Yhden testisession kesto oli kokonaisuudessaan noin 40 minuuttia. Testiin osallistui kahdeksan koehenkilöä. Heistä miehiä oli kuusi ja naisia kaksi. Koehenkilöiden iät vaihtelivat välillä 21-36 ja keski-iäksi muodostui 30 vuotta. Aikaisemman tietokoneen käytön osalta testiin osallistuneet henkilöt olivat kokeneita, sillä kaikki olivat käyttäneet tietokoneita yli kymmenen vuotta. Lisäksi koehenkilöt käyttivät tietokoneita päivittäin.

Interaktiivisia tussitauluja oli aikaisemmin kokeillut koehenkilöistä kaksi. He olivat käyttäneet SMART Boardia esityksen pitämiseen tai tussitaululaitteen toimintoihin tutustumiseen. Graafiset editorit olivat tuttuja kaikille testiin osallistuneille ja niitä käytettiin keskimäärin muutaman kerran kuukaudessa. Yleisimmät käyttötarkoitukset olivat kuvankäsittely ja kaavioiden, kuten UML-kaavioiden, piirtäminen. Käytetyistä

sovelluksista mainittiin esimerkiksi Adobe Photoshop, CorelDRAW, Gimp, Microsoft Visio, Fujaba, Umbrello ja PowerPoint.

Kynää tai sormea syötteiden antamisessa oli käyttänyt kuusi kahdeksasta osallistujasta. Kynällä tai sormella annettavia syötteitä käytettiin keskimäärin harvemmin kuin kerran kuussa. Käytettyjä laitteita olivat esimerkiksi tuntopalautekynä, piirtopöytä ja kosketusnäyttökioski. PDA-laitteet olivat tutuimpia, sillä käytettyjä laitteita olivat Sharp Zaurus, iPAQ, Palm Pilot ja MyOrigo.

6.4.1. Testidatan analysointi

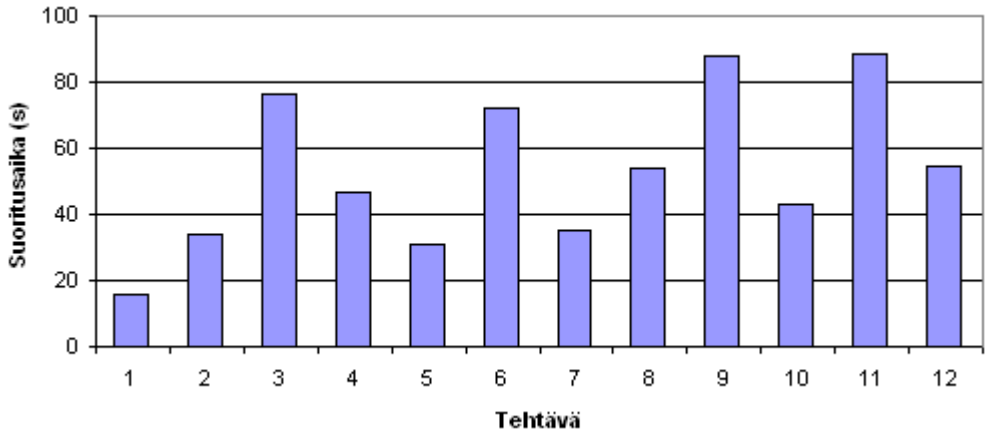
Koehenkilöt suoriutuivat varsin hyvin laskettaessa tehtävien oikeiden suoritusten prosentuaalinen määrä. Kaikkiaan 96 tehdyssä tehtävässä ohjeiden mukaisesti luotuja kaavioita oli 89,6 %. Eniten ongelmia oikeita suorituksia tarkasteltaessa oli tehtävässä 6, jossa luotiin kolme väärää kaaviota. Alla olevassa taulukossa 6.2 on eritelty tehtäväkohtaisesti onnistuneiden tehtävien määrä sekä syyt tehtävien mahdolliselle epäonnistumiselle.

Tehtävä	Onnistuneet	Syy epäonnistumiseen
1	8 / 8	
2	7 / 8	Puutteellinen tasaus
3	7 / 8	Puutteellinen tasaus
4	6 / 8	Puutteellinen tasaus
5	8 / 8	
6	5 / 8	Puutteellinen tasaus
7	8 / 8	
8	6 / 8	Luotu liikaa objekteja, puutteellinen tasaus
9	8 / 8	
10	8 / 8	
11	8 / 8	
12	7 / 8	Puutteellinen tasaus

Taulukko 6.2. Onnistuneiden tehtävien määrä ja epäonnistumisten syyt.

Tehtävänantoa vastaamattomista suorituksista 90,0 % johtui väärin tasatuista tai kokonaan tasaamatta jätetyistä objekteista. Tasauksen takia epäonnistuneet tehtävät eivät keskittyneet yhteen koehenkilöön, vaan puutteellisia suorituksia oli eri koehenkilöillä. Tehtävien suoritusten oikeellisuutta tarkasteltaessa huomioon otettiin objektien oikea määrä ja sijainti (lapsiobjekti luodaan oikean vanhemman alle) sekä ohjeiden noudattaminen tasauksessa. Tehtävien suorituksessa ei tarvinnut noudattaa järjestystä, jossa muokkaustoimenpiteet tehtävänannossa esitettiin.

Kaikkien testitehtävien suoritusaikojen keskiarvo oli 79 sekuntia. Kuvassa 6.6 on esitetty tehtäväkohtaiset suoritusaikojen keskiarvot. Nopeimmin koehenkilöt suoriutuivat luonnollisesti tehtävästä yksi (15,5 sekuntia). Eniten aikaa vei tehtävä 11 (88,2 sekuntia). Suoritusajat on laskettu ensimmäisen ja viimeisen muokkaus-toimenpiteen väliltä. Tehtävänannon lukemiseen kulunutta aikaa ei alla olevassa kuvassa ole siis huomioitu.



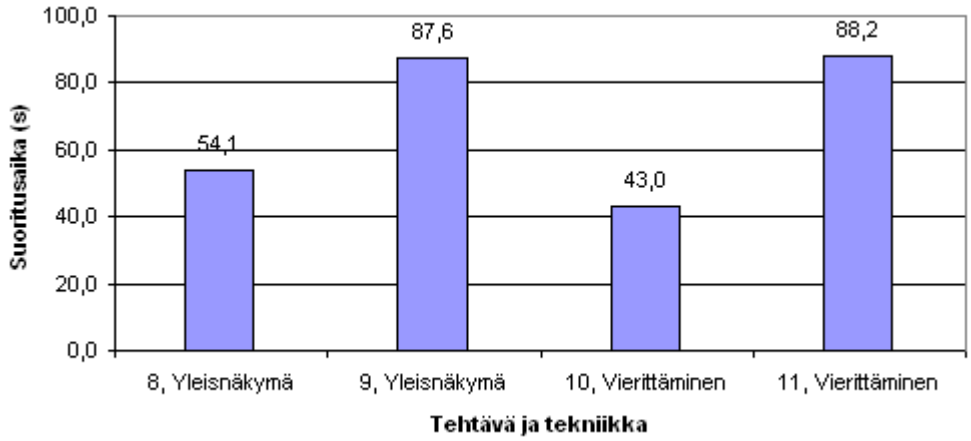
Kuva 6.6. Tehtävien suoritusaikojen keskiarvot.

Ensimmäinen tavoite testauksessa oli tarkkailla eleiden käytön toimivuutta todellisissa käyttötilanteissa. Uusien objektien luominen onnistui testitehtävissä hyvin. Tunnistamatta jääneitä luontieleitä oli kuusi yhteensä 285:sta, josta virheiden prosentuaaliseksi määräksi saadaan 2,1 %. Jokaista vaadittua objektin luontia kohti käyttäjät tekivät yhteensä 1,02 luontielettä. Kaikissa uusien objektien lisäyksissä uusi objekti luotiin oikean vanhemman alle, eikä uutta objektia tarvinnut siirtää tai poistaa. Tunnistamatta jääneissä luontieleissä ongelma oli se, etteivät eleen alku- ja loppupisteet olleet tarpeeksi lähellä toisiaan. Poistoeleitä käytettäessä kaikki käyttäjien syöttämät eleet tunnistettiin ja poistettava objekti oli oikea. Poistoeleitä suoritettiin yhteensä 25.

Eleiden käytössä vaikeimmaksi osoittautui tasaaminen. Tasauseleitä suoritettiin yhteensä 239, joista tunnistamatta jäi 18. Tunnistusvirheiden määrä oli 7,5 %, joista vaakatasausten virhemäärä oli 5,4 % ja pystytasausten 2,1 %. Eleiden tunnistuksessa vaikeutena oli eleen piirtäminen sallitun etäisyyden päähän tasattaviksi tarkoitetuista objekteista. Selvien tunnistamatta jääneiden tasauseleiden lisäksi oli useita tilanteita, joissa yhden sallitulla etäisyydellä olleen objektin lisäksi yksi tai useampi tasattavaksi tarkoitettu objekti ei ollut tarpeeksi lähellä piirrettyä elettä. Näissä tilanteissa koehenkilöiden tuli käsin siirtää tasaamatta jäänyttä objektia lähemmäksi muita tasattavia. Kun lasketaan suoritettujen tasauseleiden lukumäärä yhtä vaadittua tasausta kohti, saadaan luvuksi 1,9. Eri tasauksia eroteltaessa vaaditun vaakatasauksen suorittamiseksi koehenkilöt piirsivät keskimäärin 2,2 elettä ja vastaavasti jokaista

pystytasausta kohti 1,5 elettä. Tulokset kertovat siitä, että tasaamisessa koehenkilöt eivät tarkasti tienneet, millä periaatteella tasausele valitsee tasaukseen kuuluvat objektit.

Testauksen toisena tavoitteena oli tutkia kahden eri tilanhallintatekniikan, yleisnäkymän käyttämisen ja muokkausnäkymän vierittämisen, käyttöä. Alla olevassa kuvassa 6.7 on esitetty tilanhallintaa ensisijaisesti tutkineiden tehtävien 8-11 suoritusaikojen keskiarvot. Kuvassa vasemmalla olevat kaksi pylvästä kuvaavat tehtäviä 8 ja 9, joissa käytettiin yleisnäkymää tilanhallinnassa. Vastaavasti kaksi oikealla olevaa pylvästä kuvaavat tehtäviä 10 ja 11, joissa käytettiin tilanhallintaan muokkausnäkymän vierittämistä. Tehtävät 8 ja 10 sekä 9 ja 11 olivat keskenään peilikuvia.



Kuva 6.7. Suoritusajat tilanhallintaa testaavista tehtävistä 8-11.

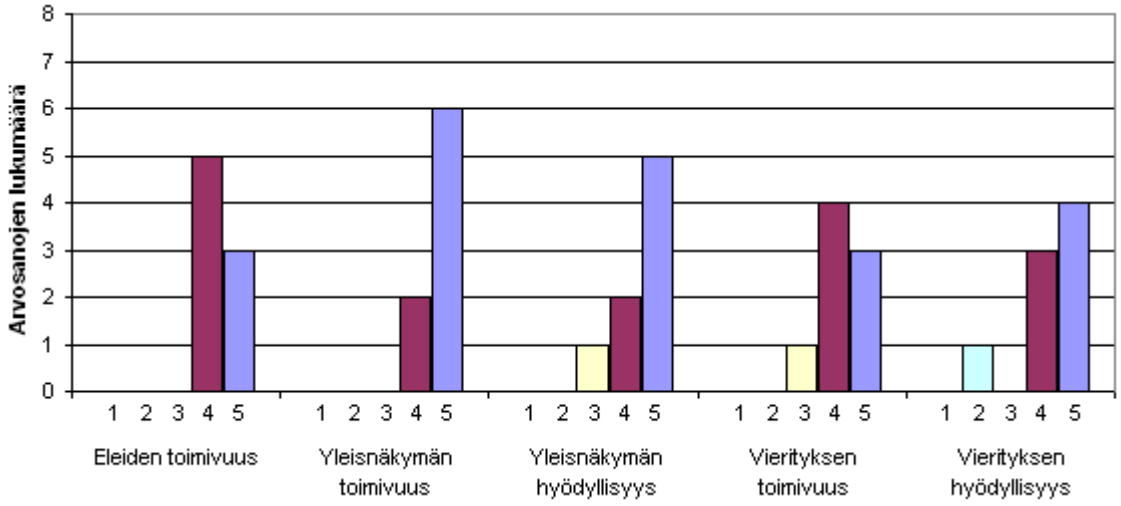
Tehtävien 8 ja 10 suoritusaikojen perusteella raahaamisen avulla toimiva vierittäminen (43,0 sekuntia) oli nopeampi kuin yleisnäkymän (54,1 sekuntia) käyttö. Tehtävissä 9 ja 11 yleisnäkymän (88,2 sekuntia) ja vierittämisen (87,6 sekuntia) käytössä ei ollut juurikaan eroja. Yleisnäkymän avulla liikuttaessa navigointiin käytettiin tehtävää kohti keskimäärin 3,6 komentoa. Muokkausnäkymän raahaamiseen perustuvaa vierittämistä käytettäessä vastaava raahaamiseleiden määrä oli 10,9 tehtävää kohden. Viimeisessä tehtävässä, jossa koehenkilöt saivat vapaasti valita navigointitavan, yleisnäkymän avulla muokkausnäkymää hallitsi seitsemän käyttäjää ja vierittämisen avulla yksi.

Tehtävissä 5-7 testattiin objektien raahaamiseen ja click-and-drop-tekniikkaan perustuvaa siirtämistä. Tehtävässä 5, jossa objekteja tuli siirtää raahaamistekniikan avulla, käyttivät koehenkilöt keskimäärin 1,6 raahaamiskomentoa yhtä siirrettävää objektia kohti. Vastaavasti click-and-drop-tekniikkaa testanneessa tehtävässä 6 koehenkilöt käyttivät siirtämiseen keskimäärin 2,1 komentoa yhtä siirrettävää objektia kohti. Click-and-drop-tekniikassa ongelmana oli sen muistaminen käyttötilanteessa sekä objektin valinnan puutteellinen ilmaiseminen. Tehtävässä 7, jossa koehenkilöt saivat käyttää objektien siirtämiseen haluamaansa tekniikkaa, raahaamista käytettiin

kymmenen objektin siirtämiseen ja click-and-drop-tekniikkaa vastaavasti 14 objektin siirtämiseen.

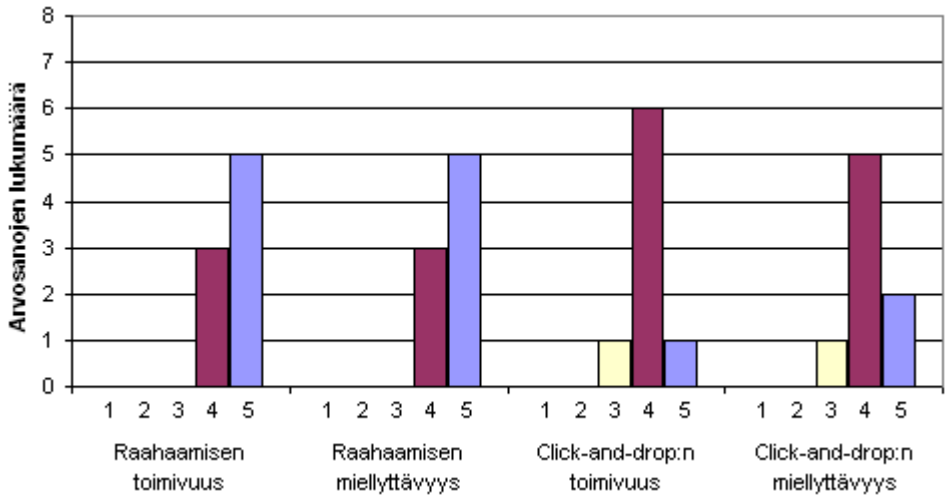
6.4.2. Subjektiiviset arviot ja palautteet

Testauksen lopuksi koehenkilöitä pyydettiin arvioimaan asteikolla 1-5 kaavio-editorisovelluksen eri ominaisuuksia (liite 2). Esimerkiksi ominaisuuden hyödyllisyyttä arvioitaessa numero 1 oli ”ei hyödyllinen” ja numero 5 taas ”hyödyllinen”. Näin numero 3 oli annetuista vaihtoehdoista siis neutraali. Alla olevassa kuvassa 6.8 on esitetty annettujen arvosanojen jakaumat viidestä ensimmäisestä arviointikohdasta (eleiden toimivuus, yleisnäkymän toimivuus, yleisnäkymän hyödyllisyys, vierityksen toimivuus ja vierityksen hyödyllisyys). Eleiden toimivuuden arvosanojen keskiarvoksi muodostui 4,4. Arvioinnissa eleiden toimivuus käsittää kaikki objektien käsittelyyn tarkoitetut eleet eli luomisen, poistamisen ja tasaamisen. Kahdesta eri tilanhallintatekniikasta yleisnäkymää pidettiin erittäin toimivana ominaisuutena, sillä sen arvosanaksi muodostui 4,8. Hyödyllisyytensä osalta yleisnäkymän arvosana oli 4,5. Vastaavasti raahaamiseen perustuvan vierityksen toimivuus ja hyödyllisyys arvioitiin hieman heikommaksi, sillä kummankin keskiarvo oli 4,3.



Kuva 6.8. Arviointilomakkeen kohtien 1-5 arvosanojen jakaumat.

Objektien siirtämiseen tarkoitettuja tekniikoita pyydettiin arvioimaan kohdissa 6-9. Kuvassa 6.9 näkyvät raahaamiselle ja click-and-drop-tekniikalle annettujen arvosanojen jakaumat. Raahaamisen toimivuudelle ja miellyttävyydelle muotoutui keskiarvoksi 4,6. Click-and-drop-tekniikan arvosanojen keskiarvot olivat raahaamista heikompia. Click-and-drop-tekniikan toimivuus sai keskiarvoksi 4,0. Miellyttävyyden osalta tekniikka sai keskiarvoksi 4,1.



Kuva 6.9. Arviointilomakkeen kohtien 6-9 arvosanojen jakaumat.

Kokonaisuudessaan lomakkeeseen annetut numeeriset arviot eri ominaisuuksista olivat varsin positiivisia. Parhaaksi toimivuudeltaan arvioitiin yleisnäkymä ja vastaavasti heikoimmaksi click-and-drop-tekniikka.

Arviointilomakkeen lopuksi annetuissa kommenteissa ja parannusehdotuksissa tulevat hyvin pitkälti esille samat asiat, jotka ovat nähtävissä myös data-analyysin pohjalta. Eleiden käyttöä koskevissa palautteissa tasaus tuli esiin useimmiten. Ongelmia tuotti se, että ”tasausele [oli] pirun tarkka”. Tämä on yhteydessä siihen, että editori ei aina tunnistanut kaikkia käyttäjän tasattaviksi haluamia objekteja. Myös eleiden ja muokkausnäkymän vierittämisen välillä esiintyi ongelmia, sillä ”joskus […] kävi niin, että kun halusi tasata objekteja, niin näyttö alkoikin liikkua”.

Tasauksen käyttöä haittasi se, että eleen suorittamisen jälkeen objektin valinta katosi. Tämä huomioitiinkin palautteissa, sillä ”tasauksen jälkeen valinnan tulisi säilyä”. Palautteissa tuli esille vaihtoehtoinen tekniikka tasauksen suorittamiselle, jossa elettä piirrettäessä osoitettaisiin tasattavaksi haluttavat objektit eksplisiittisesti. Käyttäjä voisi esimerkiksi aloittaa viivan piirtämisen tietyltä tasauskorkeudelta ja tämän jälkeen koukkausliikkeen avulla valita halutut objektit mukaan tasaukseen. Tasaamisen käyttö helpottui useilla koehenkilöillä loppua kohti. ”Monipuolisempi tasaus oli ok, kun sen ymmärsi”.

Sanallisista palautteista kävivät myös ilmi click-and-drop-tekniikan käyttöön liittyvät ongelmakohdat. Koehenkilöiden mielestä ”click-and-drop-tekniikassa pitäisi olla parempi indikaattori, että elementti on valittuna”. Lisäksi ehdotettiin, että click-and-drop-tekniikka voisi toimia päinvastaisella periaatteella. Nyt käyttäjän tulee napauttaa siirrettäväksi haluttavaa objektia kaksi kertaa ja tämän jälkeen osoittaa yhdellä painalluksella uusi paikka. Jos haluttu sijoituspaikka osoitettaisiinkin kaksoispainalluksen avulla, voisi objektin valita normaaliin tapaan yhden painalluksen

avulla. Näin käyttäjän ei tarvitsisi pohtia, kumpaa valintatapaa tulee eri tilanteissa käyttää.

Tilanhallintaan liittyvässä palautteessa pohdittiin sitä, voisiko sovellukseen yhdistää kannettavien tietokoneiden *levyhiirestä* (touch pad) tutun tekniikan, jossa näyttöä vieritetään esimerkiksi pystysuunnassa liikuttamalla sormea levyhiiren oikeassa reunassa. Saman tekniikan avulla muokkausnäkymää voitaisiin vierittää käyttämällä taulualan reunoilla olevia alueita. Toisaalta tämä voisi aiheuttaa kurottelua taulun reunoja kohti, joka aikaisemman tutkimustiedon perusteella ei ole suotava elementti vuorovaikutuksessa. Kahteen ikkunaan jaettua kaavioeditorisovellusta koehenkilöt pitivät onnistuneena, sillä ”ylänäkymä [on] hyödyllinen”.

Palautteessa oli myös kommentteja yleisesti interaktiivisia tussitaululaitteita koskien. ”Tasausviivaa piirtäessä viiva piirtyi hieman sormen alle”, joka johtuu laitteen kalibroinnista. Vaikka laitteen mukana toimitettavan kalibrointiohjelman avulla sormen painalluksen ja vastaavan näytön kohdan epätarkkuutta voidaan vähentää, ei tämä kaikissa tilanteissa riitä täydellisen tunnistustarkkuuden saamiseen. Lisäksi koehenkilöt joutuivat aluksi totuttelemaan projektorin käytöstä aiheutuvaan varjoon tussitaulun pinnalla. Niin ikään sormen ”pomppiminen” taulupintaa kosketettaessa oli monille käyttäjille odottamaton seikka. Sormen liu’uttaminen taulun pinnalla ei onnistu, jos sormea painetaan liian voimakkaasti taulupintaa kohti. Testin edetessä koehenkilöt kuitenkin oppivat nopeasti uuden käyttöympäristön.

Sanallisessa palautteessa eniten kiitosta saivat eleet, jotka olivat ”luonnollisia” ja niiden avulla oli ”helppoa luoda ja poistaa [objekteja]”. ”Piirteleminen [oli] luonnollista ja nopeaa”. Koehenkilöillä käytön sujuvuus parani poikkeuksetta tehtävien edetessä. ”Yleiskuva käytöstä oli miellyttävä ja käyttö tuli helpommaksi tehtävien edetessä”. Yleisesti kaavioeditorisovellusta pidettiin sanallisissa palautteissa toimivana. ”[…] yllätyin kuitenkin myönteisesti siitä, mitä kaikkea ja kuinka helposti ohjelmistolla pystyi tekemään. Ehdottomasti jatkotutkimusta ja jatkokehittelyä lisää.”

6.5. Yhteenveto

Olen tässä kappaleessa kuvannut jatkokehittämälleni kaavioeditorisovellukselle suoritetun testauksen. Testauksessa päätettiin keskittyä kolmen tavoitteen selvittämiseen: eleiden toimivuuteen, tilanhallintatekniikoiden vertailuun ja objektien siirtämiseen. Arvioin tässä kohdassa testauksen teknistä onnistumista sekä vertaan testauksesta saatuja tuloksia suunnitteluvaiheessa asetettuihin oletuksiin.

Testauksen läpiviennissä ei tullut esille erityisiä ongelmia. Kaikki kahdeksan koehenkilöä suoriutuivat testauksesta hyvin. Pääsääntöisesti tehtävät suoritettiin onnistuneesti, mutta myös virheellisiä kaavioita luotiin. Tehtävien vaikeusaste oli tässä suhteessa melko sopiva. Yksi koehenkilöistä kommentoi, että testitehtävien suorittaminen oli mukavaa, sillä tehtävät eivät vaikeutuneet liikaa loppua kohti, vaan myös viimeiset tehtävät pystyi suorittamaan hyvin.

Testauksen ajankäyttöön liittyen tuli hienoisena yllätyksenä se, kuinka vähän aikaa koehenkilöillä kului tehtävien suorittamiseen. Tämän vuoksi testitehtävien määrää olisi voinut kasvattaa suuremmaksikin. Toisaalta nykyisillä tehtävillä saatiin tietoa niistä seikoista, joita asetettujen tavoitteiden tutkiminen vaati. Laajemmassa testauksessa olisi voinut ottaa mukaan useampiakin testattavia ominaisuuksia, kuten esimerkiksi kontekstivalikon ja eleiden avulla annettavien komentojen vertailun. Kattavammat testiasetelmat ovatkin mahdollisia jatkokehityssuuntia, joiden avulla tutkimusta voidaan tulevaisuudessa jatkaa.

Eleiden käyttöön liittyvät testitulokset ovat keskeisimpiä siinä mielessä, että niiden avulla suoritetaan käytön kannalta olennaisimmat toiminnot. Objektien luominen ja poistaminen sujui koehenkilöiltä hyvin. Tunnistamatta jääneiden luonti- tai poistoeleiden lukumäärä jäi pieneksi. Näiden eleiden varmalla toiminnalla on suuri vaikutus siihen, kuinka sujuvaksi koehenkilöt työskentelyn tunsivat. Toisaalta tasauselettä käytettäessä tuli esille ongelmakohtia. Niin testidatan analyysistä kuin subjektiivisesta palautteesta heijastui kaksi kehitettävää kohtaa: eleen tunnistus-tarkkuuden parantaminen ja tasauseleen jälkeisen objektin valinnan säilyttäminen.

Tasauseleiden ensimmäinen ongelmakohta johtaa siihen, että käyttäjät eivät voi tasata objekteja intuitiivisesti. Heidän tuli useissa tapauksissa ensin siirtää tasattaviksi haluttuja objekteja lähemmäksi toisiaan, mikä heikentää tasauksen toimivuutta. Toinen ongelma liittyi objektin valinnan kadottamiseen tasauseleen jälkeen, jolloin esimerkiksi perättäisten vaaka- ja pystytasausten välillä koehenkilön tuli valita uudelleen toinen tasattavista objekteista. Tasauseleiden käytön lisääntyessä koehenkilöt oppivat paremmin tietämään eleen piirtämiseen liittyvät vaatimukset. Toteutetun tasauseleen ongelmattoman käyttämisen mahdollistaminen on haastava tehtävä, sillä eleellä pystyy tasaamaan objekteja monipuolisesti. Yleensä tasauskomentojen avulla ei ole mahdollista tasata objekteja vastakkaisten sivujen perusteella. Tässä suhteessa eleen toimintavarmuuden kehittäminen on mielenkiintoinen kysymys, johon tämän testauksen avulla saatiin useita hyviä parannusehdotuksia.

Tilanhallintatekniikoiden toimivuudesta testin avulla saatiin mielenkiintoista tietoa. Peilikuvina toteutettujen tehtävien 8-11 suoritusajat olivat keskenään melko yhteneviä. Ensimmäisten tilanhallintatehtävien suoritusaikoja vertailtaessa raahaamista käyttäen tapahtuva vierittäminen oli keskimäärin 11,1 sekuntia yleisnäkymän käyttöä nopeampaa. Jälkimmäisissä tehtävissä taas yleisnäkymän käyttö osoittautui vierittämistä keskimäärin 0,6 sekuntia nopeammaksi. Näiden suoritusaikojen perusteella vierittäminen oli oletusten vastaisesti yleisnäkymän käyttöä nopeampaa. Tilanhallinta-tekniikoiden testiasetelman jatkokehittämistä varten olisi huomioitava eri tekniikoita käyttävien tehtävien tasapainottaminen niin, että koehenkilöiden jo suorittamien tehtävien ja näin käyttökokemuksen määrä olisi ennen kummankin tekniikan testaamista sama.

Tilanhallintatekniikoiden väliltä valittaessa koehenkilöt käyttivät tehtävässä 12 yhtä henkilöä lukuun ottamatta yleisnäkymää pidempien siirtymien tekemiseen. Oletusta yleisnäkymän käytön nopeudesta tukee osaltaan yleisnäkymän avulla suoritettujen komentojen määrän vertaaminen vierittämiseleiden määrään. Yleisnäkymän painalluksia oli tehtävässä keskimäärin 3,6, kun taas raahaamiseleiden määrä vierittämisessä oli 10,9. Tilanhallintatekniikoita koskevien subjektiivisten palautteiden perusteella yleisnäkymä oli hyödyllinen ominaisuus. Kaikista annetuista arvioista yleisnäkymän toimivuuden keskiarvo, 4,8, oli korkein. Tämän testauksen perusteella yleisnäkymä oli kaavionmuokkauksessa hyödylliseksi koettu apuväline, mutta sen mahdolliselle ajalliselle hyödylle vierittämiseen verrattuna ei tuloksista saatu tukea.

Objektien siirtämisen testaamisessa raahaamisen ja click-and-drop-tekniikan välillä raahaaminen osoittautui suoritettujen eleiden perusteella toimivammaksi tekniikaksi. Yhtä raahaamalla siirrettävää objektia kohti käytettiin 1,6 elettä, kun taas click-and-drop-tekniikkaa käytettäessä jouduttiin suorittamaan 2,1 komentoa objektia kohti. Myös subjektiivisten arvioiden perusteella raahaaminen oli toimivampi ja hyödyllisempi tekniikka. Raahaaminen on useille käyttäjille tuttu ja intuitiivinen tekniikka, jonka käyttäminen on helppoa ilman erillistä opettelua. Click-and-drop-tekniikan kanssa vaikeutena oli tuplanapautuksen muistaminen ennen uuden sijoituspaikan osoittamista. Samoin objektin valinnan indikaattori ei ollut paras mahdollinen. Toisaalta objektien vapaata siirtämistä tutkineessa tehtävässä 7 click-and-drop-tekniikalla siirrettiin enemmän objekteja kuin raahaamalla. Tästä voidaan päätellä, että click-and-drop-tekniikka on idealtaan lupaava ja kiehtova. Teknisen toteutuksen ja käytettävyyden parantamisella sen tehokkaampi hyödyntäminen on mahdollista.

Kokonaisuudessaan testaus palveli hyvin ennalta asetettuja tavoitteita. Kaikista testaustavoitteista saatiin tietoa, jonka avulla toteutettujen ominaisuuksien onnistumista ja kehitystarpeita voidaan paremmin arvioida. Uusista ominaisuuksista tasaaminen ja click-and-drop-siirto vaatisivat vielä muokkaamista ja kehittelyä, jotta niiden käytöstä saataisiin tehokasta ja ongelmatonta. Kokonaisuudessaan testatun kaavioeditorisovelluksen tarjoamia vuorovaikutustapoja pidettiin kuitenkin luonnollisina ja helppoina tiedon käsittelemiseen interaktiivisella tussitaululla. Luonnollisen vuorovaikutuksen mahdollistaminen kaavioeditorin käytössä on ollut keskeinen aspekti, johon interaktiivisten tussitaulujen avulla on tässä työssä pyritty.

7. Lopuksi

Tässä tutkielmassa tutkin, kuinka olemassa oleva kaavioeditorisovellus voidaan muokata interaktiivisilla tussitauluilla käytettäväksi. Tarkastelussa olivat erilaiset vuorovaikutustekniikat, joiden avulla tussitaulukäytöstä pyrittiin tekemään mahdollisimman helppoa ja toisaalta myös tehokasta. Toteutin käytännössä useita erilaisia ominaisuuksia, joiden ensisijaisina tarkoituksina oli mahdollistaa komentojen antaminen ja taulun tilan hallinta.

Tutkimuksessa selvisi, että toteutettujen ominaisuuksien avulla kaavioeditorisovelluksen käyttö interaktiivisella tussitaululla oli luonnollista. Sovelluksen testaus ei kattanut kaikkia ominaisuuksia, vaan tavoitteeksi asetettiin eleiden, tilanhallinnan ja objektien siirtämisen tutkiminen. Kaksiosaisessa tutkimusongelmassa pyrittiin helppoon tapaan antaa komentoja interaktiiviselle tussitaululle. Testauksen perusteella eleiden käyttö komentojen antamisessa oli pääsääntöisesti sujuvaa, vaikka erityisesti tasauksen käytössä tuli esille kehitettäviä kohtia. Toinen keskeinen tavoite oli tehdä tussitaulun fyysisen ja virtuaalisen tilan hallinnasta tehokasta. Testaustulosten perusteella yleisnäkymän ja muokkausnäkymän yhtäaikainen käyttö on onnistunut ratkaisu, sillä yleisnäkymää pidettiin hyödyllisenä ja sitä myös käytettiin kaavioissa liikkumiseen.

Pohdittaessa tutkimuksen onnistumista yleisellä tasolla voidaan todeta, että olemassa olevan sovelluksen räätälöiminen interaktiiviselle tussitaululle onnistui hyvin. Olennaista on huomioida interaktiivisten tussitaulujen vuorovaikutukselle asettamat vaatimukset ja sovittaa ne kohdesovelluksen lainalaisuuksiin. Omassa tutkimuksessani tämä näkyi esimerkiksi objektien luonnin ja poistamisen yhdistämisessä eleisiin. Ensimmäisessä vaiheessa tulee selvittää vuorovaikutustavat, joilla alkuperäisen sovelluksen tärkeimpien ominaisuuksien käyttö on mahdollista uudessakin ympäristössä. Tämän jälkeen on mahdollista kehittää uusia tapoja, joilla työskentelyä voidaan tehostaa edelleen hyödyntämällä erilaisen käyttöalustan luonteenpiirteitä. Esimerkkinä tästä on kiihdytyksen käyttö näkymän vierittämisessä, joka ei työpöytäympäristössä tapahtuvassa kaavionkäsittelyssä erilaisesta käyttöympäristöstä johtuen tarjoa vastaavia mahdollisuuksia käytön tehostamiseen.

Esiteltyyn tutkimukseen sisältyi merkittävästi tussitauluympäristöön suunnattujen vuorovaikutustekniikoiden suunnittelua ja toteuttamista myös käytännössä. Erillisiä kokonaisuuksia olivat kontekstivalikot, eleiden käyttö, tekstinsyöttö vapaalla kädellä, työalan jakaminen, skaalautuvuuden käyttö ja kiihtyvyyden hyödyntäminen. Jokaisen aihealueen tarkempi tutkiminen olisi mahdollista, mikä tarjoaa jatkotutkimukselle oivia vaihtoehtoja.

Mahdolliset jatkotutkimushankkeet voidaan jaotella kahteen kokonaisuuteen. Ensimmäisessä vaihtoehdossa käytettäisiin pitkälti nykyistä kaavioeditorisovellusta, jonka avulla suoritettaisiin kattavampia, yhteen aihekokonaisuuteen keskittyviä

testauksia. Esimerkiksi eri tilanhallintatekniikoiden eli yleisnäkymän käytön, tavallisen vierittämisen ja kiihdytystä käyttävän vierittämisen vertailusta saisi luotua mielenkiintoisen koeasetelman. Laajassa testauksessa tasapainotettaisiin eri tekniikoiden käyttö koehenkilöiden välillä, jotta sovelluksen käyttökokemus olisi kokonaisuudessaan sama ennen jokaisen tekniikan testausta.

Samoin olisi mahdollista tutkia objektien siirtämiseen käytettäviä tekniikoita niin, että siirrettävät matkat olisivat huomattavan pitkiä. Tällöin click-and-drop-tekniikan ja raahaamisen välille voisi muodostua selviä eroja, sillä ensin mainitun tekniikan avulla objekteja siirrettäessä voidaan käyttää hyödyksi yleisnäkymää. Objektin tuplanapautuksen jälkeen käyttäjä voi valita yleisnäkymästä kohdealueen, jonne valittu objekti siirretään. Mahdollisissa jatkotestauksissa testitehtävien ja koehenkilöiden määrän tulisi olla riittävän suuri, jotta tulosten analysoinnista olisi mahdollista saada esille selviä tuloksia tai trendejä.

Toisessa jatkotutkimusvaihtoehdossa keskityttäisiin kehittämään sovellusta toteutusvaiheessa ilmi tulleiden ja koehenkilöiltä saatujen ideoiden pohjalta. Esimerkiksi tasauseleen tutkimista olisi mahdollista syventää. Tasauskomennossa voitaisiin ottaa huomioon objektien koot ja keskinäiset suhteet, joiden perusteella olisi mahdollista käytön aikana mukauttaa etäisyyttä, jonka sisällä tasattaviksi haluttujen objektien tulee olla. Lisäksi nykyisen tasauseleen rinnalle olisi mahdollista kehittää vaihtoehtoisia tekniikoita, kuten tasattaviksi haluttujen objektien eksplisiittistä osoittamista.

Myös virtuaalisen tilanhallinnan konstruktiiviselle jatkotutkimukselle on olemassa monia mahdollisuuksia. Olisi esimerkiksi mielenkiintoista vertailla, kuinka käyttäjät hyödyntäisivät zoomausta, jos he saisivat itse kontrolloida näkymän tarkkuustasoa. Näin voitaisiin tutkia, kuinka vapaampi zoomaaminen suhteutuisi nykyiseen yleisnäkymän automaattiseen skaalaamiseen.

Tässä tutkielmassa on tehty läpileikkaus tekniikoista, jotka ovat tutkimisen arvoisia kehitettäessä kaavioeditorisovelluksia interaktiivisia tussitauluja varten. Suoritetun testauksen avulla saatiin lupaavia, joskin jatkotutkimusta vaativia, tuloksia siitä, että automaattista skaalautuvuutta, tasauseleitä ja click-and-drop-tekniikkaa voidaan hyödyntää interaktiivisille tussitauluille tarkoitetuissa kaavioeditoreissa.

Viiteluettelo

[Albinsson and Zhai, 2003] Pär-Anders Albinsson and Shumin Zhai, High precision touch screen interaction. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2003, ACM Press, 105-112.

[Bardohl, 1997] Roswitha Bardohl, Application of Graph Transformation to Visual Languages State of the Art and Further Ideas. Technical Report, March, 1997. Available as *http://www.cs.pitt.edu/~chang/231/UML/tr97-10.pdf*. (Tarkastettu 13.5.2007)

[Baudisch *et al.,* 2001] Patrick Baudisch, Nathaniel Good, and Paul Stewart, Focus plus context screens: combining display technology with visualization techniques. In: *Proceedings of the 14th annual ACM symposium on User interface software and technology,* 2001, ACM Press, 31-40.

[Baudisch *et al.,* 2002] Patrick Baudisch, Nathaniel Good, Victoria Bellotti, and Pamela Schraedley, Keeping things in context: a comparative evaluation of focus plus context screens, overviews, and zooming. In: *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, 2002, ACM Press, 259-266.

[Baudisch and Tandler, 2002] Patrick Baudisch and Peter Tandler, ContextWall: A multi-user workbench for independent work, peer-to-peer interaction, and whiteboard collaboration. Unpublished manuscript, 2002. Available as *http://www.patrickbaudisch.com/publications/2002-Baudisch-CSCW02-ContextWall.pdf*. (Tarkastettu 13.5.2007)

[Beauchamp and Parkinson, 2005] Gary Beauchamp and John Parkinson, Beyond the 'wow' factor: developing interactivity with the interactive whiteboard. *School Science Review*, **86** (316), 2005, 97-103.

[BECTA, 2003] What the research says about interactive whiteboards. Available as *http://www.becta.org.uk/page_documents/research/wtrs_whiteboards.pdf*. (Tarkastettu 13.5.2007)

[Bederson and Hollan, 1994] Benjamin B. Pederson and James D. Hollan, Pad++: a zooming graphical interface for exploring alternate interface physics. In: *Proceedings of the 7th annual ACM symposium on User interface software and technology*, 1994, ACM Press, 17-26.

[Benko *et al.,* 2006] Hrvoje Benko, Andrew D. Wilson, and Patrick Baudisch, Precise selection techniques for multi-touch screens. In: *Proceedings of the SIGCHI conference on Human Factors in computing systems*, 2006, ACM Press, 1263-1272.

[Berque *et al.,* 2001] Dave Berque, David K. Johnson, and Larry Jovanovic, Teaching theory of computation using pen-based computers and an electronic whiteboard. *ACM SIGCSE Bulletin*, **3** (33), 2001, ACM Press, 169-172.

[Büring *et al.,* 2006] Thorsten Büring, Jens Gerken, and Harald Reiterer, Usability of overview-supported zooming on small screens with regard to individual differences in spatial ability. In: *Proceedings of the working conference on Advanced visual interfaces,* 2006, ACM Press, 233-240.

[Böhringer and Newbery Paulisch, 1990] Karl-Friedrich Böhringer and Frances Newbery Paulisch, Using constraints to achieve stability in automatic graph layout algorithms. In: *Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people*, 1990, ACM Press, 43-51.

[Chen *et al.,* 2003] Qi Chen, John Grundy, and John Hosking, An e-whiteboard application to support early design-stage sketching of UML diagrams. In: *Proceedings of 2003 IEEE Human-Centric Computing Conference*, 2003, IEEE Computer Society Press, 219-226.

[Chok and Marriott, 1998] Sitt Sen Chok and Kim Marriott, Automatic construction of intelligent diagram editors. In: *Proceedings of the 11th annual ACM symposium on User interface software and technology*, 1998, ACM Press, 185-194.

[Chong and Sakauchi, 2000] Ng S. T. Chong and Masao Sakauchi, Back to the basics: a first class chalkboard and more. In: *Proceedings of the 2000 ACM symposium on Applied computing*, **1**, 2000, ACM Press, 131-136.

[Citrin and McWhirter, 1995] Wayne Citrin and Jeffrey McWhirter, Diagram entry mechanisms in graphical environments. In: *Conference companion on Human factors in computing systems*, 1995, ACM Press, 294-295.

[Collomb *et al.,* 2005] Maxime Collomb, Mountaz Hascoët, Patrick Baudisch, and Brian Lee, Improving drag-and-drop on wall-size displays. In: *Proceedings of the 2005 conference on Graphics interface*, 2005, Canadian Human-Computer Communications Society, 25-32.

[Cox *et al.,* 1998] Donald A. Cox, Jasdeep S. Chugh, Carl Gutwin, and Saul Greenberg, The usability of transparent overview layers. In: *CHI 98 conference summary on Human factors in computing systems*, 1998, ACM Press, 301-302.

[Damm *et al.,* 2000] Christian H. Damm, Klaus M. Hansen, Michael Thomsen, and Michael Tyrsted, Creative object-oriented modelling: support for intuition, flexibility, and collaboration in CASE tools. In: *Proceedings of the 14th European Conference on Object-Oriented Programming*, 2000, Springer-Verlag, 27-43.

[Di Battista *et al.,* 1999] Giuseppe Di Battista, Peter Eades, Roberto Tamassia, and Ioannis G. Tollis, *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice-Hall Inc., Upper Saddle River, New Jersey, 1999.

[eBeam, 2007] Luidia, eBeam-tussitaululaitteen kotisivut, http://www.e-beam.com/. (Tarkastettu 13.5.2007)

[Elrod *et al.,* 1992] Scott Elrod, Richard Bruce, Rich Gold, David Goldberg, Frank Halasz, William Janssen, David Lee, Kim McCall, Elin Pedersen, Ken Pier, John Tang, and Brent Welch, Liveboard: a large interactive display supporting group meetings, presentations, and remote collaboration. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1992, ACM Press, 599-607.

[Forsberg *et al.*, 1998] Andrew Forsberg, Mark Dietrich, and Robert Zeleznik, The music notepad. In: *Proceedings of the 11th annual ACM symposium on User interface software and technology,* 1998, ACM Press, 203-210.

[Friedland *et al.,* 2004] Gerald Friedland, Lars Knipping, Raúl Rojas, and Ernesto Tapia, Teaching with an intelligent electronic chalkboard. In: *Proceedings of the 2004 ACM SIGMM workshop on Effective telepresence*, 2004, ACM Press, 16-23.

[Furnas and Zhang, 1998] George W. Furnas and Xiaolong Zhang, MuSE: a multiscale editor. In: *Proceedings of the 11th annual ACM symposium on User interface software and technology,* 1998, ACM Press, 107-116.

[Geißler, 1998] Jörg Geißler, Shuffle, throw or take it! working efficiently with an interactive wall. In: *CHI 98 conference summary on Human factors in computing systems*, 1998, ACM Press, 265-266.

[Goldberg and Richardson, 1993] David Goldberg and Cate Richardson, Touch-typing with a stylus. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1993, ACM Press, 80-87.

[Guimbretière *et al.,* 2001] François Guimbretière, Maureen Stone, and Terry Winograd, Fluid interaction with high-resolution wall-size displays. In: *Proceedings of the 14th annual ACM symposium on User interface software and technology*, 2001, ACM Press, 21-30.

[Hornbæk *et al.,* 2002] Kasper Hornbæk, Benjamin B. Bederson, and Catherine Plaisant, Navigation patterns and usability of zoomable user interfaces with and without an overview. *ACM Transactions on Computer-Human Interaction (TOCHI)*, **9** (4), 2002, ACM Press, 362-389.

[Hutchins *et al.,* 1986] Edwin L. Hutchins, James D. Hollan, and Donald A. Norman, Direct manipulation interfaces. In: D.A. Norman and S.W. Draper (eds.), *User Centered System Design*. Lawrence Erlbaum, 1986, 87-124.

[Igarashi *et al.,* 1997] Takeo Igarashi, Satoshi Matsuoka, Sachiko Kawachiya, and Hidehiko Tanaka, Interactive beautification: a technique for rapid geometric design. In: *Proceedings of the 10th annual ACM symposium on User interface software and technology*, 1997, ACM Press, 105-114.

[Igarashi *et al.,* 1999a] Takeo Igarashi, Elizabeth D. Mynatt, W. Keith Edwards, and Anthony LaMarca, Demonstrating flatland user interfaces. *CHI '99 extended abstracts on Human factors in computing systems*, 1999, ACM Press, 27-28.

[Igarashi *et al.,* 1999b] Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka, Teddy: a sketching interface for 3D freeform design. In: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 1999, ACM Press/Addison-Wesley Publishing Co., 409-416.

[Igarashi *et al.,* 2000] Takeo Igarashi, W. Keith Edwards, Anthony LaMarca, and Elizabeth D. Mynatt, An architecture for pen-based interaction on electronic whiteboards. In: *Proceedings of the working conference on Advanced visual interfaces*, 2000, ACM Press, 68-75.

[Katz, 1948] David Katz, Hahmopsykologia. Suom. Martti Takala. Otava, Helsinki, Kauppalehti Oy:n Kirjapaino, 1948.

[Kramer, 1994] Axel Kramer, Translucent patches - dissolving windows. In: *Proceedings of the 7th annual ACM symposium on User interface software and technology*, 1994, ACM Press, 121-130.

[Krueger, 1983] Myron W. Krueger, *Artificial Reality.* Addison-Wesley, 1983.

[Krueger *et al.,* 1985] Myron W. Krueger, Thomas Gionfriddo, and Katrin Hinrichsen, VIDEOPLACE - an artificial reality. In: *Human Factors in Computing Systems, CHI '85 Conference Proceedings*, ACM Press, 1985, 35-40.

[Lank *et al.,* 2000] Edward Lank, Jeb S. Thorley, and Sean Jy-Shyang Chen, An interactive system for recognizing hand drawn UML diagrams. In: *Proceedings of the 2000 conference of the Centre for Advanced Studies on Collaborative research*, 2000, IBM Press, 7.

[Levy, 2002] Philippa Levy, Interactive whiteboards in learning and teaching in two Sheffield schools: a developmental study. Sheffield Excellence in Cities Partnership. Available as *http://dis.shef.ac.uk/eirg/projects/wboards.htm.* (Tarkastettu 13.5.2007)

[Liu *et al.,* 2003] Ray Liu, Lisa Wong, and John Grundy, Experiences Developing an E-whiteboard-based Circuit Designer. In: *Proceedings of the 2003 International Conference on Software Engineering Research and Practice,* 2003, CSREA Press, 739-744.

[Masui, 1994] Toshiyuki Masui, Evolutionary learning of graph layout constraints from examples. In: *Proceedings of the ACM UIST ´94 Symposium on User Interface Software and Technology*, ACM Press, 1994, 103-108.

[Minas, 1997] Mark Minas, Diagram editing with hypergraph parser support. In: *Proceedings of the 1997 IEEE Symposium on Visual Languages (VL '97)*, 1997, IEEE Computer Society, 226.

[Minas, 2002] Mark Minas, Concepts and realization of a diagram editor generator based on hypergraph transformation. *Science of Computer Programming*, **44** (2), 2002, Elsevier North-Holland, Inc., 157-180.

[Minas, 2006] Mark Minas, Syntax analysis for diagram editors: a constraint satisfaction problem. In: *Proceedings of the working conference on Advanced visual interfaces*, 2006, ACM Press, 167-170.

[Minsky, 1984] Margaret R. Minsky, Manipulating simulated objects with real-world gestures using a force and position sensitive screen. In: *Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, 1984, ACM Press, 195-203.

[MKConnect, 2007] Milton Keynes Council, Choosing an Interactive Whiteboard, http://m99.e2bn.net/e2bn/leas/m99/frontpage/new/. (Tarkastettu 13.5.2007)

[Moran *et al.,* 1997] Thomas P. Moran, Patrick Chiu, and William van Melle, Pen-based interaction techniques for organizing material on an electronic whiteboard. In: *Proceedings of the 10th annual ACM symposium on User interface software and technology*, 1997, ACM Press, 45-54.

[Moran *et al.,* 1998a] Thomas P. Moran, William van Melle, and Patrick Chiu, Spatial interpretation of domain objects integrated into a freeform electronic whiteboard. In: *Proceedings of the 11th annual ACM symposium on User interface software and technology*, 1998, ACM Press, 175-184.

[Moran *et al.,* 1998b] Thomas P. Moran, William van Melle, and Patrick Chiu, Tailorable domain objects as meeting tools for an electronic whiteboard. In: *Proceedings of the 1998 ACM conference on Computer supported cooperative work*, 1998, ACM Press, 295-304.

[Mynatt, 1999] Elizabeth D. Mynatt, "The Writing on the Wall". *Proceedings of INTERACT '99*, 1999, IOS Press, 196-204.

[Mynatt *et al.,* 1999] Elizabeth D. Mynatt, Takeo Igarashi, W. Keith Edwards, and Anthony LaMarca, Flatland: new dimensions in office whiteboards. In: *Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit*, 1999, ACM Press, 346-353.

[Mynatt *et al.,* 2000] Elizabeth D. Mynatt, Takeo Igarashi, W. Keith Edwards, and Anthony LaMarca, Designing an augmented writing surface. *IEEE Computer Graphics and Applications*, **20** (4), 2000, IEEE Computer Society Press, 55-61.

[Nakagawa *et al.,* 1999] Masaki Nakagawa, Koichiro Hotta, Hirokazu Bandou, Tsuyoshi Oguni, Naoki Kato, and Shin-ichi Sawada, A revised human interface and educational applications on ideaboard. In: *Conference on Human Factors in Computing Systems*, 1999, ACM Press, 15-16.

[Nam, 2005] Tek-Jin Nam, Sketch-based rapid prototyping platform for hardware-software integrated interactive products. *CHI '05 extended abstracts on Human factors in computing systems*, 2005, ACM Press, 1689-1692.

[NextWindow, 2007] NextWindow, Comparison of touchscreen technologies, http://www.nextwindow.com/benefits/comparative.html. (Tarkastettu 13.5.2007)

[Otsuki *et al.,* 2001] Yoko Otsuki, Hirokazu Bandoh, Naoki Kato, and Masaki Nakagawa, Development of an interactive electronic whiteboard system with multiple electronic pens for educational applications. In: *Proceedings of ICCE/SchoolNet 2001,* **3**, 2001, 1438-1445.

[Ovaska *et al.,* 2003] Saila Ovaska, Pentti Hietala, and Marjatta Kangassalo, Electronic whiteboard in kindergarten: opportunities and requirements. In: *Proceeding of the 2003 conference on Interaction design and children*, 2003, ACM Press, 15-22.

[Pedersen *et al.,* 1993] Elin Rønby Pedersen, Kim McCall, Thomas P. Moran, and Frank G. Halasz, Tivoli: an electronic whiteboard for informal workgroup meetings. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1993, ACM Press, 391-398.

[Perlin and Fox, 1993] Ken Perlin and David Fox, Pad: an alternative approach to the computer interface. In: *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, 1993, ACM Press, 57-64.

[PhatWare, 2007] PhatWare, Calligrapher-tekstintunnistusohjelman kotisivut, http://www.phatware.com/calligrapher/index.html. (Tarkastettu 13.5.2007)

[Plimmer and Apperley, 2002] Beryl Plimmer and Mark Apperley, Computer-aided sketching to capture preliminary design. In: *Proceedings of the Third Australian conference on User Interfaces,* 2002, Australian Computer Society Inc., 9-12.

[Plumlee and Ware, 2006] Matthew D. Plumlee and Colin Ware, Zooming versus multiple window interfaces: cognitive costs of visual comparisons. *ACM Transactions on Computer-Human Interaction (TOCHI),* **13** (2), 2006, ACM Press, 179-209.

[Pook *et al.,* 2000] Stuart Pook, Eric Lecolinet, Guy Vaysseix, and Emmanuel Barillot, Context and interaction in zoomable user interfaces. In: *Proceedings of the working conference on Advanced visual interfaces*, 2000, ACM Press, 227-231.

[Presentation Technologies Overview, 2007] Presentation Technologies Overview, Interactive Whiteboards - an Overview, http://www.presentationtek.com/?p=21. (Tarkastettu 13.5.2007)

[Raisamo, 1995] Roope Raisamo, Kaksi kättä käyttöliittymässä. Tampereen yliopisto, Tietojenkäsittelyopin laitos, Pro gradu -tutkielma, toukokuu 1995.

[Raisamo, 1998] Roope Raisamo, A Multimodal user interface for public information kiosks. In: *Proceedings of 1998 Workshop on Perceptual User Interfaces (PUI '98),* Matthew Turk (ed.), San Francisco, November 1998, 7-12.

[Raisamo, 2002] Jukka Raisamo, Tuntopalautteen tukemat suorakäyttöiset kaavionmuokkaustyökalut. Tampereen yliopisto, Tietojenkäsittelyopin laitos, Pro gradu -tutkielma, kesäkuu 2002.

[Raisamo and Räihä, 1996] Roope Raisamo and Kari-Jouko Räihä, A new direct manipulation technique for aligning objects in drawing programs. In: *Proceedings of ACM UIST´96, Symposium on User Interface Software and Technology*, 1996, ACM Press, 157-164.

[Rakkolainen *et al.,* 2006] Ismo Rakkolainen, Tanju Erdem, Çiğdem Erdem, Mehmet Özkan, and Markku Laitinen, Interactive "immaterial" screen for performing arts. In: *Proceedings of the 14th annual ACM international conference on Multimedia*, 2006, ACM Press, 185-188.

[Read and Marlin, 1996] Michael Read and Chris Marlin, Generating direct manipulation program editors within the multiview programming environment. *Workshop on Multiple Perspectives in Software Development,* 1996, ACM Press, 232-236.

[Rekimoto, 1998] Jun Rekimoto, A multiple device approach for supporting whiteboard-based interactions. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1998, ACM Press/Addison-Wesley Publishing Co., 344-351.

[Ringel *et al.,* 2001] Meredith Ringel, Henry Berg, Yuhui Jin, and Terry Winograd, Barehands: implement-free interaction with a wall-mounted display. *CHI '01 extended abstracts on Human factors in computing systems*, 2001, ACM Press, 367-368.

[Rubine, 1991] Dean Rubine, Specifying gestures by example. In: *Proceedings of the 18th annual conference on Computer graphics and interactive techniques*, 1991, ACM Press, 329-337.

[Sanford, 2007] Sanford Brands, Mimio-tussitaululaitteen kotisivut, http://www.mimio.com/. (Tarkastettu 13.5.2007)

[Shneiderman, 1982] Ben Shneiderman. The future of interactive systems and the emergency of direct manipulation. *Behaviour and Information Technology* **1** (1982), 237-256.

[Shneiderman, 1983] Ben Shneiderman. Direct manipulation: a step beyond programming languages. In: Ronald M. Baecker and William Buxton (eds.), *Readings in Human-Computer Interaction: A Multidisciplinary Approach*. Morgan Kaufmann Publishers Inc., 1987, 461-467.

[Smarttech, 2007] SMART Technologies, SMART Board -tussitaululaitteen kotisivut, http://www.smarttech.com/. (Tarkastettu 13.5.2007)

[Summet *et al.*, 2005] Jay Summet, Gregory D. Abowd, Gregory M. Corso, and James M. Rehg, Virtual rear projection: do shadows matter? *CHI '05 extended abstracts on Human factors in computing systems*, 2005, ACM Press, 1997-2000.

[Virtual Learning, 2002] Interactive Whiteboards: new tools, new pedagogies, new learning? Some views from practitioners. Available as *http://www.virtuallearning.org.uk/whiteboards/*. (Tarkastettu 13.5.2007)

[Vlissides and Linton, 1990] John M. Vlissides and Mark A. Linton, Unidraw: a framework for building domain-specific graphical editors. *ACM Transactions on Information Systems (TOIS)*, **8** (3), 1990, ACM Press, 237-268.

[Wellner, 1991] Pierre Wellner, The digitaldesk calculator: tactile manipulation on a desk top display. In: *Proceedings of the 4th annual ACM symposium on User interface software and technology*, 1991, ACM Press, 27-33.

[Ziegler and Fähnrich, 1988] Jürgen E. Ziegler and Klaus-Peter Fähnrich, Direct manipulation. In: M. Helander (ed.), *Handbook of Human-Computer Interaction*, Elsevier Science Publishers B.V., 1988, 123-133.

Liite 1

Taustatietolomake

Nimi: _____

Ikä: _____

Sukupuoli:

- Mies
- Nainen

Montako vuotta olet käyttänyt tietokoneita?

- Alle vuoden
- 1 - 5 vuotta
- 5 - 10 vuotta
- Yli kymmenen vuotta

Kuinka usein käytät tietokoneita?

- Päivittäin
- Muutaman kerran viikossa
- Muutaman kerran kuukaudessa
- Harvemmin

Interaktiivisten tussitaulujen käyttö

Oletko aikaisemmin käyttänyt interaktiivisia tussitauluja?

- Kyllä
- En

Jos olet aikaisemmin käyttänyt interaktiivisia tussitauluja, niin kuinka usein?

- Päivittäin
- Muutaman kerran viikossa
- Muutaman kerran kuukaudessa
- Harvemmin

Jos olet aikaisemmin käyttänyt interaktiivisia tussitauluja, niin minkälaisia laitteita ja mihin tarkoitukseen?

_____

_____

_____

Graafisten editorisovellusten käyttö

Oletko aikaisemmin käyttänyt graafisia editorisovelluksia, kuten piirto-ohjelmia tai kaavioeditoreja?

- Kyllä
- En

Jos olet aikaisemmin käyttänyt graafisia editorisovelluksia, niin kuinka usein?

- Päivittäin
- Muutaman kerran viikossa
- Muutaman kerran kuukaudessa
- Harvemmin

Jos olet aikaisemmin käyttänyt graafisia editorisovelluksia, niin mitä sovelluksia ja mihin tarkoitukseen?

_____

_____

_____

Kynän tai sormen käyttö syötteiden antamisessa

Oletko aikaisemmin käyttänyt kynää tai sormea syötteiden antamiseen, kuten esimerkiksi PDA-laitteissa tai älypuhelimissa?

- Kyllä
- En

Jos olet aikaisemmin käyttänyt kynää tai sormea syötteiden antamiseen, niin kuinka usein?

- Päivittäin
- Muutaman kerran viikossa
- Muutaman kerran kuukaudessa
- Harvemmin

Jos olet aikaisemmin käyttänyt kynää tai sormea syötteiden antamiseen, niin millä laitteilla ja mihin syötteitä on käytetty?

_____

_____

_____

Lupa testisuorituksen tallentamiseen

Testin aikana sovellus tallentaa tietoja koehenkilön suorittamista komennoista ja objektien liikuttamisesta. Tätä tietoa käytetään testauksen analyysivaiheessa, jossa arvioidaan kaavionkäsittelytekniikoiden käyttöä. Koehenkilöiden antamia taustatietoja ja testistä tallennettuja tietoja ei yhdistetä toisiinsa. Koehenkilöiden nimet eivät tule esille testitulosten yhteydessä, vaan ne ovat ainoastaan testin järjestäjän (Jussi Rantala) tiedossa.

☐ Olen tietoinen tallennetun tiedon käyttötarkoituksesta ja annan luvan tallentamiseen

Liite 2

Arviointilomake

Valitse mielestäsi paras vaihtoehto ympyröimällä yksi viidestä numerosta.

1. Arvioi eleiden toimivuutta objektien luomisessa, poistamisessa ja tasaamisessa:

   (vaikeakäyttöinen) 1 2 3 4 5 (helppokäyttöinen)

2. Arvioi yleisnäkymän käytön toimivuutta navigoinnissa:

   (vaikeakäyttöinen) 1 2 3 4 5 (helppokäyttöinen)

3. Arvioi yleisnäkymän hyödyllisyyttä navigoinnissa:

   (ei hyödyllinen) 1 2 3 4 5 (hyödyllinen)

4. Arvioi muokkausnäkymän raahaamisen toimivuutta:

   (vaikeakäyttöinen) 1 2 3 4 5 (helppokäyttöinen)

5. Arvioi muokkausnäkymän raahaamisen hyödyllisyyttä:

   (ei hyödyllinen) 1 2 3 4 5 (hyödyllinen)

6. Arvioi raahaamalla tapahtuvan objektin siirron toimivuutta:

   (vaikeakäyttöinen) 1 2 3 4 5 (helppokäyttöinen)

7. Arvioi raahaamalla tapahtuvan objektin siirron miellyttävyyttä:

   (ei miellyttävä) 1 2 3 4 5 (miellyttävä)

8. Arvioi click-and-drop-tekniikalla tapahtuvan objektien siirron toimivuutta:

   (vaikeakäyttöinen) 1 2 3 4 5 (helppokäyttöinen)

9. Arvioi click-and-drop-tekniikalla tapahtuvan objektien siirron miellyttävyyttä:

(ei miellyttävä) 1 2 3 4 5 (miellyttävä)

10. Voit kirjoittaa vapaasti sovellukseen tai tehtäviin liittyviä kommentteja ja parannusehdotuksia:

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____