

OBTINEREA PERCEPȚILOR

Percepțiile sunt colectate în mai multe moduri folosind diferiți senzori (vizuali, sonori, distanță) sau prin software-uri de procesare de imagini.

Masina autonoma este dotată cu camere de vedere montate în direcția parbrizului și deasupra mașini astfel tot mediul înconjurător este observat. Cu ajutorul camerelor 3D imaginile detectate sunt foarte realiste și detaliate. Aceste camere conțin senzori care detectează automat obiectele din jur (alte mașini, pietoni, semne de circulație), le clasifică în funcție de categorii și determină distanța dintre ele și autovehicul. Totuși există unele slăbiciuni, cum ar fi în cazul unor condiții meteo nefavorabile (zapadă, ploaie, ceață), camerele nu pot detecta clar obstacolele a căror culori sunt asemănătoare. Un conducător auto prin intermediul oglinzilor are posibilitatea doar de a vedea dacă există un vehicul auto în spatele lui, prin aceste camere se poate observa o distanță mai mare în spatele lui.

Cu ajutorul acestor senzori pe board-ul de navigație vor apărea semnele de circulație în timp real, ba chiar preventiv pentru a putea lua decizii optime în prealabil.

Acuratețea GPS-ului este foarte importantă pentru o percepție corectă, deoarece este nevoie de informații exacte, fără greșeli. GPS-ul este preîncărcat cu hărți și semnele de circulație ale acestora, deoarece în cazul în care o detecție a camerei dă un răspuns fals pozitiv (după cum am menționat mai sus), agentul să se bazeze pe date predefinite.

Senzorii radar au de asemenea o contribuție crucială la funcția de conducere autonomă. Ei trimit unde radio care detectează obiecte și le măsoară distanța și viteza în raport cu mașina autonomă, în timp real. Există 2 tipuri de senzori: cu rază scurtă care monitorizează punctul mort, menținerea benzii și oferă ajutor la parcare; cu rază lungă asigură distanța dintre diferite obiecte și oferă asistență la frânare. Spre deosebire de senzori camerei, sistemele radar nu au probleme atunci când identifică obiecte în timpul ceații sau ploii. Senzori radar scanează doar pe orizontală, ceea ce poate cauza o varietate de probleme atunci când urmează să conduci sub un pod.

O mașină autonomă prezintă și senzori Lidar, care funcționează asemănători cu cei radar, singura diferență fiind că folosesc lasere în loc de unde radio. Acești senzori creează o hartă de 360 grade în jurul autovehiculului.

Deep Learning-ul este baza mașinilor autonome. Pentru clasificarea tipurilor obiectelor detectate se va folosi un algoritm de procesare a imaginilor (object detection). Algoritmul de recunoaștere a pietonilor necesită îmbunătățiri deoarece vehiculul auto identifică corect numai 90%-95% din pietoni.

1. (ag_percept (percept_probj crossing1) (percept_pname isa) (percept_pval ped_crossing))
(ag_percept (percept_probj crossing1) (percept_pname state) (percept_pval free/busy))

Detecția treceri de pietoni se face prin intermediul GPS-ului care după cum am specificat mai sus conține harta drumului cu toate semnele de circulație. Aceasta detecție este confirmată și de senzorii camerei montate pe masină. Camera folosește algoritmi de procesare a imaginilor (object detection/ body detection) cu ajutorul cărora detectează marcajele albe de pe drum și mișcarea oamenilor deasupra acestora. În cazul în care nu apar persoane în cadrul marcajului aceasta este free altfel este busy. Algoritmi sunt implementați prin openCV și python.

- <https://data-flair.training/blogs/pedestrian-detection-python-opencv/>
- <https://answers.opencv.org/question/226261/way-to-find-zebra-crossing-lines/>

2. (ag_percept (percept_probj sign1) (percept_pname isa) (percept_pval road_sign))
(ag_percept (percept_probj sign1) (percept_pname type) (percept_pval priority))

GPS-ul este actualizat cu toate semnele de circulație, astfel în momentul în care camera detectează un nou semn de circulație acesta este confirmat și de prezența lui pe GPS-ului. Se folosesc tot algoritmi de detecție a obiectelor. Se detectează semnele de circulație în formă de romb după care se verifică dacă culoarea din interior este galbenă.

- <https://github.com/harsh-99/Traffic-sign-detection>

3. (ag_percept (percept_probj road1) (percept_pname isa) (percept_pval road))
(ag_percept (percept_probj lane1) (percept_pname isa) (percept_pval line))
(ag_percept (percept_probj line1) (percept_pname status) (percept_pval on/busy/free))

Detecția unui drum/benzi se face cu ajutorul GPS-ului montat pe masină. Aici este un top a celor mai bune gps-uri de pe piață la momentul actual. Cu ajutorul camerei de luat vederi și a unei procesări de imagini auxiliare se poate identifica dacă există o mașină pe o anumită bandă/drum. Iar faptul că mașina mea se afla pe o anumită bandă/drum este dat de localizarea de pe GPS.

- <https://www.toptenreviews.com/best-car-navigation-system>
- <https://medium.com/mlearning-ai/road-lane-detection-using-opencv-hough-lines-transform-explained-a6c8cfc03f68>

4. (ag_percept (percept_probj car1) (percept_pname isa) (percept_pval car))
(ag_percept (percept_probj car1) (percept_pname on) (percept_pval road1/parking1))
(ag_percept (percept_probj car1) (percept_pname intention) (percept_pval right/ahead/left/none/back))

Această percepție se poate obține cu ajutorul unei camere montate pe mașină care procesează imaginile în timp real și determină forma acestora ca fiind o altă mașină, totodată îi determină poziția. Putem observa o mașină că semnalizează tot cu ajutorul unei camere care preia imagini în timp real, la un interval suficient de scurt de timp, astfel încât să își dea seama că culoarea becurilor de semnalizare se schimbă intermitent și care dintre cele 2 semnalizări se schimbă.

- <https://kalebujordan.dev/real-time-vehicle-detection-using-python/>

5. (ag_percept (percept_probj parking1) (percept_pname isa) (percept_pval parking))

Locul de parcare este detectat prin intermediul camerei și a GPS-ului, dar și prin intermediul senzorilor Lidar.

- <https://link.springer.com/article/10.1007/s13177-022-00300-w>

6. (ag_percept (percept_probj lights1) (percept_pname isa) (percept_pval traffic_lights))
(ag_percept (percept_probj lights1) (percept_pname color) (percept_pval green/red))

Detectarea semnelor de circulatie si a semafoartelor se folosesc librarii opencv si clasificatoare in cadrul camerelor de din fata masinii. Vehiculul este de asemenea echipat cu GPS cu harti pre-descarcate, asadar în situații de vizibilitate minima agentul este atenționat asupra semnelor. Sistemul de navigatie poate fi conectat la internet, asadar raimprospatarea harților este posibila. Pentru a detecta ce culoare are semaforul la momentul curent se folosesc algoritmi implementați în openCV care utilizează histograma pentru a determina culoarea:

- <https://stackoverflow.com/questions/64723166/how-to-detect-traffic-light-color-with-histogram-info-using-opencv>

7. (ag_percept (percept_probj intersection1) (percept_pname isa) (percept_pval intersection))

Camera montată în fața masini detectează prin algoritmi de procesare a imaginilor faptul ca urmează o intersecție, stiind ca marcajul de final de strada este alb, detectând linia alba.

- <https://medium.com/mlearning-ai/road-lane-detection-using-opencv-hough-lines-transform-explained-a6c8cfc03f68>

8. (ag_percept (percept_probj mark1) (percept_pname isa) (percept_pval mark))
(ag_percept (percept_probj mark1) (percept_pname type) (percept_pval discontinuous/continous))

Deteția unui marcaj de pe drum se face cu ajutorul camerei montate deasupra masini și in față masini. Iar dupa ce imaginile au fost preluate se aplică algoritmi de detecție a tipului de linie pentru a afla daca este continua sau discontinua.

- <https://ori.ox.ac.uk/media/5701/road-marking-classification-paul-ingmar.pdf>

9. (ag_percept (percept_probj _____) (percept_pname proximity) (percept_pval 0/1))

Pentru a afla care obiect se afla mai aproape de autohevicul se va folosi radar-ul masini care calculeaza distanța de la un obiect la masină. După ce toate distanțele sunt calculate se pun într-un sir în funcție de distanța cea mai mică.

- <https://pyimagesearch.com/2016/04/04/measuring-distance-between-objects-in-an-image-with-opencv/>

10. (ag_percept (percept_probj _) (percept_pname width) (percept_pval _))
(ag_percept (percept_probj car2) (percept_pname heigth) (percept_pval 3))

Imagini cu obiectele sunt preluate prin intermediul camerelor montate pe masină. Pe urmă, aceste imagini sunt procesate după cum se poate observa în link-ul de mai jos:

- <https://pyimagesearch.com/2016/03/28/measuring-size-of-objects-in-an-image-with-opencv/>

11. (ag_percept (percept_probj _) (percept_pname direction) (percept_pval left/right/ahead))

Direcția unui obiect față de mașină se dectează prin senzori camerei. Se preia imaginea de pe camera din față după care se imparte în două jumatați egale iar apoi în funcție de origine se dectează de ce parte este (stânga sau dreaptă). Un exemplu de decupare după origine și aflarea de ce partea a origine se afla:

- <https://stackoverflow.com/questions/59525065/detecting-object-location-in-image-with-python-opencv>