

# Undringsdokument

## Mobiltelefonen som strekodeskanner

Dette dokumentet omhandler preliminære tanker og forsøk rundt bruk av en moderne mobiltelefon med kamera som strekkodeskanner. Dokumentet ser på tekniske forutsetninger og begrensninger – og lister opp noen av mulighetene en slik mobiltelefonapplikasjon vil kunne gi.

### Innhold

Undringsdokument	
Mobiltelefonen som strekodeskanner.....	1
Innledning.....	1
Vår idé.....	2
Prior Art - “The Cue Cat”.....	2
Vår visjon .....	3
Teknisk bakgrunn.....	3
Strekker.....	3
Mobiltelefonkameraet.....	4
Et praktisk problem - og en løsning.....	4
Praktisk testing av konseptet.....	5
Implementasjon.....	7
Veien videre, muligheter.....	8
Fremdriftsplan for prosjektet.....	9

---

### Innledning

Moderne mobiltelefoner har i tillegg til heller opplagte egenskaper - som å kunne ringe og sende tekstmeldinger - også gjerne mer avanserte muligheter som for eksempel mulighet til å surfe på internett, høre på radio og forhåndslagret musikk<sup>1</sup>, ta bilder med innebygget kamera og så videre. Mer avanserte mobiltelefoner har i tillegg gjerne omfattende PDA-egenskaper, avanserte og kapable operativsystemer og relativt rask maskinvare. Moderne mobiltelefoner kan således sees på som en av de fremste eksponentene for den mye omtalte “digitale konvergens”, der digital teknologi bidrar til å gi en enkelt enhet muligheter som tidligere krevde mange separate enheter.

Det som derimot kan synes å mangle i denne sammenhengen er synergi. Med enkelte mer eller mindre opplagte unntak virker det som om sammensmeltingen

---

<sup>1</sup> Gjerne mer kjent som MP3-filer.

av flere forskjellige enheter til én enkelt enhet kun har bidratt til å gi brukerne færre enheter å forholde seg til, heller enn å faktisk tilby ny funksjonalitet som bygger på denne konvergensen.

## Vår idé

Vår idé er å gi mobiltelefonen mulighet til å fungere som en strekkodeskanner – hvilket vil si å gjøre en mobiltelefon med kamera i stand til å lese og dekode strekkoder av typen man finner på vanlige kommersielt tilgjengelige varer. Informasjonen i strekkoden er så tenkt sendt til en server<sup>1</sup> som så leverer tilbake en WAP-side eller lignende med relevant informasjon.

Dette vil involvere bruk av mobiltelefonens kamera, dens datakraft og muligheter for å kjøre tredjeparts applikasjoner - og dens kommunikasjonsmuligheter med omverdenen. Akkurat hva applikasjonen som utvikles konkret vil gjøre er ikke helt fastlagt, men vi har listet opp en rekke idéer under overskriften “Veien videre, muligheter”.

## Prior Art - “The Cue Cat”

“The Cue Cat” var en enhet ment for tilkobling til en datamaskin som kunne lese strekkoder fra reklamer i blader og lignende, og sende brukeren til en tilhørende internettside. Dette produktet ble gitt bort (gratis) av et firma med det treffende navnet “Digital Convergence” i USA noen år tilbake. I tillegg til å kunne lese strekkoder fra spesiallagde annonser i aviser og magasiner, ga det også brukerne direkte aksess til internettsiden til et produkt eller en produsent gjennom å skanne inn produktets strekkode.

“The Cue Cat” var kontroversiell fordi firmaet bak hadde en mer eller mindre skjult agenda; å samle demografisk informasjon om brukerne i markedsføringsøyemed, noe som sannsynligvis var med på å bidra til at firmaet bak dette produktet gikk under i 2002<sup>2</sup>.

Visjonen til dette firmaet var for øvrig også interessant og relevant for dette prosjektet: *“The dream was to connect items in the physical world to the Internet, automatically”*.

---

<sup>1</sup> Slik kommunikasjon kan for eksempel gå via GPRS.

<sup>2</sup> Mye kan også tyde på at selskapet hadde en noe “dotcom-isk” forretningsmodell, som fritt etter South Park kan parodiseres omtrent som: 1. Gi bort strekkodelesere 2. ??? 3. Profitt!



*“The Cue Cat”*

## Vår visjon

Vi skal ikke påberope oss mer originalitet enn at vi ærlig innrømmer at vi deler visjonen til menneskene bak “The Cue Cat” - altså hvordan man finner en vei over “gapet” mellom fysiske objekter og informasjon/kunnskap som måtte eksistere om objektet. Forskjellene - og ikke minst fordelene - mellom “The Cue Cat” og vår idé er naturligvis at vi stort sett ikke forutsetter bruk av dedikert maskinvare (se lenger ned i dokumentet for detaljer) og at det her naturligvis vil introduseres et mobilitetsaspekt som følge av at man her bruker en mobiltelefon, og ikke en datamaskin.

---

## Teknisk bakgrunn

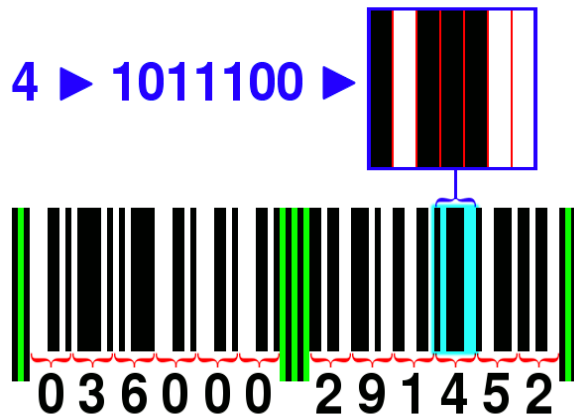
### Strekkoder

Strekkoder er tradisjonelt konstruert med tanke på robusthet og for å kunne brukes med lite kostbar og ukomplisert maskinvare. Systemet som brukes på så å i alle kommersielt tilgjengelige produkter i vesten i dag kalles UPC/EAN, og består av linjer som forekommer i fire tykkelser og i to alternerende intensiteter<sup>1</sup>.

Hvert siffer er representert med fire streker, og den totale bredden for strekene som tilhørende et bestemt siffer er syv. En strekkode i dette systemet enkoder tolv siffer, noe som gir en total bredde for strekkoden på  $7 * 12 = 84$ , pluss tre “synkroniseringsmønster” i start, midt og slutt med en samlet bredde på 11 – ergo vil den totale bredden av streker være 95.

---

<sup>1</sup> Gjerne svart/hvitt for å maksimere kontrast.



*En figur som illustrerer en strekkode og hvordan ett siffer i strekkoden er enkodet (fra wikipedia.org).*

## Mobiltelefonkameraet

Et viktig moment i denne sammenhengen er mobiltelefonens kamera (heretter refert til som “mobilkamera”), og dets begrensninger. Mobilkameraer i dag har som regel en oppløsning som varierer fra cirka 0.3 megapiksel til cirka 1.3 megapiksel. Kvaliteten på bildene som en slik mobiltelefon er i stand til å ta er varierende, men vil på en subjektiv skala som regel havne et sted mellom “elendig” og “akseptabel”.

Det finnes to hovedårsaker til begrensningene hos dagens mobilkameraer: Optikk og bildebrikke. Optikken lider under kostnadshensyn og plasshensyn, og er derfor uten unntak i dag<sup>1</sup> basert på billige fiks-fokus linser – og er dermed blottet for en mulighet man regner som selvfølgelig på vanlige kameraer - muligheten til å fokusere på et vilkårlig objekt, uavhengig av avstand. Bildebrikken lider også under tilsvarende hensyn, og kvalitet er derfor tilsynelatende ikke noe produsenter av mobiltelefoner med kamera “fokuserer” på.

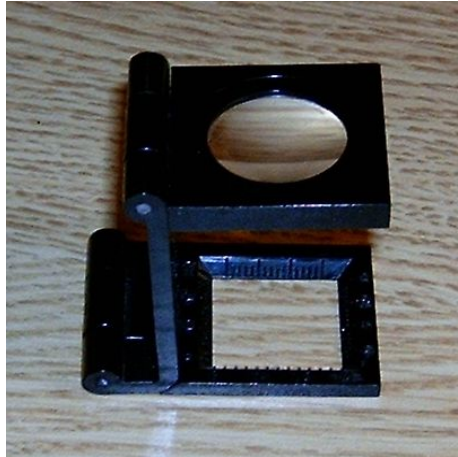
### Et praktisk problem - og en løsning

I sammenheng med dette prosjektet vil det faktum at mobilkameraer har en fiks-fokus linse som er fiksert på et punkt cirka 1.0 – 1.5 meter fra linsen utgjøre et problem. Kameraet vil av på grunn av denne begrensningen ikke være i stand til å gi et tilstrekkelig skarpt bilde av en strekkode som holdes nær linsen, samtidig som en strekkode som er i fokus vil være for lite detaljert til å kunne brukes.

Løsningen på dette problemet er å feste en linse fra en billig, enkel lupe foran

<sup>1</sup> På det europeiske markedet per februar 2005.

linsen til mobilkameraet, som da vil flytte mobilkameraets fokuspunkt til cirka 10 centimeter fra linsen. Dette vil måtte sies å være en ganske “ad hoc”-løsning, men vil duge som “proof of concept” i dette prosjektet.



*En enkel lupe med en enkel linse*

At mobiltelefoner generelt ikke har en spesielt god bildebrikke og derfor ikke tar spesielt gode bilder vil ikke være noe stort problem i denne sammenhengen, siden strekkodesystemet som tidligere nevnt er ganske robust og således i stor grad er i stand til å gjøre problemer som måtte oppstå relatert til støy, fargebalanse og lignende mindre problematiske.

---

## Praktisk testing av konseptet

For å undersøke om det faktisk var mulig å trekke ut informasjon av bilder av strekkoder tatt med en mobiltelefon, ble det implementert en fungerende modell på en vanlig datamaskin i Octave<sup>1</sup>. Bilder av strekkoder ble sendt fra mobiltelefonen til datamaskinen, og kjørt gjennom en algoritme som gikk gjennom en horisontal linje i bildet og prøvde å trekke ut informasjonen i strekkoden. Hvis algoritmen klarte å finne og dekode en strekkode fra denne linjen ble den tilhørende sifferrepresentasjonen til strekkoden skrevet ut – i motsatt fall ble det opplyst at ingen strekkode ble funnet.

På den absolutte majoriteten av bilder der dette forsøket ble gjennomført var det mulig å finne en rekke horisontale skanlinjer der algoritmen dekodet strekkoden korrekt. Gitt at dette kun var et forsøk og at algoritmen som ble brukt åpenbart har rom for forbedringer, må dette kunne sees på som et oppløftende resultat.

---

<sup>1</sup> Octave er et Matlab-lignende verktøy som brukes blant annet til vitenskapelige beregninger.



Bilde av en strekkode som testalgoritmen klarte å dekode – tatt med en Nokia 3650.

```
7 8 0 8 0 5 3 1 1 9 6 9
octave:402> imagesc(im2)
octave:403> barcode(im2, 200,11)
Scanner linje 200 med synkterskel 11.000000 !
Hvit er 205.814607, svart er 85.485915 og terskel er 145.650261
Synkdata (offset 1): 1926.333333 0.333333 0.333333 0.333333 2.333333
Synkdata (offset 2): 1.333333 0.333333 0.333333 0.333333 1.000000
Fant offset 2 i bildet : OK, synk funnet !
Fant enhetslengde 4.818182
Summen av streklengder er 94 :Feil!!!
ans = -1
octave:404> barcode(im2, 201,11)
Scanner linje 201 med synkterskel 11.000000 !
Hvit er 206.488636, svart er 86.625000 og terskel er 146.556818
Synkdata (offset 1): 1925.333333 0.333333 0.333333 0.333333 2.333333
Synkdata (offset 2): 0.000000 0.333333 0.333333 0.333333 1.000000
Fant offset 2 i bildet : OK, synk funnet !
Fant enhetslengde 4.727273
Summen av streklengder er 94 :Feil!!!
ans = -1
octave:405> barcode(im2, 203,11)
Scanner linje 203 med synkterskel 11.000000 !
Hvit er 205.740113, svart er 86.909091 og terskel er 146.324602
Synkdata (offset 1): 1925.333333 0.333333 0.333333 0.333333 4.333333
Synkdata (offset 2): 0.000000 0.333333 0.333333 0.333333 0.333333
Fant offset 2 i bildet : OK, synk funnet !
Fant enhetslengde 4.636364
Summen av streklengder er 95 :OK
ans =
7 8 0 8 0 5 3 1 1 9 6 9
octave:406> []
```

Screenshot fra testing av algoritmen (som viser at den fungerer...)

---

## Implementasjon

Siden dette prosjektet sannsynligvis vil kreve maskinvare med ytelse utover det som leveres av en typisk mobiltelefon med noe spartansk Java-støtte, vil vi utvikle dette prosjektet på en “smartphone” av typen Symbian/Series 60 – mer spesifikt en Nokia 3650<sup>1</sup>.

Series 60 fra Nokia<sup>2</sup> støtter utvikling i tre programmeringsspråk: Java, C++ og Python – hvor de to siste er de mest interessante i denne sammenhengen.

Vi har planlagt å implementere systemet med en kombinasjon av Python og C++, der C++ vil bli brukt til dekodingsalgoritmen og kommunikasjon med kamera og Python vil bli brukt til kommuniasjon med en sentral server og brukerinteraksjon. Python er velegnet i så henseende siden det tilbyr rask og effektiv prototyping der man ikke har noe utpreget behov for høy ytelse. Ved å lage en modul i C++ som er tilgjengelig fra Python vil det være relativt enkelt å bygge programmer i Python som bruker denne modulen, og dermed utvikle applikasjoner som kan lese og tolke informasjon fra strekkoder for forskjellige formål.

For implementasjon av en tjener er det naturlig nok flere muligheter for valg av programmeringsspråk og plattform, men vi vil nok også her satse på et scriptspråk som Python i kombinasjon med en web-tjener<sup>3</sup>. Dette vil ha den åpenbare fordelen at vi enkelt kan bruke våre hjemmeområder ved universitetet til testing og utvikling. Den ekstra “overhead'en” som vil måtte følge når man bruker HTTP vil neppe være så stor at det har noen betydning.

---

1 Først og fremst siden dette er telefonen vi har tilgjengelig.

2 Series 60 er en utvidelse/tilretteleggelse av Symbian-plattformen for enhåndsbruk på mobiltelefoner uten trykkfølsom skjerm.

3 Hvilket vil si at kommunikasjonen vil foregå over port 80 via HTTP.

---

## Veien videre, muligheter...

Ved bruk av en mobiltelefon med muligheten for å skanne strekkoder som er koblet til en datamaskin er det relativt enkelt å implementere tilsvarende funksjonalitet som "The Cue Cat" kunne tilby. Siden man som tidligere nevnt også har med seg en relativt kapabel datamaskin sammen med selve skanneren (i form av en relativt avansert mobiltelefon), vil det her også introduseres et mobilitetsaspekt som gjør det hele mer interessant (og mer relevant i denne sammenhengen).

Prinsippet med å gjøre dekodning av strekkoder mulig på mobiltelefoner har mange muligheter for videre bruk i forskjellige typer applikasjoner, som for eksempel:

- Enkel summering av prisen for varer i en handlekurv slik at man vet eksakt hvor mye varene vil koste. Dette kan også kombineres med at man på forhånd har satt opp en handleliste slik at varer man har plassert i handlekurven automatisk strykes ut fra listen.
- Sikker identifikasjon av tilbehør og lignende – Gitt at man trenger tilbehør til et produkt (for eksempel blekkpatroner til en skriver) kan dette systemet gjøre det mulig å sikkert identifisere riktig produkt.
- Allergi/diett – Mange lider av en eller annen form for matvareallergi. Dette systemet vil kunne tilby en enkel måte å identifisere potensielt problematiske matvarer.
- Umiddelbar prissammenligning – Ved hjelp av en sentralisert database over priser kan man sjekke om prisen på produktet er god eller mindre god. Ved at denne databasen har kategorier over produkter vil det også være mulig å få opp en liste over tilsvarende produkter (produkter med samme funksjon).
- Kaloritelling for konsumerte næringsmidler.
- Rask katalogisering – man kan raskt og enkelt bygge opp en database over for eksempel sin CD/DVD-samling ved å skanne strekkoden på CD/DVD-coveret.
- Eget bruk – Programvare for å skrive ut strekkoder er fritt tilgjengelig og kan i kombinasjon med en skanner som skissert over brukes til for eksempel logistikk, lagerføring, osv.



---

## Fremdriftsplan for prosjektet

- 9. februar – Innlevering av undringsdokument (dette dokumentet).
- Cirka 1. mars – Satt opp fungerende utviklingsmiljø for Series 60.
- Midten av mars ? - Innlevering av midtveisrapport (kjenner ikke eksakt dato).
- Ca. 1. april – Klar med en fungerende “proof-of-concept” applikasjon som kjører på mobiltelefon og en tilhørende kjørende tjenerapplikasjon.
- Ca. 1. mai – Ferdig med implementasjon.
- Midten av mai ? - Innlevering av sluttrapport (kjenner ikke eksakt dato).