

Prosjektrapport i Anvendt Digital Arbeidsflyt -
Enkel visning av 3D-modeller i et AR-miljø

Øyvind Larsen Nygård

2. desember 2009

Innhold

| | | |
|----------|---|-----------|
| 1 | Sammendrag | 2 |
| 2 | Introduksjon | 3 |
| 2.1 | Problemstilling | 3 |
| 3 | Implementasjon | 5 |
| 3.1 | Lisensiering | 7 |
| 4 | Mål/Resultat | 8 |
| 4.1 | Plan/Milepæler | 9 |
| 4.2 | Hva kunne vært gjort annerledes | 9 |
| 4.3 | Muligheter for utvidelse | 9 |
| 4.4 | Hva jeg har lært av prosjektet | 10 |
| 5 | Merknader | 12 |

Kapittel 1

Sammendrag

Denne rapporten beskriver et prosjekt i Anvendt Digital Arbeidsflyt. Prosjektet har vært å utvikle en applikasjon/arbeidsflyt for å vise 3D-modeller i et Augmented Reality miljø. Prosjektet har blitt gjort i samarbeid med David Aasteruds masteroppgave hvor han blant annet ser på å bruke AR i sammenheng med læring.

Kapittel 2

Introduksjon

Dette er et prosjekt gjort som en del av kurset *imt4951 Applied Digital Workflow*[1]. Oppgaven har gikk ut på å lage en del av en arbeidsflyt for å hente, og vise 3D-modeller i et *AR (Augmented Reality, også kalt Utvidet Virkelighet på norsk)*[2] miljø, i en læringssammenheng. Augmented reality er teknologi som kombinerer virtuell data, med data fra den virkelige verden, i denne sammenheng 3D-modeller oppå en video-bakgrunn, posisjonert, rotert og skalert til å passe inn i videoen. Dette gjøres ved hjelp av *markører*; mønster vanligvis skrevet ut på papir, som kan gjenkjennes av et program/bibliotek. Utifra rotasjon/skala/vrengning på markøren som blir gjenkent kan det beregnes hvordan den sannsynligvis er posisjonert i virkeligheten. I denne sammenheng blir det brukt til en enkel visning av 3D-modeller. Dette ble gjort i sammenheng med David Aasterud sin masteroppgave. David skal blant annet se på muligheter for bruk av AR-teknologi for læring i grunnskolen. I dette tilfellet historieundervisning.

Masteroppgaven er utført med Simon McCallum som veileder, som har sett en del på bruk av AR-teknologi tidligere. Simon McCallum var også den som kom opp idéen til dette prosjektet.

2.1 Problemstilling

Arbeidsflyten som var planlagt i utgangspunktet, og som ble beskrevet i forprosjektplanen, gikk ut på at en lærer/annen bruker skulle kunne hente en 3D-modell fra for eksempel Google Earth/Google 3D warehouse[3], eller terrengdata og modeller fra andre kilder, og la elever/andre kunne se på den ved hjelp av en applikasjon som viser den i et AR-miljø. Hovedmålet med prosjektet var å utforme, og lage denne arbeidsflyten. Planen var også å utvikle modellvisningsapplikasjonen selv. Dette fordi David og Simon hadde planer om å etter hvert å kunne prøve ut noen funksjoner som ikke eksisterte i noen lett tilgjengelige applikasjoner allerede. Blant annet å bruke en kube med 6 markører på, slik at modellene kan roteres i alle retninger, og å ved hjelp av kuben også kunne

zoome, skalere og panorere modellen. Målet for applikasjonen i dette prosjektet var i utgangspunktet bare å lage en helt enkel applikasjon som kunne vise en modell ved hjelp av AR. I tillegg var det også å finne verktøy for å kunne sjekke om modellene var i rett format, og konvertere de hvis de ikke var det. Modeller fra *Google 3D warehouse* er tilgjengelige som *.skp*-filer som er et lukket format som brukes av *Google Sketchup*, og som *.kmz* og *.zip* filer som inneholder en *KML*-fil, og en modell i det åpne formatet *COLLADA*[4] og teksturene brukt på modellen. Disse modellene kan blant annet åpnes i Google Earth, der modellen vil vises på kartet på plassen KML-filen beskriver.

Filstrukturen er laget for Google Earth og det ville derfor kunne være nødvendig med et skript for å pakke ut, og eventuelt konvertere til et format som var støttet i modellviseren. *OpenSceneGraph*[5] som er et ab bibliotekene som er brukt i AR-modellviseren, støtter lastning av *COLLADA*-modeller med et “plugin”.

For andre formater, er *Meshlab*[6] er et verktøy som kan brukes til å konvertere modellen i kommandolinja, som kan håndtere noen formater som *OpenSceneGraph*, ikke støtter.

Kapittel 3

Implementasjon

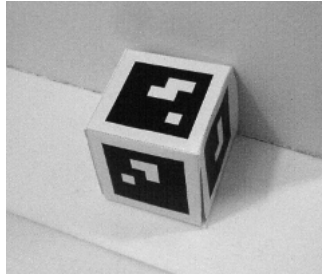
Modellvisningsapplikasjon C++ ved hjelp av åpne biblioteker:

- *ARtoolkit*[7] er et bibliotek for utvikling av AR-applikasjoner, som tar seg av tracking og integrasjon med *OpenGL*, og gir et enkelt grensesnitt for å bruke dette.
- *OpenSceneGraph*[5] er et bibliotek for utvikling av 3D-applikasjoner, som blant annet inneholder en scenegraf, og et API for lasting av modeller.
- *OSGART*[8] er et bibliotek som integrerer ARtoolkit i OpenSceneGraph.

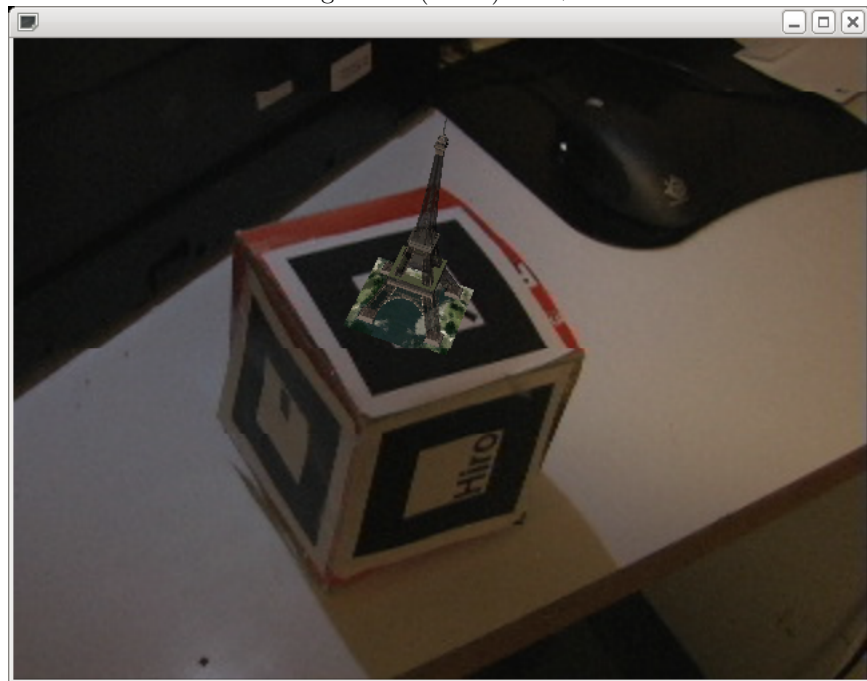
Den er fortsatt på et ganske tidlig stadie, og har begrenset funksjonalitet. Den laster inn 3D-modellen som blir oppgitt som argument til programmet, og ved hjelp av et kamera viser den videoen fra kameraet, og den oppgitte modellen, orientert oppå markøren hvis den er synlig for kameraet (og ARToolkit klarer å oppfatte den.) Planen var å lage en framviser som kunne utnytte en kube med markører. Modellviseren håndterer for øyeblikket 3 markører og vil bruke markøren ARToolkit er mest sikker på er rett for å posisjonere modellen oppå kuben. (Posisjoneringen vil ikke være helt rett, siden posisjoneringen er ikke så veldig polert, og kuben/kameraet/deteksjonen vil aldri være nær perfekt, på grunn av støy, vregning og lignende.)

Mye av tiden har gått med til å implementere modellviseren. Det tok opp mer tid en antatt, da David Aasterud og Simon McCallum var mest interessert i modellviseren, og ikke fokuserte så mye på hvordan modellene skal skaffes og tilrettelegges i denne omgang.

Jeg har også laget to skript for å åpne modellene fra Google Earth da de kommer som en pakket fil som programmet ikke kan åpne direkte. Et Bash-skript for UNIX-systemer (*unzipconv.sh*), som bruker *unzip*-kommandoen til å pakke ut filen, og et Windows batch-skript (*loadModel.bat*), som ved hjelp av det åpne verktøyet *7-zip*[9] utfører den samme operasjonen i Windows. De fungerer ved å pakke filen som blir oppgitt som argument til skriptene til en midlertidig mappe, og åpner modellviseren med modellen, og sletter de midlertidige



Figur 3.1: (Kube)markør



Figur 3.2: Skjermskudd fra modellviseren

modellene når modellviseren blir avsluttet. Å sende filnavnet som argument kan vanligvis også gjøres ved å dra og slippe filen på skriptet i GUI-miljøer, for eksempel på Windows-systemer. I forprosjektplanen var det foreslått å bruke *Python* for å lage eventuelle skript. Dette fordi det er kryss-plattform, mens hvis en skulle bruke *bash* i Windows, ville en måtte kjøre *Cygwin* eller lignende, som kan være litt mye arbeid å sette opp i forhold til behovet. Siden det ikke endte opp med å bli mye skripting, ble det enklere å lage to skript, enn å skrive et i et kraftigere, men litt mer krevende språk som *Python*. (I den sammenheng at *Bash* (og *Windows Batch* skript) stort sett benytter seg av kommandolinjeprogrammer/skript for å utføre oppgavene, men i tilfellet *Python* skulle vært brukt ville det bli å benytte standardbibliotekene for å utføre ting som utpakkings.) Den mest effektive løsningen for åpningen av filene, ville vært å programmert denne funksjonaliteten inn i applikasjonen. Det ville dog ha krevd mye mer arbeid å gjøre på en god måte. (Dette kunne vært å utvidet *COLLADA*-pluginet/lage et ekstra plugin, som ville vært veldig langt utenfor rammene i dette prosjektet.)

Det er også laget en enkel web-presentasjon i henhold til oppgaven[10].

3.1 Lisensiering

Alt innhold på nettsiden, og skriptene, vil bli lagt ut med *CC-NC-BY-SA*[11] lisensen. Formålet med *Creative Commons* er å bidra til spredning av arbeider som folk står fritt til å bruke og arbeide videre på. *Creative Commons* og kopibeskyttelse var en sentral del av det innledende faget.

Å legge ut resultater med åpne lisenser, med kildekode, og å benytte åpne formater er også viktig for framtidig bruk av slike ressurser. Det sikrer at det er lesbart selv langt fram i tid. Modellviseren vil bli lagt ut under *GNU General Public License v3 (GPL)*[12]. *ARToolkit* og *OSGART* også er lisensiert under *GPL*. *GPL*-lisensen krever at verk som bygger på et *GPL*-lisensiert verk, også må publiseres under samme vilkår.¹ *GNU General Public License* er en lisens utviklet av *Free Software Foundation*, og formålet med den var å fremme bruk av åpen kildekode. En businessmodell som har blitt mer vanlig de siste årene er å lisensiere kommersiell programvare under *GPL* eller lignende lisenser, og ofte i tillegg en kommersiell lisens. Dermed kan en bruke bruke programvaren fritt, og produsenten vil få inn penger på konsulent/support og lignende i stedet. I dette tilfellet, med biblioteker kan de brukes så lenge programvaren som benytter seg av de slippes på samme vilkår, hvis det skal brukes til en kommersiell applikasjon har en muligheten til å kjøpe en lisens og dermed slippe å gi ut kildekode.

¹Dette gjelder også dynamisk linking mot *GPL*-lisensierte biblioteker.

Kapittel 4

Mål/Resultat

I forprosjektrapporten var det listet opp mål for prosjektet. Første mål var å ende opp med en arbeidsflyt som dekket det å kunne hente ut modeller, få dem i riktig format, og vise dem i et AR-miljø. Modellviseren er funksjonell nok til å vise formålet med prosjektet. Den kan også åpne modeller direkte fra Google 3D Warehouse, ved hjelp av skriptet som pakker dem ut.

Det var også et mål at verktøyene skulle være kryss-plattform. Modellviseren kjører i både Windows, og Linux, og burde kompilere og kjøre i OS X også, men jeg har ikke hatt mulighet til å teste det. Bash-skriptet for å pakke ut modellen kjører i Linux, og burde fungere i andre UNIX-miljøer også.

Allerede i forprosjektplanen var det nevnt at det var flere risikoer i prosjektet. Prosjektet ble gjort i sammenheng med en masteroppgave som fortsatt var i planleggingsfasen, og det var derfor en del usikkerhet i hva resultatet kunne bli. Det var også nevnt at implementeringen av modellviseren lett kunne bli vanskeligere en antatt, dette har delvis skjedd da modellviseren ble mer avansert en det som var planlagt i utgangspunktet. Planen var å lage en applikasjon som kunne vise en modell med en markør. David og Simon hadde bruk for et slikt program for et forsøk, så det viktigste for dem var å ha en program som kunne vise modellen med en kubemarkør. Dette gjorde at det tok lengre tid enn beregnet å skrive programmet, og å få det i et brukbart stadie ble derfor prioritert. Dette, og at Simon og David Aa ikke var så opptatt av arbeidsflyten ennå, gjorde også at det ble lagt mindre arbeid i å automatisere arbeidsflyten. Planen var også å samarbeide med David Tverbak, en annen student på samme studiet som også tar faget, om deler av arbeidsflyten. På grunn av dårlig kommunikasjon har det dog ikke blitt noe særlig samarbeid mellom meg og han. Hovedmålet med prosjektet har derfor endret seg fra å utvikle den nevnte arbeidsflyten, til å utvikle et av verktøyene benyttet i arbeidsflyten.

Et mål var også å se på verktøy for eventuell konvertering, hvis det var behov å bruke modeller i formater som ikke var direkte støttet av OpenSceneGraph. *Meshlab*, som ble nevnt tidligere, er et åpent verktøy som kan konvertere mellom en del modellformater, også via kommandolinja.

4.1 Plan/Milepæler

Prosjektplanen og milepælene som ble oppgitt i forprosjektrapporten ble fulgt, bortsett fra at webpresentasjonen ble startet tidligere enn nevnt, og Automatisering/skripting tok mindre tid enn beregnet, mens implementasjon av modellviseren tok mer.

| Navn | Arbeid | Uke 46, 2009 | | | | | Uke 47, 2009 | | | | | Uke 48, 2009 | | | | | Uke 49, 2009 | | | | | | | | | | | | | |
|--|--------|--------------|----|----|----|----|--------------|----|----|----|----|--------------|----|----|----|----|--------------|----|----|----|----|----|----|---|---|---|--|--|--|--|
| | | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 1 | 2 | 3 | | | | |
| Finne nødvendige verktøy for arbeidsflyten | 7d | [Bar] | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Skaffe testdata | 7d | [Bar] | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Implementere modellviser | 15d | [Bar] | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Sette opp arbeidsflyten | 10d | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Automatisering av arbeidsflyt/Skripting | 10d | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Prosjektrapport | 8d | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Webpresentasjon | 8d | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Testdata fra *Google 3D Warehouse* [3] og på *oyonale.com* [13] ble benyttet under utviklingen.

4.2 Hva kunne vært gjort annerledes

Jeg har kommet fram til at jeg tok på meg mer arbeid enn jeg hadde mulighet til å få gjennomført med et godt resultat. Det har endt med at mer tid enn nødvendig gikk med til å fullføre modellvisningsapplikasjonen. En mulighet ville ha vært å kutte ut å implementere kubemarkøren, og heller ha konsentrert arbeidet rundt å skape en mer fullstendig arbeidsflyt. Da ville det ha vært mer tid til å kunne se på å utføre litt av det som er nevnt i neste avsnitt. Resultatet har blitt at prosjektet har kommet litt på kanten av kravene som ble spesifisert for oppgavene. Etter å ha startet på prosjektet, kom jeg ganske fort fram til at å velge en av de forhåndsforeslåtte oppgavene nok ville ha vært lettere gjennomførbart, passe bedre med målene i faget, og kanskje mer givende karaktermessig, enn å jobbe med oppgaven som jeg har gjort.

4.3 Muligheter for utvidelse

Selve arbeidsflyten har en del muligheter for utvidelse. Dette prosjektet har hovedsakelig blitt konsentrert rundt framvisningen av modellene. I forprosjektplanen var det nevnt et eksempel på en applikasjon, *ARSights* [14] som hadde noe av samme formålet, å vise modeller av konstruksjoner fra virkeligheten i et AR-miljø. ARSights er laget med formålet å vise modeller fra Google Earth ved hjelp av AR. På nettsiden er det brukt Google Earth sitt nettleter-API [15] og lagt til linker til de tilgjengelige modellene i en Google Earth-visning. Nettleser-APIet er et Javascript-API som gjør det mulig å benytte nettleter-utgaven av Google

Earth gratis på nettsider (med en del betingelser), og å tilpasse den til eget behov. (Det finnes også andre/åpne alternativer[16]) I den tenkte brukssituasjonen til arbeidsflyten som ble jobbet med i dette prosjektet, ville en lignende løsning gjøre kunne gjøre arbeidet lettere for folk som er mindre datakyndige, og det kan være mer “visuelt tiltrekkende”. Modellviseren kan bli brukt som basis for modellvisningsappkulasjonen i David Aa sitt masterprosjekt. Den kan også lett utvides med mye mer av den tidligere nevnte funksjonaliteten.

4.4 Hva jeg har lært av prosjektet

Kjell-Are Refsvik, faglæreren, har nevnt flere ganger at prosjektene i dette faget er en slags form for “mini-mini-masteroppgaver”. Det kan trekkes flere paralleller fra utførelsen av disse oppgavene til utførelsen av en masteroppgave. Forskjellene er at det ikke ligger en like stor grad av forskning i prosjektene, samt at valg av oppgave bør ligge innenfor rammene av faget og dermed er mer begrenset. I tillegg er dette et prosjekt på 5 studiepoeng, mens masteroppgaven kan være 35 eller mere, og er dermed mye mindre. Dette prosjektet har derfor gitt meg mer innsikt i hvordan gjennomføringen av et slikt masterprosjekt kan fungere. Jeg har lært at det er veldig viktig å vurdere omfanget av oppgaven en velger, før en starter å jobbe på den, og regne med at det kan bli en del mer arbeid en forventet. Å ha klare mål

Prosjekter vil normalt sett ikke gå helt som planlagt. Hvis det går altfor perfekt kan det tyde på smøring av resultatene. Dette prosjektet gikk heller ikke helt som planlagt. Jeg føler dog at jeg stort sett har klart å gjennomføre målene jeg satte meg i prosjektplanen, men samtidig har jeg hele veien følt at implementasjonen av prosjektet ikke helt har truffet alle målene med faget. Jeg har dog i rapporten prøvd å diskutere hvordan prosjektet kan kunne utvikles videre i tråd med dette. En viktig del av slike prosjekter, som også er nevnt som en av utfordringene er nettopp å ta hensyn til hvordan resultatene kan brukes videre (selv om de nødvendigvis ikke blir brukt til noe.)

Bibliografi

- [1] Applied digital workflow. <http://www.ansatt.hig.no/kjellr/imt4951/>.
- [2] Augmented reality. http://en.wikipedia.org/wiki/Augmented_reality/.
- [3] Google 3D Warehouse. <http://sketchup.google.com/3dwarehouse/>.
- [4] COLLADA format. <http://www.khronos.org/collada/>.
- [5] OpenSceneGraph. <http://www.openscenegraph.org/projects/osg>.
- [6] Meshlab. <http://meshlab.sourceforge.net/>.
- [7] ARtoolkit. <http://www.hitl.washington.edu/artoolkit/>.
- [8] OSGART - ARtoolKit for OpenSceneGraph. <http://www.artoolworks.com/community/osgart/>.
- [9] 7-zip. <http://www.7-zip.org/>.
- [10] Web-presentasjon av prosjektet. <http://www.stud.hig.no/~091210/>.
- [11] Creative commons navngivelse-ikkekommersiell-del på samme vilkår 3.0 unported lisens. <http://creativecommons.org/licenses/by-nc-sa/3.0/deed.no>.
- [12] GNU General Public License v3. <http://www.gnu.org/licenses/gpl.html>.
- [13] Oyonale. <http://www.oyonale.com>.
- [14] ARsights. <http://www.arsights.com/>.
- [15] Google Earth API. <http://meshlab.sourceforge.net/>.
- [16] Online virtual globes. http://en.wikipedia.org/wiki/Virtual_globe#Online_virtual_globes.

Kapittel 5

Merknader

Jeg mistet en del tid den siste uka pga. helserelaterte problemer, men valgte å ikke be om utsettelse siden jeg uansett har 3 andre prosjekter som også må fullføres etter dette, og følte jeg ikke hadde ville hatt så mye utbytte av det.

Alt arbeid (bortsett fra kompillering i windows) er gjort med åpne verktøy:

- NetBeans og GCC for utvikling av modellviser.
- GNU Emacs for skript/nettside.
- L^AT_EX for rapport. (Krevd av oppgaven)
- GIMP for illustrasjoner.

Det meste av arbeidet er gjort i Ubuntu Linux.

Takk til Simon McCallum og David Aasterud for oppgaveforslag og støtte.

Filer

- loadModel.* - modellastningsskript
- model_viewer_src.tar.bz2 - kildekode for modellviser

En utgave av modellviseren kompillert for windows, og en zip-fil med nødvendige dll-filer ligger på hjemmesiden. (Har valgt å ikke legge disse med i prosjektbesvarelsen siden disse filene (bortsett fra modellviseren) ikke er jeg som har produsert.)

Denne rapporten er lisensiert under en [Creative commons Navngivelse-Ikkekommersiell-Del på samme vilkår 3.0 Unported lisens](#).