Interaktivt Computer Vision System: Genkendelse af Dyr på Aalborg Zoos Savanne

10. Semester Kandidatspeciale,
Computer Vision, Grafik og Interaktive Systemer

Daniel Valsby-Koch
Aalborg Universitet
4. Juni 2014

AALBORG UNIVERSITET

STUDENTERRAPPORT

Institut for Elektroniske Systemer
Fredrik Bajers Vej 7
DK-9220 Aalborg Ø
http://es.aau.dk

Titel:
Interaktivt Computer Vision System: Genkendelse af Dyr på Aalborg Zoos Savanne

Tema:
Kandidatspeciale

Projektperiode:
Forårssemester 2014

Projektgruppe:
Gruppe 1040

Deltager:
Daniel Valsby-Koch

Vejleder:
Thomas B. Moeslund

Oplagstal: 3

Sidetal: 52

Bilag: 2

Afleveringsdato:
4. juni 2014

Synopsis:

I dette kandidatspeciale er et interaktivt computer vision genkendelsessystem udviklet. Brugeren kan med systemet tage et billede af et dyr på Aalborg Zoos Savanne, og brugeren oplyses hvilket af fem mulige dyr dette er. Foruden genkendelsesdelen er der ligeledes et stort fokus på brugervenligheden, da systemet netop er et komplet system med en dertilhørende brugergrænseflade.
Til genkendelse er der brugt simple pixel baserede features som hue, saturation, kanter og varians. Derefter bruges en KNN til klassificering af dyrene.
En 2-gange krydsvalidering giver en god genkendelsesrate på hele 86,78%. Ved en egentlig test af det samlede system ender genkendelsen dog på 34,91%, hvilket ikke er tilfredsstillende. Årsagen her til er blandt andet misforståelse af hvorledes systemet skal bruges, hvilkes også diskuteres.

Rapportens indhold er frit tilgængeligt, men offentliggørelse (med kildeangivelse) må kun ske efter aftale med forfatterne.

Forord

Denne rapport er en del af dokumentationen for mit 10. semesters kandidatspeciale, udarbejdet i foråret 2014, som led i specialet Computer Vision, Grafik og Interaktive Systemer på Aalborg Universitet. Projektet inddrager to af de tre områder som kandidatuddannelse handler om, nemlig Computer Vision og Interaktive Systemer. Foruden denne rapport er der i tillæg udarbejdet en videnskabelig artikel på skabelonen fra `www.isvc.net`. Disse to dokumenter udgør tilsammen kandidatspecialet.

Formaliteter

Igennem hele rapporten vil der være refereret til kilder som er samlet i en bibliografi til sidst i rapporten. Harvard metoden er brugt som reference stil. Dette betyder, at kilder refereres som [efternavn, udgivelsesår]. Figurer, tabeller og ligninger er nummereret i henhold til det kapitel de optræder i. Ved brug af tekniske forkortelser, så står det fulde navn første gange forkortelsen bruges.

Anerkendelser

Jeg ønsker at takke Aalborg Zoo for deres hjælp i forbindelse med kandidatspecialet. Denne hjælp har helt praktisk fungeret ved muligheden for at komme gratis ind i zoo, samt medbringe hjælpere og testpersoner. Desuden ønsker jeg at takke Claus B. Madsen (Aalborg Universitet) for hjælp med lån af prototype udstyr. Til sidst ønsker jeg at rette en tak til min vejleder Thomas B. Moeslund som har bistået mig med kompetent hjælp under hele specialeskrivningen.

Daniel Valsby-Koch

Indhold

1 Introduktion **1**
1.1 App idé til en zoologisk have . . . 3

2 Idéudvikling, projektudformning og -afgrænsning **5**
2.1 Stationær udgave . . . 5
2.2 Projektudformning og -afgrænsning . . . 6

3 Systemudvikling **9**
3.1 Hardware . . . 9
3.2 Software . . . 9

4 Kommunikation mellem kamera og computer **11**

5 Analyse af brugergrænsefladen **13**

6 Analyse af genkendelsesmodulet **15**
6.1 Segmentering . . . 17

7 Feature extraction **19**
7.1 Farve features . . . 20
7.1.1 Farverum . . . 20
7.1.2 Farve som feature . . . 21
7.2 Tekstur features . . . 22
7.2.1 Middelværdi og varians . . . 22
7.2.2 Sobel . . . 23
7.2.3 Canny kantdetektering . . . 24

8 Klassificering **27**

9 Træningsdata **29**

10 Prototypen **31**
10.0.4 Segmentering, feature extraction og klassificering . . . 32
10.0.5 Brugergrænsefladen . . . 33

11 Resultater fra prototypetest **35**
11.1 Test på træningsdata . . . 35
11.2 Prototypetest i Aalborg Zoo . . . 36
11.2.1 Resultater af prototypetest i Aalborg Zoo . . . 36
11.2.2 Spørgeskemaresultater . . . 38
11.3 Analyse af testresultater . . . 38

12 Analyse af krav og forbedringer **41**

12.1 Forbedringer af systemet 41

Litteratur **43**

A Test af antal udklip og størrelsen på k **45**

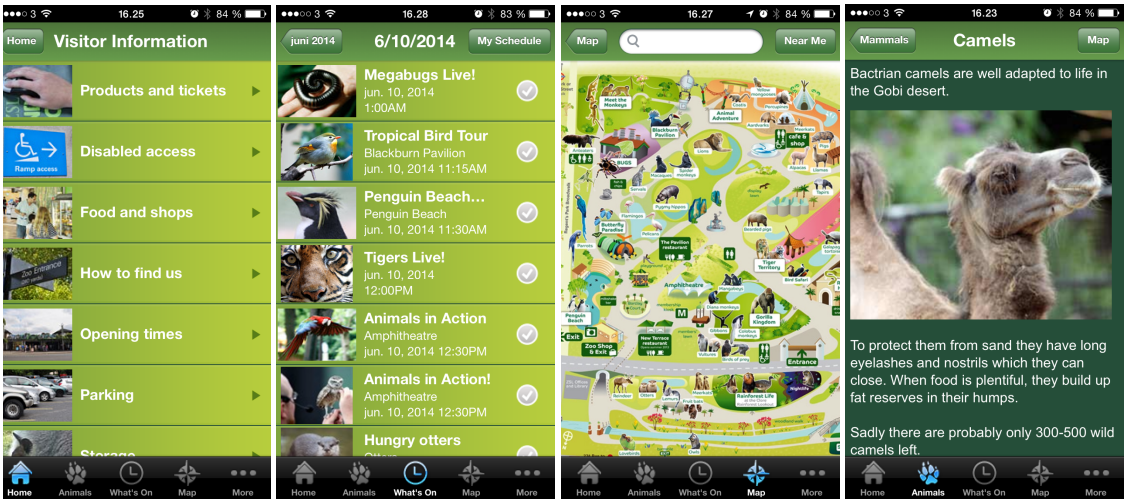
B Spørgeskema resultater **47**

Introduktion 1

I Danmark og resten af verden findes der mange forskellige seværdigheder, som man naturligvis kan besøge og opleve. Næsten alle former for seværdigheder ønsker at formidle nogle informationer i et eller andet omfang, hvad enten det er et museum, en forlystelsespark eller en dyrepark. Disse informationer kan formidles på mange forskellige måder, hvor den mest udbredte, og bedst kendte er et gammeldags skilt med tekst. Foruden denne formidlingsform, så findes der en del andre måder, hvorpå man kan få informationer om det man nu engang besøger eller kigger på. Et eksempel er The Metropolitan Museum of Art (MET) i New York, USA, hvor man kan leje en audio guide [The Metropolitan Museum of Art, 2014]. Med denne audio guide i hånden kan man få oplæst informationer om de ting man kigger på, ved blot at taste det rigtige tal ind, som man får oplyst når man står ved fx et bestemt maleri. Det smarte ved en audio guide er selvfølgelig, at man får det læst op. Det giver brugeren mulighed for at kigge på fx maleriet samtidig med, at man får informationer om maleriet. Desuden kan man nemmere have noget mere information end man kan på en tavle. Det sidste gode ved en audio guide er selvfølgelig muligheden for at have mange forskellige sprog. De fleste seværdigheder har informationerne til at stå på lokalsproget, og eventuelt en eller to andre sprog. Med en audio guide kan man altså have mange forskellige sprog, uden det skal fylde en masse tavler med tekst.

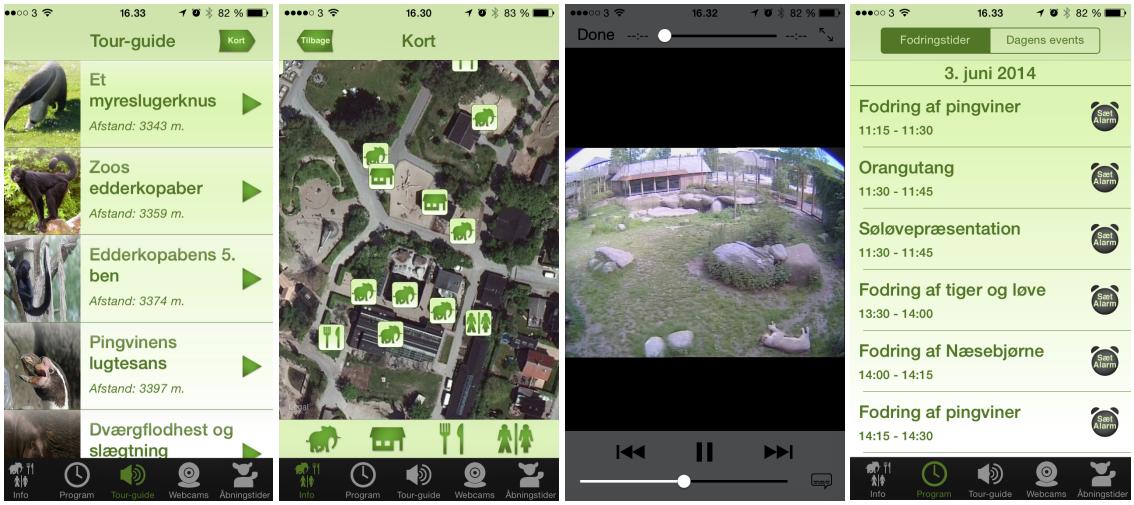
Et andet eksempel på en formidlingsform er brugen af smartphones. London Zoo har en app til iPhones [ZSL London Zoo, 2014] hvor man har mulighed for at få nogle forskellige informationer. Her kan man både få generelle oplysninger om London Zoo, men man kan også læse om de enkelte dyr. Samtidig kan man få et kort over haven, og man kan vha. gpsen i telefonen få sin egen placering vist, og derved navigere sig hen til det dyr man ønsker at se. Denne formidlingsform bliver mere og mere populær, da det giver gæsten mulighed for blot at bruge sin egen telefon for at opnå flere informationer. I modsætning til audio guiden ved MET, så er denne app gratis, og gæsterne kan altså uden ekstra omkostninger få flere informationer. Samtidig er denne app også en form for reklame, da man uden at tage i London Zoo kan hente og installere appen, og derved blive fristet til at besøge London Zoo.

London Zoo er selvfølgelig ikke den eneste zoologiske have med en smartphone app. Aalborg Zoo har også udviklet en app, som foruden de muligheder man har i appen fra London Zoo, også har nogle flere muligheder. Her er det muligt at få oplyst afstanden til alt fra dyrene og zooshoppen til det nærmeste toilet. Dette giver gæsterne en moderne og nem måde at finde rundt på, og dette sammenkobles også med Google Maps hvor man via satellitbilleder også kan navigere rundt i parken. En anden mulighed med appen fra Aalborg Zoo er også at se live webcams fra flere af dyrene. Denne mulighed er både brugbar hvis man er i parken, men også interessant hvis man ikke er i Aalborg Zoo. På denne måde skaber Aalborg Zoo



Figur 1.1. Figuren viser fire screenshot af London Zoos iPhone app. Første billede viser generelle informationer om zooen. Billede to er en oversigtskalender med de kommende events. Billede tre er et grafisk kort over zoo. Det sidste billede er oplysninger om de forskellige dyr i zoo.

en besøgslyst hos alle dem der har appen installeret. På figur 1.2 ses fire screenshot af Aalborg Zoos iPhone app.



Figur 1.2. Figuren viser fire screenshot af Aalborg Zoos iPhone app. Første billede viser afstanden til dyrene i zoo, samt giver brugeren mulighed for at lytte til informationer om dyret. Næste billede er et Google Maps satellitbillede med dyrene og faciliteterne i Aalborg Zoo. Tredje billede viser live video fra løveindhegningen. Det sidste billede oplyser brugeren om de kommende fodringer, og giver brugeren mulighed for at indstille en alarm.

I tilgift til en hovedsageligt informationsbaseret app som dem London Zoo og Aalborg Zoo har, så udvikles der også mere interaktive apps, som kan bruges ved seværdigheder. Her kan et samarbejdsprojekt mellem Aalborg Universitet og Syddansk Turisme fremhæves[Claus B. Madsen, 2011]. Kort fortalt, så handler det om en app til iPad, hvor besøgende på borgen

Koldinghus, kan opleve Koldinghus gennem en iPad og ved brug af augmented reality. Appen er beregnet til børn, hvor man via iPaden og nogle spillekort bliver ledt gennem dele af museet og oplever museet interaktivt. Denne måde at formidle informationer på bliver mere populær, især i forbindelse med udbredelsen af smartphones og tablets.

Andre eksempler er også stationære tavler med tekst, hvor man så kan trykke på en knap, og derved få oplæst teksten. Dette er blandt andet brugbart for børn og andre der ikke kan læse hvad der står.

1.1 App idé til en zoologisk have

Eftersom alting forandrer sig og udvikler sig, så er det vigtigt for de forskellige seværdigheder, at sørge for at være uptodate med de nyeste formidlingsmuligheder. Med udgangspunkt i dette, og med et ønske om, at gøre informationerne tilgængelige til så mange gæster som muligt, er følgende idé opstået:

En app til en smartphone, hvorved brugeren kan filme et dyr og få informationer om det pågældende dyr.

Med denne app er det altså muligt at filme et tilfældigt dyr i en given zoologisk have og derved få nogle informationer om dyret. Disse informationer kan så være tilgængelige på alverdens sprog, da dette blot er et spørgsmål om oversættelse, og desuden kan man få alle informationerne oplæst, således at børn og andre med læsevanskeligheder også kan få gavn af informationerne. En anden mulighed med sådan en app er, at man fra zoos side nemt kan opdatere informationerne og tilføje nyheder såsom en fødsel.

Idéudvikling, projektudformning og -afgrænsning 2

Idéen med at genkende dyr via en app på en smartphone eller tablet, kræver en væsentlig ting, hvis det skal kunne udføres. Det er selvfølgelig nødvendigt, at en genkendelsesrutine kan laves, således at hvert eneste dyr kan genkendes. Efterfølgende skal denne rutine kunne eksekveres på en smartphone og/eller tablet i tilfredsstillende hastighed, da brugeren ellers vil kunne være utilfreds med brugeroplevelsen. Derved vil behandlingskraften på den håndholdte enhed være afgørende for, hvor kompliceret en algoritme der kan afvikles. Foruden dette punkt, så er der et par andre ting der skal overvejes ved sådan en app, hvilke er listet nedenfor:

- Behandlingskraft på den håndholdte enhed
- Kvaliteten af kameraet
- Dyrets indhegning
- Brugerens placering i forhold til dyret

Foruden behandlingskraften er kvaliteten af kameraet også en enhedsspecifik begrænsning/udfordring. Her er der mange muligheder, alt efter hvilken enhed der bruges, hvilket er en faktor der skal tænkes meget over. Udover de to udfordringer, er der to overvejelser som ikke har noget med den håndholdte enhed at gøre, men som relaterer til stedet hvor appen skal bruges. Den ene faktor er dyrets indhegning. Her findes der mange forskellige former for indhegning, herunder glas, trådhegn og træhegn. Glas vil kunne påvirke optagelsen pga. eventuel genspejling og derved besværliggøre processen. Brugerens placering i forhold til dyret er også en væsentlig faktor. I nogle tilfælde vil brugeren kunne være relativt tæt på dyret, og her er det muligvis nemmere at genkende dyret ved brug af enhedens kamera. Omvendt vil der i andre tilfælde være langt, måske 100 meter hen til dyret, hvilket vil sætte selv nutidens smartphonekameraer på prøve. Disse to eksterne faktorer er vigtige at tage med i overvejelsen.

2.1 Stationær udgave

En anden mulighed er at bruge en anden platform end brugerens egen smartphone. I stedet for at bruge en håndholdt enhed kan man opsætte et system hvor programmet installeres på. Idéen med at bruge en touchskærm bibeholdes, da dette ofte er en intuitiv måde at tilgå et program på. Koblet på touchskærmen er så et kamera, og det hele er påsat et stativ, således at brugeren kan dreje skærmen og kameraet. Herved kan brugeren filme et dyr og få

relevante oplysninger om selve dyret. Hele systemet kan altså opsættes på relevante steder i en zoologisk have.

2.2 Projektudformning og -afgrænsning

I Aalborg Zoo er der en stor Savanne, hvorpå der går fem forskellige dyrearter, nemlig giraffer, strudse, zebraer, store kuduer og sabeloryxer. Ved Savannen er der et udsigtsplateau, hvor gæsterne også har mulighed for at spise deres medbragte mad. I forbindelse med plateauet er der opstillet fem infotavler som hver oplyser gæsterne om de fem forskellige dyrearter på henholdsvis dansk og engelsk. En skitse samt et satellitbillede af Savannen ses på figur 2.1. Her ville opsætningen af et dyregenkendelsessystem være fordelagtigt, da der netop er mulighed for at se flere forskellige slags dyr, uden at gæsterne nødvendigvis ved hvad de forskellige dyr er. En problemformulering for dette lyder som følger:

Er det muligt at konstruere et simpelt, børne- og brugervenligt, interaktivt, computer vision system, som ved hjælp af et kamera kan genkende et dyr blandt fem forskellige dyr på Aalborg Zoos Savanne i nær real-tid, og informere brugeren om den bestemte dyreart?

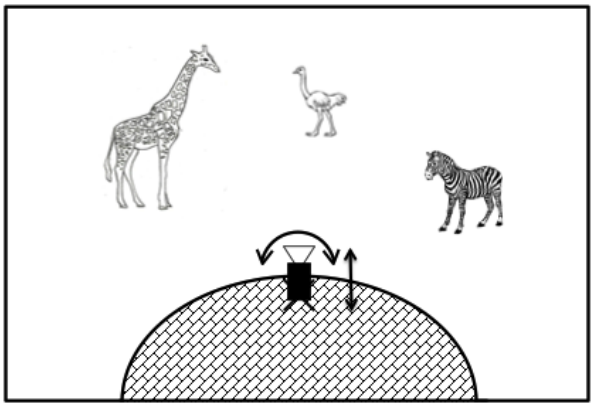
Med udgangspunkt i dette opstilles nogle krav til prototypen:

- Prototypen skal kunne genkende og adskille de fem forskellige dyr på Savannen i Aalborg Zoo
- Prototypen vil bestå af en skærm på et stativ, med et påmonteret kamera og numpad tastatur. Alt sammen koblet til en bærbar computer
- Prototypen har to grader af frihed, nemlig horisontalt og vertikalt
- Prototypen indeholder en gennemarbejdet brugergrænseflade med brugervenlighed som et fokus, og hvor brugeren kan benytte systemet uden menneskelig vejledning
- Responstiden må ikke være på mere end 3 sekunder
- Genkendelsesraten skal være på mindst 85% ved krydsvalidering
- Genkendelsesraten skal være på mindst 75% ved den endelige test

Desuden vil der være følgende afgrænsninger:

- Prototypen kan ikke zoome
- Prototypen vil kun være på dansk
- Prototypen vil kun i nogen grad kunne oplæse teksten
- Prototypen har ikke touch funktionalitet

Foruden de ovenstående krav og afgrænsninger, så vil hele projektet følge en rød tråd omhandlende simplicitet. Det vil sige at der oftest vil blive afprøvet simple metoder og teknikker, inden mere komplicerede og avancerede løsninger testes.



Figur 2.1. Figuren til venstre viser en skitse af Savannen i Aalborg Zoo. Halvcirklen repræsenterer plateauet hvor det er muligt at spise sin mad, og hvor en prototype kan opstilles. Selve Savannen er knap 100 meter dyb og 150 meter bred og i alt 10.000 m^2. En afmærkning af Savannen ses på billedet til højre, hvor plateauet er markeret med en rød plet. Billedet er taget fra `www.krak.dk`

Systemudvikling 3

I afsnit 2.2 er kravene og afgrænsningerne for systemet opstillet. På baggrund af disse krav skal en prototype udvikles og testes. Da systemet skal bruges af rigtige mennesker, så er det selvfølgelig nødvendigt at kigge på brugervenligheden. Overvejelser i forbindelse med brugergrænsefladen diskuteres i kapitel 5.

3.1 Hardware

Systemet vil bestå af tre væsentlige hardwareelementer:

- Kamera
- Stativ med skærm
- Computer

Computeren er i denne prototype en almindelig bærbar computer mere specifikt en MacBook Pro model MB990 (2,26 gHz Intel Core 2 Duo og 4GB ram). Kameraet er et digitalt spejlrefleks model D7000 fra Nikon. Påsat er et 18-200 mm objektiv, hvor brændvidden konstant er på 200 mm. Denne brændvidde er bestemt ved forsøg, da der derved dækkes mest muligt af Savannens dybde. Den sidste hardwaredel er stativet og skærmen. Her bruges et stativ fra et tidligere projekt ved Aalborg Universitet, hvor en 21 tommer skærm er påmonteret stativet og man har mulighed for at dreje skærm og stativ både horisontalt og vertikalt. Herpå er et beslag påmonteret, så det er muligt at fastspænde kameraet lige bagved skærmen.

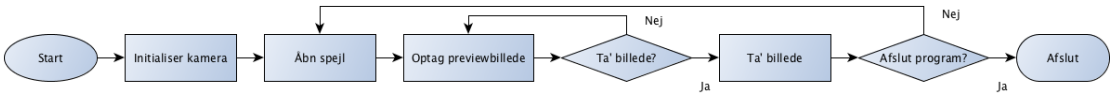
3.2 Software

Selve softwaren til systemet består af tre hovedelementer. Først og fremmest er et kommunikationsmodul programmeret, som håndterer kommunikationen mellem computeren og kameraet. Dette modul er beskrevet i kapitel 4. Dernæst er selve genkendelsessystemet med feature extraction og klassificeringen designet, hvilket er beskrevet i 6. Ovenpå det hele er en brugervenlig brugergrænseflade lagt, som også præsenterer brugeren for informationerne om dyret der genkendes. Overvejelser omkring dette findes i kapitel 5. Standardproceduren for genkendelse af et objekt ved brug af billeddata følger figur 3.1. Først og fremmest optages et billede, som ofte skal forbehandles. Forbehandlingen kan være alt fra ændring af farverum til at finde kanter i et billede. Efter forbehandlingen segmenteres billedet, da det sjældent er hele billedet der skal bruges til noget. Herefter fokuseres der på den segmenterede del af billedet, hvor forskellige interessante og relevante features udtrækkes. Til sidst bruges disse features til at klassificere objektet, i dette tilfælde som et af fem dyr eller som en fejl.

Figur 3.1. Et flowchart der viser den overordnede procedure ved genkendelse af et objekt ved brug af billededata.

Kommunikation mellem kamera og computer 4

Kameraet er, som beskrevet tidligere, et Nikon D7000 digitalt spejlreflekskamera. Med dette kamera er det muligt at forbinde computeren og kameraet via en USB forbindelse. For at kommunikere med kameraet, så bruges der et C bibliotek kaldet *libgphoto2*. Med dette bibliotek er det muligt at kommunikere med kameraet, og derved modtage billeder. Da hele programmet er skrevet i python, er der brugt et python-bindings bibliotek, til at bruge *libgphoto2*. Dette bibliotek hedder *piggyphoto*, og er udelukkende python-bindings til nogle af funktionerne i *libgphoto2*. Programmet viser et livestream fra kameraet, og når brugeren ønsker at genkende et dyr, så tages der et billede som bruges til genkendelse. Kommunikationsrutinen er vist i figur 4.1. Først initialiseres kameraet, hvorefter spejlet i kameraet åbnes, og der kan optages previewbilleder. Disse previewbilleder (640x426 pixels) modtages løbende på computeren og fremvises på skærmen. Når brugeren så ønsker at tage et billede, så lukkes spejlet, hvorefter kameraet tager et rigtigt billede (2464x1632 pixels).



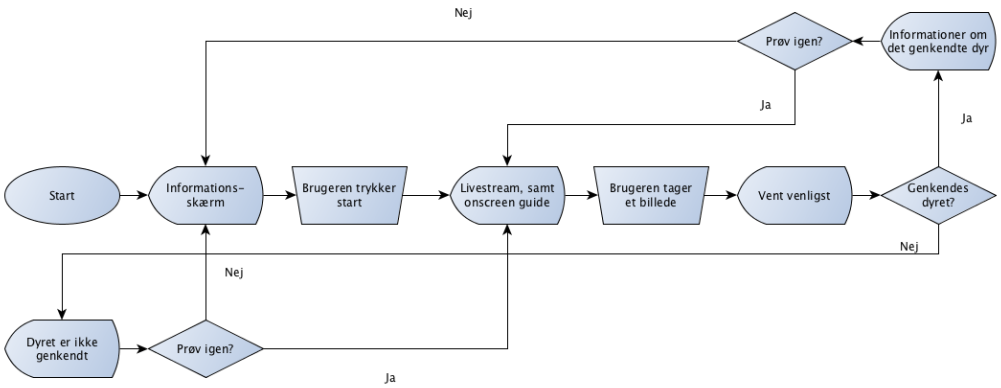
Figur 4.1. Et flowchart der viser flowet for kommunikationen mellem kameraet og computeren. C biblioteket libgphoto2 bruges til at kommunikere med kameraet.

Et problem med dette bibliotek er dog, at med et Nikon D7000 er det ikke umiddelbart muligt at fokusere med kameraet inden et billede tages. Dette er en væsentlig faktor, da et uskarpt billede vil være anderledes end et skarpt billede. Løsningen på dette har været at bruge en meget høj brændvidde, hvilket giver den størst mulige dybdeskarphed.

Analyse af brugergrænsefladen 5

En væsentlig del af et program er naturligvis brugergrænsefladen. Prototypen indeholder ikke en touchskærm, men da det er tanken at et færdigt produkt skal fungere med en touchskærm, så vil brugergrænsefladen også tage udgangspunkt i dette. I stedet vil der dog være små indikationer, der fortæller brugeren hvilken tast på tastaturet der skal trykkes på.

Prototypen vil komme til at stå ved Savannen uden nogen menneskelig guide til at fortælle hvordan det skal betjenes. Det er derfor vigtigt at programmet er så intuitivt som muligt, og at brugeren får de informationer der er nødvendige for, at han eller hun kan betjene prototypen. Figur 5.1 viser hvordan brugeren guides gennem systemet og hvilke informationer brugeren præsenteres for. Som det fremgår af figuren, så indeholder brugergrænsefladen fem præsentationer. Den første er informationsskærmen, dernæst livestreamskærmen, så en venteskærm, efterfulgt af en fejlskærm eller en dyreinformationsskærm.



Figur 5.1. Flowchart der viser brugerens vej gennem systemet, set fra en brugergrænsefladevinkel.

Nedenfor er listet de fem forskellige skærme som brugergrænsefladen skal indeholde, og hvilke overvejelser der er omkring disse.

Informationsskærmen er den skærm som programmet konstant vil fremvise. Denne skærm har derfor til sinde at fange gæsterne i zoo, og gøre dem interesseret i at bruge systemet. Skærmen bør indeholde en startknap og en oplæsningsknap.

Livestreamskærmen er den første skærm der præsenteres for brugeren, efter at personen har trykket på start. Denne skærm skal indeholde videofeedet fra kameraet, så brugeren kan benytte systemet og finde et dyr. Desuden skal vejledningen til brug af systemet være på denne skærm, gerne støttet af en eller flere eksempler på rigtige og forkerte billeder. Først og fremmest skal skærmen indeholde en ta'-billede-knap, men den bør også indeholde en afslut-/tilbageknap og en oplæsningsknap.

Venteskærmen er nødvendig så brugeren ved at systemet arbejder. Dette kan enten være et popup vindue eller eventuelt bare en opdatering af livestreamskærmen, med en tydelig note om at systemet arbejder på sagen.

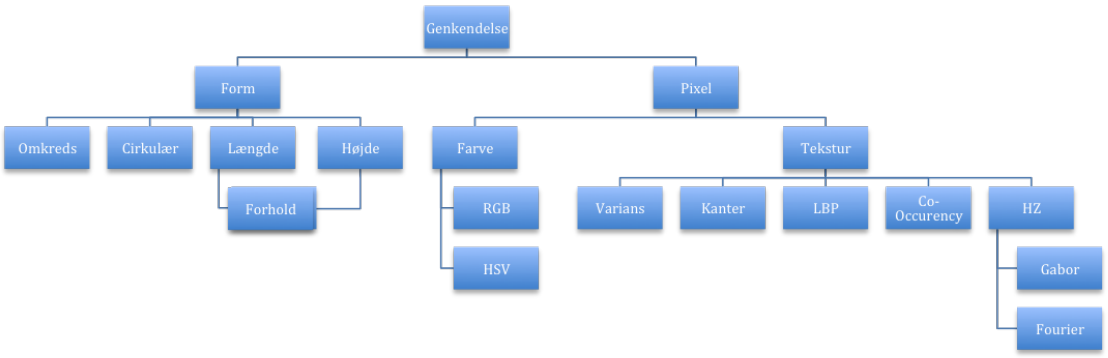
Fejlskærmen informerer brugeren om, at systemet ikke kunne genkende noget dyr ud fra billedet som brugeren tog. Denne skærm kunne meget vel være en mindre popupskærm med muligheden for at prøve igen eller afslutte programmet.

Dyreinformationsskærmen skal indeholde alle relevante informationer om det dyr som systemet har genkendt ud fra billedet. Her kan stå alt fra dyrets kost til størrelse og levesteder. Her er det især vigtigt, at teksten kan læses op, så bl.a. børn kan få de interessante oplysninger. Desuden skal skærmen indeholde muligheden for at prøve igen eller afslutte programmet.

En afslutning af programmet vil ikke afslutte programmet fuldstændigt, men blot sende brugeren tilbage til den første informationsskærm, således at systemet er klar til næste bruger. Den endelige brugergrænseflade er beskrevet og fremvist i kapitel 10.

Analyse af genkendelsesmodulet 6

Den del af softwaren som kan genkende dyret, kan udvikles på mange forskellige måder. En overordnet fremgangsmåde for Computer Vision systemer, med billeder som input, ses på figur 3.1 på side 10. Den første ting der skal overvejes er hvorledes objektet, i dette tilfælde dyret, skal segmenteres. Dette kan gøres på flere forskellige måder, hvilket er afhængigt af hvilken fremgangsmåde der bruges ved feature extraction. Derfor skal feature extraction delen først overvejes. Når billedet er taget, forbehandlet og segmenteret (de første tre punkter af figur 3.1), så er det netop tid til feature extraction. Denne feature extraction kan tage udgangspunkt i mulighederne fremvist på figur 6.1. Overordnet er der to kategorier, nemlig formbaseret og pixelbaseret feature extraction. En af de største forskelle på disse to kategorier er, at man ved brug af formen skal have hele dyret, hvor man i stedet kan nøjes med dele af dyret når man kun bruger pixels.



Figur 6.1. Figuren viser to kategorier af features i et genkendelsessystem, samt eksempler på tilhørende features.

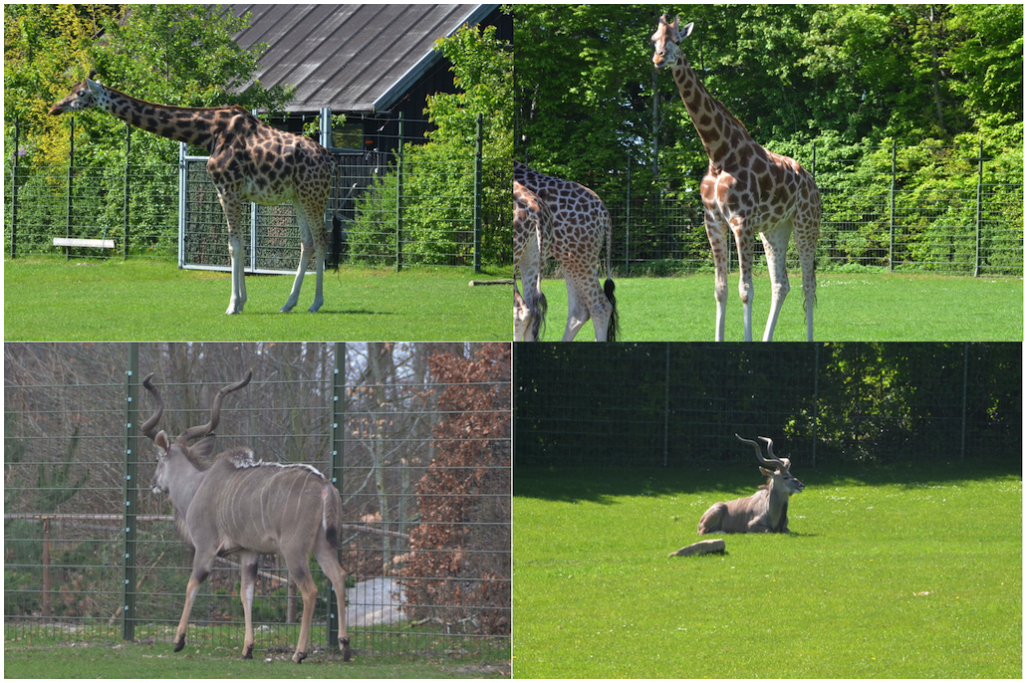
Dette skaber igen behovet for nogle overvejelser vedrørende designet af genkendelsessystemet. Alt efter hvilken kategori der vælges, så afføder det nogle krav til segmenteringen.

Formbaseret feature extraction Her kræves det at hele dyret bliver segmenteret korrekt hver gang. Det vil sige, at brugeren skal tage et billede af hele dyret, hvorefter systemet skal adskille baggrunden fra dyret, og sørge for at kun dyret er segmenteret. Denne fremgangsmåde vil kræve behovet for en baggrundsfjernelsesmetode. En måde er at bruge

en baggrundssubtraktionsalgoritme, hvor man forsøger at estimere baggrunden og bagefter trække denne baggrund fra det nye billede[Moeslund, 2009]. Derved får man forgrunden, hvilket i dette tilfælde er dyret. Selvom der findes måder hvorpå man kan generere en adaptiv baggrund, så vil bevægelser med kameraet gøre denne metode ubrugelig.

En anden metode der kan bruges er GrabCut algoritmen [Rother et al., 2004], som også er algoritmen der bruges i nogle af Microsofts produkter til segmentering af billeder. Algoritmen er udviklet til at segmentere forgrund fra baggrund ved brug af graph-cut metoden. Denne algoritme er dog interaktiv, hvilket betyder, at brugeren selv skal hjælpe til med at finde dyret på billedet. Da idéen med systemet indeholder en touchskærm, så er det ikke udelukket at brugeren selv kan hjælpe til med at segmentere dyret. Dette er dog ikke ideelt, da det vil være meget afhængig af brugeren, og derfor er svært at vide hvordan brugeren præcis vil gøre det.

Hvis det antages at man kan implementere en metode hvorved dyret er komplet segmenteret hver gang, så er der stadig en del udfordringer ved brug af formen på dyret. Som figur 6.2 viser så er giraffen og kuduen to eksempler på dyr der har to vidt forskellige former, alt efter hvilken positur den står i. Da systemet skal kunne tage højde for dette, så er det som sagt en stor udfordring.



Figur 6.2. Øverst er to eksempler på positurer som en giraf kan stå i. Nederst er ligeledes to eksempler på hvordan en stor kudu kan opleves. Billederne er taget på Aalborg Zoos Savanne.

Pixelbaseret feature extraction I modsætning til metoden hvor man bruger dyrets form, så kan man i stedet bruge features hvor man finder det for hver enkelt pixel i stedet. Ved brug af denne form for features er det derfor ikke længere lige så nødvendigt at have hele dyret. Selvfølgelig vil en komplet og perfekt segmentering af dyret stadig være godt, da man derved kan bruge pixels fra hele dyret til at finde features. Dette er dog ikke

nødvendigt, og man kan i stedet nøjes med at bruge dele af dyret. Dette kræver dog, at systemet ved hvor der med sikkerhed er dyr henne på et billede. For at løse dette er der flere muligheder:

- Sørge for at zoome så meget, at der ikke er nogen baggrund på billedet, og derved kan bruge hele billedet som forgrund
- Sørge for at hele dyret er på billedet, og fylder en vis størrelse. Derved kan man estimere hvor der altid vil være noget af dyret, afhængigt af hvilket dyr det er
- Sørge for at et vist udsnit, fx midten af billedet, altid indeholder et dyr

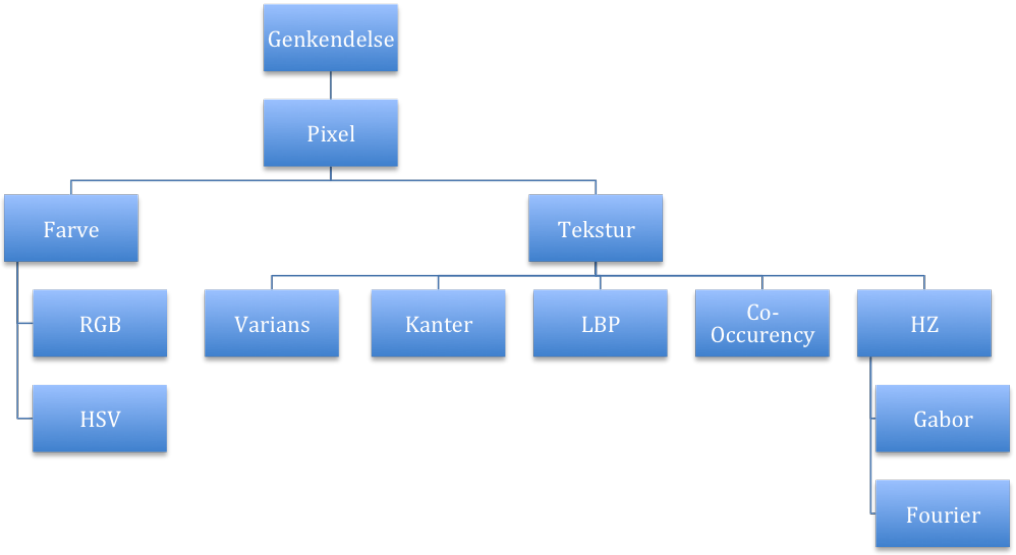
Med udgangspunkt i de ovenstående muligheder, så virker den nederste mulighed mest oplagt. Første mulighed er udelukket, da det ikke er muligt at zoome med prototypen, hvilket beskrives i afsnit 3.1. Den midterste mulighed er mere oplagt, dog er der et vist problem med positionen af dyret. Det er muligt at estimere hvor dyret er afhængigt af hvilket dyr det er, men da et dyr ikke altid står i samme positur, som det ses på figur 6.2, så er det heller ikke altid det samme sted på et billede hvor dyret vil være. Derfor virker den sidste mulighed som den bedste mulighed.

6.1 Segmentering

Begge former for feature extraction som nævnes ovenfor er interessante, og vil muligvis begge kunne bruges med et godt resultat tilfølge. Med udgangspunkt i udfordringerne ved segmentering af hele dyret vælges pixelbaseret feature extraction. Dette gør også med ønsket om at lave et så simpelt, men dog fungerende, system som muligt. Mht. selve segmenteringen, så tages der udgangspunkt i metoden, hvor et kendt udsnit af billedet altid indeholder et dyr. Her tænkes der specifikt på midten af billedet, hvor brugeren vil blive bedt om at sørge for, at en firkant i midten af billedet kun indeholder dyr og ingen baggrund. Efter segmenteringen er næste punkt feature extraction, som undersøges i kapitel 7. Derudover gennemgås klassificeringsdelen i kapitel 8.

Feature extraction 7

I kapitel 6 blev to forskellige tilgange til feature extraction præsenteret. Efter overvejelser omkring segmenteringsmåden, faldt valget på den simple måde hvor den midterste del af billedet klippes ud og bruges til genkendelse af dyret. Dette medfører også, at featurene er baseret på de enkelte pixels. Figur 7.1 viser de pixelbaserede features der også nævnes i kapitel 6.

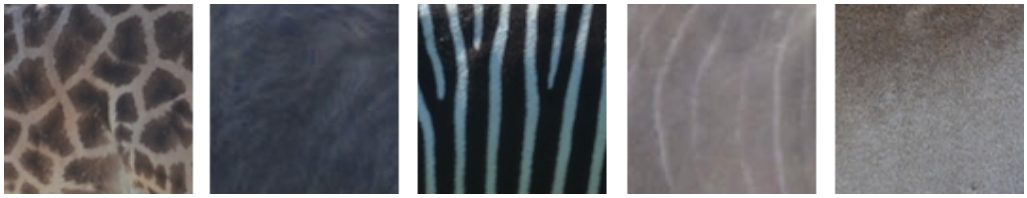


Figur 7.1. Figuren viser eksempler på pixelbaserede features, inddelt i de to kategorier farve og tekstur.

For at finde de features der er brugbare i et genkendelsessystem, så er det nødvendigt at kigge på de billeder som featurene skal findes i. På Savannen i Aalborg Zoo er der fem forskellige slags dyr, nemlig giraffer, strudse, zebraer, store kuduer (omtales som kuduer) og sabeloryxer (omtales som oryxer). Billeder taget af de fem forskellige dyr i Aalborg Zoo fremgår af figur 7.2.

Figur 7.2. Billedet viser de fem slags dyr fra Savannen i Aalborg Zoo. Øverst fra venstre er det: giraf, struds og zebra. Nederst fra venstre er det kudu og oryx.

Når man ser på disse dyr, så ser man straks, at de to kategorier nævnt på figur 7.1, beskriver dyrenes forskelligheder. Dyrene har forskellige farver, mønstrer og strukturer, hvilket inddeles i kategorierne farve og tekstur. Et udsnit fra hvert af de fem dyr ses i figur 7.3. Det er et lignende udsnit som vil blive segmenteret fra det oprindelige billede. De tilhørende features beskrives i henholdsvis afsnit 7.1 omhandlende farven og afsnit 7.2 omhandlende teksturen.



Figur 7.3. Billedet viser et udsnit af de fem dyr fra Savannen. Udsnittene svarer til hvordan en segmenteret del af dyret kan se ud. Fra venstre er det: giraf, struds, zebra, kudu og oryx.

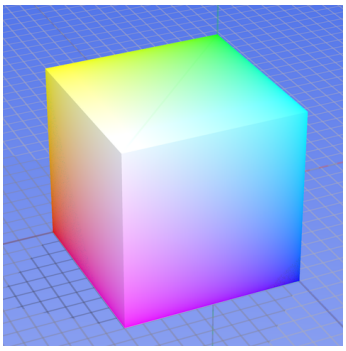
7.1 Farve features

Den første relevante feature at fokusere på er dyrets farve. Ved første øjekast, så ser det ud som om, at dyrene har ret forskellige farver. Måden hvorpå farven bruges som en feature kan være forskellig. Først og fremmest skal der vælges et farverum at arbejde med. Herefter skal der tages stilling til hvordan dyrets farve skal indgå som en feature.

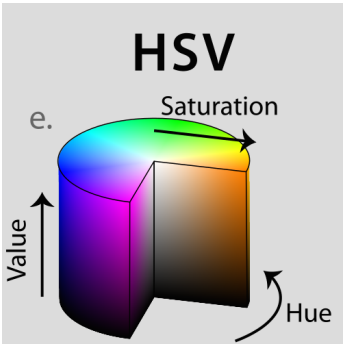
7.1.1 Farverum

Mennesket opfatter verdenen bestående af tre farver, nemlig rød, grøn og blå. Dette gøres da menneskets øje består af tre slags tappe som hver især opfanger bestemte bølgelængde områder. Disse tre farver er selvfølgelig rød, grøn og blå. På grund af menneskets opfattelse af farverne, så er det også på denne måde at man repræsentere fx billeder på en computerskærm. Et farvebillede består altså af pixels med en additiv kombination

af farverne rød, grøn og blå, og det farverum kaldes derfor RGB (Red, Green, Blue) [Moeslund, 2009]. Problemet med dette farverum er, at lysintensiteten udendørs er en stor variabel faktor, alt efter hvor meget sol der er på en given dag. I RGB farverummet arbejder man med additionen af tre forskellige farver, og man har derfor ikke nogen adskillelse af farve og lysintensitet. Dette problem kan løses ved at bruge et andet farverum fx HSV (Hue, Saturation, Value). Med dette farverum har man adskilt nuance (hue), mætning (saturation) og lysintensitet (value). Teoretisk set betyder dette, at man nu kan nøjes med fx at bruge hue og saturation til at beskrive en farve, og derved er man ligeglad med hvor lyst eller mørkt der er. De to farverum ses på figur 7.4 og 7.5. Her ses det hvordan RGB er formet som en terning med rød, grøn og blå i tre af hjørnerne. HSV farverummet er vist som en cylinder, hvilket opstår da hue værdien er mellem 0° og 360°.



Figur 7.4. Terning der viser alle tilgængelige farver i RGB farverummet. Figuren er fra Wikimedia Commons `en.wikipedia.org/wiki/File:RGB_color_solid_cube.png`



Figur 7.5. Cylinder der viser alle farverne i HSV farverummet. Figuren er fra Wikimedia Commons `en.wikipedia.org/wiki/File:Hsl-hsv_models.svg`

Billedet der bruges til at finde farve featuren på er i RGB. Det betyder at farverne skal konverteres fra RGB til HSV. Konverteringen i OpenCV foregår via nedenstående formler [Moeslund, 2009].

$$H = \begin{cases} \frac{G-B}{V-min\{R,G,B\}} \cdot 30°, & \text{if V=R and G} \geq \text{B;} \\ (\frac{B-R}{V-min\{R,G,B\}} + 2) \cdot 30°, & \text{if G=V;} \\ (\frac{R-G}{V-min\{R,G,B\}} + 4) \cdot 30°, & \text{if B=V;} \\ (\frac{R-B}{V-min\{R,G,B\}} + 5) \cdot 30°, & \text{if V=R and G} < \text{B;} \end{cases} \qquad H \in [0°, 180°[\tag{7.1}$$

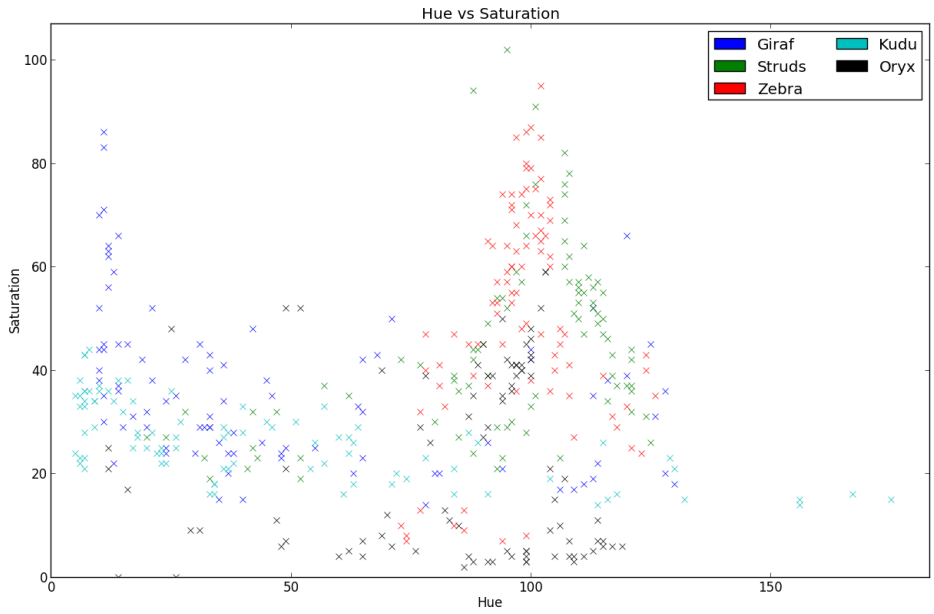
$$S = V - min\{R, G, B\} \qquad S \in [0, 255] \tag{7.2}$$

$$V = max\{R, G, B\} \qquad V \in [0, 255] \tag{7.3}$$

7.1.2 Farve som feature

Når billedet er i HSV farverummet, så bruges de to af kanalerne som to features. Som tidligere nævnt, så er det foretrukket, at en farve feature ikke er afhængig af lysintensiteten.

Derfor bruges kun hue og saturation delene af billedet. Figur 7.6 viser et plot af hue og saturation for et udsnit af træningsdata.



Figur 7.6. Figuren viser et plot af hue og saturation for en del af træningsdataet.

7.2 Tekstur features

De fem forskellige dyr som findes på Savannen har forskellige former for teksturer og mønstre. Disse egenskaber adskiller dyrene, og er også en af de måder hvor man som menneske kan se forskel på dyrene, såfremt man kan så et udsnit af dyrene svarende til figur 7.3. I dette afsnit analyseres de fem tekstur features som benyttes til genkendelse af dyrene, nemlig sobel, canny kant, middelværdien og variansen. Desuden nævnes til sidst nogle andre features som eventuelt også kunne benyttes.

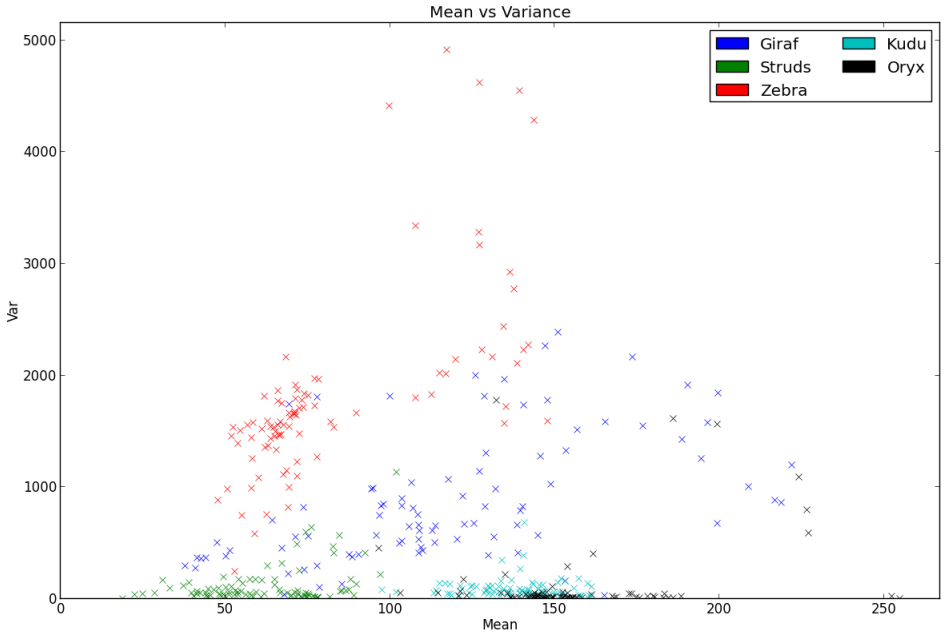
7.2.1 Middelværdi og varians

To features der kan bruges er middelværdien og variansen. Middelværdien er muligvis mere en farve feature end det er en tekstur feature, da det er et udtryk for den gennemsnitlige gråtone i billedet. Årsagen til at middelværdien beskrives i dette afsnit er, at middelværdien og variansen til sammen giver en idé om billedet. Variansen er tilgengæld et udtryk for forskelligheden i billedet. Hvis et billede har en høj varians, så betyder det, at der er en stor spredning mellem de højeste og laveste pixelværdier. Rent matematisk så beregnes middelværdien og variansen på følgende måde:

$$\mu_x = \frac{1}{N}\sum_{n=0}^{N} x_n \tag{7.4}$$

$$\sigma^2 = \sum_{n=0}^{N} \frac{(x_n - \mu_x)^2}{N} \tag{7.5}$$

På forhånd forventes det, at dyr som zebraer og giraffer vil have en relativ høj varians, hvorimod at dyr som strudse, oryxer og kuduer vil have en lavere varians. Dette forventes da zebraer og giraffer begge er dyr med et mønster, og derfor har en forskellighed i teksturen. Et plot af variansen og middelværdien ses på figur 7.7. Som forventet har både giraffer og zebraer generelt en højere varians end de andre dyr.



Figur 7.7. Figuren viser et plot af middelværdien (mean) og variansen (var). Som forventet er variansen for zebraer og giraffer generelt højere end for de andre dyr. Dataet brugt til at generere dette plot er det samme som blev brugt i figur 7.6

7.2.2 Sobel

En af de væsentlige tekstur features, som kan bruges til at adskille dyrene fra hinanden, er kanter. Kanter i et billede kan findes på forskellige måder. I dette program er der brugt Sobel kerner til at finde horisontale og vertikale kanter. Desuden er der også brugt en kant algoritme kaldet Canny Edge Detection, præsenteret af John F. Canny i 1986. Denne kant detekteringsmetode beskrives i afsnit 7.2.3.

En kant i et billede defineres som et sted, hvor der er en væsentlig ændring i gråtone værdien[Moeslund, 2009]. Måden hvorpå man rent praktisk finder dette i et billede, er ved at bruge en kerne. I dette tilfælde bruges en Sobel kerne som anvendes på et billede og outputtet beskriver hvor hurtigt eller langsomt intensiteten ændres. Sobel kernen som bruges er vist på figur 7.8.

-1	0	1
-2	0	2
-1	0	1

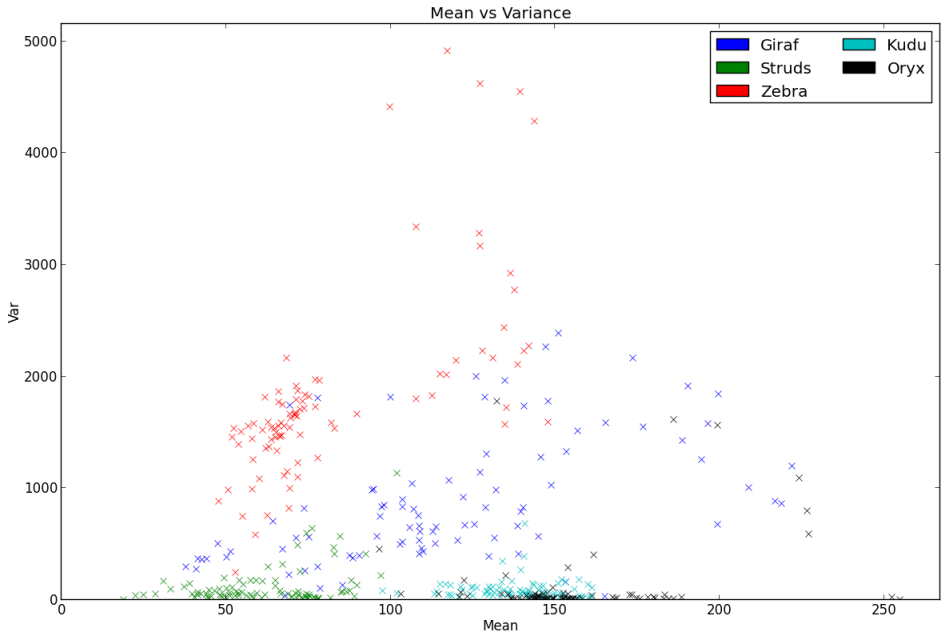
Vertikal

-1	-2	-1
0	0	0
1	2	1

Horisontal

Figur 7.8. Horisontal og vertikal Sobel kerne.

Når Sobel kernen anvendes vil resultatet være et billede med de horisontale kanter og de vertikale kanter. Et plot hvor antallet af kanter sammenlignes ses på figur 7.9. Igen forventes det at giraffer og zebraer har væsentlig flere kanter end de tre andre dyr. Desuden vil giraffer ofte have lige mange vertikale og horisontale kanter, hvorimod zebraer har et overtal af vertikale kanter.



Figur 7.9. Figuren viser et plot af de horisontale kanter (Sobelx) og vertikale kanter (Sobely). Dataet brugt til at generere dette plot er det samme som blev brugt i figur 7.6

7.2.3 Canny kantdetektering

I 1986 præsenterede John F. Canny sin kantdetekteringsalgoritme, som i dag ofte bruges til at finde kanter i et billede. Canny detekteringen bruger blandt andet Sobel kerner til at finde kanterne, men i modsætningen til beskrivelsen i afsnit 7.2.2, så er Canny detekteringen væsentlig mere detaljeret. Algoritmen består af følgende fem trin:

1. Udjævning af billede
2. Udregning af hældning

3. Fjernelse af ikke-maksimale værdier
4. Dobbelt thresholding
5. Efterbehandling

Det første trin består i at udjævne billedet ved at bruge en Gaussian kerne. I OpenCV bruges kernen som ses på figur 7.10.

$$K = \frac{1}{159}\begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

Figur 7.10. En 5x5 Gaussian kerne med en standard afvigelse på 1,4, hvilket er den anvendte kerne i algoritmen i OpenCV.

Det næste trin er en udvidelse af brugen af Sobel kernerne, som også beskrives i afsnit 7.2.2. Efter kernerne er anvendt på billedet, så beregnes størrelsen på hældningen (G). Dette gøres ved at finde den euklidiske afstand mellem den horisontale (G_x) og vertikale (G_y) hældning, som ses i ligning 7.6. Bagefter udregnes retningen på hældningen, hvilket fremgår af ligning 7.7.

$$|G| = \sqrt{G_x^2 + G_y^2} \tag{7.6}$$

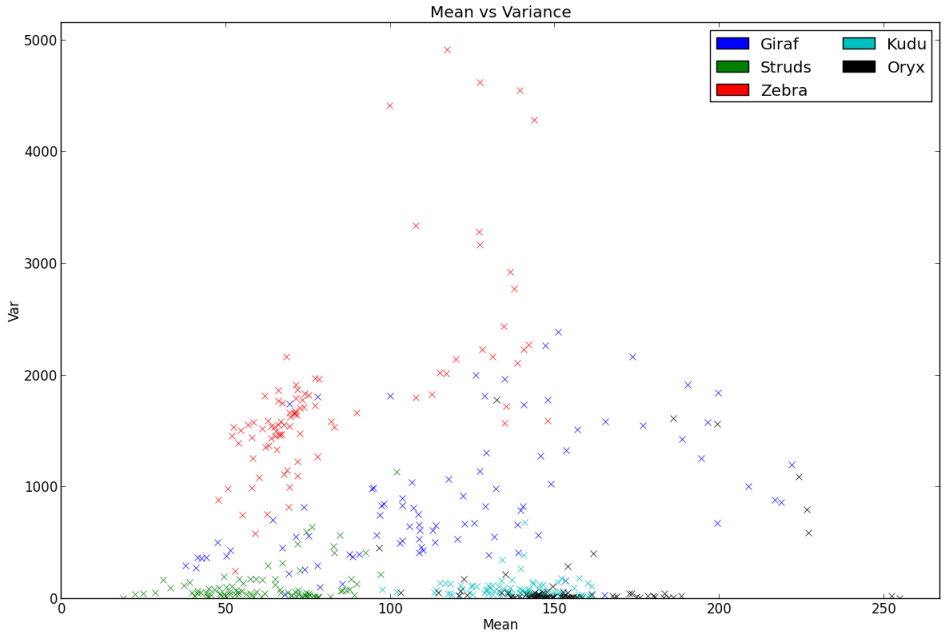
$$\theta = atan(\frac{|G_y|}{|G_x|}) \tag{7.7}$$

Trin tre består i at reducere størrelsen på kanterne der fremkommer af Sobel kernerne. Dette sørger for at kun helt tynde kanter er tilbage.

Det næste trin er en dobbelt thresholding. Her sorteres kanterne i kategorierne stærke, svage og ikke-eksisterende. Dette gøres ved at sætte to threshold værdier, nemlig en øvre og en nedre værdi.

Til sidst efterbehandles billedet, hvor de sammenhængende kanter som kun består af svage kanter fjernes.

På figur 7.11 ses et plot af antallet af kantpixels i forhold til hue værdien. Hue værdien er blot valgt for at lave et 2D plot. Som forventet er det hovedsageligt zebraer og giraffer som har kanter, netop på grund af deres mønster.



Figur 7.11. Figuren viser et plot af kanterne fundet ved brug af Canny algoritmen, sammenlignet med hue værdien af dyret. Dataet brugt til at generere dette plot er det samme som blev brugt i figur 7.6

Klassificering 8

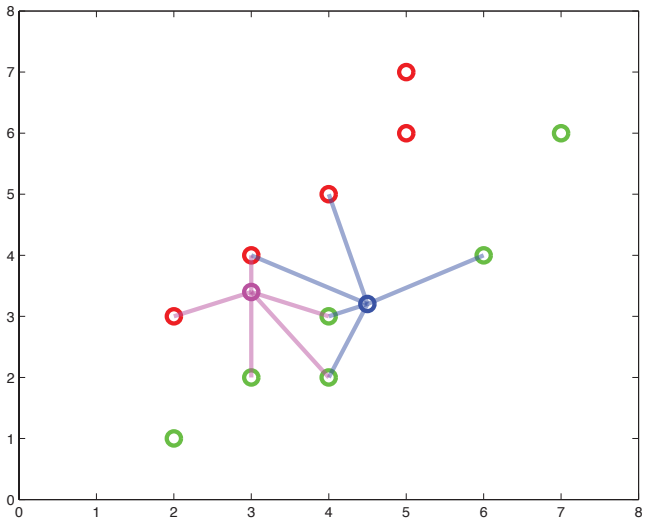
Klassificeringen er den del af et computer vision system, hvor det afgøres hvilken klasse en given sample hører til. I dette tilfælde handler det altså om, ved brug af featurene beskrevet i afsnit 7, at bestemme hvilket dyr som brugeren har taget et billede af. Med udgangspunkt i ønsket om at lave et simpelt system, så er en K-Nearest-Neighbour (KNN) algoritme brugt til klassificering af dyrene.

En KNN algoritme forsøger at tildele en featurevektor en klasse, ved at finde afstanden fra featurevektoren til de k nærmeste træningsvektorer. K'et repræsenterer altså antallet af naboer der bruges til sammenligning, når klassen for inputvektoren skal bestemmes. Matematisk forsøger KNN reglen at estimere a posteriori probability $P(\omega_i|\mathbf{x})$, hvilket er beskrevet i følgende ligning:

$$P_n(\omega_i|\mathbf{x}) = \frac{p_n(\mathbf{x},\omega_i)}{\sum_{j=1}^{c} p_n(\mathbf{x},\omega_j)} = \frac{k_i}{k} \tag{8.1}$$

hvor:
$p_n(\mathbf{x},\omega_i) = \frac{k_i/n}{V}$ er estimatet for joint probability $p(\mathbf{x},\omega_i)$, hvor V er volumen rundt om $\mathbf{x}$, inkluderende k_n ud af n samples. Det vil sige at volumen V udvides indtil den indeholder k samples.

Et eksempel med to klasser, klasse rød og grøn, er vist på figur 8.1. To nye samples ønskes klassificeret, hvilket er den blå og den lilla cirkel. I dette tilfælde bruges $k = 5$, hvilket er vist med afstandsstregerne.



Figur 8.1. Figuren et plot af to sæt træningsdata, nemlig klasserne grøn og rød. Desuden er to nye samples plottet ind, vist ved den blå og lilla cirkel. Ved brug af KNN algoritmen, med $k = 5$, klassificeres begge samples som værende den grønne klasse.

I eksemplet på figur 8.1 bliver begge af de nye samples klassificeret som tilhørende den grønne klasse. Dette virker umiddelbart korrekt for den blå sample, men ikke for den lilla. En løsning til dette problem kan være at vægte afstandene, således at de nærmeste naboer betyder mere for klassificeringen end de fjerneste.

For at finde afstanden fra den nye sample til al træningsdataet, så bruges den euklidiske afstand. Den euklidiske afstand findes ved følgende beregning:

$$d_2(A, B) = \sqrt{\sum_{i=0}^{m} (a_i - b_i)^2} \tag{8.2}$$

For at bestemme størrelsen af k, så er en test med forskellige størrelser af k udført. Resultatet af denne test blev, at $k = 3$ gav den bedste klassificering.

Træningsdata 9

For at træne klassificeringsalgoritmen kræves der en vis mængde mærket data. Da systemet bør fungere i al slags vejr, så vil den optimale træningsdata ligeledes bestå af billeder optaget i forskellige vejrsituationer. Der er dog ikke taget højde for dette, og al træningsdata er optaget i solskinsvejr.

Træningsdataet består af 310 billeder på 986x657 pixels. Billederne er taget med det samme Nikon D7000 som bruges til prototypen. Fordelingen af dyrene fremgår af tabel 9.1.

	Giraf	Struds	Zebra	Kudu	Oryx	I alt
Antal	78	44	58	46	84	310

Tabel 9.1. Fordelingen af træningsdata på de forskellige dyr.

For den bedst mulige genkendelse, så bør billederne tages fra forskellige vinkler og med forskellige afstande til dyrene. Dette er forsøgt opfyldt, der er dog hovedsageligt brugt billeder hvor midten af billedet svarer til midt på kroppen af dyret.

Al data er inkluderet på den vedlagte cd.

Prototypen 10

Som en del af projektet er en prototype blevet udviklet og bygget. Prototypen består af både hardware og software, og dette kapitel starter med en kort gennemgang af den samlede prototype. Gennemgangen er gentagelse af udviklingen fra de foregående kapitler, og er blot samlet her for at få et overblik.

Hardware:

- Stålstativ med påmonteret fladskærm. Stativet kan drejes omkring sig selv, og skærmen kan tippes op og ned
- Spejlreflekskamera med et 200 mm objektiv påmonteret midtpå bagved skærmen, således at kameraet bevæges sammen med skærmen
- Numpad tastatur påsat stativet til styring af programmet
- Bærbar computer forbundet til kamera, numpad tastatur og skærmen. Computeren er ikke påmonteret stativet, men står for sig selv

Software:

- Brugergrænseflade som kontrolleres ved brug af numpad tastaturet
- Segmentering bestående af et udklip af midten af billedet taget af brugeren
- Extraction af syv features: hue, saturation, middelværdi, varians, horisontal Sobel, vertikal Sobel og Canny kantdetektering
- KNN klassificering af featurevektoren
- Præsentation af resultatet

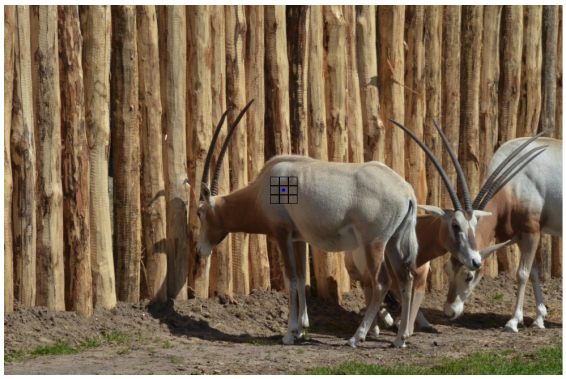
To billeder af prototypen ses på figur 10.1. Billederne er taget under test af systemet i Aalborg Zoo.

Figur 10.1. To billeder af prototypen, som er opstillet ved Savannen i Aalborg Zoo. På billedet til venstre ses det hvordan computeren er koblet til skærmen. På billedet til højre ses det hvordan kameraet er påmonteret stativet.

I de følgende to afsnit er implementeringen af brugergrænsefladen og genkendelsessoftwaren beskrevet.

10.0.4 Segmentering, feature extraction og klassificering

Den implementerede segmenteringsdel er afhængig af, at brugeren håndterer systemet korrekt. Da det ikke kan forventes at brugeren altid håndterer systemet helt korrekt, så er der taget en foranstaltning der skulle gøre det muligt at genkende korrekt alligevel. I stedet for at lave ét udklip af billedet, så laves der i stedet flere udklip. Selve segmenteringen fungerer altså ved at udklippe ni firkanter i et felt på 3x3 fra midten af billedet. Hver af disse firkanter er 15x15 pixels og bruges til at genkende dyret. Antallet af firkanter der udklippes fra hvert billede er bestemt ved test som ses i bilag A. På denne måde burde det være muligt at genkende et dyr, selvom kun fem af udklippene er af dyret. Et eksempel på et korrekt billede, hvor alle ni firkanter er indenfor dyret, og et hvor kun nogle af dem er indenfor dyret, er vist på figur 10.2.



Figur 10.2. Billedet til venstre viser et eksempel hvor alle ni firkanter er indenfor dyret. På billedet til højre er det seks af firkanterne som ikke indeholder baggrund.

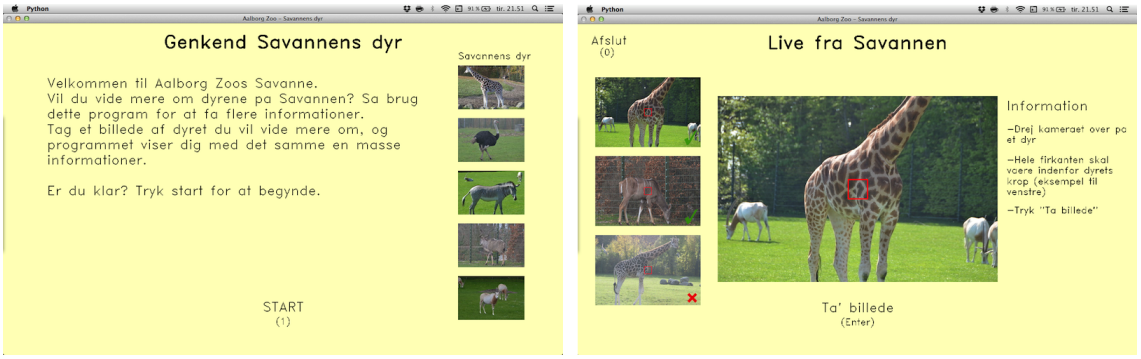
Efter segmenteringen bliver en featurevektor udtrukket af hver af de ni firkanter. Hver

firkant bliver så klassificeret ved brug af KNN algoritmen, og til sidst vurderes det hvilket dyr der er tale om. Hvis resultaterne fra de ni firkanter ikke peger mod et bestemt dyr, så bliver det endelige resultat en fejlgenkendelse. Vurderingen foretages på følgende måde:

1. Hvis mere end $k/2$ af resultaterne er den samme klasse, tildel firkanten den klasse. Ellers tildel fejlklassen til firkanten
2. Hvis mere end $k/2 \cdot n$ af resultaterne fra alle firkanterne er den samme klasse, hvor n er antallet af firkanter, tildel den overordnede genkendelse denne klasse. Hvis ikke, tildel fejlklassen.
3. Hvis klassen fra første betingelse er lig klassen fra anden betingelse, så genkendes dyret som denne klasse. Hvis ikke, så kan dyret ikke genkendes

10.0.5 Brugergrænsefladen

Brugergrænsefladen i et system tiltænkt bl.a. børn er meget vigtig. Et af kravene til prototypen er derfor også, at programmet skal være så intuitivt og selvforklarende, at brugeren selv kan bruge systemet uden anden forklaring end hvad der fremgår af skærmen. I afsnit 5 sættes behovet for antallet af forskellige interaktionsskærme til fem. Prototypens udgave indeholder dog seks skærme som fremgår af figur 10.3, 10.4 og 10.5. Den sjette skærm er en venteskærm der fremkommer når noget af informationsteksten læses op.



Figur 10.3. Billedet til venstre viser velkomstskærmen som brugeren mødes af. Billedet til højre viser liveview skærmen, med vejledning til hvordan prototypen bruges.

Figur 10.4. Billedet til venstre viser venteskærmen efter brugeren har taget et billede. Billedet til højre viser den popup som brugeren ser hvis der ikke kunne genkendes et dyr på billedet.

Figur 10.5. Billedet til venstre viser informationerne om det dyr der er genkendt. Billedet til højre viser venteskærmen der fremvises mens prototypen oplæser en af informationerne.

Flowet for brugen af prototypen forløber på følgende måde:

1. Brugeren ser velkomstskærmen og får en introduktion til hvad prototypen kan. Her kan brugeren trykke start, så snart han eller hun er klar.
2. Brugeren ser nu liveview skærmen som viser et direkte videostream fra kameraet. Samtidig informeres brugeren til højre omkring brugen af systemet. Til venstre på skærmen er der to eksempler på rigtige optaget billeder og et forkert billede. Brugeren kan her trykke ta' billede eller afslut. Hvis der trykkes afslut ryger brugeren tilbage til punkt 1.
3. Brugeren oplyses af en "vent venligst"besked, indtil systemet er færdig med at genkende dyret. Brugeren kan ikke gøre noget her. Hvis dyret er genkendt, så præsenteres brugeren for punkt 5, ellers præsenteres brugeren for punkt 4.
4. Brugeren bliver nu mødt af en fejlbesked der oplyser, at systemet ikke kunne genkende et dyr ud fra billedet. Brugeren kan enten trykke prøv igen og springe til punkt 2, eller trykke afslut og springe til punkt 1.
5. Brugeren ser nu en informationsskærm der fortæller om det dyr der blev genkendt ud fra billedet. Til venstre vises øverst det billede brugeren selv tog, og de to andre billeder er eksempler på det samme dyr. Resten af skærmen er fyldt med forskellige informationer om dyret. Såfremt brugeren trykker på et tal mellem 2 og 6 (svarende til tallene på skærmen), så oplæses teksten for brugeren. Til sidst kan brugeren enten trykke afslut og blive sendt tilbage til punkt 1, eller trykke prøv igen og starte ved punkt 2.

Resultater fra prototypetest 11

For at kunne reflektere over systemets funktionalitet er det nødvendigt at teste systemet. Systemet er blevet testet på to måder. For at kende en teoretisk opnåelig genkendelsesrate, så testes der på træningsdata. Dette fremgår af afsnit 11.1. Den endelige test af prototypen udføres "i marken", hvor prototypen er opsat i Aalborg Zoo og afprøves af testpersoner. Fremgangsmåden og resultaterne af denne test findes i afsnit 11.2.

11.1 Test på træningsdata

For at finde den øvre grænse for systemet, så laves der en test på træningsdataet. Træningsdataet består af 310 billeder fordelt på de fem dyr. Fordelingen er ikke lige, og den præcise fordeling fremgår af tabel 11.1.

	Giraf	Struds	Zebra	Kudu	Oryx	I alt
Antal	78	44	58	46	84	310

Tabel 11.1. Fordelingen af træningsdata på de forskellige dyr.

For at teste systemet bruges der 2-gange krydsvalidering. Dette fungerer ved en tilfældig opdeling af træningsdataet i to datasæt. Den tilfældige opdeling sørger for, at halvdelen af billederne fra hvert dyr er i henholdsvis datasæt d_1 og d_2. Efter opdelingen testes systemet to gange. I første test bruges d_1 som træning og d_2 som valideringsdata. I test to byttes der om på datasættene, og d_1 bruges som valideringsdata og d_2 bruges som træningsdata. Resultaterne af testene fremgår af tabel 11.2.

	Korrekt genkendt	Forkert genkendt	Ikke genkendt
Test 1	83,23%	13,55%	3,23%
Test 2	90,32%	5,16%	4,52%
Gennemsnit	86,78%	9,36%	3,88%

Tabel 11.2. Resultaterne fra 2-gange krydsvalideringen. Korrekt genkendt betyder, at dyret er genkendt som det dyr det er. Forkert genkendt betyder, at dyret er genkendt som et andet dyr end det er. Ikke genkendt betyder, at dyret ikke er genkendt som noget dyr, men blot har fået non-klassen.

11.2 Prototypetest i Aalborg Zoo

Den endelige prototypetest blev udført i Aalborg Zoo. Her blev prototypen sat op ved Savannen og testet af 16 testpersoner. 12 af testpersonerne var fundet via et opslag på Facebook, og var altså delvis kendte personer, men dog personer uden indgående kendskab til systemet. De sidste 4 personer var tilfældige personer som testede systemet. Testen foregik på følgende måde:

1. En kort introduktion til systemet gives. Introduktionen indeholder informationer om hvad prototypen kan, at det er en del af et kandidatspeciale, at det er en første prototype og at de til sidst bedes udfylde et spørgeskema.
2. Testpersonen bruger systemet så længe som de har lyst til. Informationer angående brug af prototypen fremgår af skærmen.
3. Efter testen udleveres et kort spørgeskema, som også fremgår af bilag B.

Indholdet på skærmen, som testpersonen bruger, vises også på computeren. Testen kan derfor overvåges, uden testpersonen føler sig overvåget. I enkelte tilfælde forstod testpersonen ikke opgaven korrekt, og efter at have prøvet selv et par gange, så informeres testpersonen om hvordan prototypen skal bruges. Foruden spørgeskemaerne gemmes de tekniske data også fra testen. Dette foregår ved, at hver gang testpersonen tager et billede, som gemmes billedet på computeren, og navngives efter hvilket dyr systemet genkender ud fra billedet. Disse billeddata kan efterfølgende analyseres. Et par billeder fra testdagen ses på figur 11.1.

Figur 11.1. Tre billeder taget på testdagen, hvor en af testpersonerne ses i funktion.

11.2.1 Resultater af prototypetest i Aalborg Zoo

For at evaluere resultaterne opstilles der tre test:

1. Alle billeder taget af testpersonerne bruges
2. Kun korrekt tagede billeder af testpersonerne bruges (her kan resultatet "korrekt afvist"ikke forekomme)
3. Billeder taget af en person med indgående kendskab til systemet bruges

På baggrund af dette er resultaterne i tabel 11.3 opnået. Korrekt genkendt beskriver et billede med et dyr som genkendes som det rigtige dyr. Forkert genkendt beskriver enten et billede med et andet dyr end det genkendes som, eller et billede uden et dyr, men som genkendes som et dyr. Korrekt afvist beskriver et billede uden et dyr, og hvor systemet heller ikke kan genkende et dyr (non-klasse). Forkert afvist beskriver et billede af et dyr, hvor systemet klassificerer det som ikke indeholdende et dyr.

	Billeder	Korr. genkendt	Fork. genkendt	Korr. afvist	Fork. afvist
Test 1	106	34,91%	29,25%	6,69%	29,25%
Test 2	60	58,33%	15,00%	0,00%	26,67%
Test 3	21	66,67%	14,29%	0,00%	19,05%
Gennemsnit	62,33	53,30%	19,51%	2,23%	24,99%

Tabel 11.3. Resultaterne fra prototypetesten. Korrekt forkortes som korr., og forkert forkortes som fork. Test 1 indeholder alle billeder taget af testpersoner. Test 2 indeholder korrekt tagede billeder af testpersonerne. Test 3 indeholder billeder taget af en person med indgående kendskab til systemet.

Resultaterne er meget forskellige fra de tre udførte test. Som forventet fungerer systemet bedst, når en person med indgående kendskab til systemet er testperson. Dernæst er der en stor forbedring af resultatet når kun de korrekte tagede billeder medtages i beregningerne. Dette giver et klart syn af, at testpersonerne enten ikke altid har forstået at bruge systemet, eller ikke har kunnet finde ud af at bruge systemet.

På figur 11.2 og 11.3 ses eksempler på hvilke billeder testpersonerne har taget med prototypen. På figur 11.3 fremgår det tydeligt hvorfor systemet ikke altid kan genkende dyret. Systemet sørger dog ikke altid for at afvise korrekt, hvilket giver et dårligere resultat.

Figur 11.2. To billeder taget af testpersoner på testdagen i Aalborg Zoo. Begge billeder er taget korrekt, og genkendt korrekt som henholdsvis en giraf og en zebra. De sorte firkanter viser hvilke udklip fra billedet der bruges til at genkende dyret.

Figur 11.3. To billeder taget af testpersoner på testdagen i Aalborg Zoo. Begge billeder er taget forkert, hvilket ses ved, at de sorte firkanter ikke er indenfor dyret. De sorte firkanter viser hvilke udklip fra billedet der bruges til at genkende dyret.

11.2.2 Spørgeskemaresultater

Efter endt test af prototypen blev testpersoner bedt om at udfylde et spørgeskema. Efterfølgende er resultaterne inddelt i seks aldersgrupper. Inddelingen og antallet af testpersoner i hver aldersgruppe fremgår af tabel 11.4.

	5-14	15-24	25-34	35-44	45-54	65-74	5-74
Testpersoner	3	4	4	3	1	1	16

Tabel 11.4. Fordelingen af testpersonerne på seks forskellige aldersgrupper.

Alle svarene fra spørgeskemaundersøgelsen findes i bilag B. Af resultaterne kan fremhæves følgende:

- De fleste synes at prototypen var nem eller meget nem at bruge
- De fleste synes programmet fungerede nogenlunde flydende eller bedre
- De fleste synes programmet genkendte dyrene nogenlunde korrekt eller bedre
- De fleste synes prototypen hjælper børn og andre med læsevanskeligheder nogenlunde eller bedre
- Alle kunne finde på at bruge systemet hvis det var at finde i en zoo

Desuden kan følgende kommentarer og forslag til forbedringer fremhæves:

- Mulighed for zoom
- Tydeligere instruktion til systemet
- Oplæsning af al tekst
- Bedre genkendelse
- Sjovt at interagere med dyrene

11.3 Analyse af testresultater

På baggrund af de fire forskellige test, følger her en analyse af resultaterne. Resultaterne fra 2-gange krydsvalideringen er umiddelbart meget tilfredsstillende. Med en gennemsnitlig

genkendelsesrate på 86,78% viser det, at systemet med dets nuværende konfiguration, kan genkende dyrene tilfredsstillende. Desværre er resultaterne fra prototypetesten i Aalborg Zoo ikke nær så tilfredsstillende. Her er de dårligste resultater helt nede på 34,91%, hvilket på ingen måder er tilfredsstillende. Det ses dog, at en af årsagerne er, at der ikke tages korrekte billeder med systemet. Hvis kun de rigtigt optaget billeder medtages, så forbedres genkendelsen væsentlig, men er dog stadig relativt lav med en værdi på 58,33%. Dette forbedres lidt mere, når en person med indgående kendskab bruger systemet, her er resultatet 66,67%. Den umiddelbare konklusion på disse tal er, at brugerne ikke kan finde ud af bruge systemet korrekt. Med denne tanke i hovedet, er det dog forunderligt, at de fleste af testpersonerne anså det for værende nemt eller meget nemt at bruge systemet. Det antyder altså, at brugeren tror at han/hun bruger systemet korrekt, hvilket dog viser sig ikke at være rigtigt. Iagttagelser under testen medførte følgende overvejelser:

- Nogle af dem havde problemer med at håndtere prototypen. Dette var især gældende for børnene. Årsagen er formentlig størrelsen på stativet der gør det en smule svært at bevæge
- Mange af testpersoner enten glemte, vidste ikke eller ignorerede måden hvorpå systemet skulle bruges. Det medførte mage optagelser hvor den røde firkant i midten ikke kun indeholdt et dyr
- Af og til var dyrene enten for tæt på eller for langt væk
- Dyrene bevægede sig nogle gange så hurtigt, at billederne blev meget uskarpe
- Pga. den skarpe sol, så opstod der af og til meget skarpe skygger på nogle af dyrene. Dette er et problem for de dyr som ikke har kanter, da der pludselig vil være en kant på dyret mellem det lyse og det mørke
- Af og til tog brugeren et billede af hovedet af dyret, eller af benene. Da disse situationer ikke er tilstrækkelig dækket af træningsdataet, så medførte det ofte en fejlgenkendelse

Resultaterne fra spørgeskemaet viste, at børn og voksne i alle aldre er interesserede i at bruge systemet. Desuden er der flere kommentarer som indeholder positive tilbagemeldinger angående idéen. Dette lægger op til, at et velfungerende system vil være eftertragtet i en zoo.

Analyse af krav og forbedringer

12

Efter færdigudvikling og test af systemet, så er det tid til at undersøge hvorvidt systemet opfylder kravene stillet i kapitel 2.2. For nemhedens skyld, så kommer kravene lige en gang til:

1. Prototypen skal kunne genkende og adskille de fem forskellige dyr på Savannen i Aalborg Zoo
2. Prototypen vil bestå af en skærm på et stativ, med et påmonteret kamera og numpad tastatur. Alt sammen koblet til en bærbar computer
3. Prototypen har to grader af frihed, nemlig horisontalt og vertikalt
4. Prototypen indeholder en gennemarbejdet brugergrænseflade med brugervenlighed som et fokus, og hvor brugeren kan benytte systemet uden menneskelig vejledning
5. Responstiden må ikke være på mere end 3 sekunder
6. Genkendelsesraten skal være på mindst 85% ved krydsvalidering
7. Genkendelsesraten skal være på mindst 75% ved den endelige test

Et tilsvarende skema som evaluerer hvert krav følger her:

1. Prototypen kan genkende og adskille alle fem dyr på Savannen
2. Prototypen består netop af en skærm på et stativ med et kamera og et numpad tastatur. Alt sammen er forbundet til computeren
3. Prototypen har netop to grader af frihed. Den kan roteres omkring sig selv og tiltes op og ned. Dette medfører at prototypen kan dække hele Savannen
4. Prototypen indeholder en gennemarbejdet brugergrænseflade, dog med enkelte mangler, herunder æøå. Test af systemet viser dog, at brugeren kun tager korrekte billeder 56,60% af gangene, hvilket tyder på at vejledningen ikke er god nok.
5. Responstiden er en del svingende fra 2 og helt op til 10 sekunder. En af årsagerne kan blandt andet være, at programmet ved en fejl blev eksekveret gennem iPython, hvilket gør programmet langsommere. Desuden tyder det på, at der er noget memory lækage, da programmet bliver langsommere jo længere det har kørt
6. Genkendelsesraten er på 86,78% og opfylder derfor kravet
7. Genkendelsesraten er på 34,91% og opfylder derfor slet ikke kravet

12.1 Forbedringer af systemet

Efter at have udført flere test af systemet har det medført en masse idéer til forbedring af systemet. Forbedringerne inddeles i tre kategorier, nemlig brugergrænsefladen, hardware og genkendelsesmodulet.

Hardware

Hardwaren var rimelig afgrænset på forhånd, og der blev derfor ikke taget mange overvejelser indenfor dette felt. Dette medfører selvfølgelig, at der er nogle forslag til forbedringer indenfor dette punkt også:

- Touchskærm
- Lysstærkere skærm, eller en form for skygge for solen
- Kamera/objektiv med mulighed for optisk zoom via computeren
- Implementering af digital zoom og autofokus
- Bedre stativ som nemmere kan håndteres af børn, med mulighed for at justere højden

Brugergrænseflade

Som en naturlig del efter prototypetesten, så er der en del forbedringer til systemet, enten foreslået af testpersonerne eller iagttagelser gjort under testen:

- Al teksten skal kunne læses op, eventuelt med syntetisk tale
- Systemet skal fungere på flere sprog
- Brugervejledningen skal gøres tydeligere, således at brugeren ikke er i tvivl om hvornår et billede er brugbart og hvornår det ikke er. En form for feedback hvor brugeren oplyses om, at dyret fylder tilpas meget af skærmen, ville være en stor forbedring

Genkendelsesmodulet

Genkendelsesraten ved brug af krydsvalideringen var tilfredsstillende. Dette var dog ikke gældende til prototypetesten. Derfor bør eventuelle forbedringer også overvejes her:

- Tilføjelse af flere features, fx en skaleringsuafhængig feature som SURF
- Analyse af de features der bruges. Eventuelt vil en form for feature space reduktion kunne gøre systemet bedre
- Hvis KNN beholdes, så bør en vægtning af afstandene tilføjes
- Eventuelt bruge en anden mere avanceret klassificeringsalgoritme

Litteratur

Claus B. Madsen, **2011**. Claus B. Madsen. *Koldinghus Augmented: The Walls Remember*. `http://vbn.aau.dk/files/58116502/ProjektBeskrivelseKoldinghusAugmentedv20.pdf`, 2011. Downloaded: 29-05-2014.

Moeslund, **2009**. Thomas B. Moeslund. *Image and Video Processing, 2nd Edition*. ISBN: 978-87-992732-1-8. Aalborg University, 2009.

Rother et al., **2004**. Carsten Rother, Vladimir Kolmogorov and Andrew Blake. *GrabCut: Interactive foreground extraction using iterated graph cuts*. Proceedings SIGGRAPH '04 ACM SIGGRAPH 2005 Papers, pages 309–314, 2004.

The Metropolitan Museum of Art, **2014**. The Metropolitan Museum of Art. *Audio Guide*. `http://www.metmuseum.org/visit/plan-your-visit/audio-guide`, 2014. Downloaded: 29-05-2014.

ZSL London Zoo, **2014**. ZSL London Zoo. *London Zoo App*. `https://itunes.apple.com/gb/app/london-zoo/id394794108?mt=8`, 2014. Downloaded: 29-05-2014.

Test af antal udklip og størrelsen på k

A

For at teste hvilken størrelse af k der er bedst, og hvor mange udklip der bør bruges i segmenteringen, er der udført en test. Denne test udføres som en 2-gange krydsvalidering hvor der prøves med henholdsvis 3 og 9 udklip, samt en k på 3, 5, 7, og 9. Dette giver i alt 8 kombinationer af test, hvor resultaterne fremgår af tabel A.1

	$k = 3$	$k = 5$	$k = 7$	$k = 9$
4 udklip	83,32/14,19/2,58%	85,16/11,62/3,22%	83,23/12,26/4,51%	80,64/12,90/6,46%
9 udklip	90,32/4,52/5,16%	89,68/3,87/6,45%	90,32/2,58/7,1%	89,68/3,78/6,54%

Tabel A.1. Genkendelsesraten afhængig af antallet af udklip og størrelsen af k. De tre tal i hvert felt står for genkendt korrekt/genkendt som non-klasse/genkendt forkert. Resultatet viser, at $k = 3$ og $k = 7$ kombineret med 9 udklip giver den bedste genkendelsesrate. $k = 3$ giver dog samtidig en lavere forkert genkendelse end $k = 7$.

Spørgeskema resultater B

Spørgeskema ang. Dyregenkendelse på Aalborg Zoos Savanne

Aldersgruppe: 5 – 14; 3 Mand: 1 Kvinde: 2

De følgende spørgsmål bedes du vurdere fra 1 til 5, hvor 1 er dårligst:

	1	2	3	4	5
Hvor nemt var det at bruge programmet?	0	0	0	1	2
Hvor flydende fungerede programmet?	0	0	2	0	1
Hvor godt var programmet til at genkende dyrene?	0	0	2	1	0

Programmet er bl.a. tiltænkt børn og folk med læsevanskeligheder, så de kan få oplæst informationer.

Hvor godt synes du programmet løser dette (1 er dårligst)?	1	2	3	4	5
	0	0	0	1	2

Kunne du forestille dig at bruge sådan et program i fx en zoo? Ja: 3 Nej: 0

Har du nogle forslag til forbedringer?

- Den kunne godt være lidt hurtigere til at genkende dyrene

Andre kommentarer?

- Det var sjovt

Figur B.1. Spørgeskema fra aldersgruppen 5-14.

Spørgeskema ang. Dyregenkendelse på Aalborg Zoos Savanne

Alder: 15 – 24; 4 Mand: 1 Kvinde: 3

De følgende spørgsmål bedes du vurdere fra 1 til 5, hvor 1 er dårligst:

	1	2	3	4	5
Hvor nemt var det at bruge programmet?	0	0	0	1	3
Hvor flydende fungerede programmet?	0	0	0	3	1
Hvor godt var programmet til at genkende dyrene?	0	0	1	2	1

Programmet er bl.a. tiltænkt børn og folk med læsevanskeligheder, så de kan få oplæst informationer.

Hvor godt synes du programmet løser dette (1 er dårligst)?	1	2	3	4	5
	0	0	1	0	3

Kunne du forestille dig at bruge sådan et program i fx en zoo? Ja: 4 Nej: 0

Har du nogle forslag til forbedringer?

- Evt. zoom funktion
- Hurtig brugsforklaring ved tryk på start. Enten pop-up tekst eller oplæsning
- Zoom funktion

Andre kommentarer?

- Evt. headset til lyden

Figur B.2. Spørgeskema fra aldersgruppen 15-24.

Spørgeskema ang. Dyregenkendelse på Aalborg Zoos Savanne

Alder: 25 – 34; 4 Mand: 3 Kvinde: 1

De følgende spørgsmål bedes du vurdere fra 1 til 5, hvor 1 er dårligst:

	1	2	3	4	5
Hvor nemt var det at bruge programmet?	0	0	1	1	2
Hvor flydende fungerede programmet?	0	0	0	3	1
Hvor godt var programmet til at genkende dyrene?	0	1	2	1	0

Programmet er bl.a. tiltænkt børn og folk med læsevanskeligheder, så de kan få oplæst informationer.

Hvor godt synes du programmet løser dette (1 er dårligst)?	1	2	3	4	5
	0	0	2	0	1

Kunne du forestille dig at bruge sådan et program i fx en zoo? Ja: 4 Nej: 0

Har du nogle forslag til forbedringer?

- Det kunne være sjovt med genkendelse af det enkle dyr og nogle "personlige" oplysninger om dét – evt. alder, navn, hvor den er fra osv.
- Noget mere æstetik
- At den kan læse alt tekst. Ellers god idé
- Bedre genkendelse, zoom kunne evt. løse dette

Andre kommentarer?

- Sjovt lille program
- Interaktion er sjovt. Denne løsning giver følelse af at sigte og skyde et dyr = sjovt

Figur B.3. Spørgeskema fra aldersgruppen 25-34.

Spørgeskema ang. Dyregenkendelse på Aalborg Zoos Savanne

Alder: 35 – 44; 3 Mand: 1 Kvinde: 2

De følgende spørgsmål bedes du vurdere fra 1 til 5, hvor 1 er dårligst:

	1	2	3	4	5
Hvor nemt var det at bruge programmet?	0	0	0	0	3
Hvor flydende fungerede programmet?	0	0	1	2	0
Hvor godt var programmet til at genkende dyrene?	0	2	0	1	0

Programmet er bl.a. tiltænkt børn og folk med læsevanskeligheder, så de kan få oplæst informationer.

Hvor godt synes du programmet løser dette (1 er dårligst)?	1	2	3	4	5
	0	0	0	0	3

Kunne du forestille dig at bruge sådan et program i fx en zoo? Ja: 3 Nej: 0

Har du nogle forslag til forbedringer?

- Bedre genkendelse

Andre kommentarer?

- Genialt projekt! Fortsæt det gode arbejde
- Der var lidt mangler i genkendelsesdelen på både dyr i bevægelse + dyr på afstand. Hvis det skal anvendes til børn vil det være smart at lave det i lav højde – evt. stander med fleksibel højde, i stedet for skammel. Stort potentiale – især ved dyr man ikke kender – men kræver større genkendelsesgrad. Informationerne var relevante (godt niveau). Tænkte først at det ville være smart hvis man skulle trykke på skærmen, for at tage billede – men efter lidt brug fungerede det "eksterne" tastatur fint, men svært at bruge for fx børn.

Figur B.4. Spørgeskema fra aldersgruppen 35-44.

Spørgeskema ang. Dyregenkendelse på Aalborg Zoos Savanne

Alder: 45 – 54; 1 Mand: 0 Kvinde: 1

De følgende spørgsmål bedes du vurdere fra 1 til 5, hvor 1 er dårligst:

	1	2	3	4	5
Hvor nemt var det at bruge programmet?	0	0	0	0	1
Hvor flydende fungerede programmet?	0	0	0	0	1
Hvor godt var programmet til at genkende dyrene?	0	0	0	1	0

Programmet er bl.a. tiltænkt børn og folk med læsevanskeligheder, så de kan få oplæst informationer.

Hvor godt synes du programmet løser dette (1 er dårligst)?	1	2	3	4	5
	0	0	0	0	1

Kunne du forestille dig at bruge sådan et program i fx en zoo? Ja: 1 Nej: 0

Har du nogle forslag til forbedringer?

- Zoom funktion

Andre kommentarer?

Figur B.5. Spørgeskema fra aldersgruppen 45-54.

Spørgeskema ang. Dyregenkendelse på Aalborg Zoos Savanne

Alder: 65 – 74; 1 Mand: 0 Kvinde: 1

De følgende spørgsmål bedes du vurdere fra 1 til 5, hvor 1 er dårligst:

	1	2	3	4	5
Hvor nemt var det at bruge programmet?	0	0	0	0	1
Hvor flydende fungerede programmet?	0	0	0	0	1
Hvor godt var programmet til at genkende dyrene?	0	0	0	1	0

Programmet er bl.a. tiltænkt børn og folk med læsevanskeligheder, så de kan få oplæst informationer.

Hvor godt synes du programmet løser dette (1 er dårligst)?	1	2	3	4	5
	0	0	0	0	1

Kunne du forestille dig at bruge sådan et program i fx en zoo? Ja: 1 Nej: 0

Har du nogle forslag til forbedringer?

Andre kommentarer?

- Virker fint også ved børn og ordblinde!

Figur B.6. Spørgeskema fra aldersgruppen 65-74.