Artificiell intelligens I, 5p

Laboration 2 – Fördjupning i perception och objektigenkänning

Martin Burström [dit02mbm]
Robert Eriksson [dit02ren]
Filip Sjögren [dit02fsn]

Handledare:
Therese Edvall
Daniel Ölvebrink

2009-07-06 22:13

Abstract

An intelligent agent needs to percept its environment to navigate and interact with it. Perhaps the most important input source is the vision. The agent gets an image of the environment through a camera. First, the agent will process the image to reduce noise. Then, it will apply an algorithm to recognize objects. It can either use a brightness-based technique or a feature-based. The brightness-based uses the brightness value of each pixel in the image to compare it with values for known objects in its knowledge base. The feature-based technique picks out edges and other features to compare with the known objects. Common applications of object recognition include hand written text recognition, face recognition and other types of object recognition.

Innehållsförteckning

Introduktion till ämnet

Med mänsklig perception menar man hur hjärnan tar emot och bearbetar information från sinnesintryck och detsamma gäller i princip datorbaserad perception. Det slutgiltiga målet med datorbaserad perception är att skapa en dator som upplever och reagerar på samma sätt som en hjärna. Frågan är, kommer det ska lyckas? Ett steg i den riktningen är en så kallad mjukvaruagent som kan urskilja och känna igen olika objekt med hjälp av igenkänningsalgoritmer. Denna rapport kommer att handla om just det, hur en agent kan urskilja och känna igen visuella objekt.

Syfte

Syftet med uppgiften är att fördjupa sig inom ämnet artificiell objektperception samt att redovisa en övergriplig och lättförståelig bild av ämnet. Redovisning av ämnet kommer att göras skriftlig i form av denna rapport samt i en muntlig redovisning.

Metodbeskrivning

I huvudsak kommer vi att använda oss av litteraturstudier. Kursboken är den centrala källan. Vi kommer att sammanställa materialet i det aktuella kapitlet från boken samt resonera själva kring ämnet.

Litteraturstudie, teoretisk fördjupning

Introduktion till perception inom AI

Människan använder sig av sina fem sinnen för att uppfatta och förstå världen runt henne. Inom AI perception har man försökt att efterlikna dessa mänskliga sinnen i form av olika sensorer och i dags dato finns det en hel del sensorer som är lika kraftiga som de mänskliga och i vissa fall även bättre. Sensorer inom syn, hörsel och känsel är exempel på sensorer som är kraftigare än de mänskliga sinnena. Däremot finns det än så länge inga sensorer som kommer upp i samma nivå som människans lukt- och smaksinnen. Vi kommer att inrikta oss på sensorer inom synområdet och hur de kan användas för att känna igen olika objekt.

En AI agent kan använda sig av sin perception på två olika sätt, *feature extraction* och *model-based approach*.

Feature extraction approach

Med detta angreppssätt fungerar agenten på så sätt att den utifrån input som den får från sina sensorer känner av och plockar ut olika särdrag i strömmen av data från och skickar dessa direkt till programmet i agenten som styr hur agenten ska reagera. Hur det fungerar kommer vi till senare. Detta gör att agenten kan reagera relativt snabbt. Ett exempel från djurvärlden som utnyttjar sig av detta angreppssätt är flugor som kopplar sina synintryck direkt till sina muskler och på så sätt kan reagera extremt snabbt.

Model-based approach

Med denna metod använder agenten inputdatan från sensorerna för att skapa en modell av världen som den befinner sig i. Detta görs efter en specifik matematisk formel:

$W = F^{-1}(S)$

W är världen som ska beskrivas. F^{-1} är en definierad funktion och S är inputdata från sensorn. Problemet för agenten är att inputdatan som registreras och skall analyseras är alldeles för mycket information för att kunna behandlas även för en superdator. Men all information om omvärlden krävs inte för att agenten ska kunna skapa sig en ungefärlig modell av sin omgivning. Därför filtrerar agenten bort onödig indata. T ex behöver agenten inte veta exakt vart varje hårstrå sitter på en hund utan endast att det är en hund som befinner sig framför dess sensor.

Hur en bild blir till

”Syn” tar upp reflekterat ljus från objekt i våran omgivning. Sedan, på ett bildplan, skapas en tvådimensionell bild. Bildplanet är även täckt med fotokänsligt material; Rhodopsinmolekyler på näthinnan hos en människa, silveroxider i en filmrulle och en ”charged-coupled device” (CCD) i en digitalkamera. På varje plats i CCD’n finns det en pixel som reagerar på ljus under en viss tid. Bildplanet är uppdelat i en digitalkamera, uppdelat ungefär som ett rektangulärt nät (detta nät kan innehålla ungefär fem miljoner pixlar). Ögat har ett liknande nät fast med 100 miljoner stavar och 50 miljoner tappar. Med ett öga kan man se en stor omgivning men, bildplanet är relativt litet så här gäller det att fokusera ljuset från omgivningen till/på bildplanet. Detta kan göras med eller utan lins. Hur som helst gäller det att ha koll på geometrin så att vi kan hitta varje punkt från omgivningen på bildplanet.

Det enklaste sättet att skapa en bild är nog med en pinnhålskamera, även kallad camera obscura. Detta är ett bra exempel på bilder utan lins. Kameran består av en pinnhålsöppning, på framsidan av lådan och ett bildplan på motsatta sidan av öppningen. Märk väl att bilderna i en pinnhålskamera blir inverterade från verkligeheten. D.v.s. vänster blir höger och ner blir upp. Vanliga ögon och mer moderna kameror (än en camera obscura) använder en lins. En lins är mycket vidare än ett litet pinnhål och kan därför släppa in mer ljus. Nackdelen med att använda en lins är att inte hela bilden/omgivningen inte kan vara i fokus samtidigt. För att fokusera ändrar ögats lins form medan en kameralins rör sig (framåt eller bakåt). Ljus är en oerhörd viktig faktor när det gäller syn. För utan ljus så skulle alla objekt vara enhetligt mörka, oavsett hur intressant omgivningen är.

Tidiga bildprocessoperationer

Alla bilder har någon sort brus i sig till en början. För att reducera detta finns det två metoder som på ett tidigt plan kan användas för att effektivt ”städa” bilden. Dessa metoder kallas ”low-level”-metoder eftersom de oftast är de första redigeringsfunktioner som används på bilden. Dessa metoder använder väldigt lite datorkraft eftersom de kan användas på ett separat område av pixlar och behöver heller inte veta vad bilden föreställer.

Smoothing

Med smoothing menar man inom bildbehandling att man använder sig av angränsande pixlar för att förutspå hur nästkommande pixel ska se ut. Denna teknik används för att skapa en jämnare färg över hela bilden genom att ta medelvärdet av en pixels grannar och på så sätt hålla nere extremvärdena på pixlarna.

Edge detection

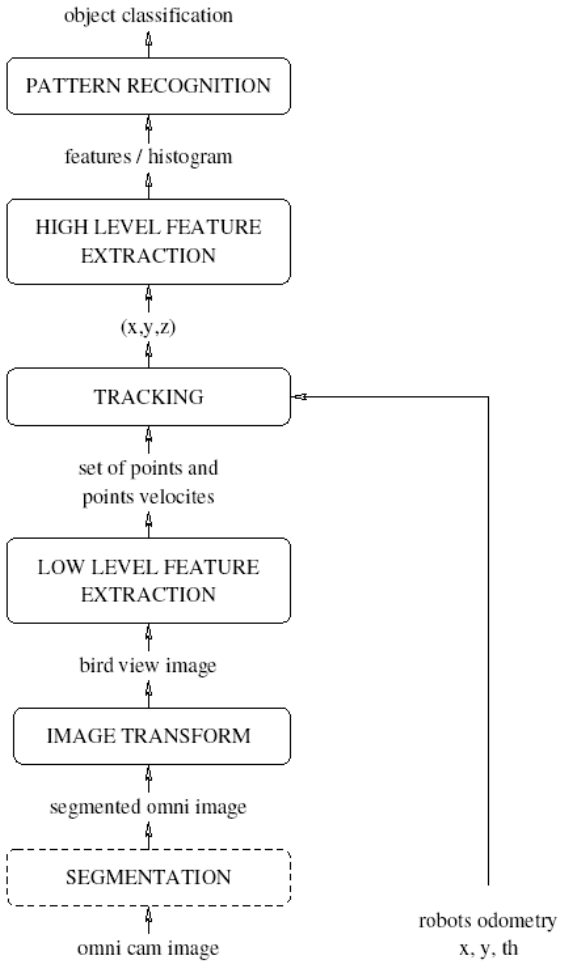
En annan ”low-level” – operation är edge detection. Som det låter används denna metod för att hitta olika kanter i bilden och filtrera fram dessa. Detta gör man för att höja abstraktionsnivån på bilden genom att plocka bort alla färger och endast spara det mest väsentliga egenskaperna. Agenten lokaliserar kanter genom att lokalisera områden i bilden där det är stor skillnad i ljusstyrka, det vill säga i hörn och kanter.

Datorbaserad objektigenkänning

Det behöver inte vara just objekt som ska kännas igen, det kan även vara rörelser, handlingar, miner etc. etc. Denna rapport kommer dock att handla om stillbildsobjekt. Det handlar dels om att lokalisera ett objekt i en bild (”var på bilden finns text som ska undersökas?”), dels om att kunna avgöra vilket objekt det är (”är det en 8 eller ett B”)Det finns en mängd praktiska tillämpningar av datorbaserad objektigenkänning: biometrisk (fingeravtryck, ögonskanning etc.), skriftigenkänning (avläsning av formulär, signaturer etc), kunna urskilja ett objekt bland många i en bild, och mycket mer.

Tekniken för objektigenkänning går huvudsakligen ut på att man lär agenten hur ett objekt ser ut, en sorts ”mall”, och sedan jämföra objekt som ska undersökas med de mallar som finns lagrade. Hur man lär agenten, och framför allt hur man får den att jämföra och känna igen, finns det många olika tekniker för.

Det första man måste göra är att dela upp en bild i olika delar för olika objekt som finns på bilden (*image segmentation*). En bild som innehåller en bil vi vill identifiera innehåller oftast andra objekt också, t.ex. bara en så enkel sak som bakgrunden. Efter man har plockat ut alla objekt kan man jämföra dem med de befintliga mallarna en och en. Denna bottom-up-approach ger dock inte bäst resultat i dagsläget, så man använder oftast en mer top-down-orienterad metod där man letar efter en del av bilden som har ett mönster som matchar mallen tills man lyckas. Den metoden är ganska datorkrävande, men de används ändå i stor utsträckning eftersom bottom-up-metoden helt enkelt är för dålig för närvarande. Figuren till höger visar en modell för hur rekognitionsprocessen kan gå till när det gäller en robot som ska känna igen 3D-objekt (Andreasson & Duckett, sida 2). Inte exakt samma uppgift som vår agent, men principen för arbetsgången är liknande.



Hur går själva jämförelsen och matchningen med mallen till då? När det gäller färger är det ganska lätt att representera en bilds färginnehåll med hjälp av histogram, som lätt kan jämföras. Men när det gäller att urskilja former i en bild är det lite svårare. Särskilt om man tänker på att agenten ska kunna känna igen ett objekt ur olika vinklar. Det finns olika tekniker, men de kan delas in i ljusstyrkebaserad (brightness-based) respektive formbaserad (feature-based) igenkänning.

Ljusstyrkebaserad igenkänning går ut på att jämföra rena pixelvärden - med avseende på lusstyrkan i pixlarna - i en viss del av bilden där man har kunnat urskilja ett objekt. Man lagrar värden för kända objekt (en mall) som man sedan använder för att jämföra det men testar med. Implementationer av denna teknik finns t.ex. i form av neurala nätverk, beslutsträd och Bayesian modeller. Ett problem med den här tekniken är att mängden data kan göra processen långsam. För att lösa det kan man reducera datat eftersom närliggande pixlar ofta är väldigt lika, så man tar inte med precis all data. Igenkänning av handskriven text är ett exempel där ljusstyrkebaserad igenkänning fungerar utmärkt.

Formbaserad igenkänning använder sig av spatiala element i bilden, istället för att arbeta med rena pixlar. Typiskt använder man kanter (konturer), ytor och enskilda punkter som formelement. Det är alltså själva platsen för dessa element i bilden som är informationen. Även här lagrar man information om kända objekt för att kunna jämföra med. Jämförelsen sker genom att beräkna skillnaden mellan positionen för respektive element för bilden vi undersöker och mallen.

När vi har hittat konturerna för ett objekt kan vi sampla dessa till ett antal punkter. Formen för objektet kommer givetvis fortfarande att synas. Om vi plockar ut en av punkterna, punkt *p*, ur mängden kan vi utifrån den skapa en så kallad *shape context*. Den erhålls genom att skapa vektorer mellan *p* och samtliga övriga punkter. Vektorerna kan sedan representeras i histogram. Om man gör så för varje samplad punkt fås en bra beskrivning av objektet. Nu kan man jämföra två liknande men inte identiska objekt genom att ta motsvarande punkter i de båda och undersöka deras shape context. Om de är lika bör deras shape context överensstämma bra. Man kan beräkna skillnaden mellan varje par av punkter (motsvarande punkter på respektive bild) med hjälp av χ^2. Ju mindre den totala skillnaden är, desto mer lika är bilderna.

Kort om CCH

Färghistogram kan även användas för igenkänning av former och framför allt ”poser”, d.v.s. vilken vinkel ett objekt har i förhållande till betraktaren. Då använder man så kallade Color Cooccurrence Histograms, CCH (Ekvall, Hoffmann, Kragic). De kan bevara viss information om former eftersom de representerar antal förekomster av pixelpar med vissa färger. På så sätt kan man lokalisera geometriska former i en bild. Att bestämma vilken pose eller ställning, i vilken vinkel i förhållande till observatören, ett objekt har kan vara viktigt. När en maskin ska greppa ett föremål till exempel måste den först veta i vilken ställning föremålets stor.

Diskussion och slutsats

För en människa är objektigenkänning till synes en väldigt enkel sak. Perceptionen sker per automatik. För en dator eller maskin är det däremot en mycket komplex process. När det gäller att få en dator att känna igen objekt och reagera på omgivningen överhuvudtaget har vi kommit ganska långt, men det är ändå en ofantlig bit kvar till att få dem ”mänskliga” eller verkligen medvetna om sin omgivning. Den typen av medvetande som en dator har, kan det verkligen kallas medvetande? Det kan snarare liknas vid instinkt: stimuli och respons.

De tekniker som beskrivits i denna avhandling fungerar mycket bra för att känna igen handskrift, enkla objekt, ansikten etc. Vi anser att formbaserad objektigenkänning är mer intelligent och borde vara bättre, dels eftersom den bygger på en mer avancerad teknik. Men även för att den verkar ha bättre utvecklingspotential, det borde vara enklare att skapa lärande agenter med denna teknik. Eftersom den bygger på former och drag känns det som att den ska kunna kombinera kunskaperna bättre än vad en ljusstyrkebaserad igenkännare kan.
Det som kan vara svårt är att känna igen aktiviteter och saker som inte går att definiera som ”objekt”. Att få en dator att känna igen alla möjliga objekt och händelser och kunna kombinera dem fritt – som en människa kan göra – tror vi är orealistiskt inom en överskådlig framtid.

Objektigenkänning och perception är oerhört viktigt för artificiell intelligens. Utan att kunna uppfatta och reagera på omgivningen, hur ska man då kunna interagera med den? Att få en dator att uppfatta saker, att ge den sinnen, måste vara första steget mot något som kan liknas vid medvetande. Hur kan man vara medveten utan sinnen?

Referenser

Russell, Stuart. Norvig, Peter — Artificial Intelligence: A Modern Approach
Pearson Education Inc. 2003

Andreasson, Henrik. Duckett, Tom — Object Recognition by a Mobile Robot using Omni-directional Vision
ftp://aass.oru.se/pub/tdt/scai03.pdf

Ekvall, Staffan. Hoffmann, Frank. Kragic, Danica — Object Recognition and Pose Estimation for Robotic Manipulation using Color Cooccurrence Histograms
http://www.nada.kth.se/~ekvall/iros2003.pdf

Skrivet af: