

Bullfighting Sumo: Kan een minidrone stiervechten?

Dico van Leeuwen
Universiteit van Amsterdam

Jan Erik van Woerden
Universiteit van Amsterdam

Ozzy Ulger
Universiteit van Amsterdam

Tim Smit
Universiteit van Amsterdam

Bullfighting Sumo is een minidrone van Parrot van het type Parrot Jumping Sumo. Bullfighting Sumo kan stiervechten spelen met een persoon door middel van beeldherkenning.

Inleiding

In de vierde week van het vak Zoeken, Sturen en Bewegen is het team de opdracht gegeven een toepassing voor een drone of robot te ontwikkelen. Dit was een nieuwe uitdaging ten opzichte van de voorafgaande drie weken, waarin een boter-kaas-en-eieren-spelend programma en een schakende robotarm werd ontwikkeld. Hierbij is recent opgedane kennis over programmeertalen zoals PROLOG en Java toegepast en veel ervaring opgedaan over het gebruik van zoekalgoritmes voor praktische zoekproblemen.

Voorwaarde voor de te ontwikkelen toepassing was "go, where no one has gone before", oftewel iets wat niemand ooit eerder heeft gemaakt. Alvorens echter ideeën uit te wisselen, was het verstandig om eerst een type drone uit te kiezen. De keuze viel al snel op de Parrot Jumping Sumo minidrone. Dit type drone was niet zo populair als de vliegende drone, maar had wel evenveel potentie voor creatieve toepassingen. Bovendien vormde dit type drone een extra grote uitdaging, omdat er minder documentatie over te vinden was.

Dankzij dhr. Arnoud Visser werd inspiratie toegereikt voor een autonoom rijdende Jumping Sumo uit een document met reglementen voor de Autonome Driving challenge tijdens de Portugese Robotics Open. Vooral de onderdelen "3.2 - Obstacles", "3.5 - Signaling Panels" en "3.6 - Vertical Traffic Signs" hebben het team geïnspireerd een autonoom stiervechtende Parrot Jumping Sumo te ontwikkelen. De vraag die hierbij centraal staat is: Kan een minidrone stiervechten?. Om deze vraag te kunnen beantwoorden is per ontwikkelingsdeel een deelvraag gesteld. Is "Bullfighting Sumo" in staat om i. een rood object in het gezichtsveld te volgen? ii. een bepaalde afstand te bewaren van een rood object wanneer deze van zich af beweegt? iii. te springen richting een rood object wanneer deze naar zich toe beweegt? iv. een rood object te zoeken door te draaien totdat deze wel gedetecteerd is, wanneer deze niet in het gezichtsveld te detecteren was en v. middels een schrikreactie een zwarte doek, dat is geplaatst over het gezichtsveld, weg te werken?

De kernhypothese luidde dat het mogelijk is om een autonoom stiervechtende Parrot Jumping Sumo te ontwikkelen,

dankzij de ingebouwde camera van de Parrot Jumping Sumo. Deze hypothese werd opgesteld dankzij de deelhypotheses: de drone is in staat om i. een rood object in het gezichtsveld volgen; ii. een bepaalde afstand bewaren van een rood object wanneer deze van zich af beweegt; iii. springen richting een rood object wanneer deze naar zich toe beweegt; iv. een rood object zoeken door te draaien totdat deze wel gedetecteerd is, wanneer deze niet in het gezichtsveld te detecteren was en v. middels een schrikreactie een zwarte doek, dat is geplaatst over het gezichtsveld, weg te werken.

Benodigde software om dit de stiervechtende Parrot Jumping sumo te kunnen ontwikkelen waren Open Source Computer Vision (kort: OpenCV), NodeJS en een editor.

Alhoewel het project vooral een leuk eindresultaat opgeleverd heeft bieden de bevindingen en ontwikkelingen mogelijkheden om serieuzere toepassingen te bedenken. Om maar iets op te noemen, een militaire toepassing waarbij autonoom gezocht moet worden naar zeer specifieke objecten is maar één van de talloze mogelijke opties. Wat tijdens dit project is ontwikkeld in een toy wereld kan getoets worden aan de echte wereld om het ons leven makkelijker te maken.

Voortgangsrapportage

Maandag

Op de eerste dag van een veelbelovende week werd de Parrot Jumping Sumo opgehaald. Om eerst een beeld te krijgen van de mogelijkheden met de Parrot Jumping Sumo, werd al door Parrot ontwikkelde iOS-app FreeFlight 3 op een iPhone 5 gedownload. Dankzij deze app werd in de "Free Ride" modus kennis gemaakt met alles waartoe de Parrot Jumping Sumo in staat was qua bewegingen.

In deze eerste ronde van kennismaking vergaarde het team al snel veel inspiratie en begon zij al snel met brainstormen over de mogelijkheden om de bewegingen te implementeren. Een aantal van de ideeën die voortkwamen waren onder andere om i. de drone autonoom een rood object te benaderen en ii. een bepaalde afstand te bewaren van een rood rechthoekig object wanneer deze van zich af beweegt. Om deze eigenschappen te kunnen implementeren zou gebruik

Maandag	
Tijd	Taken
9:00 - 11:00	Parrot J.S. opgehaald. Kennis gemaakt met Parrot J.S. Parrot J.S. mogelijkheden bekeken.
11:00 - 13:00	Gebrainstorm over onderwerpen. Indeling gemaakt met taken per dag.
13:00 - 15:00	Documentatie opgezocht en bekeken. OpenCV gedownload
15:00 - 17:00	Git opgezet Met NodeJS geëxperimenteerd

Tabel 1

Voorbeeld van een dagindeling.

gemaakt moeten worden van beeldverwerking, gekoppeld aan de motoren van de drone.

Alvorens te starten met documentatie op te zoeken en te beginnen met programmeren, was een goede planning van essentieel belang. Daarom is een dagindeling per twee uur gemaakt over wat er moet komen. Op deze manier kon in het belang van het project vlothedigheid en efficiëntie gestimuleerd worden en eventueel gereflecteerd worden waarom een bepaalde taak niet lukte binnen de aangegeven tijd.

Een deel van het team startte een zoektocht naar bruikbare informatie en documentatie, terwijl een ander deel ging onderzoeken welke tools er nodig zijn om de beeldverwerking en het contact met de drone te bewerkstelligen. Tijdens deze zoektocht werd een package gevonden die het mogelijk maakte om door middel van enkele simpele commands de drone te besturen. Dankzij deze vondst werd het mogelijk om met succes een eerste test uit te voeren met de drone via commands vanaf de laptop. In de voorbeelden bij de package werd ook een aparte package gevonden die het mogelijk en makkelijk maakte om te communiceren met OpenCV, het programma voor de beeldverwerking. Automatisch vielen onze keuzes voor de tools op OpenCV voor de beeldverwerking en NodeJS voor de interactie met de drone. Hieruit volgde ook meteen de keuze voor de programmeertaal waarin ontwikkeld werd, namelijk JavaScript, omdat NodeJS JavaScript gebruikt. Helaas slaagde niet ieder lid er in om zowel NodeJS als OpenCV werkend te hebben. Dit grote probleem werd zeer serieus genomen, omdat het team streefde naar optimale betrokkenheid van ieder individueel lid bij de ontwikkelingen. Hierop werd het begin van de dagindeling van dinsdag aangepast, met extra aandacht voor het werkend krijgen van NodeJS en OpenCV op de laptops van alle teamleden.

Dinsdag

De problemen van enkele teamleden met het werkend krijgen van NodeJS en OpenCV werden vroeg op de dag opge-

lost door alle reeds geïnstalleerde en gedownloade bestanden verwant aan het onderzoek te verwijderen en het hele installatieproces opnieuw te belopen.

De ideeën die eerder voortkwamen uit het uittesten van de drone met de FreeFlight 3-app van Parrot werden nu vertaald naar concrete stukken theorie. De precisie waarmee de drone via de FreeFlight 3-app reageerde op 90 en 180 graden-bochten, was waarschijnlijk te danken aan een implementatie van rotatie op basis van graden. In de gevonden NodeJS package werd echter gebruik gemaakt van bochten die roteerden op basis van tijd in milliseconden. Alhoewel dit eerst een struikelblok leek, bleek dit geen belemmering te vormen binnen het onderdeel van autonoom rijden van dit project. Dankzij het gebruik van milliseconden was het juist gemakkelijk een rustige, vloeiende en gecontroleerde rotatie te implementeren, terwijl de bochten die gemaakt werden via de FreeFlight 3-app ondanks de precisie overkwamen als chaotisch en niet vloeiend.

Door middel van beeldverwerking met OpenCV werd het mogelijk beeld binnen te krijgen waar informatie uit gehaald kon worden. Om de dummy-vlag, een rode broodtrommel, te kunnen detecteren, was eerst meer informatie over de eigenschappen nodig. Met behulp van Photoshop CS6 en sites zoals Colorizer.org werd een analyse gemaakt van de eigenschappen.



Afbeelding 1: Informatie over de gebruikte kleur rood.

Door het binnenkomende beeld te filteren op de kleur rood werd het herkennen van objecten een stuk beter. Dit wordt gedaan door een range van kleuren aan te geven. De kleurenrange is ingesteld op (160, 110, 50) tot (180, 255, 255). Deze kleurenrange is rood van lichtrood tot donkerrood. Hierdoor werd het heel eenvoudig om contour detectie uit te voeren om objecten te herkennen. Door de filtering van kleuren werden er alleen de contouren genomen van locaties waar de pixels overgingen van een kleur die binnen de kleurenrange viel en de pixels die daarbuiten vielen. Hierdoor detecteert de drone in het beeld rode objecten. De drone pakt het grootste rode object en corrigeert zijn houding daarop in een rechte lijn tot het object en benadert deze totdat de kleur rood vijf procent van het scherm inneemt.

```

per interval of 50 ms
make a copy of the current frame
filter everything out that isn't red
find all the contours of objects in the copy
for every object do
  | check if the size is big enough to be useful
  | check if the size is bigger than the biggest object so
  | far
  | calculate the coordinates of the center of the object
end
draw the biggest object in the original image
if an object is found then
  | check how far the object is from the center of the
  | image
  | if the object is close to the center of the image then
  |   | if the object is big then
  |   | | stand still
  |   | else
  |   | | move forward
  |   | end
  | else
  | | rotate towards the object
  | end
end

```

Woensdag

Een volgend onderdeel van het onderzoek was het springelement, waarbij de drone een äfschrikreactie"vertoont tegenover een groter wordend rood rechthoekig object (oftewel "de stierenflag"). Hiermee wordt de woede van een stier gesimuleerd wanneer de stierenvechter de stier uitdaagt. Hierbij was het kwestie om een minimale percentage aan inbetslagname van het beeld door de kleur rood te nemen, alvorens via een if-statement opdracht te krijgen om te springen. Om de simulatie van een echte stier te stimuleren werden extra elementen toegevoegd, zoals het "tappen" met de staart op de grond wanneer het rode object op een bepaalde afstand staat. Hiermee werd de intimidatie door het vegen van de stier gesimuleerd.

```

per interval of 50 ms
make a copy of the current frame
filter everything out that isn't red
find all the contours of objects in the copy
for every object do
  | check if the size is big enough to be useful
  | check if the size is bigger than the biggest object so
  | far
  | calculate the coordinates of the center of the object
end
draw the biggest object in the original image
if an object is found then
  | check how far the object is from the center of the
  | image
  | if the object is close to the center of the image then
  |   | if the object is big then
  |   | | stand still
  |   | | if this is the first interval where the object is
  |   | | | big then
  |   | | | | do the tap-animation
  |   | | | end
  |   | | | if the object is even bigger then
  |   | | | | jump and take a short brake
  |   | | | end
  |   | | else
  |   | | | move forward
  |   | | end
  |   | else
  |   | | rotate towards the object
  |   | end
  | else
  | | start rotating
  | end

```

Om een verdere uitbreiding van het autonoom rijden te implementeren, werd besloten de drone zelf te laten zoeken naar een rood object in het gezichtsveld, in plaats van deze door een persoon te laten plaatsen. Hiervoor roteert de drone om haar eigen as en besluit autonoom wanneer te stoppen met roteren en het rode object te benaderen. Voor de al bestaande code had dit weinig gevolgen, want deze werd als het ware "verpakt in een if-else-statement. Als het rode object niet in het gezichtsveld was, moest de drone hiernaar opzoek. Dit betekende dat in het else-deel van de if-else-statement een nieuwe beweging werd geïmplementeerd, namelijk een rotatie met een snelheid van 180 graden per seconde. Dankzij deze toevoeging kon de drone eindeloos lang, totdat de batterij leeg was, autonoom interacteren met rode objecten.

Een bug die naar voren kwam is het te vaak laten springen van de drone als hij te dichtbij een object is. Tijdens het uitvoeren van de sprong herkende hij weer het object en kreeg hij door dat hij nog een keer de animatie moest uitvoeren terwijl hij nog in de lucht was. Om dit te voorkomen hebben we een stop ingesteld. Tijdens een animatie zoals een sprong analyseert hij geen beelden en zorgt hij ervoor dat hij dus niet

onbedoelt animaties meerdere malen uitvoert.

```

per interval of 50 ms
make a copy of the current frame
filter everything out that isn't red
find all the contours of objects in the copy
for every object do
  check if the size is big enough to be useful
  check if the size is bigger than the biggest object so far
  calculate the coordinates of the center of the object
end
draw the biggest object in the original image
if an object is found then
  check how far the object is from the center of the image
  if the object is close to the center of the image then
    if the object is big then
      stand still
      if this is the first interval where the object is big then
        | do the tab-animation
      end
      if the object is even bigger then
        | jump and take a short brake
      end
    else
      | move forward
    end
  end
  rotate towards the object
else
  | start rotating
end

```

Naast software is ook aandacht besteed aan de opmaak van de hardware. Om de Parrot Jumping Sumo om te toveren naar Bullfighting Sumo is hulp ingeschakeld van creatieve experts. De belangrijke taak van opmaak is daarom uitbesteed aan het zusje en nichtje van één van de teamleden.



Afbeelding 2: Creatieve experts aan het werk.



Afbeelding 3: Bullfighter Sumo.

Donderdag

Een laatste implementatie die werd bewerkstelligd was een afschrikreactie van de drone wanneer een zwart doek op de drone werd geplaatst, waardoor deze slechts een zwart beeld binnenkreeg. Hiermee werd het gedrag van een stier gesimuleerd wanneer deze wordt geblinddoekt. Om deze beweging te implementeren werd gebruik gemaakt van de gemiddelde kleur in het beeld. Van de gemiddelde kleur werden de RGB waarden genomen en deze bij elkaar opgeteld. Hoe lager de totale RGB waarde is, hoe donkerder de kleur. Als de totale RGB waarde laag genoeg is, dan wordt er een spin jump ingezet.

Helaas ervaarde het programma een enorme lag tijdens het testen hiervan. Om te bekijken of dit te maken had met de zojuist ontwikkelde afschrikreactie voor zwart beeld, werd het deel uitgecomment. Echter, het probleem deed zich voort. Door de lag kwam het beeld tijdens de rotatie van de drone, wanneer deze op zoek is naar een rood object, vaak te laat binnen waardoor twee of drie 360 graden rotaties gedaan moesten worden alvorens het rode object te detecteren. Vanwege deze reden is de gehele code omvat in een if-statement, zodat acties pas worden verricht als er geen sprake is van lag. Ondanks veel moeite en aanpassingen kreeg de drone de oude, goede en vloeiende werking niet meer helemaal terug. Het team besloot hulp in te schakelen van Kyriacos Shiarlis, die concludeerde dat het te maken had met een overbelast netwerk vanwege het intensieve testen door alle teams. Ondanks deze teleurstellende oorzaak had het team er vertrouwen in dat de drone goed zou presteren tijdens de presentatie, wanneer het netwerk waarschijnlijk wat minder belast zou zijn.

```

per interval of 50 ms
if there is no lag then
    check the average color of all the pixels in the
    current frame
    if the average color is dark then
        | turn and jump
    end
    make a copy of the current frame
    filter everything out that isn't red
    find all the contours of objects in the copy
    for every object do
        check if the size is big enough to be useful
        check if the size is bigger than the biggest
        object so far
        calculate the coordinates of the center of the
        object
    end
    draw the biggest object in the original image
    if an object is found then
        check how far the object is from the center of
        the image
        if the object is close to the center of the image
        then
            if the object is big then
                stand still
                if this is the first interval where the
                object is big then
                    | do the tab-animation
                end
                if the object is even bigger then
                    | jump and take a short brake
                end
            else
                | move forward
            end
        end
        turn towards the object
    else
        | start rotating
    end
end

```

Vrijdag

Dankzij de eerdere vier dagen een strakke planning te hebben gevolgd, ontstond de vrijheid om te experimenteren met een efficiëntere code. Hierbij werd vooral gefocust op het reduceren van de lag, die eerder voor grote problemen zorgde. Daar waar het eerder niet wilde lukken het probleem op te lossen, werden nu zeer efficiënte oplossingen ontwikkeld. De lag werd met een enorme hoeveelheid verminderd door het aantal opgenomen pixels tweemaal door vier te delen. Hierdoor werd niet iedere individuele pixel beschouwd, maar groepen van pixels. Dankzij deze aanpassing werkte het geheel aanzienlijk beter en efficiënter.

Ook werd duidelijk dat het beeld vooral lag vertoonde als de drone zelf veel bewegingen maakte. Door de drone te laten stoppen als hij een lag heeft van meer dan tweehonderd milliseconden, heeft de drone veel minder lang lag en kan hij veel sneller bij het object komen.

De rest van de laatste dag stond in het thema van de presentatie. Zo is er nagedacht over de opbouw van de presentatie. Er is hier gekozen voor structuur die dicht bij het gedachte proces deze week stond. Ook is de Bullfightig Sumo uitgebreid getest. Hier kan je denken aan verschillende ondergronden en in een ruimte met verschillende kleuren. Er is hierbij ook gezocht naar een eenvoudigere manier om de broodtrommel te verschuiven en een alternatief te vinden voor het te zware t-shirt. Wat een nog steeds te zware hoed is geworden, maar het ziet er wel heel leuk uit.

Het grootste gedeelte van de tijd op vrijdag stond in het teken van het labboek netjes in orde krijgen. Zo zijn er plaatjes in gezet en staat alles netjes op de juiste plek. Ook is er nog vanalles uit geschreven en bijgevoegt.

Discussie

Het onderzoek naar de vraag of het mogelijk is een autonoom stiervechtende Parrot Jumping Sumo te ontwikkelen gaf als resultaat een drone die een rood voorwerp kan zoeken en er op af kan gaan, op een bepaalde afstand stil kan gaan staan en over dit object heen te springen zodra deze te dichtbij is. Daarnaast zal de drone proberen dingen van zich af te werpen als hij bedekt is. Dit resultaat is in lijn met de voor de start van het onderzoek verwachte resultaat.

Ook al is een goed resultaat bereikt in dit onderzoek, zijn er wel problemen geweest die voor oponthoud zorgden. Het eerste probleem waar bij dit onderzoek tegenaan werd gelopen was het probleem dat de packages zeer lastig te installeren waren en zeer lastig aan de draad te krijgen waren. Om dit probleem op te lossen moest er hulp komen van mensen die dit probleem al eerder gezien hadden. Het tweede probleem waar tegenaan gelopen werd was het laten reageren van de robot op een doekje dat over hem heen werd gegooid, om dit te bereiken moest berekend worden wat de gemiddelde kleur was van het beeldscherm, alleen dit zorgde voor veel lag. Nadat deze code iets minder precies gemaakt werd en er dus iets minder computatie kracht nodig was voor het berekenen van de gemiddelde kleur, werd de lag wel iets minder. De lag die de connectie naar de drone ondervond werd wel steeds erger tijdens dit project. Waar we de eerste dagen niet heel veel last hadden van de lag, werkte de drone niet meer gewenst op donderdag door te veel lag. Deze lag bleek niet te liggen aan het berekenen van de gemiddelde kleur en dus kon hier niet heel veel aan gedaan worden.

Deze resultaten zijn niet heel baanbrekend op gebied van nieuwe technologie, maar zijn meer een voorbeeld om te laten zien hoe makkelijk het is om een simpele taak autonoom uit te laten voeren door een drone. Het resultaat kan als ar-

gument dienen voor mensen die autonome drones bijvoorbeeld als militaire toepassing willen gebruiken, maar natuurlijk ook talloze andere toepassingen in het echte leven.

Voor de conclusie van dit onderzoek moet eerst gekeken worden hoe de deelvragen beantwoord zijn. Op de eerste plaats hebben we aan kunnen tonen dat de drone in staat is een rood object in het gezichtsveld te volgen. Daarnaast kan de drone ook een bepaald afstand bewaren van een rood ob-

ject wat zich van de drone af beweegt. Ook kan de drone springen als het rode object zich verder richting de drone beweegt. Verder zal de drone rond draaien op zoek naar het object als hij deze niet kan vinden. Tot slot zal de drone proberen dingen van zich af te gooien zodra de drone bedekt is met iets donkers. Hieruit volgt dat het antwoord op de hoofdvraag "Kan een minidrone stiervechten?" is dat een minidrone inderdaad tot staat is om te stiervechten.