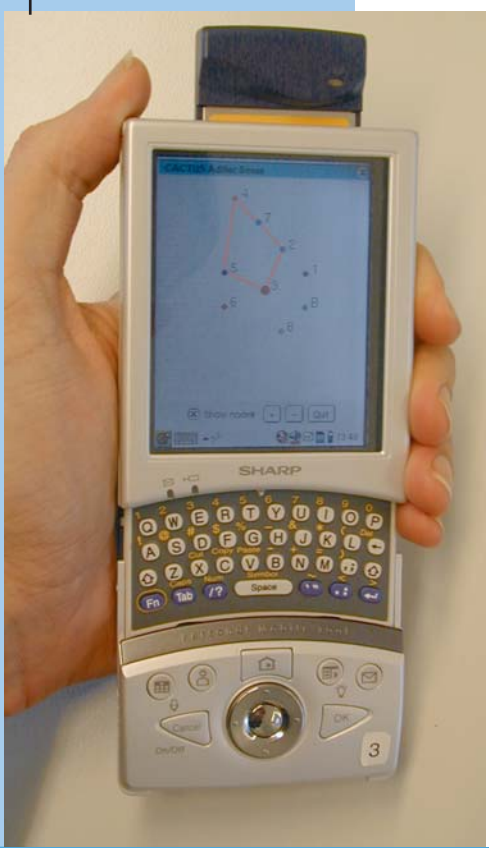
Cactus project

W. Pasman

Binnenkort zullen de apparaten en gebouwen om ons heen vele stukjes software bevatten, die de besturing van apparaten en de interfacing naar de gebruiker verzorgen. Zo bevat de auto al vele tientallen processoren die voor een deel ook door de gebruiker moeten worden aangesproken, bijvoorbeeld als hulp bij navigatie. Op dit moment zijn al die stukjes software nog met draden met elkaar verbonden, maar we verwachten dat dit in toenemende mate via een draadloos netwerk zal gebeuren. Dit kan natuurlijk via de bestaande mobiele telecomnetwerken, maar in vele gevallen zal het wenselijk zijn direct boodschappen over en weer te sturen, zonder tussenkomst van telecom operatoren. Zulke apparaatjes zullen zichzelf grotendeels moeten configureren, want er zijn er veel en de situatie verandert voortdurend. Zulke 'ad-hoc' netwerken kennen echter geen vaste routing en door het beperkte bereik kan alleen over grotere afstand worden gecommuniceerd door het bericht door meerdere tussenliggende apparaten te laten 'hoppen'.



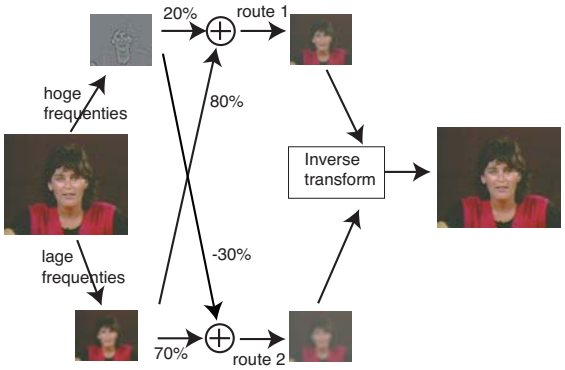
Figuur 1.
Sharp Zaurus met ad hoc multi-pad route. Op dit moment geven de rode lijnen aan dat er twee ad hoc routes vanaf deze machine (nummer 3 in het midden) naar machine nummer 4 zijn: via 2 en 7, of via 5. Machines 1, 6, 8 en B zijn op dit moment onbereikbaar. Dit plaatje is licht gefaked, omdat onze huidige demo slechts 1 route laat zien.

Het CACTUS project ("Context Aware Communication, Terminal and User") doet onderzoek naar zulke ad hoc netwerken en agent systemen. Een ad hoc agent systeem is een toekomstbeeld waarin agents (stukjes min of meer 'zelfstandige' software) rondzwerven door de wereld en proberen hun service aan de man te brengen. Deze agents leveren allerlei typen services, zoals het bedienen van lichtschakelaars, het instellen van de sfeer van de kamer, reisplanning, bemiddeling bij onderhandeling over een gezamenlijke trip, etc.

Op netwerkgebied rijzen er vragen als hoe mobiele computertjes (op de mobiele telefoon, in het horloge, in de auto, etc) elkaar kunnen vinden, en hoe communiceren ze betrouwbaar als de communicatielink op elk moment onderbroken kan worden. Op gebruikersgebied zijn er vragen als hoe vindt de gebruiker de agents die hij nodig heeft, en begrijpt hij het nog als bijvoorbeeld een video ten gevolge van netwerkstoringen wordt teruggeschaald naar een geluidsfragment of als hij plots okay moet zeggen in plaats van op de OK knop drukken. We belichten een paar kernideeën.

Multipad routing

Een leuk voorbeeld om de verschillende netwerk mogelijkheden van een ad hoc systeem uit te buiten is multipad routing. Om van machine 3 naar 4 te komen zijn er vaak meerdere mogelijkheden, bijvoorbeeld via 5 of via 6 en 8. In de tot nu toe bestaande ad hoc



Figuur 2.
Splitsing en reconstructie. Om het eenvoudig te houden wordt het originele beeld hier in maar 2 kanalen gesplitst. De verzonden data zijn 'gemangelde' beelden, omdat de componenten in onnatuurlijke hoeveelheden aanwezig zijn. Na de inverse transformatie komt het originele beeld weer te voorschijn.

netwerk standaards wordt er altijd maar één van de beschikbare paden tussen de machines gebruikt. In het Cactus project wordt gewerkt aan een aangepast protocol, zodat meerdere beschikbare paden tegelijk gebruikt kunnen worden, dus via 5 én via 6 en 8. Figuur 1 laat dit zien op ons testplatform (de Sharp zaurus).

Videostroom robuust splitsen

Een datastroom moet in meerdere stromen gesplitst worden om gebruik te maken van multipad routing. Dit splitsen kan dom gebeuren, door gewoon de helft van de pakketjes die de data bevatten over het ene pad en de andere helft over het andere pad te sturen. Maar als er dan een pakketje uitvalt kan de ontvanger meestal niets met de pakketjes die wel aankwamen, omdat huidige compressietechnieken zeer foutgevoelig zijn. We ontwikkelen binnen Cactus een coder die een videostroom zo kan splitsen dat alleen de beeldkwaliteit vermindert, maar voorkomt dat de video helemaal stokt.

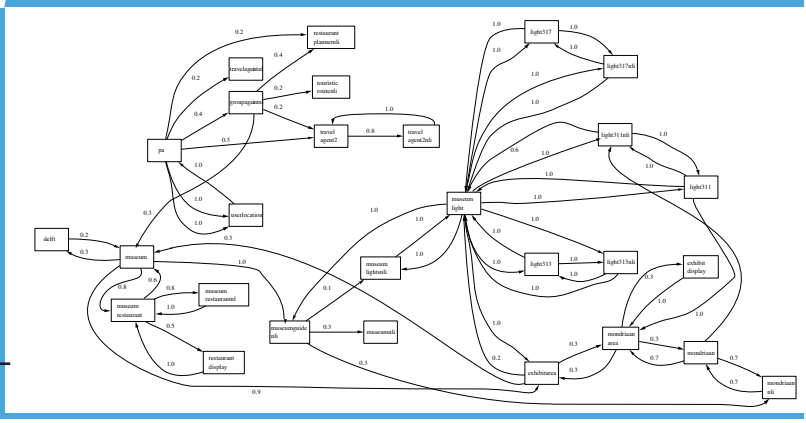
Het splitsen van de videostroom gaat ongeveer als volgt. De beelden worden -zoals gewoonlijk- opgesplitst in laagfrequente data (zeg maar geblurde versie van het plaatje) en hoogfrequente data (de contouren in het beeld). Deze splitsing zorgt voor een aantal kleinere beelden, die tezamen het grote origineel representeren. De laagfrequente data wordt belangrijker beschouwd voor het totaalbeeld, en we geven daarom laagfrequentere data 'zwaarder' mee in elke stroom dan de hoogfrequentere data. Als alle data binnenkomt kan alle oorspronkelijke informatie worden terug gerekend en komt het originele beeld weer tevoorschijn. Figuur 2 schetst het proces.

Als een stroom uitvalt hebben we relatief veel over van de laagfrequente stroom, ten koste van meer verlies in de hoogfrequente stroom. In dat geval moet met kennis over de ingebrachte correlaties tussen de verzonden stromen een schatting gemaakt worden van de ontbrekende stroom, waarna de inverse transformatie gedaan kan worden.

Ad-hoc agents organiseren

Om voor de gebruiker relevante services betrouwbaar te kunnen vinden in een ad hoc systeem is het nodig om de kennis over services en hun relaties te distribueren over de agents. In het Cactus systeem heeft elke agent kennis over agents die gaan over de ruimtes waar hij in zit en die hij bevat, over taak gerelateerde agents en over gebruiker-gerelateerde informatie. Een koffie automaat agent heeft bijvoorbeeld referenties naar de keukenagent, andere koffiemachine agents in de buurt, een cola-automaat agent, de koffievoorraad agent en de wateraansluiting agent. De keukenagent weet dat hij een gebied van een paar meter bestrijkt. De personal agent weet van gebruikelijke taken als de agenda agent en de gebruikerslocatie agent (Figuur 3, volgende bladzijde).

Al deze relatie informatie kan worden gebruikt door context-afhankelijke services. Een zo'n voorbeeld service is het laten meehoppen van windows met de gebruiker, bijvoorbeeld als de gebruiker naar de keuken gaat worden relevante



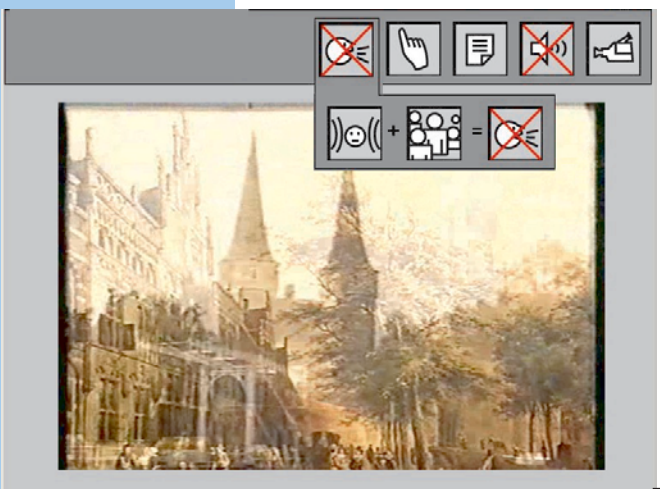
Figuur 3.
Kleine testsituatie voor agents en hun relaties. Zoeken start bij de personal agent (PA). De zoekruimte kan met bepaalde kans (het nummer bij een edge) worden uitgebreid naar een gerelateerde agent.

windows meegeschoven naar het display in de keuken. Een ander voorbeeld is de 'servicematcher' service, die op een context-afhankelijke manier naar agents kan zoeken die een gebruikersverzoek begrijpen. Zo kan de gebruiker vragen om 'licht', en afhankelijk van zijn locatie daarmee een kamerlamp of straatlantaarn bedienen. Of een verzoek 'naar Amsterdam' kan resulteren in een keuzemenu tussen 9292ov en ns.nl als hij thuis is, terwijl een compleet routeplan zou verschijnen als hij al bij een tramhalte staat. Als er wat agents uitvallen blijft het netwerk grotendeels intact, en kunnen de meeste services gewoon doorwerken.

Consistentie en vertrouwen

We deden een experiment om te zien hoe veranderende interface modaliteit de gebruiker beïnvloedt. We verwachtten dat de gebruiker tijdens gebruik verwachtingen ('mentaal model') opbouwt van hoe het systeem zich gedraagt, en dat hij het systeem meer vertrouwt naarmate het systeem beter past bij zijn verwachtingen. Om dit te testen maakten we een multimodale videospeler (Figuur 4). Het systeem liet een aantal video's zien in verschillende omstandigheden, bijvoorbeeld in een te heldere omgeving (buiten) werd er teruggeschakeld naar alleen geluid, en bij teveel lawaai of bij te laag batterijniveau verschenen er alleen ondertitels. De helft van de proefpersonen kreeg uitleg over de situatie in de vorm van iconen (rechtsbovenin Figuur 4).
Inderdaad hebben de mensen die uitleg krijgen meer vertrouwen in het systeem, en ze blijken het systeem ook beter te begrijpen. Maar verrassend genoeg kunnen zij na het experiment slechter voorspellen wat een systeem in een bepaalde situatie zal doen dan mensen die geen uitleg kregen. De mentaal model hypothese heeft wat aanpassing nodig!

Het Cactus project wordt voor de helft gefinancierd door Freeband (www.freeband.nl), en voor de andere helft door de deelnemers Technische Universiteit Delft en TNO. Meer over het Cactus project vindt U op www.cactus.tudelft.nl.



Figuur 4. Gebruikers kregen o.a. een video over Delft te zien. Rechtsbovenin uitleg waarom het systeem bepaalde modaliteiten gebruikt. Hier zien we bijvoorbeeld dat er geen geluid en spraakherkenning beschikbaar is vanwege een lawaaierige omgeving.