

3D IR Motion Tracking

Fontys Technische Informatica
T7 Embedded Linux

20 september 2008

T7 Embedded Linux: Bert Tilmans, Joris van Rooij, Niels Keeman, Ruud Jansen, Ruud van Heugten.
Aan dit document kunnen geen rechten ontleend worden. Vrijgegeven onder de CCPL BY-SA 3.0 licentie.

Inhoudsopgave

1	Inleiding	3
2	Technologie	3
2.1	Theorie	3
2.2	Praktijk	3
3	Toepassing	5
3.1	Positioning	5
3.2	Interfacing	6
4	Development Traject	8
4.1	Design	8
4.2	IR Input (evdev)	8
4.3	MPX	9
4.4	2D - 3D Conversion	9
4.5	3D Motion Protocol	9
4.6	Aansturing Robotics	9
5	Conclusie	10

1 Inleiding

Dit document is opgesteld als alternatief op het document van 2008-09-08. Het document beschreef een methode om op een innoverende wijze te interfacen met een applicatie, weergegeven op een wandgemonteerd scherm.

Om ruimte te scheppen in het ontwikkelproces is er gekeken naar andere mogelijkheden om hetzelfde doel te bereiken. Hetzelfde principe kan worden uitgevoerd door middel van andere, evengoed vernieuwende, technologie. Dit document beschrijft het resultaat van een zogenaamde *brainstormsessie*.

2 Technologie

2.1 Theorie

Het doel van deze manier van interfacen is het toegankelijk maken van applicatiefunctieiteit door middel van handgebaren. Deze handgebaren dienen op een of andere manier door het systeem herkend en omgezet te worden naar applicatiesturing.

Naast de methode die beschreven is in het voorgaande document is er een andere veelgebruikte methode. Deze methode maakt gebruik van IR¹. Het voordeel van deze methode is dat de herkenningspunten gemakkelijker uit het beeld gefilterd kunnen worden.

Het systeem werkt vrij eenvoudig. Met een IR-camera wordt het beeld van de gebruiker voor het scherm opgenomen. De gebruiker kan door middel van IR LEDs herkenningspunten op zijn lichaam aanbrengen, bijvoorbeeld aan de vingertoppen. De onderlinge posities van de IR LEDs worden vervolgens omgezet naar referentiepunten op een twee dimensionaal vlak. Aan de hand van veranderingen aan de posities kan een beweging in 3 dimensies berekend worden.

2.2 Praktijk

Het beste voorbeeld om aan te tonen dat deze technologie werkt is de Wii spelcomputer van Nintendo. Bij deze opstelling heeft de gebruiker de IR camera in de hand terwijl de IR LEDs op het scherm bevestigd zijn. In principe is dit exact hetzelfde idee, enkel omgedraaid.

Johnny Chung Lee van de universiteit van Pittsburgh heeft enkele proeven uitgevoerd met de Nintendo IR apparatuur. Hij heeft succesvol zonder modificaties de apparatuur kunnen gebruiken als HID². Op zijn website³ is beeldmateriaal te vinden waarin te zien is hoe hij te werk is gegaan.

¹infrarood licht

²Human Interface Device

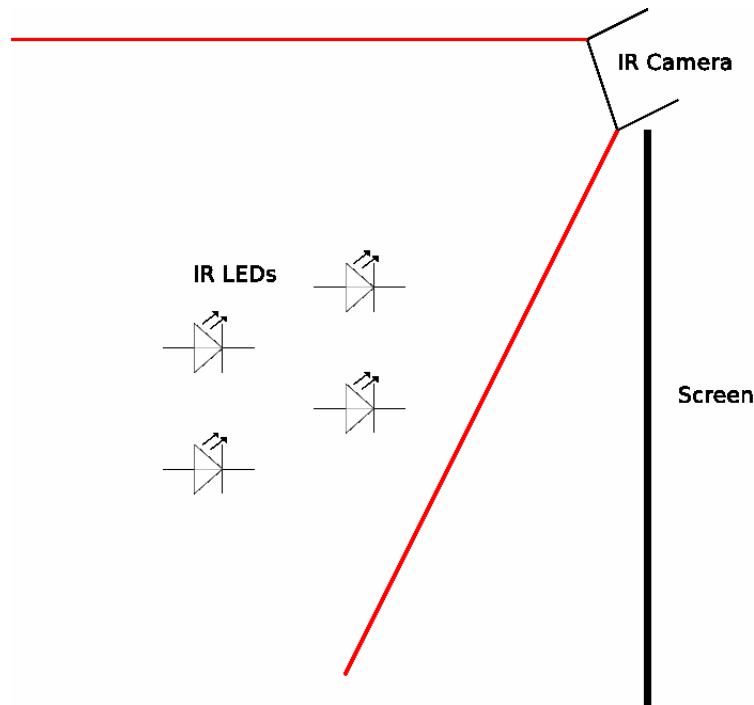
³<http://www.cs.cmu.edu/~johnny/>

Enkele Apple-gebruikers zijn er eerder al in geslaagd ingebouwde iSight webcams te gebruiken als HID door middel van IR LEDs.

3 Toepassing

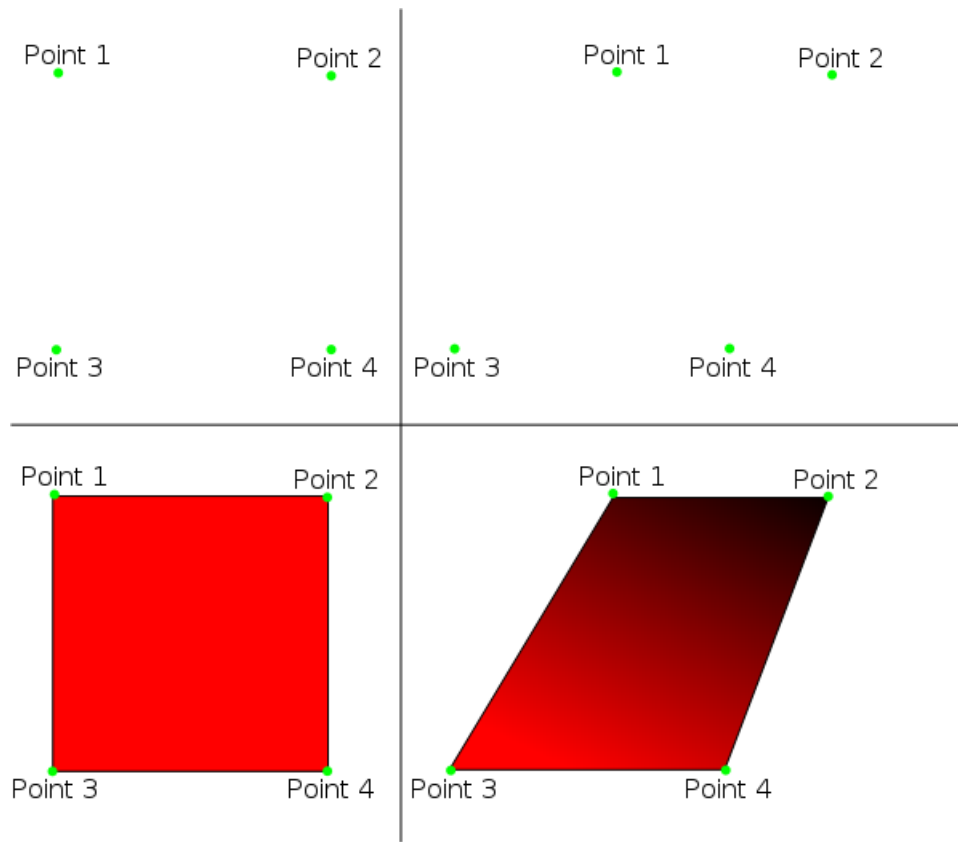
3.1 Positioning

Voordat er ook maar iets kan gebeuren zullen de relatieve coördinaten van de herkenningpunten door het systeem gedetecteerd moeten worden. In dit geval wordt er een camera gebruikt met een vaste positie relatief aan het scherm. Deze camera is enkel gevoelig voor IR, dit maakt het herkennen van de IR LEDs eenvoudiger.



Zoals in de bovenstaande afbeelding duidelijk wordt is de nodige perspectiefcorrectie noodzakelijk voordat de punten in twee dimensies overeenkomen met de benodigde relatieve coördinaten aan het scherm. Het gewenste resultaat is natuurlijk dat wanneer de gebruiker een referentiepunt gelijk brengt met de rechtsbovenhoek van het scherm de interface ook daadwerkelijk reageert in de rechtsbovenhoek. De onderkant van het scherm mag niet minder gevoelig reageren dan de bovenkant.

Vervolgens kan door middel van het volgen van bewegingen van verschillende referentiepunten een 3D-beweging berekend worden. Hiervoor zijn minstens vier punten benodigd. Het is aannemelijk dat deze vier punten aan de duimen en wijsvingers van de gebruiker worden gekoppeld.



Door de beweging van *point 1* en *point 2* te volgen is een kanteling van de rechthoek tussen de vier punten waar te nemen. Natuurlijk is niet alleen een kantelende beweging van de handen resulterend in dergelijke puntveranderingen. Door een duidelijke (real time) feedback te genereren vanuit de interface kunnen overwachte bewegingen door de gebruiker gecorrigeerd worden. Dit effect, gebruik makende van de oog-hand-coördinatie van de gebruiker, is een veel voorkomend verschijnsel bij mens-machine-interactie.

De 3D-beweging kan uitgedrukt worden in roll-, pitch- en yaw-waarden. Deze drie waarden beschrijven de rotatie (in graden) van een bepaald object in respectievelijk de x-, y-, en z-as. In de bovenstaande afbeelding zou de roll-waarde op ongeveer -15 staan, terwijl pitch en yaw 0 blijven.

3.2 Interfacing

Zoals al beschreven is het van belang dat de interface in real-time duidelijke feedback verschaft aan de gebruiker. Aangezien de gebruiker in drie dimensies bewegingen maakt is het zaak om ook de interface in 3D weer te geven.

Wanneer bijvoorbeeld de gebruiker een kubus bestudeert zal de kubus op het scherm aan de hand van de berekende 3D-bewegingen mee moeten draaien.

De interface kan natuurlijk meerdere 3D-objecten bevatten. Aangezien er een veelvoud aan referentiepunten aanwezig is kan er eenvoudig teruggevallen worden op “standaard” multitouch gedrag, waarbij ieder referentiepunt een eigen muisaanwijzer krijgt. Op deze wijze kunnen er binnen een interface mogelijkheden worden geschapen om van 3D-objecten en zelfs applicaties te wisselen door middel van handgebaren.

De gegenereerde 3D-bewegingen kunnen naast gebruikt in een 3D applicatie ook geëxporteerd worden in een protocol, waarna een embedded software-omgeving de yaw-, pitch-, en roll-waarden kan omrekenen naar bijvoorbeeld motorsturing.

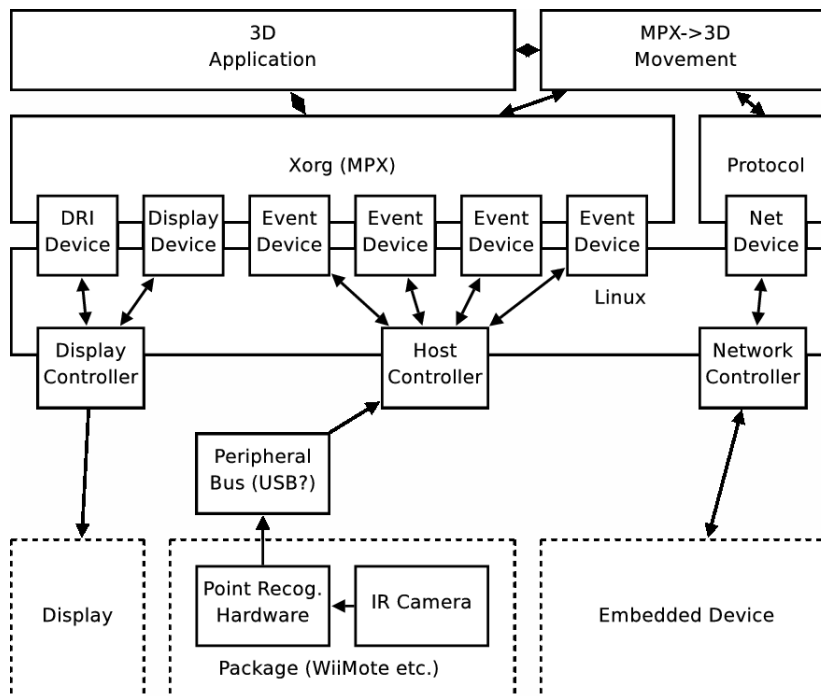
4 Development Traject

Het komende hoofdstuk geeft een opzetje voor een mogelijk development traject voor dit project.

4.1 Design

Allereerst zal een gedetailleerd design opgesteld moeten worden, tezamen met ondersteunend onderzoek.

Een globaal schematisch design van het systeem kan er als volgt uit zien.



4.2 IR Input (evdev)

Er zal een driver geschreven moeten worden die óf de output van de point recognition hardware omzet naar een event per punt, óf zelf IR beelden direct omzet naar events per gevonden punt.

Er zal uitgezocht moeten worden hoe event interfaces werken, hoe het muis-protocol eruit ziet en welke calibraties en correcties er toegepast moeten gaan worden. Er zijn enkele open implementaties beschikbaar, samen met een vrij volledige open documentatie.

4.3 MPX

Multi Pointer X (MPX) is een speciale aanpassing aan de X.org server waardoor er meerdere muisaanwijzers beschikbaar komen voor X11 clients.

Er dient uitgezocht te worden hoe deze afzonderlijke muisaanwijzers te benaderen zijn binnen een enkele client. Daarbij kan globale kennis over X11 natuurlijk niet ontbreken.

4.4 2D - 3D Conversion

Wanneer vier punten beschikbaar komen binnen een applicatie kan er begonnen worden aan het omrekenen van 2D veranderingen naar 3D bewegingen. Hier komt veel goniometrie en triangulatie bij kijken, wellicht is dit een punt waarop wiskundeboeken tevoorschijn kunnen worden gehaald.

Het resultaat worden de welbekende drie waardes van yaw, roll en pitch.

4.5 3D Motion Protocol

De drie waardes die resulteren uit de vorige stap kunnen zowel door applicaties als andere (externe) devices worden gebruikt. Om dit zo toegankelijk mogelijk te maken dient dit protocol licht en simpel inelkaar te zitten. Het is de vraag of positiecoördinaten (x , y , z) ook wenselijk zijn.

Het protocol zal door zowel lokale UNIX sockets als externe IP sockets kunnen worden gevoerd. Ook busverbindingen zijn niet uitgesloten.

4.6 Aansturing Robotics

Op het moment dat het protocol in een werkende staat verkeert kan externe randapparatuur in de vorm van microcontrollers worden aangesloten. Een voorbeeldtoepassing zou bijvoorbeeld een TI OMAP kunnen zijn waarop een embedded RT-Linux omgeving motorsturing verzorgt van een robotarm. Deze robotarm zou dan aan de hand van de handgebaren aangestuurd kunnen worden, dit terwijl een 3D model van deze arm synchroon op het scherm mee beweegt.

5 Conclusie

De IR methode voor het interfacen met een (RT-)Linux-omgeving is een goed uitvoerbaar project. In de praktijk is het door verschillende mensen grotendeels al eens uitgevoerd. Het unieke element wat door dit project wordt toegevoegd is een stuk standardisatie, tezamen met de mogelijkheid handgebaren van de gebruiker naar buiten beschikbaar te maken.

Het voorgestelde traject is naar alle waarschijnlijkheid te lang voor een enkel project. In overleg met de betrokkenen kan een grens en dus een eindproduct worden bepaald.