

FÖRSTÄRKT VERKLIGHET I MONTERINGSPROCESSER

En användarstudie av videobaserat virtuellt stöd för operatörer i tillverkningsindustrin

AUGMENTED REALITY IN ASSEMBLY PROCESSES

A user study of video based virtual support for operators in the manufacturing industry

Examensarbete inom huvudområdet Datalogi
Grundnivå 30 högskolepoäng
Vårtermin 2014

Oscar Danielsson

Handledare: Daniel Sjölie
Examinator: Mikael Johannesson

Sammanfattning

Förstärkt verklighet ger användare möjlighet att ta del av mer information via digital väg än vad deras sinnen ensamma kan erbjuda. I detta arbete undersöks om förstärkt verklighet applicerad på liknande sätt som handledning i spel kan effektivisera en monteringsprocess. I detta syfte har en huvudmonterad och videobaserad utrustning för förstärkt verklighet utvecklats och jämförts mot pappersinstruktioner i användartester för att undersöka om effektiviseringar kan göras. De två aspekter av effektivitet som har undersökts är tidsåtgång och mängd fel. Ur ett tidsperspektiv har effektiviseringar inte kunnat påvisas men sett till mängden fel har en effektivisering kunnat påvisas. Då testpopulationen enbart bestod av tolv personer finns det ett behov av att i framtiden göra större undersökningar för att säkerställa resultaten.

Nyckelord: videobaserad, förstärkt verklighet, AR, användartest

Innehållsförteckning

1	Introduktion	1
2	Bakgrund	2
2.1	Förstärkt verklighet	3
2.1.1	Optiskt baserad AR	4
2.1.2	Videobaserad AR	5
2.1.3	Avvägning mellan optisk och videobaserad huvudmontering	5
3	Problemformulering	7
3.1	Metodbeskrivning	8
3.1.1	Prototyp och testmiljö	8
3.1.2	Mätning	8
3.1.3	Evaluering	9
3.1.4	Forskningsetiska aspekter	9
4	Implementation	10
4.1	Sökta kriterier för prototyp	10
4.1.1	Delmål 1 för programmering: AR-funktionalitet	10
4.1.2	Delmål 2 för programmering: Videobaserad input	11
4.1.3	Sammanflätning av de två programmeringsdelmålen	11
4.1.4	Mekanisk modifiering av Oculus Rift	14
4.1.5	Användartest av det framtagna AR-programmet	15
4.2	Experiment	16
4.2.1	Experimentets utformning	16
4.2.2	Testmiljö	16
4.2.3	Testpersoner	16
4.2.4	Frågeformulär	17
5	Utvärdering	18
5.1	Presentation av undersökning	18
5.2	Analys	18
5.2.1	Tidsaspekt	19
5.2.2	Tekniska förutsättningar	20
5.2.3	Enkät svar	21
5.3	Slutsatser	22
6	Avslutande diskussion	23
6.1	Sammanfattning	23
6.2	Diskussion	23
6.3	Framtida arbete	25
	Referenser	26

1 Introduktion

Datorspel har väldigt stor frihet att visuellt leda spelare i sina miljöer eftersom dessa miljöer är helt artificiellt skapade. Det ger möjligheter för utvecklare att tydligt peka ut vart spelaren ska förflytta sig och vilka objekt som är viktiga då den grafiska representationen av alla objekt i spelvärlden deterministiskt kan styras av utvecklarna. Ifall möjligheten att grafiskt förstärka viktiga objekt skulle kunna föras över till en mer verklighetsbaserad miljö skulle det öppna upp möjligheter för en helt ny form av handfast virtuell vägledning.

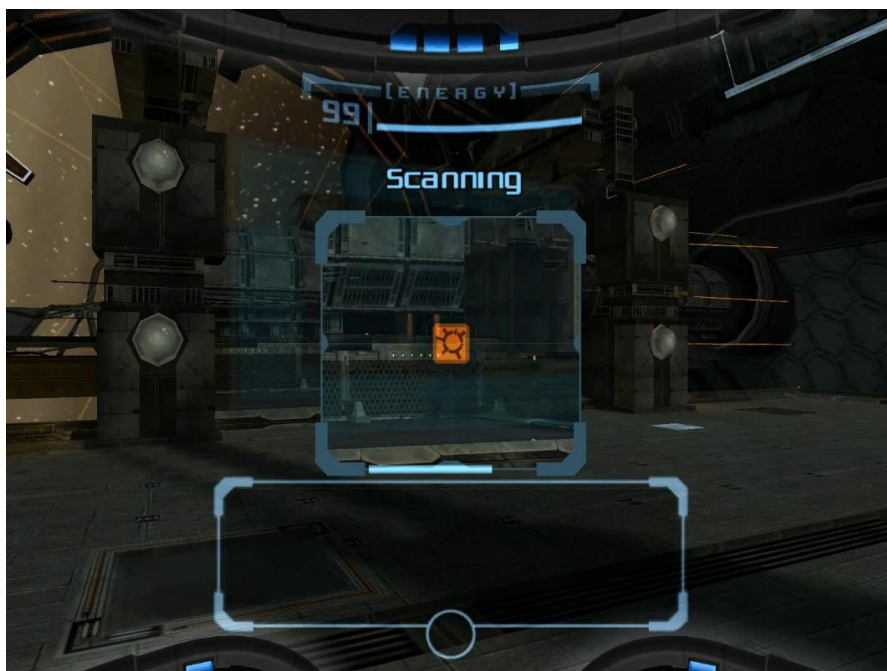
Men datorspelens möjlighet att visuellt designa exakt hur spelmiljön ska se ut ner till minsta objekt och vilket ljus som faller på det är helt beroende av att de är virtuella. Därför måste gränsen mellan virtuellt och analogt bryggas för att kunna nyttja datorspelens visuella verktyg i verklighetsbaserade tillämpningar. Detta görs inom området förstärkt verklighet som ofta förkortas till AR (av den engelska terminologin "Augmented Reality"). Detta arbete undersöker ifall AR kan användas som verktyg i en monteringsprocess för att ge operatörer ett ökat stöd.

2 Bakgrund

Detta arbete är en del av forskningsprojektet Young Operator 2020, vars mål är att tekniskt förbättra industrigolvet för att ge industrioperatörer bättre verktyg för att möta industrins krav på ständig förbättring och nya krav från omvärlden. Huvudfokus på dessa förbättringar är IT-baserade lösningar för att stödja industrioperatörer i att fatta rätt beslut och arbeta optimalt. Förhoppningen är, utöver effektiviseringsförbättringar, att en mer IT-avancerad arbetsmiljö även kan locka ung och välutbildad personal.

I en monteringsprocess måste operatörer följa instruktioner för att inom givna tidsramar korrekt utföra sina arbetsmoment. Ju mer komplexa operationer operatören kan utföra per tidsenhet desto mer värde tillförs produkten som tillverkas. Men ju mer komplexa operationerna blir desto större blir givetvis risken att operationerna utförs fel. För att kompensera för detta behöver operatören instrueras i hur den ska arbeta. Exempel på instruktioner kan vara upplärning utförd av mer erfaren personal, en checklista på arbetsmoment med beskrivning av dem eller teoretisk och/eller praktisk undervisning.

Ett annat område där en användare ofta instrueras i hur de ska utföra komplexa operationer är datorspel. Datorspel är av sin natur virtuella vilket ger dess utvecklare en stor frihet att i detalj styra den miljön som spelaren upplever. Det ger utvecklare möjligheter att bland annat visuellt förstärka instruktioner. Ett allmänt exempel är att under ett upplärningsmoment där målet är att förflytta ett virtuellt föremål till en specifik position kan utvecklaren visa i spelvärlden var positionen är genom att rendera en pil som pekar i vilken riktning spelaren måste förflytta sitt föremål. Positionen kan även i sin helhet eller med en markör markeras ifall det är i en specifik volym spelaren ska placera sitt föremål. Volymen kan då ges en annan färg som kontrasterar mot spelvärlden så att den tydligt syns.



Figur 1 I datorspelet Metroid Prime visas delvis genomskinliga orangea markörer som en slags ikon för att informera spelaren att objektet vid markörens position kan skannas för att ge mer information.

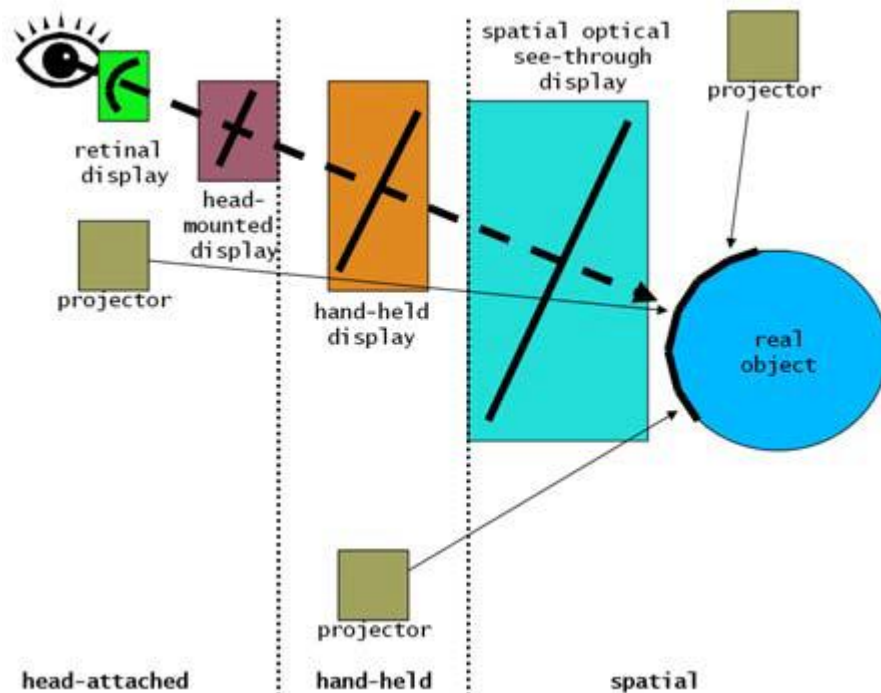
Ifall dessa undervisningsverktyg, som finns tillgängliga i virtuella miljöer, skulle kunna appliceras i verkliga miljöer har detta en stor potential att underlätta undervisning av operatörer i verklighetsbaserade tillämpningar. Tänk till exempel ifall en operatör skulle kunna se en genomskinlig virtuell tredimensionell markering var en specifik komponent ska monteras. Ett sätt att genomföra detta är att för en användare visualisera virtuella objekt i kombination med verkligheten, det vill säga förstärkt verklighet.

2.1 Förstärkt verklighet

Förstärkt verklighet har den engelska förkortningen AR (från Augmented Reality). Ett AR-system beskrivs av Gabbard et al. (2005) som ett system där information som är beroende av miljön på ett direkt sätt visas i användarens synfält. Azuma et.al. (2001) definierar AR som ett system som interaktivt kompletterar verkligheten med virtuella objekt där de virtuella objektens position deterministiskt avgörs av verkligheten. De understryker också att denna definition inte begränsar sig till vare sig vilken teknologi som används i implementeringen eller vilket sinne som förstärks. Dessa källor är förvisso väldigt gamla, inte minst sett till att AR är ett område där mycket förändring har skett sen de skrevs. De förändringar som har skett är dock inom teknisk implementation där AR-teknologi exempelvis har blivit snabbare, detaljrikare och mer lättillgängligt. Men vad som generellt utgör AR som teoretiskt begrepp är dock oförändrat.

För det här arbetet begränsas dock AR både i sin diskussion och sin implementation till visuell förstärkning och framtida referenser av termen AR hänvisar specifikt till visuellt baserad AR. Avgränsningen motiveras främst av tillgänglig teknik och omfattning av arbetet, fler än ett sinne skulle innebära ett behov att även evaluera skillnader mellan virtualiseringsprocessen av de olika sinnena. Det har även en säkerhetsaspekt i att för varje sinne som virtualiseras ökar risken för användaren vid ett eventuellt systemfel då sinnena blir teknikberoende och kan komma att avskärma användaren.

Det finns ett antal olika sätt att tekniskt implementera AR och Krevelen & Poelman (2010) delar upp de olika implementationerna i tre övergripande kategorier: huvudburna, handburna och spatiala. Dessa har olika underkategorier. Huvudburna implementationer kan, enligt Krevelen & Poelman (2010) ytterligare delas upp i retinaprojicering, optisk, video och projektiv. Handburna delas inte upp i ytterligare underkategorier men spatiala implementationer delas upp i video, optiska och projektiva.



Figur 2 Översikt över de olika formerna av AR-implementation gjord av Bimber & Raskar (2006)

För det här arbetet används en huvudburen, videobaserad implementation. En handburen lösning valdes bort eftersom den binder upp användarens händer som är viktiga för det primära arbetet. En spatial lösning, som enligt Krevelen & Poelman (2010) innebär att AR implementeras genom fast monterad utrustning i miljön, valdes bort av hänsyn till för projektet tillgänglig teknologi och tänkta framtida implementationer. En spatial implementation kan dock vara ett intressant alternativ, inte minst vid ett kollaborativt fokus. Valet för detta projekt stod främst mellan en optisk lösning och en videobaserad varför dessa två undersöktes närmare.

2.1.1 Optiskt baserad AR

Optiskt baserad AR beskrivs av Azuma (1997) som att fungera genom att placera optiska kombinerare, som visar en kombination av verkligheten och virtuellt material, framför användarens ögon. Källan är förvisso 17 år gammal, men Azuma (1997) bör ändå ses som en relevant källa ur perspektivet teoretisk definition då den fortfarande är väl citerad. Vid en citeringssökning i databasen Web of Science utförd den 10 april 2014 var artikeln citerad 78 gånger 2011, 73 gånger 2012 och 90 gånger 2013 vilket också är de tre år artikeln har citerats mest enligt databasen Web of Science. Genom att dessa kombinerare är delvis genomskinliga och reflektiva så kan användaren se den verkliga världen och virtuella objekt, som reflekteras mot kombinerarna, samtidigt. På liknande sätt, men mer specificerat mot huvudmonterad visningsteknik, beskrivs optisk huvudmonterad visning av Rolland & Fuchs (2000) som att den verkliga världen syns genom halvgenomskinliga speglar framför användarens ögon och datorgenererade bilder visas på samma speglar. Gemensamt för dessa definitioner är att renderingen av de virtuella objekten läggs till användarens befintliga syn vilket innebär att de områden av användarens synfält där virtuella objekt inte renderas bibehåller användaren sin normala syn undantaget den halvgenomskinliga skärmen.

2.1.2 Videobaserad AR

Videobaserad AR beskrivs av Azuma (1997) och Rolland & Fuchs (2000) som att fungera genom att en eller två huvudmonterade videokameror utgör användarens vy av verkligheten och att kamerornas data kompletteras med virtuell information. Man kan beskriva det som att till skillnad från optiskt baserad AR bygger videobaserad AR på att helt digitalisera användarens syn. Användarens normala synintryck ersätts helt av en digital motsvarighet.

Biocca & Rolland (1998) fann att videobaserad AR kräver att användarens syn förskjuts. De jämförde användares öga-hand-koordination när de var utrustade med en huvudmonterad videobaserad AR-uppkoppling och utan utrustning. Kamerorna som gav användarna syn var placerade 165 mm framför och 62 mm ovanför ögonen och de fann att med utrustningen presterade användarna sämre. Prestationen blev bättre med tiden men uppnådde aldrig samma nivå som vid normal syn. De fann även försämrad prestation en period efter att utrustningen tagits av.

Även Park et al. (2008) undersökte effekten av öga-hand-koordination vid kameraförskjutning. I sitt första experiment testade de ett antal olika förskjutningar och fann att en höjdförskjutning av 35 mm ovanför eller under ögonen gav ett bättre resultat än vid placering i ögonhöjd för alla testade djupnivåer (130 mm, 165 mm och 200 mm). Detta var ett oväntat resultat och Park et al. (2008) misstänkte att resultatet berodde på att vid ögonhöjdspositionering skylde testpersonernas händer vyn av målet. De något förhöjda/försänkta positionerna gav då enligt dem förmodligen en bekvämare position att få en översikt över målet. Men de än mer förhöjda/försänkta positionerna på 70 mm var förmodligen för avlägsna från användarens normalläge. I sitt andra experiment användes ett spegelsystem som gjorde att kamerorna kunde placeras praktiskt taget i ögonens nivå. De fann då stora och genomgående skillnader till fördel för spegelsystemlösningen.

Av detta kan slutsatsen dras att om den videobaserade lösningen som används inte kompenserar för kamerans förskjutna position i förhållande till ögonen kommer initialt användarna att få en stor försämring av koordinationsförmåga. Efter en tids invänjning kan de dock till viss del kompensera för detta, dock inte till sitt normalläge. Om försämringen i koordination inte är kritisk i förhållande till krävd koordination kan en videobaserad lösning vara ett bra alternativ.

2.1.3 Avvägning mellan optisk och videobaserad huvudmontering

Rolland & Fuchs (2000) beskriver avvägningen mellan optisk och videobaserad lösning på följande vis:

“Optical see-through HMDs take what might be called a ”minimally obtrusive” approach; that is, they leave the view of the real world nearly intact and attempt to augment it by merging a reflected image of the computer-generated scene into the view of the real world. Video see-through HMDs are typically more obtrusive in the sense that they block out the real-world view in exchange for the ability to merge the two views more convincingly.”

Rolland & Fuchs, 2000, s.293

Vidare menar Rolland & Fuchs (2000) att avvägningen mellan videobaserade och optiskt baserade lösningar beror på om de ökade utvecklingsmöjligheterna videobaserat erbjuder är värt den extra störningen av den verkliga världssynen. De två olika diskuterade implementationerna skiljer sig alltså främst i hur stor plats tekniken får ta av användarens upplevelse. En optisk lösning är den som minst stör normaltillståndet men en videolösning öppnar upp för bättre överlappning mellan virtuellt och verkligt eftersom den ger total kontroll över vad användaren ser.

Eftersom båda implementationerna är huvudmonterade är de båda beroende av att kunna uppdatera bilden för användaren när huvudet rör på sig. Jeon & Kim (2008) fann i sina observationer att de långsammaste huvudrörelserna motsvarade en vinkelhastighet av 8 grader per sekund och de snabbaste upp emot 80 grader per sekund. I runt 95 % av fallen fann de dock att användarna roterade huvudet med en hastighet av ca 40 grader per sekund. Azuma (1997) fann att 50 grader per sekund är en måttlig huvudhastighet. Vid en systemlatens på 100 millisekunder motsvarar detta enligt Azuma (1997) ett dynamiskt fel av 5 grader vilket vid 68 centimeters avstånd ackumuleras till ett fel på 60 millimeter. Av detta kan vi se att normala huvudrörelser kan skapa ett stort behov av att uppdatera den digitala bilden för att ge användaren en rättvisande bild av sin omgivning och de virtuella objekten som ska läggas till.

I och med att videoupptagning, bearbetning till digitalt format och rendering till skärm omöjlig kan utföras i samma hastighet som ljuset färdas i luft så innebär en digitalisering av synen en viss fördröjning. Det innebär att alla former av videolösningar inom en överskådlig framtid kommer att innebära en fördröjning av användarens syn som förmodligen användare kommer märka sett till hur reaktiv mänsklig syn är. Enligt neurologiska mätningar utförda av Tovée (1994) behöver enskilda neuroner enbart vara aktiva i mellan 20 och 30 millisekunder för att skapa synintryck.

3 Problemformulering

Kan huvudmonterad, videobaserad förstärkt verklighet effektivisera en monteringsprocess stegvist utförd av en montör?

En monteringsprocess avgränsas i den här frågan till en process där objekt placeras i en specifik ordning vid specifika positioner och inbegriper inte någon form av fastsättning av objekten till varandra som exempelvis limning, svetsning eller skruvning. Att fastsättning inte innefattas är en begränsning av omfattning snarare än en indikation på ämnets intresse.

De övriga avgränsningar som har gjorts i frågan motiveras utifrån det tänkta tillämpningsområdet, tillverkningsindustrin. Huvudmonterad utrustning frigör händerna vilket underlättar vid tillverkning. Huvudmonterad utrustning förkortas ofta som HMD (av engelskans Head-Mounted Display). Av motsvarande anledning valdes handburen utrustning bort. En spatial lösning hade kunnat vara ett möjligt alternativ eftersom industrin normalt sett har många statiska arbetsstationer vilket passar bra för en spatial implementering. Men sett till att detta arbete är del av det större målet att modernisera industrimiljön så finns det ett egenvärde i att inte binda sig till lösningar som är beroende av statiska implementeringar.

Valet av videobaserad AR är främst en avgränsning av omfattning. De andra befintliga alternativen är optikbaserad AR, retinaprojicering och projektionsbaserat men valet faller på videobaserat. De övriga alternativen överför alla virtuell information direkt på verkligheten. Därigenom finns det ingen synkronisering av den virtuella informationen och användarens uppfattning av verkligheten vilket leder till att den virtuella informationen blir skakig som tidigare nämnts. Rent praktiskt är även en videobaserad lösning lättare att implementera utan specialiserad utrustning. En videobaserad lösning ger även en större kontroll av vad användaren ser och möjliggör för större precision i kopplingen mellan det virtuella och det verkliga. Precisionen mellan den visuella informationen och verkligheten blir dock lidande eftersom all visuell data måste bearbetas i ett digitalt medium vilket innebär en teknisk fördröjning. I praktiken innebär detta att det användaren ser har blivit mer försenat än om den inte hade sett det genom ett digitaliserande video-skärm-filter och det finns därmed en risk att användaren upplever att verkligheten "laggar efter".

Det är enbart visuell AR som kommer undersökas. Motiveringen till att göra denna begränsning är dels en fråga om omfattning men även en prioriteringsfråga. Det sinne utöver syn som med dagens teknik verkar genomförbar är hörsel, men då industrimiljöer ofta är väldigt bullriga miljöer innebär en hörselbaserad applikation att användningsområdet begränsas enligt Schwerdtfeger et al. (2009). En hörselbaserad implementation ihop med en videobaserad skulle även innebära en väldigt stor avskärmning av användare av tekniken ifrån verkligheten.

Miljön är vald till en monteringsprocess då det finns ett stort intresse från industrin att hitta nya och bättre processer. Det finns även potentiella samhällsvinster som står att finnas som färre produktfel och kortare och/eller lättare inlärningsprocesser för tillverkningsarbetare. Genom att avgränsa sig till en monteringsprocess med fokus på industriellt användande så blir omkringliggande faktorer lättare att styra över då en industriplats är en väldigt kontrollerad miljö. Eftersom en industrimiljö är designad för tillverkning och operatörers

säkerhet så kan miljön ändras väldigt drastiskt så länge effektiviteten ökar. Denna motivering gör att det går att tänka väldigt fritt i design av testmiljö.

Med valet av tillverkningsprocess följer, som nämnts, den industriella arbetsmiljön och därmed även säkerhetsaspekter. Industrimiljön är generellt en högriskmiljö där automatiserade maskiner, truckar med varuleveranser och farliga kemikalier är vanliga. Videobaserad lösning av AR innebär att operatörens syn blir helt digitaliserad och beroende av teknik vid användande. Vid mjukvaru- eller strömfel i HMDn kan operatören inte se sin omvärld förrän felet försvinner eller HMDn tas av vilket kan vara en oacceptabel risk om miljön kräver full visuell kontroll i alla lägen, t.ex. vid operation nära svetsrobotar och/eller truckar. I det här projektet kommer inte specifikt en lösning sökas för att helt arbeta bort risken för temporär förblindning. Detta begränsar användningsområdet och testmiljöer för prototyper i detta arbete till miljöer där temporär blindhet som varar till dess att användaren hinner ta av sig den aktuella prototypen inte utgör en oacceptabel säkerhetsrisk.

3.1 Metodbeskrivning

Eftersom frågan vars svar sökes är fokuserad på användare i en specifik miljö behöver användare och miljö undersökas. Det som ämnas uppnås är en positiv förändring av användarens förutsättningar att navigera i miljön för att utföra specifika arbetsuppgifter. Därför är det lämpligt att göra en jämförelse mellan två i stor utsträckning likvärdiga miljöer där en miljö tillämpar förstärkt verklighet och där en inte gör det. För att bedöma effektivitet är det lämpligast att arbetsmomentet som ska utföras är samma i båda miljöerna och kvantifierbart. Då kan skillnader i tidsåtgång och felprocent utgöra ett effektiviseringsmått.

Initialt identifierade risker är först och främst temporär förblindning ifall kameror eller skärmar får strömbrott som kommer lämna användaren blind tills systemfunktionalitet återfås eller utrustningen tas av. Detta utesluter arbetsmiljöer som är i nära anslutning till trucktrafik eller robotar. En annan risk är illamående relaterat till lång utsatthet av att se på två skärmar som täcker synfältet.

3.1.1 Prototyp och testmiljö

En prototyp har utvecklats för kamerabaserad AR ger användare visuell återkoppling på moment som utförs i verkligheten. Det fanns två viktiga delmål som behövde uppnås för att prototypen skulle vara testbar, dels att den uppnådde interaktivitet, d.v.s. användarens handlingar ska kunna ändra den visade informationen. Men även så pass hög objektigenkänningsförmåga att tillräckligt avancerade arbetsuppgifter kunde visualiseras.

Anpassat till prototypens komplexitet kan därefter en simulerad arbetsmiljö upprättas. För att det ska finnas ett behov av instruktioner har arbetsuppgifterna vara så pass komplexa att instruktioner behövdes. Tidsåtgång måste kunna mätas vilket innebär att start och slutmoment tydligt definierats. Ett tydligt arbetsflöde på vilka moment som ska göras och i vilken ordning har också definierats för att kunna mäta hur väl de olika formerna av instruktioner följdes.

3.1.2 Mätning

I sina mätningar av sina AR-modeller använde Looser et al. (2007) subjektiva mätningar i form av frågeformulär till testpersonerna där upplevd svårighetsgrad samt fysisk och mental påfrestning angavs. Även Schwerdtfeger et al. (2009) använde frågeformulär, där med fokus

på upplevda fysiska påfrestningar. Med bakgrund i detta finner jag stöd i att det är lämpligt att subjektivt evaluera testpersonernas upplevelser av att använda den utvecklade prototypen i kontrast med att inte använda den. För de objektiva mätningarna använde både Looser et al. (2007) och Schwerdtfeger et al. (2009) tidsmätning av arbetsuppgifter och bokföring av fel. Looser et al. (2007) mätte även huvud- och handrörelser men alla tre modeller som utvärderades var olika AR-tillämpningar. Eftersom specifika mätinstrument monterade på användarna krävs för att mäta sådana data finns risken att mätinstrumenten i sig i för betydande utsträckning påverkar mätdata i kontrollfallet. Av denna anledning kommer inte huvud- och handrörelser att mätas.

Eftersom Biocca & Rolland (1998) fann försämringar i koordination hos användare relaterat till synförskjutningen genom kameraplaceringen är det även av värde att mäta skillnaden mellan koordination med och utan utrustning. Detta moment är dock inte kritiskt för resultatet men värdefullt för att bättre differentiera skillnader.

3.1.3 Evaluering

Genom att såväl subjektiva som objektiva mätningar enligt ovanstående uppställning ger goda förutsättningar för kvantifiering av numerisk data kommer en jämförelse mellan prototyp- och kontrollmätningar visa vad det blir för eventuella skillnader mellan de två arbetssätten. Givet att frågorna är relevanta och korrekt formulerade samt arbetsuppgifterna tillräckligt komplexa för att anvisningar krävs och likvärdiga mellan de två modellerna kommer data kunna numeriskt visa skillnader mellan effektivitet och upplevd komfort.

3.1.4 Forskningsetiska aspekter

Som tidigare kapitel beskrivit kommer användartester att utföras. Då detta betyder att mänskligt deltagande kommer vara centralt följer ett antal aspekter som är viktiga att ta hänsyn till. Frivilligt deltagande är den viktigaste aspekten, alla deltagare måste vara informerade om att deras deltagande bygger helt på frivillig basis. De ska ha rätt att närsomhelst avbryta experimentet utan att behöva motivera sig. Information om experimentet och dess syfte måste delges till testpersonerna innan experimenten. De ska även kunna ange om de är intresserade av att höra resultatet efteråt.

Tillverkningsindustrin är en hårt konkurrensutsatt arbetsmiljö och många jobb har effektiviserats bort. Bästa testmiljön, om experimentets uppsättning tillåter det, är i en industrimiljö och med industriarbetare då det är den tänkta tillämpningsmiljön och de tänkta användarna. Ett möjligt orosmoment hos testpersonerna kan då vara risk för att individers prestationer i testet kan spåras och indirekt eller direkt påverka arbetsgivarens omdöme. Därför är det av stor vikt att all prestationsbaserad data anonymiseras och att tillgången till rådata minimeras till så få personer som möjligt. Kopplat till detta kan det även finnas oro för att framtida tillämpningar av experimentet kan leda till effektiviseringar som i slutändan resulterar i färre anställda.

Säkerhetsaspekter har tidigare diskuterats. Om testmiljön är placerad i ett industriområde är det av största vikt att området är avskärmat från tidigare nämnda riskmoment som trucktrafik, robotar och farliga kemikalier. Dessa aspekter är baserade på riktlinjerna uppsatta av Vetenskapsrådet (2002). Även om dessa områden skiljer sig från datalogi är det fortfarande relevanta aspekter att ta hänsyn till då riktlinjerna primärt är för forskning med mänskligt deltagande.

4 Implementation

Detta kapitel tar upp de delmål i implementeringen som har behövt uppfyllas för att nå den färdiga prototypen, en videobaserad, huvudburen AR-implementation. Det går även igenom utformningen av det experiment som tagits fram och genomförts för att, i kombination med prototypen, besvara den sökta frågan.

4.1 Sökta kriterier för prototyp

Viktiga aspekter för den tänkta prototypen är stöd för spårning av flera bildmarkörer, möjlighet till kollisionsdetektering av virtuella objekt, möjlighet att uppdatera bilder baserat på input och möjlighet att bygga mot en plattform som har stöd för Oculus Rift. Det främsta målet med den tänkta prototypen är inte att få en färdig produkt utan att testa ett koncept. Jag misstänkte att det skulle kunna uppstå en del svårigheter med att foga samman två bibliotek som inte var tänkta för detta ändamål vid deras design. Eftersom systemet ska fungera som användarens ögon räknade jag med att det skulle bli en svårighet att mekaniskt konstruera prototypen tillräckligt exakt. Av denna anledning valde jag att prioritera kraftfulla utvecklingsverktyg framför full insyn i biblioteket. Eftersom jag hade viss erfarenhet av spelmotorn Unity (2005) sedan innan och då Oculus Rift har stöd för Unity (2005) såg jag det som en viktig aspekt att AR-biblioteket hade stöd för denna motor. I det första sökskedet hade jag dock inte specificerat dessa aspekter som viktiga utan hade som första mål att bekanta mig med AR som koncept och teknik.

4.1.1 Delmål 1 för programmering: AR-funktionalitet

Det första målet var att implementera någon form av AR som skulle kunna användas i prototypen. Funktionalitet som behövs är: bildigenkänning och modellprojicering (grundläggande AR) samt kollisionsdetektion (för att kunna registrera progression i monteringen). Eftersom detta är ett område som har börjat få mycket fokus de senaste åren finns det ett antal AR-bibliotek med olika för- och nackdelar.

Många av de bibliotek jag fann till en början och innan jag hade en helt klar bild av vilka kriterier som var viktigast för det här projektet hade enbart stöd för mobila plattformar, främst Android. Jag jobbade även parallellt med delmål två och kom där fram till att min slutgiltiga lösning skulle komma att kräva stöd för att bygga mot en skrivbordsmiljö.

Ett av de tidigaste biblioteken som övervägdes var ARToolKit (1999). Den största fördelen med detta bibliotek är att det är ett öppet bibliotek då det är licensierat med GNU-licensen. Alternativt har de även en kommersiell licens vilket skulle ge ökad flexibilitet i framtida kommersiella applikationer. Att det är öppet ger maximal insyn i hur biblioteket fungerar vilket är en stor fördel ifall det finns ett stort behov av detaljkontroll i den tänkta implementationen. Eftersom det inte hade naturligt stöd för Unity (2005) och då jag inte hade tiden att sätta mig in på den nivån som jag bedömde att stöd för Oculus Rift skulle kräva valde jag dock att undersöka andra alternativ.

Initialt föll valet på att använda biblioteket Qualcomm Vuforia (2011). Den främsta motiveringen var att det har ett inbyggt stöd för spelmotorn Unity (2005) vilket, som tidigare nämnts, prioriterades högt för att underlätta i följande delmål. Vid en första anblick verkade det även uppfylla alla sökta aspekter. Det kom tyvärr senare att visa sig att biblioteket vid den aktuella tidsperioden inte hade stöd för att bygga mot desktopmiljöer,

något jag inte upptäckte i den första prototypen då den utvecklades mot android. Jag gjorde det felaktiga antagandet att alla projekt i Unity (2005) automatiskt har stöd för alla plattformar som listas bland byggalternativen. Hur jag löste detta tas upp i kapitel 4.1.3.

4.1.2 Delmål 2 för programmering: Videobaserad input

Oculus Rift har inbyggt stöd för Unity (2005) vilket var en av anledningarna till att det var högt prioriterat att ha stöd för Unity (2005) i AR-delmålet. Det är fullt möjligt att det finns andra hårdvarualternativ som är lämpliga eller till och med lämpligare, men då Oculus Rift uppfyllde de behov som fanns för projektet fann jag det av tidsskäl inte motiverat att fortsätta söka. Det inbyggda stödet för Oculus Rift består i ett digitalt paket som heter OVR (Oculus Virtual Reality). Paketet har två kameror färdiguppsatta för att återge det digitala innehållet som skapas i Unity (2005). För att skapa videokopplingen behövde enbart indata från två webbkameror läsas in och återges till respektive kamera. Detta löstes av Tom Ekblom, en forskningsassistent vid Högskolan i Skövde kopplad till projektet Young Operator 2020, genom att skapa två webbkameratexturer, placera dem framför respektive kamera och därefter ställa in varje kamera till att rendera sin textur och inte den andra kamerans textur.

4.1.3 Sammanflätning av de två programmeringsdelmålen

I detta mål upptäckte jag en allvarlig brist i val av AR-bibliotek i delmål 1. En av styrkorna med spelmotorn Unity (2005) är stöd för multipla utvecklingsplattformar. I normalfallet kan man i en meny enkelt välja en av flera plattformar som projektet ska byggas emot och sen sköter motorn kompileringen till den aktuella plattformen. Här visade det sig tyvärr att Qualcomm Vuforia vid den aktuella tiden inte hade stöd för att bygga mot en windowsmiljö utan enbart Android och IOS. På motsvarande sätt hade Oculus Rift enbart stöd för att bygga mot desktopmiljöer i Unity (2005). Jag ställdes då inför ett val där alternativen var: att finna ett sätt att bygga in stöd i Vuforia-biblioteket för att bygga mot en desktop-miljö, att söka svar på en annan fråga baserat på handburen teknologi eller att välja ett AR-bibliotek med stöd att bygga mot en desktop-miljö. Det första alternativet valde jag snabbt bort då jag bedömde min kunskap och tillgänglig tid som för begränsad för att finna en lösning inom ramen för detta arbete. Det andra alternativet hade varit en möjlig lösning, i det läget hade jag en i grova drag färdig prototyp för Android-kompatibla mobila enheter. Men det alternativet hade också inneburit begränsningar i möjliga tillämpningar. Med en mobil handburen enhet blir användarens händer uppbundna vilket gör lösningen väldigt opraktiskt i många appliceringar i en industrimiljö varför detta alternativ valdes bort. Det sista alternativet var även det som valdes, att finna ett nytt AR-bibliotek med stöd för desktop-kompilering.

Det bibliotek som slutligen valdes för att implementera AR var Metaio (2006). Följande utmaning låg i att koppla samman de två biblioteken. I Unity (2005) använder båda biblioteken varianter av de kameror som finns tillgängliga i Unity (2005) för sin grundläggande funktionalitet. Metaio (2006) använder två kameror, en kamera som läser in vad som syns i en extern kamera, t.ex. en webbkamera och projicerar detta på ett plan. Den andra kameran används för den virtuella projiceringen. Baserat på vad den första kameran ser letar biblioteket efter mönster som används för igenkänning. När mönster hittas avgörs dess position och rotation sett ur den första kamerans perspektiv. Baserat på detta positioneras de virtuella objekt som är kopplade till det aktuella mönstret och renderas som

ett lager ovanför kameraplanet. De två lagren, extern kameravy och virtuell överläggning, slås ihop och renderas till skärmen för visning till användaren.

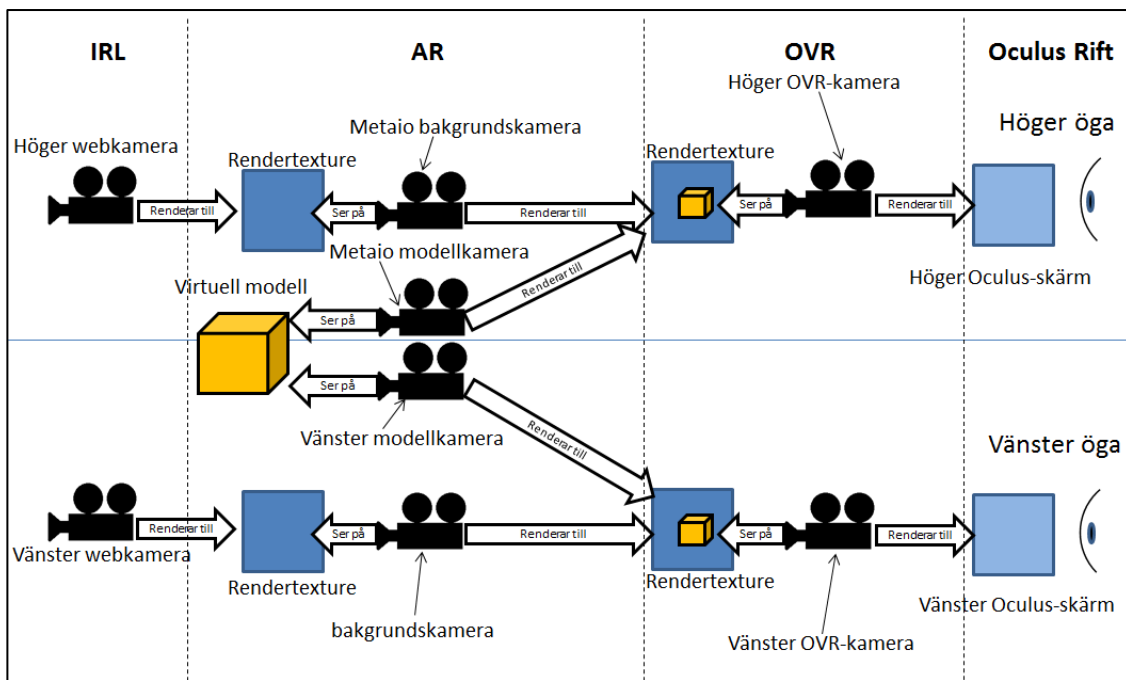
Biblioteket för Oculus Rift, OVR, använder även det två kameror. Dessa kameror renderar den virtuella världen de befinner sig i men med en liten förskjutning mellan varandra för att motsvara förskjutningen av en människas ögon. Det första försöket att rendera AR till Oculus Rift bestod i att skapa två uppsättningar av Metaios (2006) prefab, rendera deras resultat (som normalt skickas till datorskärmen för rendering) till varsin render-texture, placera två plan framför OVR-kamerorna och applicera render-texturerna på dessa planen. Detta visade sig dock inte görbart på grund av en begränsning i Metaio-biblioteket; Metaio (2006) tillåter enbart en AR-kamera per projekt. Användare har frågat efter stöd för flera AR-kameror och detta är något som utvecklarna av Metaio (2006) jobbar på i dagsläget men nu finns inte stöd för detta.

För att lösa detta försökte jag själv utveckla stöd för flera kameror. Det första hindret för flera kameror som jag fann var att ett antal av de klasser som hanterar kameraobjekt hade attributet "static", det vill säga att det bara får finnas en instans av klassen. Första försöket bestod i att lägga till en heltalsvariabel kallad "cameraIndex" till Metaio-kamera-prefaben och lägga till den här variabeln i alla metodanrop av kamera för att på detta sätt kunna skapa flera kameror. En stor utmaning med denna konvertering var att jag var tvungen att göra väldigt många ändringar i flera filer innan jag ens teoretiskt skulle kunna provköra projektet. Dessvärre lyckades jag aldrig få denna lösning att fungera. Längst ner i Metaios hierarki finns en klass, "UnityCommunicationProtos.cs", som använder ett bibliotek som heter ProtocolBuffers, utvecklat av företaget Google. Googles bibliotek är en språk- och plattformsnöjneutral förlängningsbar mekanism för att serialisera strukturerad data enligt projektets beskrivning (Google 2012). Googles bibliotek används i Metaio (2006) för att generera data som därefter används i statistiska funktioner för att implementera AR-funktionalitet. När variabeln "cameraIndex" läggs till blir det kompilersfel därför att ProtocolBuffers inte har metoder med indata som kan ta ett extra heltal. För att finna en lösning via detta spår såg jag två alternativ. Det första alternativet var att bygga bort ProtocolBuffers ur Metaios (2006) klass UnityCommunicationProtos. Den klassen är dock på 4146 rader och protokollet används på över 250 ställen vilket fick mig att göra bedömningen att det var en för tidskrävande lösning där jag heller inte visste säkert att det skulle ge ett fungerande resultat i slutändan. Det andra alternativet var att skriva om i ProtocolBuffers för att skapa stöd för ett extra heltal. Baserat på att det var på premisen att "bara" lägga till ett heltal i Metaio (2006) som jag började denna lösningskedja och att jag inte hade någon erfarenhet av biblioteket så avskrevs även denna lösning som alltför tidskrävande vilket gjorde att jag fick återvända till grundproblemet.

Då jag började få ont om projekttid och ännu inte fått en grundläggande koppling mellan de två bibliotekens funktionalitet var jag nödgad att prova även en väldigt omständigt och designmässigt icke rekommenderad lösning. Jag duplicerade alla script som hanterade kameror på något sätt i Metaio (2006) och döpte dem med prefixen 0 och 1 (t.ex. UnityCommunicationProtos0 och UnityCommunicationProtos1). Detta gick att göra utan att skapa kompilersfel, däremot blev det problem när jag placerat två Metaiokameror med varsin uppsättning script och försökte köra eftersom biblioteket även anropar dynamiska länkbibliotek (.dll-filer). Felet bestod i att jag även behövde duplicera filen metaioSDK.dll och när jag körde i Unity (2005) kunde jag inte se och duplicera filen. Däremot kunde jag bygga programmet, gå in i en mapp, Plugins, där dll-filerna låg och där duplicera filerna till

metaioSDK0.dll och metaioSDK1.dll. Då kunde programmet köras och en lösning såg ut att vara nära. Effekten var att båda kamerorna indikerade att de var aktiva och båda skärmarnas plan renderade input från kameror. Dock renderade båda planen samma kamera och de skiftade mellan de två kamerorna i ett mönster jag inte kunde utröna. Det jag lyckades med var att få Metaio (2006) att byta vilken kamera som agerade AR-kamera men jag lyckades inte få biblioteket att få två kameror att simultant agera AR-kamera.

Slutligen testade jag en annan lösning som innebar en viss begränsning i funktionalitet. I inställningarna för hur AR ska renderas i Metaio (2006) finns det två alternativ. När ett mönster hittas är standardläget att kameran som sköter renderingen av det virtuella innehållet förflyttar sig till ett läge där den kan rendera innehållet i rätt vinkel. Genom att sätta variabeln "transformCamera" (i min implementation är denna variabel omdöpt till "transformCamera") till false förflyttas de aktuella modellerna istället för kameran. Detta ger två fördelar. Dels är detta en förutsättning för att kunna rendera modeller från fler än en markör samtidigt. Det ger även fördelen att det blir lättare att rendera de virtuella objekten för en annan kamera genom att placera den AR-ansvariga kameran vid samma position som denna kamera. I min slutgiltiga lösning finns det enbart AR-funktionalitet på en Metaio-kamera som är placerad vid samma position som OVR-kamerorna. Den högra kameran i OVR-biblioteket tittar på ett plan som har en render-texture placerad på sig. Denna textur får sitt innehåll bestämt av Metaios kamera som hanterar inläsning av data från en extern kamera. På detta sätt får OVR ta del av samma kameradata som Metaio (2006) och den vänstra kameran är satt att se på ett plan som får indata från den andra kameran renderat till sig. I Unity (2005) går det att ställa in vad en kamera ska rendera genom att ge objekt en lagertillhörighet. Kamerorna är satta på att se sina respektive lager vilket gör att lagren kan placeras på samma position utan att de stör varandra. Alla virtuella objekt som är kopplade till att renderas när markörer syns har tilldelats ett specifikt lager, "ModelRendering". Genom detta kan de externa kamerornas vy visas tillsammans med de virtuella objekten. Dock ska de virtuella objekten alltid renderas över det som kommer ifrån de externa kamerorna. Därför fick jag modifiera en shader som lades på planen så att planen inte använde z-buffern. Jag tog den shader som i Unity heter "Diffuse" och lade till "Zwrite Off". Effekten av detta blir att planen ritas bakom övriga objekt eftersom inga av dessa har sin z-buffer avstängd.



Figur 3 Schematisk översikt av flödet i Unity-projektet.

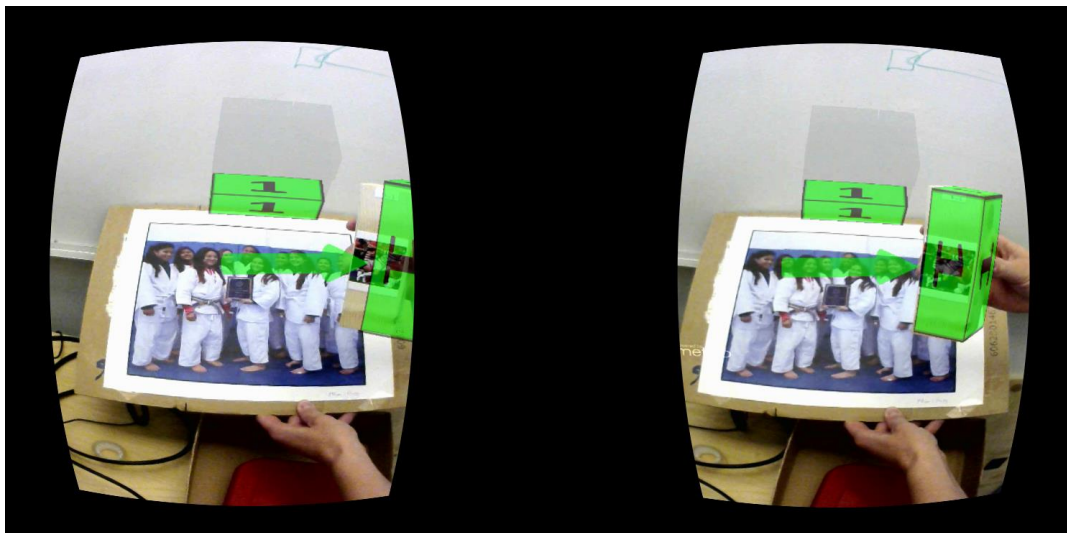
4.1.4 Mekanisk modifiering av Oculus Rift

För att skapa den huvudbaserade aspekten utgick jag ifrån den första allmänt tillgängliga prototypen av Oculus Rift (2012). Inom det virtuella implementationsområdet är Oculus Rift (2012) en huvudmonterad utrustning som är anpassad för VR. Steptoe (2013) har monterat två kameror på en Oculus Rift (2012) och kopplat deras input till skärmarna i den vilket ger resultatet att användarens syn filtreras genom ett digitalt filter. Genom det upplägget blir resultatet en videobaserad och huvudmonterad implementation. För att få ett bra resultat måste kamerorna som monteras på Oculus Riften vara justerbara i sidled för att kompensera för olika användares avstånd mellan ögonen, ett avstånd som benämns "Inter Pupillary Distance" (IPD) av Oculus VR. Min fysiska montering av kamerorna är baserad på den som Steptoe (2013) redovisar för men med ett tillägg för att kunna justera kamerorna i höjddled i enlighet med de resultat som Park et. al (2008) fann.



Figur 4 Front och sidvy av modifierad Oculus Rift.

Den färdiga monteringen illustreras i figur 3 och består av två uppsättningar av en bockad aluminiumplåt som kan träs på från sidan. Dessa plåtar, som utgör fäste för kamerorna, har tre hål i sig och varje hål har en mutterfäst skruv. Skruvarna används för att trä på en plastskiva varpå kamerorna är fastlimmade. Slutligen fästs plastskivorna mot sina respektive plåtar med mutter och bricka. Plastskivorna har för varje skruv 25 mm långa skårar vilket möjliggör förskjutning och fästning av kamerorna i höjdlid. Genom att skruvarna utgör tre punkter för respektive plastskiva och därmed spänner upp dem på ett plan kan plastskivorna vid behov vinklas. Då behöver ett elastiskt material placeras mellan plåt och plastskiva för att genom tryck och fäste från muttrarna fixera plastskivan. För att underlätta kalibrering har även måttenheter fästs. Närmast Oculus Riften finns ett måttband som har mitten, 85 mm, utmarkerad. Vardera aluminiumplåt har även ett måttband där mitten av kameran är utmarkerad (24 mm för kameran längst till vänster i figur 3 och 26 mm för kameran bredvid). I figuren är kamerorna placerade 32 mm från mitten, vilket motsvarar en IPD på 64 mm.



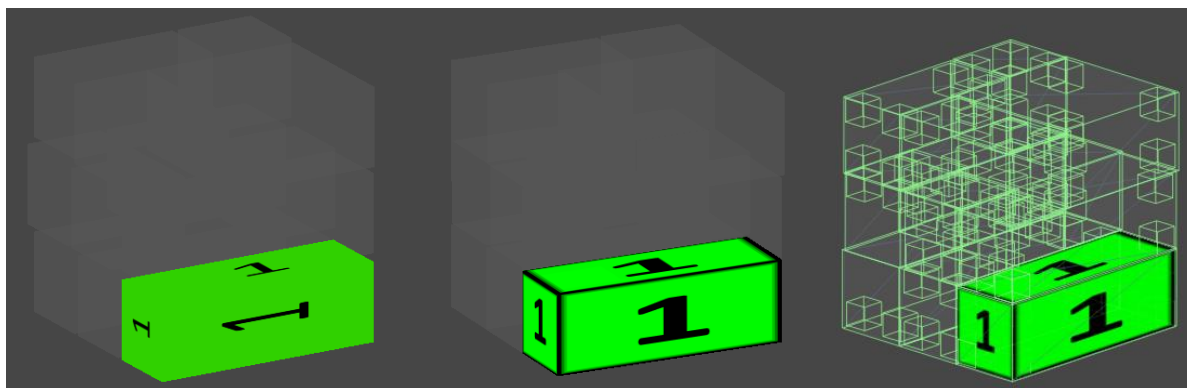
Figur 5 Skärmdump från vad en användare ser med utrustningen. De två uppsättningarna av bilder skapar stereosyn. Notera att pusselbiten inte täcks av den virtuella bilden lika mycket i de två vyerna. Detta blir effekten om den inställda IPDn inte överensstämmer med den faktiska förskjutningen av de verkliga kamerorna.

4.1.5 Användartest av det framtagna AR-programmet

Innan det huvudsakliga experimentet utfördes gjordes ett användartest av den framtagna prototypen för att undersöka om den uppnått tillräckligt hög stabilitet och användarvänlig design för att personer som inte använt prototypen kunde förstå och utföra uppgiften. Vid testet upptäcktes ett flertal designval som behövde förbättras. De pusselbitarna som behöver två primitiva kuber i Unity för att helt representeras hade problemet att varje enskild kub testades enbart mot sin enskilda motsvarighet i pussellayouten. En del testpersoner råkade placera pusselbitar så att bara en av de två delarna uppdaterades vilket resulterade i att pusselbiten de höll på att placera var enbart delvist upplyst. Relaterat till detta upptäcktes att pusslet uppdaterade för snabbt för att användarna skulle förstå. Eftersom kollision kontrollerades enbart genom att varje pusselbit var en eller två kollisionsboxar som hade en eller två lika stora helt virtuella motsvarigheter så räckte det för testpersonen att pusselbiten precis vidrörde området pusselbiten skulle placeras på för att pusslet skulle uppdatera vilket skapade stor förvirring. Den virtuella uppmärkningen av pusselbitarna var också förvirrande

eftersom siffrorna inte var konsekvent orienterade på de olika paren av en pusselbits markering och den motsvarande virtuella markeringen.

Utifrån den återkoppling användarna gav utfördes förbättringar i programmet. Pusselbitarnas virtuella motsvarigheter förtydligades genom att kanterna målades svarta och siffrorna som angav pusselbitens ordning orienterades på samma sätt för varje par av pusselbit och pusselbitens position. Istället för att enbart kontrollera kollision mot hela pusselbitsvolymen och dess motsvarande tänkta plats lades det till kollisionsboxar i hörnen. I den nya designen uppdaterade inte pusslet förrän kollision uppstått med varje hörn vilket tvingade fram att användaren var mer exakt i sin placering innan progression tilläts.



Figur 6 Från vänster till höger är detta hur pusselbitarna förtydligades utefter det första användartestet. Notera grafiskt förtydligande av kanter, orientering på siffror och extra kollisionsboxar i hörn.

4.2 Experiment

Detta delkapitel redogör för upplägget av det experiment som tagits fram för att tillsammans med prototypen kunna besvara den sökta frågan.

4.2.1 Experimentets utformning

För att simulera en monteringsprocess har ett enkelt tredimensionellt pussel med nio bitar konstruerats. Uppgiften som testpersonerna ställs inför är att med hjälp av utrustningen placera pusselbitarna i en bestämd ordning och på bestämda positioner. Detta ska simulera den bestämda ordningen i vilken montering av delkomponenter normalt utförs i en monteringsprocess.

4.2.2 Testmiljö

Då skapandet av testutrustning tog mycket tid i anspråk fanns det inte utrymme för att designa en testmiljö för att efterlikna industrin. Fokus förflyttades istället till att skapa en testmiljö med begränsad stimuli. I en sal där få personer rörde sig i närheten sattes det upp ett vitt bord varpå pusslet placerades samt en stol vid bordet som testpersonerna kunde sitta i. På ett bord i direkt anslutning till vänster stod en dator där AR-utrustningen kunde kopplas in. På ungefär en meters avstånd från bordet omgärdades testytan med vita skiljeväggar för att begränsa yttre stimuli.

4.2.3 Testpersoner

Då testmiljön inte efterliknade en industrimiljö ändrades även fokus på testpersoner. Testpersonerna bestod av personal och studenter vid Högskolan i Skövde. Utöver inspelning

av experimenten och ett frågeformulär insamlades ingen data om testpopulationen såsom kön, ålder, teknikvana och liknande.

4.2.4 Frågeformulär

För att kunna utvärdera testpersonernas upplevelse av experimentet har ett frågeformulär tagits fram med fokus på frågor kring upplevelsen av att använda prototypen. Frågorna är:

1. Jag fann systemet lätt att förstå.
2. Jag fann det lätt att placera en pusselbit.
3. Det kändes som att jag presterade snabbt med det här systemet.
4. Om jag var tvungen att använda utrustning som denna regelbundet skulle jag uppskatta att ha tillgång till den.
5. Jag fann systemet fysiskt ansträngande.
6. Jag fann systemet mentalt ansträngande.
7. Jag fann systemet frustrerande.

Då testpersoner kan komma att vara engelskspråkiga togs även en engelsk version fram:

1. I found the system easy to understand.
2. I found it easy to place a puzzle-piece.
3. It felt like I worked fast with the system.
4. If I had to use equipment like this regularly I would appreciate to have access to it.
5. I found the system physically straining.
6. I found the system mentally straining.
7. I found the system frustrating.

Frågorna utgår från den enkät som Looser et. al (2007) använde men modifierade utefter det skiftade fokuset från interaktionsverktyg för val till monteringsinstruktioner. Svarsalternativen graderas som i ursprungsenkäten med en sjugradig Likert-skala mellan 1 = håller inte med till 7 = håller med.

5 Utvärdering

5.1 Presentation av undersökning

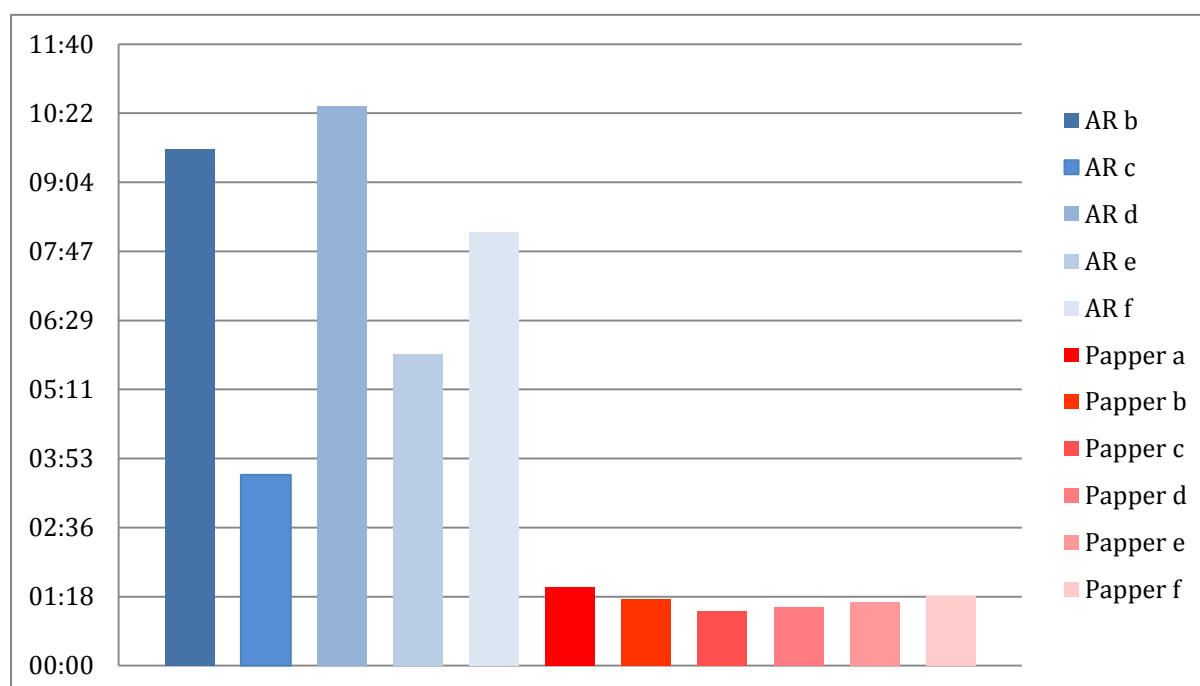
Undersökningen utfördes i två experiment, båda med uppgiften för testpersonerna att montera ett tredimensionellt pussel. Pusslet som lades var ett träpussel med nio bitar och det var exakt samma pussel som lades i båda experimenten, skillnaden bestod enbart i hur testpersonerna fick instruktioner i hur pusslet skulle läggas. Den framtagna utrustningen beskriven i kapitel 4.1 användes i ett av experimenten och en schematisk pappersbaserad ritning användes i det andra experimentet för att ge något att relatera data med. De båda experimenten utfördes med sex olika testpersoner vardera och deras deltagande filmades med syfte att få en bra tidsmätning men även möjlighet att närmare studera eventuella skillnader i utförande mellan de två experimenten. På grund av tekniska svårigheter under mätningen kunde dock inte tid mätas vid ett av AR-experimenten. Under samtliga AR-experiment gjordes ett försök att logga vad testpersonen såg genom programmet Bandicam (2014). Inspelningen misslyckades dock i samtliga fall. Den schematiska pappersbaserade ritningen och den muntligt upplästa informationen till deltagarna finns återgivna i Appendix A.

Vägledning under experimentet fanns men minimerades till efter att testpersonen själv hade försökt men tydligt indikerade att den inte förstod eller om testpersonen uttryckligen bad om instruktioner. Den vägledning som erbjöds begränsades till att upprepa den information som tidigare lästs upp, om än inte alltid ordagrant. Vid två tillfällen behövde dock ytterligare vägledning när testpersonen upprepade gånger placerat på rätt position men utan att pusslet uppdaterades till att visa nästa moment i pusslet. Vid de här tillfällena valde jag att hjälpa testpersonerna genom att lyfta lite på huvudbilden så att pusslet uppdaterades. Dessa ingrepp motiverades med att progressionen hindrades av tekniska begränsningar i prototypen snarare än användarnas förståelse och interaktion med uppgiften.

5.2 Analys

I samtliga testfall förutom två av fallen i pappersversionen utförde testpersonen uppgiften, att lägga pusslet, korrekt vilket innebär att en liten skillnad i felprocent uppstod. Pusslet består av nio bitar vilket innebär att 2 av 54 totala steg utfördes inkorrekt, en felprocent på knappt 4%. Inga av felen ledde till följdfel. Det första felet bestod i att testpersonen placerade pusselbit 1 felorienterad och gick vidare till pusselbit 2. Vid pusselbit 2 upptäcktes felet som uppstått och testpersonen korrigerade felet. Inga fel fanns därför kvar i det färdiglagda pusslet. Det andra felet bestod i att pusselbit 8 började placeras innan pusselbit 7 placerats klart.

5.2.1 Tidsaspekt

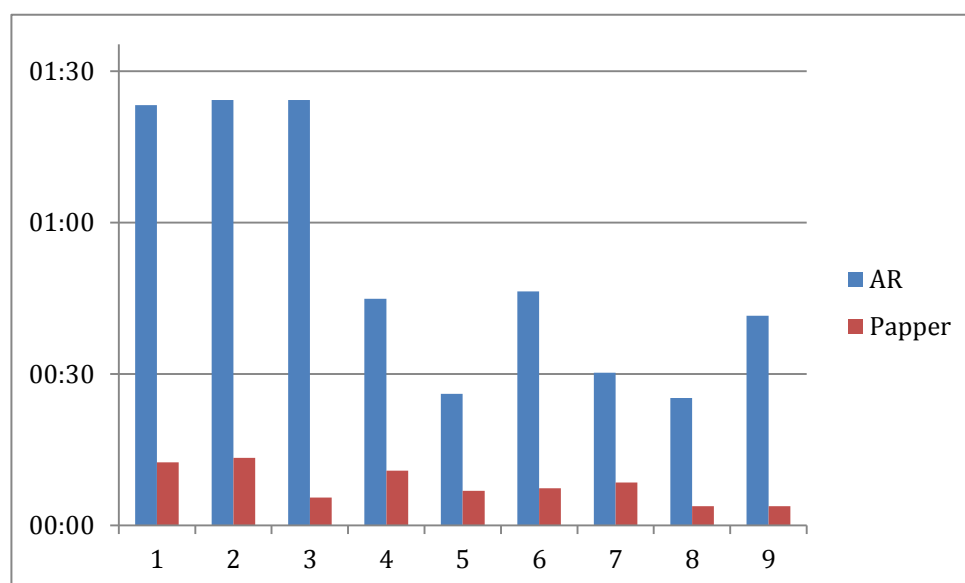


Figur 7 Diagram över tiden det tog för testpersonerna i respektive test att utföra hela uppgiften. Notera att AR a saknas på grund av tekniska svårigheter vid mätningen.

Tidåtgången som återges i figur 7 visar på framför allt två intressanta aspekter. Pappersversionen av experimentet var det snabbaste. Den testperson som tog längst tid för att lägga hela pusslet med pappersinstruktion behövde en minut och 28 sekunder och den som tog minst tid med AR-utrustningen behövde tre minuter och 35 sekunder. En tidsaspekt är därför att pappersversionen var klart snabbare i samtliga testfall; även den testperson som lade AR-versionen av pusslet snabbast behövde fortfarande mer än dubbelt så lång tid som den som lade pappersversionen av pusslet långsammast. En annan intressant tidsaspekt var spridning. I pappersversionen låg samtliga testpersoner inom intervallet 1:01 - 1:28, en skillnad mellan snabbast och långsammast på 27 sekunder, vilket motsvarar att den längsta tiden var ungefär 44 % längre än den kortaste tiden. I AR-versionen låg samtliga tidsmätta testpersoner inom intervallet 3:35 – 10:30, en skillnad mellan snabbast och långsammast på 6 minuter och 55 sekunder, vilket motsvarar att den längsta tiden var ungefär 193 % längre än den kortaste. Den stora skillnaden i tid och mängden frågor under experimentet påvisar en spridning bland testpersonerna i hur bra de var på att anpassa sig till tekniken alternativt vilken vana de hade att hantera teknik av denna typ. Testpersonerna fick ingen möjlighet att bekanta sig med systemet genom att prova det innan testet började. De fick ta på sig utrustningen, ställa in sin IPD (Inter Pupillary Distance) och därefter startade testet.

Figur 8 visar de genomsnittliga tiderna för varje pusselbit i de båda testen. Tiderna i pappersversionen ligger inom ett relativt litet tidsspann i förhållande till AR-versionen. Då testpopulationen enbart bestod av 6 deltagare för vart och ett av de två experimenten var populationen för liten för att kunna dra några generella slutsatser enbart baserat på tiderna, men de gav incitament för en djupare analys av de inspelade experimenten. Det första steget i uppgiften tog mellan 6 och 27 sekunder. Den sistnämnda tiden särskiljer sig markant från

de övriga första tiderna som ligger mellan 6 och 13 sekunder. Framför allt de senare tiderna är generellt väldigt låga i förhållande till AR-versionen, ibland så lågt som 2 sekunder.



Figur 8 Genomsnittlig tid som testpersoner behövde för varje enskild pusselbit

5.2.2 Tekniska förutsättningar

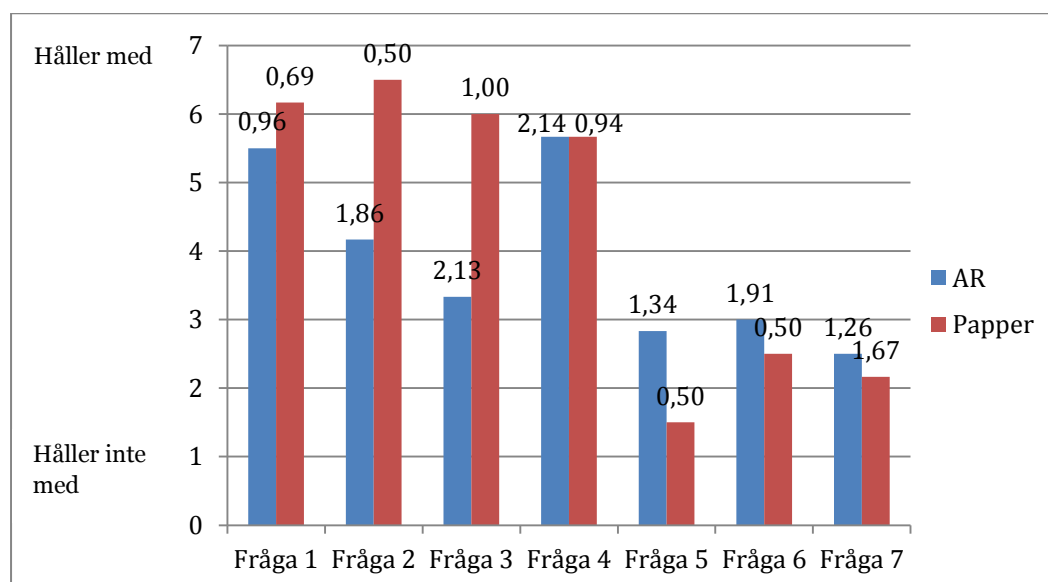
AR-utrustningen baseras på att AR-kameran måste se och tolka bilder för att kunna ge användaren rätt information och detta satte andra begränsningar på AR-experimentet än pappersexperimentet. För att se var på pusslet testpersonen skulle placera nästa bit var den tvungen att hålla kvar den nyckelbild som var knuten till layouten för pusslet i kamerans synfält. Dessutom behövde bilden på pusselbiten synas för kameran för att en korrekt placering skulle kunna loggas. Utöver dessa begränsningar är testpersonens normala synfält begränsat av kamerornas upptagningsförmåga vilket gör att testpersonerna i AR-experimentet hade ett mindre synfält. Vid testen märktes detta av som en begränsning tydligt vid ett antal tillfällen när testpersonen placerade pusselbiten korrekt men fick trots detta inte gå vidare eftersom kameran tappade kontakt med en av de involverade bilderna. I vissa fall behövde testpersonen repetera det korrekt utförda steget upp till tre gånger innan programmet lät testpersonen gå vidare i uppgiften.

Det är inte känt vilken eller vilka felorsaker som faktiskt orsakade dessa uppdateringsfel men de möjliga felfaktorer som identifierats är låg upplösning på kameran, begränsningar i mönsterigenkänningslogiken i programmet, felaktig implementering av mjukvaran, dåligt val av bilder och felaktigt användande. Eftersom inspelningen av vad testpersonerna såg i programmet misslyckades gick det inte att se vad kameran såg när projiceringarna misslyckades. Utifrån min kunskap om hur programmet fungerar samt egen användning av prototypen studerade jag inspelningarna av när testpersonerna lade pusslet med fokus på vad de gjorde när problem uppstod jämfört med när det fungerade.

Pusselbit 1 och 2 skapade viss förvirring dels för att det var testpersonernas första interaktion med AR-utrustningen och dels för att de hade samma form. I startpositionen kunde inte testpersonerna se rätt pusselbits projicering utan att flytta sig närmare pusselbitarna eller hålla pusselbitarna närmare vilket gjorde några testpersoner osäkra. De virtuella kollisionsboxarna som placerats i varje pusselbits hörn gjorde att orienteringen av

pusselbitarna var viktig och viss förvirring följde av att testpersonerna placerade en pusselbit felorienterat utan att pusslet uppdaterades. De fick då de tidigare instruktionerna repeterade att orienteringen var viktig och hur de skulle tolka den pil som syntes vid varje pusselbit. Pusselbitar som hölls på långt avstånd när de placerades tappade ibland sin virtuella koppling till utrustningen då kameran inte kunde registrera bilden när en kombination av såväl avstånd som vinkel gjorde att bilden inte syntes. I vissa fall tappades även kontakten när testpersonen täckte för bilden på den aktuella pusselbiten. Ytterligare ett moment som ledde till svårigheter för testpersonerna var att pusslet inte uppdaterades trots rätt placering därför att det inte registrerats av utrustningen. För att AR-utrustningen skulle kunna registrera korrekt placering krävdes att såväl pusselbitens bild som arbetsytans referensbild syntes samtidigt. Arbetsytans referensbild behövdes enbart för kameran och var i sig inte en del av pusslet och därför tappade en del testpersoner kontakt när de fokuserade enbart på pusslet i sitt synfält.

5.2.3 Enkät svar



Figur 9 Sammanställning av enkät svar. Frågorna återfinns under kapitel 4.2.2. Siffrorna ovanför varje kolumn visar standardavvikelsen bland de svarande på den aktuella frågan.

I enkäten fick samtliga deltagare beskriva sin upplevelse av uppgiften. På samtliga frågor utom den sista var det större skillnader sett till standardavvikelse bland AR-testpersonerna (som kan ses i figur 9). AR-systemet var något svårare att förstå än pappersversionen men för båda systemen höll testpersonerna klart mer med påståendet att systemet var lätt att förstå än de inte höll med, standardavvikelsen var också låg i båda fallen, under ett steg på Likert-skalan. Fråga två tog upp hur lätt arbetsuppgiften var att utföra och där syns en tydlig skillnad, mer än 2 steg på skalan i genomsnitt till fördel för pappersversionen. AR-gruppen ligger väldigt nära 4 på skalan och gränsar på att vara närmare att inte hålla med än att hålla med. Det var också större spridning bland AR-gruppen. Även fråga tre visar en klar skillnad där testpersonerna för AR-systemet i mindre utsträckning upplevde sig prestera snabbt, vilket korrelerar med de uppmätta tiderna, AR-gruppen är närmare att inte hålla med än att hålla med påståendet. Den fjärde frågan visar att båda systemen i lika hög grad accepteras av testpersonerna. Det var dock en dubbelt så hög standardavvikelse i AR-gruppen. Fråga 5 – 7 tar upp olika former av påfrestningar för användaren där den fysiska ansträngningen (fråga

5) sticker ut till nackdel för AR-systemet men i övrigt är skillnaderna små, inom en standardavvikelse på Likert-skalan för båda grupperna, dock till viss fördel för pappersversionen.

5.3 Slutsatser

Det är viktigt att poängtera att testpopulationen bestod enbart av sex individer i vardera två grupper, mer omfattande studier skulle behövas för att kunna dra riktigt säkra slutsatser. De data som finns visar dock några tydliga mönster. Ur ett tidsperspektiv syns klara skillnader till fördel för pappersversionen som var ungefär sex gånger snabbare. Detta är ett rimligt resultat sett till enkätsvaren där AR-versionen av testet genomgående upplevs som svårare och mer ansträngande att använda.

Huruvida mängden fel som har begåtts skiljer sig beror mycket på hur stringent man väljer att tolka situationerna som uppstod. I samtliga fall fanns inget fel kvar i den färdiga produkten, det lagda pusslet. Det första felet upptäcktes först efter att steget hade lämnats och ett nytt påbörjats. Instruktionerna var att ordning och orientering av pusselbitarna var viktig och det här felet bröt mot båda aspekterna. Däremot upptäcktes felet av användaren och det fanns ingen mekanisk begränsning som gjorde det tydligt att stegen byggde på varandra och därför kan det ha varit svårare för testpersonen att respektera begränsningen att göra stegen helt i ordning. Det andra felet bestod i att ett steg påbörjades innan det föregående steget var färdigt vilket också är relaterat till ordningskravet. Inom industrin är feltoleransen mycket låg och eftersom det var samma testperson som utförde båda felen innebär det att två av nio steg inte utfördes enligt den fastslagna standarden vilket är en väldigt hög felmängd.

Med en sträng tolkning av fel som uppstod är AR-utrustningen bättre, inga fel kvarstod när testpersonen gick vidare till nästa steg eftersom systemet inte tillät testpersonen att gå vidare till nästa steg förrän nuvarande steg var korrekt utfört. Om en testperson placerade en pusselbit fel men korrigerade detta innan den gick vidare har detta inte tolkats som ett fel. Den inbyggda kontrollen i AR-utrustningen ger en extra säkerhet i processen som hjälper till att skydda mot den mänskliga faktorn.

6 Avslutande diskussion

6.1 Sammanfattning

Den fråga som har sökts ett svar till var: Kan huvudmonterad, videobaserad förstärkt verklighet effektivisera en monteringsprocess stegvist utförd av en montör? För att kunna besvara denna fråga har en prototyp för huvudmonterad förstärkt verklighet tagits fram och testats i jämförelse mot ett pappersbaserat test. De effektiviseringsaspekter som undersöktes var tid som behövdes av testpersonerna för att utföra uppgiften samt hur många fel testpersonerna gjorde. De utförda testen visade att prototypen inte lyckades effektivisera ur en tidsaspekt. Däremot visade de utförda testen att prototypen lyckades effektivisera ur en felmängdsaspekt. Testpopulationen bestod dock enbart av sex personer i varje testgrupp och de utförde enbart en version av testet vilket gör det svårt att dra några generella slutsatser från underlaget.

6.2 Diskussion

Den framtagna AR-utrustningen bör främst ses som en prototyp avsedd att testa ett koncept. Arbetsuppgiften är ytterst förenklad i att den består i enkla geometriska former med bilder som skiljer dem åt tydligt och det är enbart nio steg att utföra. Detta kan dock ha varit till AR-utrustningens nackdel i att uppgiften var relativt enkel att förstå. När testpersonerna i pappersversionen presterade som snabbast utförde de ett steg på ca två sekunder och när uppgiften är att känna igen, plocka upp och placera en tråkloss är det svårt att fysiskt utföra steget mycket snabbare vilket kan indikera att uppgiften var för lätt eftersom testpersonerna inte verkade behöva lång tid för att förstå uppgiften. Om uppgiften hade haft fler steg, inte enbart pusselbitarna som behövdes funnits att tillgå och pusselbitarna varit mer lika hade det kunnat vara svårare att utan instruktioner själv härleda vad som skulle göras vilket hade kunnat ge AR-utrustningens potentiella förmåga att snabbt identifiera sökta mönster bland annan information en större fördel. De sista två stegen är lätta att gissa sig till hur de ska utföras eftersom testpersonen då enbart har två pusselbitar kvar att välja och de är tydligt olikformade, vilket även de lägre genomsnittstiderna för just de två sista stegen indikerar.

Relaterat till enkelheten i uppgiften är familjaritet med pappersinstruktioner. Att följa schematiska pappersinstruktioner är vanligt förekommande i de flesta människors liv i allt från leksaker som Lego till möbler från IKEA. Det är därför rimligt att anta att pappersversionen var lättare att relatera till för testpersonerna. AR-versionen bygger på teknik som ännu inte finns kommersiellt och det är utefter det rimligt att anta att samma familjaritet inte finns för detta system. I figur 8 framgår att de första tre pusselbitarna tog tydligt längre tid att placera för testpersonerna i AR-versionen vilket även stämmer överens med att många frågor då ställdes kring hur systemet fungerade. Testpersonerna gavs inte möjlighet att bekanta sig med utrustningen mer än att deras IPD ställdes in innan testet började vilket krävde att de hade på sig utrustningen i några minuter innan testet började. Under inställningen var dock inte kamerorna aktiverade så testpersonerna fick enbart bekanta sig med att ha utrustningen på sig och att se något på skärmarna. Utifrån detta är det rimligt att testpersonerna hade presterat bättre med AR-systemet om flera tester hade utförts under en längre tid med samma testpersoner alternativt gett testpersonerna mer tid att bekanta sig med systemet innan testet påbörjades. Till viss del stämmer detta även med pappersversionen men då många av tiderna där redan var så låga som enbart några

sekunder blir det svårt i och med fysiska begränsningar att utföra arbetet snabbare i mycket större utsträckning än i AR-systemet där skillnaden mellan lägst och högst genomsnittlig tid är över en minut. Förbättringspotentialen genom att lära sig systemet är helt enkelt större med AR-systemet.

Vid ett flertal tillfällen utförde testpersonerna i AR-versionen steget korrekt men var tvungna att upprepa steget flera gånger för att systemet misslyckades med att uppdatera korrekt. Förbättringsutrymmet är stort ur ett tekniskt perspektiv och många aspekter spelar in i att AR-utrustningen inte var så exakt. Den huvudmonterade displayen består av första versionen av Oculus Rift Developer Kit och är inte en färdig produkt. Upplösningen på skärmarna är enbart 640 x 800 pixlar och det går tydligt att se de enskilda pixlarna, inte minst eftersom de är placerade med ett litet mellanrum mellan varandra. Effekten blir att det man ser verkar ha ett finmaskigt svart nät lagt på sig. AR-biblioteket Metaio (2006) har inte ett officiellt stöd för vare sig mer än en kamera eller Oculus Rift (2012) vilket har inneburit begränsningar i vad som kunnat implementeras. Kamerorna som används för AR-detektering och som användarens ögon har begränsad vinkel de kan se i vilket gör att hela upplösningen på Oculus Rift (2012) inte används vilket kraftigt begränsade testpersonernas synfält. Upplösningen på kameran sätter även en gräns på hur långt avstånd bilder kan kännas igen på, en högre upplösning skulle ge användare möjlighet att se AR-information längre ifrån sig själv. Om dessa begränsningar kunde förbättras är det rimligt att anta att det kommer ge bättre resultat för framtida testpersoner.

Förhoppningen är att de framtagna resultaten ska kunna bidra till att arbetet fortskrider att effektivisera industrin genom att ge operatörer ett effektivare verktyg för att lära sig och vägledas i sitt arbete. Säkerhetsaspekten av att operatörers synfält blir digitalt har redan diskuterats under kapitel 3.1.4. Men en annan viktig aspekt att ta i beaktning är interaktion mellan människa och maskin. Om den här tekniken får mogna till det stadiet att den kan användas av operatörer i sitt dagliga arbete, vad blir då effekten av att en hel arbetsdag vägledas exakt i vad man ska göra av en maskin? Industriarbete är förvisso väldigt styrt och standardiserat redan idag, operatörers arbetssteg mäts i sekunder med detaljerade arbetsbeskrivningar men kräver samtidigt personal kunniga i sitt hantverk. Finns risken att tekniken hjälper för mycket, att den helt eliminerar människors beslutande i den dagliga processen? Styrkan att potentiellt kunna utföra en monteringsuppgift enligt maskinens instruktioner utan att ens ha sett produkten innan kan även vara en svaghet i att operatören kan bli för fränkopplad från sitt arbete och bara bli en förlängning av maskinens arbete. I och med att tekniken kan registrera att arbetsuppgifter utförts är det kanske också möjligt att tekniken i framtiden kan användas för att vägleda maskiner i sitt arbete och ytterligare minska antalet arbetstillfällen inom industrin.

Inbyggt i den teknik som använts finns även en integritetsaspekt. Genom att operatören med tekniken skulle komma att få kameror som ögon är det givetvis möjligt att dessa kameror i framtiden även spelar in vad de ser. Arbetsgivare skulle på en tidigare aldrig uppnådd nivå kunna bevaka arbetstagare ner till vad de tittat på varenda sekund av deras arbetsdag, föra exakt statistik av vilka arbetsuppgifter de utfört när och hur. Vad innebär det för arbetsmiljön att inte bara vara ständigt övervakad hela arbetsdagen utan också att operatören själv är den som står för övervakningen genom sina arbetsverktyg?

I kapitel 2.1.3 togs de fysiologiska begränsningarna i människors registrering av synintryck upp i relation till den oundvikliga tekniska begränsningen med den här typen av AR att

synintryck blir fördröjda då teknik ska tolka det inkommande ljuset innan det sänds vidare till användarens ögon. Olika känslighet hos individer för denna fördröjning skulle kunna innebära att alla användare inte kan använda utrustningen utan att känna illamående. Implikationerna av detta blir då att om denna teknik blir standardiserad arbetsutrustning och en lösning inte hittas på illamåendet kan de människor som är känsliga för det diskvalificeras från arbetstillfällen de ansågs kapabla till innan teknikens införande.

6.3 Framtida arbete

Som diskussionen ovan tagit upp finns det mycket förbättringspotential som blir möjlig med precisare teknik men det finns även ett par faktorer som inte hann testas mer. Att testa olika höjdnivåer på kamerorna för att undersöka ifall Biocca och Rollands (1998) resultat även kan appliceras på den framtagna prototypen hade varit intressant. Ett mer avancerat pussel eller annan monteringsuppgift där testpersonernas förståelse av uppgiften utanför uppgifterna lättare kunnat räknas bort är också av intresse. Den framtagna prototypen har möjlighet att justera kamerorna i höjddled såväl som sidled. I experimentet fanns det dock enbart tid att justera kamerorna i sidled för deras individuella IPD och låta alla testpersoner testa med en höjd. Kamerorna var placerade något nedanför ögonen, mitten av linsen var ca 55 mm från Oculus Riftens nederkant att jämföra med mitten av Oculus Riftens lins som är ca 60 mm från dess nederkant. Det hade varit intressant att se om justering av höjddled kunnat ge annorlunda resultat och i så fall vilka.

Såväl AR som Oculus Rift är tekniker där det i dagsläget sker mycket utveckling. I takt med att tekniken mognar mer kommer det öppna upp för mer avancerade implementeringar. När Oculus Rift är färdigutvecklad och släpps som kommersiell produkt kommer den ha bättre skärmapplösning och uppdateringsfrekvens vilket kommer möjliggöra att mer användarvänliga och exakta implementeringar av huvudmonterad AR kommer kunna skapas. Redan version två av utvecklingsversionen av Oculus Rift (2012) kommer ha en upplösning på varje skärm på 960 x 1080 pixlar och den kommersiella versionen kommer ha ännu högre upplösning. Den nuvarande prototypen har främst brister relaterade till att bättre teknik ännu inte finns tillgänglig vilket gör detta till ett mycket intressant område allt eftersom tekniken utvecklas.

Referenser

- Azuma, R. (1997) *A Survey of Augmented Reality*. Malibu, Hughes Research Laboratories
- Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S. & MacIntyre, B. (2001) Recent Advances in Augmented Reality. *Computer Graphics and Applications, IEEE*. 21 (6), s. 34-47.
- Bandisoft (2014) *Bandicam* (Version 1.9.4.505) [Datorprogram]. Bandisoft. Tillgänglig på Internet: <http://www.bandicam.com>
- Bimber, O. & Raskar, R. (2006) Modern approaches to augmented reality. *International Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '06. New York, NY, USA, ACM*.
- Biocca, F. & Rolland, J.(1998) Virtual Eyes Can Rearrange Your Body: Adaptation to Visual Displacement in See-Through, Head-Mounted Displays. *Presence: Teleoper. Virtual Environ.* 7 (3), s. 262-277.
- Forskningsetiska principer inom humanistisk-samhällsvetenskaplig forskning.* (2002). Stockholm: Vetenskapsrådet Tillgänglig på Internet: http://www.cm.se/webbshop_vr/pdf/etikreglerhs.pdf [Hämtad Februari 7, 2014].
- Gabbard, J., Swan, J., Hix, D., Schulman, R., Lucas, J. & Gupta, D (2005) An Empirical User-based Study of Text Drawing Styles and Outdoor Background Textures for Augmented Reality. *Proceedings of the 2005 IEEE Conference 2005 on Virtual Reality. Washington, DC, USA, IEEE Computer Society.* s. 11-18, 317.
- Krevelen, D. & Poelman, R. (2010) A Survey of Augmented Reality Technologies, Applications and Limitations. *The International Journal of Virtual Reality.* 9 (2), s. 1-20.
- Looser, J., Billingham, M., Cockburn, A. & Grasset, R. (2007) An Evaluation of Virtual Lenses for Object Selection in Augmented Reality. *Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia.* New York, NY, USA, ACM. s. 203-210.
- Metaio (2006) *Metaio* (Version: 5.3) [Datorprogram]. Metaio. Tillgänglig på Internet: www.metaio.com/products/sdk
- Nara Institute of Science and Technology (1999) *ARToolKit* (Version: 2.72.1) [Datorprogram]. University of Washington HIT Lab. Tillgänglig på Internet: www.hitl.washington.edu/artoolkit/download
- Oculus VR (2012) *Oculus Rift* (Version: Developer kit) [Utvecklingsmiljö]. Oculus VR. Tillgänglig på Internet: www.oculusvr.com
- Park, M., Serefoglou, S., Schmidt, L., Radermacher, K., Schlick, C & Luczak, H. (2008) Hand-Eye Coordination Using a Video See-Through Augmented Reality System. *The Ergonomics Open Journal.* 1 (1), s. 46-53.
- Retro Studios & Nintendo (2002) *Metroid Prime* [Datorprogram]. Nintendo.

- Rolland, J. & Fuchs, H. (2000) Optical Versus Video See-Through Head-Mounted Displays in Medical Visualization. *Presence: Teleoperators and Virtual Environments*. 9 (3), s. 287-309.
- Schwerdtfeger, B., Reif, R., Günthner, W., Klinker, G., Hamacher, D., Schega, L. , ... Tümler, J. (2009) Pick-by-Vision, A First Stress Test. *Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on*. S. 115-124.
- Steptoe, W (2013) AR-Rift: Stereo camera for the Rift & immersive AR showcase. *Oculus Developer Forums*. Tillgänglig på Internet: <https://developer.oculusvr.com/forums/viewtopic.php?f=28&t=5215> [Hämtad April 7, 2014].
- Tovée, M. (1994) Neuronal Processing: How fast is the speed of thought? *Current Biology*. 4 (12), s. 1125-1127.
- Qualcomm (2011) *Vuforia* (Version: 2.8) [Datorprogram]. Qualcomm. Tillgänglig på Internet: <https://developer.vuforia.com/resources/sdk/android>,
- Unity Technologies (2005) *Unity* (Version: 4.3.4) [Datorprogram]. Unity Technologies. Tillgänglig på Internet: <https://unity3d.com/unity/download>.

Appendix A - Testinstruktioner

Följande text lästes upp för testpersonerna vid testets genomförande. För svenskspråkiga användes den svenska versionen och för engelskspråkiga den engelska versionen. Det första stycket i varje språkversion var allmän information som lästes upp för samtliga testpersoner och av de följande två styckena lästes enbart den text som var aktuell för det specifika testet.

Den svenska versionen:

Generell information

Syftet med detta experiment är att undersöka effektiviteten i olika monteringsinstruktioner. Ditt deltagande är helt frivilligt och kan närsomhelst avbrytas utan motivering. Data kommer att hämtas men kommer inte att kunna kopplas tillbaka till dig som person. Ditt deltagande kommer att filmas i syfte att inhämta data. Om du känner dig obekvämt med att filmas behöver du enbart säga ifrån och då filmen kommer enbart ses av mig. Då kommer även filmen att förstöras när all data är uthämtad.

Pappersinstruktioner

Instruktionsmanualen visar en stegvis guide för hur ett 3-D-pussel med tråklossar ska läggas. Den stora siffran visar ordningsföljden på instruktionerna, den orangea teckningen visar formen på den aktuella klossen och bilden visar hur klossen ska läggas. Det är alltid bara en bit per steg. Notera att ordning och orientering av bitarna är viktig.

Oculus-instruktioner

När vi ska kalibrera utrustningen kommer du få ta på dig detta headsetet. De två kamerorna på utsidan kommer att se åt dig och visa vad de ser på de två skärmarna inuti headsetet. Den högra kameran har även en mönsterigenkänningsfunktion. När den ser bilderna i uppgiften projicerar den information som ska guida dig genom uppgiften. Den stora bilden visar med grå genomskinlig färg hur hela pusslet ska se ut. Den visar även med en klargrön färg var nästa pusselbit ska läggas. När en pusselbit har lagts rätt och kameran sett detta kommer informationen från den stora bilden uppdateras. Lagda pusselbitar markeras med en mörkgrön färg. Pusselbitarna har en bild var som hjälper kameran att känna igen dem. När kameran ser rätt pusselbits bild kommer information att projiceras på pusselbiten, en grön markering av biten och en pil som visar rätt orientering. Pilen ska peka från dig snett ned på pusselbiten om den är rätt orienterad. Placera då pusselbiten där den stora bilden anger att den ska vara. Siffror på bitarna ska ha samma orientering som på den stora bildens projicering. Om pusslet inte uppdateras trots att du är säker på att du har lagt rätt kan det bero på att kameran inte registrerat att du har lagt biten rätt. Den högra kameran måste se både den stora bilden och den lilla bilden på pusselbiten samtidigt för att se att de är på samma ställe. Det är alltid bara en bit per steg. Notera att ordning och orientering av bitarna är viktig.

Den engelska versionen:

General information

The purpose of this experiment is to investigate the efficiency of different assembly-instructions. Your participation is completely voluntary and can be aborted whenever you want without motivation. Data will be collected but will not be connected to you as a person. Your participation will be filmed with the purpose to collect data. If you feel uncomfortable with being filmed you only need to ask for it and the recording will only be watched by me. Then the film will be destroyed once all data is collected.

Paper-instructions

The instruction-manual shows a stepwise guide for how a 3D-puzzle with wooden blocks should be placed. The big number shows the order of the instructions, the orange sketch shows the shape on the current block and the picture shows how the block should be placed. There is always only one piece per step. Note that order and orientation of the pieces are important.

Oculus-instructions

When we shall calibrate the equipment you will get to wear this headset. The two cameras on the outside will see for you and show you what they see on the two screens inside the headset. The right camera also has a pattern-recognition-functionality. When it sees the pictures in the assignment it will project information that will guide you through the assignment. The big picture shows with gray transparent color how the entire puzzle will look. It also shows with a clear green color where the next piece of the puzzle is to be laid. When a piece of the puzzle is laid correctly and the camera has seen this the information from the big picture will be updated. Puzzle-pieces already placed are marked with a dark green color. The puzzle-pieces have a picture each that helps the camera to recognize them. When the camera sees the right puzzle-piece's picture, information will be projected on the puzzle-piece in the form of a green marking of the piece and an arrow that show the correct orientation. The arrow should point away from you down to the piece if it is correctly oriented. Then place the puzzle-piece where the big picture indicates that it should be. Numbers on the pieces should have the same orientation as on the big picture's projection. If the puzzle isn't updated even though you are sure that you have placed it correctly it can be due to the camera not having registered that you have placed the piece correctly. The right camera must see both the big picture and the little picture on the puzzle-piece at the same time to see that they are on same place. There is always only one piece per step. Note that order and orientation of the pieces are important.

Följande tre sidor visar hur pappersinstruktionerna såg ut. Oculusinstruktionerna går inte att återskapa på ett bra sätt i tryckt form men har beskrivits i rapporten.

1

1_1



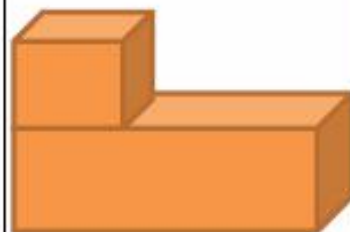
2

1_2



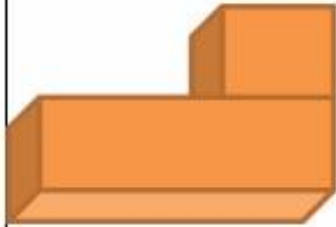
3

1_3



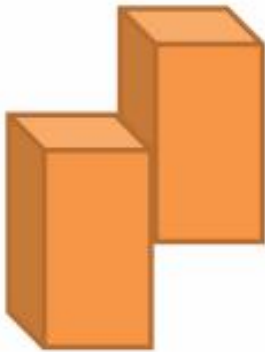
4

2_1



5

2_2



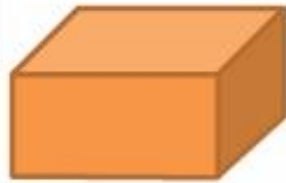
6

2_3



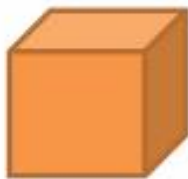
7

3_1



8

3_2



9

3_3

