



M Ű E G Y E T E M 1 7 8 2

Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Méréstechnika és Információs Rendszerek Tanszék

Pataki Milán

**FORGALMI MANŐVER
FELISMERÉS VALÓSZÍNŰSÉGI
MODELLEL**

KONZULENSEK

Krébesz Tamás István
Bodnár Attila

BUDAPEST, 2015



SZAKDOLGOZAT-FELADAT

Pataki Milán (HUFXED)

szigorló villamosmérnök hallgató részére

Forgalmi manőver felismerés valószínűségi modellel

A korszerű autók vezetést segítő rendszereinek szerves része a forgalmi manőverek felismerése. A szituáció-elemzés során felhasznált adatok bizonytalan ismereten alapulnak, ezért sok esetben a szituáció meghatározása során nem tehetünk biztos állításokat. Ilyenkor az egyes manővereket valószínűségi modell alapján kezeljük.

A hallgató feladata egy olyan valószínűségi modell felépítése, amely alkalmas a gépjármű kamera és radarrendszere által szolgáltatott adatok alapján a forgalomban résztvevő járművek különböző forgalmi manővereinek (elsősorban sávváltás) felismerésére, illetve adott esetben előrejelzésére.

A hallgató feladatának a következőkre kell kiterjednie:

- Mutassa be a forgalmi manőverek felismerésének témakörét és szerepét nagyobb rendszerekben illetve az esetlegesen felmerülő nehézségeket!
- Ismerje meg a különböző (grafikus) valószínűségi modelleket és válasszon egy alkalmas modellt a feladat megoldására, továbbá indokolja meg a választását!
- Készítsen egy alkalmazást a forgalmi manőverek felismerésére, mutassa be ennek használatát valós mérési szekvenciák segítségével!
- Mutassa be eredményeit, továbbá ismertesse a továbbfejlesztési lehetőségeket!

Tanszéki konzulens: Krébesz Tamás István

Külső konzulens: Bodnár Attila (Bosch Magyarország Kft.)

Budapest, 2015. március 12.

.....
Dr. Jobbágy Ákos
tanszékvezető

Tartalomjegyzék

Összefoglaló	5
Abstract	6
1 Bevezetés	7
2 Felhasznált autóiipari rendszerek.....	9
2.1 Szenzorok.....	9
2.1.1 Radar	10
2.1.2 Gépi látás.....	12
2.1.3 Szenzorok összefoglalása.....	14
2.2 A sávmodell.....	15
3 Valószínűségi modellek	17
3.1 Rejtett Markov modell	19
3.2 Bayes-hálók.....	23
4 A Sáv váltás	26
4.1 Sávok és járművek kapcsolata.....	26
4.2 A sáv váltás folyamata.....	30
4.3 A sáv váltás modellje.....	33
5 A LaneChange program.....	35
5.1 Függvények	36
5.2 Taszkok	40
5.2.1 Tanítás	40
5.2.2 Szimuláció	41
5.2.3 ROC	41
6 Eredmények.....	42
7 Továbbfejlesztési lehetőségek.....	45
Ábrajegyzék.....	48
Irodalomjegyzék.....	49
Függelék.....	50

HALLGATÓI NYILATKOZAT

Alulírott **Pataki Milán**, szigorló hallgató kijelentem, hogy ezt a szakdolgozatot meg nem engedett segítség nélkül, saját magam készítettem, csak a megadott forrásokat (szakirodalom, eszközök stb.) használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Hozzájárulok, hogy a jelen munkám alapadatait (szerző(k), cím, angol és magyar nyelvű tartalmi kivonat, készítés éve, konzulens(ek) neve) a BME VIK nyilvánosan hozzáférhető elektronikus formában, a munka teljes szövegét pedig az egyetem belső hálózatán keresztül (vagy hitelesített felhasználók számára) közzétegye. Kijelentem, hogy a benyújtott munka és annak elektronikus verziója megegyezik. Dékáni engedéllyel titkosított diplomatervek esetén a dolgozat szövege csak 3 év eltelte után válik hozzáférhetővé.

Kelt: Budapest, 2015. 05. 15.

.....
Pataki Milán

Összefoglaló

Napjainkban egyre nagyobb szerepet kapnak az autóiparban a különböző vezetést segítő technológiák, mind a kényelmi, mind a biztonsági funkciók tekintetében. Ezen rendszerek fontos feladata a forgalmi szituációk felismerése és elemzése. Ilyen feladat lehet annak meghatározása, hogy a forgalom résztvevői milyen forgalmi manővereket hajtanak végre pillanatnyilag, illetve mit szándékoznak a közeljövőben végrehajtani. Erre a szituáció-elemzés nyújthat segítséget.

A szituáció-elemzés során felhasznált mérési adatok bizonytalan ismereten alapulnak, ezért sok esetben a szituáció meghatározása során sem tehetünk biztos állításokat. Ilyenkor az egyes manővereket valószínűségi modell alapján kezeljük.

A szakdolgozatomban a szükséges elméleti tudás elsajátítása után kidolgoztam egy megoldást a forgalomban résztvevő járművek sávváltásának felismerésére, illetve ennek előrejelzését. A gépjárművek radar- és kamerarendszere által szolgáltatott adatokat grafikus valószínűségi modellbe illesztettem be. A program fejlesztését, és a kiértékeléshez szükséges tesztelést MATLAB környezetben végeztem.

A kidolgozott megoldás akár a tömeggyártásba kerülő autók vezetést segítő szoftverrendszerébe is integrálható, így lehetőséget ad arra, hogy a járművek a sofőrnél korábban észleljék a veszélyhelyzeteket, és ezáltal biztonságosabbá tegyék a közlekedést.

Abstract

Nowadays, the driving assistance technologies have been getting more and more significance in the automotive industry. An important task of driving assistant systems is the recognition and evaluation of traffic situations. One specific task is the determination of the current traffic maneuvers and those to be performed. Situation analysis can help to do that.

The situation analysis is based on uncertain data, therefore in most cases no confident decision can be made during the evaluation of the situation. In these cases the maneuvers have to be handled in a probabilistic approach.

In my BSc thesis, first, I introduce basics of the probabilistic models, and then I describe my development in details. The lane changing recognition and its forecasting method is written in MATLAB environment. The information is provided by the radar and camera systems of the vehicle and applied to the model

The developed software component can provide a basis for the software of cars in mass production. The advanced situation recognition facilitates the avoidance of dangerous traffic situations and makes the travelling safer.

1 Bevezetés

Napjaink modern technikával felszerelt járműveiben rengeteg balesetmegelőző, vezetést segítő, illetve részben, vagy teljes mértékben automatizált vezetést biztosító szolgáltatás található. Ezen szolgáltatások biztonságkritikus rendszert alkotnak, ami megköveteli, hogy a rendszer ne kerülhessen nem megengedett állapotba. Továbbá ezeknek a rendszereknek az emberi élet védelme előírás, ezért az egészséget veszélyeztető tényezők bekövetkezését meg kell előzniük. A gazdasági vagy környezeti károkat okozó szituációkat a lehetőségekhez képest minimalizálniuk kell [1][2].

Az mai értelemben vett vezetést segítő rendszerek (továbbiakban: DAS, driver assistance systems) kezdetben még csak a felsőkategóriás járművekben voltak jelen, az utóbbi időkben viszont kezdenek egyre általánosabbá válni. Elterjedésének fő okai, hogy egyre olcsóbbak és megbízhatóbbak ezen technológiák, így szinte bármelyik autóba lehet igényelni ilyen szolgáltatásokat. Ezek közül elsőként az autós helymeghatározó rendszerek jelentek meg a piacon közvetlenül az után, hogy a GPS (Global Positioning System) általánosan is elérhetővé vált. 1995-ben az adaptív sebességszabályozás még csak nagyon kezdetleges formában létezett: a fék használata nélkül, a gázadás kontrollálásával érték el a megfelelő sebességet. Ekkor még csak lézer technológiával (Ladar) dolgoztak, később vezették be a rádióhullám alapú (Radar) érzékelőket. Az elektronikai eszközök széleskörű elterjedése előtt csak formatervezési megoldásokat alkalmaztak a sofőrök látóteréből kitakart holtterek csökkentésére, például jobb vezetési pozícióval, vékonyabb tetőoszlopokkal, illetve domború visszapillantó tükrökkel kívántak javulást elérni. Viszont a korszerű autókban már számítógépes felügyelet, hangjelzések és vizuális megjelenítés is segítheti a holtter megszüntetését, és így a veszélyes helyzetek megelőzését.

Az ilyen rendszerek tervezése és megvalósítása hatalmas feladatot és nagy felelősséget ró a mérnökökre, hiszen a kifejlesztett szoftverek a gépjárművekben biztonságkritikus beágyazott rendszerek elemeiként szolgálnak. Ezen járművek számos aktív és passzív szenzor segítségével figyelik meg a környezetüket, majd a kinyert adatok előfeldolgozása után különböző szoftvermodulok használják fel az érzékelők jeleit. A legtöbb biztonsági funkció (vészfékezés, elektronikus menetstabilizáló rendszer,

légszákokat vezérlő rendszer) és a komfort funkciók is (start-stop rendszer, parkoló automata) támaszkodnak ezen információkra.

Munkám során egy DAS szoftvermodul fejlesztésében vettem részt, ezen belül a nagy hatótávolságú radar és a fedélzeti kamera adatainak a feldolgozásával foglalkoztam. Ez a két rendszer egymástól függetlenül (többek között a redundancia elvének megfelelően) detektál a különböző típusú tárgyakat, amelyek fúziójával egységeket generál (objektumok). A fúzió annak érdekében történik, hogy az objektumok pozíciója, típusa és egyéb funkciókhoz szükséges további jellemzői pontosabbak legyenek.

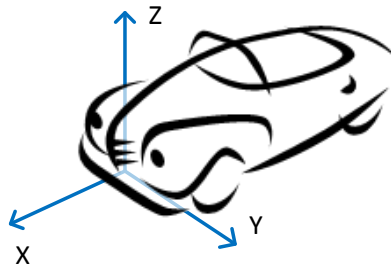
Szakdolgozatomban egy általam fejlesztett szoftvermodult ismertetek, ami becslést ad a környezetben lévő autók sávváltási szándékára. Ehhez egy valószínűségi értéket rendel annak becslésére, hogy egy előre meghatározott időintervallumon belül bekövetkezik-e sávváltás. Ezt a valószínűségi értéket később a DAS egyes funkciói a döntéseikhez is fel tudják használni. Így szolgáltathat információt az adaptív sebességszabályozás (Adaptive Cruise Control, ACC) vagy az önjáró autók mesterséges intelligenciája részére. Ebből a megoldásból az az előny származhat, hogy a veszélyes szituációkat hamarabb felismeri a rendszer, illetve komfortosabb utazást tesz lehetővé, ha egy másik jármű sávváltását korábban detektáljuk.

A megoldás során a magyar közutakon hatályos jobboldali közlekedést, és a KRESZ szabályait vettem alapul. A kifejlesztett modul felhasználja a környezetben észlelt objektumok fizikai jellemzőit (koordináta rendszerbeli pozíciók, objektumok egymáshoz viszonyított sebessége, stb.), objektumtípus osztályozásának eredményét, illetve a felismert forgalmi elrendezés modelljét.

2 Felhasznált autóiipari rendszerek

Ebben a fejezetben bemutatom az általam felhasznált rendszer legfontosabb részegységeinek a működését, amely során két szenzort elemzek autóiipari felhasználásuk szempontjából, illetve a sávfelismerésért felelős modul is ismertetem.

A fejezetben bemutatott rendszerek mind a környezet feltérképezését hivatottak szolgálni. A detektált céltárgyakat egységesen egy derékszögű koordinátarendszerben helyezik el. Középpontjában a mérést végző jármű található (továbbiakban: EGO, a latin eredetű „én” szóból). A tengelyek irányítása a 2.1 ábrán látható.



2.1. ábra Autóiipari koordinátarendszer

2.1 Szenzorok

A szenzorokat két nagy csoportba sorolhatjuk a működési elv alapján: aktív illetve passzív szenzor. Az utóbbi eszközök a mérendő elem által kibocsátott jeleket, vagy más forrásból származó visszavert jeleket detektálnak. Az aktív szenzorok ezzel szemben saját forrást használnak ahhoz, hogy detektálják a megfigyelni kívánt elemek tulajdonságait. Alább látható néhány autóiipari alkalmazásuk [3].

Aktív szenzorok:

- Radar (távolság és sebességmérés, részletesen alább olvasható)
- Ultrahang szenzor (távolságmérés, parkolás segéd)
- Lambda-szonda (oxigénszint mérés, kipufogógáz)

Passzív szenzorok:

- Segédfény nélküli kamerák
- Hall szenzor

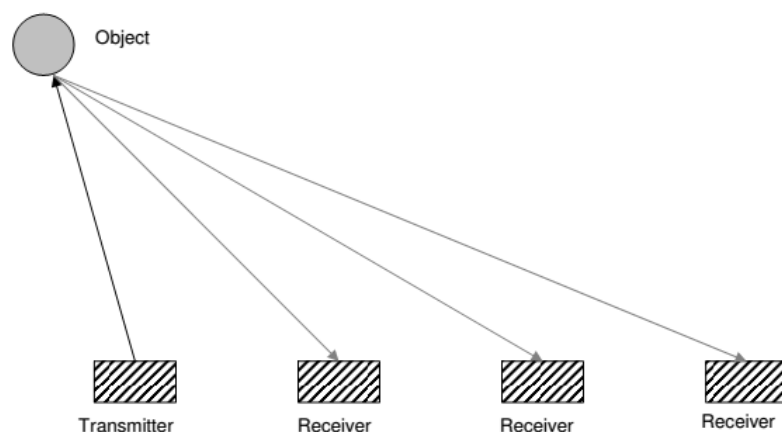
- Hőmérséklet mérő
- PIR szenzor (objektum detektálás és osztályozás, éjjel látó kamera)

Az autópárhban az alábbi nagy hatótávú távolság- és sebességmérésre alkalmas technológiák [3] állnak jelenleg rendelkezésre:

- Radar
- Ladar (Laser detection and ranging), Lidar (Light detection and ranging)
- Passzív IR
- Gépi látás (kamera)

2.1.1 Radar

A radar egy aktív elven működő, távolág és sebességérzékelésre alkalmas eszköz. Az általam használt eszköz „multistatic” felépítésű (több antennával végzi a mérést és mozgóalkatrész nélküli kialakítással rendelkezik), amelyet a 2.2. ábra mutat. A radar FMCW elven működik (Frequency-Modulated Continuous-Wave). A szétválasztott adó és vevő antennáknak köszönhetően lehetővé teszi a jel egyidejű adását és vételét is. A folytonos jelsugárzás miatt rövidebb mérési időt szolgáltat, mint az egyantennás, impulzus alapú eszközök [4].



2.2. ábra Mérés multistatic radar segítségével [5]

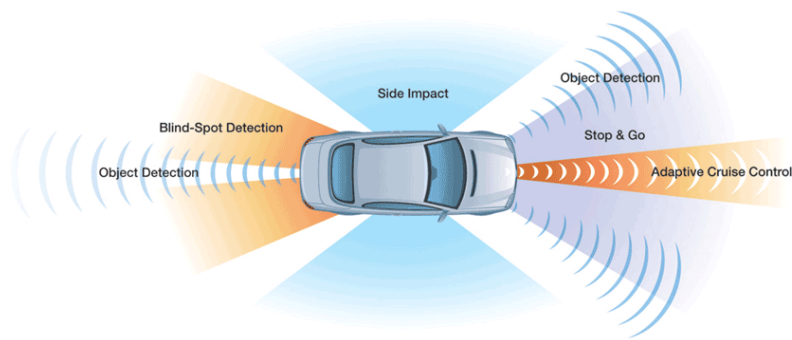
A radar alkalmazásának szemléletes példája az objektumok sebességének meghatározása. Ehhez a doppler-effektust lehet kihasználni, aminek lényege, hogy az

eltérő sebességű objektumokon a rádiósugárzás meghatározott frekvencia eltolódást szenved.

Többobjektumos rendszerekben problémát okozhat a több vett jel érzékelése (különböző sebességek esetén, több különböző vett frekvencia formájában jelenik meg). A megoldást a lineáris frekvenciamoduláció szolgálja, aminek a lényege, hogy időben változó frekvenciával sugározzuk a jelet. Az adás és a vétel időkülönbségének alapján pedig a jel által megtett utat lehet meghatározni, ebből pedig az objektum távolságát. A többobjektumos rendszer miatt további problémákat okozhat a többszörös visszaverődés. Az útról, a falakról és más járművekről közvetetten is érkezik jel, ami számos fantom objektum észlelését okozza. Megoldást jelenthet a bonyolultabb frekvenciamoduláció, a redundáns mérések, a szoftveres szűrőalgoritmusok, illetve a frekvenciasáv növelésének együttes alkalmazása [5].

Pontos pozíció adatok szolgáltatásához szükség van még az irányszög meghatározására. Ennek számítására a „multistatic” eszközökben két megoldást használnak leggyakrabban: a fázis- illetve a térerősségek különbsége az elvét. Az előbbi teljesen egyforma vevőantennákat használ, és ezek térbeli elhelyezkedéséből eredendően mindegyikhez különböző fázissal érkeznek be a jelek, majd az időkülönbségük alapján határozza meg az irányszöveget. Ennek a megoldásnak a hátránya, hogy nagy távolságból (>100 m) a kis fáziskülönbség növeli a pontatlanságot, jellemzően nagy látószögű eszközökben alkalmazzák. A szakdolgozatomban használt radar eltérő karakterisztikájú antennákat használ a nagyobb hatótávolság eléréséhez. Az eszköz működése során adott irányból érkező jel pontosan meghatározott térerősség eléréseket hoz létre a különböző antennákon, és ez alapján határozza meg az objektum irányszögét.

Az aktív eszközök által kisugárzott teljesítmény miatt nagyobb villamos energia igényrel rendelkeznek a passzív érzékelőkhöz képest. A radar esetén a stabil áramforrást, a nyomtatott áramkörök hűtését, és az antennák megfelelő hőmérsékleten tartását is biztosítani kell.



2.3. ábra Radar felhasználása a járművekben

A radarokat elterjedten használják az autóiparban is, 2.3. ábra látható DAS funkciók az alábbi tulajdonságú radarokat igényelnek:

- Nagy hatótávolságú radar:
 - akadály detektálás (Object Detection)
 - ACC (Adaptive Cruise Control)
- Nagy látószögű radar:
 - akadály detektálás (Object Detection),
 - Stop & Go
 - Oldalsó becsapódás érzékelés (Side Impact)
 - Holttér figyelés (Blind-Spot Detection)

2.1.2 Gépi látás

A szakdolgozat feladat megoldása során használt passzív elven működő sztereó videó kamerát alkalmazok, amely két egyforma szenzort tartalmaz, így a mono kamerák előnyeit 3 dimenziós térérzékeléssel egészíti ki. Ez a megoldás jelentősen javítja a távolságmérést, és ebből kifolyólag a sebességmérést is. A felismert objektumok irányszögét egyértelműen meg lehet határozni a rögzített képen való elhelyezkedés alapján. Ennek az értéknek a pontosságát nagymértékben korlátozza az objektum távolsága, illetve a kamera felbontása. Az útra való legjobb rálátás érdekében a 2.4. ábra által szemléltetett helyen, a szélvédő mögött, a legmagasabb pozícióban szokás elhelyezni a kamerát [6].



2.4. ábra A beépített fedélzeti kamera elhelyezése

A kamera két érzékelője nemlineáris CMOS félvezető technológiát alkalmaz, és a látható fény tartományában detektálja a beérkező sugárzást. Ez a technológia az emberi szem működéséhez hasonlóan dinamikusan képes a fényerősség érzékelését szabályozni, ami az eltérő fényviszonyok mellett is nagy kontraszt értékű és éles képet rögzít [7].



2.5. ábra A fedélzeti kamera képe

Digitális képfeldolgozás útján számos további jellemzőket lehet detektálni az objektumokról, színinformációkon túl élek és alakzatok felismerésére is alkalmas. A kamera szoftvere a gépi tanulás segítségével mintákhoz objektumokat rendel és ezeket különböző objektum osztályokba sorolja. Ennek segítségével lehetővé válik a forgalmi akadályok, a gyalogosok, a járművek, az útburkolati jelzések, a közúti jelzőtáblák, vagy

akár a forgalomirányító jelzőlámpák azonosítása. Az általam felhasznált célobjektumok a járművek osztályán belül, a személygépkocsik és a tehergépkocsik. A motorkerékpárok eltérő menetdinamikai tulajdonságai miatt jelen szakdolgozatban nem kerül tárgyalásra.

A pozíció és sebesség információk szolgáltatása, illetve objektum osztályozási funkciója alkalmassá teszi a kamerát, hogy vezetést segítő rendszereket támogató eszközzé váljon.

2.1.3 Szenzorok összefoglalása

Annak érdekében, hogy a vezetést segítő rendszerek, és egyéb kiszolgált funkciók minél korszerűbb szintet érjenek el, a multiszenzoros mérési adatokat fuzionálják. A különböző érzékelők előnyeit egyesítve, a legpontosabb, és legmegbízhatóbb adatokat szolgáltatja a rendszer. Az általam használt szenzorok technikai jellemzőinek összehasonlítása a 2.1. táblázatban látható [4].

Szenzor	Radar	Kamera
Detektálás hatótávolsága	< 250 m	< 100 m
Horizontális látószög	30°	45°
Frissítési ráta	80 ms	180 ms
Előnyök	- időjárási körülményektől független működés - pontosabb távolság és sebességmérés - robosztus	- pontosabb objektum osztályozás - széles látószög - pontosabb irányszög mérés
Hátrányok	- drága telepítési költség - pontatlan irányszög mérés	- pontatlan távolságmérés - jó fényviszonyok szükségesek
Egyéb adatok	frekvenciasáv: 77 GHz moduláció: FMCW	nemlineáris CMOS szenzor

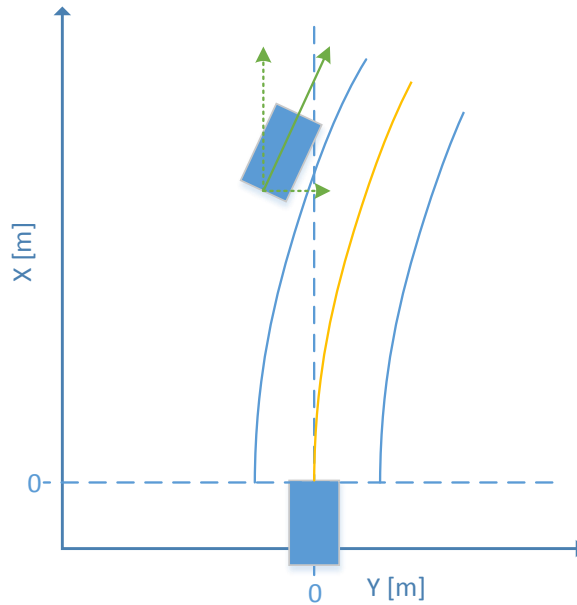
2.1. táblázat Szenzorok összehasonlítása

A 2.1. táblázatban ismertetett szenzor adatok fúziójával átlagos autópályás körülmények között biztosítható a helyes működés, és így biztonságkritikus vezetést segítő rendszerek alapjául szolgálhat.

2.2 A sávmodell

A fedélzeti kamera másik fontos feladata az útdatok és sávelrendeződések felismerése. A perspektivikus kép bázistranszformációval van átalakítva axonometrikus (felülnézeti) képpé egy szoftvermodul által. Ez a térképszerű megjelenítés egy kétdimenziós, derékszögű koordinátarendszeren alapul, középpontjába a mérést végző járművet helyezi (továbbiakban: EGO, a latin eredetű „én” szóból). A fejlesztéshez csak egyetlen sáv adatait használom, azt, amelyikben az EGO legnagyobb valószínűséggel tartózkodik. E korlátozás oka, hogy csak az EGO sávjával kapcsolatos eseményekre teszek következtéseket, amelyeket a 4. fejezetben tárgyalok részletesen.

A bázistranszformáció első lépése, hogy a felismert világos színnel jelzett útburkolati jelzéseket egymáshoz rendeli, majd ez alapján polinom függvényekkel közelítve írja le azokat. E függvények szakaszonként jó közelítéssel írják le az EGO előtt és után található forgalmi sávok vonalvezetését (a szakaszok és így a leíró függvények mennyisége függ az EGO sebességtől, illetve az adott forgalmi szituációtól). A gyorsabb feldolgozás, illetve pontosabb eredmények érdekében a radar által detektált sávelválasztó korlátokat, és a kormányzó alapján feltételezett útvonalat is figyelembe veszi a fúzió során. A fedélzeti számítógép adatbázisából irányított lekérdezés során kinyerem az adott sávhoz tartozó függvények együtthatóit, amelyeket a sávváltás vizsgálatához használok fel. A 2.6. ábra mutatja az alkalmazott sávmodellt, amelyen narancsszínű görbével szemléltetem, hogy kanyargós útvonal esetében hogyan követhetőek a sávok íveit. Teljesen egyenes útszakasz esetén konstans függvényeket kapok, amelyek egyenes szakaszként jelennek meg.



2.6. ábra Az alkalmazott sávmodell

A hálózatba kötött beágyazott rendszerekben a függvényekkel való közelítés előnnyel bír a koordináta pontok tárolásával szemben. Az autóiiparban használt adatbuszokon (CAN, LIN) a továbbított információk mérete, és az üzenetek mennyisége kritikus lehet az adattárolások alkalmával. A polinom együtthatós módszer sokkal kevesebb tárhelyet igényel és a feladathoz illeszkedő pontosságú adatokat szolgáltat.

A sávmodell által szolgáltatott információkból a következő adatokat dolgozom fel:

- X-koordináta
- Y-koordináta
- Y-koordináta varianciája
- Sáv választó típusa
- Sáv létezésének valószínűsége

3 Valószínűségi modellek

Jelen fejezetben a feladatom megoldására alkalmazható valószínűségi modelleket röviden ismertetem. Ezek a modellek a Bayes-hálók és a Markov láncok.

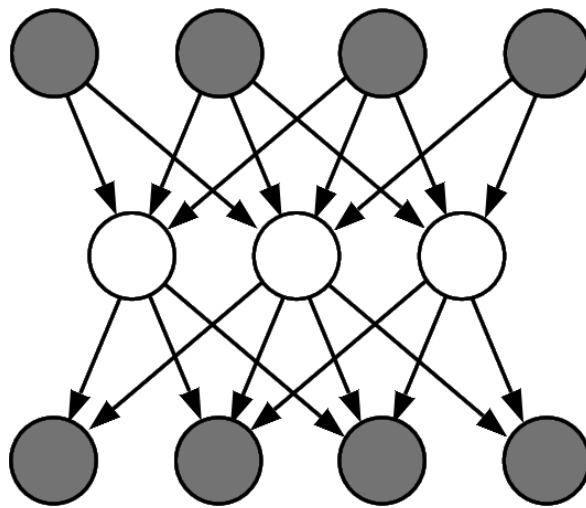
A matematika azon ágát, ami véletlenszerű jelenségekkel foglalkozik, valószínűségi számításnak nevezzük [17]. A tárgyalt valószínűségi modellek eszközként használhatóak fel az eseménytér szimulálásához, múltbéli adatok (statisztikák) alapján becsülnek, és megadják az esemény újbóli bekövetkezésének valószínűségét.

Alapfeltevéseink:

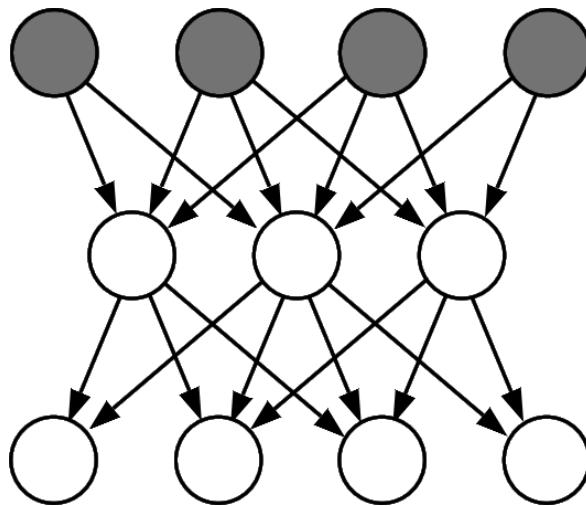
$P(A)$	„A” esemény valószínűsége a teljes eseménytéren (relatív gyakoriság)
$P(A, B)$	„A” és „B” események valószínűsége a teljes eseménytéren (események szorzata)
$P(A B)$	„A” esemény feltételes valószínűsége , ha „B” esemény teljesült
„A” és „B” események függetlenek , ha a két esemény együttes bekövetkezésének valószínűsége megegyezik a két esemény független bekövetkezési valószínűségeik szorzatával: $P(A, B) = P(A) * P(B)$	

Grafikus modelleknek nevezzük az olyan valószínűségi modelleket, amik szemléltetéséhez gráfokat használunk. A gráf csúcsai a vizsgált eseménytér valószínűségi változói, a folyamat állapotát vagy vizsgált értéket írja le. A gráf élei reprezentálják a csúcsok között lévő függőségeket. A továbbiakban csak az irányított élekkel rendelkező típusokat ismertetem.

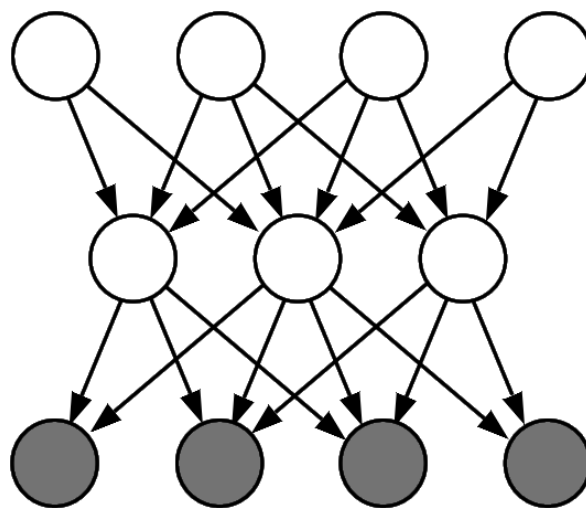
A 3.1. ábrán a grafikus modellek 3 használati módját szemléltetem:



(a)



(b)



(c)

3.1. ábra (a) Tanítás, (b) Előrejelzés, (c) Diagnózis, szabályozás [12]

A fenti ábrákon legfelső sorban vannak a bemenő adatok (input), középső sorban a közbenső viszonyok (lehetséges többszintű is), alsó sor pedig az eredményt (output) szimbolizálja. Sötét kiemeléssel vannak jelölve azok a csomópontok, amiket megfigyelünk, vagyis pontosan tudjuk az állapotát vagy az értékét. A világos színű csúcsok az ismeretlen változókat ábrázolják, korábbi ismereteink alapján ezekre teszünk következtetéseket illetve előrejelzéseket [11][13].

A 3.1. (a) ábrán a modell tanítása látható, mely során a be- és a kimenetek között egy függőségi rendszert hozunk létre.

A 3.1. (b) ábrán csak a bemeneteket figyeljük meg, és a kimenetre (esetlegesen a közbenső pontok állapotára) a statisztikáink alapján teszünk következtetéseket.

A 3.1. (c) ábrán a kimenő állapotokat érzékeljük, és a kiváltó okokra adunk valószínűségi becslést. Az orvosi alkalmazásokban a tünetektől következtetünk a betegségekre, például a májelégtelenség diagnosztizálása [18].

A grafikus modellek használatának legfőbb előnyei:

- Hűen reprezentálják a bizonytalanságot és az adatokban lévő zajt
- Automatizálhatóak
- Robusztus szerkezettel rendelkeznek
- Jól kezelik a nagy mennyiségű adatot

3.1 Rejtett Markov modell

A Rejtett Markov modell (továbbiakban: RMM) egy széles körben alkalmazott statisztikai modellező eszköz, amely diszkrét idejű valószínűségi következtetéseket határoz meg [11] [13].

Az RMM tipikus alkalmazásai:

- hang- és beszéd felismerés;
- fordítógépek;
- mikrobiológiai kutatások (fehérjefelismerés)
- viselkedés felismerés.

A modellben az állapotot egyetlen diszkrét valószínűségi változó írja le (X). Ennek a változónak a véges hosszúságú vektorba rendezett időbeli állapotai Markov folyamatot alkotnak ($\mathcal{F} = [X_1, X_2, \dots, X_N]$). Ez a folyamat Markov-tulajdonsággal rendelkezik, azaz annak a valószínűsége, hogy a rendszer hogyan viselkedik a jövőben, csak az aktuális állapottól függ, az előzményeknek nincs szerepe. Azaz:

$$P(X_i = x | \mathcal{F}) = P(X_i = x | X_{i-1}) \quad (1)$$

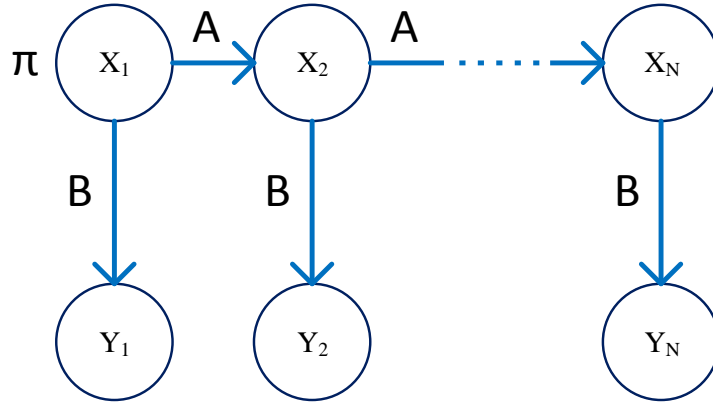
Ezt a tulajdonságot szemléletesen memóriamentességnek nevezhetjük. Az (1) egyenletértékeit $N * N$ -es méretű állapotátmeneti táblába tudjuk rendezni, amelyet az 3.1. táblázat mutat, ahol N az állapotok számát jelöli.

$X_i \backslash X_{i-1}$	x_1	x_2	x_N
x_1	A(1,1)	A(1,2)	A(1,N)
x_2	A(2,1)	A(2,2)	A(2,N)
	⋮	⋮		⋮
x_N	A(N,1)	A(N,1)	A(N,N)

3.1. táblázat Az állapotátmeneti tábla

Az RMM alkalmas változó hosszúságú (elemszámú) jellegvektorok felismerésére.

Az állapotot nem tudjuk teljes bizonyossággal megmondani, csak a környezet egyéb tulajdonságairól következtetünk rá. Ezt a következtetést kiolvasásnak nevezzük. A kiolvasási művelet is a feltételes valószínűséggel zajlik a mért értékeink és az állapot közötti függőségek alapján (statisztikai alapon).



3.2. ábra RMM modell ábrázolása

- X_i csúcsok: a rejtett állapotok
- Y_i csúcsok: leolvasott tulajdonság
- B élek: kiolvasás
- A élek: állapotátmenet
- π csúcs: kezdő állapot (X_1)

Egyszerű, bináris példa lehet egy RMM típusú rendszerre az a vezetést segítő funkció, ami vizsgálja, hogy egy út szélén álló jármű ki szándékozik-e hajtani elénk. Az előrejelzés eredményét felhasználhatná az ütközés megelőző funkció, a parkolási asszisztens a szabad helyek keresésekor, vagy egy önjáró autó elengedné a buszmegállóból elindulni készülő buszt.

Az állapothalmaz kizárólag annyiból épül fel, hogy álló jármű kihajt-e az adott pozíciójából. Erre a rejtett állapotra kizárólag abból következtetünk, hogy mennyi indexlámpa villanást detektáltunk. Az egyelemű kiolvasási tér bizonytalansága abból adódik, hogy a feltételezett indexvillanás helyett esetleg a napfény vagy egy másik jármű fényszórója tükröződött vissza, amely téves detekciót okozhat.

A kiolvasási tér illetve az állapotátmeneti valószínűség képletek alább láthatóak (K: kihajtás, I: index villogása)

$$A = P(K_{i+1}|K_i) \quad (2)$$

$$B = P(K | I) \quad (3)$$

Az autó szoftvere által gyűjtött adatok alapján előállított statisztikai táblázatok:

A

	K_i^0	K_i^1
K_{i-1}^0	0,95	0,05
K_{i-1}^1	0,2	0,8

3.2. táblázat: Állapotátmeneti táblázat

B

	K^0	K^1
I^0	0,7	0,3
I^1	0,6	0,4
I^2	0,1	0,9

3.3. táblázat Kiolvasási táblázat

A jármű szenzorai már távolról detektálják a parkoló autót, és a járművünkben található DAS azt becsli, hogy 4 másodpercen belül mekkora eséllyel indul el (a rendszerünk csak egész másodperces időközzel számol).

$$\begin{aligned}
 P(K_{t < 4}^1 | I = i) = & B(i + 1, 2) + \\
 & B(i + 1, 1) * A(1, 2) + \\
 & B(i + 1, 1) * A(1, 1) * A(1, 2) + \\
 & B(i + 1, 1) * A(1, 1) * A(1, 1) * A(1, 2)
 \end{aligned}$$

Az előbbi egyenlet adja meg a példánkban a kihajtás valószínűségét, ahol az i jelöli a legutóbbi periódusban számolt index villanások számát (2 Hz-es villogást feltételezve, i -hez [0,1,2] értékek társulhatnak).

Index jelzés nélkül $P=0,271$ valószínűségi értéket ad a képlet a kihajtásra. Egy ideális világban ez 0-hoz tart, de a jelen statisztikában előfordult, hogy a figyelmetlen sofőrök indexelés nélkül kihajtottak elénk. 2 index villanás esetén $P=0,914$ a kihajtás esélye. A téves riasztásokat okozhat a korábban említett hibás index észlelés, vagy a vészvillogó villogása.

A projectemet RMM-el megközelítve a folyamatban lévő sávváltás tekinthető a rejtett állapotnak, ennek az a következménye, hogy csak diszkrét állapotként van meghatározva. Ebben a módban osztályozóként üzemel, célom pedig egy elosztott valószínűségi változó számolása. Mivel az általam felhasznált adatok viszonylag nagy zajjal terheltek, a szűrőalgorithmusom nagymértékben támaszkodik a múltbéli értékekre annak ellenére, hogy ez ellentétben áll RMM-ek követelményeivel. Továbbá ez a modell arra az időtartamra sem tesz következtetést, hogy az adott jármű mikor fogja befejezni a sávváltási manővert.

3.2 Bayes-háló

A Bayes-háló (továbbiakban: BH) egy grafikus modell, ami valószínűségi kapcsolatokat jellemez a megfigyelt változók alapján. Irányított aciklikus gráf formájában ábrázolható, ahol a csomópontok a vizsgált tér halmazai, és az élek a halmazok között lévő függőségek. [8]

A Bayes tétel szerint, „B” hipotézis valószínűsége „A” megfigyelt tények esetén:

$$P(B|A) = \frac{P(B, A)}{P(A)} = \frac{P(A|B) * P(B)}{P(A)} \quad (4)$$

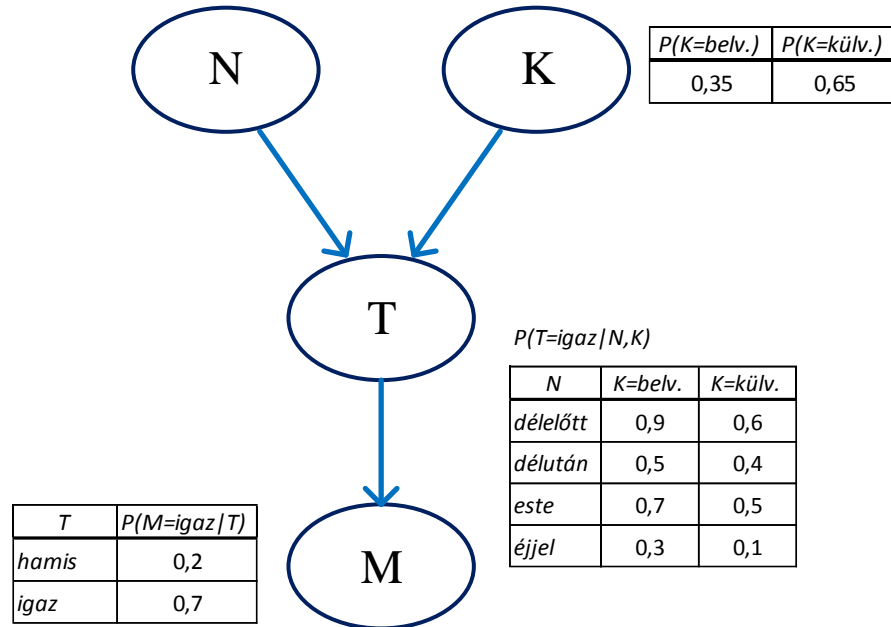
Ahol $P(B)$ illetve $P(A)$ a hipotézis illetve a tények valószínűsége. $P(A|B)$ pedig a feltételes valószínűsége a megfigyelt ténynek, a hipotézis teljesülése esetén. A Bayes tanulás célja, hogy a tényhalmaz összes lehetséges kombinációjára meghatározza a posteriori következtetést. „N” darab megfigyelt tény esetén általánosan egy N-változós függvény tartozhat a tapasztalati következtetésekhez. A gyakorlatban csak diszkrét eseteket vizsgálunk, amit egy feltételes valószínűségi táblázatba foglalunk (CPT, conditional probability table).

A valós feladatoknál abba a problémába lehet ütközni, hogy a teljes CPT olyan nagyméretű lesz, ami kezelhetetlen még a számítógépek számára is (túl nagy tárhelyet igényel, sokáig tart benne a keresés). Ennek a problémának a feloldására dekompozíciót alkalmazunk, amikor a teljes valószínűséget felbontjuk függőségi valószínűségekre:

$$P(x_1, x_2, \dots, x_n) = \prod_{i=1}^N P(x_i | pa_i),$$

ahol pa_i jelöli az megfigyelt tények részhalmazát, amiktől x_i feltételesen függ.

A 3.3. ábra egy példa egy egyszerű BH-t ábrázol, ahol az általános gyakorlatot alkalmazva a feltételes függést élékekkel jelöljük, ahol az él iránya a vizsgált hipotézis felé mutat [13] [16].



3.3. ábra Példa BH modell

A Bayes-hálók használatát egy egyszerű példán keresztül szemléltetem. A járművek különböző kényelmi funkcióinak szüksége van arra az információra, hogy egyes kereszteződéseknel mekkora a valószínűség, hogy megáll az autó. A 3.3. ábra táblázataiban szemléltetem a rendelkezésre álló városi közlekedési statisztikát. A forgalom torlódását (T) befolyásolja az adott napszak (N) és a kerület jellege (K). Ebben a világban a bel- és külváros kerületei egyértelműen szétválaszthatóak, adott napszakon belül homogén a forgalom eloszlása, illetve diszkrétén meghatározható, hogy van-e torlódás. A megállás valószínűségére (M) kizárólag a torlódás esélyéből következtetnek.

A példában egy útvonalat tervezek, ami számos kereszteződést tartalmaz. Az egyik kereszteződést bizonyosan az délelőtti időszakban érintem. Annak valószínűsége, hogy meg szükséges állni a járművel:

$$\begin{aligned}
 P(m_1) &= P(M = i | T = h) * P(T = h | N = de., K = belv.) + \\
 &P(M = i | T = i) * P(T = i | N = de., K = belv.) = \\
 &= 0,65
 \end{aligned}$$

Egy másik kereszteződésen az éjszakai napszakban haladok keresztül. Sajnos arról nincs információ, hogy milyen városi kerületben van. Így megállás valószínűsége:

$$\begin{aligned}
P(m_2) &= P(M = i | T = h) * (\\
&\quad P(T = h | N = de., K = belv.) * P(K = belv.) + \\
&\quad P(T = h | N = de., K = külv.) * P(K = külv.)) + \\
&\quad P(M = i | T = i) * (\\
&\quad P(T = i | N = de., K = belv.) * P(K = belv.) + \\
&\quad P(T = i | N = de., K = külv.) * P(K = külv.)) \\
&= 0,285
\end{aligned}$$

A megállásról szolgáltatott információt felhasználja az üzemanyag fogyasztásmérő, az akkumulátor töltő berendezés, és az útvonaltervező a hátralévő menetidő kiszámításához.

A Bayes-háló használatának előnyei [8]:

- változók közötti függőségeket kezeli hiányos bemeneti adatok esetén is
- ok-okozati viszonyokat képes felismerni és előre jelezheti a beavatkozások következményét
- valószínűségelméletet és ok-okozati téziseket egyesíti
- algoritmusokkal szolgál a tanításhoz és a predikcióhoz
- hatékonyan, és nagy pontossággal tud kezelni nagy adatbázisokat is.

A modell alkalmazásánál általában a legnehezebb feladat a változók közötti függőségek meghatározása.

A bizonytalan változók kezelése és a kimenet elosztott valószínűségi értéke miatt a Bayes-hálók alkalmasak a sávváltás előrejelzésének modellezésére, továbbá statisztikai alapú és előre definiált CPT-eket is kezel. A modellhez egyszerűen lehet új változókat hozzáadni illetve a változók felbontásának a növelése is megoldott.

4 A Sáv váltás

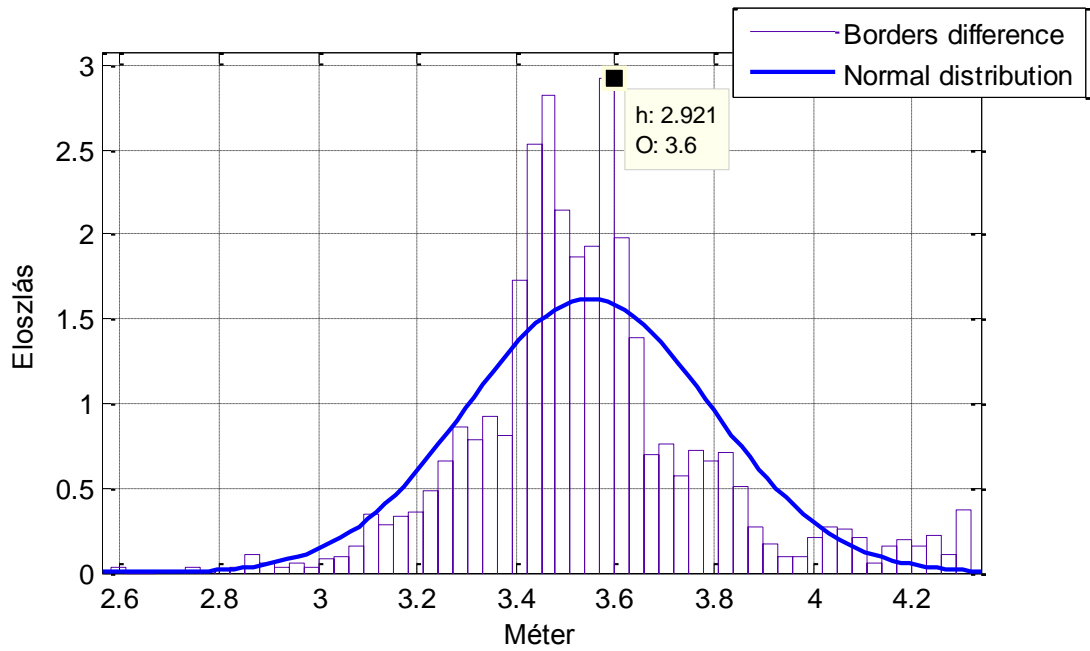
Ebben a fejezetben betekintést nyújtok a sáv váltás felismerésének módszeréről és a valószínűségi modell szükségességéről. Első pontban elemzem a sávokat és a sávokban haladó járművek jellemzőit, a második pontban a sáv váltás folyamatát tárgyalom.

4.1 Sávok és járművek kapcsolata

A sáv váltás észlelését a jármű-sáv illetve jármű-jármű kapcsolatával közelítem meg [14]. Továbbiakban kizárólag az EGO haladási sávjának eseményeivel foglalkozom, vagyis az ide való besorolással (szomszédos sávból való belépés) ill. az innen történő kisorolással (az EGO sávjának valamelyik irányba történő elhagyása). Ezt a korlátozást elsősorban azért teszem meg, mert a jelenleg használt DAS funkcióknak csak erre van szüksége, másodsorban megbízható adatok erről a területről állnak rendelkezésre.

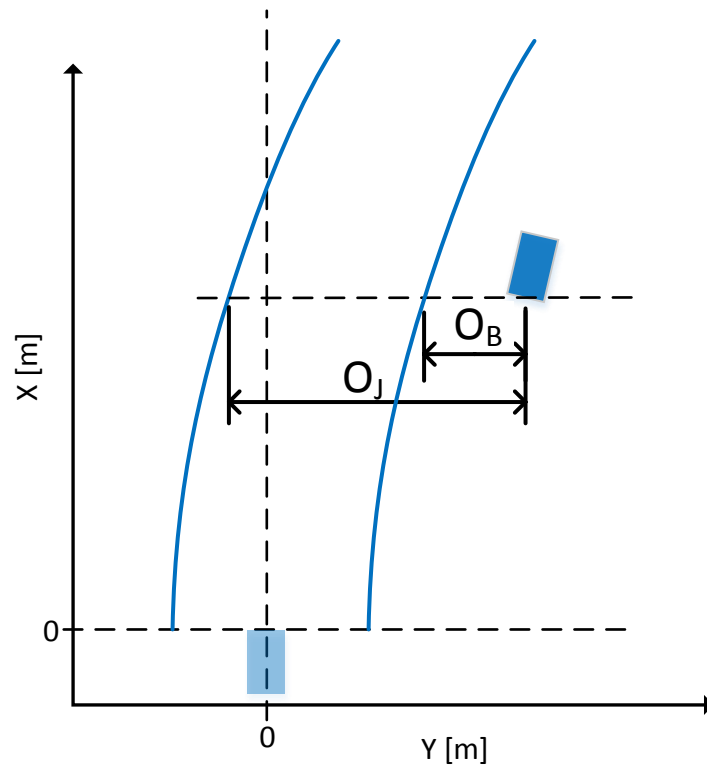
A sávmodell által szolgáltatott információk alapján a 4.1. ábrán szemléltetem az EGO sáv szélességének hisztogramját. A statisztika szerint a sávválasztók közötti távolság leggyakoribb értéke 3,6 méter, a várható értéke 3,54 méter, szórása 0,2465 méter, és ezt a grafikonra illesztett normáleloszlás is reprezentálja.

A MATLAB grafikonokon a *Datatip* eszközzel lehetőség van az ábrázolt pontokról további információkat kiírni. A *Datatip* alapértelmezetten a grafikonon szereplő x illetve y értékeket írja ki, azonban ez félreértésre adhat okot. Saját *Datatip* függvényekkel mindig az éppen megjelenített ábrákhoz igazítottam az eszközzel kiírt információkat.



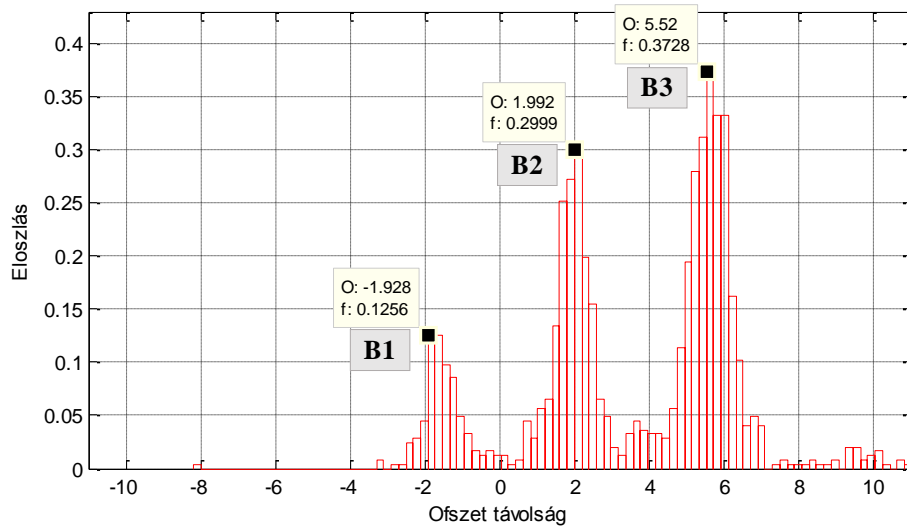
4.1. ábra Az EGO sáv szélességének hisztogramja és az illesztett normál eloszlás függvénye

A célobjektumok távolságát az EGO sáv két elválasztó jelölésétől az Y koordináták különbsége alapján határozom meg. A 4.2. ábra ezt mutatja be a bal ofszet (O_B) és jobb ofszet (O_J) mérésén keresztül.

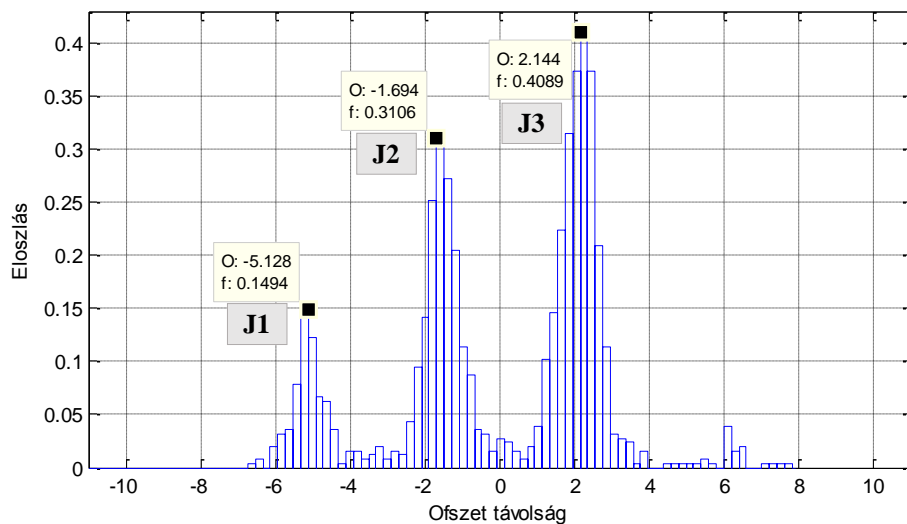


4.2. ábra Ofszet távolságok ábrája

A két ofszet érték hisztogramját a 4.3. ábra illetve a 4.4. ábra mutatja be. Ha pontosan az adott jelölés fölött van a jármű, akkor 0 értéket rendel az ofszethez, pozitív számot, ha tőle balra, és negatívot, ha ettől jobbra detektálta a rendszer. A diagramon „O” jelöli az ofszet értékét, „f” pedig a téglalap magasságát.



4.3. ábra A bal ofszet hisztogramja



4.4. ábra A jobb ofszet hisztogramja

A fenti két diagramot összehasonlítva első közös jellegzetesség a három nagy csúcs, ennek magyarázata, hogy a sávrendszerbe helyezett célobjektumokat ebbe a három sávba osztályozza a rendszer. A csúcsok maximum helyei:

$$O_{B1} = -1,928m; O_{B2} = 1,992m; O_{B3} = 5,52m$$

$$O_{J1} = -5,128m; O_{J2} = -1,694m; O_{J3} = 2,144m$$

A csúcsok maximuma közötti eltérés a 4.3 és a 4.4 ábra alapján:

$$O_{B2} - O_{B1} = 3,92$$

$$O_{B3} - O_{B2} = 3,53$$

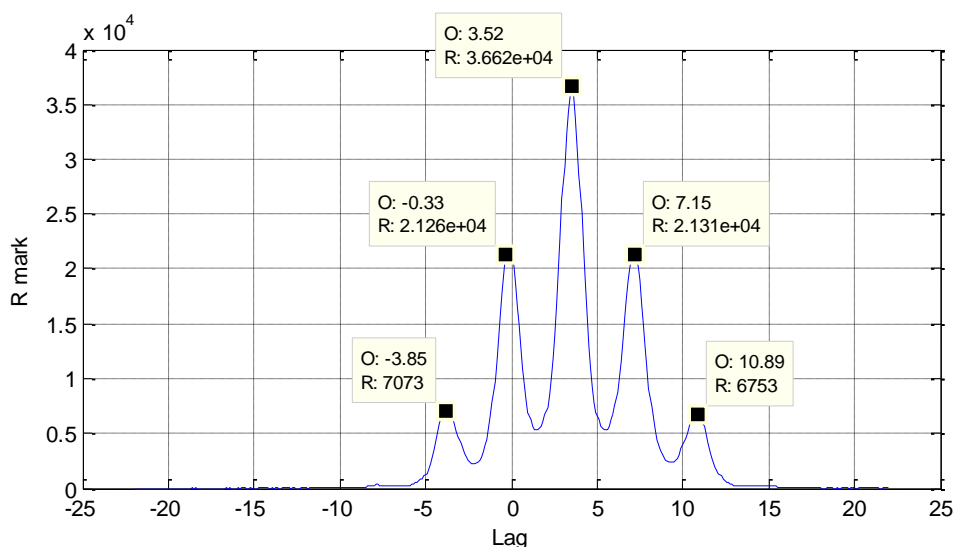
$$O_{J2} - O_{J1} = 3,44$$

$$O_{J3} - O_{J2} = 3,83$$

Az egyik diagram ofszet tengelyét eltolva nagymértékű átfedés fedezhető fel, ami pedig a fejezet elején magyarázott EGO sáv jelöléseinek laterális különbségére utal. A két hisztogram összehasonlítását keresztkorrelációval végeztem el, amihez a MATLAB `xcorr` beépített függvénye felhasználásával. Az összehasonlítás azon alapszik, hogy két jelet egymáshoz képest eltolva elemenként összeszorozza, majd összegzi. Diszkrét 1-dimenziós esetben $f[x]$ és $g[x]$ függvények keresztkorrelációs függvényének képlete:

$$R[t] = \sum_x f[x]g[x - t] \quad (5)$$

Az (5) kifejezésben, ha t olyan eltolást ad, amely mellett $f[x]$ és $g[x - t]$ hasonlóak, akkor magas lesz a szorzatösszeg értéke. A 4.5 ábrán látható a 4.3 és 4.4 ábrákon ismertetett ofszet hisztogramok keresztkorrelációja, legmagasabb pont 3,52 eltolás értéknél van, ami megfelel az elvárásainknak, mivel csak 0,57%-kal tér el az átlagos sáv szélesség értéktől. Az oldalsó csúcsok is ennek az értéknek a többszöröseinél helyezkedik el, ami csak tovább erősíti a modell helyességét. Ezt az összehasonlító módszert számos más területen is használják, hang- vagy magas frekvenciás jelen alapuló helymeghatározás, vagy aktív zajelnyomás területén is.



4.5. ábra Ofszet hisztogramok keresztkorrelációja

A 4.3. és a 4.4. ábrákból következtetni lehet az autók sávon belüli elhelyezkedésére is. Az esetek többségében közelebb helyezkednek el a jobb oldali sávhoz, mint a bal oldalihoz. Ezt bizonyítja, hogy a középső sávban tartózkodó járművek várható bal illetve jobb ofszet értéke rendre 1,99 illetve -1,69 méter, de a szomszédos sávokban is ugyanilyen eltérés figyelhető meg. A jelenségre magyarázatot adhat, hogy a vezetők a szembejövő forgalomtól nagyobb távolságot igyekeznek tartani, illetve kissé jobbra húzódnak, hogy az őket megelőző járműnek nagyobb oldaltávolságot hagyjanak.

A 4.3. és a 4.4. ábrák növekedő oszlopait, amely az autók relatív gyakoriságát jelentik, a jobbra tartási kötelezettség magyarázza, vagyis párhuzamos közlekedés (többsávos utak) esetén a belső sávokat lehetőleg csak előzésre használják a járművek.

4.2 A sávváltás folyamata

Sávváltásnak tekintem az EGO-val azonos sávban tartózkodást leíró valószínűségi érték tartós megváltozását. Ha sávváltás előtt az EGO-val azonos sávban volt a célobjektum, utána pedig valamelyik szomszédosban akkor azt kisorolási eseménynek értelmezem, fordított esetben pedig besorolásnak.

Minden feldolgozandó célobjektumról rendelkezésre áll két ofszet érték és a hozzájuk tartozó két relatív sebesség is (a számítást a következő fejezetben részletezem). Az ofszet (O) és a sebesség (V_{rel}) hányadosa határozza meg a sávváltáshoz szükséges időt (TLCH, time to lane change).

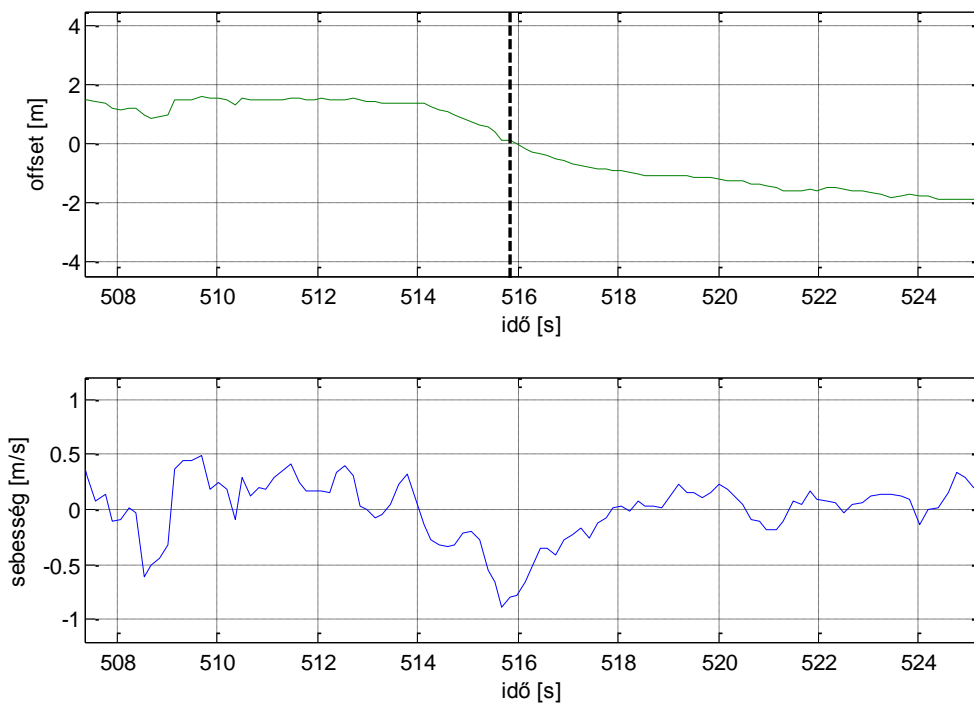
$$TLCH = \frac{O}{V_{rel}} \quad (6)$$

TLCH számolásához segédletként 4.1. táblázat értékeit használom, ahol az egyes ofszet és relatív sebesség (ofszet időegység szerinti megváltozása) tartományokhoz tartozó várható TLCH érték szerepel másodpercben kifejezve.

		sebesség						
		V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇
ofszet	O ₁	3,15	1,05	0,63	0,45	0,35	0,29	0,24
	O ₂	7,88	2,63	1,58	1,13	0,88	0,72	0,61
	O ₃	11,04	3,68	2,21	1,58	1,23	1,	0,85
	O ₄	14,19	4,73	2,84	2,03	1,58	1,29	1,09
	O ₅	17,34	5,78	3,47	2,48	1,93	1,58	1,33
	O ₆	20,5	6,83	4,1	2,93	2,28	1,86	1,58
	O ₇	23,65	7,88	4,73	3,38	2,63	2,15	1,82
	O ₈	28,38	9,46	5,68	4,05	3,15	2,58	2,18
	O ₉	34,68	11,56	6,94	4,95	3,85	3,15	2,67
	O ₁₀	44,14	14,71	8,83	6,31	4,9	4,01	3,4

4.1. táblázat TLCH értékek sebesség és ofszet függvényében

A 4.6. ábrán jelenítem meg egy tipikus sávváltáskor fellépő ofszet és relatív sebesség értékeket, továbbá vastagított fekete vonallal jelenítettem meg az általam detektált sávváltást.



4.6. ábra Egy tipikus sávváltás ofszet és relatív sebesség időgrafikonja

A 4.7. ábra jeleníti meg egy sávváltás elméleti nyomvonalát, ahol a longitudinális távolság a célobjektum haladási irányával párhuzamos. [14] Az ofszet kezdeti és végértékét O_0 és O_1 , a manőver longitudinális kezdeti illetve végértékét pedig X_0 ill. X_1 jelöli, illetve az X tengely szimbolizálja a sávelválasztót. A függvényt $[X_0, X_1]$ tartományon egy harmadfokú polinommal modelleztem, amelynek képlete:

$$y(x) = a_3(x - X_0)^3 + a_2(x - X_0)^2 + a_1(x - X_0) + a_0$$

Az együtthatók számolásához használt kiinduló adatok:

$$y(0) = O_0$$

$$y(X_1) = O_1$$

$$\frac{dy}{dx}(0) = 0$$

$$\frac{dy}{dx}(X_1) = 0$$

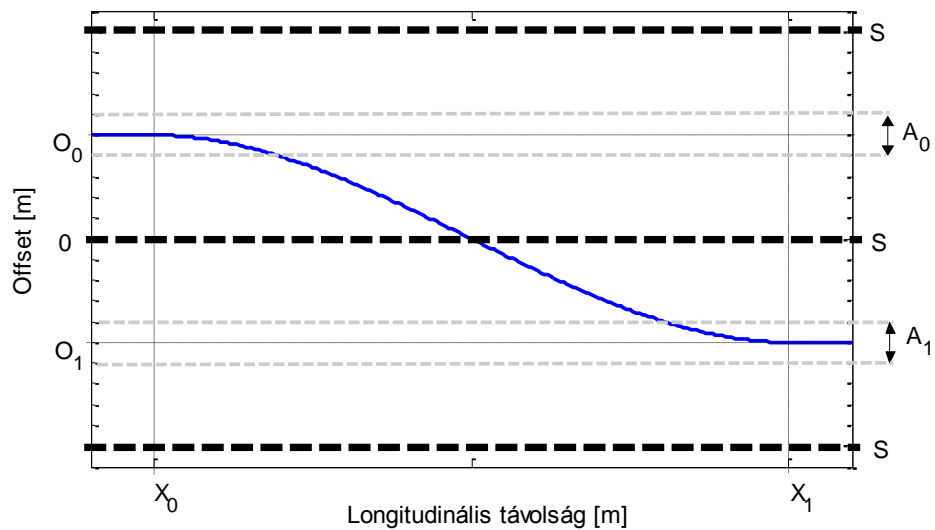
Az együtthatókra az alább látható eredmények adódtak:

$$a_0 = O_0$$

$$a_1 = 0$$

$$a_2 = 3 \frac{(O_1 - O_0)}{(X_1 - X_0)^2}$$

$$a_3 = -2 \frac{(O_1 - O_0)}{(X_1 - X_0)^3}$$



4.7. ábra Sávváltás térbeli nyomvonalának a modellje

A 4.7. ábrán szürke szaggatott vonallal jelöltem sávközép területeit (A_0 és A_1). Ezek mindig a sávok közepén helyezkednek el és a szélességük pontosan 20%-a. A sávváltás időtartama az az időtartam, ami a sávváltás folyamán a kiinduló sáv sávközepének elhagyása és a cél sáv sávközepébe érkezése között eltelik. A mérési szekvenciák szerint a sávváltás időtartama átlagosan 3,9 másodperc. Ennek meghatározása azért szükséges, hogy a teszteléskor megfelelő időtartamú szeletekre bonthassam fel a mérési szekvenciákat.

4.3 A sávváltás modellje

A sávváltás előrejelzés modellezésére a 3.2 fejezetben foglaltak alapján a Bayes-hálókat választottam. A sávváltás bekövetkezésének valószínűségét közvetlenül 3 független valószínűségi változóból származtatom:

1) Laterális bizonyíték (P_{lat})

A vizsgált jármű és a sávváltás Y-irányú viszonyából következtetek a sávváltás laterális bizonyítékára.

- O : offset [m]
- V_{rel} : relatív sebesség (a jármű és a sávváltás között) $\left[\frac{m}{s}\right]$

2) Infrastrukturális lehetőség (P_{layout})

Az infrastrukturális lehetőség az a valószínűségi változó, ami azt jellemzi, hogy a pillanatnyi forgalmi szituációban mekkora a valószínűsége egy esetleges sávváltásnak. A sávmodellből rendelkezésre álló adatok:

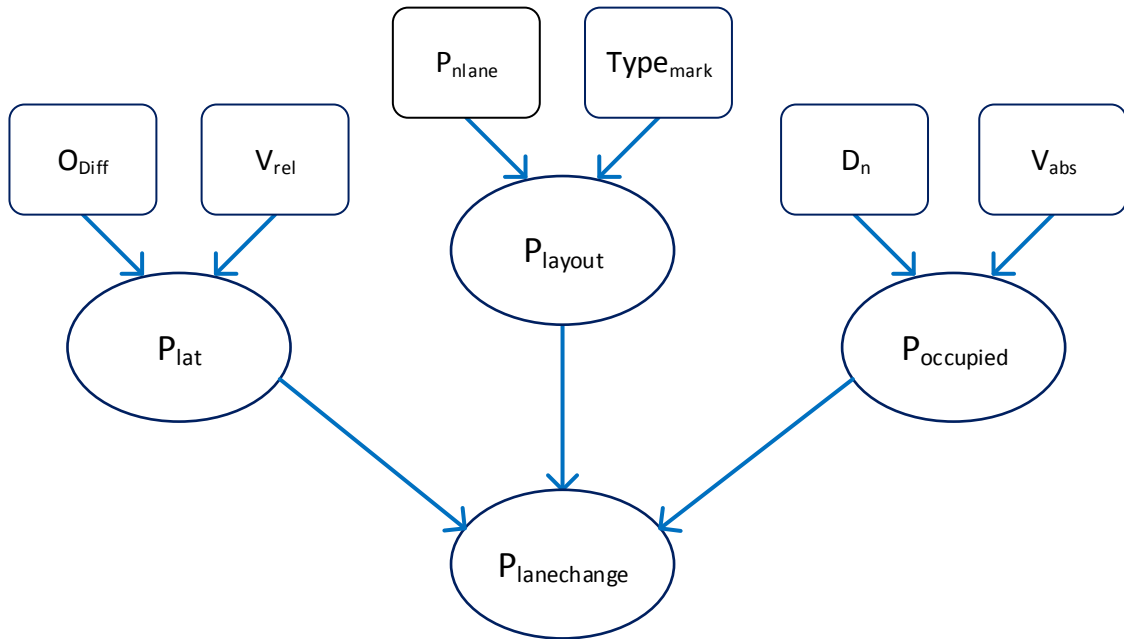
- P_{lane} : Szomszédos sáv létezésének valószínűségi értéke [1]
- $Type_{mark}$: A két sáv között húzódó elválasztó vonal típusa
[nincs vonal, szaggatott vonal záróvonat]

3) Sáv foglaltsága (P_{occup})

A vizsgált jármű sávváltás esetén bizonyos méretű helyet igényel a szomszéd sávban. Ennek a területnek a hosszát növeli a jármű utazási sebessége. Ennek a valószínűségét, hogy foglalt-e ez a terület, befolyásolja a vizsgált jármű és

szomszédos sávban haladó jármű távolsága (több jármű esetén a legkisebb távolság).

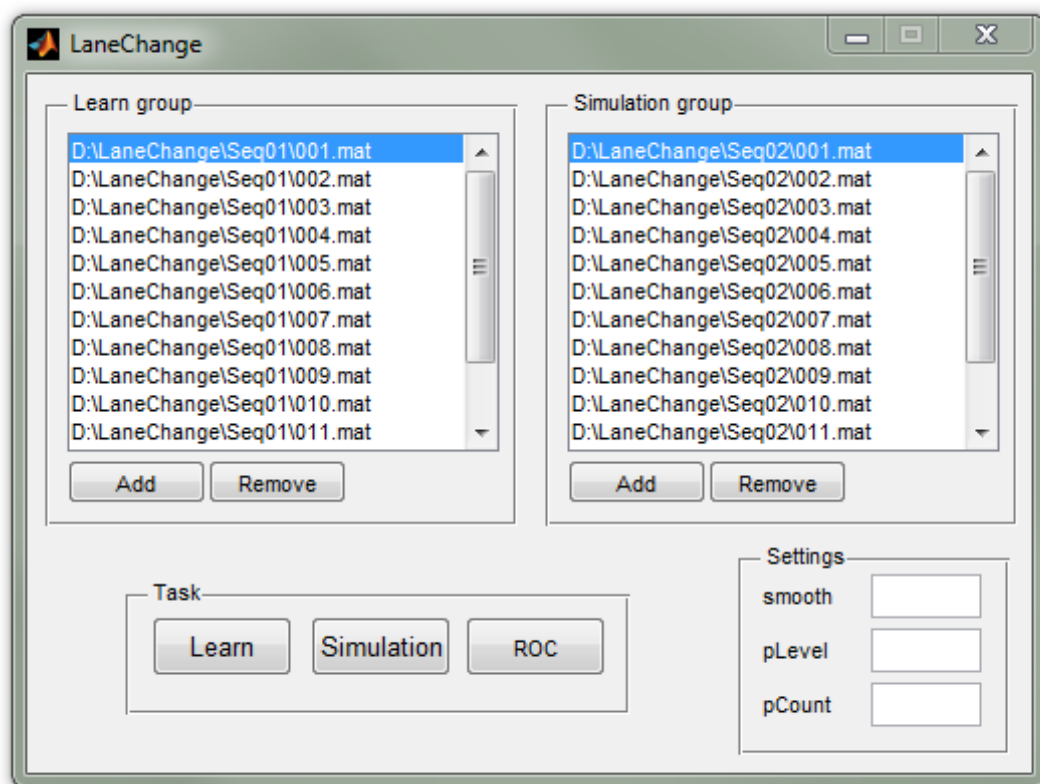
- D_n : A szomszéd sávban lévő jármű távolsága [m]
- V_{abs} : A jármű abszolút sebessége [$\frac{m}{s}$]



4.8. ábra Rendszerterv

5 A LaneChange program

Autóipari fejlesztések során tesztautók naplózzák a szenzorok méréseit, a folyamatok állapotát, és az adatbuszon küldött üzeneteket. Ezeket a napló fájlokat (továbbiakban szekvenciák) tároljuk további fejlesztésekhez, ellenőrzésekhez és későbbi kiértékelésekhez. A sávváltás felismeréshez és előrejelzéséhez fejlesztettem a LaneChange programot, ami ezeket a szekvenciákat használja fel. Az 5.1. ábra mutatja be a program grafikus kezelőfelületét.



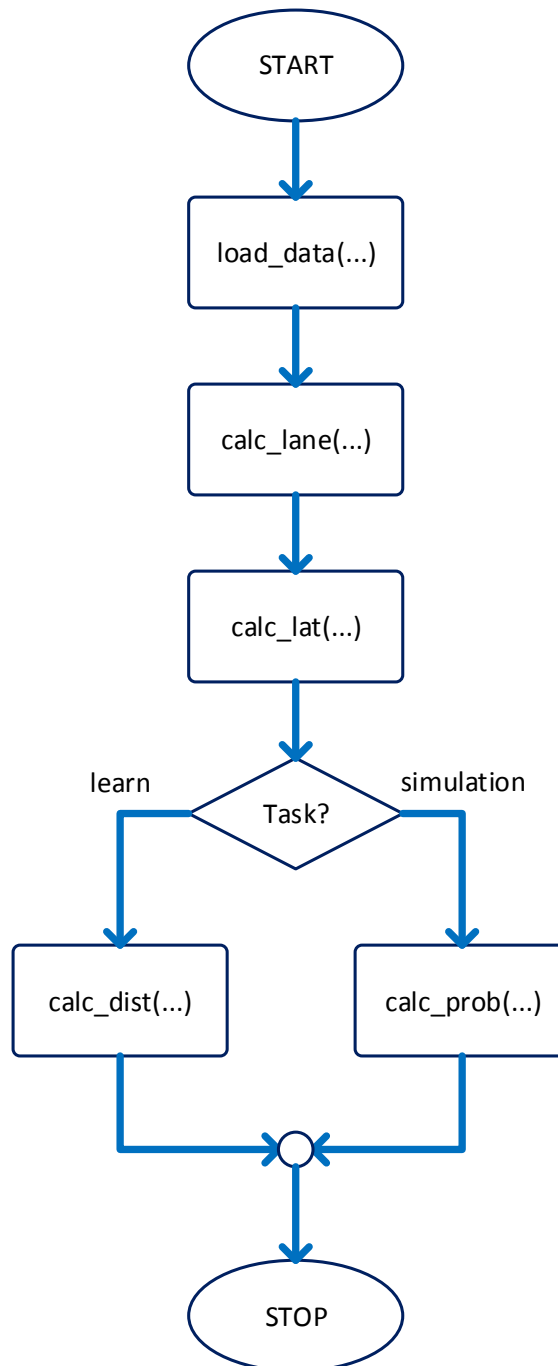
5.1. ábra A program kezelőfelülete

Az 5.1. ábrán szereplő panelok magyarázata:

- Learn group: a tanító szekvenciák kezelése
- Simulation group: a teszteléshez megjelölt szekvenciák kezelése
- Task: a feladatvégző taszkok indítása
- Settings: bemeneti küszöbérték paraméterek

5.1 Függvények

Az 5.2. ábrán látható a mérési szekvenciák feldolgozásának folyamatábrája.



5.2. ábra Fájlfeldolgozás folyamatábrája

A LaneChange program legjelentősebb függvényei:

Data_collect(...)

A `Data_collect` vezérli a fájlfeldolgozást és kapcsolódási felületet (interface) biztosít a GUI és a műveletvégző függvények között. A fájlfeldolgozás típusai:

- a) Új mérés: paraméterként kapott elérési úton található mérési fájl dolgozza fel, és létrehozza a "DAT" struktúrát.
- b) Mérés frissítése: paraméterként kapott "DAT" struktúra mezőit tölti fel (esetleg újraszámolja)

```
DAT = struct( ...  
            'filename',    '', ...  
            'AUX',        struct(), ...  
            'LANE',       struct(), ...  
            'LAT',        struct(), ...  
            'CPT',        struct(), ...  
            'Probs',      struct() );
```

A MATLAB függvényhívási struktúrájából adódóan, egy fájlban elhelyezhető több függvény, de csak az első látható kívülről (main function). További lokális függvények implementálhatóak a fájlban, de csak a main számára láthatóak. A szakdolgozatom során a GUI kezelést és a fájlfeldolgozási részfeladatokat is ilyen lokális függvényekkel oldottam meg. A műveletvégző függvények aktiválása a feladat típusától függően változik.

load_data(...)

Az sorban az első műveletvégző függvény szerepe:

- mérési adatokat betöltése
- funkció szerinti rendezés és csoportosítás
- mértékegység átváltások, normalizálások, és eltolások beállítása
- "DAT.AUX" segédstruktúrába kerül az eredménye

calc_lane(...)

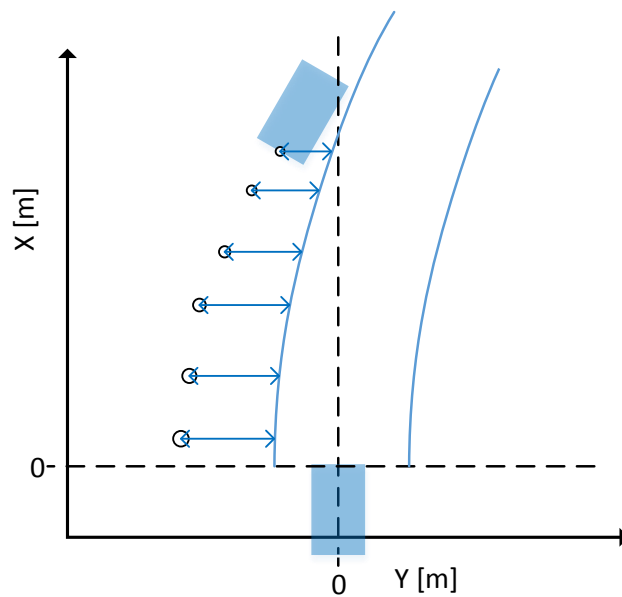
Az EGO központú koordináta rendszer folyamatos mozgásban van. Ahhoz, hogy a jelenlegi koordináta rendszerben is érvényesek legyenek célobjektumok múltbéli pozíció adatai, bázistranszformációt végez a jármű szoftvere (eltolás és forgatás). A korábbi pozíciók nyomvonalként álnak a rendelkezésekre, és ezt az 5.3. ábrán körökkel jelöltem. A nyomvonalat arra használok fel, hogy a lehető legpontosabban határozzam meg a jármű aktuális oldalirányú pozícióját és sebességét.

A jármű különböző szoftvermoduljai más-más módon tárolják el az objektumok adatait. A calc_lane műveletvégző függvénnyel a jármű különböző szoftvermoduljainak az egyesítését végzem. A célobjektumok, a sávmodell, és a nyomvonal adatokat indexeléssel rendelem egymáshoz. Továbbá rendelkezésekre áll egy további állomány, amiben a mérési szekvencia felvétele után, nem automatizált módon meg vannak jelölve forgalmi események. Ezen jelölések és a korábbi indexelt állományok segítségével címkézem meg a sávváltásokat. A címkézett sávváltásokat további műveletvégző függvényekben és a tesztelés során használok fel.

calc_lat(...)

A harmadik műveletvégző függvény a járművek és a sávváltások közötti laterális viszonyok számítására szolgál.

A 4. fejezetben taglalt ofszet és relatív sebesség értékek nyers formában nem alkalmasak arra, hogy biztonságkritikus rendszert építsek rájuk. A zajszint csökkentéséhez többféle filterező algoritmus is használható, amelyek adatai a függelék 7.1. táblázatában vannak. A késleltetési idő csökkentése, és a számításigény tekintetében egy saját filtert terveztem, ami tendencia alapján számol.



5.3. ábra A nyomvonal-sázelválasztó ofszet

Az aktuális ofszet érték meghatározásához egy N hosszú nyomvonal áll rendelkezésre. Az algoritmus α súllyal veszi figyelembe a mért, $(1-\alpha)$ súllyal a tendencia értékét. Képlettel:

$$\begin{aligned}
 y_1 &= x_1 \\
 y_2 &= x_2 \\
 y_i &= \alpha x_i + (1 - \alpha) \left(y_{i-1} + (y_{i-1} - y_{i-2}) * \frac{dt_i}{dt_{i-1}} \right) \quad (7)
 \end{aligned}$$

Ahol x a nyers ofszet értékek, dt a nyomvonal pontok időkülönbségei, az y pedig a tendencia értéke. Az 1-es indexhez a legrégebbi, az N indexhez pedig a legfrissebb adat tartozik.

calc_dist(...)

A `calc_dist` függvénnyel a szekvenciához tartozó priori értékeket gyűjtöm össze annak érdekében, hogy a modell CPT tábláit meghatározzam.

Két statisztikát állítok össze a célobjektumok ofszet és relatív sebesség adatairól. Egyikben az összes tanítóminta ($P(O, V_{rel})$), másikban csak azok, ahol egy előre

meghatározott időintervallumon belül címkézett sávváltás történik ($P(O, V_{rel} | SV)$).

A (4) egyenlet szerint: adott O és V_{rel} esetén a laterális bizonyíték értéke:

$$P_{lat} = P(SV | O, V_{rel}) = \frac{P(SV) * P(O, V_{rel} | SV)}{P(O, V_{rel})} \quad (8)$$

calc_prob(...)

A `calc_prob` műveletvégzővel a szimulációhoz szükséges valószínűségi értékeket számolom. A paraméterként kapott CPT táblák segítségével határozom meg az alábbi változókat:

- P_{lat}
- P_{layout}
- P_{occup}
- $P_{lanechange}$

5.2 Taszkok

A LaneChange program használatát 3 elkülönített részre bontottam (taszkok). Ebben az alfejezetben tárgyalom a taszkok működését, és a feldolgozás során betöltött szerepüket.

5.2.1 Tanítás

A tanítás taszk a modell statisztikai alapon számolt CPT tábla értékeit határozza meg. A taszk lefutása engedélyezi a szimuláció futtatását.

Az 5.1 ábrán látható „Settings” paneljén az alábbi küszöbértékre határozza meg a valószínűségi táblák értékeit:

- *smooth*: szűrőalgorithmus α paramétere, (7) egyenlet

Egy mezőben több bemeneti paraméter is szerepelhet szóközzel elválasztva. Ebben az esetben összes lehetséges bemeneti kombinációra számítja az eredményeket.

A (8) egyenlet összes értelmezett O és V_{rel} értékeire összeáll a P_{lat} tábla. A modell további CPT tábláit általam definiált táblázatokkal használom.

5.2.2 Szimuláció

A szimuláció taszk a teszt csoport szekvenciáin valós idejű mérést szimulál és folytonosan értékeli ki a sávváltás valószínűségének értékét.

Futása során a tanítás taszk kimenetétül kapott CTP táblákat használja fel. A befejeztével lehetséges futtatni a ROC taszkat.

5.2.3 ROC

A ROC (Receiver Operating Characteristic) görbéket statisztikában használjuk, hogy az osztályozó algoritmusok hatékonyságát vizualizáljuk. Leggyakoribb felhasználási területei az orvosi kutatások, gépi tanulás és adatbányászat. A szakdolgozatomban arra használom fel, hogy megtaláljam azokat a paramétereket és konstansokat, amivel a leghatékonyabban működik az algoritmus.

A ROC taszk futtatásakor a $P_{\text{lanechange}}$ értékekhez egy sávváltást jelző indikátort rendel hozzá. Az 5.1 ábra „Settings” paneljában az alábbi küszöbértékekre határozza meg az indikátort:

- *pLevel*: $P_{\text{lanechange}}$ minimális értéke a sávváltás indikátorhoz
- *pCount*: az indikátorhoz szükséges küszöbérték feletti $P_{\text{lanechange}}$ minimális mennyisége

A tanításhoz használt mérési szekvenciákat kisebb mérési részletekre bontom, és megjelölöm azokat, amelyekben sávváltás történt.

A kiértékeléshez figyelembe veszem a címkézett sávváltásokat és indikátor jelzéseket. Az eredményeket ROC görbéken vizualizálom.

6 Eredmények

A program tesztelésére ROC görbés paraméter beállítást használtam, ami a különböző határértékek esetén vizsgálja a sávváltás felismerésének a sikerességét.

A mérési szekvenciákat rövid szeletekre bontottam és ezekben vizsgáltam a sávváltásra vonatkozó valószínűségi értéket. Indikátor jelzi, hogy elér-e a sávváltásra vonatkozó szintet, a sávváltás előrejelzés sikerességét pedig az indikátor és [0,9 - 4] másodperc tartományon belül bekövetkezett (vagy be nem következett) sávváltás viszonya jelzi.

Az egyes tesztek eredményei:

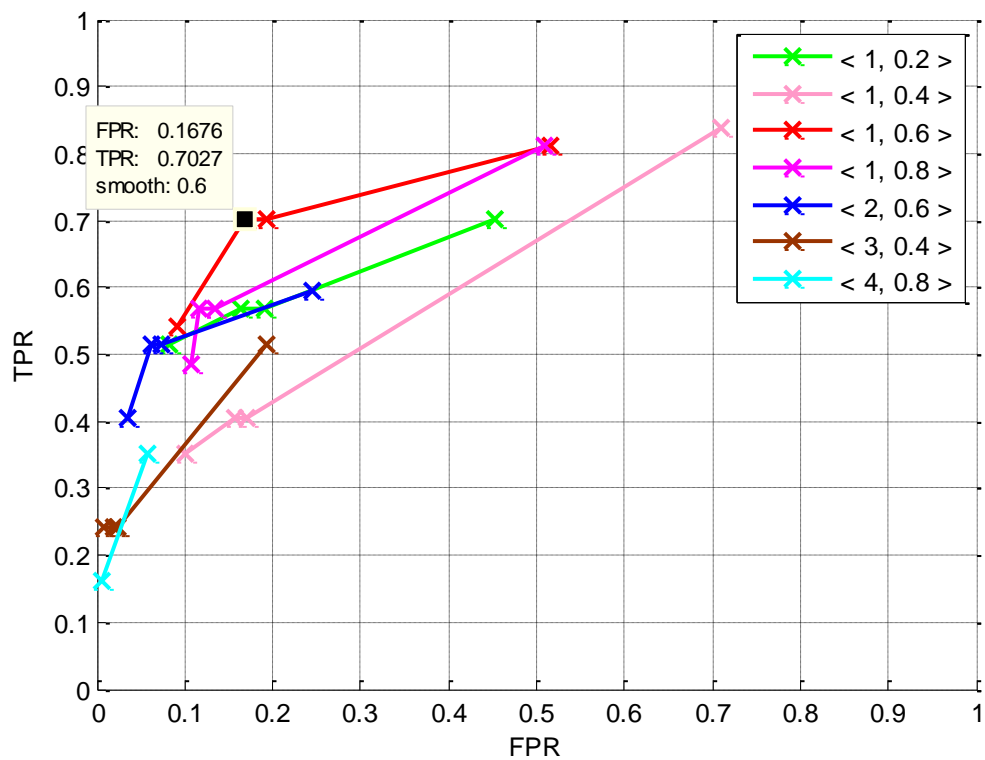
- TP: helyes előrejelzés indikátorral (true positive)
- FP: helytelen előrejelzés indikátorral, téves riasztás (false positive)

ROC arány mutatók:

- TPR: helyes-pozitív teszt és összes sávváltás minták számának hányadosa (sensitivity, recall)
- FPR: téves-pozitív teszt és az összes nem-sávváltás minták számának aránya (fall-out)

A 6.1. ábrán a LaneChange program ROC grafikon látható. A jelmagyarázatban szereplő $\langle x, y \rangle$ név rögzített *pCount* ill. *smooth* értékű görbéket jelöl. Az egyes görbék a *pLevel* változó függvényében fejezi ki a program hatékonyságát.

A tökéletes osztályozó - ami az összes sávváltást felismeri és nem ad téves riasztást - helye a (0,1) pontban van. A legjobb küszöbérték beállítást úgy választottam meg, hogy tökéletes osztályozóhoz képest minimális távolságra legyen a ROC grafikonon.



6.1. ábra ROC görbék

A legjobb küszöbérték beállítás:

- $smooth = 0,6$
- $pCount = 1$
- $pLevel = 0,3$

Teszteredmények:

összes minta	389
sávváltás minta	37
felismert sávváltás(TP)	26
TPR	70,27%
nem-sávváltás minta	352
téves jelzés (FP)	59
FPR	16,76%

További, a hatékonyságot leíró mérőszámok:

- az indikátor jelölések hányad részében történt valóban sávváltás:

$$precision = \frac{\sum TP}{\sum TP + \sum FP} = 0,306$$

- helyes előrejelzések aránya az összes mintához képest:

$$accuracy = \frac{\sum TP + \sum TN}{\sum P + \sum N} = 0,82$$

- *precision* és a *recall* harmonikus közepe:

$$F - measure = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} = 0,4262$$

7 Továbbfejlesztési lehetőségek

Az általam fejlesztett funkció jól mutatja, hogy a sávváltást lehetséges Bayes hálós megoldással modellezni. A fejlesztés során figyelembe vettem, hogy a megoldás beágyazott környezetben is használható legyen. A modellt autóiipari DAS rendszerbe való integrálásával javíthat a biztonsági funkciók hatékonyságán vagy komfortosabbá teheti a vezetést.

A modell további finomításával lehet javítani a felismerés hatékonyságán. Ha több fizikai jellemzőt használtunk bemenő paraméterként, akkor lehetséges a valósághoz sokkal jobban illeszkedő modellt készíteni. Ennek vitathatatlan előnye, hogy pontosabb becslést ad a sávváltásokra, illetve a szomszédos sávok foglaltságára.

Nagyságrendekkel nagyobb mérési szekvenciák használatával érdemes tanítani a valószínűségi mátrixok paramétereit. Ebben az esetben a legtöbb mérésből adódó hiba kiátlagolódik.

Ha a bemeneti paramétereket finomabb felbontású tartományokba osztanánk, az nagyban hozzájárulna, hogy pontosabban ismerjük fel a forgalmi szituációt. Esetleg az is jó megoldás lehet, hogy változó méretű tartományokat alkalmazzunk, vagyis jobban optimalizáljuk a körülményekhez mérten.

A tendencia alapú filterezés helyett jobb szűrési algoritmussal tovább lehetne csökkenteni a jel zajosságát és késleltetését. Kalman filter adaptációja a hosszú távú fejlesztési tervzetben szerepel.

A fent említett továbbfejlesztési lehetőségek mindegyike nagyon számításigényes, a munkám során használt számítógépnek közel sem elegendő teljesítménye hozzá. A továbbfejlesztett funkciót mindenképpen szerver számítógépeken érdemes tanítani.

Rövidítések és kifejezések jegyzéke

ACC	<u>A</u> daptive <u>C</u> ruise <u>C</u> ontrol
BH	<u>B</u> ayes- <u>h</u> áló
TC	<u>T</u> ime <u>C</u> ycle (a legutolsó adatfrissítés óta eltelt idő)
DAS	<u>D</u> riuing <u>A</u> ssistance <u>S</u> ystem (vezetést segítő rendszer)
EGO	a mérést végző jármű
GUI	<u>G</u> raphical <u>U</u> ser <u>I</u> nterface (grafikus kezelőfelület)
Ladar	<u>L</u> aser <u>D</u> etection <u>A</u> nd <u>R</u> anging
Lidar	<u>L</u> ight <u>D</u> etection <u>A</u> nd <u>R</u> anging
mérési szekvencia	a mérés során az autó érzékelőinek jeleit, és a buszon küldött üzeneteket rögzítik, és szekvenciákba darabolják
Radar	<u>R</u> adio <u>D</u> etection <u>A</u> nd <u>R</u> anging
RMM	<u>R</u> ejtett <u>M</u> arkov <u>m</u> odell
sávmodell	a mindenkori sávinformációk tárolására használt struktúra
TLC	<u>T</u> ime to <u>l</u> ine <u>c</u> rossing
TLCH	<u>T</u> ime to <u>l</u> ane <u>c</u> hange
DLC	<u>D</u> istance to <u>l</u> ine <u>c</u> rossing
DLCH	<u>D</u> istance to <u>l</u> ane <u>c</u> hange

Köszönetnyilvánítás

Szeretnék köszönetet mondani Krébesz Tamás Istvánnak és Bodnár Attilának, akik hasznos tanácsaikkal segítettek a szakdolgozatom elkészítésében.

Ezúton köszönetet mondanék a Robert Bosch Kft.-nek a szakdolgozatírási és gyakorlatvégzési lehetőségért.

Ábrajegyzék

2.1. ábra Autóipari koordinátarendszer	9
2.2. ábra Mérés multistatic radar segítségével	10
2.3. ábra Radar felhasználása a járművekben	12
2.4. ábra A beépített fedélzeti kamera elhelyezése	13
2.5. ábra A fedélzeti kamera képe	13
2.6. ábra Az alkalmazott sávmodell	16
3.1. ábra (a) Tanítás, (b) Előrejelzés, (c) Diagnózis, szabályozás	18
3.2. ábra RMM modell ábrázolása	21
3.3. ábra Példa BH modell	24
4.1. ábra Az EGO sáv szélességének hisztogramja és az illesztett normál eloszlás függvénye	27
4.2. ábra Ofszet távolságok ábrája	27
4.3. ábra A bal ofszet hisztogramja	28
4.4. ábra A jobb ofszet hisztogramja	28
4.5. ábra Ofszet hisztogramok keresztkorrelációja	29
4.6. ábra Egy tipikus sávváltás ofszet és relatív sebesség időgrafikonja	31
4.7. ábra Sávváltás térbeli nyomvonalának a modellje	32
4.8. ábra Rendszerterv	34
5.1. ábra A program kezelőfelülete	35
5.2. ábra Fájlfeldolgozás folyamatábrája	36
5.3. ábra A nyomvonal-sávválasztó ofszet	39
6.1. ábra ROC görbék	43

Irodalomjegyzék

- [1] John C. Knight: Safety Critical Systems: Challenges and Directions
- [2] Heckenast Tamás: Szoftver-minőségbiztosítás
<http://www.sze.hu/~heckenas/okt/swmin2.pdf> (elérés ideje: 2015. 05 15.)
- [3] William J. Fleming: Overview of Automotive Sensors
- [4] Chassis Systems Control LRR3: 3rd generation Long-Range Radar Sensor
http://products.bosch-mobility-solutions.com/media/db_application/downloads/pdf/safety_1/en_4/lrr3_datenblatt_de_2009.pdf (elérés ideje: 2015. 05 15.)
- [5] Tomas Henriksson:
Driver Assistance Systems with focus on Automatic Emergency Brake
- [6] Bosch Driver assistance systems hivatalos honlapja
http://www.bosch-mobility-solutions.us/en_us/us/driving_comfort_1/driving_comfort_systems_for_passenger_cars_2/driver_assistance_systems_5/driver_assistance_systems_5.html (elérés ideje: 2015. 05 15.)
- [7] Jan Becker: Driver Assistance and Automated Driving
- [8] David Heckerman: A Tutorial on Learning With Bayesian Networks
- [9] Dobrowiecki Tadeusz, Eredics Péter: Mesterséges Intelligencia MI
- [10] Jelasity Márk: Mesterséges Intelligencia I. előadásjegyzet
- [11] Molnár-Sáska Gábor: Rejtett Markov Modellek statisztikai vizsgálata
- [12] Michael I. Jordan: An introduction to graphical models
- [13] Stuart Russell, Peter Norvig: Mesterséges Intelligencia Modern megközelítésben
- [14] Dietmar Kasper, Galia Weidl, Thao Dang:
Object-Oriented Bayesian Networks for Detection of Lane Change Maneuvers
- [15] J. P. Lewis: Fast Normalized Cross-Correlation
- [16] Judea Pearl: Bayesian networks
- [17] Vetier András: Valószínűségszámítás
- [18] Agnieszka Onisko, Marek J. Druzdzet, Hanna Wasyluk:
A Bayesian Network Model for Diagnosis of Liver Disorders

Függelék

Függelék A

`data_collect(...)`

Bemeneti paraméterek:

a)	logical	Recalc	a futtatandó függvények listája
	char	filename	a mérési szekvencia elérési útja
	struct	CTP	a teszteléshez használt valószínűségi mátrixok tárolója
b)	logical	Recalc	a futtatandó függvények listája
	struct	DAT	a korábbi analízis eredménye
	struct	CTP	a teszteléshez használt valószínűségi mátrixok tárolója

Kimeneti paraméterek

struct	DAT	mérési adatok
--------	-----	---------------

`load_dat(...)`

Bemeneti paraméterek

char	filename	a mérési fájl elérési útja vagy a fájlnev
------	----------	---

Kimeneti paraméterek

struct	AUX	segéd struktúra
--------	-----	-----------------

calc_lane (...)

Bemeneti paraméterek

struct	AUX	a mérési fájl elérési útja vagy a fájlnev
--------	-----	---

Kimeneti paraméterek

struct	LANE	sávmodell
--------	------	-----------

calc_offs (...)

Bemeneti paraméterek

struct	AUX	segéd struktúra
struct	LANE	sávmodell

Kimeneti paraméterek

struct	LAT	laterális értékek
--------	-----	-------------------

calc_dist (...)

Bemeneti paraméterek

struct	AUX	segéd struktúra
struct	LAT	laterális értékek

Kimeneti paraméterek

struct	CPT	feltételes valószínűségi táblák
--------	-----	---------------------------------

`calc_prob(...)`

Bemeneti paraméterek

struct	AUX	segéd struktúra
struct	LAT	laterális értékek
struct	CPT	feltételes valószínűségi táblák

Kimeneti paraméterek

struct	Probs	sávvalással kapcsolatos valószínűségi értékek
--------	-------	---

Függelék B

Mozgó átlag	$y_i = \frac{x_i + x_{i-1} + \dots + x_{i-n}}{n}$	
Exponenciális filter	$y_i = \alpha x_i + (1 - \alpha) * y_{i-1}$	
Butterworth filter	$y_i = \frac{b_1 x_i + b_2 x_{i-1} + \dots + b_n x_{i-n}}{a_1 x_i + a_2 x_{i-1} + \dots + a_n x_{i-n}}$	MATLAB: [b,a] = butter(n,Wn)

7.1. táblázat Szűrőalgoritmusok