



Fakulteten för ekonomi, kommunikation och IT
Avdelningen för Datavetenskap

Mathias Andersson och Henrik Bäck

Flerpunktspekskärm med användargränssnitt för multipla operationer

Multi-touch screen with interface
for multiple operations

Examensarbete 30 poäng
Civilingenjörsprogrammet IT

Datum/Termin: XX-XX-XX
Handledare: Katarina Asplund
Examinator: Donald. F. Ross
Ev. löpnummer: X-XX XX XX

Denna uppsats är skriven som en del av det arbete som krävs för att erhålla en civilingenjörsexamen i datavetenskap. Allt material i denna rapport, vilket inte är mitt eget, har blivit tydligt identifierat och inget material är inkluderat som tidigare använts för erhållande av annan examen.

Mathias Andersson

Henrik Bäck

Godkänd, Date of defense

Opponent: NN

Handledare: Katarina Asplund

Examinator: Donald F. Ross

Sammanfattning

Svenskt abstrakt här

Multi-touch screen with interface for multiple operations

Put the text of your english abstract here

Tack

Ett stort tack till Magnus Andersson för tillverkningen av det chassi som all hårdvara monterats i.

Innehåll

1	Inledning	1
1.1	Problem	1
1.2	Motivering	2
1.3	Mål	2
1.4	Genomförande	2
1.5	Primära resultat	2
1.6	Denna uppsats disposition	2
1.7	Sammanfattning	3
2	Bakgrund	5
2.1	Introduktion till flerpunktspekaskärmar	5
2.1.1	Simultan interaktion	5
2.1.2	Tillämpningar	6
2.2	Historia	6
2.2.1	En-punktsinmatning	6
2.2.2	Fler-punktsinmatning	9
2.3	Pekskärmar och flerpunktspekaskärmar idag	13
2.4	Sammanfattning	14
3	Potentiella lösningar	15

3.1	Hårdvara	15
3.1.1	Bildåtergivning	15
3.1.2	Pekpunktsregistrering	16
3.1.3	Chassi	19
3.2	Mjukvara	19
3.2.1	Spårare	19
3.2.2	Klient	19
3.2.3	Spårare och klientkommunikation	20
3.3	Begränsningar och krav	20
3.3.1	Hårdvara	20
3.3.2	Mjukvara	20
4	Hårdvaruprototyp	21
4.1	Beräkningar	21
4.1.1	Kamerans synvinkel	21
4.1.2	Chassits höjd	21
4.1.3	Spegelns storlek	21
4.2	Problem	21
4.2.1	DI	21
4.2.2	Lysa ovanifrån	21
4.2.3	Penna	21
4.3	Implementation	22
4.3.1	Bildåtergivning	22
4.3.2	Bildinläsning	22
4.3.3	Chassi	23
4.3.4	Bildinhämtning	23
4.3.5	Pekning	23
4.4	Komponentlista	24

5	Mjukvaruprototyp	25
5.1	Spårare	25
5.1.1	Funktionalitet	25
5.1.2	Olika spårare	26
5.2	Klient	27
5.2.1	Användargränssnitt	27
5.2.2	Gester	27
5.3	Kommunikation mellan spårare och klient	27
5.3.1	TUIO	27
5.3.2	Ramverk	28
5.3.3	TUIO i mjukvaruprototyp	29
6	Utvärdering av prototyp	31
7	Slutsats	33
	Litteraturförteckning	35

Figurer

2.1	Historisk tidslinje för inmatningssystem	7
2.2	The Rand Tablet	8
2.3	AccuTouch™ 1977 (License pending)	9
2.4	PLATO 1978	10
2.5	reactTable*	11
2.6	iPhone 3G	12
2.7	Microsoft Surface	13
3.1	Utspridd upplysning	17
3.2	Frustrerad total intern reflektion	18
3.3	Lysdiodspenna	19
4.1	Plattksärmens panel	22
4.2	Plattskärm monterad i chassi. På bilden syns undersidan av skärmen.	23
5.1	TUIO, Meddelandetyper	28
5.2	TUIO, Profiler	29

Tabeller

5.1	Olika spårare som implementerar TUIO	26
5.2	TUIO, Meddelandeargument	29

Kapitel 1

Inledning

Den här uppsatsen kommer att visa hur man kan lösa problemet med att flera användare samtidigt kan interagera med ett system via en och samma in-dataenhet. Som in-dataenhet kommer en flerpunktspeksskärm¹ att användas. Flerpunktspeksskärmen kommer att konstrueras från standardkomponenter. En applikation kommer att utvecklas för att demonstrera delar av den funktionalitet som flerpunktspeksskärmen tillhandahåller.

I det här kapitlet ...

1.1 Problem

dyrt

försäljning???

En utmaning inom försäljning är att sälja icke fysiska produkter, så som exempelvis resor. Att kunna presentera icke fysiska produkter på ett interaktivt sätt ger säljaren möjlighet att anpassa presentation och dessutom låta kunden vara delaktig i anpassningen. En nuvarande teknik är att använda sig av en dator för demonstration av produkterna. En sådan dator skulle kunna vara utrustad med en peksskärm för att ge mer direkt manipulation av objekt än vad som är möjligt med tangentbord och mus [24]. Peksskärmen kan enbart ta emot indata från en samtidig användare och dessutom kan inte inmatningen innehålla mer än en in-datapunkt (koordinat).

För att lösa problemet med simultan interaktion från ett flertal användare kan peksskärmen

¹Engelska: Multi-Touch Screen, ej officiell översättning men korrekt enligt Svenska Datatermsgruppen [4].

utvecklas. Istället för att skärmen enbart kan ta emot en in-datapunkt (koordinat) måste den ta emot flera. En sådan förändring skulle medföra nya möjligheter för hur inmatning kan ske och vilka operationer som kan utföras [21].

En säljare som exempelvis visar bilder på produkter får då möjlighet att förstora, förminska samt rotera bilderna med fingrarna på skärmen, utan att använda speciella skalnings- eller roteringsverktyg. Säljaren kan till och med ha en 3D-modell av den icke fysiska varan som denne eller kunden kan navigera på genom att enbart använda fingrarna på ytan.

1.2 Motivering

Varför blir detta bra? Bättre än ngn annan? Likvärdigt? Varför görs jobbet? Dyrt med andra?

1.3 Mål

Målet med den här uppsatsen är lösa problemet med att flera användare simultant kan interagera med ett och sammas system via samma in-data enhet. Målet består av två delar, en hårdvaru- och en mjukvarudel. allmänt om vad hårdvaran och mjukvaran är

1.4 Genomförande

Låda-Skärm-Mjukvara

1.5 Primära resultat

1.6 Denna uppsats disposition

Bakgrund kap2

Lösningar kap3

Mjukvara4 Hårdvara5

Utvärdering6 Slutsats7

1.7 Sammanfattning

Kapitel 2

Bakgrund

Flerpunktspekaskärmar är ingen ny teknik men det är först nu som de har kommit till användning i olika applikationer [25]. Mobiltelefoner [9], datorer med mera utrustas idag med flerpunktspekaskärmar av olika slag och användningsområdet varierar. I det här kapitlet kommer en introduktion till begreppet flerpunktspekaskärmar att ges. Utöver detta kommer en kort historik om olika inmatningsmetoder, relaterade till pekaskärmar eller flerpunktspekaskärmar, att ges.

2.1 Introduktion till flerpunktspekaskärmar

Till skillnad från pekaskärmar har flerpunktspekaskärmar, som tidigare nämnt, möjlighet att ta emot flera simultana pekpunkter. Det är även så att pekaskärmar har egenskapen att de operationer en användare utför på ett objekt, med hjälp av pekaskärmen, är direkt manipulation av objektet. För en ovan användare blir det då lättare för dennes hjärna att koordinera händerna och synintrycket vid användande av pekaskärmar än vid användande av tangentbord och mus. [24] [25]

2.1.1 Simultan interaktion

Möjligheten att kunna använda två händer när ett problem utforskas är naturligt. Det går därför fortare att utföra en viss åtgärd med två händer än med en [21]. Här måste mjukvaran utvecklas eftersom

Två personer samtidigt vill manipulera objekt i ett system

2.1.2 Tillämpningar

Flerpunktspekskärmar kan ha många olika användningsområden.

De lämpar sig exempelvis bra som interaktiva whiteboards då de har möjlighet att ta emot inmatningar från flera samtidiga användare. Detta gör att flera personer samtidigt kan använda tavlan. Till skillnad från en vanlig whiteboard får man dessutom möjlighet att kunna lagra den information som matats in och på så sätt göra informationen på tavlan bestående.

Från whiteboard är steget inte alls långt till anslagstavlor. Det skulle kunna gå att tillverka stora anslagstavlor där flera personer samtidigt kan läsa och sätta upp digitala anslag. Anslagstavlor skulle kunna användas på företag, skolor eller i köpcentrum.

I presentationssammanhang finns flera andra tänkbara användningsområden. I nyhetssändningar analyseras olika scenarion i direktända program. Exempelvis skulle en väderkarta inte behöva vara animerad i förväg för att visa närmare vyer av områden. Presentatören skulle i stället kunna navigera i kartan med bara händerna.

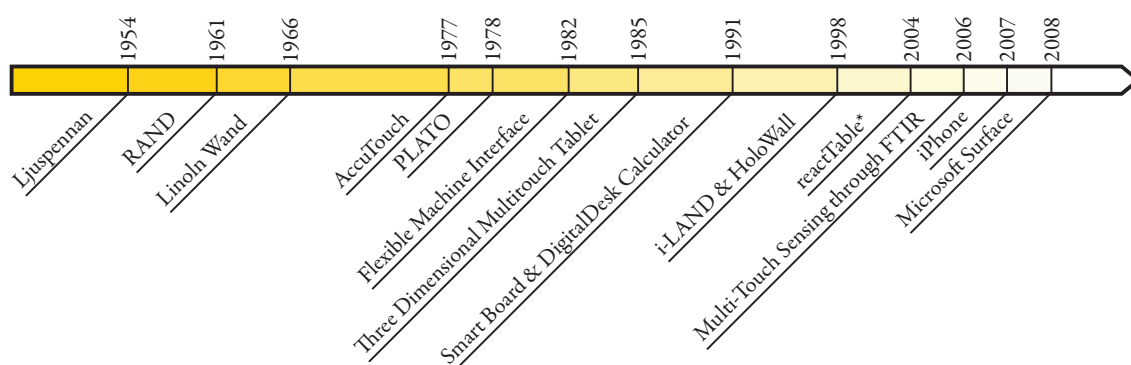
Tekniken lämpar sig också inom design. Det finns en fördel med att kunna arbeta med båda händerna när man gör kurvor. Det mer naturligt att arbeta med två händer och därmed går det också både enklare och fortare att utföra uppgifter [21]. Exempel på en teknik som passar in här är "TapeDrawing" och man slipper då problemet med att omsätta analoga ritningar digitalt senare.

2.2 Historia

Alternativa inmatningsmetoder till tangentbordet har funnits i över 50 år. I detta avsnitt presenteras historiska hållpunkter inom områden som kan relateras till pekskärmar och flerpunktspekskärmar samt lite information kring dem. Figur 2.1 visar hållpunkter i historien.

2.2.1 En-punktsinmatning

I detta avsnitt kommer inmatningsmetoder relaterade till pekskärmar att presenteras.

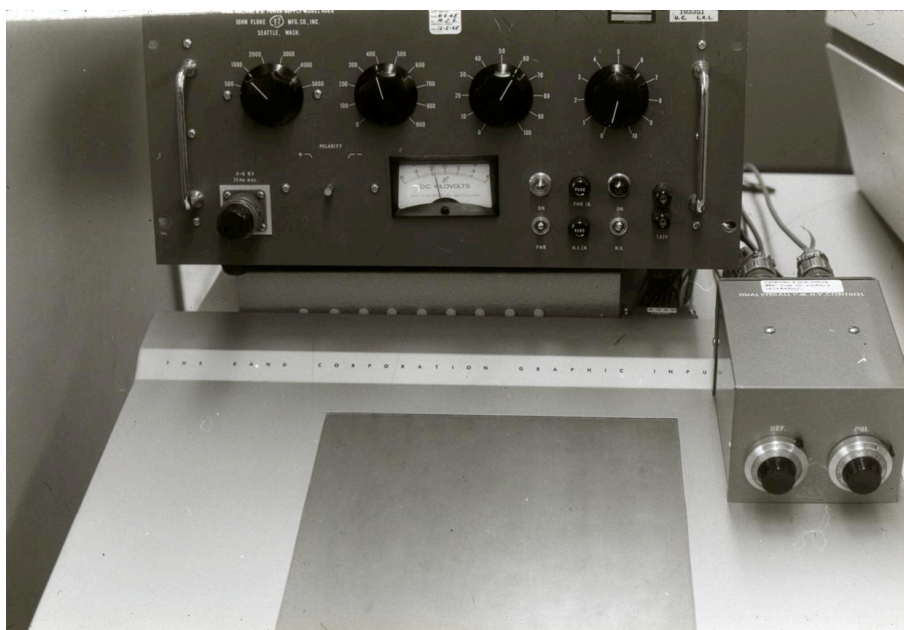


Figur 2.1: Historisk tidslinje för inmatningssystem

1954 introducerades ljuspennan som inmatningsmetod [19]. Med ljuspennan arbetade användaren direkt på skärmen med en för ändamålet utvecklad penna. Pennan hade en inbyggd ljussensor som kände av när CRT-skärmens katodstråle träffade densamma. Katodstrålens position är känd och utifrån denna information kan man härleda pennans position i två dimensioner.

THE RAND TABLET, se figur 2.2, lanserades 1961 [3] och hade en 10 gånger 10 tum stor yta vilken kunde läsa diskreta punkter i planet. Ytan bestod av en elektriskt laddad platta. På ytan användes en speciell penna (sond) för att göra inmatningar. Pennan hade en tryckkänslig spets för att simulera en verklig penna och på så sätt mata in data i tre dimensioner. Då inmatning och utmatning var fysiskt separerade kan THE RAND TABLET främst liknas vid dagens ritbord (tablet)[5].

1966, fem år efter RAND, utvecklades en in-dataenhet kallad THE LINCOLN WAND [15], där användaren kunde peka direkt på skärmen. Fortfarande används en speciell penna för att göra kopplingen mellan pekpunkten på skärmen och datorn. Pennan var utrustad med en ultraljudsmottagare och dess position gick att utläsa i tre dimensioner, till skillnad från ljuspennans två dimensioner. Tekniken fungerar, i kort, så att skärmen har fyra stycken inbyggda ultraljudssändare som periodiskt sänder ut en ultraljudssignal. När användarens penna tar emot ljuden från sändarna kan tiden beräknas för varje sändare. På så sätt kunde positionen

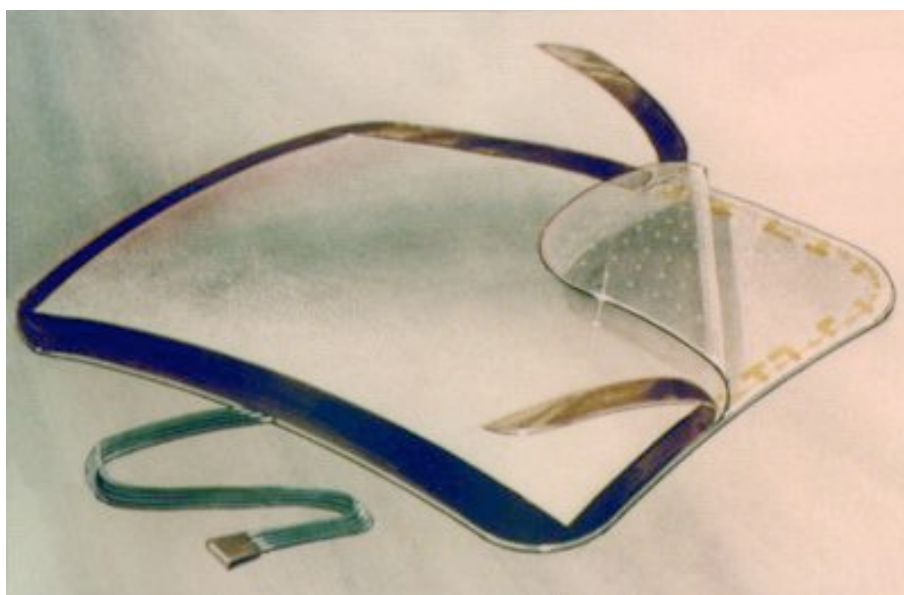


Figur 2.2: The Rand Tablet

för pennan beräknas.

Elva år senare, 1977, tillverkades en transparent yta kallad AccuTouch™[11], se figur 2.3, som kunde avläsa en punkt baserat på fingrets placering på skärmen. Systemet består av två lager av elektriskt ledande material. När ett finger trycker på ytan sluts en elektrisk krets mellan de två lagren. Fingrets position beräknas genom att mäta potentialen, horisontellt och vertikalt, i punkten [6].

Året efter AccuTouch, 1978, användes en pekskärm i en utbildningsprodukt vid namn PLATO [7], se figur 2.4. PLATO hade en plasmaskärm om 512 gånger 512 bilpunkter som dessutom var utrustad med möjligheten att ta emot data genom att placera ett finger på skärmen. En användare kunde på så sätt göra val i olika applikationer utan att använda tangentbordet. Pekskrmen hade en upplösning på 16 gånger 16 punkter. Detta innebar att det var möjligt att känna av 256 olika områden på skärmen. Runt skärmen placerades IR-Ljussändare och mottagare. Dessa skapade ett rutnät av ljusstrålar över skärmen. När ett finger sattes på skärmen bröts två ljusstrålar och detta gav fingrets position som vidarebefordrades till appli-



Figur 2.3: AccuTouch™ 1977 (License pending)

kationen [1].

Från 1980-talet och framåt började pekskärmar leta sig in i allt fler kommersiella produkter [23]. Hewlett-Packard lanserade kommersiella arbetsstationer för personligt bruk med inbyggd pekskärm. Dessa byggde på samma teknik som den i PLATO.

Under nittiotalet, närmare bestämt 1991, presenterades Smart BOARD™. Smart BOARD, är en interaktiv whiteborard som är menad att användas i utbildningssammanhang.

2.2.2 Fler-punktsinmatning

Det här avsnittet kommer presentera inmatningsmetoder relaterade till flerpunktspekskärmar.

Redan 1982 togs en prototyp fram för att kunna känna av flera simultana inmatningspunkter från en eller flera användare. Prototypen fick namnet The Flexible Machine Interface [23] och fungerade genom bildinläsning. En yta läses av med en kamera och fingrarna uppträder som svarta punkter som kameran kan avläsa.

Tre år senare, 1985, presenterades en yta med möjlighet att ta emot flera samtidiga punk-



Figur 2.4: PLATO 1978

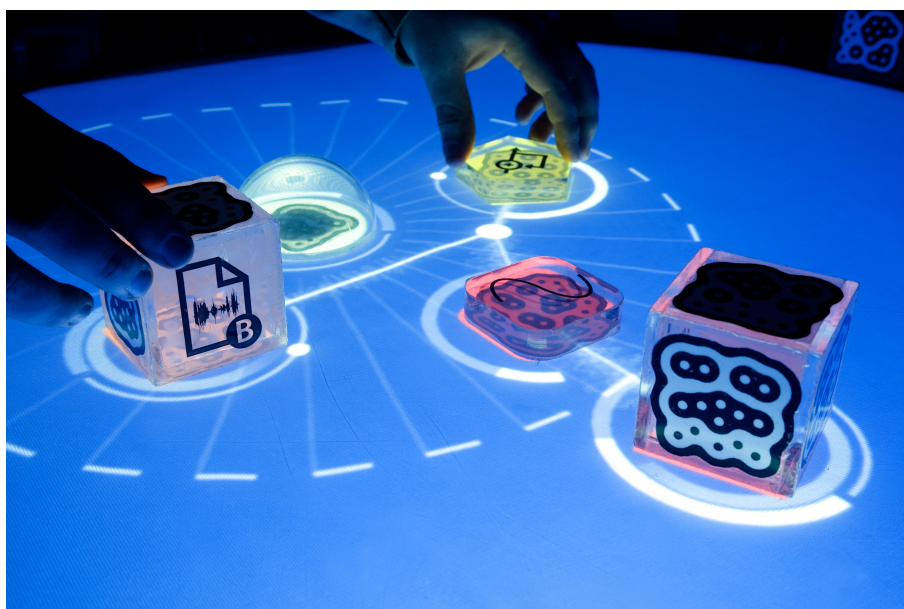
ter och som dessutom talar om graden av kontakt [16]. Ytan, som var indelad i 64 gånger 32 punkter, tillät att användaren arbetade direkt med fingrarna på ytan. Avläsningen skedde genom att ytan periodiskt skannades av med hjälp av en algoritm där punkter med lägre elektrisk laddning upptäcks. Ytan rapporterade plankoordinater samt hur hårt användaren tryckte på punkten. Detta gav en tredimensionell inmatning. Ytan kunde dock inte visa någon bild och bildskärmen var därför tvungen att placeras separat vilket gör att denna yta inte riktigt kan klassas som en äkta pekskärm.

The DigitalDesk Calculator [28] presenterades 1991. Enheten projicerade ett gränssnitt på toppen av ett bord. Bredvid projektorn fanns även en kamera som registrerade användarens rörelser över bordet och dessutom objekt på bordet. Systemet kunde då ta emot flera samtidiga inmatningar och känna igen dokument placerade på bordet.

Sju år senare, 1998, presenterades projektet i-LAND [26]. i-LAND var ett projekt där två enheter togs fram. DynaWall och InteracTable. DynaWall var en stor interaktiv whiteboard som hade möjligheten att ta emot information från flera samtidigt användare och InteracTable var ett bord runt vilket upp till sex personer kunde stå. Bordet kunde ta emot in-data från flera samtidigt användare.

Samma år presenterades även HoloWall [17]. HoloWall var en vägg designad för att kunna ta emot fler samtidigt inmatningar men även känna igen föremål som hölls upp framför ytan. HoloWall fungerar genom att en yta projekteras bakifrån. Vid projektorn finns en kamera som filmar ytan och IR-ljus som belyser upp ytan. Kameran kan registrera det IR-ljus som reflekteras och på så sätt räkna ut pekpunkter.

Under tjugohundratalet, närmare bestämt 2004, presenterades en fungerande prototyp av reactTable*[14], figur 2.5. reactTable* är ett interaktivt musikinstrument som har stöd för att kunna känna igen flera fysiska objekt placerade på ytan samtidigt. Objekt identifieras optiskt med hjälp av en kamera och märkningar på varje objekt. Utifrån objekten, dess riktning och deras inbördes förhållande genereras musik [12].



Figur 2.5: reactTable*

De tre åren 2006, 2007 och 2008 innebar stora språng på marknaden för flerpunktspeksskärmar. 2006 presenterade Jefferson Y. Han en flerpekspunktsyta som kunde konstrueras både enkelt och icke kostnadskrävande [8]. Han presenterade sina idéer på konferensen TED (Technology, Entertainment, Design)[27] vilket fick världen att öppna ögonen för vad som kan åstadkommas.

Apple iPhone™ [9], se figur 2.6, lanserades i början av 2007 och var en av de första telefonerna med en inbyggd flerpunktspeksskärm.



Figur 2.6: iPhone 3G

2008 levererades Microsoft® Surface™ [2], se figur 2.7, för första gången. Microsoft Surface består av att en bild projekteras bakifrån på en yta. Ytan belyses även med infrarött ljus från projektorns håll. Kameror på undersidan registrerar ytan och kan känna igen flera samtidiga användare och dessutom objekt.



Figur 2.7: Microsoft Surface

2.3 Pekskämar och flerpunktspeksskärmar idag

JÄMFÖR

Marknaden för flerpunktspeksskärmar börjar idag ta fart. Microsoft® Surface™ finns idag i ett antal AT&T-butiker [2] och Apple säljer iPhone framgångsrikt [10].

Utöver dessa produkter är interaktiva whiteboards populärt i utbildningssammanhang. Här är det dock främst traditionella peksskärmar som är populära och inte flerpunktspeksskärmar.

När det gäller flerpunktspeksskärmar levererar ett företag vid namn N-trig ytor för detta. N-trig levererar skärmar till bland annat Dell och Intel [20] som har implementerat dem i sina datorer.

2.4 Sammanfattning

Flerpunktspekskärmar ger flera fördelar före pekskärmar. Bland annat låter de användaren använda flera händer och fingrar när denne utforskar ett problem och dessutom låter de flera personer samtidigt interagera med ett system. Flerpunktspekskärmar passar inte i alla situationer. Dock kan de med fördel användas i flera specifika tillämpningar.

Pekskärmar är ingen nyhet på marknaden och så inte heller flerpunktspekskärmar. Det har funnits varianter av pekskärmar sedan 1954 och redan från början har de använts för att mata in data till olika applikationer. På senare år har pekskärmar utvecklats så att de kan ta emot fler än en in-datapunkt.

Kapitel 3

Potentiella lösningar

I det här kapitlet kommer olika lösningsförslag ges till problemet med att skapa hård- och mjukvara för flerpunktspekskärmar. Först tas aspekter angående hur hårdvaran kan konstrueras och byggas upp. Olika metoder för både bildåtergivning och pekpunktsbelysning ges. Senare i kapitlet diskuteras olika lösningsförslag för mjukvaran och hur denna kan bli expanderar.

3.1 Hårdvara

3.1.1 Bildåtergivning

För att flerpunktspekskärmen skall fungera måste in- och utdata presenteras på samma yta. Det finns flera sätt att lösa detta på. Det går kombinera olika sorters bildåtergivningsmetoder med pekpunktsavläsning.

Digitalprojektor

Ett sätt är att använda sig av en digitalprojektor¹. En projektor monteras i chassit och projicerar en bild på en duk eller ett plexiglas i chassits topp. Detta ger en klar och bra bild av gränssnittet. Projektorn kan monteras i sitt ursprungliga skick men kan behöva kompletteras med speglar eftersom avståndet för projicering kan bli kort. [8]

¹Engelska: Video projector

Plattskärm

Ett annat alternativ, som i mindre skala, är billigare än projektorn är att använda sig av en plattskärm². Plattskärmen måste monteras i sär så att bakstycket först och främst kan tas bort. Kretskorten som sitter monterade bakom skärmen måste också flyttas så att de inte sitter i vägen för pekpunktsregistrering, se avsnitt 3.1.2. [18]

3.1.2 Pekpunktsregistrering

För att flerpunktspeksskärmen skall kunna användas måste samtliga pekpunkter som befinner sig på skärmen registreras av systemet.

Elektrisk avläsning

Genom historien har flera olika produkter använt sig av elektronik registrering av pekpunkter. [16][11] Inköp eller utvecklande av en sådan kan rimligtvis inte ligga innanför både budget och tidsrymd för detta projekt. Därför kommer detta alternativ inte att utredas.

Optisk avläsning

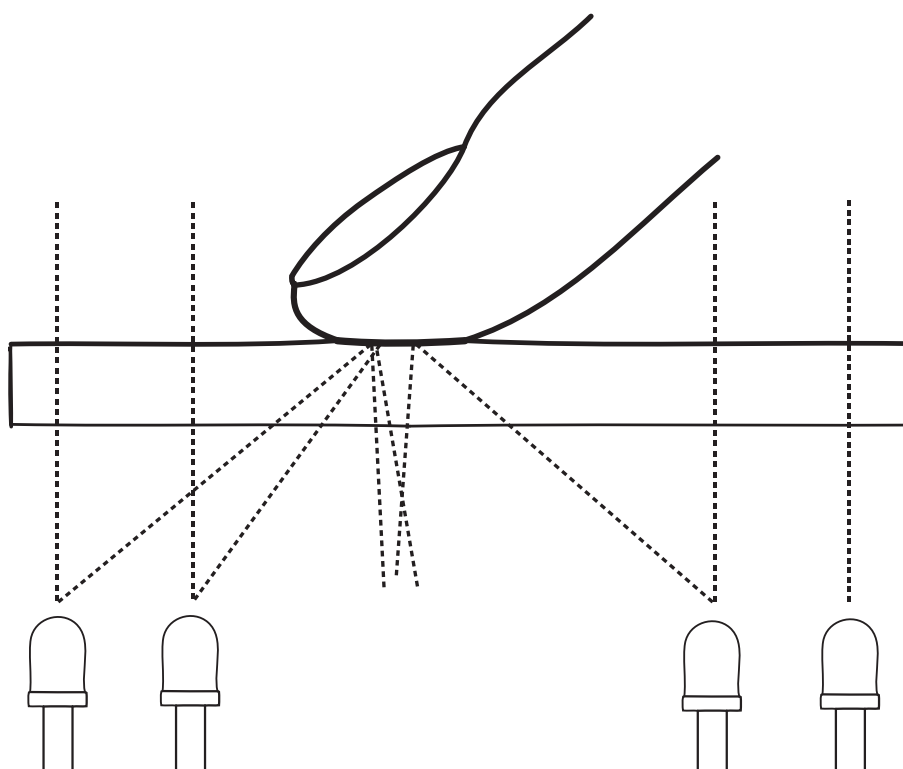
Ett annat sätt att läsa av aktiva pekpunkter på skärmen är genom optisk avläsning. Detta är en simpel metod som fungerar bra.

En kamera monteras i botten på chassit, eller på annat plats med hjälp av speglar, så att den kan registrera undersidan av hela bildytan. Avläsning av ytan sker underifrån så att användarens händer och armar inte är i vägen för avläsningen. Detta gör att det enda som syns är de punkter som är nära skärmens yta eller i kontakt med ytan. För att kameran skall kunna se pekpunkterna måste dessa belysas. För att inte störa skärmens bild bör ett, för ögat, osynligt ljus användas för att skapa pekpunkter. Detta gör att kameran enbart skall se detta för ögat osynliga ljus. [17][22]

²LCD, med eller utan TFT

Utspridd upplysning

Ett sätt att lysa upp punkterna är en teknik som kallas utspridd upplysning³. Ytan som skall registreras belyses underifrån med infrarött ljus. När ett finger placeras på ytan kommer detta att reflektera ljuset nedåt så att detta kan registreras av kameran, se figur 3.1.



Figur 3.1: Utspridd upplysning

Frustrerad total intern reflektion

Ett annat sätt att belysa pekpunkter är via så kallas frustrerad total intern reflektion⁴ (FTIR) [8]. Tekniken går ut på att man i en plexiglasskiva lyser med ett infrarött ljus med en speciell infallsvinkel. När ljuset försöker bryta sig ut från glasskivan kommer det att träffa på luft som

³Engelska: Diffused Illumination

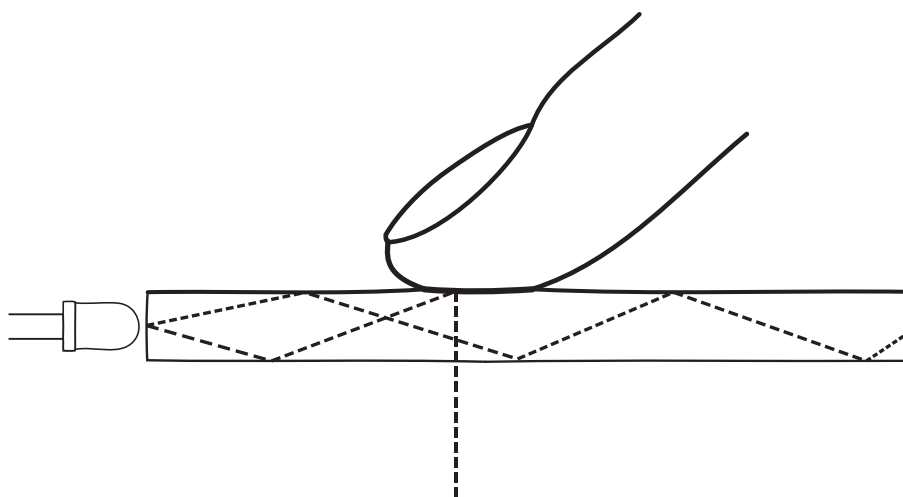
⁴Engelska: Frustrated Total Internal Reflectoion

har ett lägre brytningsindex än glas. Det gör att allt ljus som är större än den så kallade *kritiska vinkeln* kommer att reflekteras och stanna kvar inuti plexiglasskivan.

När ett finger sätts på ytan kommer detta medföra att det frustrerade ljuset bryts ut från plexiglasskivan och därmed träffar kameran.

Den *kritiska vinkeln* kan beräknas med Snells lag.

$$n_2 \sin \theta_c = n_1 \sin 90^\circ \Leftrightarrow \theta_c = \arcsin \left(\frac{n_1}{n_2} \right) \quad (3.1)$$



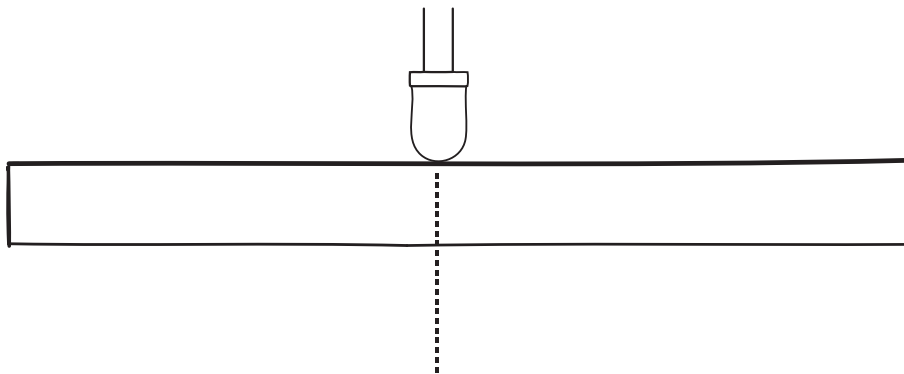
Figur 3.2: Frustrerad total intern reflektion

Belysning i fingerhöjd

Denna lösning belyser fingrarna direkt med IR-ljus. Belysningen sitter i höjd med skärmens yta och kommer på så sätt att träffas av fingrarna när de sätts ned. [18]

Penna

Eftersom vi registrerar pekpunkter underifrån kan vi använda en penna. Pennan ger bra träffsäkerhet



Figur 3.3: Lysdiodspenna

3.1.3 Chassi

Design, ritningar, med mera.

3.2 Mjukvara

3.2.1 Spårare

En spårare⁵ kommer att behöva användas

3.2.2 Klient

Tar emot data från spårare. Gör ngt med den

⁵Engelska: Tracker

3.2.3 Spårare och klientkommunikation

TUIO

3.3 Begränsningar och krav

3.3.1 Hårdvara

Hårdvaran kommer att användas för att visa och navigera mjukvaran. Ytan, för ut- och indata placeras i liggande position, med bildytan uppåt.

3.3.2 Mjukvara

Ska kunna...

Kapitel 4

Hårdvaruprototyp

En prototyp har tagits fram för detta projekt enligt specifikationen i avsnitt 1.3.

4.1 Beräkningar

4.1.1 Kamerans synvinkel

4.1.2 Chassits höjd

4.1.3 Spegelns storlek

4.2 Problem

4.2.1 DI

Gick inget bra, vi såg inte mycket

4.2.2 Lysa ovanifrån

Potentitellt fungerar det, svaga punkter

4.2.3 Penna

Få plats

4.3 Implementation

4.3.1 Bildåtergivning

Skärm har valts för att det är billigt i jämförelse med projektor.

Plattskärm

Skärmen isärmonterad Krestskort kan inte sitta på baksidan, det stör kamerans bild.



Figur 4.1: Plattskärmens panel

Plattskärmen monteras i chassit

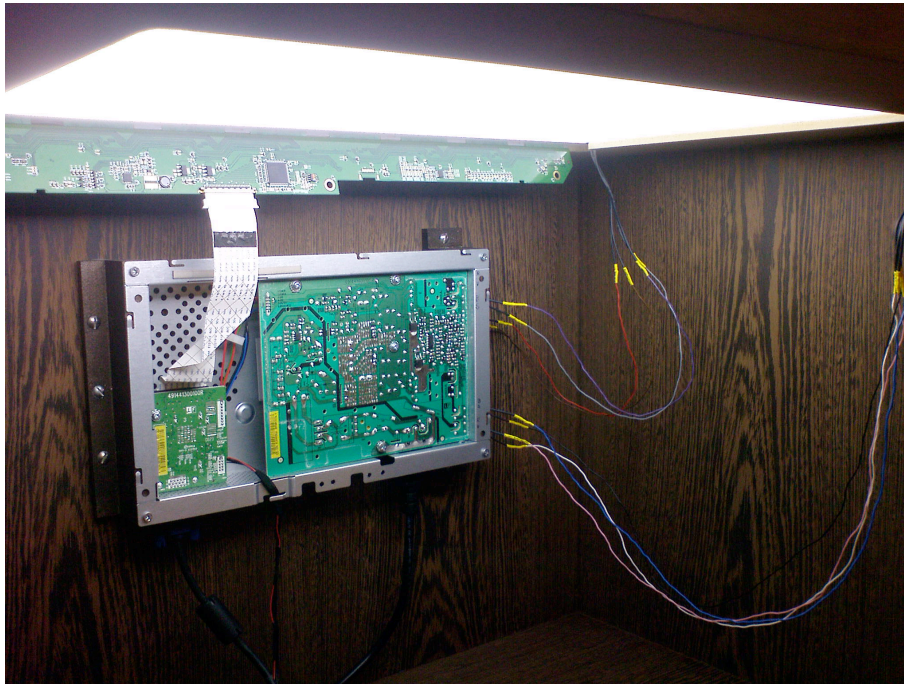
Tog bort ett lager som flyttar bilden åt olika håll. Den förstör.

4.3.2 Bildinläsning

Bilder tas med kameran

Kameran har filter enligt modellen i kap 3

Bilderna skickas till datorn via USB



Figur 4.2: Plattskärm monterad i chassi. På bilden syns undersidan av skärmen.

4.3.3 Chassi

Chassit tillverkades av Magnus Andersson enligt ritningar, bilaga...

4.3.4 Bildinhämtning

Kamera monteras ovan krestkorten, spegel

4.3.5 Pekning

I denna implementation kan man peka med speciella pennor. förhoppningsvis fingrar

Pennor

4.4 Komponentlista

Chassi:

Kapitel 5

Mjukvaruprototyp

I detta kapitel kommer detaljerade specifikationer om mjukvaruprototypens konstruktion att förklaras. Utöver detta kommer bakomliggande tekniker för mjukvaran att förklaras. Mjukvaruprototypen kommer att använda en befintlig spårare.

5.1 Spårare

Den information som kan erhållas från hårdvaruprototypen, se avsnitt 4.3.2, är en sekvens av stillbilder. För tillämpningen är de erhållna bilderna oanvändbara i sitt ursprungliga format. Användbar information utläses ur bilden genom att behandla och tolka bilden, detta är spårarens uppgift.

5.1.1 Funktionalitet

Bilden innehåller en mängd information. Enbart en delmängd av denna information är intressant för tillämpningen, exempelvis är färg inte intressant. Genom att applicera lämpliga filter på bilden gallras irrelevant information bort. Den intressanta information har olika utseende beroende på vilka förhållanden som rådde när bilden togs. På grund av detta behövs olika filter vid olika förhållanden.

Efter filtrering skall spåraren tolka och analysera den kvarvarande information. Spåraren måste identifiera pekpunkter och följa dessa mellan olika bilder i sekvensen. Till sist lämnar spåraren ifrån information om de aktiva pekpunterna och deras position.

Filtrering

Filtreringstekniker

Tolkning och analys

För att kunna finna de pekpunkter som är intressanta finns, förutom den aktuella bilden, en referensbild. Referensbilden är tagen av kameran när inga pekpunkter finns aktiva. Genom att subtrahera informationen i referensbilden från den aktuella bilden har all bakgrundsinformation tagits bort [22].

För att spåraren skall kunna

Behandla med filter Tolkar resultatet Meddelar vilka blobar (kontrulös r

5.1.2 Olika spårare

Det finns flera spårare med öppen källkod som kan användas. Spårarna implementerar olika sätt att lämna i från sig data, gemensamt för de som studerats är att de implementerar TUIO. TUIO, se avsnitt 5.3.1, är ett väl-specifierat protokoll för nätverksbaserad kommunikation mellan spårare och klient. I tabell 5.1 finns exempel på ett antal spårare som implementerar TUIO.

<i>Spårare</i>
BBTouch
Touché
TouchLib
Reactivation

Tabell 5.1: Olika spårare som implementerar TUIO

Val av spårare

Till denna mjukvaruprototyp har spåraren **Touché** valts men vilken spårare som helst, som implementerar TUIO, kan användas.

5.2 Klient

Lime > A slight touch of lime

5.2.1 Användargränssnitt

5.2.2 Gester

5.3 Kommunikation mellan spårare och klient

Klienten måste få information om pekpunkter från spåraren. Hur spåraren tillhandahåller denna information till klienten kan variera beroende på implementation.

5.3.1 TUIO

TUIO [13] är ett protokoll för kommunikation mellan spårare och klient. Protokollet arbetar över valfritt transportprotokoll, men UDP används med fördel. Fördelen med TUIO är att det är framtaget enbart för syftet att förmedla information om optiskt identifierade objekt.

TUIO utvecklades som en del av reactTable* [14], se avsnitt 2.2.2, och bygger på OSC-protokollet. Detta gör det möjligt att använda TUIO på alla enheter som har stöd för OSC, exempelvis Flash [13].

OSC

OSC [30], OpenSound Control, är ett protokoll för kommunikation mellan datorer och multimediaenheter som exempelvis synthar. OSC är optimerat för nätverkskommunikation och är konstruerat för klient/server-kommunikation. OSC är utöver detta oberoende av vilket transportprotokoll som används.

OSCs datablock kallas för paket och ett paket innehåller ett eller flera meddelanden. Ett meddelande består av en adress, en typ samt noll eller fler argument. De datatyper som stöds

enligt OSC är ASCII-strängar, 32-bitars flyttal ,heltal (integer) samt 64-bitars tidstämplar och binär data [30][29].

Adressen i meddelandet ger destinationen [29], adresserna är inte förbestämda i protokollet vilket gör att man kan skapa nya sorters adresser. Adressen är uppbyggd av noder som är separerade med ett “/”, ex. “/sverige/karlstad/universitetsgatan”. Sista noden i adressen är mottagaren av meddelandet.

TUIO över OSC

TUIO definierar en uppsättning meddelanden i OSC, se figur 5.1, med uppgift att förmedla information om objekt. I huvudsak finns två meddelandetyper. En meddelandetyper, *alive*, talar om vilka objekt som befinner sig på ytan. En annan meddelandetyper, *set*, talar om information om de objekt som är placerade på ytan. Utöver dessa definierar TUIO ett meddelande som talar om sekvensnummer (fseq) samt ett meddelande med namnet på servern (source). [13]

```
/tuio/[profileName] source name  
/tuio/[profileName] set [sessionID parameterList]  
/tuio/[profileName] alive [list of active sessionIDs]  
/tuio/[profileName] fseq [int32]
```

Figur 5.1: TUIO, Meddelandetyper

TUIO-meddelanden är uppdelade i profiler, se figur 5.2. Profilnamnet finns lagrad i OSC-adressen och är där ändnoden. Profilen föregås av “/tuio/”. [13]

Varje meddelandetyper har ett antal argument, dessa framgår av figur 5.1. [13]

5.3.2 Ramverk

Ramverk kan användas att länk mot

```

2D Interactive Surface
/tuio/2Dobj set s i x y a X Y A m r
/tuio/2Dcur set s x y X Y m

2.5D Interactive Surface
/tuio/25Dobj set s i x y z a X Y A m r
/tuio/25Dcur set s x y z X Y Z m

3D Interactive Surfaces
/tuio/3Dobj set s i x y z a X Y Z A B C m r
/tuio/3Dcur set s x y z X Y Z m

raw profile
/tuio/raw_[profileName]
/tuio/raw_dtouch set i x y a

custom profile
/tuio/_[formatString]
/tuio/_sixyP set s i x y 0.5

```

Figur 5.2: TUIO, Profiler

s	sessionID, temporary object ID, int32
i	classID (e.g. marker ID), int32
x, y, z	position, float32, range 0...1
a, b, c	angle, float32, range 0..2PI
X, Y, Z	movement vector (motion speed and direction), float32
A, B, C	rotation vector (rotation speed and direction), float32
m	motion acceleration, float32
r	rotation acceleration, float32
P	free parameter, type defined by OSC packet header

Tabell 5.2: TUIO, Meddelandeargument

5.3.3 TUIO i mjukvaruprototyp

Mjukvaruprototypen använder sig enbart av en profil från TUIO. Profilen som används är “/tuio/2Dcur” eftersom unikt identifierbara objekt inte är intressant i denna implementation.

BBOSC

För mjukvaruprototypen har biblioteket BBOSC används för att ta emot och behandla OSC-meddelanden över nätverk. Biblioteket tar emot alla slags OSC-meddelanden och en filtrering måste göras för att se till att endast läsa den profil som är intressant.

```
for (BBOSCMessages* theMessage in messages)
{
    if ([[theMessage address] address] isEqualToString:@"tuio/2Dcur")
    {
        //Ta hand om meddelande
    }
}
```


Kapitel 6

Utvärdering av prototyp

Användandet av Stylus inte så bra som fingrar.

Kapitel 7

Slutsats

Litteraturförteckning

- [1] Bill Buxton. Internetsida, Sep 2008. Multi-Touch Systems that I Have Known and Loved.
- [2] Microsoft Corporation. Microsoft Surface, Sep 2008. <http://www.microsoft.com/surface>, läst Sept. 2008.
- [3] RAND CORPORATION. Rand 2007 annual report. Årsrapport, 2007.
- [4] Anders Lotsson DATATERMSGRUPPEN KTH. E-post mellan anders lotsson och henrik bäck. Sep 2008.
- [5] M.R Davis and T.O Ellis. The rand tablet: A man-machine communication device. pages 1–21, May 1964.
- [6] Morrison et al. Passive touch system and method of detecting user input. Patent 20 070 075 982, United States Patent and Trademark Office, Apr 2007.
- [7] M Grossman and D Walter. Teaching with interactive computer capabilities (plato: Computer-based education for animal breeding). *Dairy Sci*, (61):1308 – 1311, Sep 1978.
- [8] Jefferson Y Han. Low-cost multi-touch sensing through frustrated total internal reflection.
- [9] Apple Inc. Pressmedelände, SAN FRANCISCO, MACWORLD, Jan 2007. <http://www.apple.com/pr/library/2007/01/09iphone.html>, läst Sept. 2008.
- [10] Apple Inc. Pressmedelände, California, CUPERTINO, Jul 2008. <http://www.apple.com/pr/library/2008/07/14iphone.html>, läst Sept. 2008.
- [11] Tyco Electronics Inc. Internetsida, Sep 2008. <http://www.elotouch.com/AboutElo/History>, läst Sept. 2008.
- [12] S Jordà. Sonigraphical instruments: from fmol to the reactable. *Proceedings of the 2003 conference on New interfaces for ...*, Jan 2003.
- [13] M Kaltenbrunner, T Bovermann, and R Bencina. Tuio: A protocol for table-top tangible user interfaces. *Proc. of the The 6th Int'l Workshop on Gesture in Human- ...*, Jan 2005.

- [14] M Kaltenbrunner, G Geiger, and S Jordà. Dynamic patches for live musical performance. *Proceedings of the 2004 conference on New interfaces for . . .*, Jan 2004.
- [15] G Roberts Lawrence. The lincoln wand. *Proceedings of the AFIPS Fall Joint Computer Conference*, pages 223–227, Sep 1966.
- [16] S.K Lee, W Buxton, and K.C Smith. A multi-touch three dimensional touch-sensitive tablet. *CHI '85 PROCEEDINGS*, pages 21–25, 1985.
- [17] Nobuyuki Matsushita and Jun Rekimoto. Holowall: designing a finger, hand, body, and object sensitive wall. *UIST '97*, pages 209–210.
- [18] Nima Motamedi. Hd touch: multi-touch and object sensing on a high definition lcd tv. *CHI '08: CHI '08 extended abstracts on Human factors in computing systems*, Apr 2008.
- [19] Brad A Myers. A brief history of human computer interaction technology. *ACM interactions*, 5(2):44–54, Mar 1998.
- [20] N-trig. Pressmedelände, N-trig's DuoSense™ Digitizer Embedded in Intel's New UrbanMax Mobile Computer, Aug 2008. <http://www.n-trig.com/Content.aspx?Page=PressReleases&PressReleaseId=315>, läst Sept. 2008.
- [21] Russell Owen, Gordon Kurtenbach, George Fitzmaurice, Thomas Baudel, and Bill Buxton. When it gets more difficult, use both hands – exploring bimanual curve manipulation. page 8, Mar 2005.
- [22] J Rekimoto and N Matsushita. Perceptual surfaces: Towards a human and object sensitive interactive display. *Workshop on Perceptual User Interfaces (PUP'97)*, Jan 1997.
- [23] Dan Saffer. *Interactive Gestures: Designing Gestural Interfaces*. O'REILLY, 2008.
- [24] B Shneiderman. Touch screens now offer compelling uses. *IEEE Software*, 8(2):93–94, Mar 1991.
- [25] Ben Shneiderman and Cathrene Plaisant. *Designing the user interface*. Addison-Wesley, fourth edition.
- [26] Norbert A Streitz, Jörg Geißler, Torsten Holmer, Shinichi Konomi, Christian Müller-Tomfelde, Wolfgang Reischl, Petra Rexroth, Peter Seitz, and Ralf Steinmetz. i-land: an interactive landscape for creativity and innovation. *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 120–127, 1999.
- [27] Entertainment Design TED, Technology. Speakers Jeff Han: Human-computer interface designer, Sep 2008. http://www.ted.com/index.php/speakers/jeff_han.html, läst Sept. 2008.

-
- [28] P Wellner. The digitaldesk calculator: Tangible manipulation on a desk top display. *ACM UIST*, pages 27–33, Nov 1991.
- [29] Matthew Wright. Open Sound Control 1.0 Specification, Sep 2002. http://opensoundcontrol.org/spec-1_0, läst Okt. 2008.
- [30] Matthew Wright, Adrian Freed, and Ali Momeni. Opensoundcontrol: Stateof theart 2003. *Proceedings of the 2003 Conference on New Interfaces for Musical Expression*, pages 154–159, May 2003.