POP RUXANDRA MARIA 30643

OBȚINEREA PERCEPȚILOR

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

Masina autonoma este dotată cu camere de vedere montate în direcția parbrizului și deasupra masini astfel tot mediul înconjurator este observat. Cu ajutorul camerelor 3D imaginile detectate sunt foarte realistice și detaliate. Aceste camere conțin senzori care detectează automat obiectele din jur (alte masini, pietoni, semne de circulație) , le clasifică în funcție de categori și determină distanța dintre ele și autovehicul. Totuși există unele slabiciuni, cum ar fi în cazul unor conditii meteo nefavorabile (zapadă, ploaie, ceață), camerele nu pot detecta clar obstacolele a căror culori sunt asemânătoare.

Un conducător auto prin intermediul oglinzilor are posibilitatea doar de a vedea dacă exisra 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 decizi optime în prealabil.

Acuratețea GPS-ului este foarte importantă pentru o percepție corecta, deoarece este nevoie de informații exacte, fără greseli. GPS-ul este preîncărcat cu hârți și semnele de circulație ale acestora, deorece 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 masina 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 ceții 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 masină autonoma prezintă și senzori Lidar, care funcționează asemânatori cu cei radar, singura diferență fiind că folosesc lasere în loc de unde radio. Acesti senzori crează o hartă de 360 grade în jurul autovehicului.

Deep Learning-ul este baza masinilor autonome. Pentru clasificarea tipurilor obiectelor detectate se va folosi un algoritm de procesare a imaginilor (object detection) . Algoritmul de recunoastere a pietonilor necesită îmbunătățirii 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-ul 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 foloste algoritmi de procesare a imaginilor (object detention/ body detection) cu ajutorul cărora detectează marcajele albe de pe drum și miscarea oamenilor deasupra acesteia. 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 dectează 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 momentat 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 procesari de imagini auxiliare se poate indentifica dacă există o masina pe o anumita banda/drum. Iar faptul că masina 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-

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))

Aceasta percepție se poate obține cu ajutorul unei camere montate pe mașină care procesează imaginile în timp real și determina forma acesteia ca fiind o altă masină, totodată îi determina 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 încat să îsi dea seama că culoarea becurilor de semnalizare se schimbă întermitent și care dintre cele 2 semnalizari 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. Vehicolul 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 dectecta ce culoare are semaforul la momentul curent se folosesc algoritmi implemtați în openCV care utilizează histograma pentru a determina culoarea:

https://stackoverflow.com/questions/64723166/how-to-detect-trafic-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-

8. (ag\_percept (percept\_probj mark1) (percept\_pname isa) (percept\_pval mark))

(ag\_percept (percept\_probj mark1) (percept\_pname type) (percept\_pval discontinuous/continous))

Detecția 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 intr-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 preaia 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