



CONCEPTION D'UN SYSTÈME D'AIDE A LA CONDUITE POUR VEHICULE DE TOURISME (ANTICOLLISION)

Arthur Cartel Foahom Gouabou, Leslie Ngwa Ngwa

► To cite this version:

Arthur Cartel Foahom Gouabou, Leslie Ngwa Ngwa. CONCEPTION D'UN SYSTÈME D'AIDE A LA CONDUITE POUR VEHICULE DE TOURISME (ANTICOLLISION). [Rapport de recherche] Ecole Nationale Supérieure Polytechnique de Douala (ex Faculté de Génie Industriel). 2017. hal-02308475

HAL Id: hal-02308475

<https://amu.hal.science/hal-02308475>

Submitted on 8 Oct 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITE DE DOUALA

FACULTE DE GENIE INDUSTRIEL

www.fgi-ud.org



UNIVERSITY OF DOUALA

FACULTY OF INDUSTRIAL ENGINEERING

www.fgi-ud.org



DÉPARTEMENT DE TECHNOLOGIE AUTOMOBILE (TAU)

**Mémoire rédigé et soutenu en vue de l'obtention du Diplôme d'Ingénieur de
Conception en Génie Industriel**

OPTION : MÉCATRONIQUE

***CONCEPTION D'UN SYSTÈME D'AIDE A LA CONDUITE POUR
VEHICULE DE TOURISME (ANTICOLLISION)***

Par :

FOAHOM GOUABOU Arthur Cartel

Licence en Génie Industriel

NGWA Leslie NGWA

Licence en Génie Industriel

Sous l'encadrement de :

**Dr. OFFOLE Florence
Dr. PERABI Steve
M. FOHOUÉ Kennedy**

Devant le jury composé de :

**PRESIDENT : Dr. ESSOLA
RAPPORTEUR 1 : Dr. OFFOLE Florence
RAPPORTEUR 2 : Dr. PERABI Steve
EXAMINATEUR : M. MAH
MEMBRE : M. MBIA**

ANNEE ACADEMIQUE : 2016-2017



DÉPARTEMENT DE TECHNOLOGIE AUTOMOBILE (TAU)

Ce mémoire intitulé :

***CONCEPTION D'UN SYSTÈME D'AIDE A LA CONDUITE POUR
VEHICULE DE TOURISME (ANTICOLLISION)***

Présenté par

FOAHOM GOUABOU Arthur Cartel

NGWA Leslie NGWA

Mémoire présenté en vue de l'obtention du diplôme d'Ingénieur de Conception en Génie
Industriel

a été soutenu le : 20 Décembre 2017

a été dûment accepté par le jury d'examen constitué de :

M. ESSOLA, Dr, président

Mme OFFOLE Florence, Dr, rapporteur et encadreur académique

M. PERABI Steve, Dr, rapporteur et encadreur académique

M. MAH, examinateur

M. MBIA, membre

DÉDICACE

*A mes parents **M. et Mme FOAHOM.***

*A mes sœurs et mon frère **Ariane, Blondelle et Alain.***

Arthur Cartel FOAHOM

*To my grandfather and father Mr **Frederick and Martin NGWA.***

NGWA Leslie NGWA

REMERCIEMENTS

Nous rendons grâce à Dieu tout puissant pour ses grâces, le souffle de vie, la santé, l'intelligence et le courage qu'Il nous a donné et qui nous a accompagné tout le long de ces cinq années d'étude jusqu'à l'aboutissement de ce travail.

Aussi nous tenons à exprimer notre gratitude à ceux et celles qui par leur assistance morale et physique, par leur expérience, leur sympathie, leur encouragement, leur collaboration et leur disponibilité ont contribué de près ou de loin à la réalisation de ce travail. Nous pensons particulièrement à :

- ☯ Au Pr Robert NZENGWA, Doyen de la FGI, pour notre formation exemplaire au sein de son établissement.
- ☯ Au Dr PERABI Steve pour sa disponibilité et son orientation dans la réalisation de ce projet.
- ☯ Au Dr Florence OFFOLE, Responsable du Département de TAU pour ses directives.
- ☯ A Mr FOHOUE Kennedy pour le suivi méticuleux de ce travail.
- ☯ A Mr NYATTE Steve pour ses remarques et suggestions.
- ☯ Aux enseignants Mr KUATE Roméo, Mr MATOP et à tous les autres enseignants de la FGI pour les enseignements dispensés et les conseils prodigués.
- ☯ A nos frères et sœurs, spécialement A Edith, N Carely pour leur amour, conseil et soutien constant.
- ☯ A toutes nos familles respectives aussi larges qu'elles puissent être pour leur soutien et accompagnement, nous pensons tout particulièrement à la famille AMBINGPOH, à la famille NINTA, à la famille NGWA BONWEI, à la famille NADJO, à la famille NZALE, à la famille TAPTCHEM et aux GUAGUEFFA MAYOUEGO.
- ☯ A Mr Peter MAMBO et tout le personnel de l'entreprise SMARTECH Sarl.
- ☯ Au Pr Christian KOUAM et à tout le personnel de l'entreprise AUTOHAUS.
- ☯ A Mr Alain TSOBGNY et Mr Dorian KAMSU et tout le personnel de l'entreprise MAPON Africa.
- ☯ A nos amis pour leur sympathie, nous pensons tout particulièrement au G7, Ousmaela H, Estelle F, Blanche Ayah, Laurel L, Dunnil, Suzanne E, Narcisse B, Cédric B, Marcel A, César K.
- ☯ A tous nos camarades de la 7^{ème} promotion de la Faculté de Génie Industriel.

A tous ceux et celles que nous oublions, qu'ils trouvent ici la marque de notre profonde sympathie.

RÉSUMÉ

Une étude sur le parc automobile du Cameroun montre que les véhicules de tourisme représentent 56,4% de la flotte automobile camerounaise et ces véhicules sont les plus impliqués dans les accidents dans les villes urbaines comme Douala. Le rapport de l'Organisation mondiale de la santé (OMS), sur la sécurité routière de 2013, précise qu'au Cameroun, 1 véhicule tue en moyenne 123,8 fois plus que dans les pays développés. Le même rapport stipule qu'au Cameroun, on enregistre en moyenne 2000 cas d'accidents par an, donc 87% sont dus à des causes humaines (inattention, excès de vitesse). Les piétons et les occupants des véhicules légers (bicyclettes, motocyclettes et tricycles) représentant respectivement 25% et 44% des blessés impliqués dans ces accidents. Les constructeurs automobiles tentent de résoudre ce fléau en développant des systèmes d'aide à la conduite avancée (ADAS) qui inclut également des systèmes d'anticollision (CAS). Ce mémoire a pour objet le développement d'un nouveau CAS basé sur des solutions à base de capteurs qui permettra aux véhicules de « regarder devant eux », de détecter les obstacles (véhicules et piétons) dans leur environnement et d'éviter les collisions soit en freinant, en déviant ou en freinant et déviant en cas de situation accidentogène où le conducteur ne peut réagir à temps. Ce mémoire fait une étude de l'état de l'art dans ce domaine, en passant en revue les méthodes délibératives et actives d'évitement d'obstacle, les approches d'identification par vision (caméra), les technologies impliquant des capteurs actifs (Radar, Lidar, Ultrason) et l'intelligence artificielle (IA).

Mots-clés: conception, véhicule, évitement de collision, intelligence artificielle.

ABSTRACT

A study of Automobile fleet in Cameroon shows that, tourisms vehicles represent over 56.4% of Cameroon's Automobile fleet and these vehicles are the most involved in accidents in urban towns like Douala. The World health Organisation's (WHO) report on road security in 2013 precise that, in Cameroon one vehicle kills an average of 123.8 times more than in developed countries. The same report states that, Cameroon registers an average of 2000 accidents yearly and that 87% are due to human causes (inattentiveness, over speeding). With pedestrians and light vehicles (bicycles, motorcycles and tricycles) representing 25% and 44% of casualties often involved in these accidents. Vehicle manufacturers are addressing these challenges by developing Advance Drivers Assistance Systems (ADAS) which also involve Collision Avoidance systems (CAS). This thesis aims at developing a new CAS based in sensor solutions that will make vehicles « to look ahead » and detect obstacles (vehicles and pedestrians) in their surroundings and avoid collision either by braking, deviating or by braking and deviating at the same time in accidental condition where the driver could not react on time. It investigates the state of the art in this domain, reviewing deliberative and active methods, video-based approaches (stereo vision camera), approaches involving active sensors (Radar) and Artificial intelligence (AI).

Keywords : conception, vehicle, collision avoidance, artificial intelligence.

SOMMAIRE

DÉDICACE.....	I
REMERCIEMENTS	IV
RÉSUMÉ.....	V
ABSTRACT.....	VI
SOMMAIRE	VII
LISTE DES TABLEAUX.....	VIII
LISTE DES FIGURES.....	IX
LISTE DES SIGLES ET ABRÉVIATIONS	XI
LISTE DES ANNEXES.....	XIII
INTRODUCTION GENERALE	1
PARTIE 1: REVUE DE LITTERATURE.....	3
CHAPITRE 1 : LES ADAS ET LES METHODES DE CONCEPTION DES SE.....	4
CHAPITRE 2 : RECONNAISSANCE DE PIETONS ET MÉTHODOLOGIE DE NAVIGATION AUTONOME.....	17
PARTIE 2: MATERIELS ET METHODES.....	45
CHAPITRE 3 : ETUDE DE CONCEPTION.....	46
CHAPITRE 4 : OUTILS ET MATERIELS	70
PARTIE 3 : RESULTATS ET DISCUSSION.....	75
CHAPITRE 5 : SIMULATIONS ET DISCUSSIONS.....	76
CHAPITRE 6 : REALISATION DU PROTOTYPE	86
CONCLUSION GENERALE	93
BIBLIOGRAPHIE	94
ANNEXES	XIV
TABLE DES MATIÈRES	XXXVIII

LISTE DES TABLEAUX

<i>Tableau 1-1 : Différents capteurs disponibles dans un véhicule [8]</i>	<i>6</i>
<i>Tableau 3-1 : Paramètres des distances RC, RG et RD.....</i>	<i>62</i>
<i>Tableau 3-2 : Paramètres des distances d1, d2, d3, d4</i>	<i>62</i>
<i>Tableau 3-3 : Paramètres de la vitesse V</i>	<i>62</i>
<i>Tableau 3-4 : Paramètres de l'angle de déviation Ø.....</i>	<i>63</i>
<i>Tableau 3-5 : Paramètres de la pression de freinage P</i>	<i>63</i>
<i>Tableau 3-6 : Table des règles du contrôleur flou d'évitement d'obstacle</i>	<i>64</i>
<i>Tableau 3-7 : Conclusions des règles proposées pour les valeurs linguistiques de sortie de la variable angle de déviation Ø</i>	<i>65</i>
<i>Tableau 3-8 : Conclusions des règles proposées pour les valeurs linguistiques de sortie de la variable pression de freinage P.</i>	<i>65</i>
<i>Tableau 3-9 : Règles du contrôleur flou d'alignement</i>	<i>67</i>
<i>Tableau 4-1 : Caractéristiques des radars utilisés</i>	<i>71</i>
<i>Tableau 4-2 : Caractéristiques moyennes des caméras stéréovision adoptées</i>	<i>72</i>
<i>Tableau 4-3 : Caractéristiques des microprocesseurs intégrant l'IA pour l'électronique embarquée automobile.....</i>	<i>73</i>
<i>Tableau 6-1 : Caractéristiques des cartes arduino Uno et Mega.....</i>	<i>90</i>
<i>Tableau 6-2 : Caractéristique de la caméra pixy</i>	<i>90</i>

LISTE DES FIGURES

<i>Figure 1.1 : Histogramme présentant l'impact des ADAS sur la réduction du nombre d'accidents de la circulation [11].</i>	8
<i>Figure 1.2 : Architecture simplifiée d'un système embarqué [17].</i>	10
<i>Figure 2.1 : Procédure de détection de piéton, tiré de [45]</i>	24
<i>Figure 2.2 Illustration du principe de fonctionnement des SVMs pour un cas linéaire simple. Les éléments d'une classe (ronds blancs) sont séparés des éléments d'une autre classe (rond noirs) par l'hyperplan séparateur (trait plein) qui maximise la marge (traits en pointillés), tiré [55]</i>	27
<i>Figure 2.3 : Schéma de principe de la logique floue, tiré de [68]</i>	29
<i>Figure 2.4 : Exemples des fonctions d'appartenance (a) Logique classique (b) logique floue, tiré de [63]</i>	30
<i>Figure 2.5 : Structure d'un contrôleur flou [59]</i>	31
<i>Figure 2.6 : Différents modèle d'inférence flous, tiré de [63]</i>	35
<i>Figure 2.7 : Représentation de l'apprentissage automatique [63]</i>	37
<i>Figure 2.8 : Représentation de l'apprentissage supervisé, tiré de [73]</i>	38
<i>Figure 2.9 : Représentation de l'apprentissage non supervisé, tiré de [64]</i>	38
<i>Figure 2.10 : Représentation de l'apprentissage par renforcement, tiré [74]</i>	40
<i>Figure 3.1 : Représentation schématique du système</i>	46
<i>Figure 3.2 : Illustration de l'architecture de commande des systèmes (a) « brake by wire » et (b) « steer by wire »</i>	48
<i>Figure 3.3: Architecture de communication du système (multiplexage)</i>	49
<i>Figure 3.4 : Provenance de l'obstacle heurté par un véhicule [7]</i>	51
<i>Figure 3.5 : Architecture matérielle du BP</i>	52
<i>Figure 3.6 : Présentation des niveaux d'alerte et de leurs actions spécifiques</i>	59
<i>Figure 3.7 : Planificateur de trajectoire floue</i>	60
<i>Figure 3.8 : Illustration des variables linguistiques d'entrée</i>	60
<i>Figure 3.9 : Illustration de l'orientation du véhicule à travers le contrôleur flou de remplacement</i>	66
<i>Figure 3.10 : Illustration de la commande du système de direction par le module de Manœuvres</i>	68
<i>Figure 4.1 : Photo radar automobile BOSCH [93]</i>	71
<i>Figure 4.2 : Photo caméra stéréovision [93]</i>	72
<i>Figure 4.3 : Photo microprocesseur Xavier [93]</i>	73

Figure 5.1 : Simulation de l'identification d'un piéton dans le cas d'un piéton longeant la route...	76
Figure 5.2 : Simulation de l'identification d'un piéton dans le cas d'un piéton traversant la route	77
Figure 5.3 : Vue globale du contrôleur flou d'évitement à partir de Matlab	78
Figure 5.4 (a) (b) (c) : Règles du contrôleur flou d'évitement	79
Figure 5.5 : Surfaces de commande de (a) l'angle de déviation et de (b) la pression pour le couple d'entrée (RG, V)	79
Figure 5.6 : (a) Surface de commande de l'angle de déviation pour le couple d'entrée (V, RC) (b) Surface de commande de la pression pour le couple d'entrée (V, RC)	79
Figure 5.7 : Illustration de l'inférence des valeurs de commande du contrôleur flou d'évitement à partir du logiciel.....	80
Figure 5.8 : Variation pour $V=90\text{km/h}$ de (a) P en fonction de RC ; (b) $\emptyset$ en fonction de RC	81
Figure 5.9 : Variation pour $RC= 3\text{m}$ de (a) $\emptyset$ en fonction de V ; (b) P en fonction de V	81
Figure 5.10 : Illustration de l'apprentissage de l'agent.....	82
Figure 5.11 : Vue globale du contrôleur flou d'alignement	83
Figure 5.12 : Illustration de l'inférence des valeurs de commande du contrôleur flou d'évitement.	83
Figure 5.13 : (a) Surface de commande pour le couple d'entrées ($d1$, $d2$); (b) Surface de commande pour le couple d'entrées ($d3$, $d4$).	84
Figure 5.14 : (a) Surface de commande pour le couple d'entrées ($d1$, $d3$); (b) Surface de commande pour le couple d'entrées ($d2$, $d4$).	84
Figure 5.15 : (a) Variation de $\emptyset$ en fonction de $d1$ pour $d2=0.5\text{m}$ (b) Variation de $\emptyset$ en fonction de $d2$ pour $d1=0.5\text{m}$	85
Figure 6.1 : Schéma bloc du prototype réalisé	86
Figure 6.2 : Interface du logiciel arduino IDE version 1.8.0	87
Figure 6.3 : Interface du logiciel ISIS PROTEUS	88
Figure 6.4 : (a) Photo pont H L-298N; (b) Photo moteurs CC; (c) Photo capteur ultrason; (d) Photo carte arduino uno	89
Figure 6.5 : Photo caméra pixy	91

LISTE DES SIGLES ET ABRÉVIATIONS

ABS	Anti-lock Braking System
ACC	Adaptative Cruise Control
ADAS	Advanced Driver Assistance System
AFNOR	Agence Française de Normalisation
AEBS	Advanced Emergency Braking System
AR	Apprentissage par Renforcement
BAD	Banque Africaine de Développement
CAN	Controller Area Network
CAS	Collision Avoidance System
CNN	Convolutional Neural Network
ECU	Electronic Control Unit
EMA	Evasive Manoeuver Assist
ESP	Electronic Stability Program
FMCW	Frequency Modulated Continuous Wave
GPS	Global Positioning System
HOG	Histograms of Oriented Gradients
IA	Intelligence Artificielle
ILSVRC	ImageNet Large-Scale Visual Recognition Challenge
LDW	Lane Departure Warning
LKA	Line Keeping Assist
PDM	Processus de Décision Markovien
PUFFIN	Pedestrian User-Friendly Intelligent Crossing
PUSSYCATS	Pedestrian Urban Safety System and Comfort at Traffic Signal
RF	Radio-Fréquence

RONE	Région de Non Echappement
SAE	Society of Automotive Engineers
SE	Système Embarqué
SIF	Système d'Inférence Floue
SIG	Système d'Information Géographique
SIT	Système de Transport Intelligent
SRR	Slope Stability Radar
SVM	Support Vector Method
TS	Takagi-Sugeno

LISTE DES ANNEXES

<i>ANNEXE 1 : Les différentes stratégies de choix d'action [63].....</i>	<i>xiv</i>
<i>ANNEXE 2 : Algorithme de Q-learning [63].....</i>	<i>xv</i>
<i>ANNEXE 3 : Diagramme Gantt du projet.....</i>	<i>xv</i>
<i>ANNEXE 4 : Modèle SIMULINK de simulation du contrôleur flou d'évitement.....</i>	<i>xvi</i>
<i>ANNEXE 5 : Modèle SIMULINK de simulation du contrôleur flou d'alignement</i>	<i>xvi</i>
<i>ANNEXE 6 : Circuits électriques du prototype.....</i>	<i>xvi</i>
<i>ANNEXE 7 : Organigramme programme du prototype.....</i>	<i>xviii</i>
<i>ANNEXE 8 : Budget prototype.....</i>	<i>xix</i>
<i>ANNEXE 9 : Cahier des charges du projet.....</i>	<i>xx</i>
<i>ANNEXE 10 : Photos du prototype</i>	<i>xxi</i>

INTRODUCTION GENERALE

Le contexte social africain en général et camerounais en particulier est marqué par un grand nombre d'accident de la route, ce qui constitue un véritable problème de santé et un frein au développement. Pour appuyer cela, le rapport [1] de l'OMS de 2013 sur la sécurité routière précise qu'au Cameroun, 1 véhicule tue en moyenne 123.8 fois plus que dans les pays développés. Le même rapport fait état du fait qu'au Cameroun, on enregistre en moyenne 2000 cas d'accidents chaque année, dont 87% sont dus à des causes humaines (Inattention, excès de vitesse), 69% des décès dus à ces accidents sont ceux de piétons, de cyclistes et des occupants de véhicules légers. Grâce aux nombreuses avancées technologiques et aux nouvelles normes de sécurité qui rendent les véhicules et la conduite de plus en plus sûrs, des solutions existent pour améliorer la sécurité routière. Il s'agit entre autre des messages de prévention routière diffusés par les autorités dans les médias enfin de sensibiliser les usagers. Mais cette solution se heurte à la difficulté voir l'impossibilité de s'assurer du comportement exemplaire des usagers. D'autres solutions possibles, employées cette fois par l'industrie automobile, sont l'automatisation totale de la conduite ou l'installation de systèmes d'aide à la conduite (ADAS) qui assistent le conducteur et le préviennent, voire se substituent à lui lorsqu'il se met dans des situations potentiellement accidentogènes. Mais le cas de l'automatisation totale de la conduite nécessiterais des changements important dans tout l'écosystème des transports et s'avère pour l'heure être une solution pas applicable. Les ADAS, par contre, s'avère être une solution abordable, plus particulièrement, les systèmes d'aide à l'évitement de collision (CAS) sont ceux qui sont de plus en plus abordés par les chercheurs et les constructeurs automobiles. Outre les raisons économiques et commerciales, l'engouement de ces derniers pour les CAS augmente. La vulnérabilité des usagers tels que les piétons et les cyclistes fait en sorte que la sécurité des personnes est devenue un facteur primordial lors de la conception des véhicules, car les normes imposées deviennent de plus en plus strictes. Comme exemple récent, depuis le 24 février 2013, 43 pays ont décidé d'appliquer la réglementation n°13 de la commission économique des Nations Unies pour l'Europe rendant obligatoire la présence de l'AEBS (Système de freinage Automatique) dans les camions et autocars ; l'objectif étant d'étendre cette réglementation aux autres catégories de véhicule [2]. D'un point de vue plus large, l'évitement de collision basé sur la détection d'obstacle fait l'objet d'une attention particulière. Ainsi, la distraction et le non-respect du code de la route à la fois par les usagers de la route comme les conducteurs, ajouté à cela, l'inadéquation des manœuvres exécutées par les conducteurs dans

certaines situations accidentogènes posent le problème du moyen d'évitement efficace des collisions par les véhicules avec une priorité faite aux piétons.

Ce mémoire a pour objet la **conception d'un système d'aide à la conduite pour véhicule de tourisme (anticollision)**. L'objectif est d'obtenir une application embarquée qui détecte et identifie les obstacles dans un environnement sur route, planifie et exécute une trajectoire d'évitement de collision lorsque le risque de collision est jugé important. Plus particulièrement, il sera question pour nous de proposer une méthodologie efficace d'identification de piétons, de réduire le temps de collision et d'utiliser des techniques de l'intelligence artificiel(IA) pour planifier la trajectoire permettant d'éviter les obstacles. La commande des actionneurs du véhicule pour suivre au mieux cette trajectoire est brièvement décrite dans ce mémoire et n'a pas fait l'objet d'étude approfondie.

Pour mener à bien notre projet, la démarche employée, à partir de consultations documentaires et de personnes ressources, sera tout d'abord de faire une revue scientifique des méthodes de conception de système embarqué, des méthodologies de détection et d'identification d'obstacle, ainsi que des stratégies de planification de trajectoire de mobile à l'aide des outils de l'IA. Ensuite, à partir des résultats de nos recherches, nous nous attèlerons à concevoir et à réaliser un système embarqué qui répond au mieux aux objectifs que nous nous sommes fixés à travers le cahier de charges établi. Ce travail est destiné aux étudiants et chercheurs dans le domaine des sciences de l'ingénieur ainsi qu'aux départements recherche et développement de constructeurs automobiles.

Ce mémoire est subdivisé en trois parties réparties comme suit : Dans la première partie qui comporte deux chapitres, nous faisons une étude bibliographique des travaux antérieurs relatifs à notre projet. Le premier chapitre fait une revue de littérature des systèmes d'aide à la conduite existants et présente les méthodes de conception des systèmes embarqués. Le deuxième chapitre expose les principales technologies et méthodologies d'identification de piéton ainsi qu'une description des différentes approches de planification de trajectoire pour des mobiles à base de l'IA. La deuxième partie présente notre contribution dans la conception du système à proprement parlé. Il comporte deux chapitres. Le premier chapitre donne un aperçu global du système à concevoir et une description détaillée du système à partir de l'application des différentes méthodologies retenues. Le deuxième chapitre présente les outils logiciels exploités pour simuler nos choix de conception ainsi que les caractéristiques des principaux composants du système. La dernière partie, divisée en deux chapitres, présente les résultats commentés de nos simulations ainsi que le prototype du système réalisé.

PARTIE 1: REVUE DE LITTERATURE

Un véhicule automobile est un engin doté de roues propulsées par un moteur et destiné au transport terrestre de personnes et des marchandises. Les véhicules automobiles peuvent être classifiés en plusieurs catégories entre autre les véhicules particuliers (ou de tourisme), les véhicules utilitaires et les véhicules légers. Un véhicule de tourisme étant défini comme une voiture destinée au transport des personnes et de leurs bagages pouvant contenir au maximum sept passagers [3]. Le rapport [4] de la BAD (Banque Africaine de Développement) de 2015 stipule que le parc automobile Camerounais était estimé en 2008 à 380 000 véhicules dont 56,40% sont des véhicules de tourisme. Dans la suite de notre mémoire, nous assimilerons le terme véhicule à un véhicule de tourisme. Les statistiques de la sécurité routière en Afrique en générale et au Cameroun en particulier du rapport de l'OMS [4], présentées à l'introduction, relève la nécessité de développer des solutions qui permettent de réduire durablement les accidents routiers. Les piétons, cyclistes et occupants de véhicules légers représentant 69% des cas de décès dus à ces accidents tels qu'illustré par la **figure 1** ci-dessous, démontrent que ces derniers représentent un facteur incontournable dans toute tentative de résolution de ce fléau.

Cette partie fait un état de l'art sur les systèmes d'anticollision qui existe, et présente une étude sur les différentes méthodologies de conception de système embarqué, d'identification de piétons ainsi que des approches de l'IA sur la planification des trajectoires pour l'évitement de collision.

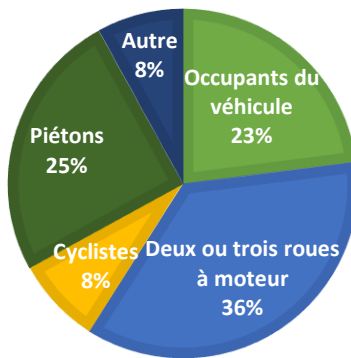


Figure 1 : Diagramme présentant la répartition des décès dus aux accidents de la route par type d'utilisateur en Afrique de l'OMS [1].

CHAPITRE 1: LES ADAS ET LES METHODES DE CONCEPTION DES SE

1.1 Les systèmes d'aide à la conduite pour véhicule

La question de l'amélioration de la sécurité des usagers de la route est un enjeu majeur de la société depuis bien des décennies et est encore au cœur des préoccupations sociétales. Les accidents de la circulation causent chaque année encore de nombreux décès et blessés graves dans le monde. Grâce aux nombreuses avancées technologiques et aux nouvelles normes de sécurité visant à rendre la conduite de plus en plus sûre, les constructeurs automobiles ont développé bon nombre de solutions ayant pour objectif l'amélioration de la sécurité routière. Ces solutions peuvent être classées en deux catégories : celles visant l'automatisation totale de la conduite et celles servant d'aide à la conduite (ADAS : Advanced Driver Assistance System).

1.1.1 Les véhicules autonomes

Un véhicule autonome ou véhicule automatisé, est un véhicule qui se sert de capteurs et d'analyses informatiques pour reconnaître son environnement et naviguer sans intervention humaine [5]. La norme SAE J3016 [6] classe les véhicules autonomes en six niveaux en fonction de leur comportement vis-à-vis du conducteur.

Le cas de la conduite automatique implique par définition que toute erreur liée à un facteur humain serait supprimée. La conduite en serait plus sûre, puisque les règles imposées seraient mieux respectées par les véhicules autonomes. En cas de danger imprévisible, les temps de réaction seraient optimisés, ou du moins mieux maîtrisés par rapport à une conduite humaine où ils peuvent être très variables suivant les individus et suivant leurs états, ce qui assurerait pratiquement une sécurité routière sans faille. Néanmoins, l'avènement des véhicules autonomes impliquerait des changements importants dans tout l'écosystème des transports et prendra certainement du temps à se réaliser. Cependant, les recherches effectuées dans le domaine des véhicules autonomes (appelés parfois « automobile ») ont de nombreuses retombées pour les ADAS [7].

1.1.2 Les systèmes avancés d'aide à la conduite

Les ADAS sont des systèmes conçus pour être intégrés dans les véhicules afin de renforcer la quantité et la qualité des informations transmises au conducteur et de l'assister voire se substituer à lui lorsqu'il se met dans des situations potentiellement accidentogènes. Les ADAS interviennent dans la conduite au niveau tactique comme par exemple sur l'accélérateur, les freins ou sur la direction. Parmi les systèmes embarqués pour l'aide à la navigation d'un véhicule dans un environnement urbain et périurbain, deux fonctions sont envisagées : la première porte sur l'exploitation des informations fournies par des capteurs installés sur le véhicule afin de percevoir l'environnement autour du véhicule. Le **tableau 1.1** ci-dessous présente différents capteurs qui sont actuellement disponibles et nombreux dans les véhicules. D'autres sources d'information sont exploitables par les ADAS comme par exemple des informations venant d'autres véhicules, selon le système de communication V2V (Vehicle-to-Vehicle) ou venant d'infrastructure selon le système de communication V2I (Vehicle-to-Infrastructure) comme des téléphones mobiles ou des réseaux de données WI-FI. La deuxième a pour objectif de proposer un ensemble de services embarqués dotant le véhicule de nouvelles fonctionnalités comme par exemple le platooning¹, la détection d'obstacles, la gestion des distances, la planification des trajectoires et des trajets [8].

¹ Le platooning est un regroupement de véhicule par pelotons, les ADAS utilisent cette fonctionnalité principalement dans des situations de convoi routier ou d'embouteillage.

Tableau 1-1 : Différents capteurs disponibles dans un véhicule [8]

Type de capteurs	Rôle
Capteurs d'imagerie	Modules de caméras, les caméras gamme 3D, détecteurs lasers, détecteurs infrarouges, les capteurs de l'état de la route.
Capteurs de distance	Capteurs infrarouges, radar, capteur de collision par l'avant, capteur de collision par l'arrière, capteur de collision latéral et capteurs anticollision.
Les cartes numériques	Système de positionnement Global (cartes GPS), Système d'Information Géographique (SIG)
Les dispositifs de communication	Communication sans fil et diffusion météorologique locale.
Tachymètres	Indicateurs de vitesse, thermomètres, horloges, capteur vitesse de roue
Capteurs mécaniques	Capteur mécanique de l'état du moteur, capteur de pression des pneus, capteur de l'état de la pneumatique
Autres capteurs	Pollution, détection de pluie...

Beaucoup d'ADAS existent sur le marché, ils sont pour la plupart des systèmes de sécurité actifs. Un système de sécurité actif peut être défini comme étant un ensemble d'éléments liés au véhicule, à l'homme et à l'environnement, qui par leur présence ou leur fonctionnement peuvent minimiser la gravité de l'accident ou éviter qu'il ne se produise. Ils entrent en action avant l'accident (contrairement aux systèmes de sécurité passive qui entrent en action pendant l'accident) [9]. On retrouve par exemple, comme ADAS, l'ESP (Electronic Stability Program) qui aide au maintien de la contrôlabilité du véhicule dans les virages, l'ACC (Adaptive Cruise Control ou DISTRONIC chez Mercedes) qui aide à réguler la vitesse du véhicule afin de maintenir une distance de sécurité raisonnable par rapport au véhicule précédent, ou encore le LKA (Line Keeping Assist, ou LDW, Lane Departure Warning) qui prévient le franchissement involontaire d'une ligne de démarcation de voie [7]. Nous avons également l'ATTENTION ASSIST, qui est un système novateur de détection de somnolence qui alerte le conducteur dès que des signes caractéristiques de fatigue ou d'inattention sont détectés [10].

Les exemples d'ADAS cités ci-dessus ayant surtout pour portée la contrôlabilité du véhicule par le conducteur, les ADAS appliqués aux fonctions d'évitement de collision (CAS : Collision Avoidance System) sont ceux qui sont de plus en plus abordés par les chercheurs et les constructeurs automobiles et également les plus prometteurs en matière d'impact sur la réduction du nombre d'accident comme le démontre les pourcentages présentés dans l'histogramme de la **figure 1.1** ci-dessous et tiré de LOUWERSE [11] . Dans ce cas également, des systèmes ont déjà vu le jour ou sont encore en phase de développement.

Parmi les CAS ayant déjà été développés au moment de la rédaction de ce mémoire, nous pouvons citer l'AEBS (Advanced Emergency Braking System ou Foward Collision Assist) qui permet de freiner automatiquement lorsqu'une collision est imminente. VOLVO a développé sur la Volvo S60 un CAS qui allie les performances d'une caméra et d'un radar pour détecter et reconnaître les piétons afin d'effectuer un arrêt d'urgence si nécessaire [12]. Le radar détecte la présence et la distance des obstacles pendant que la caméra en identifie la nature. Une étude réalisée par ABI Research publiée en 2015 montre que les constructeurs automobiles Mercedes-Benz, Volvo et BMW dominent le marché des systèmes CAS pour la protection des piétons. À partir de 2013, les voitures BMW ont été équipées d'un kit d'assistance au conducteur pour l'avertissement des piétons, basé sur la vision nocturne infrarouge et la vision par camera monoculaire. Récemment, Mercedes a combiné des caméras de vision stéréo avec des radars à longue, moyenne et courte portée pour surveiller la zone

en face du véhicule. En 2016, le groupe Volkswagen a proposé un capteur radar avancé (standard pour son modèle Tiguan) capable de détecter les objets et les piétons, à une distance allant jusqu'à 170 mètres [13].

Pour ce qui est des CAS qui sont encore en développement au moment de la rédaction de ce mémoire, et dont la plupart sont prévus être mis sur le marché en 2018, nous avons le système EMA (Evasive Manoeuver Assist) de la marque VOLVO qui consiste à éviter tout obstacle en cas de situation accidentogène en agissant sur le système de direction et de freinage [14]. La société Nissan développe un système qui détecte l'environnement de la voiture, y compris la route, les autres véhicules et les piétons. Plus poussés que l'EMA, les CAS du groupe Daimler, de la société BOSCH avec son système PEDESTRIAN AVOIDANCE SYSTEM [15] et de Toyota avec le PEDESTRIAN AVOIDANCE STEER ASSIST SYSTEM [16] intégreront la reconnaissance de piétons dans leurs systèmes de direction automatisée en cas d'urgence.

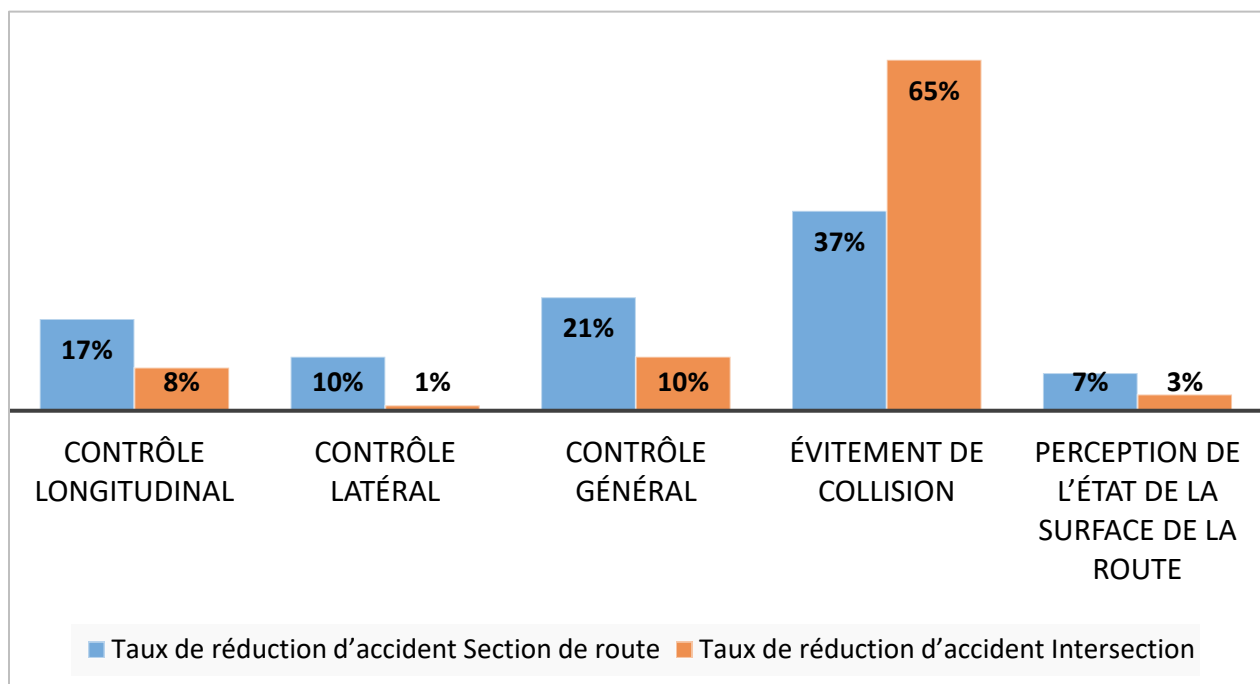


Figure 1.1 : Histogramme présentant l'impact des ADAS sur la réduction du nombre d'accidents de la circulation [11].

L'intégration massive des ADAS dans les véhicules est une solution, pour l'instant, plus abordable que l'avènement des véhicules autonomes. Nous avons vu dans cette section que les ADAS pouvaient être regroupés en deux catégories : ceux destinés à la sécurité du véhicule et de ses occupants à l'instar de l'ESP, l'ACC et le LKA; et ceux destinés à la sécurité des usagers de la route comme l'AEBS, l'EMA et le PEDESTRIAN AVOIDANCE STEER ASSIST SYSTEM. Pour ces derniers, qui sont regroupés comme étant des CAS, nous avons pu observer qu'hormis le fait que ces systèmes soient encore en phase de développement, seuls ceux développés par Toyota et Bosch prévoient combiner l'automatisation du système de direction et de freinage pour l'évitement de piéton. La combinaison de l'automatisation de ces deux systèmes pour éviter la collision avec les piétons, assure une plus grande efficacité. Néanmoins, ces systèmes ADAS présentent encore des difficultés dans la différenciation entre les êtres humains et les objets voisins ainsi que dans les manœuvres d'évitement par freinage et déviation, en particulier dans un environnement urbain bondé où ils ne sont pas capables de détecter tous les piétons partiellement cachés, de plus, ils ne fonctionnent pas efficacement dans des conditions météorologiques extrêmes. En outre, il est difficile de trouver un système ADAS capable d'assurer une fonctionnalité stable et efficace en temps réel [13].

L'objectif principal de notre travail est de concevoir un système d'anticollision agissant, à la fois et de manière automatique sur le système de direction et le système de freinage, lorsque le risque de collision sera jugé important. Plus particulièrement, la technologie et les méthodes employées doivent pouvoir identifier le piéton et les occupants des véhicules légers. De même l'évitement des obstacles doit être efficace.

1.2 Conception des systèmes embarqués

Un Système Embarqué (SE) peut être défini comme étant tout système conçu pour résoudre un problème ou une tâche spécifique mais qui n'est pas un ordinateur d'usage général [17]. Il utilise généralement un microprocesseur combiné avec d'autre matériel et logiciel pour résoudre un problème de calcul spécifique. La **figure 1.2** ci-dessous illustre l'architecture simplifiée d'un système embarqué en ressortant les différents blocs le constituant. La conception de tout système en général et des systèmes embarqués en particulier nécessite donc l'utilisation de méthodes et d'outils de conception guidant le concepteur dans sa démarche de création d'un nouveau produit.

Dans cette section, nous présenterons un bref aperçu de l'historique des SE. Nous en viendrons à leur application dans le domaine de l'automobile. Nous classifierons les types de conceptions et

exposerons ensuite quelques méthodes et outils de conception applicables dans le cas de la conception d'un SE. Nous concluons cette section par l'identification de la méthode de conception adaptée à notre projet.

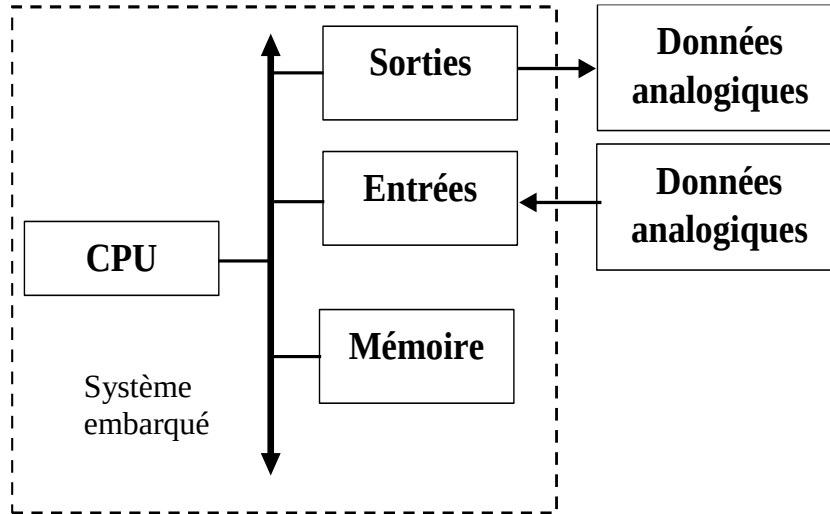


Figure 1.2 : Architecture simplifiée d'un système embarqué [17].

1.2.1 Historique des SE et présentation des types de conception

Nous pouvons débiter l'historique des SE dans la fin des années 1940 où le processeur Whirlwind du MIT est conçu pour des applications temps réel, ceci à l'origine pour contrôler un simulateur de vol. Le premier microprocesseur apparaîtra en 1970 dénommé l'Intel 4004, un peu plus tard, la calculatrice HP-35 utilise plusieurs circuits pour implémenter un microprocesseur en 1972. Les applications dans l'automobile ont débuté depuis les années 1970 avec des systèmes de contrôle du moteur avec microprocesseurs ayant pour objectifs : le contrôle du mélange fuel/air, la gestion du moteur, la baisse des émissions polluantes, la consommation optimisée [17].

POMIAN [18] définit une méthode de conception comme étant « Une méthode proposant une démarche cherchant à définir l'ordonnancement de toutes les activités à mener à bien pour atteindre un objectif final. Une technique est un procédé permettant de réaliser une ou plusieurs de ces activités. Une méthode peut donc demander l'emploi successif de plusieurs techniques. ». Les techniques (ou outils) désignent l'ensemble des moyens utilisés lors des différentes activités. Ce peut être des logiciels, des normes, des questionnaires, du prototypage, des outils d'analyse. Ainsi, l'activité de conception comporte deux phases : la définition puis le développement d'un produit. Le produit (ou

service, procédé...) doit répondre à un ensemble de spécifications (techniques et fonctionnelles) décrites dans un cahier des charges. La rédaction d'un cahier des charges est une tâche complexe ; elle consiste à recueillir les besoins des futurs utilisateurs du produit (clients) et la façon donc le produit peut y répondre [19].

CHANDRASEKARAN [20] a identifié trois types de conception, différenciés par la connaissance initiale que l'on a de l'objectif, du domaine et de la démarche : les conceptions routinière, innovante ou créative. L'objectif étant composé de fonctionnalités attendues du produit (système) et du domaine des solutions technologiques pouvant permettre de réaliser le produit. La démarche ici, correspond aux différentes compétences et savoir-faire de l'entreprise permettant d'exploiter la connaissance du domaine pour répondre à l'objectif.

a- La conception créative

En conception créative, le concepteur ne dispose pas de connaissances sur le produit à concevoir. Le produit n'existe pas et n'a pas d'équivalent sur le marché, seul l'objectif est connu. Les technologies à utiliser pour que le produit puisse réaliser les fonctions pour lesquelles il est conçu sont à identifier et le concepteur doit s'approprier ces technologies. Ce type de conception est le processus à mettre en œuvre pour un tout nouveau produit. Pour ce type de conception, les concepteurs procèdent généralement par essais successifs.

b- La conception innovante

En conception innovante, nous disposons de plus de connaissances sur le futur produit : son objectif ainsi que les technologies qui vont permettre de réaliser le produit sont connus, mais les stratégies de conception reste à définir. Ce type de conception est le processus à mettre en œuvre pour la mise sur le marché d'un produit correspondant à un besoin exprimé par les clients, mais non encore satisfait. Pour ce type de conception, les entreprises ont généralement peu d'expérience sur le produit à concevoir.

c- Conception routinière

En conception routinière, l'objectif du produit est parfaitement connu, ainsi que les technologies permettant de le réaliser. Nous parlons de conception routinière lorsque les concepteurs ont une certaine habitude et des savoir-faire sur la démarche de conception d'une même famille de produits. Ce type de conception concerne généralement un produit de complément de gamme, de remplacement ou d'améliorations incrémentales.

Notre projet de fin d'étude s'inscrit dans la catégorie d'une conception innovante. Ceci étant, il sera question pour nous d'effectuer le choix des méthodes et des outils que nous utiliserons pour la conception de notre système.

1.2.2 Présentation des méthodes et outils de conception de système

Bon nombre de méthodes et outils existent pour la conception de système, dans le cadre de notre projet de fin d'étude, nous avons restreint notre étude à celles qui nous semblaient plus pertinentes à adapter au système que nous voulons concevoir, il s'agit entre autre, de l'analyse fonctionnelle, du QFD (Quality Function Deployment), de l'analyse de la valeur et de la méthode TRIZ.

1.2.2.1 Analyse fonctionnelle [21] [22]

Créée à la fin des années 40 par la société General Electric, l'analyse fonctionnelle a vite été utilisée par les industriels ayant à faire face à des défis économiques et stratégiques importants.

D'après la norme AFNOR NF X 50-151 de 1991, l'analyse fonctionnelle est une démarche qui amène à s'interroger sur les réelles attentes des utilisateurs par rapport à un produit ou à un service. L'analyse fonctionnelle repose sur une méthode dont les étapes sont les suivantes :

a- Énoncer le besoin fondamental

Il est question dans cette étape d'exprimer le besoin et de vérifier sa validité. L'outil utilisé ici pour le premier point est la « Bête à cornes », qui est un outil de réflexion emprunté au cabinet Application aux Techniques d'Entreprise (APTE). En ce qui concerne le deuxième point, la validation du besoin se fait en répondant aux questions suivantes, le but étant de cerner l'espérance de vie commerciale du produit :

- Pourquoi ce besoin existe-t-il ?
- Qu'est ce qui pourrait le faire évoluer ?
- Qu'est ce qui pourrait le faire disparaître ?
- Qu'est ce qui pourrait le faire disparaître ?

b- Rechercher les fonctions

La recherche des fonctions s'effectue en étudiant les relations du futur produit avec son environnement. Elle s'effectue selon une méthodologie axée sur le recensement exhaustif des fonctions. Chaque fonction doit être exprimée exclusivement en terme de finalité et être formulée par un verbe à l'infinitif suivi d'un ou plusieurs compléments. Les fonctions se déclinent en deux

catégories, les fonctions de services, elles même comprenant des fonctions principales et des fonctions contraintes, et les fonctions techniques. Les fonctions de services permettent de définir chacun des services attendus par le client en termes de finalités tandis que les fonctions techniques le font en termes de solutions. Les outils permettant d'effectuer la recherche des fonctions sont la rosace des milieux extérieurs encore appelé diagramme pieuvre développé par APTE, le FAST (Function Analysis System Technic), et le SADT (Structured Analysis and Design Techniques).

c- Caractériser et quantifier les fonctions

Une fois les fonctions identifiées, il faut définir les critères qui nous permettront d'effectuer le choix d'une solution technique. Cela consiste à énoncer pour chaque fonction de service les critères d'appréciation, les niveaux de chaque critère et la flexibilité de chaque niveau. Les critères sont retenus pour apprécier la manière donc une fonction est remplie ou une contrainte respectée. Une fois définis, les critères de valeur devront, chaque fois que possible, être assortis d'un niveau. Ce dernier est exprimé dans une unité correspondant aux propriétés et caractéristiques préalablement identifiées et peut être celui recherché en tant qu'objectif ou celui atteint par une solution proposées. Enfin, le niveau pourra être modulé par des indications de flexibilité si l'on souhaite rechercher une certaine optimisation du rapport qualité/prix. Ces indicateurs sont pondérés par quatre chiffres.

d- Hiérarchiser les fonctions

Dernière étape de l'analyse fonctionnelle, elle a pour objectif d'indiquer aux futurs prestataires ou aux concepteurs, les services essentiels sur lesquels il faudra concentrer leur savoir-faire ; pour cela la hiérarchisation des fonctions peut se faire soit en associant directement un coefficient à chaque fonction, soit en comparant chaque fonction à toutes les autres en jugeant si elle est « plus importante » ou « moins importante ». La méthode dite du « Tri-croisé » est utilisée pour hiérarchiser les fonctions.

Les résultats de l'analyse fonctionnelle sont formalisés dans le Cahier des Charges Fonctionnel du besoin (CdCF).

1.2.2.2 Quality Function Deployment (QFD) [21]

Dans la perspective de s'améliorer, les industries japonaises ont cherché dans les années 70 à optimiser et à rationaliser la conception de leurs produits et de leurs procédés. Les premiers outils méthodologiques utilisés furent les diagrammes de causes à effets pour identifier la demande des clients. En 1972, le chantier de construction navale de Kobe (Mitsubishi) développa la première matrice « besoins clients/caractéristiques qualité ». Le QFD était né. AKAO [23], un de ses

fondateurs, le définit de la manière suivante : « *Le QFD propose des méthodes spécifiques pour garantir la qualité à chaque étape du procédé de développement des produits, en commençant par la conception. En d'autres termes, il s'agit d'une méthode pour introduire la qualité dès le stade de la conception, afin de satisfaire le client puis de traduire les exigences des clients en objectifs de conception et en points clés qui seront nécessaires pour assurer la qualité en phase de production.* ».

Le QFD avait été développé au départ pour réduire les coûts de développement de nouveaux produits. Les entreprises utilisant le QFD ont connu une véritable réduction du temps de développement de leurs produits. A titre d'exemple, les constructeurs automobiles américains dans la fin des années 80 et au début des années 90 nécessitaient en moyenne 5 ans pour mettre un produit sur le marché, en partant de la conception jusqu'au showroom. Parallèlement, Honda mettait deux ans et demi pour mettre un nouveau produit sur le marché et Toyota 3 ans. Ces deux dernières entreprises justifiaient cette conséquente réduction du temps de développement par l'utilisation du QFD.

Le QFD s'appuie sur un ensemble de matrices pour organiser l'information et faciliter la réflexion collective. La première matrice, outil central de la méthodologie du QFD, est la matrice de la qualité, connue également sous le nom de maison de la qualité.

1.2.2.3 TRIZ [24]

La TRIZ est une méthode d'innovation donc l'acronyme russe signifie « Théorie de résolution des problèmes inventifs ». Elle a été proposée par le soviétique Genrich Altshuller en 1946 [25]. G.Altshuller est un ingénieur dans la marine russe qui avait pour mission d'analyser les brevets inventifs internationaux (plus de 40000 brevets inventifs) et d'aider les inventeurs dans les démarches de dépôt de brevets. Ses recherches ont pour objectifs de déterminer « la mécanique de création », et trouver une approche scientifique à ce qui se passe dans la tête des inventeurs lors de la résolution d'un problème. Elle a donc pour but d'orienter le concepteur par un processus systématique permettant de guider la résolution du problème à l'aide d'outils. TRIZ considère que les problèmes rencontrés durant la conception d'un nouveau produit présentent des analogies avec d'autres et que des solutions analogues doivent pouvoir s'appliquer.

Le processus de résolution de problème inventif par la méthode TRIZ se déroule en 3 grandes étapes :

- Analyse de la situation initiale, la reformulation du problème industriel sous forme de contradiction et l'utilisation des outils d'aide au déblocage de l'inertie psychologique afin d'établir une modélisation du problème non pas industrielle mais abstraite.

- La génération des modèles de solution en utilisant les outils de résolution de problèmes inventifs qui guident le concepteur vers les solutions/concepts génériques établies par Altshuller.
- L'interprétation des modèles de solutions abstraites générés afin d'en extraire un concept de solutions applicables au problème industriel.

Les outils de la TRIZ sont divisés en deux catégories suivant leur utilisation : les outils de déblocage psychologique (Les hommes miniatures, la méthode des neuf écrans,...) utilisés lors de la modélisation du problème et les outils de résolution de problèmes (Matrice de résolution des contradictions techniques, l'algorithme ARIZ) utilisés pour la génération des modèles de solutions.

Depuis son introduction par les publication et travaux d'Altshuller, les entreprises longtemps axées sur la productivité, la qualité et les délais, s'orientent dorénavant vers l'innovation pour conserver leurs avantages concurrentiels. Par le passé, la recherche d'innovation en entreprise se faisait majoritairement de manière incrémentale, consistant principalement à l'amélioration des systèmes existants. Mais l'application de la méthode TRIZ a permis la mise en place d'une logique d'amélioration radicale créant la rupture avec les technologies utilisées antérieurement. Son utilisation est de plus en plus présente dans bon nombre d'industrie à l'instar du géant de l'électronique Samsung qui utilise la TRIZ comme méthode principale d'innovation depuis 2007 ce qui lui a permis de réaliser une forte croissance de son chiffre d'affaire (2006 : 162,9 m\$; 2011 : 220,1m\$). La méthode TRIZ bien que très efficace dans la résolution de problème innovant, son utilisation reste cependant limitée. Les limites de son déploiement viennent de plusieurs aspects [24] [26] :

- L'apprentissage de la méthode TRIZ demande un temps d'assimilation long et nécessite une expertise certaine qu'il faut acquérir par la formation et/ou le suivi d'un expert. En effet, la démarche TRIZ demande un certain recul sur le problème et les cas d'étude sont souvent présentés de manières très succinctes demandant une certaine maîtrise des outils
- La résistance psychologique aux principes de la méthode TRIZ. en effet, des notions de la méthode apparaissent comme floues et éloignées du problème industriel. Une absence de lien entre la TRIZ et les procédés maîtrisés par l'entreprise et les moyens de production donc elle dispose limite fortement le nombre de concepts de solution pouvant être mis en pratique

- Elle a du mal à s'imposer en tant que méthode et est d'avantage utilisée sous forme de « boîte à outils » pouvant aider le concepteur du fait de la nécessité pour son application d'y allouer de grande ressource

Concevoir un SE nécessite pour le concepteur de prendre en compte un certain nombre de contrainte entre autres les performances de son futur système, ses fonctionnalités, son coût de fabrication, etc. Nous avons vue dans cette section qu'une procédure s'avérait donc nécessaire pour assurer au concepteur qu'il fera le bon choix des paramètres importants. Pour cela, 4 méthodes de conception qui se rapprochaient de la typologie de notre projet à l'instar de l'analyse fonctionnelle, de l'analyse de la valeur, du QFD et de la TRIZ ont été étudiés. L'étude de ces différentes méthodes, leurs origines, leurs méthodologies et les différents avantages et limites de chacune de ses méthodes nous a permis de mieux les appréhender et de retenir l'analyse fonctionnelle comme méthode de conception que nous allons employer pour la conception de notre système.

CHAPITRE 2 : RECONNAISSANCE DE PIETONS ET MÉTHODOLOGIE DE NAVIGATION AUTONOME

2.1 Les méthodes d'identification de piétons

La conception d'un système d'anticollision pour véhicule implique, pour le système, de le doter de capacités fonctionnelles lui permettant d'analyser l'environnement, pour estimer les caractéristiques des objets qu'il contient et donc le véhicule peut entrer en collision. La sécurité des piétons étant un problème prioritaire dans l'environnement de trafic urbain, de plus, le rapport de l'OMS [1] faisant état du fait que les piétons représentent 25% des décès dus aux accidents de la route en Afrique, par conséquent, un accent particulier doit être mis sur la détection et l'identification de ces usagers vulnérables de la route. Dans cette section, nous ferons une revue de littérature des différentes technologies des capteurs utilisés pour la détection des piétons, ensuite nous présenterons les principales méthodes d'identification de piéton par vision, avant d'exposer les méthodes que nous utiliserons dans le cadre de notre projet.

2.1.1 Les technologies des capteurs [27]

Les piétons sont des utilisateurs importants et vulnérables de la route, en particulier dans les milieux urbains. Les accidents de la circulation impliquant des piétons entraînent généralement des blessures graves ou des décès. Du point de vue de la technologie, différentes technologies de détection telles que des capteurs piézoélectriques, capteurs ultrasoniques, radar à micro-ondes, scanner laser et la vision par ordinateurs peuvent être utilisées pour la détection des piétons. Une revue de ces différents capteurs est résumée ci-dessous où ils sont principalement classés selon les types de technologie qu'ils utilisent. Il existe déjà un travail important qui a été consacré à la détection automatique des piétons. Nous nous sommes basés sur cette revue afin d'identifier les méthodes et technologies les plus appropriées à notre projet.

2.1.1.1 Capteur piézoélectrique

La piézoélectricité, ou l'électricité à pression, est la propriété de certains matériaux qui produisent une modification des propriétés électriques avec une pression mécanique. Pour l'application à la détection des piétons, les câbles piézoélectriques à matériau piézoélectrique sont habituellement fabriqués sous forme de tapis. Lorsqu'une personne passe sur un tel tapis, un signal

électrique est généré jusqu'à ce que la personne quitte le tapis. Dans [28] et [29], les détecteurs piézoélectriques sont utilisés pour détecter la présence d'un piéton en attente lors d'une traversée contrôlée par route pour PUFFIN (Pedestrian User-Friendly Intelligent Crossing) qui signifie Traversée intelligente facile à utiliser pour les piétons et PUSSYCATS (Pedestrian Urban Safety System and Comfort at Traffic Signal) pour Système urbain de sécurité et confort pour piétons dans les signaux de circulation. Un détecteur piézoélectrique est un capteur simple et fiable pour la détection des piétons. Il ne nécessite pas de traitement de signal complexe. Cependant, il nécessite un contact physique entre un piéton et le tapis du capteur. Par conséquent, un détecteur piézoélectrique est habituellement utilisé à un passage piéton et doit être utilisé conjointement avec d'autres capteurs tels que des radars ou des capteurs infrarouges. Le capteur piézoélectrique est idéal lorsque le contact physique direct entre le piéton et le capteur est assuré, comme à un endroit où les piétons sont canalisés dans un passage à niveau.

2.1.1.2 Capteur ultrason

Les détecteurs ou capteurs à ultrasons transmettent des ondes ultrasonores (haute fréquence). Lorsque les véhicules ou les piétons passent, une partie de l'onde sonore transmise est réfléchi vers le récepteur. Sur la base de l'analyse du signal réfléchi, les objets peuvent être détectés avec leur distance et leur vitesse. Il existe deux types basiques de capteurs à ultrasons en fonction de la forme d'onde ultrasonore [30]. Les capteurs à ultrasons à impulsions mesurent la distance ou la présence d'objets en envoyant une onde à ultrasons pulsés, puis en mesurant le temps de vol de l'écho sonore réfléchi. Les capteurs à ultrasons à ondes continues produisent des ondes ultrasoniques continues de certaines fréquences et utilisent le principe Doppler pour détecter un objet mobile et sa vitesse. Les détecteurs à ultrasons peuvent détecter des objets jusqu'à neuf mètres. Une discussion sur les limites des capteurs à ultrasons a été développée par BECKWITH [31] et SONG [32]. Tout d'abord, afin de minimiser l'échappement de l'énergie ultrasonique perdue à partir d'une cible, les configurations d'installation préférées sont directement orientées vers le bas au-dessus de la zone de visée ou vers une position de vision latérale horizontale. Deuxièmement, les vêtements ont des effets sur la détection des piétons. Les vêtements en fibres naturelles sont généralement plus absorbants pour les ondes sonores que les vêtements en fibre synthétique. Par conséquent, les piétons qui portent des fibres synthétiques (par exemple, du nylon) sont facilement détectés par rapport aux piétons portant des fibres naturelles (par exemple, du coton). Enfin, les conditions météorologiques (la température, la pression, l'humidité, ...) affectent la performance des capteurs à ultrasons. Ceci parce que la vitesse

du son varie en fonction de la température et de la pression du milieu. Dans [31], les capteurs à ultrasons placés à des passages non signalés pour la détection des piétons ont reçu plus de faux appels pendant les pluies. Dans [33], l'effet du vent de différentes directions a été étudié pour un système de capteurs à ultrason.

2.1.1.3 Radar

Le radar à micro-ondes fonctionne de manière similaire aux capteurs à ultrasons. Au lieu des ondes sonores, les ondes électromagnétiques sont transmises depuis l'antenne radar à micro-ondes. Sur la base de l'analyse des signaux arrières rebondis, les objets peuvent être détectés avec leur distance et leur vitesse. Il existe trois principaux types de radars hyperfréquences, en fonction de leur forme d'onde électromagnétique transmise (Doppler, FMCW : Frequency modulated continuous wave et UWB : Ultra Wide Band). Le radar Doppler transmet une onde continue de fréquence constante. De telles ondes électromagnétiques, lorsqu'elles sont réfléchies par un objet en mouvement, auront un décalage de fréquence. En analysant le décalage de fréquence, la vitesse de l'objet peut être calculée. Le radar Doppler seul ne peut détecter que l'objet mobile avec une vitesse supérieure à un certain seuil. Le deuxième type de radar à micro-ondes transmet des signaux modulés en fréquence ou modulés en phase. La distance à la cible est déterminée par la temporisation du signal de retour [30] [32]. Le radar ultra large bande (UWB) est une technologie nouvelle, émergente, qui présente un grand potentiel dans les applications STI (Système de Transport Intelligent) [34]. Le concept de base est de transmettre et de recevoir un éclat de fréquence radio (RF) à durée extrêmement courte, généralement quelques dizaines de picosecondes à quelques nanosecondes. Afin d'offrir plus de liberté et d'information, différentes technologies radar sont souvent intégrées ensemble. Par exemple, la technologie de radar à courte portée (SRR : Slope Stability Radar) de 24 GHz proposée par l'industrie automobile est une combinaison de deux fonctions [35]:

- 1) un mode radar classique Doppler fournissant des informations de vitesse sur un objet qui s'approche;
- 2) un mode radar UWB fournissant des informations de position sur des objets d'une résolution d'environ 10-15 cm.

Les capteurs radar peuvent fournir une distance et une vitesse précises de l'objet sans traitement complexe du signal (par rapport à la vision par ordinateur). La technologie radar peut fonctionner dans différentes conditions environnementales (par exemple, mauvais temps, mauvaise visibilité ou

impacts environnementaux difficiles comme la couverture de glace, de neige ou de poussière). Si le radar est installé sur le véhicule, il peut être caché derrière des matériaux sans blindage et ne perturbe donc pas la conception du véhicule puisqu'il n'a aucune influence sur l'apparence de celle-ci. Pour différencier davantage les objets détectés (par exemple les piétons ou d'autres usagers), les caractéristiques de densité spectrale de puissance du signal réfléchi peuvent être analysées [36]. Par exemple, le contenu de l'eau dans le corps humain rend la densité spectrale de puissance du signal réfléchi très différente de celle des voitures ou des pôles. Lorsqu'un seul capteur radar installé sur un véhicule peut ne pas être suffisant pour couvrir l'ensemble de la zone intéressée, un réseau radar composé de plusieurs radars avec différents modèles de faisceaux [37] peut être utilisé pour atteindre une grande surface de couverture.

2.1.1.4 Scanner laser

Un scanner laser est un détecteur de gamme laser haute résolution. Le scanner laser émet des impulsions laser infrarouge et détecte les impulsions réfléchies. Le principe de mesure est basé sur la méthode du temps mort, où la distance à la cible est directement proportionnelle à l'intervalle de temps entre la transmission et la réception d'une impulsion. Le balayage du faisceau de mesure est réalisé par un prisme rotatif et couvre un angle de vision jusqu'à 360 degrés. Les données originales d'un scanner laser ressemblent beaucoup aux données d'image de vision. Par conséquent, une procédure similaire au traitement d'image est appliquée. Tout d'abord, les points de données en bloc sont regroupés en différents objets par segmentation. Ensuite, les objets sont classés en différentes catégories en fonction de leurs caractéristiques. Par exemple, les piétons peuvent être classés selon les caractéristiques de leurs jambes mobiles [38]. Enfin, le suivi des objets est effectué par une méthode de filtrage de Kalman [39]. Un système de détection de piétons basé sur un scanner laser multicouches a été construit avec un champ de vision de 120 degrés devant un véhicule [40]. Une région de non échappement (RONE) a été introduite, décrivant une zone devant la voiture où l'accident de voiture à piétons est inévitable si le piéton est détecté dans cette zone. Une excellente précision de gamme et une résolution angulaire fine rendent les scanners laser adaptés aux applications dans lesquelles une image haute résolution de l'environnement est requise. Cependant, puisqu'ils sont des capteurs optiques, les conditions météorologiques comme le brouillard ou la neige limitent leur portée de détection. Le traitement du signal est un peu plus complexe pour le scanner laser comparé au radar à ultrasons ou aux micro-ondes, dont un microprocesseur dédié peut être nécessaire.

2.1.1.5 La vision par ordinateur

La détection des piétons basée sur la vision est un choix naturel basé sur l'expérience d'un homme. Le système de perception visuelle humaine est peut-être le meilleur exemple de performance possible avec ces capteurs de vision. Bien qu'une caméra vidéo puisse obtenir des informations beaucoup plus riches sur l'environnement par rapport à un radar ou un scanner laser, les séquences d'images ne peuvent être utilisées directement sans d'autres interprétations. L'extraction d'informations utiles à partir des séquences d'images disponibles n'est pas une tâche banale pour plusieurs raisons [41] [42]:

1. La détection des piétons implique un environnement extérieur incontrôlé complexe. Les conditions d'éclairage peuvent changer en raison de la météo, le lever du soleil ou le coucher du soleil. Les piétons se retrouvent principalement dans les conditions de circulation de la ville où la texture de fond (par exemple, les bâtiments voisins, les véhicules, les poteaux et les arbres) forme un environnement très encombré.
2. Il existe un large éventail de variations dans l'apparence des piétons en raison de vêtements, de pose, d'occlusion, d'ombre, de mouvement, de taille et de couleur de peau.
3. Si la caméra est installée sur un véhicule en mouvement, cela augmentera la difficulté de différencier les objets de fond et les piétons.
4. Le traitement de l'image nécessite généralement une grande puissance de calcul et la détection des piétons pour un véhicule intelligent nécessite généralement une réponse rapide en considérant la vitesse du véhicule en mouvement.

Divers algorithmes ont été proposés pour détecter les piétons dans les séquences d'images acquises dans les caméras vidéo. Des recherches récentes montrent deux tendances principales. Les approches basées sur les mouvements prennent en compte l'information temporelle et tentent de détecter les caractéristiques périodiques de la marche humaine dans le mouvement des modèles candidats. D'autre part, les approches basées sur la forme reposent sur des caractéristiques de forme pour reconnaître les piétons. Bien que les approches basées sur les mouvements offrent un moyen efficace de réduire le nombre de candidats faux positifs, il existe plusieurs limites aux approches basées sur les mouvements. Tout d'abord, les systèmes basés sur le mouvement ne peuvent évidemment pas détecter les piétons stationnaires ou les piétons engagés dans des mouvements inhabituels comme le saut. Deuxièmement, les pieds ou les jambes du piéton doivent être visibles afin

d'extraire des caractéristiques rythmiques ou des motifs de mouvement. Troisièmement, la procédure de reconnaissance nécessite une séquence d'images, ce qui retarde l'identification jusqu'à plusieurs images plus tard et augmente le temps de traitement. Les méthodes basées sur la forme permettent de reconnaître les piétons mobiles et stationnaires. La principale difficulté associée à cette approche est la façon de tenir compte du large éventail de variations dans les apparences des piétons en raison de la pose, de diverses articulations des parties du corps, de l'éclairage, du vêtement, de l'occlusion, etc. Les principaux problèmes ici sont: trouver une représentation concise et suffisante des caractéristiques de la forme humaine, maintenir un équilibre entre la précision de la détection et le temps de traitement.

Après avoir examiné différentes technologies de capteurs, une idée naturelle est de combiner différentes technologies afin que l'avantage d'une technologie puisse être utilisé pour compenser la limite d'une autre. Dans [43] par exemple, la vision par ordinateur avec le radar à micro-ondes est utilisée pour la détection des piétons à partir d'un véhicule en mouvement. Le radar est utilisé pour générer une liste de cible potentielle et la vision est utilisée pour effectuer une recherche fine.

Dans le cadre de notre projet, nous avons opté pour la fusion de capteurs radar et d'une caméra pour la détection de piétons. Les capteurs radar, moins influençables par les conditions météorologiques que les scanners laser et les ultrasons, seront utilisés pour la détection d'obstacles et l'évaluation de la distance du véhicule par rapport à ces obstacles, tandis que la caméra, dont la technologie de vision par ordinateur est l'approche la plus fiable, aura pour rôle d'identifier la nature des obstacles (piétons, motorcycle, véhicule, etc...). Pour ce qui est de l'identification de la nature des obstacles, notre choix s'est porté sur l'utilisation des approches reposant sur les caractéristiques de formes des différents obstacles.

2.1.2 Les méthodes d'identification de piéton par vision basées sur des caractéristiques de forme

Avec le développement accru de véhicules autonomes et d'autres technologies similaires, il était indispensable de développer des méthodes de détection d'obstacles, de sorte que le véhicule puisse naviguer en toute sécurité. Ceci est incroyablement critique lorsque les obstacles sont des vies humaines, de sorte que le modèle de détection d'image doit pouvoir fournir l'information au véhicule sur la position du piéton et la première étape consiste à détecter l'emplacement du piéton sur une image 2D. Compte tenu de l'importance de la sécurité, le modèle de détection des piétons doit être très précis [44].

2.1.2.1 Historiques [45]

La détection de piéton a été un élément important et une partie difficile de problèmes de vision par ordinateur. L'identification des piétons est une sous-partie de la détection d'objet où l'objectif est de prendre soit une image, soit une vidéo et détecter la présence de personnes dans l'image.

Des techniques d'apprentissage supervisées et non supervisées ont été appliquées à ce problème. Le travail de ROWLEY [46] qui a appliqué des réseaux de neurones pour la détection du visage a été une première étape dans cette direction en mettant l'accent sur la détection des fonctionnalités. Le travail de VIOLA et JONES [47] entre 2001 et 2003 a été une étape majeure dans la détection des objets et des piétons. Ils ont utilisé une représentation d'image intégrale et un nouvel algorithme de détection et d'apprentissage de fonctionnalité basé sur la méthode de boosting² AdaBoost. Les travaux de DALAL et TRIGGS [48] sur le HOG (Histograms of Oriented Gradients) pour la détection humaine, ont permis d'améliorer davantage les systèmes de détection prédictifs et les systèmes de détection des piétons. Les travaux qui ont suivis visaient le développement des techniques d'amélioration des méthodes basées sur les fonctionnalités afin d'améliorer la détection des fonctionnalités. La plupart de ces techniques ont utilisé la détection par fonctionnalité couplée aux méthodes de régression SVM pour effectuer la détection des piétons. [49] [50].

Dans la vision par ordinateur et le traitement d'image, le concept de détection de fonctionnalité (ou caractéristique) se réfère à des méthodes qui visent à calculer des abstractions d'informations d'image et à prendre des décisions locales à chaque point d'image, qu'il y ait ou non une image d'un type donné. Une fonctionnalité peut être définie ici comme une partie intéressante d'une image. De nombreux algorithmes de vision par ordinateur utilisent la détection de fonctionnalité comme étape initiale, de sorte qu'un très grand nombre de détecteurs de caractéristiques ont été développés [51].

Cependant, depuis 2012, les CNN (Convolutional Neural Network) qui désignent les réseaux neuronaux de convolution ont révolutionné le domaine de la vision par ordinateur et détection d'image. Des architectures comme AlexNet [52] et GoogleNet [53] ont permis de réduire considérablement les erreurs des précédents algorithmes de détection pendant ILSVRC (ImageNet Large-Scale Visual Recognition Challenge), qui est la compétition annuelle mondiale de la vision par ordinateur [54].

² Le boosting est un domaine de l'intelligence artificielle dont le principe est de regrouper de nombreux algorithmes qui s'appuient sur des ensembles de classifieurs binaires.

Cela a provoqué un changement important dans l'approche de la communauté avec l'application avec succès des CNN à un grand nombre de problème de classification de vision et d'image. Les CNN étant un concept assez récent, leur application à la détection de piétons est cependant encore à sa phase embryonnaire [45].

En s'inspirant des travaux de VARIYAR [45], la résolution du problème de détection suit une procédure présentée sur la **Figure 2.1** ci-dessous. La procédure contient une méthode de délimitation de région, une extraction de fonctionnalités et une classification des régions. Les algorithmes de détection d'objet basés sur l'apprentissage en profondeur plus connu sous son appellation anglaise « Deep Learning » comme les CNN, ne nécessitent pas l'étape d'extraction des fonctionnalités.



Figure 2.1 : Procédure de détection de piéton, tiré de [45]

Dans ce mémoire, nous proposons la fusion de la méthode HOG avec la méthode SVM pour l'identification de piéton, ceci du fait de la simplicité de leurs implémentations, et l'accessibilité à un nombre conséquent de travaux à ce sujet. Nous emploierons la méthode HOG pour la délimitation de la région et l'extraction des fonctionnalités, et la méthode SVM nous permettra de classifier les différentes régions.

2.1.2.2 Le descripteur de caractéristiques HOG

L'image d'entrée d'un système de vision par ordinateur contient trop d'informations supplémentaires qui ne sont pas nécessaires pour la classification. Par conséquent, la première étape de la classification des images consiste à simplifier l'image en extrayant les informations importantes contenues dans l'image et en laissant de côté le reste. Un algorithme d'extraction de fonctionnalités convertit une image de taille fixe en un vecteur caractéristique de taille fixe.

Dans le cas de la détection des piétons, le descripteur de la caractéristique HOG définit dans une région les proportions de pixels dont l'orientation du gradient appartient à un certain intervalle. Ces proportions caractérisent la forme présente dans cette région. La distribution (histogramme) des directions des gradients représente la fonctionnalité du descripteur HOG. Les contours d'une image

donnent une bonne représentation des formes qui y sont présentes. Le calcul du gradient, notamment par la méthode de Canny cité par MIGNIOT [55] permet une détection performante des contours. Le calcul du descripteur HOG peut être regroupé en cinq étapes [56]:

- ⑧ **Prétraitement** : Cette étape consiste à extraire un patch³ de l'image. Ici, les images peuvent être de n'importe quelle taille, mais la contrainte étant néanmoins, que les patches aient un rapport d'aspect fixe.
- ⑧ **Calcul du gradient de l'image** : Il est question dans cette étape de calculer les gradients horizontaux et verticaux de l'image. Ceci est réalisé en filtrant tout d'abord l'image avec les masques $[-1 \ 0 \ 1]$ et $[-1 \ 0 \ 1]^T$. On obtient alors les valeurs G_x et G_y , représentant respectivement les valeurs des gradients horizontaux et verticaux. Ensuite, on effectue le calcul de la magnitude (équation (2-1)) et de la direction (équation (2-2)) du gradient.

$$g = \sqrt{g_x^2 + g_y^2} \quad (2-1)$$

$$\theta = \text{actan} \frac{g_y}{g_x} \quad (2-2)$$

Cette étape supprime beaucoup d'informations non essentielles (par exemple un arrière-plan coloré constant) et met en évidence des contours. A chaque pixel⁴, le gradient a une amplitude et une direction.

- ⑧ **Calcule de l'histogramme des gradients par cellules de $X \times X$** : Dans cette étape, l'image est divisée en $X \times X$ cellules, et un histogramme de gradients est calculé pour chaque $X \times X$ cellules. X représente le nombre de pixel dont le choix dépend de l'échelle des fonctionnalités recherchées. Il est à souligner que plus X est petit, plus la représentation de l'histogramme sera moins sensible au bruit. L'histogramme est essentiellement un vecteur (ou tableau) de 9 cases (nombres) correspondant à des valeurs d'angle prédéfini.
- ⑧ **Normalisation des blocs** : Les gradients d'une image étant sensible à l'éclairage, si l'on assombri par exemple une image en divisant toutes les valeurs de pixels par 2, l'amplitude du gradient changera de moitié, et les valeurs de l'histogramme également. Cette étape consiste

³ Patch : en vision par ordinateur, un patch est une partie d'une image. Source : www.wikipedia.org

⁴ Pixel : unité de base permettant de mesurer la définition d'une image numérique matricielle. source : www.wikipedia.org

donc à normaliser l'histogramme afin qu'il ne soit pas affecté par les variations d'éclairage. La norme utilisée ici est la norme L2 [57]

- ☞ **Calcul du vecteur de fonction HOG** : Cette dernière étape consiste simplement à concaténer dans un vecteur géant les différents vecteurs obtenus par bloc.

2.1.2.3 La méthode de classification SVM

Nous avons vu précédemment comment convertir une image en vecteur de fonctionnalité. Dans cette sous-section, nous allons apprendre comment un algorithme de classification prend ce vecteur de caractéristique comme entrée et produit une étiquette de classe (par exemple, un piéton ou arrière-plan). Avant qu'un algorithme de classification puisse faire cette distinction, nous devons l'entraîner en montrant des milliers d'exemples de piétons et d'arrière-plans. Différents algorithmes d'apprentissage apprennent différemment, mais le principe général est que les algorithmes d'apprentissage traitent les vecteurs de caractéristiques comme des points dans un espace dimensionnel supérieur et essaient de trouver des plans (ou surfaces) qui divisent l'espace dimensionnel supérieur de telle sorte que tous les exemples appartenant à la même classe soient sur un côté de plan (ou surface).

Le SVM est l'un des algorithmes de classification binaire par apprentissage supervisé les plus populaires. Bien que les idées utilisées dans SVM aient été développées depuis 1963, la version actuelle a été proposée en 1995 par CORTES et VAPNIK [58]. La classification se réalise sur des vecteurs à N dimensions. Dans notre cas, ces vecteurs doivent représenter des caractéristiques des images. L'élément testé ainsi que les éléments de la base de données sont dans ce format pour permettre les comparaisons. La première partie du procédé consiste à créer un modèle à partir de la base de données. Il s'agit de l'apprentissage de la classe. La base de données se décompose en un ensemble d'éléments positifs (contenant un élément de la classe) et un ensemble d'éléments négatifs (ne contenant pas d'élément de la classe). Un hyperplan séparant les éléments de chacun des deux ensembles est calculé de façon à maximiser la marge (**Figure 2.2**), c'est-à-dire la distance entre les échantillons et l'hyperplan. Pour cela, l'espace d'étude est retranscrit sur un espace de plus grande dimension où l'existence d'un séparateur linéaire est possible. Finalement, l'ensemble des exemples positifs se trouve d'un côté de l'hyperplan tandis que l'ensemble des exemples négatifs se trouve de l'autre côté. Dans la seconde partie du procédé, ce modèle permet la décision. Si le vecteur testé se trouve du côté de l'hyperplan relatif aux exemples positifs, alors il s'agit d'un élément de la classe.

Dans le cas contraire, il ne s'agit pas d'un élément de la classe. On peut remarquer que la distance du vecteur de caractéristiques à l'hyperplan donne une évaluation de la fiabilité de la décision. En effet, si cette distance est très réduite, la décision sera moins tranchée car l'exemple est très proche des deux classes [55].

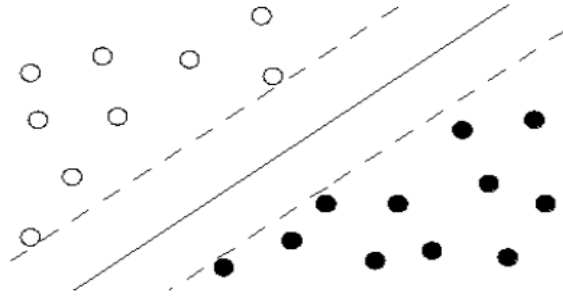


Figure 2.2 Illustration du principe de fonctionnement des SVMs pour un cas linéaire simple. Les éléments d'une classe (ronds blancs) sont séparés des éléments d'une autre classe (rond noirs) par l'hyperplan séparateur (trait plein) qui maximise la marge (traits en pointillés), tiré [55]

2.2 L'intelligence artificielle appliquée à la navigation

Plusieurs définitions peuvent être données à la tâche de navigation selon le but recherché. Il existe néanmoins une très grande variété de travaux et des méthodes permettant d'aborder ce problème difficile. Les stratégies de navigation permettant à un système mobile d'évoluer de manière autonome pour rejoindre un but sont extrêmement diverses, elles dépendent de la connaissance de son environnement qui peut être figé ou évolutif et du mode de définition de la trajectoire (apprentissage préalable, planification en ligne) [59]. Les performances du système de navigation sont étroitement liées à la précision, à la fiabilité et au temps de réponse des capteurs et des méthodes mises en œuvre pour localiser le véhicule [60]. La navigation autonome d'un véhicule s'articule autour de trois niveaux principaux : la localisation et la planification, la perception de l'environnement et enfin le raisonnement et la décision [59] [61] [62].

Le niveau de raisonnement et de prise de décision constitue l'intelligence du système de navigation autonome. A l'instar de l'homme, le raisonnement du système autonome doit lui permettre de décider d'une action appropriée à une situation donnée, compte tenu d'une mission à réaliser. Cette décision est ensuite transmise au niveau fonctionnel pour opérer les différentes parties qui satisfont cette mission, la mission ici pouvant être le suivi d'une trajectoire, l'évitement d'obstacles ou le parking [63].

L'évitement d'obstacles constituant le nœud de notre projet, nous allons, dans ce chapitre, nous intéresser à cet aspect de la navigation d'un véhicule. Plus particulièrement, nous allons définir le concept d'évitement d'obstacles en présentant les différentes approches et méthodologie à ce sujet. Nous allons, par la suite, nous pencher sur deux techniques de l'intelligence artificielle en occurrence la logique floue et l'apprentissage automatique. Nous terminerons ce chapitre par le choix d'une méthode que nous appliquerons dans la conception de notre système.

2.2.1 Les approches d'évitement d'obstacle

Il existe plusieurs approches pour l'évitement d'obstacles. Nous allons citer les plus utilisées [59] :

- Celles qui calculent un ensemble de solutions potentielles compte-tenu des informations sur l'environnement, puis sélectionnent une solution particulière afin de satisfaire des

contraintes de tâche. Les solutions peuvent prendre une forme de direction privilégiée du véhicule ou d'une consigne en vitesse. Il s'agit des méthodes dites délibératives.

- Celles qui calculent directement une commande à partir des informations disponibles sur l'environnement. Il s'agit des approches de l'intelligence artificielle inspirées du vivant (réseaux de neurones artificiels [64] , approche bayésienne [65], logique floue [66]), ou des approches basées sur des phénomènes physiques naturels (les champs de potentiels [67]). Il s'agit des méthodes dites réactives.

Les approches délibératives présentent un problème majeur, elles n'implémentent aucun comportement intelligent de haut niveau, réduisant ainsi leur capacité de traité des situations complexes. Ce qui présente un risque de ne pas avoir une solution fiable au problème d'évitement d'obstacles. Pour pallier à ces limites, les chercheurs se sont orientés vers leur utilisation en tant que solution partielle en les injectant dans des approches réactives où ils deviennent capables de traiter des situations plus complexes [68].

Les approches réactives, contrairement aux approches délibératives sont basées sur le caractère d'intelligence. Elles sont capables de modéliser un savoir-faire de haut niveau de l'aspect cognitif du système mobile. Tout particulièrement, nous nous intéresserons relativement dans notre mémoire, aux approches réactives d'évitement d'obstacle basées sur des contrôleurs flous entraînés par apprentissage.

2.2.2 La logique floue

La logique floue est une théorie de l'intelligence artificielle basée sur les ensembles flous. La méthode fut introduite par ZADEH [69]. L'objectif de la logique floue telle que illustrée sur la **Figure 2.3** ci-dessous, est de fournir une fondation pour le raisonnement approximatif, qui utilise des propositions imprécises basées sur la théorie des ensembles flous.

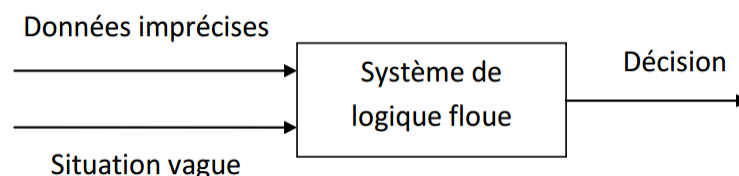


Figure 2.3 : Schéma de principe de la logique floue, tiré de [68]

2.2.2.1 Concept de base de la logique flou [63]

La logique floue peut être vue comme une extension de la logique booléenne; elle permet de traiter des variables linguistiques dont les valeurs sont des mots ou expressions du langage naturel.

Soit U un univers de discours, A un sous ensemble de U et x un élément de U . Un ensemble classique (**figure 2.4 (a)**) A de U est défini par une fonction caractéristique notée $\mu_A(x)$ telle que :

$$\mu_A(x) = \begin{cases} 1, & x \in A \\ 0, & \text{sinon} \end{cases} \quad (2-3)$$

Un ensemble flou par contre, repose sur la notion de sous ensemble flou. Il peut être considéré comme une généralisation de l'ensemble classique. La fonction d'appartenance d'un ensemble classique peut prendre seulement deux valeurs $\{0,1\}$, tandis qu'un ensemble flou peut être représenté comme un ensemble de paires ordonnées :

$$A = \{x, \mu_A(x) / x \in U\} \quad (2-4)$$

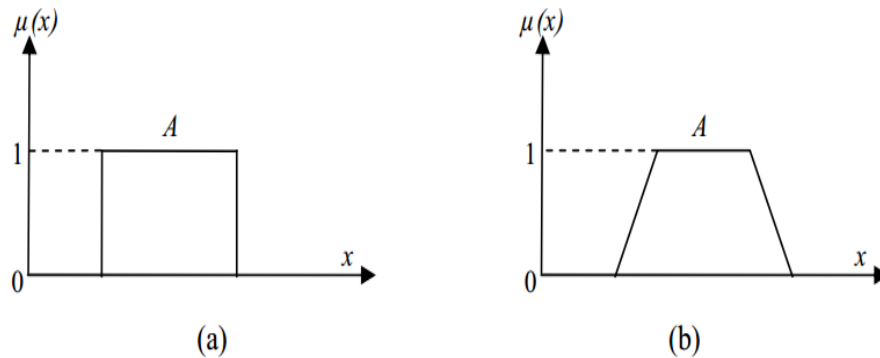


Figure 2.4 : Exemples des fonctions d'appartenance (a) Logique classique (b) logique floue, tiré de [63]

Les ensembles flous ont le grand avantage de constituer une représentation mathématique des termes linguistiques utilisés dans le langage naturel des experts. Les notions de base de la logique floue sont : les fonctions d'appartenance, les variables linguistiques, les opérations sur les ensembles flous et le raisonnement flou.

- a- Fonctions d'appartenance :** Ce sont les fonctions qui expriment le degré d'appartenance d'une grandeur à une variable linguistique. Elles peuvent prendre n'importe quelle forme,

mais les plus utilisées sont les fonctions d'appartenances de forme trapézoïdales, triangulaires, ou en forme de cloche (gaussienne).

- b- Variable linguistique :** C'est une variable dont les valeurs ne sont pas des nombres, mais des mots ou phrases exprimés en langage naturel. La raison pour laquelle on utilise cette représentation, est que le caractère linguistique est moins spécifique que le caractère numérique. Une variable linguistique est généralement représentée par un triplet $(x, T(x), U)$ dans lequel x représente le nom de la variable linguistique, $T(x)$ l'ensemble des valeurs linguistiques qui sont utilisées pour caractériser x , et U l'univers de discours de la variable linguistique x .
- c- Le raisonnement flou :** permet de raisonner lorsque les règles ou les faits sont connus de façon imparfaite.
- d- Opérations sur les ensembles flous :** La représentation formelle des ensembles flous par des fonctions d'appartenance a permis de généraliser les opérateurs des ensembles classiques au cas flou. On distingue parmi les opérations sur les ensembles flous; l'égalité floue, le sous-ensemble flou, la complémentation floue, l'union floue, l'intersection floue, le produit cartésien, implication floue, l'intersection.

2.2.2.2 Les systèmes d'inférence flous (SIF)

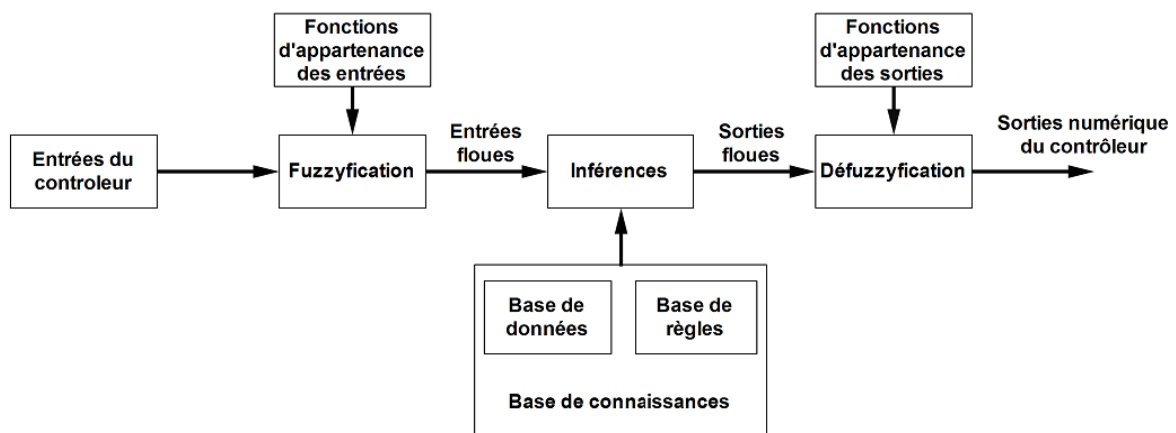


Figure 2.5 : Structure d'un contrôleur flou [59]

Un système flou (contrôleur flou) peut être interprété selon deux points de vue: mathématique ou logique. D'un point de vue mathématique, un système flou est une fonction non linéaire

reliant un vecteur de données d'entrée à un vecteur de sortie. D'un point de vue logique, un système flou est un système à base de connaissance particulière (système expert) composé de quatre modules principaux, à savoir: la fuzzification, la base de connaissances, le moteur d'inférence et la défuzzification. La **figure 2.5** ci-dessus, montre la structure d'un tel contrôleur flou. Ainsi, la description sommaire de chaque module composant le contrôleur flou est la suivante [63]:

- a- Interface de fuzzification:** L'opération de fuzzyfication permet le passage du domaine réel vers le domaine du flou. Les variables d'entrées du régulateur sont transformées en variables linguistiques avec la définition des fonctions d'appartenances. Ainsi cette opération consiste à déterminer le degré d'appartenance d'une valeur à un ensemble flou. Les fonctions d'appartenances sont généralement fixes, selon l'expertise recueillie par le concepteur.
- b- La base de règles floues ou base de connaissance du processus :** Elle est composée de l'ensemble des renseignements que nous possédons sur le processus. Elle permet de définir les fonctions d'appartenance et les règles floues qui décrivent le comportement du système. C'est le cœur du système entier dans le sens où tous les autres composants sont utilisés pour interpréter et combiner ces règles pour former le système final.
- c- Moteur d'inférence (Raisonnement flou) :** Une action, sous forme symbolique, est décidée à l'aide des techniques de raisonnement flou en fonction des variables floues précédemment calculées.
- d- Interface de défuzzification :** L'ensemble flou de sortie est déterminé à la fin de l'inférence, mais il n'est pas directement utilisable. Il est alors nécessaire de passer du *monde flou* au *monde réel*, et ainsi obtenir une valeur numérique précise qui sera appliquée au système. C'est la défuzzyfication. Il existe plusieurs méthodes de défuzzyfication parmi elles on peut citer :
 - La technique du centre de gravité : Elle consiste à tracer sur le même diagramme les différentes courbes correspondant à chacune des règles, et à calculer le centre de gravité de la zone commune.
 - La technique de la moyenne pondérée : On prend comme valeur de sortie la moyenne des différentes valeurs pondérées par leurs pourcentages.
 - La technique du maximum : C'est une technique qui consiste à prendre en considération une seule valeur possédant le degré d'appartenance maximum. Cette

technique est peu précise, car les valeurs secondaires apportent une certaine nuance dont on ne tient pas compte

2.2.2.3 Les méthodes de raisonnement flou [63]

Plusieurs méthodes de raisonnement flou ont été proposées dans la littérature suivant la réalisation des opérateurs flous "ET" et "OU" et le type des règles floues utilisées. Les trois moteurs d'inférence floue les plus utilisés sont: le moteur de Mamdani, de Sugeno et celui de Tsukumoto.

2.2.2.3.1 La méthode de Mandani

Mamdani fut le premier à utiliser la logique floue pour la synthèse des commandes. L'inférence floue par la méthode de Mandani correspond aux étapes suivantes, pour un vecteur d'entrée $x = (x_1, \dots, x_n)^t$:

1. Calcul du degré d'appartenance de chaque entrée aux différents sous-ensembles flous:

$$\mu_{A_j}^i(x_j), \text{ pour } j \in [1, n] \text{ et } i \in [1, N] \quad (2-5)$$

où n est la dimension d'espace d'entrée et N le nombre de règles floues.

2. Calcul de la valeur de vérité α de chaque règle, pour i allant de 1 à N :

$$\alpha_i(x) = \min_j \left(\mu_{A_j}^i(x_j) \right), \text{ pour } j \text{ allant de } 1 \text{ à } n \quad (2-6)$$

3. Calcul de la contribution de chaque règle :

$$\mu_i(y) = \min(\alpha_i(x), \mu_{B_i}(y)) \quad (2-7)$$

4. Agrégation des règles en utilisant l'opérateur max:

$$\mu(y) = \max_i(\mu_i(y)) \quad (2-8)$$

Le résultat est donc un sous ensemble flou caractérisé par sa fonction d'appartenance. Pour obtenir une conclusion exacte, il faut défuzzifier. En utilisant la technique du centre de gravité on parlera d'implémentation (min, max, barycentre). D'autres variantes de la méthode Madani existent comme (min, produit, barycentre) et (produit, somme, barycentre).

2.2.2.3.2 La méthode de Takagi-Sugeno

La méthode Takagi-Sugeno (TS) constitue un cas particulier important; où la conclusion n'est pas symbolique mais une fonction des entrées: $B^i = f(x_1, \dots, x_n)$.

La forme la plus utilisée est la suivante:

$$B^i = \sum_{j=0}^n b_j^i x_j \quad (2-9)$$

Un SIF de type TS réalise une fusion de modèles locaux par interpolation. Un SIF de type TS est dit d'ordre zéro ou de méthode de TS simplifiée si la conclusion est une constante, et d'ordre 1, quand la conclusion est un hyperplan (**équation (2-9)**).

Pour un vecteur d'entrée $x = (x_1, \dots, x_n)^t$, la sortie d'un système d'inférence flou de type TS d'ordre 0 est calculée selon les étapes suivantes :

1. Calcul du degré d'appartenance de chaque entrée aux différents sous-ensembles flous :

$$\mu_{A_j}^i(x_j), \text{ pour } j \in [1, n] \text{ et } i \in [1, N] \quad (2-10)$$

2. Calcul de la valeur de vérité de chaque règle, pour i allant de 1 à N :

$$\alpha_i(x) = ET(\mu_{A_1}^i(x_1), \dots, \mu_{A_n}^i(x_n)) \quad (2-11)$$

3. Calcul de la sortie :

$$y = \frac{\sum_{i=1}^N \alpha_i(x) \cdot b^i}{\sum_{i=1}^N \alpha_i(x)} \quad (2-12)$$

2.2.2.3.3 Méthode de Tsukumoto

Dans ce cas, des fonctions monotoniques sont associées aux variables de sortie. La sortie totale est une moyenne pondérée des degrés de confiance des règles floues et des valeurs des fonctions des variables de sortie. La **figure 2.6** illustre les types du raisonnement flou pour un système flou à deux entrées et une base de connaissances de deux règles floues. On constate que les différences viennent de la spécification de la partie conclusion d'une part, et de la méthode de défuzzification d'autre part.

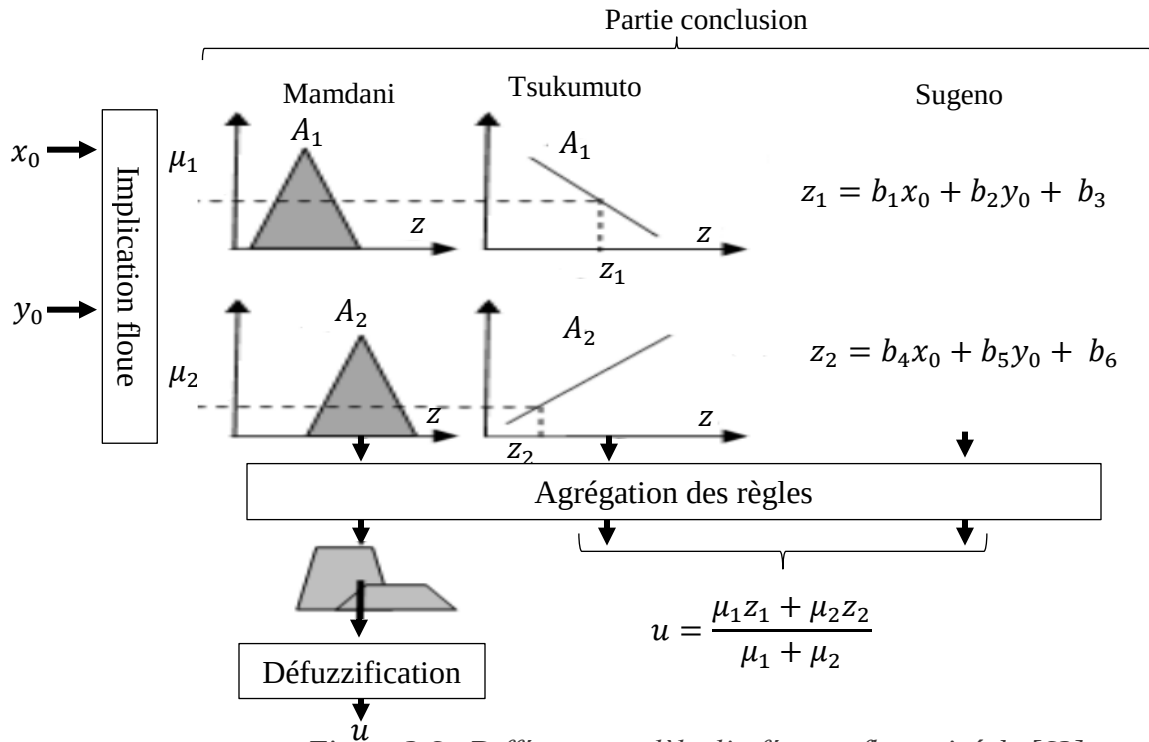


Figure 2.6 : Différents modèle d'inférence flous, tiré de [63]

Trois raisons principales amènent à utiliser les SIF comme contrôleurs des systèmes dynamiques. Premièrement, ce type de systèmes a la propriété d'approximateur universel de fonctions continues, avec un degré de précision quelconque, à condition d'utiliser un nombre suffisant de règles floues. Deuxièmement, les SIF sont construits à partir de règles floues de la forme Si-Alors, de ce fait, les informations linguistiques ou mathématiques disponibles d'un expert peuvent éventuellement être incorporées dans le contrôleur. Troisièmement, la logique floue présente comme avantage de ne pas être coûteuse en complexité de calcul, il s'agit d'un outil puissant et assez simple à implémenter. Mais ces contrôleurs présentent néanmoins quelques inconvénients, entre autres, la nécessité d'évaluer et d'affiner les contrôleurs (les fonctions d'appartenance d'entrée et de sortie et les tables d'inférence). Le réglage des différents paramètres dans un environnement pouvait fournir

de bons résultats pour cet environnement et ne pas répondre correctement face à une situation nouvelle [63]. D'autre part, un autre problème est celui de la sélection de la technique d'acquisition de connaissances et la disponibilité d'un expert humain [68].

Dans cette section, nous avons introduit les principes de base des ensembles flous qui sont une généralisation du concept d'ensembles classiques. Contrairement à la fonction caractéristique d'un ensemble net qui prend la valeur 0 ou 1, la fonction d'appartenance d'un ensemble flou prend ses valeurs dans l'intervalle $[0,1]$. A partir des ensembles flous, nous pouvons construire des systèmes d'inférence flous (SIF). Nous avons constaté qu'il existe plusieurs types de SIF. Le fonctionnement d'un contrôleur flou dépend d'un nombre important de paramètres (méthode de fuzzification, le type des fonctions d'appartenance, le type des règles floues, la méthode du raisonnement flou et la stratégie de défuzzification) qu'il faut déterminer lors de la conception. Comme ces paramètres s'influencent mutuellement, leur réglage n'est donc pas aisé. Afin d'augmenter l'adaptabilité de notre système, nous nous sommes tournés vers le domaine de l'apprentissage automatique. Une solution possible est l'application des techniques d'apprentissage pour l'ajustement et la génération des paramètres du contrôleur flou.

2.2.3 L'apprentissage automatique

L'apprentissage est considéré comme une tâche de construction de nouvelles connaissances ou amélioration des connaissances existantes. Le but est d'améliorer les performances du système en tenant compte des ressources et des compétences dont il dispose. Il existe plusieurs formes d'apprentissage et de modification de comportement, selon le type d'informations disponibles; on peut citer: l'apprentissage supervisé, non supervisé et l'apprentissage par renforcement. Chacun avec ses propriétés intéressantes. Grâce à ces différentes propriétés, l'utilisation des techniques d'apprentissage, pour l'amélioration des comportements et la commande des processus, est devenue un domaine très important dans les dernières décennies [70] [71].

Dans cette section, nous allons tout d'abord présenter les définitions et les notions de base de l'apprentissage automatique, ensuite, nous allons présenter une technique importante de l'intelligence artificielle pour l'adaptation des comportements : l'apprentissage automatique basé sur un signal critique d'évaluation (apprentissage par renforcement).

2.2.3.1 Définition

2.2.3.1.1 Intelligence artificielle (IA)

L'intelligence artificielle est la reproduction, par des moyens artificiels de toutes les formes de l'intelligence humaine pour un objectif final, qui s'intéresse à la conception des systèmes intelligents. L'IA est la capacité à raisonner, à apprendre, et à s'adapter face à de nouveaux changements [72].

2.2.3.1.2 Agent

D'après FERBER [72], un agent est une entité autonome réelle ou abstraite qui est capable d'agir sur elle-même et sur son environnement, et peut communiquer avec d'autres agents, dont le comportement est la conséquence de ses observations, ses connaissances et de ces interactions avec son environnement. Il doit être autonome, interactif, adaptatif, rationnel, coopératif et intelligent. La notion d'agent se diffère selon l'utilisation.

2.2.3.1.3 Apprentissage automatique

L'apprentissage automatique fait référence au développement, à l'analyse et à l'implémentation de méthodes qui permettent à une machine d'évoluer grâce à un processus d'apprentissage, et ainsi, de remplir des tâches qu'il est difficile ou impossible de remplir par des moyens algorithmiques plus classiques. L'objectif est d'extraire et d'exploiter automatiquement l'information présentée dans un jeu de données. La **figure 2.7** ci-dessous schématise le processus d'apprentissage. L'environnement agit sur le module d'apprentissage de l'agent, cela a pour effet, de consulter et de modifier la base de connaissances pour arriver à une exécution adéquate. De cette exécution, un retour est attendu pour évaluer le résultat obtenu [63].

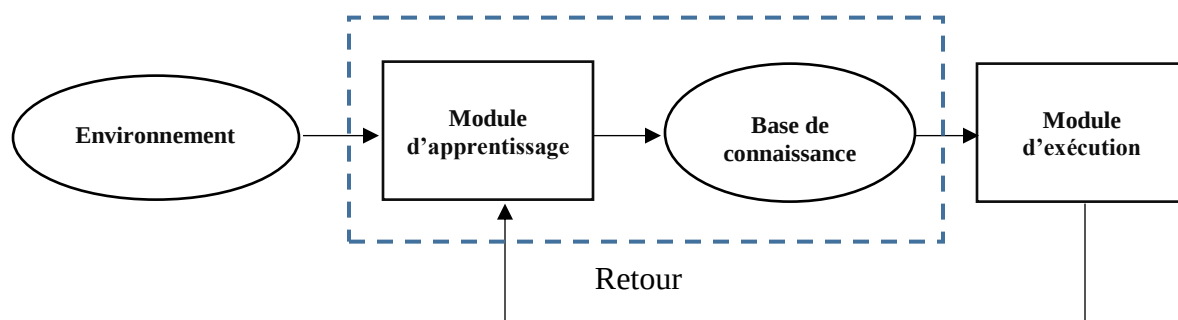


Figure 2.7 : Représentation de l'apprentissage automatique [63].

Il existe trois types d'apprentissage :

🌀 Apprentissage supervisé

Le contrôleur (ou le maître) fournit l'action qui devrait être exécutée. Un expert est employé pour étiqueter correctement des exemples (**figure 2.8**). Ces exemples doivent être représentatifs de la tâche à accomplir. L'apprenant doit alors trouver ou approximer la fonction qui permet d'affecter la bonne étiquette à ces exemples [73].

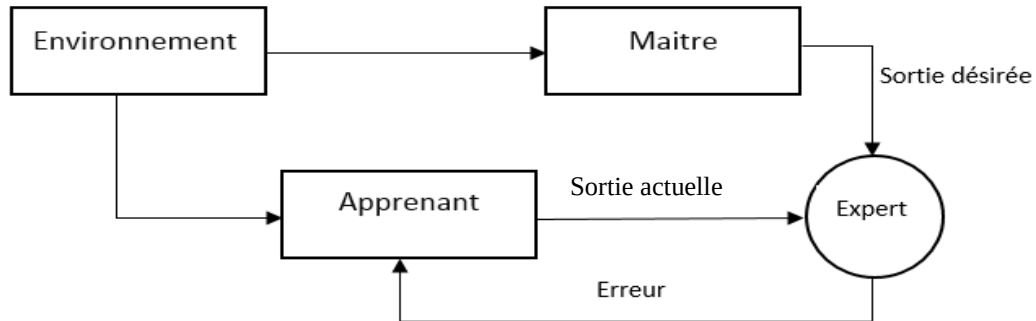


Figure 2.8 : Représentation de l'apprentissage supervisé, tiré de [73]

🌀 Apprentissage non supervisé:

Ici l'apprenant doit identifier par lui-même la meilleure réponse possible, il n'y a pas de réponse désirée, ni d'expert (**figure 2.9**). La tâche peut être par exemple, de créer des regroupements de données selon des propriétés communes (catégorisation) [64].

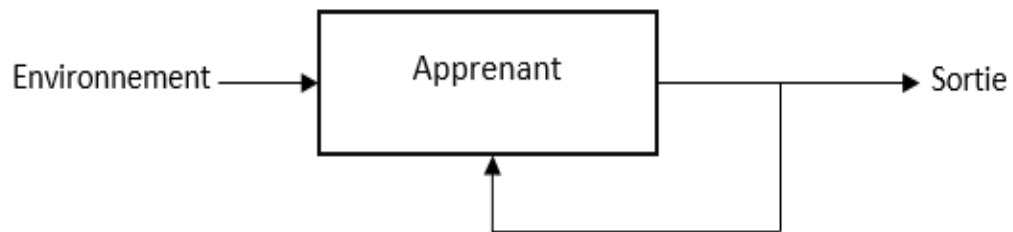


Figure 2.9 : Représentation de l'apprentissage non supervisé, tiré de [64]

☯ Apprentissage par renforcement (AR)

Le contrôleur a un rôle d'évaluateur et non pas d'instructeur, l'information disponible est un signal de renforcement, généralement appelé critique. Son rôle est de fournir une mesure indiquant si l'action générée est appropriée ou non. Le contrôleur doit déterminer et modifier ses actions de manière à obtenir une meilleure évaluation dans le futur [74].

L'apprentissage non supervisé cherchant à trouver des partitions de modèles par lui-même, il a plus un caractère descriptif, et est donc en cela pas adapté pour la commande de système. L'apprentissage supervisé par contre est adapté pour la commande de par la possibilité, à travers un maître, de fournir les actions à exécuter, il présente néanmoins l'inconvénient de nécessiter un ensemble de paires état-action définissant la base de donnée pour l'apprentissage des différents comportements (ralentissement, évitement d'obstacles). Dans le cadre de notre projet, nous utiliserons l'apprentissage par renforcement, du fait qu'il se distingue des autres approches par deux principaux aspects : l'apprentissage se fait sans supervision et il repose sur le principe d'essai/erreur, ce qui le rend très utile dans le cadre de problèmes, où des stratégies comportementales efficaces sont inconnues a priori ou sont difficilement automatisable et lorsqu'il y a de l'incertain dans la manière dont l'environnement évolue.

2.2.3.2 L'apprentissage par renforcement

2.2.3.2.1 Définition et principe

L'apprentissage par renforcement est une approche de l'intelligence artificielle qui permet l'apprentissage d'un agent par l'interaction avec son environnement; afin de trouver, par un processus essais-erreurs, l'action optimale à effectuer pour chacune des situations que l'agent va percevoir pour maximiser ses récompenses [75] [74]. A chaque étape l'agent doit définir l'état dans lequel il se trouve. A partir de cet état, il doit prendre une décision sur l'action à exécuter. En fonction du résultat obtenu lors de l'exécution de cette action, il est soit puni pour diminuer la probabilité d'exécution de la même action dans le futur, soit récompensé pour favoriser ce comportement dans les situations pareilles (**figure 2.10**).

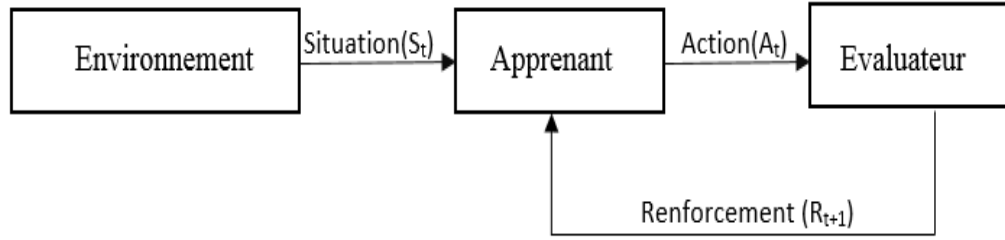


Figure 2.10 : Représentation de l'apprentissage par renforcement, tiré [74]

La formalisation mathématique des algorithmes d'apprentissage par renforcement telle que nous la connaissons aujourd'hui repose sur la notion du Processus de Décision Markoviens (PDM) proposée par Bellman [76]. L'apprentissage par renforcement est basé sur l'apprentissage à temps discret des paramètres d'une chaîne de Markov. Les chaînes de Markov sont représentées par des états et de transitions entre ces états. Elles sont définies par un quadruplet (S, A, P, R) [74] où :

S : est un ensemble d'états fini,

A : est un ensemble d'action fini. On note $A(s)$ l'ensemble des actions possibles dans l'état s .

P : est la fonction de transition décrivant la dynamique de l'environnement. Elle définit une distribution de probabilité de choisir une action à un état donné :

$$\begin{cases} P: S * A * S \rightarrow [0,1] \\ (s, a, a') \rightarrow P(s, a, a') = P_r[s_{t+1} = s' | s_t = s, a_t = a] \end{cases} \quad (2-13)$$

R : est la fonction de retour (récompense ou punition). On associe pour chaque transition une valeur de probabilité sur $\mathcal{R}$:

$$\begin{cases} R: S * A * S \rightarrow \mathcal{R} \\ (s, a, a') \rightarrow R(s, a, a') = E[r_t | s_{t+1} = s', s_t = s, a_t = a] \end{cases} \quad (2-14)$$

Les PDM respecte la propriété de Markov qui stipule : l'état courant et le retour courant ne dépendent que de l'état précédent et de l'action qui vient d'être émise. Il n'est alors nul besoin de mémoire pour prendre des décisions au mieux: seule la connaissance de l'état courant est utile. La seule trace d'une mémoire réside dans l'apprentissage de comportement effectué par l'agent (on dit que l'environnement est markovien) [77].

2.2.3.2.2 Éléments de base de l'AR

En plus de l'agent et de l'environnement, on peut identifier trois éléments principaux d'un système d'AR : la politique, la fonction de récompense, la fonction de valeur. La politique définit la manière de se comporter de l'agent à un instant donné, c'est-à-dire quelle action effectuer à cet instant. La fonction de récompense correspond au but du problème considéré. Elle associe chaque état (ou paire état-action) de l'environnement à un nombre, la récompense. Elle définit quels sont les événements mauvais et les événements bons pour l'agent. La fonction de valeur spécifie ce qui est bon à long terme. Pour parler approximativement, la valeur d'un état est la quantité totale de récompense qu'un agent peut s'attendre à accumuler dans le futur en partant de cet état.

a- La politique

Le comportement de l'agent est défini par une politique π , qui guide l'agent de manière probabiliste en spécifiant, pour chaque état s la probabilité de réaliser l'action a ($\pi(s) = a$). Le but est de trouver la politique optimale π^* maximisant la récompense à long terme défini par l'équation 2-15.

$$\begin{cases} \pi: \{S, A\} \rightarrow [0, 1] \\ (s, a) \rightarrow \pi(s, a) = P_r[a_t = a | s_t = s] \end{cases} \quad (2-15)$$

b- Choix d'action (dilemme Exploration-Exploitation)

Le choix et la génération de l'action est un problème majeur en commande. Il doit garantir un équilibre entre l'exploration et l'exploitation de l'apprentissage déjà réalisé; c'est-à-dire faire confiance à l'estimation courante de la qualité Q pour choisir la meilleure action à effectuer dans l'état courant (exploitation) où, au contraire, choisir une action à priori sous optimale pour observer ses conséquences (exploration). Naturellement, on conçoit intuitivement que l'exploration doit initialement être importante (quand l'apprentissage est encore très partiel, on explore) puis diminuer au profit de l'exploitation quand l'apprentissage a été effectué pendant une période suffisamment longue. Plusieurs stratégies sont utilisées pour le choix d'action, entre autre la stratégie Gloutonne, e-gloutonne, softmax et Boltzmann. La description de ces différentes stratégies est faite dans l'annexe 1.

c- Fonctions valeurs et récompense

Pour mesurer la performance du comportement d'un agent qui est associé naturellement à un problème de la forme d'un PDM; on utilise l'équation 2-17 représentant la formule de la fonction valeur qui est l'espérance de gain si l'on suit la politique π à partir de l'état courant s .

$$R_t = \sum_{k=1}^{\infty} \gamma^k r_{t+k}, \gamma \in [0,1] \quad (2-16)$$

$$V_{\pi}(s) = E_{\pi}(R_t | s_t = s) = E_{\pi}(\sum_{k=1}^{\infty} \gamma^k r_{t+k} | s_t = s) \quad (2-17)$$

Où R_t représente la fonction de retour, γ le facteur de dépréciation qui permet de régler l'importance que l'on donne aux retours futurs par rapport aux retours immédiats. Plus γ se rapproche de 1, plus les retours futurs sont prisent en compte.

Il faut toutefois noter que ce critère n'est pas la meilleure évaluation de la politique, mais il reste le plus pratique à mettre en œuvre. En plus d'état, d'action, fonction de récompense, et de fonction de probabilité de transition, un autre concept important du PDM est la fonction qualité $Q^{\pi}(s, a)$, qui inclut la fonction valeur d'état et d'action. De ces fonctions valeurs, seront construits les 3 algorithmes d'apprentissage par renforcement que sont : SARSA, le Q-learning et l'acteur-critique.

L'AR est une méthode de programmation ne nécessitant que de spécifier les moments pendant lesquels punir ou récompenser l'agent. Il n'est nul besoin de lui indiquer quoi faire dans telle ou telle situation, l'agent se charge d'apprendre par lui-même en renforçant les actions optimales à effectuer dans une situation donnée afin de maximiser les récompenses. C'est une méthode d'apprentissage orientée objectif qui va conduire à un contrôleur optimal pour la tâche spécifiée par les récompenses. Il existe deux approches pour l'utilisation de la méthode d'apprentissage par renforcement [63]. La première consiste à discrétiser le problème afin de fournir des espaces d'états qui pourront être utilisés directement par des algorithmes utilisant des tableaux de qualités Q . L'utilisation est difficile et exige le stockage des valeurs de la fonction qualité pour tous les couples (état, action) [75]. Dans les problèmes discrets de faible dimension, on peut utiliser des tableaux: une ligne correspond aux qualités des différentes actions pour un état donné. Mais dans le cas des espaces

d'états et d'actions continus, le nombre de situation est infini et la représentation de la fonction Q par des tableaux est impossible. La deuxième approche va permettre de travailler directement dans les espaces d'états et d'actions continus, ceci en utilisant des méthodes d'approximation des fonctions comme les réseaux de neurones et les systèmes d'inférence flous qui offrent des solutions prometteuses pour ce problème et limite les effets de parasites qui pourraient apparaître suite à un mauvais choix des espaces état-action pour la discrétisation [78] [79] [80] [81].

Dans cette section, nous avons présenté les approches d'évitement d'obstacles par des systèmes mobiles, ensuite nous avons exposé deux approches de l'intelligence artificielle, la logique floue et l'apprentissage automatique. Nous avons constaté que le fonctionnement d'un contrôleur flou dépend d'un nombre important de paramètres qui doivent être déterminés lors de la conception. Ces paramètres s'influencent mutuellement, leur réglage devient peu aisé. Nous nous sommes tournés, par la suite, vers le domaine de l'apprentissage automatique qui est une solution possible pour l'ajustement et la génération des paramètres du contrôleur flou. Après avoir fait un exposé sur l'apprentissage par renforcement, nous avons vu que les algorithmes d'apprentissage par renforcement sont basés sur les processus markoviens et demandent de discrétiser l'environnement en couples (état-action). Pour parfaitement modéliser un problème réel, la taille de la matrice V ou Q est proportionnelle au nombre de combinaisons états-actions possibles ce qui posait le problème de stockage d'autant de données sur un système embarqué. De plus l'apprentissage serait extrêmement long, puisqu'il demanderait à l'agent de visiter plusieurs fois chaque combinaison état-action possible. Une des approches de l'apprentissage par renforcement permettant de travailler directement dans les espaces d'états et d'actions continus, ceci en utilisant des méthodes d'approximation des fonctions comme les réseaux de neurones et les systèmes d'inférence flous, nous avons opté, dans le cadre de notre projet, de concevoir un contrôleur flou entraîné par renforcement pour la commande automatique de navigation et d'évitement d'obstacles (piétons) de notre système. Nous allons tout particulièrement utiliser la méthode de Takagi-Sugeno d'ordre 0 comme modèle de raisonnement flou, et l'algorithme Q-learning d'apprentissage présenté à l'**annexe 2**, pour mettre à jour notre fonction valeur d'état et la fonction valeur d'action.

Cette partie est divisée en deux chapitres; dans le premier chapitre, nous avons effectué une présentation générale des ADAS. Après avoir défini les notions de véhicule de tourisme et de véhicule autonome, nous avons vu que la plupart des ADAS sont des systèmes de sécurité actifs dont certains étaient employés pour la contrôlabilité du véhicule et d'autres pour l'évitement de collision. Ensuite, nous avons fait une étude sur les différentes méthodologies de conception de système embarqué. Nous nous sommes principalement penchés sur quatre méthodes : l'analyse fonctionnelle, l'analyse de la valeur, la Quality Function Deployment (QFD) et la méthode TRIZ. Le deuxième chapitre est consacré aux méthodes d'identification de piétons. Nous avons présenté dans ce chapitre les différentes technologies de capteurs pour l'identification de piétons; par la suite, une revue de littérature des méthodes d'identification des piétons par vision a été présentée. Ce chapitre se termine par une exposition des approches d'évitement d'obstacles applicables à un véhicule. Nous nous sommes focalisés sur les algorithmes d'évitement d'obstacles réactifs à base de la logique floue et de l'apprentissage automatique.

Dans la deuxième partie, nous présenterons notre système, se sera l'occasion pour nous d'effectuer l'application des différentes méthodologies issues de notre revue de littérature à notre problématique.

PARTIE 2: MATERIELS ET METHODES

Dans cette partie, nous présentons notre contribution dans le domaine de la conception et de la réalisation d'un système d'aide à la conduite (anticollision). À l'opposé de la partie précédente, cette partie concerne un domaine plus expérimental, la conceptualisation du système tel que défini dans le cahier de charges établi (voir **annexe 9**). La méthodologie employée ici consiste à subdiviser la problématique en bloc, concevoir une solution ou une preuve de faisabilité et la réaliser avec efficacité de manière à pouvoir analyser les résultats à la fin. Le succès d'un projet expérimental dépend de plusieurs facteurs. D'abord, il faut que le problème ou l'objectif soit choisi avec soin. Le facteur temps pour prototyper une solution est important : s'il faut six mois pour prototyper une solution, il est intéressant de le faire ; s'il faut trois ans, le problème peut être dépassé au moment où le prototype est terminé. Un moyen pour réduire ce temps consiste à réutiliser le travail des projets précédents ou une infrastructure existante. Cette partie est structurée en deux chapitres ; le premier fait une présentation succincte de notre système embarqué ainsi que le système de multiplexage adopté. Ensuite, une étude proprement dit de la conception du système est faite en faisant une application des différentes méthodologies présentées et adoptées dans la partie précédente à la réalisation de notre système. Le deuxième chapitre fait une description de caractéristiques matérielles des capteurs qui seront utilisés dans notre système, ainsi qu'une définition des différents outils logiciels avec leurs rôles que nous utiliserons afin de simuler et tester notre système.

CHAPITRE 3: ETUDE DE CONCEPTION

3.1 Présentation du système

L'architecture globale du système, telle que illustrée par la **figure 3.1**, comprend trois composantes principales: le Bloc de Perception (BP), le Bloc de Décision (BD) et le Bloc de Commande (BC). La fréquence de fonctionnement dépend de la fréquence des capteurs utilisés et de l'architecture de traitement choisie pour le fonctionnement du BP. Cet aspect est discuté dans la section suivante. A chaque itération générale, les tâches effectuées par chaque composante sont globalement séquentielles, selon le flux décrit par les flèches du schéma. Toutefois, de nombreuses tâches internes sont indépendantes et sont traitées en parallèle.

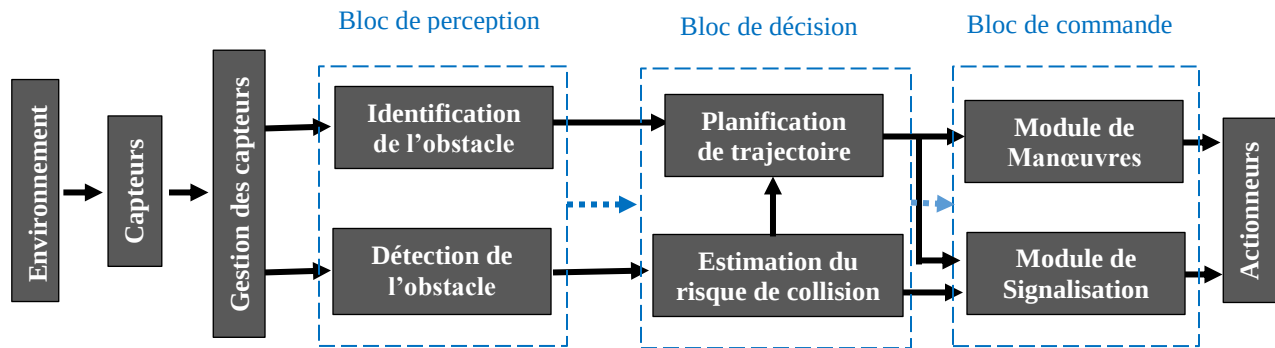


Figure 3.1 : Représentation schématique du système

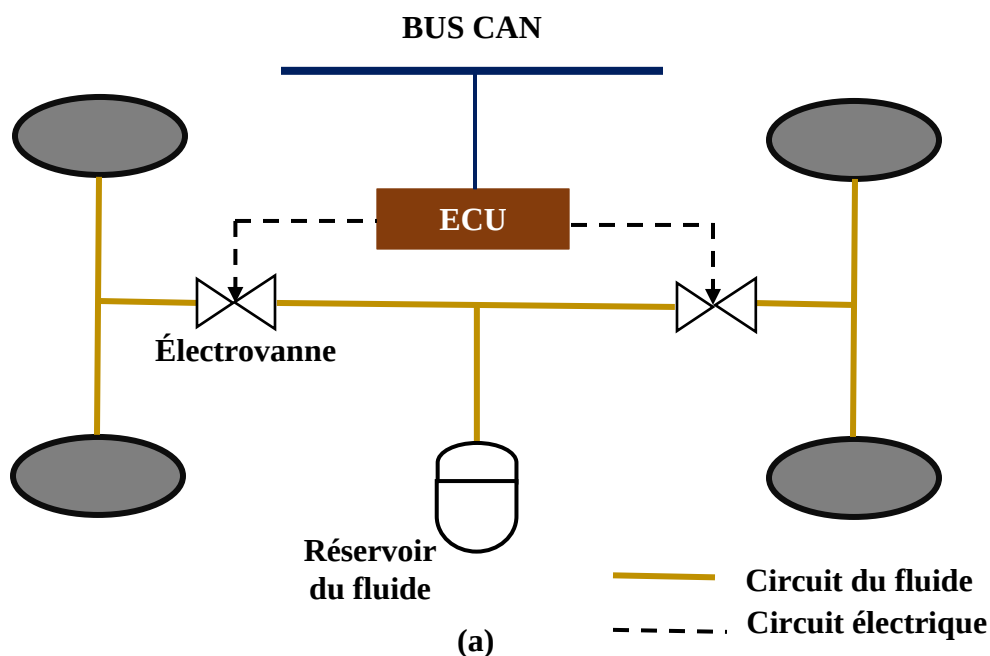
3.1.1 Présentation des entrées/sorties du système

Considérons plus en détails la **figure 3.1**, les données d'entrée du système global correspondent aux données d'entrée du BP et ne sont autres que les sorties des capteurs du système. Nous ignorons ici, les algorithmes et les paramètres des filtres mis en œuvre dans les différents capteurs que nous utiliserons. Les données fournies par les capteurs sont matériellement récupérées à travers le bus CAN du véhicule hôte sous forme de trames⁵. Ces données doivent être traitées, afin de reconstituer la liste de pistes des capteurs sous une forme exploitable. Un aspect fondamental voulu pour le système est

⁵ Trame ou protocole CAN de données est un protocole de communication série qui supporte des systèmes temps réel avec un haut niveau de fiabilité. Il s'agit en quelque sorte du langage utilisé pour la transmission de donnée.

qu'il doit être assez flexible pour permettre l'interchangeabilité des capteurs à la plateforme sans nécessiter des modifications profondes. Ceci permet la tolérance aux pannes de capteurs et l'adaptation à des véhicules équipés différemment. Les performances seront inévitablement conditionnées par les capteurs présents. La gestion matérielle des capteurs est présentée dans la section suivante. Les sorties du BP sont des pistes globales représentant des caractéristiques sur les obstacles identifiés par le système (distance et nature).

Les données de sorties du système global correspondent aux données de sortie du Bloc de Commande. Ces dernières n'étant autres que les signaux générés par le module de Manœuvres (MOMAN) et le module de signalisation (MOSIG). Ces signaux sont utilisés comme des entrées des actionneurs sur lesquels notre système va agir, les actionneurs du MOSIG pouvant être des leds, des écrans, des alarmes ou des vibrateurs. Pour ce qui est du MOMAN, une considération doit être prise en compte à ce niveau de conception. En effet, le système étant embarqué dans un véhicule, et la manœuvre d'évitement s'opérant de manière autonome, il s'avère indispensable que l'architecture matérielle du véhicule puisse permettre cette automatisation. Pour cela, nous avons été contraints de réduire l'applicabilité de notre système aux véhicules dotés d'une direction assistée électriquement (*Steer by wire*) et d'un système de freinage hybride (*Brake by wire*) [82] [83]. Les **figure 3.2 (a)** et **figure 3.2 (b)** illustrent le principe de fonctionnement de ces deux systèmes.



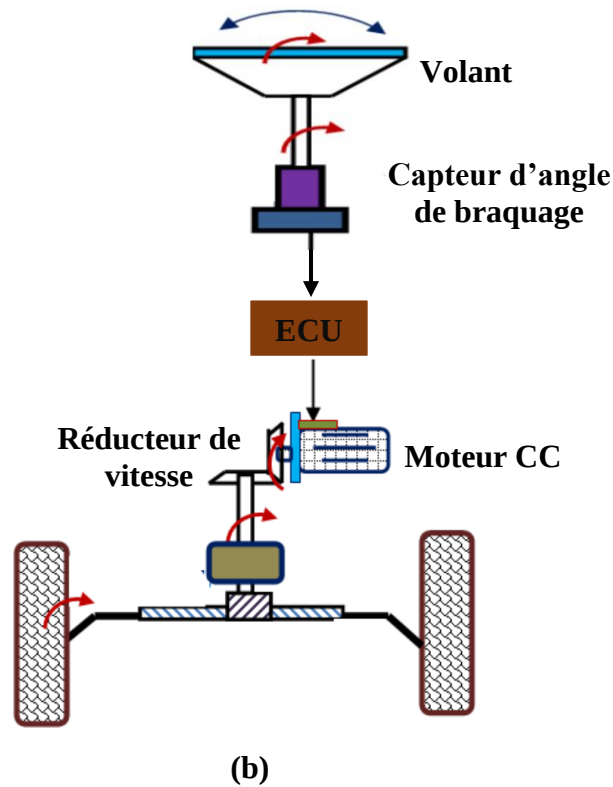


Figure 3.2 : Illustration de l'architecture de commande des systèmes (a) « brake by wire » et (b) « steer by wire »

3.1.2 Gestion des capteurs

Les capteurs utilisés ont chacun un ECU (Electronic Control Unit), un calculateur dédié qui effectue le traitement des données brutes du capteur. Chaque ECU est un esclave du bus LIN (Local Interconnect Network). Un ECU Central remplit les fonctions de LIN maître et commande les ECU esclaves situés sur son réseau LIN. L'ECU Central assure la passerelle entre les protocoles LIN et CAN, il encode les pistes obtenues par ses esclaves dans une série de trames, ces dernières sont ensuite émises sur l'un des bus CAN du véhicule [7]. Deux modules centraux, qui ne sont autres que le module de manœuvres et le module de signalisation, sont connectés aux bus CAN et permettent de récupérer les trames émises par les divers ECU dédiés et celles contenant les données proprioceptives⁶ du véhicule hôte. La **figure 3.3** illustre cette architecture ainsi décrite. Au vu du grand nombre de données qui devaient être utilisées par notre système, il était indispensable pour nous d'utiliser des protocoles

⁶ Données proprioceptives : ce sont des mesures qui désignent la perception des paramètres internes au véhicule comme les déplacements angulaires des roues, la vitesse, etc...

de communication pour le transfert des données temps réels, raison pour laquelle le protocole CAN et le protocole LIN ont été retenus. Le choix du protocole de communication LIN se justifie par le fait qu'il soit fiable et économique [84]. Il utilise un réseau en bus monofilaire et est utilisé en automobile comme sous-réseau du bus CAN pour gérer des capteurs et des micro-actionneurs [85]. Le protocole CAN quant à lui a été choisi pour sa large utilisation dans la plupart des véhicules aujourd'hui. Pour permettre l'interchangeabilité des capteurs à la plateforme sans nécessiter de profonds changements, il est nécessaire d'harmoniser les données bas niveau (proche des capteurs) afin de conserver la même structure pour les traitements de niveaux plus hauts. Les travaux de HOUENOU [7] donnent plus ample détail à ce sujet. Dans ce mémoire, nous n'aborderons pas la question d'harmonisation des données, la communication à travers les bus LIN et CAN des différents éléments cités plus haut est supposée effective.

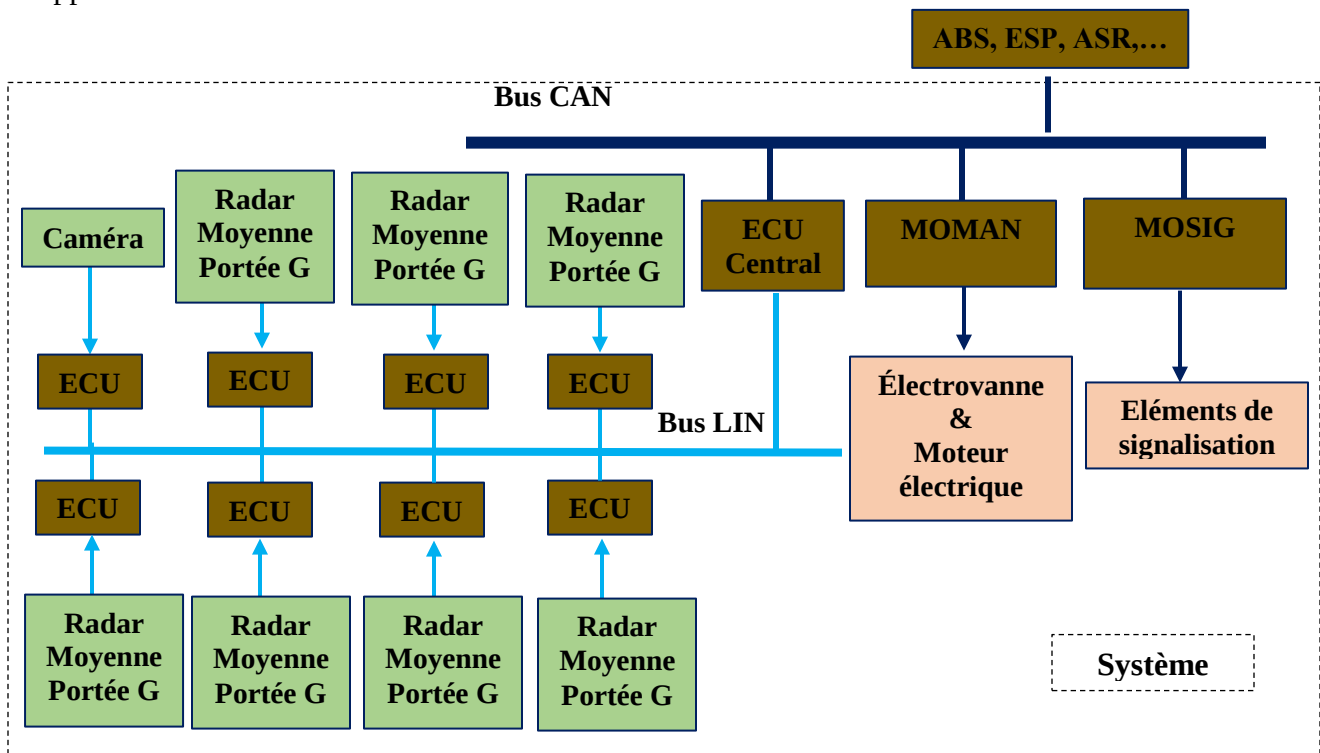


Figure 3.3: Architecture de communication du système (multiplexage)

Dans cette section, nous avons présenté notre système d'anticollision de piéton dans sa globalité en ressortant chaque composant principal ainsi que son rôle de façon macroscopique. Le Bloc de Perception permet d'obtenir une représentation en temps-réel de l'environnement dans lequel évolue le véhicule. Le Bloc de Décision analyse cet environnement afin d'évaluer le risque de collision vis-à-vis des piétons et détermine une trajectoire d'évitement de collision lorsque le risque de collision

est important. Enfin, le Bloc de Commande sert d'interface entre le Bloc de Décision et les différents actionneurs ou pré-actionneurs de notre système. Les flux de données entre les entrées du système et ses sorties ont été expliqués et nous avons vu l'architecture de communication de notre système. Dans la section suivante, nous allons aborder plus en détail le rôle et le fonctionnement de chaque bloc.

3.2 Conception du système

L'Agence Française de Normalisation (AFNOR) dans sa norme X50-170 de 2002, définit la conception comme étant un processus qui définit pas à pas le produit qui doit répondre aux besoins et aux attentes, par des choix successifs portant sur des points de plus en plus détaillés [21]. Ainsi, l'activité de conception comporte deux phases : la définition puis le développement d'un produit. Le produit (ou système) doit répondre à un ensemble de spécifications décrites dans un cahier des charges : spécifications techniques et fonctionnelles. Dans cette section, nous mettons en évidence le côté conceptuel de notre système qui constitue une étape fondamentale qui précède l'implémentation. Il s'agira de définir précisément chaque bloc de notre système, nous nous baserons pour cela sur les différentes méthodologies présentées dans le **chapitre 2**.

3.2.1 Conception du bloc de perception (BP)

3.2.1.1 Description du BP

Le Bloc de Perception de l'environnement du véhicule fournit les informations permettant de comprendre la situation de conduite. Les tâches effectuées incluent la détection des obstacles ainsi que leur identification. L'architecture du système doit permettre l'interchangeabilité des capteurs afin de s'adapter à divers niveaux d'équipement. Dans le contexte de ce mémoire, tous les capteurs utilisés sont chacun équipés d'un ECU qui effectue le suivi des objets détectés dans leur champ de perception et fournit la liste des pistes locales obtenues. Le rôle du BP est de fusionner et traiter les données des différents ECU dédiés afin de fournir, via le bus CAN, les données numérisées qui serviront d'entrée pour le bloc de décision. Il s'agit donc d'une fusion de données multi-capteur. Ce type de fusion de données touche à plusieurs problématiques. Notamment, la question de l'asynchronisme des capteurs. En effet, la fusion de plusieurs données n'a de sens que si ces données sont relatives au même instant. Puisque les capteurs sont indépendants et ont des fréquences de fonctionnement différentes, il est nécessaire de définir une stratégie adéquate. D'autre part, puisque les capteurs sont susceptibles de produire des pistes fantômes et d'avoir des non-détections, un module de gestion de la confiance en

l'existence des pistes globales a été développé par HOUENOU [7] comme solution pour assurer l'intégrité du système par rapport à la scène réelle. Dans le cadre de notre projet, la détection des obstacles se fera à l'aide des radars, très peu dépendant des conditions météorologiques, et l'identification de la nature des obstacles se fera à partir d'une caméra. Les caractéristiques des capteurs, leurs nombres ainsi que leur emplacement exact seront abordés plus en détails dans la sous-section suivante. Les travaux de HOUENOU [7] sur la provenance des obstacles heurtés par un véhicule, illustrée par la **figure 3.4** ci-dessous, nous ont inspiré dans la conception de l'architecture matérielle du BP.

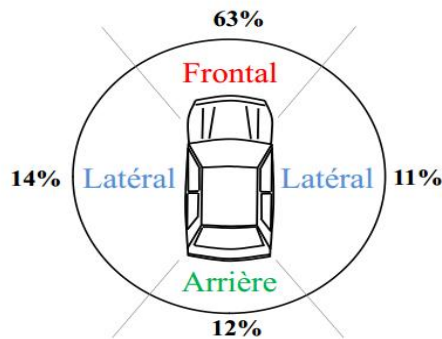


Figure 3.4 : Provenance de l'obstacle heurté par un véhicule [7]

3.2.1.2 Architecture matériel du BP

Les capteurs sont des composants de la chaîne d'acquisition d'un système embarqué, ils récupèrent des informations brutes sur leur environnement. Les informations récupérées sont envoyées à des unités de traitement qui les analysent. L'objectif étant de fournir au BD suffisamment d'information lui permettant d'analyser chaque situation, les caractéristiques, ainsi que l'emplacement de ces capteurs doivent être choisis avec soin. La **figure 3.5** ci-dessous présente l'architecture matérielle utilisée dans le cadre de notre projet. Trois zones ont été définies : la zone frontale, la zone latérale gauche et la zone latérale droite. Pour la première zone, qui est celle présentant le plus grand risque de collision, trois radars et une caméra seront utilisés. Les trois radars sont répartis de manière à couvrir une zone de vision s'étalant sur 170°. Tout particulièrement, pour le radar placé tout au centre, il doit avoir une portée plus grande, allant jusqu'à 150m. La caméra sera fixée au centre du véhicule. Les zones latérales droite et gauche ne nécessitent pas de radar ayant une grande portée, leur objectif étant de détecter des obstacles très rapprochés du véhicule qui pourrait gêner les manœuvres d'évitement, quatre radars à portée courte y seront utilisés, deux par côté.

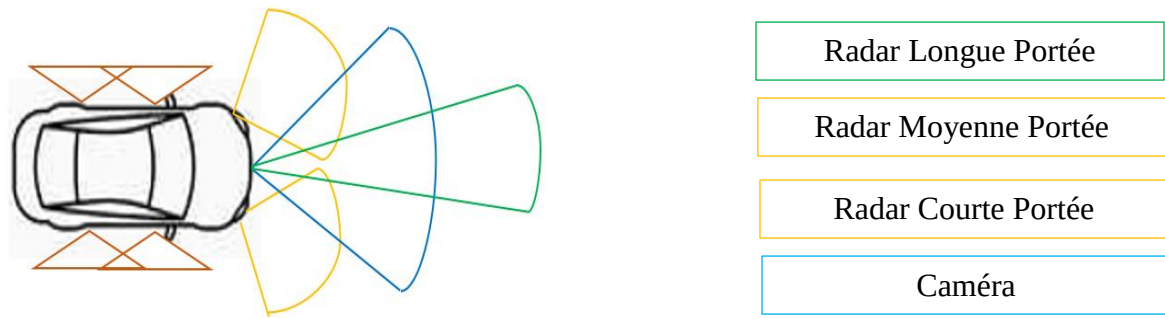


Figure 3.5 : Architecture matérielle du BP

3.2.1.3 Application de la méthode d'identification de piéton

Nous avons vu dans le **chapitre 2**, que la procédure d'identification d'un piéton dans une image suit quatre étapes : prétraitement de l'image, la délimitation des régions, l'extraction des fonctionnalités et la classification des régions.

3.2.1.3.1 Prétraitement de l'image

Comme nous l'avons mentionné précédemment, l'acquisition des images de l'environnement du véhicule se fera à partir d'une caméra. Chaque image obtenue est traitée par un ECU. L'ECU analyse à l'aide des méthodes HOG et SVM, chaque image, et génère à la sortie, la probabilité que l'image contienne un piéton ou pas. La première tâche du traitement de l'image est d'extraire un patch de l'image fournie par la caméra. Nous avons opté pour notre projet, pour un rapport d'aspect de 1:2.

3.2.1.3.2 Délimitation des régions

La délimitation des régions sur une image se fait à l'aide du descripteur HOG. Cette délimitation dépend de l'intervalle d'orientation du gradient. En général, il existe deux approches pour le choix de l'intervalle d'orientation du gradient : l'approche des gradients signés et l'approche des gradients non-signés. L'approche des gradients signés définit l'intervalle des angles d'orientation du gradient entre 0 et 2π radians. Utiliser l'approche des gradients signés permet de prendre en compte le sens du contour et donc la transition en luminosité. Néanmoins, selon les cas, le premier plan est plus clair que l'arrière-plan ou le contraire. Le sens de l'orientation du gradient n'est donc pas caractéristique de la classe des piétons. De plus, il est préférable d'avoir un descripteur de petite

dimension. Nous utiliserons donc l'approche des gradients non-signés où l'intervalle des angles d'orientation du gradient est compris entre 0 et π radians.

3.2.1.3.3 Extraction des fonctionnalités

L'extraction des fonctionnalités est l'opération qui permet de convertir l'image en un vecteur caractéristique. C'est ce vecteur caractéristique qui sera exploité par la suite, pour effectuer la classification des régions. L'extraction des fonctionnalités revient à calculer le vecteur de fonction HOG. Ceci passe par le calcul de l'histogramme de gradient par pixel. Nous utiliserons une valeur de $X = 8$ pixels, autrement dit nous allons diviser notre patch en 8×8 cellules, ceci parce que cela est suffisant pour extraire des fonctionnalités utiles [86], mais aussi afin de réduire, au minimum, la sensibilité de l'histogramme aux bruits.

Nous allons diviser l'espace généré par l'intervalle des angles d'orientation du gradient choisi précédemment en N_{bin} intervalles. Ces intervalles sont de mêmes tailles $\left(\frac{\pi}{N_{bin}}\right)$. L'ensemble Δ des intervalles d'orientation du gradient est défini par :

$$\Delta = \left\{ I_k = \left[\frac{2k-1}{2N_{bin}}\pi, \frac{2k+1}{2N_{bin}}\pi \right], k \in [1, N_{bin}] \right\} \quad (3-1)$$

Il existe deux possibilités pour déterminer le nombre de pixels concernés par chaque intervalle d'orientation :

- Soit les contours sont pris en compte. Dans ce cas, le nombre de pixels concernés par un intervalle est le nombre de pixels appartenant à un contour et dont l'orientation du gradient appartient à l'intervalle. Cette solution est cohérente avec l'utilisation des segments de contour et la recherche de silhouette.
- Soit le gradient est pris en compte. Dans ce cas, le nombre de pixels concernés par un intervalle est le nombre de pixels dont l'orientation du gradient appartient à l'intervalle pondéré par la magnitude du gradient. Cette solution permet une étude plus souple des variations de gradient.

Nous allons appliquer la dernière approche. Le décompte des pixels dont l'orientation du gradient appartient à un intervalle est binaire. Il est incrémenté de 1 si l'orientation appartient à l'intervalle et de 0 dans le cas contraire. Or, un pixel dont l'orientation du gradient est proche de la valeur moyenne de l'intervalle est plus caractéristique de celui-ci. Un pixel dont l'orientation du

gradient est éloignée de la valeur moyenne de l'intervalle est également caractéristique de l'intervalle voisin. Une interpolation est alors réalisée pour adoucir les contraintes du décompte. Soit $\theta \in \left[\frac{l\pi}{N_{bin}}, \frac{(l+1)\pi}{N_{bin}} \right]$ l'orientation du gradient d'un pixel de l'image, D_k le décompte des pixels pour l'intervalle I_k , le décompte pour chaque pixel de l'image d'orientation du gradient θ est réalisée par [55] :

$$\text{Pour tout } k \in [1, N_{bin}], D_k = \begin{cases} D_k + \frac{\theta - \frac{l\pi}{N_{bin}}}{\frac{\pi}{N_{bin}}} \text{ si } k = l \\ D_k + \frac{\frac{(l+1)\pi}{N_{bin}} - \theta}{\frac{\pi}{N_{bin}}} \text{ si } k = l + 1 \\ D_k \text{ sinon} \end{cases} \quad (3-2)$$

Ces décomptes correspondent aux fréquences d'occurrence des pixels de l'image dont l'orientation du gradient appartient à un intervalle $\Delta.V_{I_k}$, la fréquence d'apparition relative à l'intervalle I_k . L'histogramme de gradients orientés de la région est alors l'ensemble de ces fréquences pour tous les intervalles de Δ .

$$HOG = \{v_I^{region}, I \in \Delta\} \quad (3-3)$$

3.2.1.3.4 Classification des régions par la méthode SVM

SVM est une méthode qui repose sur l'existence d'un classificateur linéaire dans un espace approprié. Son principal avantage est qu'il peut être utilisé dans des espaces à grandes dimensions. Puisque c'est un problème de classification à deux classes (piétons et non piétons), cette méthode fait appel à un jeu de données d'apprentissage pour apprendre les paramètres du modèle (dans notre cas ceux d'un piéton). Le but de SVM est de trouver un classificateur qui va séparer les données et maximiser la distance entre ces deux classes. Ce classificateur est un classificateur linéaire appelé hyperplan. Les points les plus proches, qui seuls sont utilisés pour la détermination de l'hyperplan, sont appelés vecteurs de support. Il est évident qu'il existe une multitude d'hyperplan valide mais la propriété remarquable des SVM est que cet hyperplan doit être optimal. Nous allons donc, en plus de chercher les hyperplans valides, déterminer celui qui maximise la distance entre ces deux classes.

Soit m le nombre de données d'apprentissage, l'équation 3-4 représente la base de donnée d'apprentissage pour la classification. Les x_i sont les descriptions des images, ce sont des vecteurs de taille N_{bin} qui correspond à la taille des vecteurs caractéristiques définies dans la sous-section

précédente, y_i représente les labels associés : +1 ou -1. Les labels indiquent à quelle classe les données appartiennent.

$$\{(x_i, y_i), i = 1 \rightarrow m\} \quad (3-4)$$

La prediction de la classe d'une description d'image x est faite à partir de la fonction de décision f de l'équation 3-5 ; où w représente un vecteur de taille N_{bin} et b un coefficient appelé biais. x sera de la classe +1 si $(x) \geq 0$, et -1 sinon.

$$f(x) = \text{signe}(w.x + b) \quad (3-5)$$

L'hyperplan séparateur (frontière de décision) a donc pour équation $w.x + b = 0$. Nous attribuons, dans le cadre de notre projet, la classe +1 à celui des piétons, et la classe -1 aux non-piétons.

La distance $d(x)$ d'un point x au plan étant donnée par l'équation 3-6, l'hyperplan optimal est celui pour lequel la distance entre les vecteurs supports est maximale. Autrement dit, maximiser la marge revient à minimiser $\|w\|$.

$$d(x) = \frac{|w.x+b|}{\|w\|} \quad (3-6)$$

3.2.2 Conception du bloc de décision

3.2.2.1 Description du BD

Lorsqu'une collision s'annonce imminente, avec une probabilité assez élevée, le système doit proposer une manœuvre d'évitement de collision qui tient compte des données du BP afin d'éviter toute autre collision pendant la manœuvre. Dans l'optique ADAS, le rôle du système est d'alerter le conducteur, suffisamment tôt, en cas de danger afin qu'il réagisse lui-même. Le système n'intervient qu'en dernier recours lorsque le temps disponible ne permet pas de pouvoir éviter la collision plus tard. Il faut donc déterminer la manœuvre d'évitement de collision appropriée et le temps de réaction associé, qui définit le moment où la manœuvre doit commencer. En général, il s'agit de freiner avant d'atteindre l'obstacle ou d'effectuer un écart latéral pour le contourner. On peut sans doute imaginer d'autres scénarios, mais dans ce mémoire, nous nous sommes limités à ces deux cas. Le choix final se fait en fonction du temps de réaction disponible pour chaque cas de manœuvre. Nos travaux sur le BD sont détaillés dans les paragraphes suivants.

3.2.2.2 Contraintes mécaniques et cinématiques du véhicule

3.2.2.2.1 Contraintes mécaniques

Les contraintes mécaniques auxquelles est soumis notre système, se manifestent à travers l'angle de déviation du véhicule, angle matérialisant l'orientation du vecteur vitesse au centre de l'essieu avant par rapport à l'axe longitudinal du véhicule, et la pression du système de freinage. L'angle de déviation du véhicule correspond approximativement à l'angle de braquage $\varnothing$ des roues compris dans l'intervalle présenté dans l'équation 3.7 [87].

$$-78^{\circ} \leq \varnothing \leq 78^{\circ} \quad (3-7)$$

La pression maximale de freinage pouvant être appliquée à un véhicule est en moyenne de 160bar. Afin d'avoir une grande marge pour différent constructeur, nous fixerons la pression maximale à 170bar.

A très grande vitesse, les variations brusques de l'angle de déviation et de la pression de freinage (pour l'évitement d'obstacle) entraîneraient une perte de stabilité du véhicule, due aux forces centrifuges ou au blocage des roues (patinage). Dans le cadre de notre projet, nous ne traitons pas cette situation, du fait que cela soit corrigé par les systèmes d'ESP et d'ABS, systèmes préalables à toute implémentation de notre système dans un véhicule.

3.2.2.2.2 Contraintes cinématiques

La voiture la plus rapide au monde en 2017 a une vitesse maximale de 440 km/h [88] . Compte tenu de la difficulté d'atteindre de telle vitesse en conduite dû principalement aux infrastructures routières en Afrique subsaharienne et au code routier, nous avons défini la vitesse maximale que pourra atteindre un véhicule à 120km/h.

3.2.2.3 Modèle de calcul de l'estimation du risque de collision

3.2.2.3.1 Notion de risque de collision

Si le mouvement des obstacles prédits était absolument certain et si la forme géométrique des objets était parfaitement connue, l'éventualité d'une collision serait une valeur parfaitement binaire. En effet, il suffirait de vérifier sur l'horizon de prédiction, s'il existe au moins un instant où l'espace occupé par le véhicule recouvre celui d'un obstacle donné. On saurait donc de façon certaine, s'il y aura une collision et, si oui, à quel moment précis (à une période d'échantillonnage près). Mais ce

n'est pas le cas en pratique puisque, d'une part, les approches d'évitement d'obstacle ne donnent qu'une approximation de la future trajectoire et que, d'autre part, elles se basent sur des estimations d'état qui sont initialement sujettes à une incertitude ; les positions prédites sont par conséquent aussi incertaines. C'est pourquoi nous devons évaluer un risque de collision. La notion de risque de collision peut revêtir plusieurs aspects. On peut, par exemple, s'intéresser à un risque global de collision par rapport à tous les objets détectés, à la fois, ou bien chercher à évaluer individuellement le risque de collision pour chaque objet. D'autre part, de façon communément admise, le risque de collision est le produit de la probabilité de la collision annoncée et de sa gravité. La gravité dépend en particulier de la nature de l'objet ou des objets impliqués (piéton, deux-roues, voiture, poids-lourd etc.). Par exemple, on admet que la gravité est plus importante s'il s'agit d'une collision avec un piéton ou un cycliste que s'il s'agissait d'un autre véhicule. D'un point de vue ADAS, ce facteur sert surtout à pondérer les seuils qui définissent les niveaux d'alerte ou éventuellement les actions à effectuer, ce, en vue de préserver au maximum l'intégrité physique des personnes [7] [89].

Pour nos travaux, nous considérons qu'une collision implique un véhicule et un seul objet à la fois, ceci parce que l'objectif est de déterminer, dans les cas critiques, une manœuvre permettant d'éviter un obstacle particulier (principalement un piéton), sans bien sûr, causer d'autres collisions. La définition de la gravité peut très vite impliquer des notions liées à l'éthique et ne fait pas partie des objectifs de ce mémoire. L'évaluation de la gravité d'une collision n'est pas prise en compte dans nos travaux. Nous nous concentrons sur l'évaluation de la probabilité de collision et par ricochet, de la distance à partir de laquelle, cette probabilité est élevée.

3.2.2.3.2 Méthode proposée

D'un point de vue ADAS, l'objectif du calcul du risque de collision est de permettre la définition de plusieurs niveaux d'alerte correspondant chacun à des actions spécifiques. Nous nous sommes fixés comme contrainte, relativement aux challenges sur l'automatisation de la conduite exposés par NIDHI [90], de donner le maximum de temps possible au conducteur pour réagir face à un risque de collision. Pour cela, nous avons défini quatre niveaux d'alerte illustrés par la **figure 3.6** ci-dessous. Le premier niveau se manifeste par le déclenchement d'alarmes visuelles, sonores et sensibles, ceci dans le but d'éveiller le conducteur sur l'imminence d'un danger, elles se déclenchent à la distance d_a de l'obstacle. Nous avons évalué la distance d_a , en nous inspirant des travaux de GREEN [91], elle est fonction de la distance d_f et de deux constantes c_c et c_m dépendant respectivement du temps de réaction du conducteur et du temps de réaction du mécanisme de freinage

(**équation 3-8**). Le deuxième niveau consiste en la prise de contrôle du véhicule par le système, elle permet un actionnement automatique du système de freinage du véhicule et correspond au franchissement de la distance limite d_f permettant d'éviter un obstacle uniquement en freinant. Le calcul de la distance d_f est fait à partir de la formule de l'**équation 3-9** tiré des travaux de ECKERT [89]. Le troisième niveau est celui pour lequel, l'évitement de l'obstacle ne peut se faire qu'avec la combinaison du freinage et de la déviation du véhicule. Elle s'active après un certain seuil une fois que le niveau 2 ait été enclenché sans élimination du risque, ou lorsque l'obstacle apparaît subitement dans la zone frontale du véhicule à une distance inférieure à d_f . Le dernier niveau, qui est le plus critique, correspond à un risque de collision quasi inévitable. Il se manifeste lorsque la distance limite d_d permettant d'éviter un obstacle en déviant est atteinte. L'**équation 3-10** ci-dessous donne la formule permettant de calculer d_d .

$$d_a = d_f + c_c + c_m \quad (3-8)$$

$$d_f = \frac{1}{2a} V_{rel}^2 + c_m \quad (3-9)$$

$$d_d = \sqrt{\frac{2L}{a}} V_{rel} \quad (3-10)$$

$$V_{rel} = V_1 - V_2 \quad (3-11)$$

L : Décalage latéral

V_{rel} : Vitesse réelle du véhicule

V_1 : Vitesse du véhicule, V_2 : Vitesse de l'obstacle

a : Accélération du véhicule

c_c : Constante dépendant du temps de réaction du conducteur

c_m : Constante dépendant temps de réaction mécanisme

En tenant compte de la largeur moyenne des voies d'une route et de celle d'un véhicule, nous avons fixé la valeur du décalage latéral L à 3m. Aussi, en nous inspirant des travaux de HOUENOU [7], nous avons fixé l'accélération du véhicule, qui correspond en pratique dans notre cas à la décélération maximale possible de freinage, à une constante égale à $10m/s^2$. Un module de calcul (**figure 3.6**) effectuera l'estimation à partir des formules citées plus haut des différents paramètres.

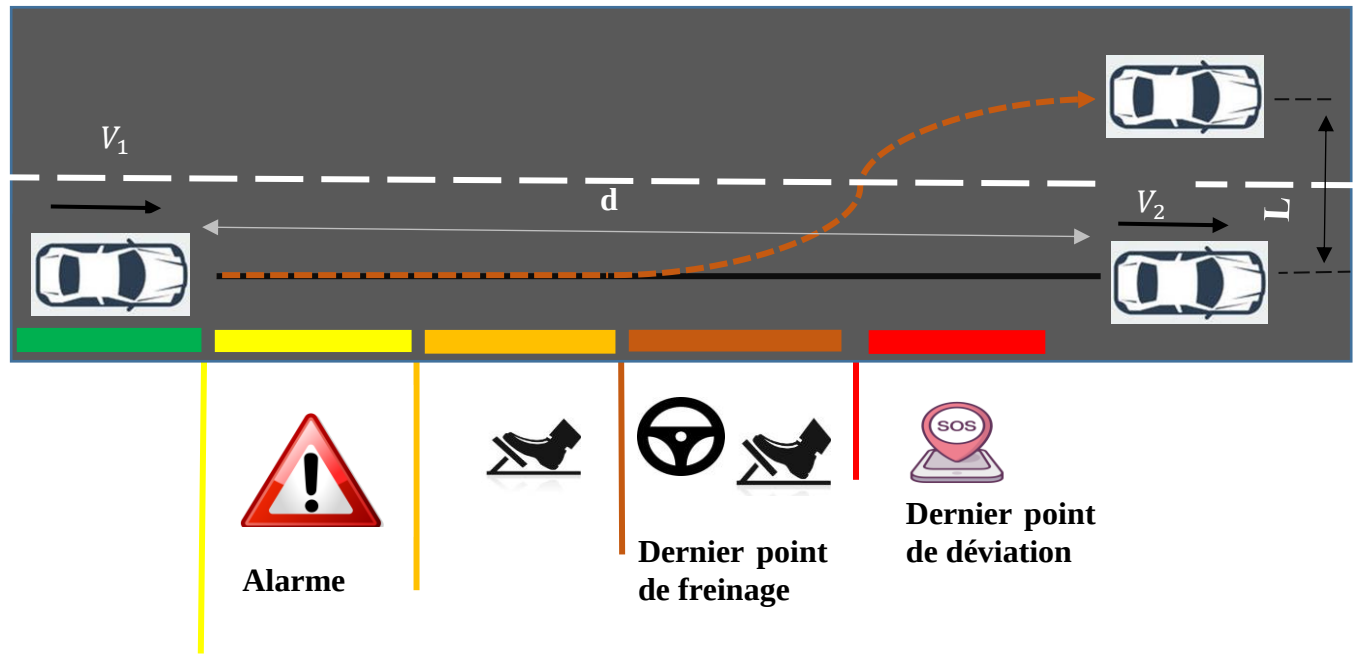


Figure 3.6 : Présentation des niveaux d'alerte et de leurs actions spécifiques

3.2.2.4 Conception des contrôleurs flous de planification de trajectoires

Comme on a vu dans le **chapitre 2**, la théorie de la logique floue est caractérisée par la capacité de modéliser et de traiter des informations incertaines et imprécises. Dans la majorité des applications de logique floue pour la navigation, le modèle mathématique de la dynamique de l'environnement du véhicule n'est pas nécessaire dans le processus de conception du contrôleur de mouvement puisque la logique floue est un outil mathématique puissant qui permet de manipuler le raisonnement humain, les concepts et les termes linguistiques. D'autre part, le système de navigation basé sur les comportements est une stratégie de commande qui utilise l'information acquise pour effectuer le choix d'une étape de raisonnement afin d'agir sur le monde externe. L'idée principale de la navigation basée sur les comportements est d'identifier des réponses différentes (comportements) selon les données sensorielles [63]. Dans le cadre de notre projet, nous avons défini deux contrôleurs flous (**figure 3.7**) en fonction de la tâche à accomplir. Un contrôleur flou pour l'évitement d'obstacle et un contrôleur flou pour l'alignement droit du véhicule. Le premier contrôleur permet au système, de prendre une décision face à un risque de collision, il s'appuiera pour cela sur l'algorithme de Q-learning flou. Le deuxième contrôleur lui, s'enclenche une fois que la collision a été évitée, il a pour

objectif de replacer le véhicule à une position droite afin de permettre aisément au conducteur de reprendre la main. Il sera réalisé à partir d'un algorithme de logique floue. Dans les deux cas, tel que précisé dans le **chapitre 2**, un SIF de type TS d'ordre zéro sera utilisé.

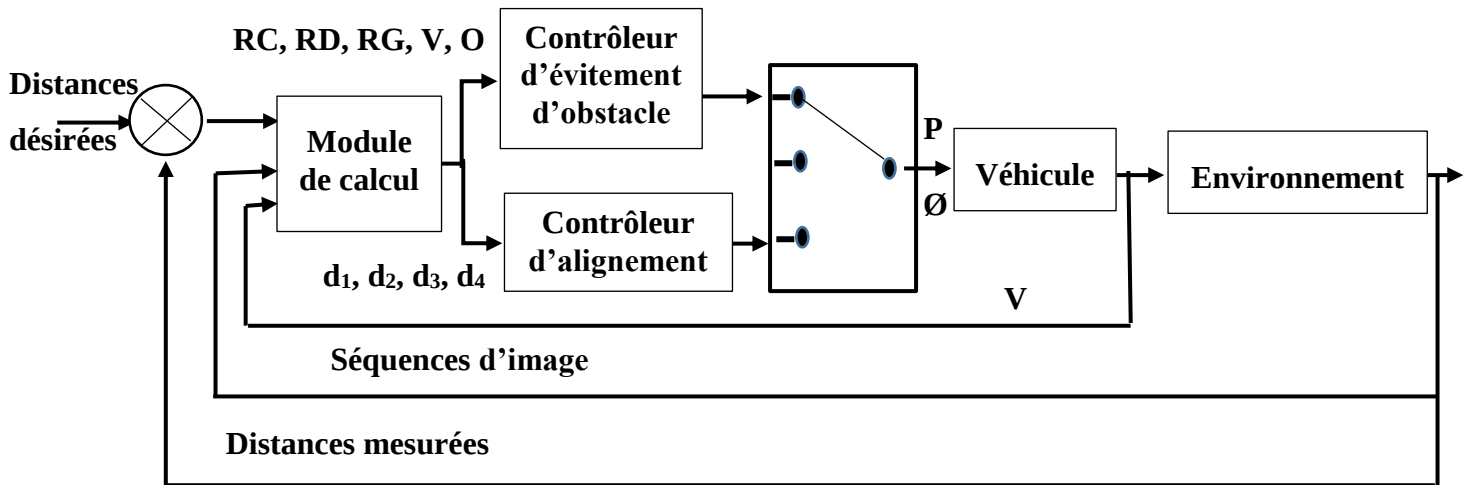


Figure 3.7 : Planificateur de trajectoire floue

3.2.2.4.1 Définition des paramètres flous

Il est question ici, de présenter les paramètres des variables linguistiques d'entrée et de sortie de notre système. Nous allons définir les différentes variables linguistiques que nous utiliserons pour la conception de nos contrôleurs flous, nous allons également effectuer le choix des méthodes des opérateurs ainsi que les fonctions d'appartenance. La **figure 3.8** ci-dessous est une illustration des variables linguistiques d'entrée de notre système.

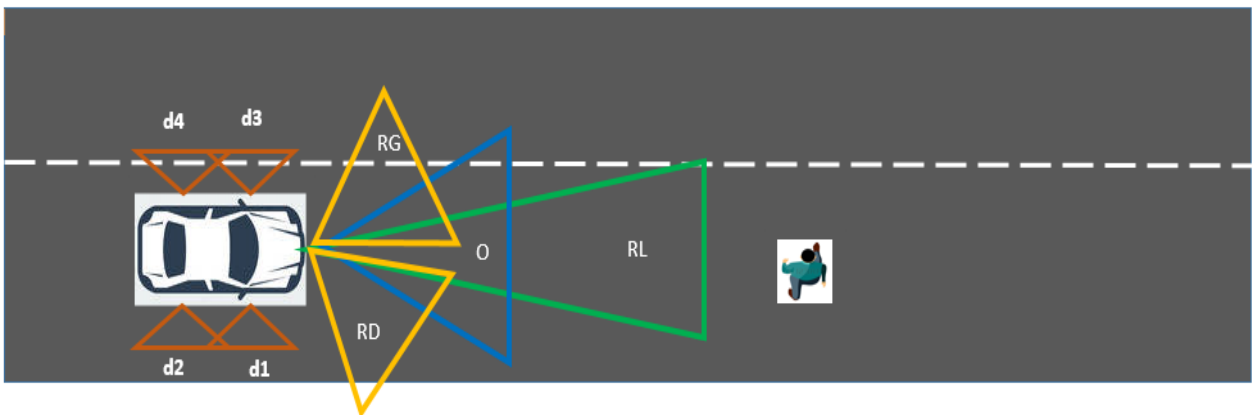


Figure 3.8 : Illustration des variables linguistiques d'entrée

a- Choix des méthodes des opérateurs

Dans le cadre de notre projet, nous utiliserons la méthode MAX pour l'opérateur OU et la méthode MIN pour l'opérateur ET.

b- Définition des variables linguistiques

Notre système est constitué de onze variables linguistiques :

❖ Variables linguistiques d'entrée

- La distance RC à l'obstacle mesurée par le radar longue portée exprimée en mètre.
- La distance RG à l'obstacle mesurée par le radar moyenne portée gauche exprimée en mètre.
- La distance RD à l'obstacle mesurée par le radar moyenne portée droit exprimée en mètre.
- La distance d1 à l'obstacle mesurée par le radar courte portée avant droit exprimée en mètre.
- La distance d2 à l'obstacle mesurée par le radar courte portée arrière droit exprimée en mètre
- La distance d3 à l'obstacle mesurée par le radar courte portée avant gauche exprimée en mètre
- La distance d4 à l'obstacle mesurée par le radar courte portée arrière gauche exprimée en mètre
- La vitesse V du véhicule mesurée par les capteurs proprioceptifs du véhicule exprimée en Km/h.
- La nature O des obstacles, déterminée par la probabilité de présence d'un piéton ou non.

❖ Variables linguistiques de sortie

- L'angle Ø de déviation des roues du véhicule exprimé en degré.
- La pression P de freinage des roues exprimée en Bar.

c- Paramètres des variables linguistiques

Les paramètres des variables linguistiques ici sont l'ensemble des valeurs linguistiques, l'univers de discours et les fonctions d'appartenance. Les tableaux suivants présentent ces paramètres pour chacune de nos variables linguistiques.

Tableau 3-1 : Paramètres des distances RC, RG et RD

Valeurs linguistiques	Plage de la fonction d'appartenance (m)	Fonction d'appartenance
Petite	0; 0; 2; 5	Trapézoïdale
Moyenne	4.5; 15; 20; 25	Trapézoïdale
Grande	24; 30; 60; 60	Trapézoïdale

Tableau 3-2 : Paramètres des distances d1, d2, d3, d4

Valeurs linguistiques	Plage de la fonction d'appartenance (m)	Fonction d'appartenance
Petite	0 ; 0 ; 1 ; 1,5	Trapézoïdale
Moyenne	1 ; 2 ; 3 ; 4	Trapézoïdale
Grande	3.5 ; 5; 8; 8	Trapézoïdale

Tableau 3-3 : Paramètres de la vitesse V

Valeurs linguistiques	Plage de la fonction d'appartenance (km/h)	Fonction d'appartenance
Petite	10; 10; 20; 25	Trapézoïdale
Moyenne	24; 30; 40; 60	Trapézoïdale
Elevée	55; 70; 120; 120	Trapézoïdale

Tableau 3-4 : Paramètres de l'angle de déviation $\varnothing^7$

Valeurs linguistiques	Angle (°)
Négatif Moyen	-14
Négatif Petit	-7
Zéro	0
Positif Petit	7
Positif Moyen	14

Tableau 3-5 : Paramètres de la pression de freinage P

Valeurs linguistiques	Pression (Bar)
Petite	60
Moyenne	96
Forte	130

3.2.2.4.2 Contrôleur flou entraîné par renforcement d'évitement d'obstacle

Nous définissons ici un contrôleur flou entraîné par renforcement pour la tâche d'évitement d'obstacle. Plus particulièrement, l'algorithme Q-learning flou est utilisé pour optimiser les conclusions des règles floues. Durant l'apprentissage, la conclusion de chaque règle est choisie au moyen de la politique d'exploration-exploitation de donnée e-gloutonne.

L'ensemble des règles du contrôleur flou d'évitement d'obstacle est présenté dans le **tableau 3-7** ci-dessous. P, G, M, E, NM, NP, Z, PP, PM, et F représentent respectivement dans ce tableau, les valeurs linguistiques petite, grande, moyenne, élevée, négatif moyen, négatif petit, zéro, positif petit, positif moyen et forte.

⁷ Le signe de l'angle de déviation est fonction du sens de la déviation. Pour une déviation dans le sens de rotation des aiguilles d'une montre, l'angle sera considéré négatif, positif dans le sens contraire.

Tableau 3-6 : Table des règles du contrôleur flou d'évitement d'obstacle

		Entrées				Sorties	
Cas		RG	RC	RD	V	Ø	P
1		P			M	NP	M
		P			E	NM	F
2				P	M	PP	M
				P	E	PM	F
3			G		E	Z	M
			M		M	Z	M
			M		E	Z	F
			P		P	Z	M
			P		M	PP	F
			P		E	PM	F
4		P	P	P	M	Z	F
		P	P	P	E	Z	F

Le renforcement d'un SIF par un algorithme d'apprentissage par renforcement consiste à approcher la fonction qualité Q (voir **Annexe 2**) par la fonction SIF suivant l'équation 3-12. Le principe de cette optimisation consiste à proposer plusieurs conclusions pour chaque règle et à associer à chaque conclusion une valeur qualité qui sera mise à jour incrémentalement au cours du temps. Il s'agit alors de trouver la conclusion appropriée pour chaque règle. Le processus d'apprentissage permet de déterminer l'ensemble des règles maximisant les renforcements futurs. Cette version floue du Q-learning est appelée Q-learning flou.

$$s \rightarrow y = Q = SIF(s) \quad (3-12)$$

Dans le cadre de notre projet, nous avons défini pour chaque règle 3 conclusions, soit 84 au total pour les 28 règles de notre contrôleur flou d'évitement d'obstacle (voir **tableau 3-7**). Après chaque phase d'apprentissage, le contrôleur choisi pour chaque règle la conclusion correspondante à la meilleur qualité $q[i, j]_{j=1}^{45}$.

La base des règles est améliorée en ligne en utilisant un signal de renforcement r définit par :

$$r = \begin{cases} -4, & P, \emptyset \text{ diminue et } V \text{ augmente} \\ -2, & Ri < 8,5, V > 50 \text{ et } \emptyset = 0, i = G, L, D \\ 4, & Ri > 40 \text{ } P \text{ augmente et } \emptyset = 0, i = G, L, D \\ 0, & \text{ailleurs} \end{cases}$$

Ce signal va servir pour déterminer la meilleure interprétation numérique des termes linguistiques utilisé en proposant trois interprétations pour chaque label de sortie (angle de braquage et pression de freinage). Les **Tableaux 3-8** et **3-9** ci-dessous présentent les 3 conclusions proposées pour les 7 valeurs linguistiques de sortie de notre contrôleur flou d'évitement d'obstacle.

Tableau 3-7 : Conclusions des règles proposées pour les valeurs linguistiques de sortie de la variable angle de déviation $\emptyset$

Valeurs linguistiques	Angle (°)
Négatif Moyen	-16;-14;-12
Négatif Petit	-9;-7;-5
Zéro	-2;0;-2
Positif Petit	5;7;9
Positif Moyen	12;14;16

Tableau 3-8 : Conclusions des règles proposées pour les valeurs linguistiques de sortie de la variable pression de freinage P .

Valeurs linguistiques	Pression (Bar)
Petite	55;60;65
Moyenne	86;96;106
Forte	120;130;140

3.2.2.4.3 Contrôleur flou d'alignement

Ce contrôleur permet de réaliser l'action d'« alignement droit » du véhicule à partir de la connaissance de sa position courante une fois l'obstacle évité (c'est-à-dire, faire aligner le véhicule droit sur la route). Les manœuvres permettant de replacer le véhicule sont effectués par l'intermédiaire des données issues des radars courte portée. Le contrôleur flou de remplacement utilise quatre variables (d_1 , d_2 , d_3 , d_4) pour générer les commandes de déviation $\emptyset$ permettant de replacer le véhicule dans la bonne direction. La stratégie de navigation réalisée ici est basée sur le principe de l'équidistance d'un triangle isocèle (**Figure 3.9**).

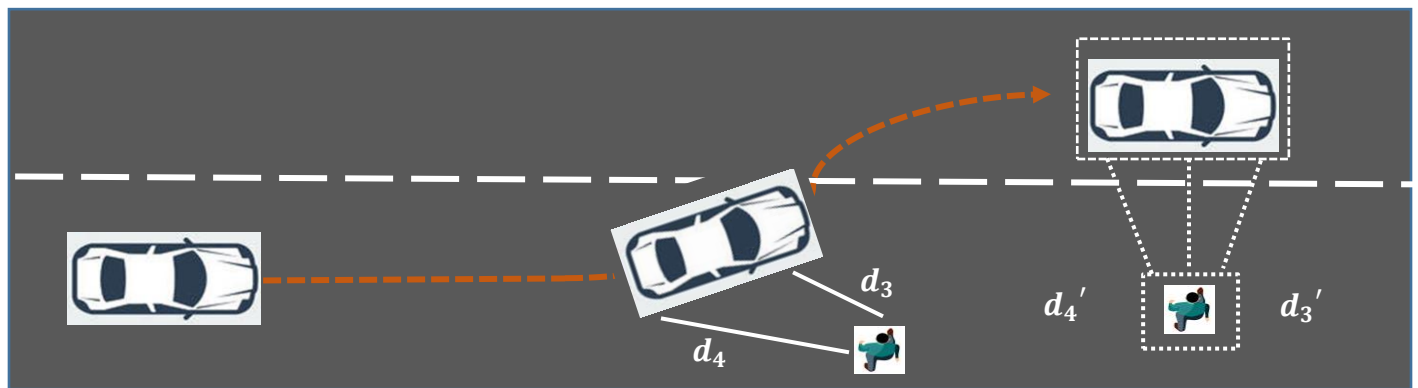
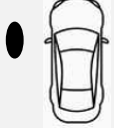
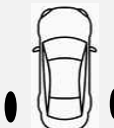


Figure 3.9 : Illustration de l'orientation du véhicule à travers le contrôleur flou de remplacement

Le **Tableau 3-10** ci-dessous présente les règles proposées pour le contrôleur flou d'alignement.

Tableau 3-9 : Règles du contrôleur flou d'alignement

		Entrées				Sorties
Cas		d1	d2	d3	d4	Ø
1		P	M			NM
		P	P			Z
		M	M			Z
		M	P			PM
2				P	M	PM
				P	P	Z
				M	P	NM
				M	M	Z
3		P		P		Z
		M		M		Z
4			P		P	Z
			M		M	Z
5		M	M	M	M	Z
		G	G	G	G	Z

3.2.3 Conception bloc de commande

Une fois le risque de collision évalué et la manœuvre d'évitement planifiée par le BD, le système doit agir par l'émission de commandes qui permettront d'appliquer les décisions prises par le BD : c'est le rôle du bloc de commande. Ces commandes permettront d'actionner l'électrovanne du

système de freinage hybride et/ou le moteur électrique du système de direction assisté électriquement, et de générer des signaux pour alerter le conducteur. Nos travaux sur le BC sont brièvement détaillés dans cette section car ils n'ont pas encore faits l'objet de résultats validés et feront partir des perspectives pour la suite de nos travaux.

L'architecture matérielle des véhicules compatibles, telle que spécifier dans la **section 3.1**, étant dotée de modules dédiés aux commandes du système de direction et du système de freinage, il sera surtout question, pour le Module de Manœuvres, de générer les trames de données, dans le bus CAN, correspondant aux commandes pour actionner les systèmes de direction et de freinage à travers leurs ECUs respectifs. Les signaux de commande générés par le MOMAN substituent ceux venant du conducteur à travers les capteurs d'angle de braquage et de pédale de frein. La **Figure 3.10** ci-dessous illustre cette architecture de commande pour le cas du système de direction.

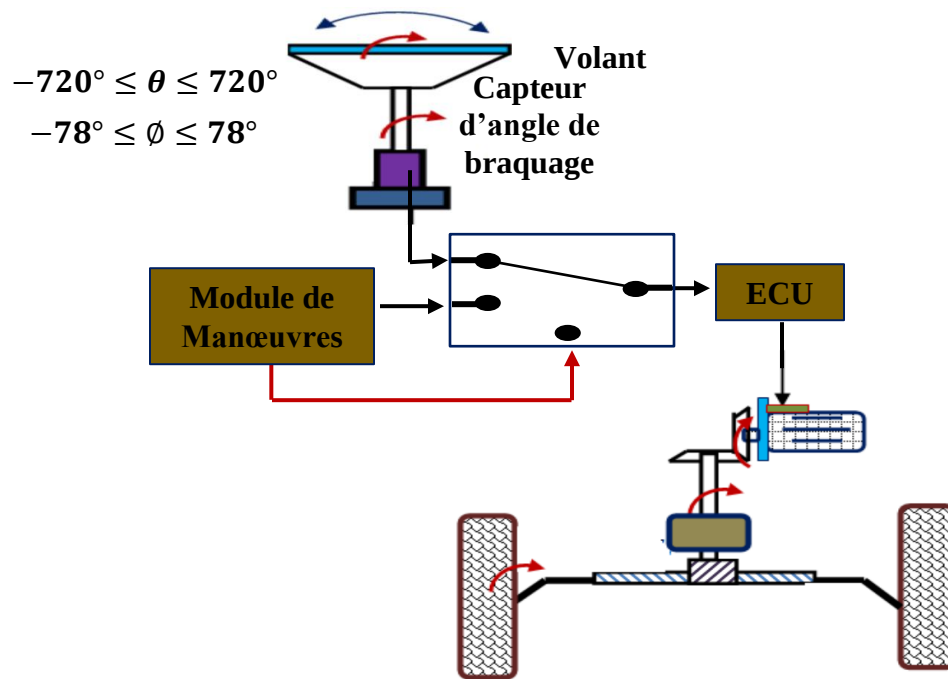


Figure 3.10 : Illustration de la commande du système de direction par le module de Manœuvres

Le module de signalisation (MOSIG) génère les signaux de commande qui sont utilisés pour actionner les éléments de signalisation (GLCD, Leds, alarmes sonores, vibrateurs, module GSM ...). Ces éléments pouvant être commandés directement ou à travers des pré-actionneurs.

Dans ce chapitre, nous avons défini et détaillé de façon précise et concise le rôle et le fonctionnement de chaque Bloc de notre système. Le Bloc de Perception est le bloc d'entrée de notre système, c'est lui qui, à partir des informations obtenues et traitées à partir des capteurs (externes et

proprioceptifs au véhicule), fournit les informations permettant de comprendre la situation de conduite. Ceci passe par la détection et l'identification des différents obstacles. L'application des méthodes HOG et SVM est faite pour l'identification de la nature des obstacles. Le Bloc de Décision quant à lui représente l'intelligence du système, c'est lui qui, sur la base des données reçues du BP, définit le comportement à adopter de façon autonome face à un risque de collision. Il utilise pour cela deux contrôleurs flous, un servant à la tâche d'évitement de l'obstacle, et l'autre pour replacer le véhicule après l'évitement. Quatre niveaux de risques ont été développés. Le Bloc de Commande enfin a pour rôle d'émettre les commandes qui permettront d'appliquer les décisions prises par le BD. Il se sert pour cela de deux modules, l'un, le MOMAN pour les manœuvres d'évitement ou d'alignement du véhicule, et le MOSIG pour l'actionnement des éléments de signalisation.

CHAPITRE 4 : OUTILS ET MATERIELS

Dans ce chapitre, nous décrivons les différents outils logiciels que nous avons exploités pour effectuer les simulations de notre système. La deuxième section de ce chapitre présente les caractéristiques des matériels qui correspondent à nos choix de conception.

4.1 Description des outils

4.1.1 Environnement Matlab

MATLAB (« *matrix laboratory* ») est un langage de programmation de quatrième génération émulé par un environnement de développement du même nom ; il est utilisé à des fins de calcul numérique. Développé par la société The MathWorks, MATLAB permet de manipuler des matrices, d'afficher des courbes et des données, de mettre en œuvre des algorithmes, de créer des interfaces utilisateurs, et peut s'interfacer avec d'autres langages comme le C, C++, Java, et Fortran. MATLAB offre également la possibilité d'utiliser des *toolbox* (boîte à outils) [92].

Nous nous sommes servis de MATLAB version R2017a pour simuler notre méthode d'identification des piétons. En effet, nous avons utilisé la bibliothèque « *Computer Vision System Toolbox* » de MATLAB pour simuler à partir de la méthode HOG et SVM, l'identification d'un piéton sur une séquence d'images obtenues à partir d'une caméra. MATLAB et SIMULINK nous a également permis de simuler nos deux contrôleurs flous ainsi que l'algorithme d'apprentissage par renforcement.

4.2 Présentation du matériel

4.2.1 Les radars

Les capteurs radar dans notre système sont responsables de la détection des objets en avant et sur les côtés du véhicule. La théorie de base des systèmes radar est expliquée dans le **chapitre 2**. Dans un système radar automobile, un ou plusieurs capteurs radar détectent les obstacles autour du véhicule et leurs vitesses par rapport au véhicule. Sur la base des signaux de détection générés par les différents radars, le BD détermine l'action appropriée nécessaire pour éviter la collision ou pour réduire les dommages (atténuation des collisions). La méthode de mesure est le balayage actif, c'est-à-dire que le

radar transmet le signal radio et que le signal réfléchi est analysé. Dans notre projet trois types de radars sont utilisés : Radar à longue portée, Radar à moyenne portée et Radar à courte portée. Le **tableau 4-1** ci-dessous présente les caractéristiques des radars utilisés. Les principaux fabricants de ces radars étant les entreprises BOSCH, HELLA, Continental, Denso et Delphi.

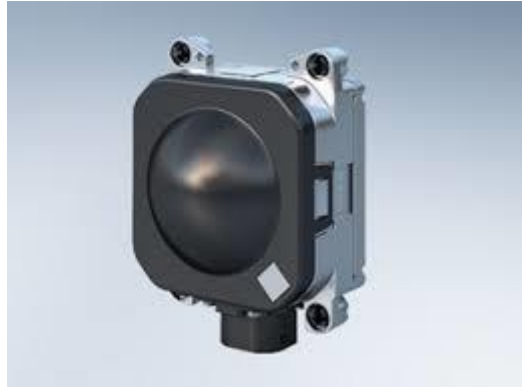


Figure 4.1 : Photo radar automobile BOSCH [93]

Tableau 4-1 : Caractéristiques des radars utilisés

<i>Types de radar</i>	Fréquence de fonctionnement (GHz)	Distance de mesure maximale (m)	Angle de balayage (°)
<i>Longue portée</i>	76-77	150-250	25-40
<i>Moyenne portée</i>	77-81	60-100	45-56
<i>Courte portée</i>	77-81	20-30	90-180

4.2.2 Caméra stéréovision

La caméra stéréovision est le type de caméra le plus utilisé par les constructeurs automobiles comme solution d'identification des obstacles dans les ADAS. C'est le seul capteur d'imagerie qui possède des capacités 3D et peut mesurer la distance des objets dans l'image. La caméra stéréovision est utilisée dans notre système pour l'identification de la nature des obstacles. Le **tableau 4-2** ci-dessous présente les caractéristiques moyennes des caméras stéréovisions adoptées pour notre système, elles sont inspirées des caractéristiques de la caméra stéréovision de Bosch.



Figure 4.2 : Photo caméra stéréovision [93]

Tableau 4-2 : Caractéristiques moyennes des caméras stéréovision adoptées

Angle de couverture (°)	Distance de mesure maximale (m)	résolution (pixels)
50	50	1280*960

4.2.3 ECU

L'automatisation d'un véhicule à partir des techniques de l'intelligence artificielle nécessite une puissance de calcul incroyable afin de pouvoir gérer toutes les données des capteurs en temps réel. Il faut de plus envisager la défaillance du matériel, et donc prévoir la redondance des capteurs et des supercalculateurs en cas de panne. De plus, le rapport de performance calcul/consommation énergétique (DL TOPS/Watt) est un élément clé dans le choix d'un microprocesseur pour une solution d'électronique embarquée dans un véhicule. Ces ainsi que de grandes firmes du domaine des microprocesseurs tel que AMD, INTEL et NVIDIA développent depuis une décennie déjà, des puces avec des puissances de calcul suffisante pour pouvoir intégrer l'IA pour des solutions de véhicule autonome [94].

Nous nous sommes inspirés du leader dans le cas des véhicules autonomes, NVIDIA avec son microprocesseur Xavier pour définir les caractéristiques des microprocesseurs que nous avons opté pour notre système présenté dans le **tableau 4-3** ci-dessous.



Figure 4.3 : Photo microprocesseur Xavier [93]

Tableau 4-3 : Caractéristiques des microprocesseurs intégrant l'IA pour l'électronique embarquée automobile

Puissance de calcul (DL TOPS ⁸)	Consommation (Watt)
30	20

L'objet de ce chapitre est de faire une description des différents outils logiciels que nous avons utilisés pour simuler et évaluer l'efficacité de notre système. Nous avons également, dans ce chapitre, présenter les caractéristiques des capteurs et des ECUs qui existent sur le marché et qui correspondent à nos choix de conception pour notre système.

⁸ DL TOPS qui est le sigle de Deep Learning Trillion operation per second, désigne l'unité de mesure de la puissance de calcul d'un microprocesseur. Il quantifie le nombre d'opération effectué par seconde.

Dans cette partie nous avons présenté la conception de notre système d'anticollision de piéton. Cette partie est divisée en deux chapitres. Le premier chapitre est une présentation sommaire de notre système. Une approche modulaire de conception y est développée en subdivisant la solution à la problématique de notre projet en trois blocs : Bloc de Perception (BP), Bloc de Décision (BD) et un Bloc de Commande (BC). Il fait également état des différentes entrées-sorties de notre système, ainsi qu'une présentation de l'architecture de communication adoptée pour les capteurs et modules du système. Il s'agit ici des protocoles de communication LIN et CAN. Par la suite, nous entrons plus en profondeur dans la spécification du système. Adoptant une démarche progressive allant de bloc en bloc, le BP est tout d'abord exposé à travers son architecture matérielle présentant la disposition des différents capteurs le constituant. Une fois les capteurs utilisés par le système présentés, en occurrence, les radars et une caméra, une application des méthodologies adoptées dans le **chapitre 2** est faite afin de permettre à notre système d'identifier la nature des obstacles : les méthodes HOG et SVM. L'intelligence de notre système est ensuite définie à travers le BD. C'est ce dernier qui permet à notre système d'évaluer à partir des données de l'environnement reçues du BP, le risque de collision avec un piéton, et par la suite, détermine la stratégie à employer afin d'éviter la collision. Ce chapitre se termine avec la description du BC dont le rôle sera de servir d'interface entre les commandes générées par le BD et les actionneurs de notre système qui sont les électrovannes du système de freinage hybride (*brake-by-wire*) et le moteur électrique du système de direction à assistance électrique (*steer-by-wire*) des véhicules hôtes. Des éléments de signalisation (leds, alarmes, vibrateurs,...) font également partie des actionneurs du système. Le deuxième chapitre de cette partie fait une description des deux outils logiciels utilisés pour simuler notre système, en occurrence le logiciel Matlab et Spyder. Une présentation des différentes caractéristiques des capteurs et des ECUs, disponible sur le marché et correspondant à nos choix de conception, est ensuite faite dans la deuxième section de ce chapitre.

PARTIE 3 : RESULTATS ET DISCUSSION

Cette partie est divisée en deux chapitres. Dans le premier chapitre, nous allons simuler les principales fonctionnalités de notre système en occurrence l'identification d'un piéton et l'intelligence du système. Nous allons tout particulièrement, à partir des outils présentés dans le **chapitre 4**, simuler les différentes méthodes retenues pour la conception de notre système dans des situations réelles auxquelles le système pourrait faire face. Ceci nous permettra de valider l'efficacité de nos choix de conception. Le deuxième chapitre présente le prototype, il décrit le fonctionnement du prototype ainsi que les différents outils et matériels utilisés pour le réaliser.

CHAPITRE 5 : SIMULATIONS ET DISCUSSIONS

Dans ce chapitre, nous allons présenter des simulations de l'application des différentes méthodes utilisées pour la conception de notre système d'anticollision de piéton. Il s'agira pour nous d'évaluer la capacité de notre système à effectuer les fonctionnalités pour lesquelles il a été conçu, en occurrence l'identification de la nature d'un obstacle, l'évitement à travers des manœuvres des obstacles identifiés et enfin, le repositionnement du véhicule afin de permettre au conducteur d'aisément reprend le contrôle de son véhicule une fois l'obstacle évité. Pour cela, nous avons choisi de présenter ces résultats bloc par bloc accompagné de commentaires.

5.1 Simulation de l'identification d'un piéton

Pour vérifier l'efficacité des méthodes utilisées pour l'identification de piéton, nous avons fait des simulations sur différentes séquences vidéo représentant des situations récurrentes relatives aux accidents de piétons [7], en occurrence, un conducteur confronté à un piéton longeant la route, à un piéton traversant la route, à un piéton jouant ou courant et à un piéton initialement masqué. Les objets détectés sont automatiquement encadrés par un rectangle portant la valeur de la probabilité que l'objet encadré soit un piéton.

a- Cas1 : Situation conducteur confronté à un piéton longeant la route



Figure 5.1 : Simulation de l'identification d'un piéton dans le cas d'un piéton longeant la route

b- Cas2 : Situation conducteur confronté à un piéton traversant la route

Figure 5.2 : Simulation de l'identification d'un piéton dans le cas d'un piéton traversant la route

Les résultats obtenus pour les cas1 et cas2 peuvent aisément s'étendre aux autres cas cités précédemment. On observe de ces résultats qu'à partir d'une probabilité supérieure à 0,05 il est plausible que la séquence d'image contienne un piéton. Nous observons également que notre code ne fait pas de distinction entre un piéton et un homme sur une motocyclette ou tout autre engin laissant transparaître la forme d'un homme.

5.2 Simulation de l'intelligence du système

Dans cette section, nous présenterons une vue globale de chacun de nos deux contrôleurs flous. Nous présenterons par la suite, pour chaque cas, les surfaces illustrant les valeurs des variables de sortie générées en fonction des valeurs des variables d'entrée correspondantes. Nous exposerons ensuite quelques cas de figure pour illustrer des situations récurrentes.

5.2.1 Simulation contrôleur flou d'évitement

Lorsque le risque de collision, à partir des données perçues par les capteurs, est imminent le véhicule doit, de manière automatique, éviter l'obstacle en actionnant le système de freinage et/ou de

direction. Ceci est réalisé à partir du contrôleur flou d'évitement. La **figure 5.3** ci-dessous montre une vue globale de ce contrôleur implémenter à partir du logiciel Matlab. Comme présenté sur la figure, le contrôleur prend en entrée les 4 variables linguistiques RG, RC, RD et V et génère à la sortie à partir d'un contrôleur de type TS, les variables d'angle $\emptyset$ et de pression P.

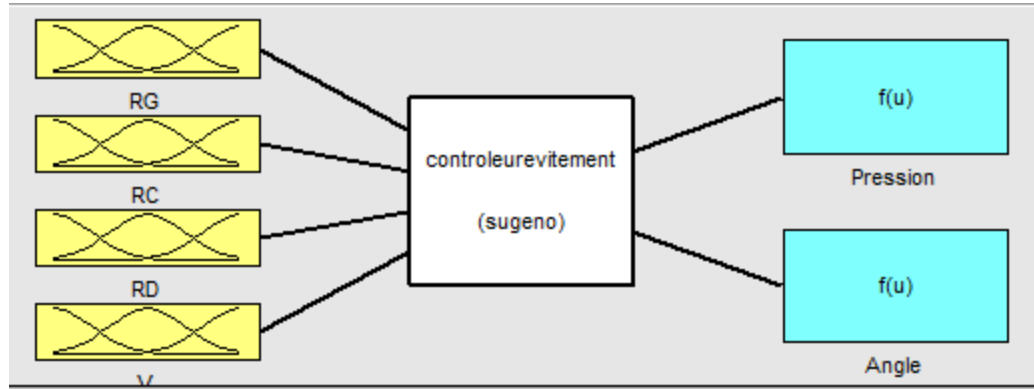
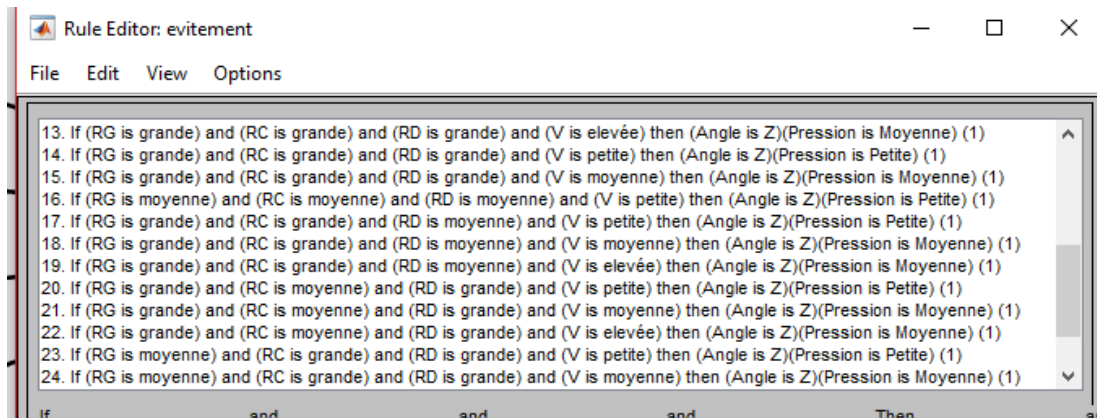


Figure 5.3 : Vue globale du contrôleur flou d'évitement à partir de Matlab



(a)

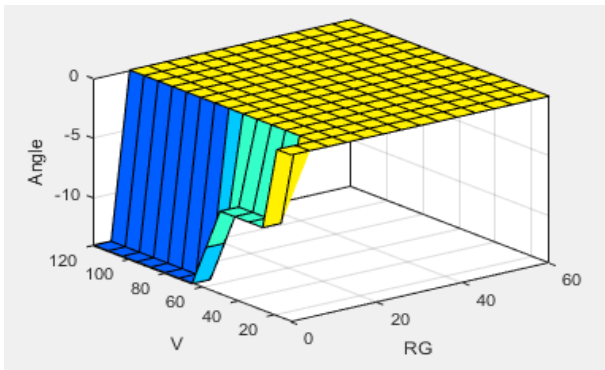


(b)

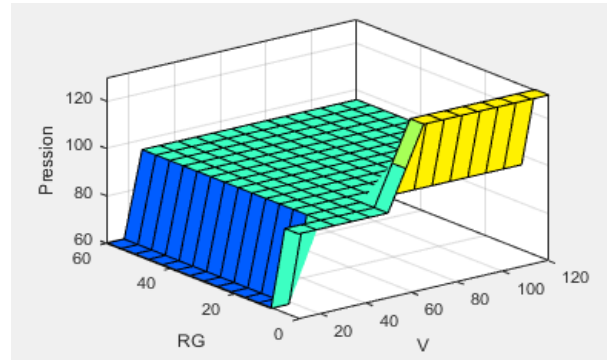


(c)

Figure 5.4 (a) (b) (c) : Règles du contrôleur flou d'évitement



(a)



(b)

Figure 5.5 : Surfaces de commande de (a) l'angle de déviation et de (b) la pression pour le couple d'entrée (RG, V)

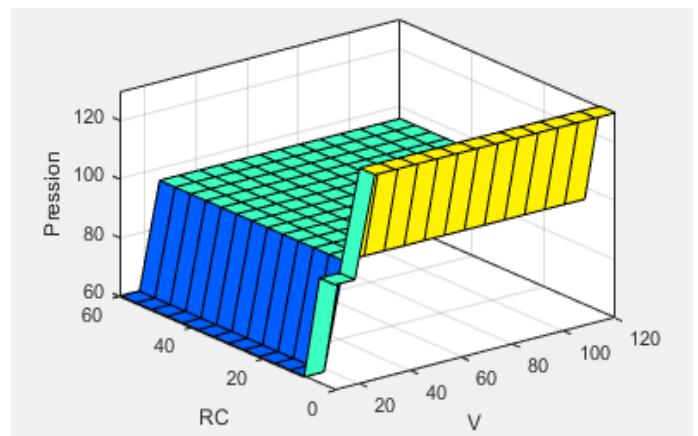
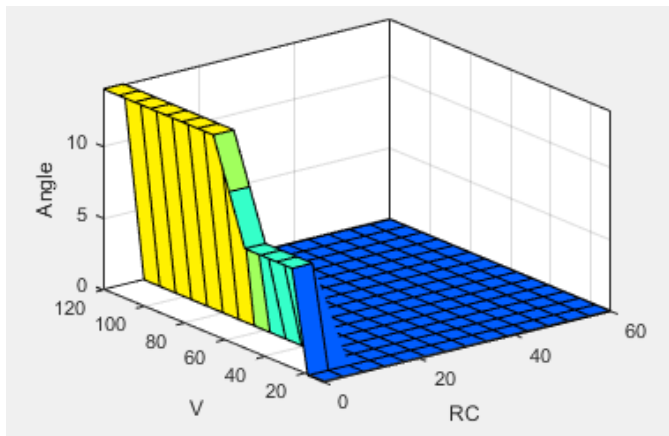


Figure 5.6 : (a) Surface de commande de l'angle de déviation pour le couple d'entrée (V, RC) (b) Surface de commande de la pression pour le couple d'entrée (V, RC)

Les figures 5.5 (a) (b) et figures 5.6 (a) (b) ci-dessus présentent les surfaces de commande de l'angle de déviation $\emptyset$ et de la pression P pour chaque combinaison des entrées (RG, V) et (V, RC).

Dans la figure 5.5 (a), qui illustre la surface de commande de l'angle $\emptyset$ généré par le contrôleur lorsque le capteur RG détecte un obstacle en fonction de la vitesse V du véhicule, on peut remarquer pour les distances mesurées appartenant à la valeur linguistique *Grande* (distance supérieur à 25m) que l'angle généré est nul. Plus cette distance diminue, la vitesse V augmentant, plus l'angle de déviation augmente du côté droit (angle de signe négatif) afin d'éviter l'obstacle situé à gauche du véhicule. La figure 5.5 (b) illustre pour le même couple d'entrée, la surface de commande de la pression P. On observe de cette dernière, que la pression de freinage augmente avec la vitesse, la distance mesurée diminuant. Les mêmes résultats ont été obtenus pour le couple d'entrée (RD, V) pour ce qui est de la surface de commande de la pression. La surface de commande de l'angle de ce cas présente les mêmes variations que celles du couple d'entrée (RG, V) à la seule différence que les angles générés ici ont un signe négatif.

L'interprétation des figures 5.6 (a) et (b) se fait par analogie avec celles des figures 5.5 (a) et (b).

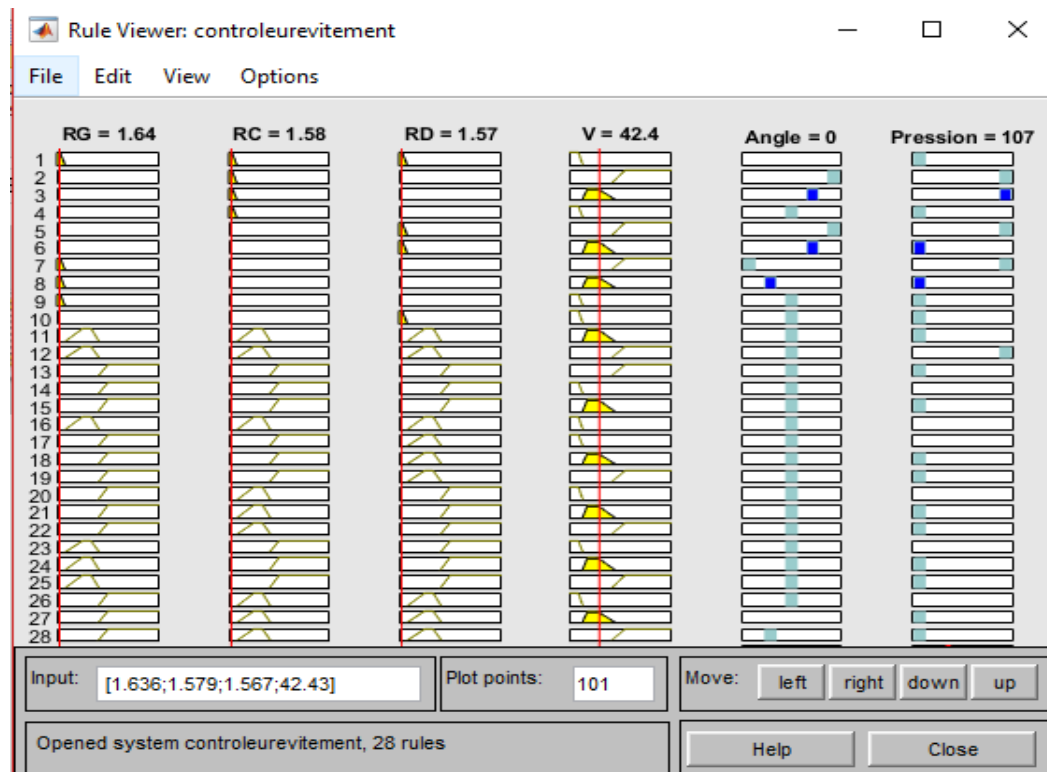
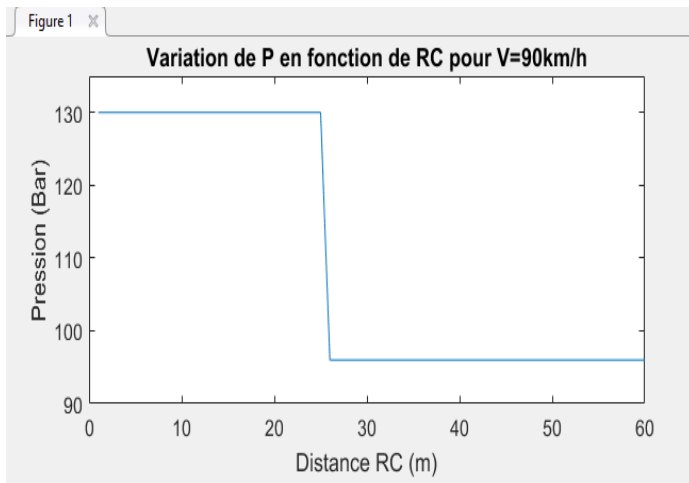
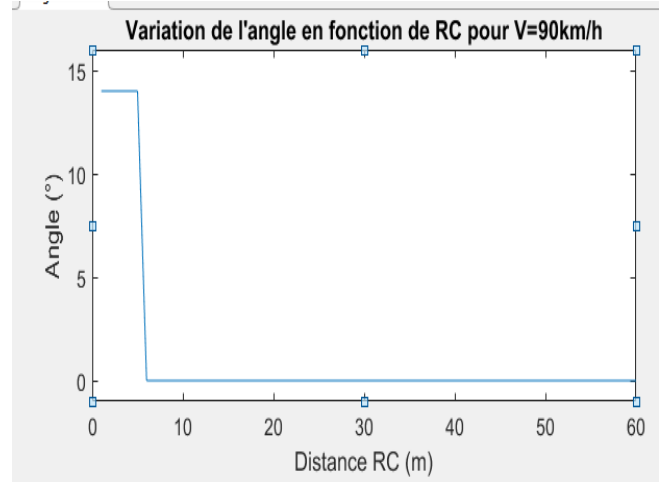


Figure 5.7 : Illustration de l'inférence des valeurs de commande du contrôleur flou d'évitement à partir du logiciel

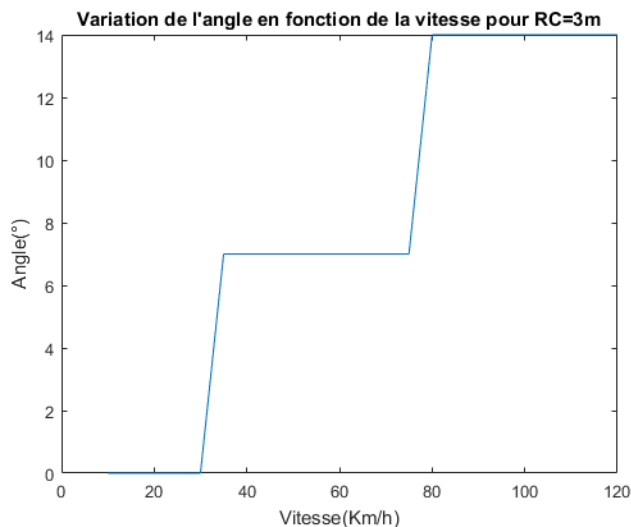


(a)

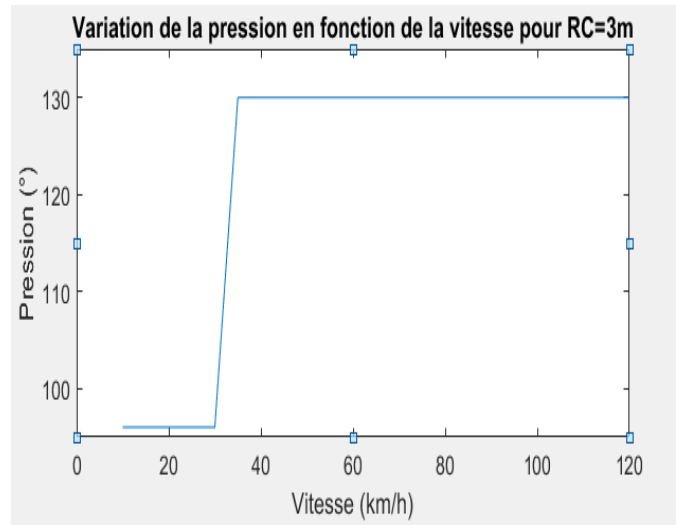


(b)

Figure 5.8 : Variation pour $V=90\text{km/h}$ de (a) P en fonction de RC ; (b) $\emptyset$ en fonction de RC



(a)



(b)

Figure 5.9 : Variation pour $RC=3\text{m}$ de (a) $\emptyset$ en fonction de V ; (b) P en fonction de V

A partir du modèle SIMULINK du contrôleur d'évitement (**annexe 4**), nous avons généré les valeurs de sortie inférées pour quelques combinaisons de variables d'entrée représentant des cas de figures récurrentes. Les **figure 5.8 (a)(b)** et **figure 5.9 (a)(b)** illustrent les valeurs de commande obtenues.

Les **figures 5.8 (a)** et **(b)** représentent le cas où la vitesse du véhicule est élevée (90km/h) et un obstacle est détecté à une distance RC en face du véhicule. On constate que les valeurs de la pression de freinage sont 96 et 130 bar qui correspondent aux valeurs linguistiques forte et moyenne.

Pour une distance inférieure à 25m, correspondant à la valeur linguistique petite, la pression de freinage est forte et moyenne dans l'autre cas. Parallèlement, pour une distance inférieure à 6 m, le véhicule dévie d'un angle de 14° correspondant à l'angle maximal de déviation et conserve sa trajectoire pour les distances supérieures.

Les figures 5.9 (a) et (b) représentent le cas où la distance RC est égale à 3m. On constate pour une vitesse comprise entre 0 et 25 km/h, le véhicule conserve sa trajectoire et la pression de freinage est égale à 0bar. Pour les vitesses moyennes comprises entre 25 et 80 km/h, l'angle de déviation est de 7° et la pression de freinage est 130 bar. Pour des vitesses comprises entre 80 et 120 km/h, toutes les variables de sortie ont leurs valeurs maximales de 130 bar pour la pression de freinage et 14° pour l'angle de déviation.

5.2.2 Simulation algorithme Q-learning de renforcement

Le contrôleur flou d'évitement présenté précédemment est amélioré grâce à une technique d'apprentissage automatique, en occurrence l'apprentissage par renforcement tel que spécifié dans 3.2.2.4.2. Ceci est fait par l'intermédiaire de l'algorithme Q-learning. Nous avons à l'aide du logiciel Matlab, simuler le comportement d'un agent entraîné par renforcement dans un environnement similaire à celui d'une route. L'agent doit à partir d'une position de départ D atteindre une cible C en évitant un obstacle placé sur son trajet.

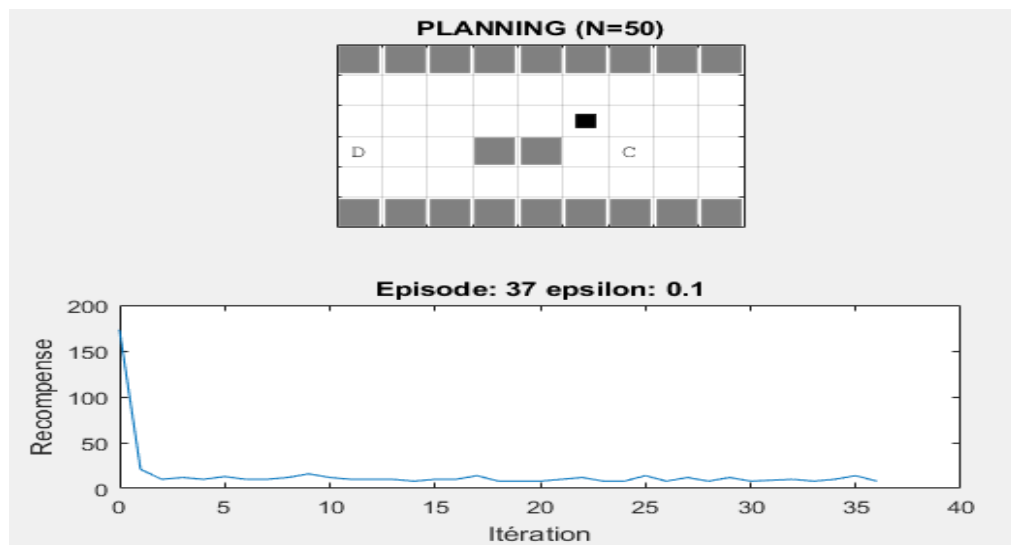


Figure 5.10 : Illustration de l'apprentissage de l'agent

La **figure 5.10** ci-dessus illustre l'apprentissage de l'agent pour quitter du point D au point C en évitant l'obstacle en face de lui. On remarque que plus le nombre d'itération augmente, plus le nombre d'erreur que le simulateur de résultats met pour arriver à la cible C diminue en tendant vers 0. On observe donc que l'algorithme conçu permet effectivement à notre agent d'atteindre sa cible et dès la 3^{ème} itération, le but de l'agent par la suite reste d'optimiser la trajectoire.

5.2.3 Simulation contrôleur flou d'alignement

Une fois l'obstacle évité, le véhicule doit être repositionné afin de permettre au conducteur d'aisément reprendre le contrôle de son véhicule. Ceci est réalisé à partir du contrôleur flou d'alignement. La **figure 5.10** ci-dessous montre une vue globale de ce contrôleur implémenter à partir du logiciel Matlab. Comme présenté sur la figure, le contrôleur prend en entrée les 4 variables linguistiques d1, d2, d3 et d4 et génère à la sortie à partir d'un contrôleur de type TS, la variable d'angle $\emptyset$.

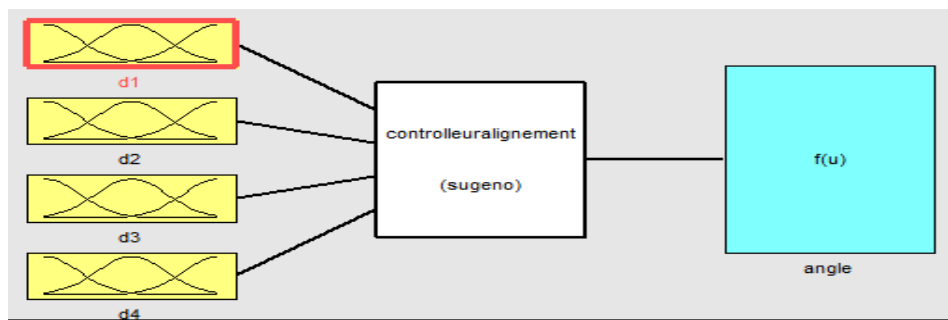


Figure 5.11 : Vue globale du contrôleur flou d'alignement

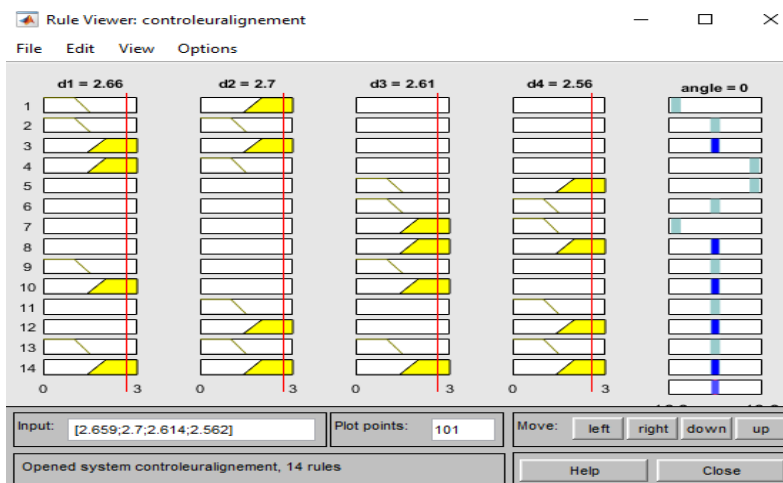
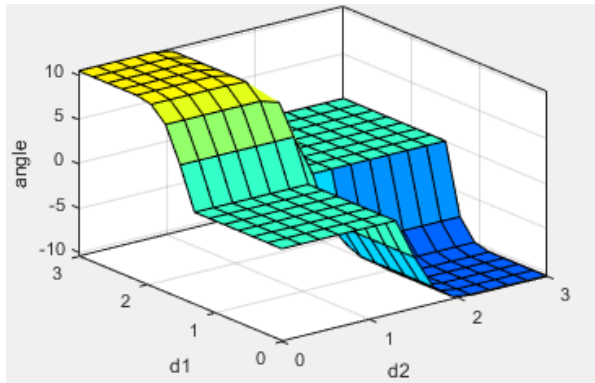
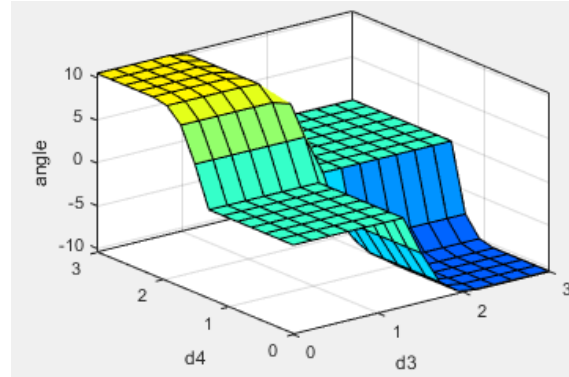


Figure 5.12 : Illustration de l'inférence des valeurs de commande du contrôleur flou d'évitement

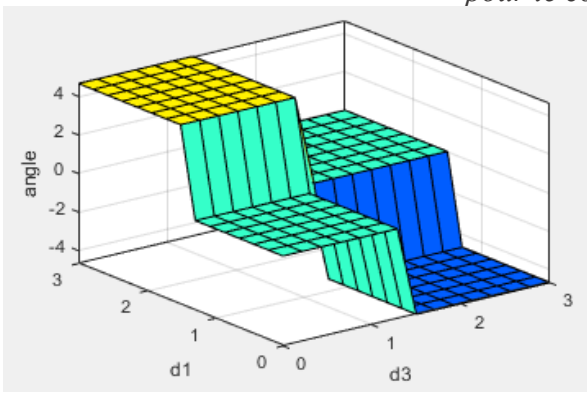


(a)

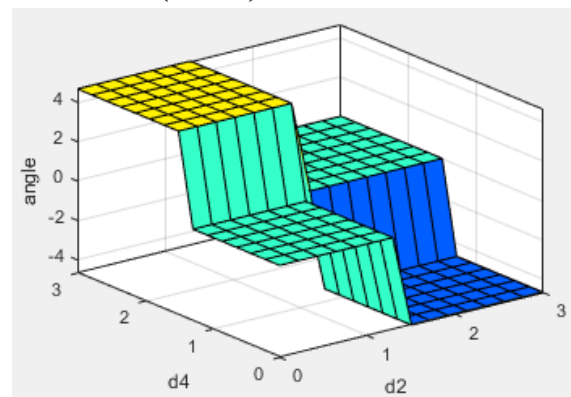


(b)

Figure 5.13 : (a) Surface de commande pour le couple d'entrées $(d1, d2)$; (b) Surface de commande pour le couple d'entrées $(d3, d4)$.



(a)



(b)

Figure 5.14 : (a) Surface de commande pour le couple d'entrées $(d1, d3)$; (b) Surface de commande pour le couple d'entrées $(d2, d4)$.

Les figures 5.13 (a) (b) et figures 5.14 (a) (b) ci-dessus présentent les surfaces de commande de $\emptyset$ pour chaque combinaison des entrées $(d1, d2)$, $(d3, d4)$, $(d1, d3)$ et $(d2, d4)$. On peut remarquer dans tous les quatre cas que 3 commandes sont générées, ceci correspondant soit à l'appartenance des deux variables d'entrée à une même valeur linguistique, soit l'appartenance des deux variables d'entrée à des valeurs linguistiques différentes.

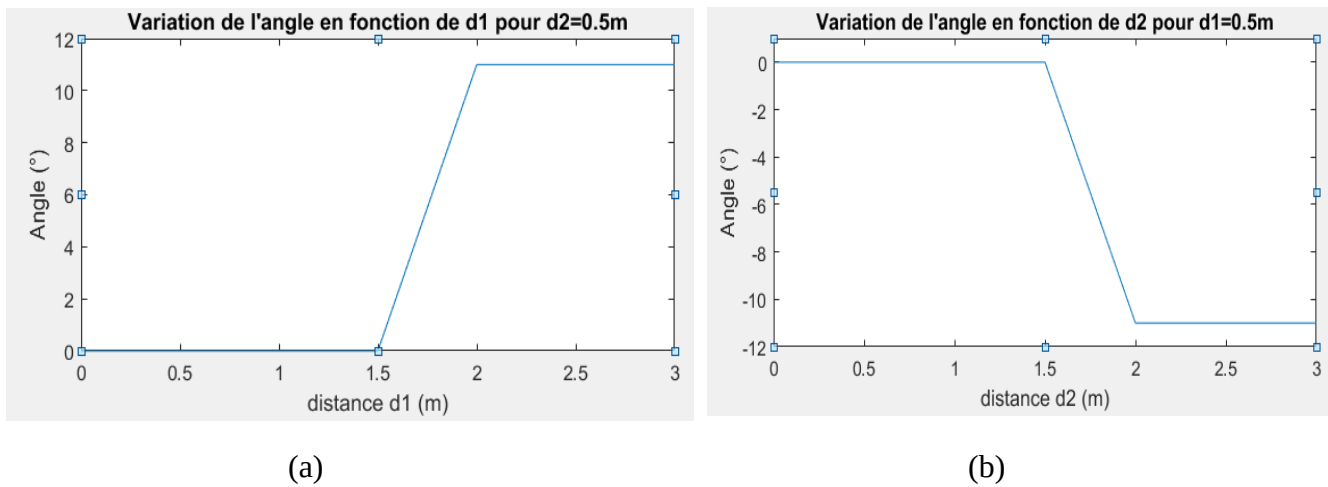


Figure 5.15 : (a) Variation de $\varnothing$ en fonction de $d1$ pour $d2=0.5m$ (b) Variation de $\varnothing$ en fonction de $d2$ pour $d1=0.5m$

A partir du modèle SIMULINK du contrôleur d'alignement (**annexe 5**), nous avons généré les valeurs de sortie inférées pour quelques combinaisons de variables d'entrée. Les **figures 5.15 (a) et (b)** ci-dessus illustrent les valeurs de commande obtenues.

Dans la **figure 5.15 (a)**, nous avons fixé la distance $d2$ à 0.5m (valeur linguistique petite). On observe dans ce cas, que pour des distances $d1$ inférieure à 1.5 m l'angle générée est nulle et égale à 11° dans le cas contraire.

Une interprétation analogique à celle de la **figure 5.16 (a)** peut être faite pour le cas de la **figure 5.16 (b)**.

Dans ce chapitre, nous avons, à l'aide d'outils logiciels, testé les différentes méthodes retenues pour la conception de notre système en occurrence HOG et SVM pour l'identification de piéton, un contrôleur flou entraîné par apprentissage pour l'évitement des obstacles et un contrôleur flou pour le repositionnement du véhicule une fois l'obstacle évité. D'après les résultats de simulation, nous avons pu constater que la méthode HOG et SVM nous permettait effectivement, à partir d'une séquence d'image, de détecter la présence d'un piéton néanmoins, nous avons fait le constat, que cette détection ne faisait pas de distinction entre le piéton et un homme placé sur une motocyclette. Le contrôleur d'évitement

CHAPITRE 6 : REALISATION DU PROTOTYPE

L'idée principale de ce projet est de réaliser un système d'anticollision de piéton pour véhicule de tourisme qui soit capable, de façon automatique, d'identifier tout obstacle (principalement les piétons) et de générer des commandes permettant d'effectuer des manœuvres afin d'éviter toute collision. Ce chapitre présente le prototypage de notre système. Il fait tout d'abord un bref exposé des fonctionnalités du prototype, ensuite présente les outils et matériels utilisés pour sa réalisation.

6.1 Présentation du prototype

Le prototype de notre projet est en réalité une voiture de petite dimension doté de capteurs et de microcontrôleurs. Cette voiture est capable de détecter les obstacles présentés en face de lui, de différencier la nature des obstacles détectés et de choisir de façon autonome une manœuvre à effectuer, en fonction du risque, pour éviter l'obstacle. Les manœuvres pouvant être de freiner ou alors de freiner et de dévier. Le prototype est également doté d'élément de visualisation permettant d'informer sur l'état du système. En cas de situation de collision inévitable, la voiture envoie automatiquement un message SOS afin d'alerter une institution compétente. La **figure 6.1** ci-dessous illustre le schéma bloc du prototype. L'annexe 5 présente l'organigramme de ce dernier.

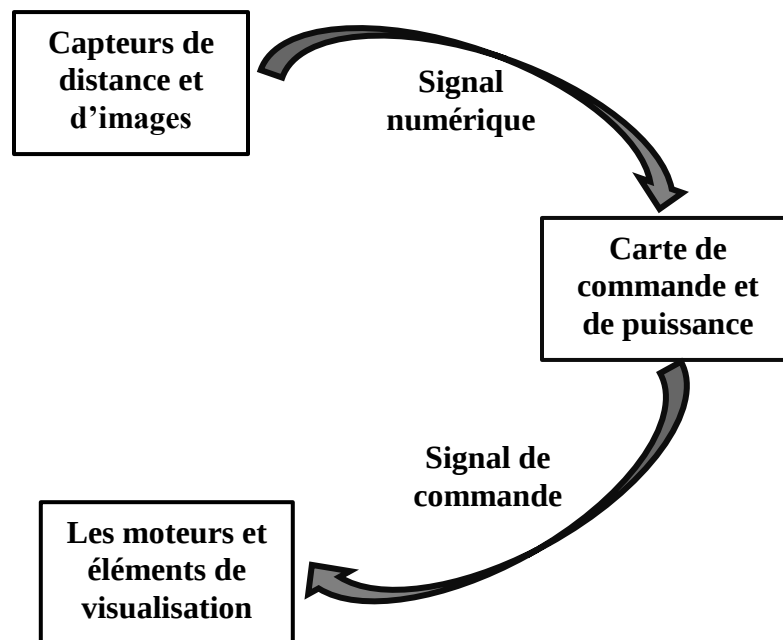


Figure 6.1 : Schéma bloc du prototype réalisé

6.2 Outils et matériels utilisés

6.2.1 Présentation des outils

6.2.1.1 Arduino IDE

Arduino IDE est un logiciel de programmation des modules Arduino. Il s'agit d'une application Java, libre et multiplateforme, servant d'éditeur de code et de compilateur, et qui peut transférer le firmware et le programme au travers de la liaison série (RS-232, Bluetooth ou USB selon le module). Le langage de programmation utilisé est le C++. La mise en place de ce langage standard rend aisé le développement de programmes sur les plates-formes Arduino, à toute personne maîtrisant le C ou le C++.

Nous avons utilisé l'Arduino IDE version 1.8.0 (**figure 6.2**) pour programmer les cartes de commande du prototype.

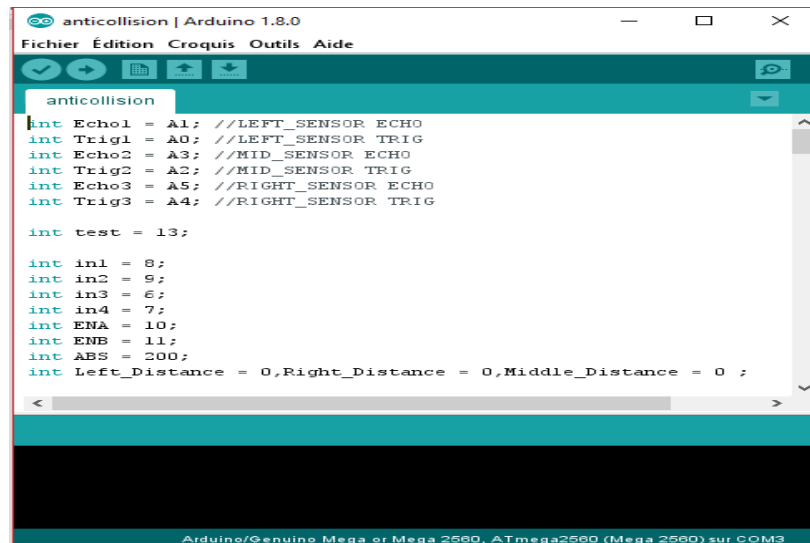


Figure 6.2 : Interface du logiciel arduino IDE version 1.8.0

6.2.1.2 Logiciel Proteus

Proteus est une suite logicielle permettant la conception assistée par ordinateur (CAO) électronique éditée par la société Labcenter Electronics. **Proteus** est composé de deux logiciels

principaux : ISIS, permettant entre autres la création de schémas et la simulation électrique, et ARES, dédié à la création de circuits imprimés.

Nous avons utilisé logiciel ISIS de PROTEUS version 8 professionnel (**figure 6.3**) pour simuler le fonctionnement de nos circuits électroniques avant implémentation.

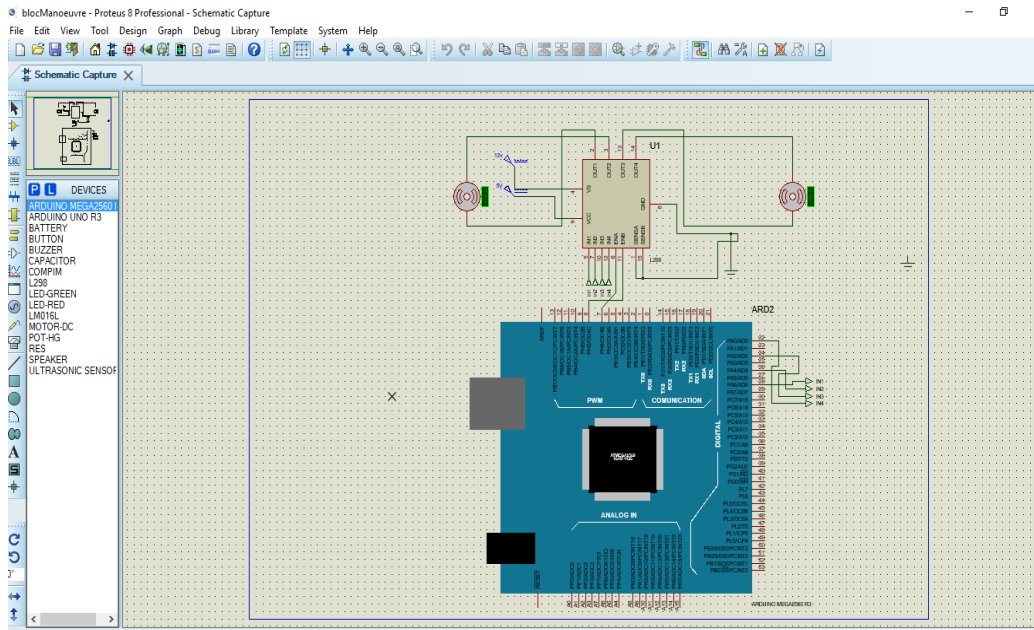


Figure 6.3 : Interface du logiciel ISIS PROTEUS

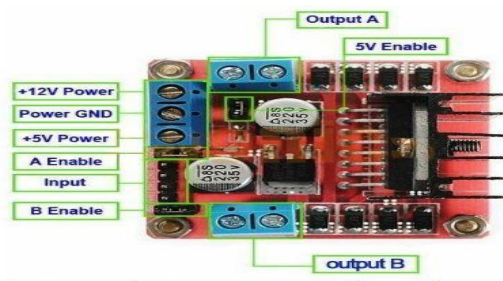
6.2.2 Description du matériel

Dans cette section, nous faisons une brève description des composants que nous avons utilisés pour réaliser le prototype du projet. Des contraintes liées à la disponibilité de certains composants sur le marché ainsi que leurs coûts sont des facteurs qui ont guidés notre choix des composants. Afin d'être le plus fidèle possible avec le système réel, nous avons opté pour les composants suivant :

- c- Cartes Arduino qui jouent le rôle d'ECUs pour notre prototype (**figure 6.4 (d)**).
- d- Des ultrasons (HCSR 04) pour la détection d'obstacle, ils sont utilisés ici à la place des radars (**figure 6.4 (c)**).
- e- Le pont-H L298N. Son rôle sera de commander les deux moteurs courant continu de notre prototype (**figure 6.4 (a)**).

- f- Une caméra Pixy qui est utilisé ici à la place de la caméra stéréovision. Son choix se justifie par sa capacité de faire du traitement d'image, étant lui-même doté d'un microprocesseur dédié.
- g- Un écran LCD qui est le principal élément de visualisation de l'état du système.
- h- Un module Bluetooth (HC-06) qui permet, lorsque la collision s'avère inévitable, d'envoyer un message SOS.
- i- Deux moteurs CC 12V qui permettent de déplacer le prototype (**figure 6.4 (b)**).

La liste détaillée des composants électroniques est précisé dans le tableau présentant le budget de réalisation du prototype de l'**annexe 8**.



(a)



(b)



(c)



(d)

Figure 6.4 : (a) Photo pont H L-298N; (b) Photo moteurs CC; (c) Photo capteur ultrason; (d) Photo carte arduino uno

6.2.2.1 Carte Arduino

Arduino, et son récent synonyme Genuino, est une marque qui couvre des cartes matériellement libres sur lesquelles se trouve un microcontrôleur. La carte arduino a de nombreux ports, en fonction de la version, permettant d'implémenter des cartes de commande. Le microcontrôleur peut être programmé pour analyser et produire des signaux électriques, de manière à effectuer des tâches très diverses comme la domotique (le contrôle des appareils domestiques - éclairage, chauffage), le pilotage d'un robot et l'informatique embarquée.

Nous avons, dans le cadre de notre projet, utilisé deux cartes arduino, une carte arduino Uno et une carte arduino Mega. Le **tableau 6-1** ci-dessous présente les caractéristiques de ces deux cartes.

Tableau 6-1 : Caractéristiques des cartes arduino Uno et Méga

Caractéristiques	Cartes Arduino	
	Uno	Méga
Microcontrôleur	ATmega328	ATmega 2560
Tension de fonctionnement	5v	5V
Tension d'alimentation	7V-12V	7-12V
Broches E/S (numérique)	14	54
Broches d'entrée analogique	6	15
Intensité maxi disponible par broche	40mA	40mA
Mémoire programme Flash	32ko	256ko
Vitesse d'horloge	16Mhz	16Mhz

6.2.2.2 Caméra Pixy

La caméra Pixy (**Figure 6.5**) est une caméra intelligente dotée d'un microprocesseur dédié. Elle utilise un algorithme de "filtrage de couleur" sur la base de "teintes" pour détecter des objets. Elle calcule la teinte et la saturation de chaque pixel RGB à partir du capteur vidéo et les utilise en tant que paramètres de filtrage primaire. Le **Tableau 6-2** ci-dessous présente ses caractéristiques.

Tableau 6-2 : Caractéristique de la caméra pixy

Processeur	Champs de vision	Alimentation	Sorties
NXP LPC4330, 204 MHz, dual core	75° à l'horizontale, 47° à la verticale	Entrée USB (5 V) ou entrée non régulée (6 à 10 V)	UART Série, SPI, I2C, USB, digitale, analogique



Figure 6.5 : Photo caméra pixy

6.3 Réalisation

Les annexes 6.a et 6.b présente les schémas électroniques des deux blocs (manœuvre et identification) que nous avons réalisés pour le prototype. L'annexe 7 présente l'organigramme de programmation du prototype et l'annexe 10 quelques photos du prototype réalisé.

Dans cette partie, nous avons présenté les résultats de simulation de la méthodologie d'identification de piéton adoptée ainsi que ceux des méthodes dotant notre système d'une intelligence. Les résultats obtenus, pour le cas de l'identification de piéton, révèlent l'efficacité de la méthode employée. Cependant, une insuffisance a été observée au niveau de la distinction entre un piéton et un homme sur une motocyclette ou tout autre engin laissant transparaître la forme d'un homme. Pour ce qui est de la méthodologie de navigation autonome, les résultats de simulation des deux contrôleurs conçus ont été satisfaisants. Ils ont présenté une conformité avec les connaissances d'expert que nous avons introduit. Par la suite, la simulation de l'algorithme d'apprentissage Q-learning, nous a permis de conforter notre hypothèse sur son efficacité et son adaptabilité dans le cas de notre projet. Cette partie se termine par la présentation de la démarche de prototypage de notre système. Une vue globale du prototype y est exposée, ainsi qu'une description des outils et matériels utilisés pour sa réalisation.

CONCLUSION GENERALE

Dans ce travail, le problème de la sécurité routière a été abordé par le développement d'un système d'aide à la conduite (ADAS) pour véhicule de tourisme. L'objectif établi dans le cahier de charges était de concevoir un système embarqué capable de détecter et identifier les objets dans l'environnement du véhicule, d'évaluer continuellement le risque de collision avec les obstacles et de planifier une trajectoire d'évitement lorsque le risque devient important. Les résultats obtenus en simulation nous ont donnés des résultats satisfaisant. Tout d'abord, la simulation de l'identification de la nature des objets a montré que nous pouvons effectivement identifier dans une séquence d'image un piéton à partir d'un capteur de vision. Les résultats de la simulation des contrôleurs flous ont démontré l'efficacité de la commande à base de la logique floue pour les différentes fonctions de notre système en occurrence l'évitement d'obstacle et le repositionnement du véhicule. Le test effectué avec l'algorithme de Q-learning nous a permis d'observer la capacité d'apprentissage d'un agent à partir des données reçues directement de son interaction avec son environnement, il s'agit donc d'une solution prometteuse pour le problème d'anticollision par son habileté à optimiser les paramètres d'un SIF et donc d'appuyer l'expertise humaine. Le coût d'implémentation du projet a été estimé à 2 258 750 FCFA.

Plusieurs perspectives sont envisageables suite à nos travaux :

- Application d'une méthode d'identification des objets qui permet une distinction plus poussée entre les piétons et les occupants des véhicules légers.
- Faire la validation du système flou entraîné par l'algorithme d'apprentissage par renforcement Q-learning.
- Faire des études de l'application de ce système dans les autres catégories de véhicule en occurrence les véhicules utilitaires et les véhicules légers.

BIBLIOGRAPHIE

- [1] Organisation Mondiale de la Santé. *Rapport de situation sur la sécurité routière dans le monde*. 2013.
- [2] Commission Economique des Nations Unies pour l'Europe. [En ligne] 2008. <http://www.unece.org/trans/main/wp29/wp29regs1-20.html>.
- [3] Mèmeteau, Hubert. *Technologie fonctionnelle de l'automobile*. 5e. s.l. : Dunod, 1981. p. 224.
- [4] Banque Africaine de Développement. *Cameroun, Note sur le secteur des transports*. Abidjan : s.n., 2015. p. 44.
- [5] Chong, Jed. Véhicules autonomes et connectés: état d'avancement de la technologie et principaux enjeux stratégiques pour les pouvoirs publics au canada. *Etude Générale*. Ottawa : Bibliothèque du parlement, 2016, p. 24.
- [6] Wendling, Barbara. *First revision of J3016-update on task force activities*. s.l. : SAE On-Road Automated vehicle Standards committee.
- [7] Houenou, Adam. *Calcul de trajectoires pour la préconisation de manoeuvres automobiles sur la base d'une perception multi-capteur: application à l'évitement de collision*. Université de Technologie de Compiègne. 2013. p. 184, Thèse de doctorat.
- [8] Salameh, Nadeen. *Conception d'un système d'alerte embarqué basé sur les communications entre véhicules*. Roen : Institut National des Sciences Appliquées de Rouen, 2011. p. 151, Thèse de doctorat.
- [9] Wikipedia. [En ligne] [Citation : 04 Septembre 2017.] https://fr.wikipedia.org/wiki/S%C3%A9curit%C3%A9_active.
- [10] Daimler. Mercedes-benz. [En ligne] [Citation : 04 Septembre 2017.] https://techcenter.mercedes-benz.com/en/attention_assist/detail.html.
- [11] *Adas safety impacts on rural and urban highways*. Louwerse, W et Hoogendoorn, S. Parma : s.n., 2004. IEEE Intelligent Vehicles Symposium.
- [12] *Collision warning with full auto brake and pedestrian detection*. Coelinght, E, Eidehall, A et Bengtsson, M. Madeira Island, Portugal : s.n., 2010. IEEE Conference on intelligence transportation systems.

- [13] *Fusion of Stereo Vision for Pedestrian Recognition using Convolutional Neural Networks*. Pop, Danut O, et al. Bruges,Belgium : s.n., 2017. 25th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning.
- [14] Volvo. Volvo car Usa Support. [En ligne] [Citation : 04 septembre 2017.] http://volvo.custhelp.com/app/answers/detail/a_id/9862/~/_city-safety-with-steering-assist.
- [15] Ford, jason. The engineer. [En ligne] 29 Octobre 2015. [Citation : 4 Septembre 2017.] <https://www.theengineer.co.uk/bosch-developing-pedestrian-avoidance-system/>.
- [16] Toyota. [En ligne] [Citation : 4 Septembre 2017.] <https://www.toyota-europe.com/world-of-toyota/safety-technology/toyota-pedestrian-avoidance-steer-assist-system>.
- [17] Grisel, Richard et Abouchi, Nacer. [En ligne] [Citation : 13 Août 2017.] http://richard.grisel.free.fr/Master_OSM/2_Introduction_Embedded_systems.pdf.
- [18] Pomian, J.L, Pradere, T et Gaillard, I. *Ingénierie et ergonomie*. Toulouse : CEPADUES Editions, 1997.
- [19] Van Oudenhove de saint gery, Thomas. *Contribution à l'élaboration d'un formalisme gérant la pertinence pour les problèmes d'aide à la conception à base de contraintes*. 2006.
- [20] *Design problem solving : a task analysis*. Chandrasekaran, B. 11, 1990, Artificial Intelligence Magazine, pp. 59-71.
- [21] Marsot, J. *Méthode et outils pour intégrer l'ergonomie dans le cycle de conception des outils à mains*. Vandoeuvre : s.n., 2002.
- [22] Aiache, Jacques et Chereau, Jean-Marc. Klubprepa. [En ligne] [Citation : 18 08 2017.] <http://www.klubprepa.fr>.
- [23] Akao, Y. *QFD-Prendre en compte les besoins du client dans la conception du produit*. Paris : AFNOR, 1993.
- [24] *La méthode d'aide à l'innovation TRIZ et les limites de son déploiement en entreprise*. Maranzana, Sébastien, Caillaud, Emmanuel et Rose, Bertrand. Prague : s.n., 2016. 23 ème Colloque des sciences de la conception et de l'innovation.
- [25] Altshuller, G. *The innovation Algorithm-TRIZ,systematic innovation and technical creativity*. Massachussets : Technical Innovation Center, 1999.

- [26] Boldrini, jean-claude. *L'accompagne des projet d'innovation.Le suivi de l'introduction de la méthode TRIZ dans une entreprise de petite taille.Business administration*. Nantes : Université de Nantes, 2005.
- [27] Chan, Ching-Yao, Bu, Fanping et Shladover, steven. *Experimental vehicle platform for pedestrian detection*. Institute of transportation studies, University of California. Berkeley : California Path Research Report, 2006. p. 74. ISSN 1055-1425.
- [28] Davies, D. *Research, development, and implementation of pedestrian safety facilities in the United Kingdom*. s.l. : FHWA research report, 1999. FHWA-RD-99_089.
- [29] Minnesota Dot. *NIT Phase II: Evaluation of Non-Intrusive Technologies for Traffic Detection*. 2002. SRF No. 3683.
- [30] Noyce, D et Dharmaraju, R. *An Evaluation of Technologies for Automated Detection and Classification of Pedestrians and Bicyclists*. Massachusetts : Massachusetts Highway Department report, 2002.
- [31] *Passive Pedestrian Detection at Un-Signalized*. Beckwith, D et Hunter-Zaworski, K. 1998. 77th Annual Transportation Research Board Meeting.
- [32] *Design and Experimental Study of an Ultrasonic Sensor System for Lateral Collision Avoidance at Low Speeds*. Song, K, Chen, C et Huang, C. 2004. IEEE Intelligent Vehicle Symposium. pp. 647-651.
- [33] Knoll, P, et al. *Predictive Safety System – Steps towards Collision Mitigation*. s.l. : SAE Technical Paper, 2004. 2004-01-1111.
- [34] *A Multi-sensor Approach for the Protection of Vulnerable Traffic Participants- the Protector Project*. Gavrila, D, Kunert, M et Lages, U. 2001. IEEE Instrumentation and Measurement Technology Conference. pp. 2044-2048.
- [35] *Short RangeDevices, RFID, Bluetooth, Ultra Wideband Systems and Automotive Short Range Radars: Overview and Latest Development*. Scherrer, D et Burgherr, R. 2003. OFCOM report.
- [36] *Radar Sensors and Sensor Platform used for Pedestrian Protection in the EC-funded Project SAVE-U*. Tons, M, et al. 2004. IEEE Intelligent Vehicles Symposium. pp. 813-818.
- [37] *Laser Scanners for Obstacle Detection in Automotive Applications*. Ewald, A et Willhoeft, V. 2000. IEEE Intelligent Vehicles Symposium. pp. 682-687.

- [38] *Object Tracking and Classification for Multiple Active Safety and Comfort Applications Using a Multilayer Laser scanner*. Fuerstenberg, K et Dietmayer, K. 2004. IEEE Intelligent Vehicle Symposium. pp. 802-807.
- [39] *Pedestrian Recognition in Urban Traffic Using Laser Scanners*. Fuerstenberg, K et Willhoeft, V. 2001. 8th World Congress on Intelligent Transport Systems. p. 551.
- [40] *Reliable Pedestrian Protection Using Laser Scanners*. Fuerstenberg, Kay C et Scholz, J. Las Vegas : s.n., june 2005. Proceedings of 2005 IEEE Intelligent Vehicles Symposium.
- [41] Zhao, L. *Dressed Human Modeling, Detection and Parts Localization*. The Robotics Institute, Carnegie Mellon University. july 2001. Ph.D Thesis.
- [42] Shashua, A, Gdalyahu, Y et Hayun, G. *Pedestrian Detection for Driving Assistance Systems: Single-frame Classification and System Level Performance*. s.l.: IEEE Intelligent Vehicle Symposium, 2004.
- [43] *Pedestrian Detection with Radar and Computer Vision*. S. Milch and M. Behrens. Septembre 2001, Proceedings of Pal 2001 - Progress in Automobile Lighting, held laboratory of lighting technology, Vol. 9. ISBN: 3-89675-971-X.
- [44] Ling, Christopher. *Image Detection Techniques on Daimler Pedestrian Monocular Data*. 10 Septembre 2017.
- [45] Variyar, Anil. *Application of Convolutud Neural Networks for Pedestrian Detection*. San Francisco : Stanford University.
- [46] *Neural Network-Based face Detection*. Rowley, A, Baluja, S et Kanade, T. January 1998. IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [47] *2nd International Workshop on Statistical and computational theories of vision, modeling ,learning, computing and sampling*. Viola, P et Jones, M. 2001.
- [48] *Histograms of oriented gradients for human detection*. Dalal, N et Triggs, B. 2005. IEEE Conference on Computer Vision and Pattern Recognition.
- [49] *Pedestrian Detection: An Evaluation of the State of the art*. Dollar, P, et al. August 2011. IEEE Transactions on Pattern Analysis and Machine Intelligence. ISSN:0162-8828.
- [50] *Pedestrain Detection: A Benchmark*. Dollar, P, et al. 2009. IEE Conference on Computer Vision and Pattern Recognition.

- [51] Wikipedia. [En ligne] [https://en.wikipedia.org/wiki/Feature_detection_\(computer_vision\)](https://en.wikipedia.org/wiki/Feature_detection_(computer_vision)).
- [52] *Image Net classification with deep convolutional neural networks*. Krizhevsky, A, Sutskever, I et Hinton, G. 2012. Advances in neural information processing systems.
- [53] *Going Deeper with Convolutions*. Szegedy, C, et al. 2014. ILSVRC.
- [54] Deshpande, Adit. The 9 Deep Learning Papers You Need To Know About (Understanding CNNs Part 3). *Disqus*. [En ligne] septembre 2017.
- [55] Migniot, Cyrille. *Segmentation de personnes dans les images et les videos* ´. Grenoble : Université de Grenoble, 2012. p. 180, Thèse de doctorat.
- [56] Mallick, Satya. *Histogram of Oriented Gradients*. 6 December 2016.
- [57] Weisstein, Eric. *L²-Norm*. s.l. : MathWorld.
- [58] Cortes, Corinna et Vapnik, Vladimir. *Support-Vector Networks*. [éd.] Lorenza Saitta. Boston : Kluwer Academic Publishers, 1995.
- [59] Mechalik, youcef et Milouddi, Ali. *Développement d'algorithmes d'évitement d'obstacles statiques et dynamiques*. s.l. : Faculté des Sciences de Technologies et Sciences de Matières, Université Kasdi Merbah–Ouargla, 2012. p. 92, Projet de fin d'étude.
- [60] *Capteurs et méthodes pour la localisation des robots mobiles*. Marie-José, Aldon. s.l. : traité Informatique industrielle, 1998, Techniques de l'Ingénieur.
- [61] Bayle, B. *Robotique Mobile*. Université Louis Pasteur. s.l. : Ecole Nationale Supérieure de Physique de Strasbourg, 2007.
- [62] Shuzhi, S et Lewis, F. *Autonomous Mobile Robots, Sensing, Control, Decision, Making and Applications*. s.l. : Taylor and Francis Group, 2006.
- [63] Cherroun, Lakhmissi. *Navigation Autonome d'un Robot Mobile par des Techniques Neuro-Floues*. Université Mohamed Khider - Biskra. s.l. : Faculté des Sciences et de la Technologie, 2014. p. 129, Thèse de doctorat.
- [64] Gauthier, E. *Utilisation des Réseaux de neurones pour la commande d'un véhicule autonome*. Grenoble : s.n., 1999. Thèse de doctorat.
- [65] Parent, O et Eustache, Julien. *Les réseaux bayésien, à la recherche de la vérité*. Lyon : Université Claude Bernard Lyon 1, 2006.

- [66] Garnier, E. *Contrôle d'exécution réactif de mouvements de véhicules en environnement dynamique et structuré*. s.l. : Institut national polytechnique de Grenoble, 1995. Thèse de doctorat.
- [67] *Real-Time Obstacle Avoidance for Manipulators and Mobile Robots*. Khatib, O. 1986, The International Journal of Robotics Research, Vol. 5.
- [68] Belkhouche, Ismail. *Planification de trajectoire et évitement d'obstacle par les réseaux de neurones pour les robots*. Université Abou Bakr Belkaid– Tlemcen. s.l. : Faculté des sciences, 2011. p. 118.
- [69] Zadeh, Lotfi. Fuzzy Sets. *Information and Control*. 1965, Vol. 8, pp. 338-353.
- [70] *Control of a nonholonomic mobile robot using neural networks*. Fierro, R et Lewis, F. 4, 1998, IEEE Transactions on neural networks, Vol. 9, pp. 589-600.
- [71] *Reinforcement learning: an overview*. Glorennec, P.Y. Aachen, Germany : s.n., Septembre 2000. ESIT 2000.
- [72] Ferber, J. *Les Systèmes Multi Agents: Vers une Intelligence Collective*. Université Pierre et Marie Curie . Paris : InterEdition, 1995.
- [73] Anderson, J et Rosenfeld, E. *Neurocomputing: Foundations of Research*. cambridge : MIT Press, 1988.
- [74] Sutton, R et Barto, A. *Reinforcement Learning: An Introduction*. Cambridge : MIT Press, 1998.
- [75] *Reinforcement Learning: A Survey*. Kaelbling, L, Littman, M et Moore, A. 1996, Journal of Artificial Intelligence Research, Vol. 4, pp. 237-285.
- [76] Bellman, R.E. *Dynamic Programming*. Princeton, New Jersey, USA : Princeton University Press, 1957.
- [77] *Apprentissage par renforcement, Notes de cours*. Preux et P. s.l. : GRAPPA, octobre 2007. cours de l'université de Lille.
- [78] *Self Improvement based on Reinforcement Learning, Planning and Teaching*. Lin, L. 1991. Proc. of 8th Workshop on Machine Learning.
- [79] *Learning and Tuning Fuzzy Logic Controllers Through Reinforcements*. Khedkar, H. R. Berenji and P. 5, 1992, IEEE Transactions on Neural Networks, Vol. III, pp. 724-740.

- [80] *Neural Reinforcement Learning for Behaviour Synthesis*. Touzet, C. 3, 1997, Robotics and Autonomous Systems, Vol. 22, pp. 251-281.
- [81] *Fuzzy Q-learning*. Glorennec, P et Jouffe, L. 1997. 6th IEEE International Conference on Fuzzy Systems. pp. 659-662.
- [82] *Vehicle steer-by-wire system control*. Yao, Yixin. Detroit : SAE International, 2006. SAE Technical Paper Series.
- [83] *Brake-by-wire system development*. Colombo, Davide. Torino : FIAT Group, 2007. Technology and development process.
- [84] Paret, Dominique et Rebaine, Hassina. *Réseaux de communication pour systèmes embarqués*. 2, Paris : Dunod, 2005. ISBN 978-2-10-070700-3.
- [85] Audi AG. *Audi A5 -Multiplexage*. 2007. Programme autodidactique 395.
- [86] Mallick, Satya. Image Recognition and Object Detection : Part 1. *Learn OpenCV*. [En ligne] 14 Novembre 2016. <http://www.learnopencv.com/>.
- [87] EXXOTEST. *Analyse des réseaux multiplexes*. Module pédagogique.
- [88] Wilov. [En ligne] 7 Mars 2017. [Citation : 28 Octobre 2017.] <https://blog.wilov.com/portfolio/10-voitures-plus-rapides-au-monde/>.
- [89] *Emergency steer & brake assist – a systematic approach for system integration of two complementary driver assistance systems*. Eckert, Alfred, et al. [éd.] Continental AG.
- [90] *Challenges and Approaches to Realizing Autonomous Vehicle Safety*. Nidhi, Kalra. [éd.] RAND Corporation. Santa Monica, USA : s.n., 2017. RAND Testimonies.
- [91] Green, Marc. Reaction time. *Institute of transportation engineers expert witness council newsletter*. 2005, pp. 2-6.
- [92] wikipedia. [En ligne] [Citation : 08 Novembre 2017.] <http://www.wikipedia.org>.
- [93] Google Image. *Google*. [En ligne] [Citation : 18 Décembre 2017.] <http://www.google.com/imghp?hl=fr>.
- [94] NVIDIA. Véhicules autonomes. [En ligne] [Citation : 15 Novembre 2017.] <https://www.nvidia.fr/self-driving-cars/drive-px/>.
- [95] Parizeau, M. *Réseaux de neurones*. s.l. : Edition université de Laval, 2004.

- [96] Anderson, J. *An Introduction to Neural Networks*. Bradford : MIT Press, 1995.
- [97] Godjevac, J. *Neuro-Fuzzy Controllers, Design and Application*. Presses Polytechniques et Universitaires Romandes. Lausanne : Collection Meta, 1997.
- [98] James, A et Skapura, David. *Neural Networks Algorithms, Applications and Programming Techniques*. s.l. : Addison Wesley, 1992.
- [99] Touzet, C. *Les Réseaux de Neurones Artificiels, Introduction au Connexionnisme*. Paris : Masson, 1992.
- [100] Personnaz, I. *Réseaux de Neurones Formels pour la Modélisation, la Commande et la Classification*. Paris : CNRS Edition, 2003.
- [101] Centre de recherche LGI2P. *Groupe heuristiques et systèmes complexes*. Alès : s.n. Réseau de neurones formels commande de processus.
- [102] Aiache, Jacques et Chereau, Jean-Marc. Klubprepa. [En ligne] [Citation : 18 08 2017.] <http://www.klubprepa.fr>.
- [103] Giard, Vincent. *Gestion de la production et des flux*. Toulouse : Economica, 2003.
- [104] Tarondau, J. *La flexibilité dans les entreprises*. Paris : P.U.F, 1999.
- [105] manuals, online. *Minimum_designs_truck_bus_turns*.
- [106] Google. google image. [En ligne] [Citation : 18 décembre 2017.] <http://www.google.com/imghp?hl=fr>.
- [107] Jose Antonio Martin H, *Dyna-Q Maze Demonstration Programmed in Matlab*, [En ligne] <https://github.com/pilami/DDP/blob/master/Dyna%20Q/MazeDemo.m>

ANNEXES

ANNEXE 1 : Les différentes stratégies de choix d'action [63]

Annexe 1.a : Méthode Gloutonne (greedy) : Consiste toujours à choisir l'action estimée comme la meilleure, soit :

$$a_{\text{gloutonne}} = \operatorname{argmax}_{a \in A(s)} Q(s_t, a)$$

Annexe 1.b : Méthode e-gloutonne (e-greedy) : Consiste à choisir l'action gloutonne avec une probabilité e et à choisir une action au hasard avec une probabilité $1-e$, soit :

$$a_{e\text{-gloutonne}} = \begin{cases} \operatorname{argmax}_{a \in A(s)} Q(s_t, a) & \text{avec probabilité } e \\ \text{action prise au hasard dans } A(s_t) & \text{avec probabilité } 1 - e \end{cases}$$

Annexe 1.c : Méthode Softmax : Consiste à choisir une action en fonction de sa qualité par rapport à la qualité des autres actions dans le même état, on associe à chaque action la probabilité d'être émise selon l'équation suivante :

$$Pr[a_t | s_t] = \frac{Q(s_t, a)}{\sum_{a \in A(s_t)} Q(s_t, a)}$$

Annexe 1.b : Méthode Boltzmann : c'est un cas particulier de sélection softmax: la probabilité est calculée avec l'équation suivante qui produit une distribution dite de Boltzmann:

$$Pr[a_t | s_t] = \frac{e^{\frac{Q(s_t, a)}{\tau}}}{\sum_{a \in A(s_t)} e^{\frac{Q(s_t, a)}{\tau}}}$$

ANNEXE 2 : Algorithme de Q-learning [63]

$Q(s,a) \leftarrow 0, \forall (s,a) \in (S,A)$

Pour un grand nombre **faire**

Initialiser l'état initial s_0

$t \leftarrow 0$

Répéter

Choisir l'action à émettre a_t en fonction de la politique de Q

Emettre l'action choisie.

Observer r_t et s_{t+1}

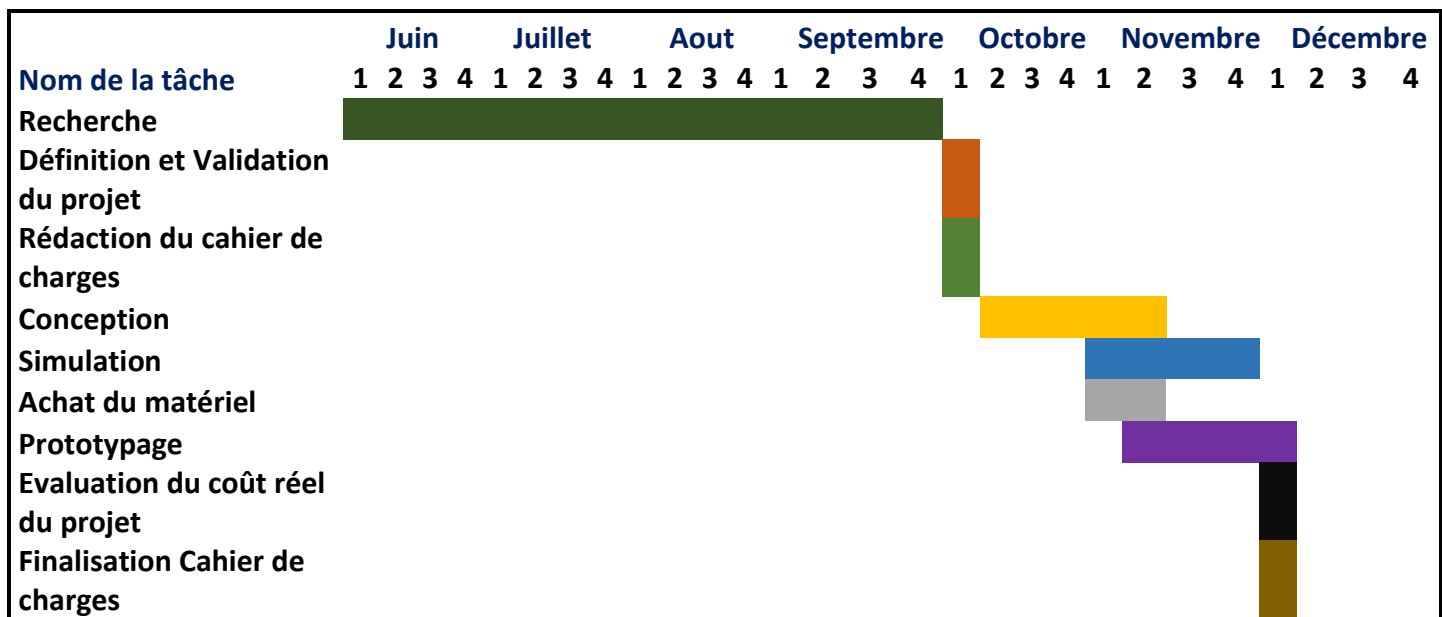
$Q(s_t, a_t) \leftarrow Q^x(s_t, a_t) + \beta[r_t + \gamma \max_{a' \in A(s_{t+1})} Q(s_{t+1}, a') - Q(s_t, a_t)]$

$t \leftarrow t + 1$

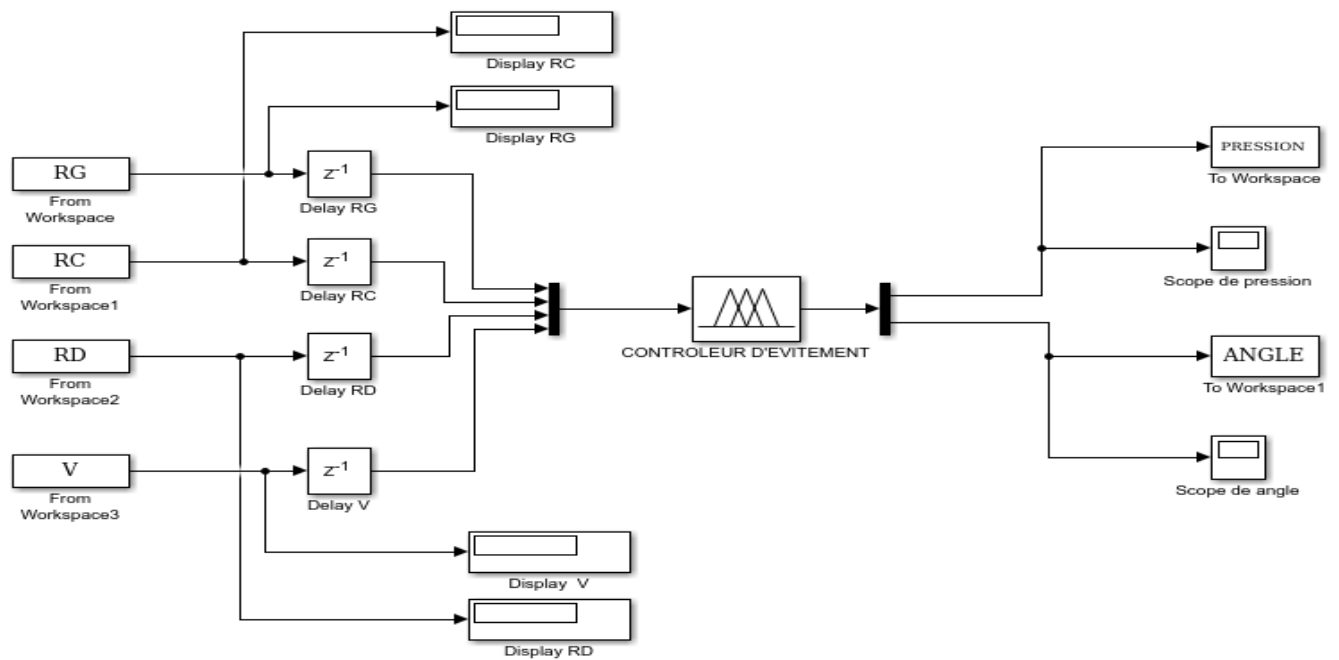
Jusqu'à $s_t \in F$

Fin pour

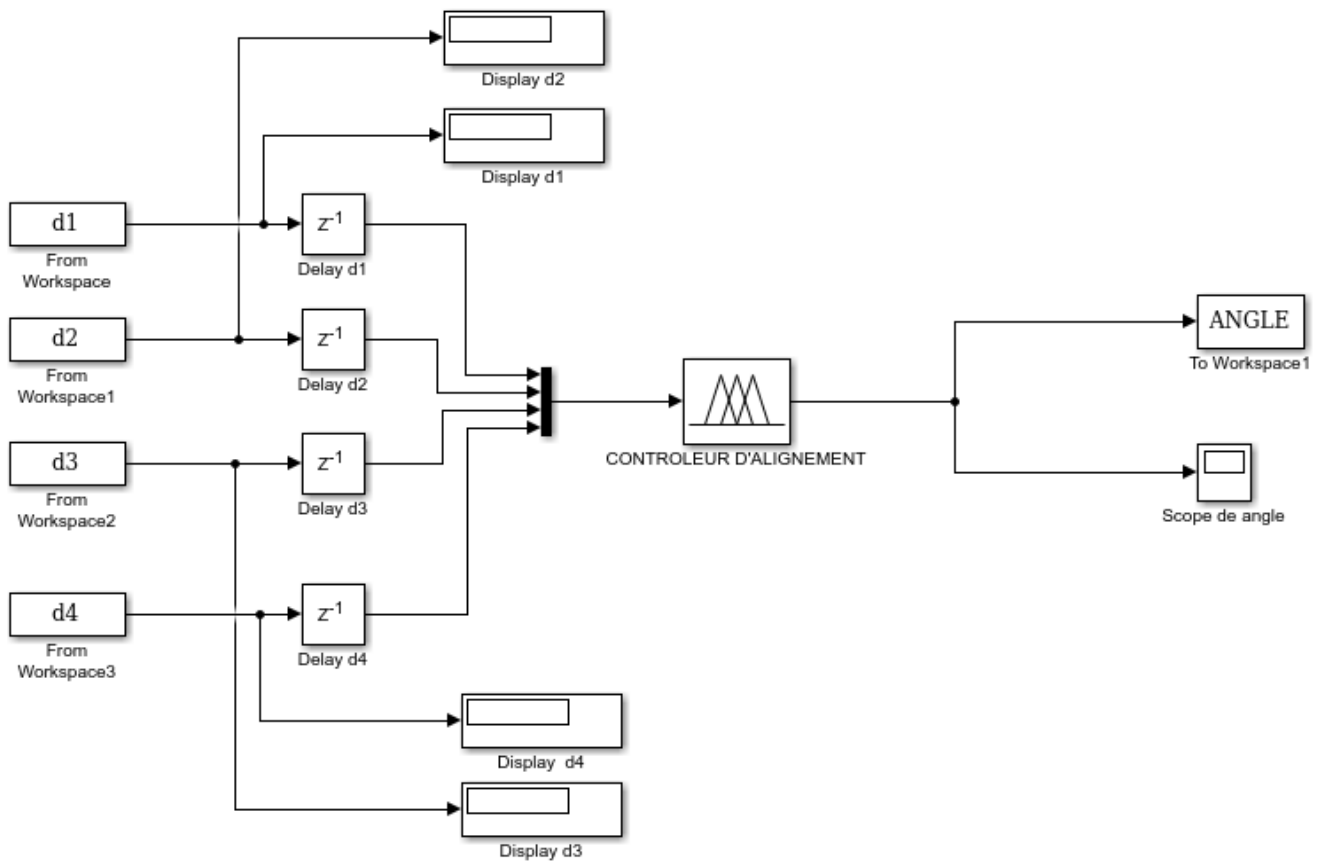
ANNEXE 3 : Diagramme Gantt du projet



ANNEXE 4 : Modèle SIMULINK de simulation du contrôleur flou d'évitement

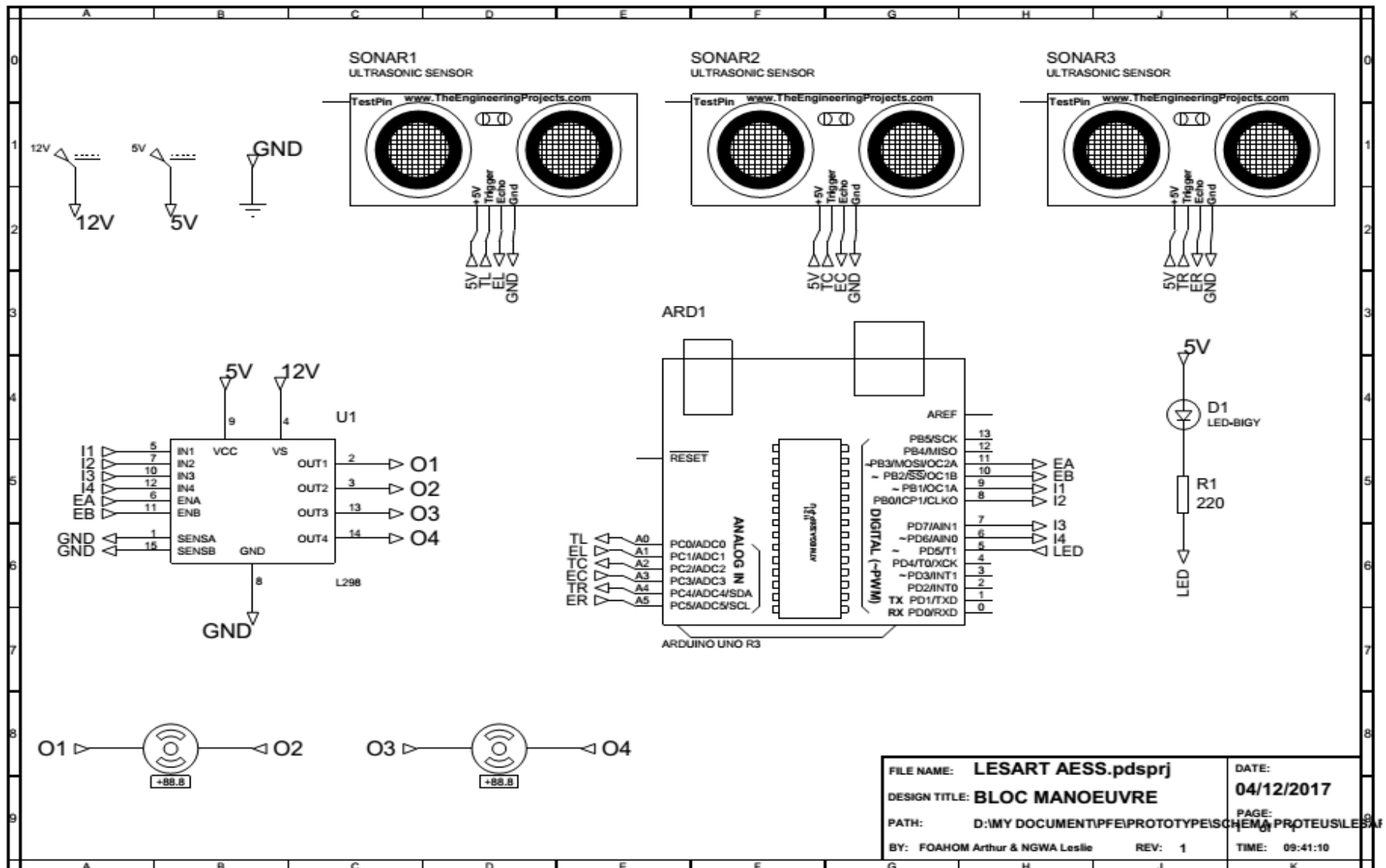


ANNEXE 5 : Modèle SIMULINK de simulation du contrôleur flou d'alignement

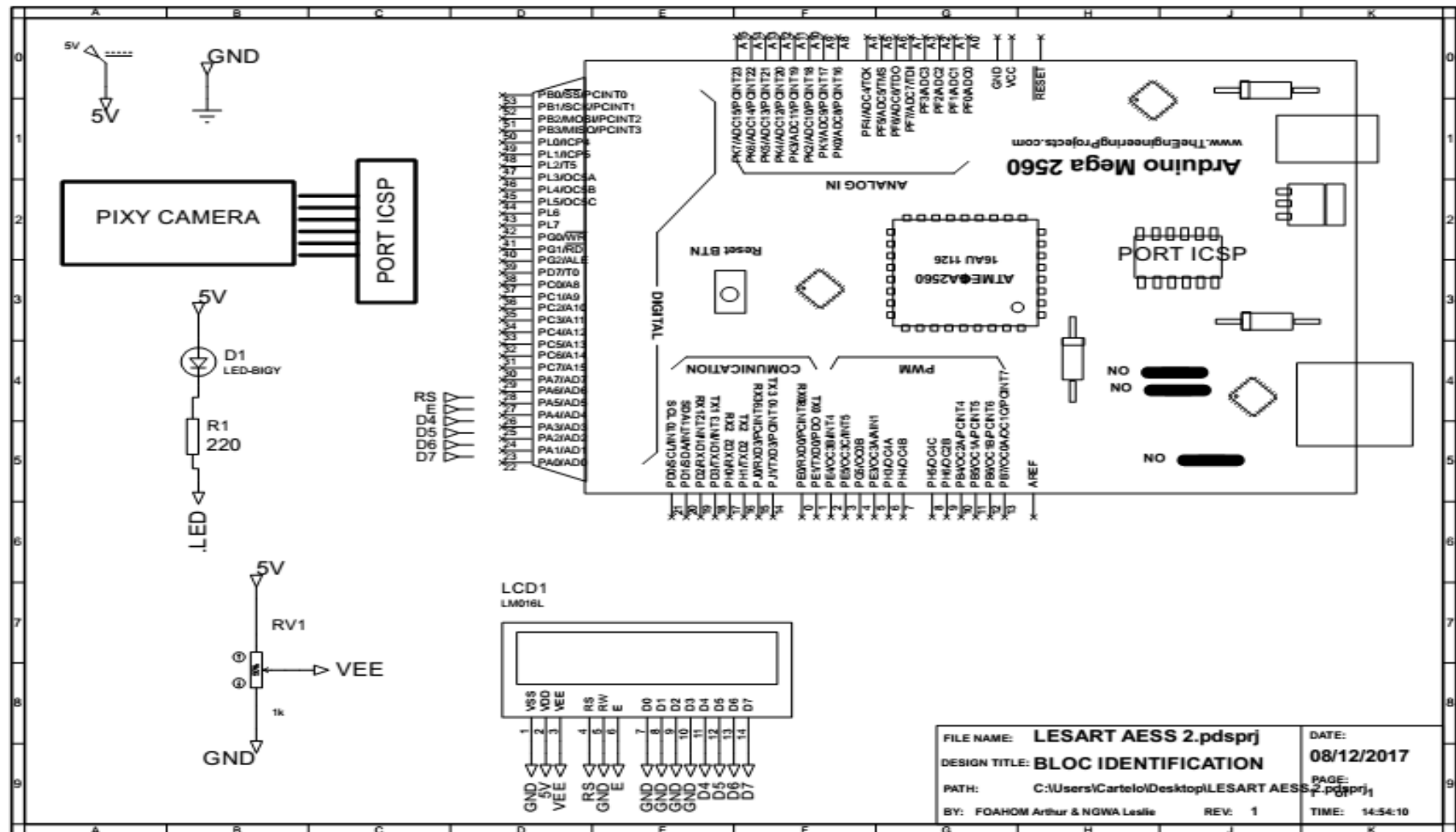


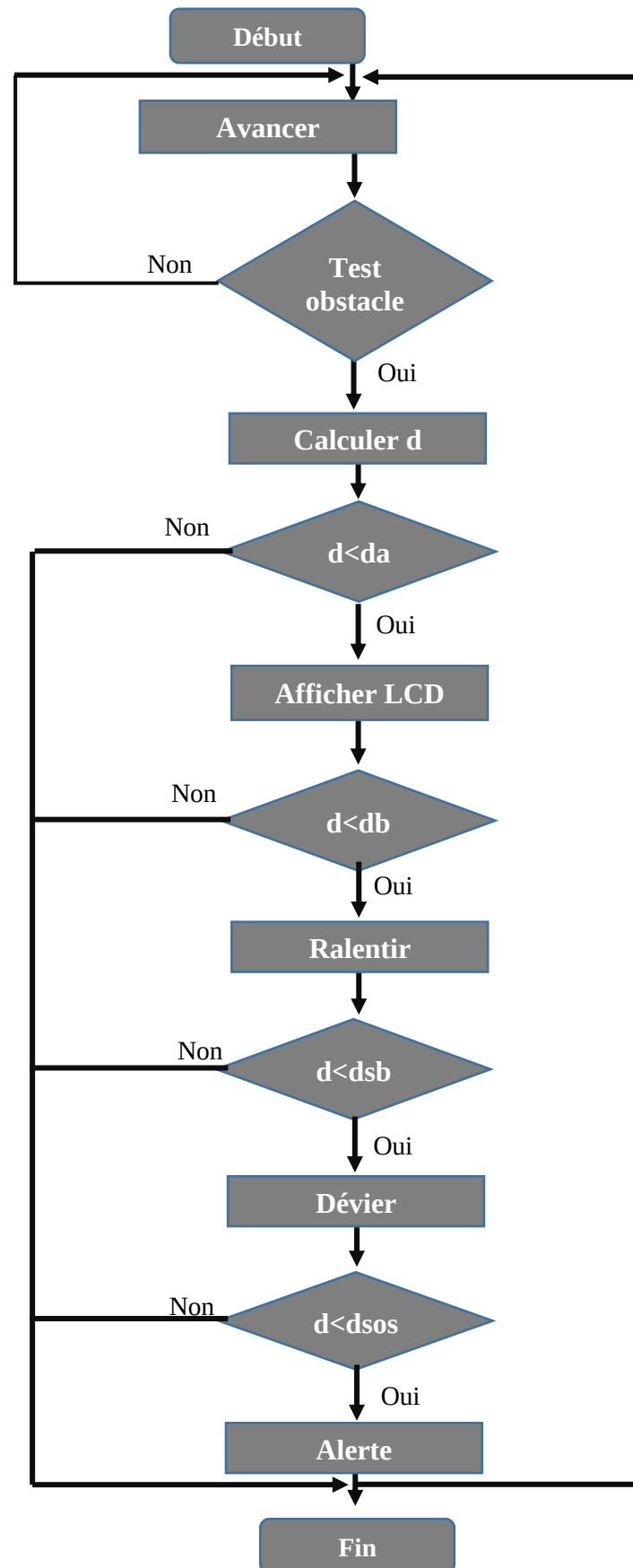
ANNEXE 6 : Circuits électriques du prototype

Annexe 6.a : Schéma électronique bloc manœuvre



Annexe 6.b : Schéma électronique bloc identification



ANNEXE 7 : Organigramme programme du prototype

ANNEXE 8 : Budget prototype

Capteurs	Caractéristiques	Quantité	Prix unitaire(FCFA)	Prix(FCFA)
Microcontrôleur	Arduino uno (8k) et Mega (15k)	1	15000+8000	23 000
Résistance		8	25	200
Led		3	50	150
Plaque à essai		1	1 500	1 500
Connecteurs	60 cables	1	2 500	2 500
Chassis + moteurs	2 moteur CC, 1 chassis, 3 roues et 1 frame	1	12 000	12 000
Caméra Pixy	Pixy (Cmucam5)	1	70 720	70 720
Capteur ultrason	HC SR004	3	2 000	6 000
Module bluetooth	HC-05	1	5 000	5 000
Batterie	5v-9V	6	500	3 000
Pont H	L298N	1	3 000	3 000
Contre-plaqué		1	3 500	3 500
Peinture	Blanche noir et rouge	2	1500	3 000
LCD	LCD1602	1	4000	4000
Pinceau		1	1	500
Pétrole		1L	650	650
Potentiomètre		2	500	1 000
Vis/ Écrou		1 paquet	500	500
colle	Plastic	1	300	300
Total				140 520

ANNEXE 9 : Cahier des charges du projet

ANNEXE 10 : Photos du prototype



Photo du bloc d'identification du prototype

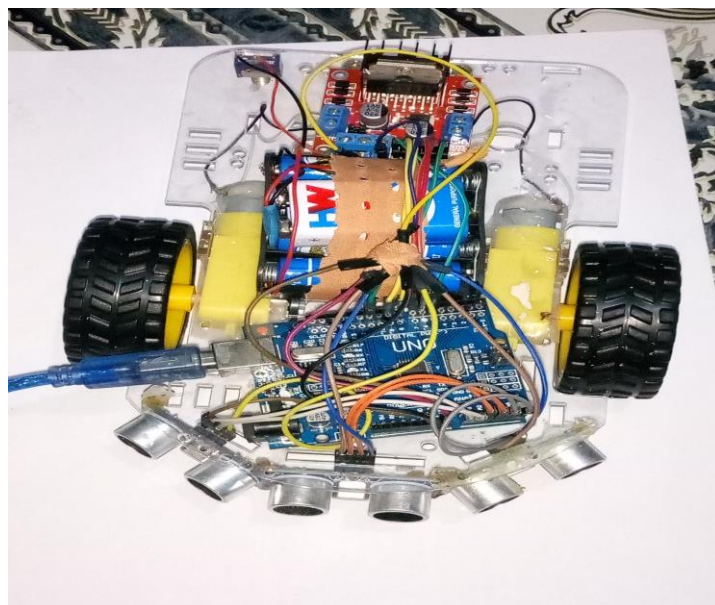


Photo du bloc d'évitement du prototype



Photo vue d'ensemble du prototype

TABLE DES MATIÈRES

DÉDICACE	I
REMERCIEMENTS	IV
RÉSUMÉ.....	V
ABSTRACT.....	VI
SOMMAIRE	VII
LISTE DES TABLEAUX.....	VIII
LISTE DES FIGURES.....	IX
LISTE DES SIGLES ET ABRÉVIATIONS	XI
LISTE DES ANNEXES.....	XIII
INTRODUCTION GENERALE	1
PARTIE 1: REVUE DE LITTERATURE.....	3
CHAPITRE 1 : LES ADAS ET LES METHODES DE CONCEPTION DES SE.....	4
1.1 Les systèmes d’aide à la conduite pour véhicule	4
1.1.1 Les véhicules autonomes.....	4
1.1.2 Les systèmes avancés d’aide à la conduite.....	5
1.2 Conception des systèmes embarqués	9
1.2.1 Historique des SE et présentation des types de conception	10
1.2.2 Présentation des méthodes et outils de conception de système.....	12
CHAPITRE 2 : RECONNAISSANCE DE PIETONS ET MÉTHODOLOGIE DE NAVIGATION AUTONOME.....	17
2.1 Les méthodes d’identification de piétons.....	17
2.1.1 Les technologies des capteurs [27].....	17
2.1.2 Les méthodes d’identification de piéton par vision basées sur des caractéristiques de forme 22	
2.2 L’intelligence artificielle appliquée à la navigation.....	28
2.2.1 Les approches d’évitement d’obstacle	28

2.2.2	La logique floue	29
2.2.3	L'apprentissage automatique.....	36
PARTIE 2: MATERIELS ET METHODES.....		45
CHAPITRE 3 : ETUDE DE CONCEPTION.....		46
3.1	Présentation du système	46
3.1.1	Présentation des entrées/sorties du système	46
3.1.2	Gestion des capteurs.....	48
3.2	Conception du système.....	50
3.2.1	Conception du bloc de perception (BP)	50
3.2.2	Conception du bloc de décision	55
3.2.3	Conception bloc de commande	67
CHAPITRE 4 : OUTILS ET MATERIELS		70
4.1	Description des outils	70
4.1.1	Environnement Matlab	70
4.2	Présentation du matériel	70
4.2.1	Les radars	70
4.2.2	Caméra stéréovision	71
4.2.3	ECU.....	72
PARTIE 3 : RESULTATS ET DISCUSSION.....		75
CHAPITRE 5 : SIMULATIONS ET DISCUSSIONS.....		76
5.1	Simulation de l'identification d'un piéton	76
5.2	Simulation de l'intelligence du système.....	77
5.2.1	Simulation contrôleur flou d'évitement	77
5.2.2	Simulation algorithme Q-learning de renforcement.....	82
5.2.3	Simulation contrôleur flou d'alignement	83

CHAPITRE 6 : REALISATION DU PROTOTYPE	86
6.1 Présentation du prototype.....	86
6.2 Outils et matériels utilisés	87
6.2.1 Présentation des outils.....	87
6.2.2 Description du matériel	88
6.3 Réalisation.....	91
CONCLUSION GENERALE	93
BIBLIOGRAPHIE	94
ANNEXES	XIV
TABLE DES MATIÈRES	XXXVIII