



HAL
open science

Localisation visuelle par approche neuromimétique dans le contexte de la délégation de conduite

Yoan Espada

► **To cite this version:**

Yoan Espada. Localisation visuelle par approche neuromimétique dans le contexte de la délégation de conduite. Automatique / Robotique. CY Cergy Paris Université, 2020. Français. NNT : 2020CYUN1070 . tel-03274926

HAL Id: tel-03274926

<https://theses.hal.science/tel-03274926>

Submitted on 30 Jun 2021

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.



UNIVERSITÉ PARIS SEINE
École doctorale science et ingénierie

THÈSE

en vue de l'obtention du grade de

Docteur d'Université

en navigation robotique et science cognitive

Présentée et soutenue par

Yoan ESPADA

Localisation visuelle par approche neuromimétique dans le contexte de la délégation de conduite

préparée au laboratoire ETIS en partenariat avec l'institut VEDECOM
soutenue le ...

Jury :

Rapporteurs : Olivier ORFILA
David FILLIAT - Président

Directeur : Olivier ROMAIN

Examineurs : Samia AINOUS
Samia BOUCHAFA-BRUNEAU

Encadrants : Nicolas CUPERLIER
Guillaume BRESSON



Abstract

The main aim of this thesis is to extend navigation models built for indoor environments to autonomous vehicles. These models follow a neurorobotics approach : they reproduce neural substrates that sustain the mechanism responsible for spatial cognition in the mammalian brain. This thesis introduces and shows the potential of bio-inspired robotics and neurorobotics to the community of autonomous driving.

This manuscript begins by giving a state of the art of robotic navigation. It presents the subsidiary question of visual recognition.

The notions of bio-inspired robotics and neurorobotics are then introduced. An overview of anatomical observations necessary to the comprehension of bio-inspired models is also given. The document describes a model for visual place recognition inspired by hippocampal place cells. This model is tested in outdoor on-road environments. The results show that the model is able to build a representation of the environment that can be used in navigation tasks.

A reactive navigation model (PerAc) is then presented. This model uses the representation built by the visual place recognition model to learn sensorimotor units linking places and actions. A set of these associations allows the model to learn robust trajectories or goal positions. This model was initially built for robots evolving in indoor environments. It is embedded and tested on an autonomous vehicle. The experiments show that the model is able to learn trajectory with a car in outdoor environments.

My last contribution presented is about a visual contexts model. A second visual processing chain inspired by the primary visual cortex is proposed to build a coarse representation of the environment that modulates the activities of the visual place cells.

The essay ends by giving a set of leads to improve different aspects of the proposed models, by modeling in more depth other brain structures or change the encoding of data for instance.

Keywords : autonomous driving, neurorobotics, bio-inspiration, visual place recognition, reactive navigation

Résumé

L'objectif premier de cette thèse est de tester et d'étendre des modèles de navigation initialement développés pour de la navigation en intérieur et de les appliquer à la délégation de conduite des véhicules autonomes. La particularité de ces architectures de contrôle pour robot mobile est qu'elles sont issues d'une approche bio-inspirée et sont dite neurorobotiques : elles reproduisent les mécanismes neuro-naux impliqués dans les comportements de navigation observés chez les animaux. Ces travaux concourent à introduire auprès de la communauté du véhicule autonome les approches bio-inspirées et leur potentiel au travers des modèles présentés.

Cette thèse commence par aborder les technologies utilisées (GPS, lidar, ...) et les différentes stratégies de navigation (réactive, avec carte). Cette présentation se termine en circonscrivant le problème de la navigation des véhicules autonomes à la reconnaissance de lieux par la vision.

Le manuscrit présente ensuite les approches bio-inspirée et neurorobotique avant de donner les notions d'anatomie facilitant la compréhension des modèles bio-inspirés décrits par la suite. Un modèle de reconnaissance de lieux par la vision est présenté en détail. Des expériences de simulations avec des données réelles permettent de tester ce modèle, initialement conçu pour la navigation intérieure, dans des environnements propres aux véhicules. Les résultats de ces tests montrent que le modèle construit une représentation de l'environnement capable de discriminer et de reconnaître des lieux.

Une architecture neuronale de navigation réactive reposant sur le précédent système, le modèle PerAc est décrite ensuite. Ce modèle utilise les lieux construits par le modèle de reconnaissance et associe chaque lieu à une direction à suivre. Par un ensemble d'associations, il est possible d'apprendre à suivre de manière robuste des trajectoires ou une position à rejoindre. Ce modèle, conçu pour de la navigation en intérieur, est testé sur un véhicule dans des expériences servant à évaluer sa capacité à fonctionner sur véhicule. Les résultats prouvent que le modèle est capable d'apprendre des trajectoires avec une voiture en environnement extérieur. Ma dernière contribution porte sur un modèle de contextes visuels permettant d'accroître les performances computationnelles de la reconnaissance de lieux. Une chaîne de traitements visuelle inspirée du cortex visuel primaire effectue un découpage de l'environnement qui vient moduler la reconnaissance de lieux du modèle précédent.

Ce manuscrit se termine en proposant un ensemble de pistes à même d'améliorer les différents aspects des modèles présentés, par exemple, en modélisant plus finement certaines zones du cerveau ou en utilisant un codage épars afin de réduire le coût computationnel.

Mots clefs : véhicules autonomes, neurorobotique, bio-inspiration, reconnaissance de lieux par la vision, navigation réactive

Remerciements

Je remercie, tout naturellement, mes encadrants de thèse Nicolas Cuperlier et Guillaume Bresson ainsi que mon directeur de thèse Olivier Romain.

Je remercie également l'ensemble des personnels de l'intitut VEDECOM et du laboratoire ETIS et plus particulièrement chacun des membres des équipes VEH08 et Neurocyber. Je tiens aussi à remercier tout spécialement Victor Bares dont la contribution a été primordiale pour la réalisation des expériences sur véhicule.

Pour terminer en beauté, je commence par remercier ma fiancée Adeline Govehovitch qui a été un soutien de tous les instant et sans qui je n'aurais probablement pas pu terminer l'écriture de ce manuscrit. Et enfin je remercie ma fille, Nausicaa sans qui ce manuscrit aurait été terminé quelques mois plus tôt.

Table des matières

1	Introduction	1
2	La navigation en robotique	5
2.1	La perception	8
2.1.1	La proprioception	8
2.1.2	La géolocalisation	9
2.1.3	La reconnaissance de lieux visuelle	10
2.1.4	Les télémètres	15
2.1.5	Synthèse	16
2.2	La navigation sans carte	17
2.2.1	La navigation vers un but	17
2.2.2	Les systèmes réactifs	18
2.2.3	Les méthodes de suivi	18
2.2.4	Les modèles de subsumption	19
2.2.5	L'algorithme des fourmis	20
2.2.6	L'intégration de chemin	20
2.2.7	Les méthodes de bout-en-bout	21
2.2.8	Applications actuelles	22
2.3	La navigation avec carte	22
2.3.1	Les cartes métriques	23
2.3.2	Les cartes topologiques	24
2.3.3	Les cartes hybrides	25
2.3.4	Les cartes cognitives	26
2.4	Le SLAM	26
2.5	Discussion	27
3	Observations biologiques	29
3.1	La modélisation issue de la biologie	31
3.1.1	La bio-inspiration	31
3.1.2	La neurorobotique	32
3.2	Observations	33
3.2.1	La vision et l'œil	33
3.2.2	Le cortex visuel et le corps géniculé latéral	38
3.2.3	La zone Hippocampique	42
3.3	Modèles bio-inspirés appliqués à la navigation de véhicules autonomes	50
3.3.1	Les réseaux de neurones profonds	50
3.3.2	RatSLAM	52
3.3.3	Modèle de mémoire visuelle	54
3.3.4	Modèle de carte cognitive de Zeng et Si	56
3.3.5	Beobot	57

3.4	Discussion	59
4	Modèle de cellules de lieu visuelles	61
4.1	Description du modèle	61
4.1.1	La chaîne de traitement visuel	62
4.1.2	La partie neuronale	64
4.1.3	Les mécanismes d'apprentissage	69
4.2	Expérimentations	72
4.2.1	Jeux de données	73
4.2.2	Méthodologie	75
4.2.3	Métriques	76
4.2.4	Résultats	80
4.3	Discussion	86
4.3.1	Comparaison à l'état de l'art bio-inspiré	86
4.3.2	Perspectives	90
5	Modèle PerAc	93
5.1	Description du modèle	93
5.1.1	Principe	94
5.1.2	Contrôle neuronal	95
5.1.3	Implémentation	98
5.1.4	Adaptation au véhicule	99
5.2	Expériences	103
5.2.1	Dispositif expérimental	104
5.2.2	Résultats	105
5.3	Discussion	108
6	Modèle de contexte visuel	111
6.1	Description du modèle	113
6.1.1	La chaîne de contexte visuel	113
6.1.2	Interactions avec le modèle de cellules de lieu visuelles	116
6.2	Expérimentations	117
6.2.1	Dispositif expérimental	117
6.2.2	Résultats	118
6.3	Discussion	124
7	Conclusion et perspectives	127
	Bibliographie	135

Table des figures

2.1	Un exemple de représentation obtenue par un lidar multi-couches Velodyne.	15
2.2	L'architecture d'un véhicule de Braitenberg avec les comportements d'évitement d'obstacle (2a) et de homing (2b).	19
2.3	Le principe de l'intégration de chemin. Les segments en bleu représentent le chemin parcouru. Le segment en rouge représente l'intégration de chemin qui donne une estimation de la position relative du point de départ au point courant.	21
2.4	Un exemple de représentation sous forme de grille d'occupation avec, en rouge, le trajet parcouru par le robot[Bellekens <i>et al.</i> 2015].	23
2.5	Un exemple de carte topologique construite à partir d'une grille d'occupation [Blöchlinger <i>et al.</i> 2017].	24
2.6	Le plan du métro parisien, un exemple de carte topologique.	25
3.1	Diagramme schématique d'un oeil humain	34
3.2	Répartition des cônes (en bleu) et des bâtonnets (en noir) par rapport au centre de la fovea. La zone sans photorecepteur correspond à la papille, la zone où commence le nerf optique.	35
3.3	Schéma détaillant les quatre couches constituant la rétine ainsi que les six types de neurones (batonnets, cônes, cellules horizontales, cellules bipolaires, cellules amacrines et cellules ganglionnaires) formant ces couches [Purves 2001]	36
3.4	Diagramme donnant l'angle d'orientation des yeux pour différents espèces de mammifères	37
3.5	Le corps géniculé latéral avec les trois voies (parvocellulaire, koniocellulaire et magnocellulaire) de projection depuis la rétine [Nieuwenhuys 2008].	39
3.6	La position du cortex visuel primaire, zone 17 de Broadman, dans le cerveau humain.	40
3.7	Le modèle dorsoventral de transmission des informations depuis le cortex visuel primaire (en gris) vers les zones dorsales (en vert) et ventrales (en violet)	42
3.8	La formation hippocampique dessinée par Santiago Ramon y Cajal. EC : Cortex Entorhinal, DG : Gyrus Dentelé , CA1-CA3 : Cornes d'Ammons (hippocampe), sub : Subiculum [Ramon 1909].	43

3.9	Un exemple de dispositif de mesure ainsi que les principaux types de neurones identifiés impliqués dans la cognition spatiale A - Exemple de dispositif utilisé pour mesurer des activités neuronales. Une caméra zénitale enregistre les déplacements du rat sur lequel est implanté une électrode. B - Un enregistrement de l'activité d'une cellule de lieu dans une arène. L'image de gauche représente le trajet parcouru par le rat en noir avec les pics d'activation dessinés en rouge. L'image de droite représente l'activité moyenne du neurone dans cette même arène. Les zones en bleu correspondent à une activité quasi nulle et plus on se rapproche du rouge plus l'activité est élevée. C - Enregistrement d'activités d'une cellule de direction de la tête. A gauche, un amer situé au nord provoque une activation du neurone quand l'animal est orienté vers le sud. A droite, déplacement de l'amer vers l'est, le neurone répond quand l'animal est orienté vers l'ouest. D - Un enregistrement d'activité de cellules de grille selon la même méthode de représentation que pour la cellule de lieu (A) [Grievés & Jeffery 2016]	44
3.10	Exemple d'activité d'une cellule de lieu. La zone de plus forte activité sur l'image de droite (en rouge) correspond au champ de lieu [Grievés & Jeffery 2016]	45
3.11	Activité d'une cellule de direction de la tête pour une condition initiale (Standard 1), une rotation d'amer de 180 degrés (180 degree Cue Rotation) et enfin un retour aux conditions initiales (Standard 2) [Taube & Burton 1995].	48
3.12	Enregistrement d'une cellule de grille dans le cortex entorhinal. On peut voir le motif défini par cette cellule [Grievés & Jeffery 2016].	49
3.13	Schéma de fonctionnement du modèle. Il extrait un vecteur de features à partir d'images d'entrée avant d'utiliser la matrice de confusion et deux filtrages avant de faire la correspondance finale [Chen <i>et al.</i> 2014].	51
3.14	Le schéma de fonctionnement du modèle RatSLAM. Des cellules de vue sont obtenues à partir senseurs externes (en général une caméra). La proprioception, par intégration de chemin, sert à obtenir une cellule de pose. Ces deux types de cellules servent à construire une carte d'expérience utilisable pour des tâches de navigation [Milford & Wyeth 2010].	52
3.15	La banlieue de Sainte Lucie où a été testée le modèle RatSLAM [Milford & Wyeth 2008].	53
3.16	Schéma du modèle de mémoire visuelle [Azouaoui & Achour 2014]. Le système visuel extrait un descripteur à partir d'une image d'entrée. Le système de mémoire visuelle compare le descripteur à ceux qu'il a en mémoire et apprend un nouveau lieu si il ne parvient pas à obtenir une correspondance [Azouaoui & Achour 2014]	55

3.17	Le robot Beobot et le schéma de son modèle de localisation et de navigation. Il utilise toute une série de capteurs afin de se localiser sur une carte en utilisant une stratégie d'extraction d'amers, une stratégie de description globale d'image et de l'intégration de chemin. Cette localisation est utilisée par plusieurs stratégies de navigation, voir [Christian <i>et al.</i> 2010] pour de plus amples détails.	58
3.18	Les erreurs de positionnements pour plusieurs trajets (LIBR, PROF et ATHL) selon que le modèle soit en mode automatique ou contrôlé par un utilisateur et en fonction de la présence ou non d'une foule sur le trajet [Christian <i>et al.</i> 2010].	59
4.1	L'architecture neuronale du modèle de reconnaissance visuelle de lieux. Il prend en entrée une image ainsi qu'une information d'orientation. Une chaîne de traitement visuel permet d'obtenir une carte de saillance qui est exploitée par un mécanisme attentionnel afin d'extraire des informations d'identification (what) et d'orientation (where). Un réseau neuronal stocke, combine et apprend ces données aboutissant à la construction de cellules de lieu visuelle. Pour plus de clarté l'ensemble des connexions neuronales sont cachées (lignes hachurées)[Espada <i>et al.</i> 2018]. Les connexions en verts représentent des liens algorithmiques (opérations mathématiques, etc), les connexions en noir représentent les liens synaptiques entre neurones dont les poids sont utilisés pour apprendre des informations.	62
4.2	La chaîne de traitement visuel encadrée en rouge, première partie du modèle de cellules de lieu visuelles [Espada <i>et al.</i> 2018].	63
4.3	Un exemple d'extraction d'une vue locale et de sa transformation en log polaire [Espada <i>et al.</i> 2018]. Haut - Sur l'image d'entrée on identifie des POI pour lesquels on va extraire une portion d'image définie par deux cercles concentriques (en bleu). Milieu - Représentation en log/polaire de la portion d'image associée à un point. Bas - Représentation de cette même signature du POI sous forme d'une carte neuronale.	65
4.4	La partie neuronale encadrée en rouge, seconde partie du modèle de cellules de lieu visuelles [Espada <i>et al.</i> 2018].	66
4.5	L'encodage d'un azimut. L'orientation donnée par une boussole magnétique est encodée sous la forme d'une gaussienne centrée sur l'orientation. Cet encodage se fait par une population de neurones simulant ceux de directions de la tête définis par l'équation 4.2 [Espada <i>et al.</i> 2018].	67
4.6	La structure de la mémoire de travail, encadrée en rouge. Elle combine une vue locale sous forme d'un log polaire encodé sur un neurone de signature avec une orientation sous forme d'une gaussienne discrétisée sur une population de neurones [Espada <i>et al.</i> 2018].	68

4.7	Les véhicules de l'institut VEDECOM [Espada <i>et al.</i> 2018].	74
4.8	Exemple d'images acquises avec les véhicules de VEDECOM [Espada <i>et al.</i> 2018].	75
4.9	Exemple de vérité terrain 4.9a et de matrices de confusion 4.9b pour la base KITTI 06. La vérité terrain, construite pour un nombre donné de cellules de lieu, indique à quelle cellule devrait correspondre chaque image. La matrice de confusion donne pour chaque image à quelle cellule de lieu correspond réellement chaque image. Pour un système avec une parfaite reconnaissance des fermetures de boucles avec des champs de lieu de taille régulière, la vérité terrain et la matrice de confusion seraient identiques.	79
4.10	schéma explicatif du calcul du 3WD	80
4.11	Tracé des activités des cellules de lieu, sur la base de données KITTI 06, avec le mécanisme basé sur un pas fixe. Les zones mise en valeurs avec un contour gris correspondent à des erreurs dans l'ordre de reconnaissance des cellules de lieu. La zone mise en valeur avec un contour marron montre une absence de recouvrement par la représentation	81
4.12	Tracé de la trajectoire de la base de données KITTI 06 pour le mécanisme d'apprentissage basé sur le seuil de vigilance. Le champ de lieu associé à chaque cellules de lieu est dessiné avec une couleur différente.	83
4.13	Le score 3WD et le nombre de cellules de lieu recrutées pour l'ensemble des seuils de vigilance pour la base de données KITTI 06.	84
4.14	Le nombre d'erreurs séquentielles et le ratio WNR pour l'ensemble des seuils de vigilance pour la base de données KITTI 06.	85
4.15	La taille des champs de lieu pour l'ensemble des seuils de vigilance pour la base de données KITTI 06.	86
4.16	Base de données KITTI 06 : la précision et le rappel pour les trois vigilances 0.30 0.42 et 0.80	87
5.1	Architecture du modèle PerAc. Le modèle de cellules de lieu visuelles recrute des lieux qui sont ensuite associés à des actions par une seconde partie du système. Une troisième partie s'appuie sur cette représentation lieu/action pour sélectionner l'action à mener et génère la consigne associée.	94
5.2	Apprentissage de huit lieux autour d'une position but permettant de réaliser des tâches de homing. Dans cet exemple, les huit cellules de lieu permettent de couvrir l'ensemble de l'espace grâce à leur champs de lieu. Chaque lieu est associé à une direction pointant vers la position cible. En suivant la direction correspondant au lieu courant, le robot ira soit vers le centre soit entrera dans un autre champ de lieu et changera donc de direction à suivre. Par une succession de changements de directions, le robot convergera vers le centre, réalisant ainsi le homing. On obtient l'équivalent d'un puits dans un champ de potentiel.	95

5.3	Test de l'apprentissage d'une trajectoire cible. La trajectoire cible , en pointillée, est obtenue par un ensemble d'associations lieu/action, point et flèches rouge. La trajectoire de test est dessinée en noir.	96
5.4	1 - Le noyau d'interaction définit à partir d'une différence de gaussienne avec le centre excitateur et la périphérie inhibitrice. 2 - Exemple de connexions latérales pour un neurone lui permettant d'exciter ces voisins proches et d'inhiber ses voisins plus éloignés. [Jauffret 2014]	96
5.5	Exemple de propriétés de coopération et de compétition d'un champ de neurones dynamiques. A - Compétition : deux stimuli sont éloignés et donc s'inhibent mutuellement jusqu'à ce que l'un d'eux disparaisse permettant au stimuli restant de se renforcer par son lien récurrent. B - Coopération : deux stimuli proches s'excitent mutuellement et finissent par converger vers une réponse correspondant à la moyenne des deux stimuli [Jauffret 2014].	97
5.6	Exemple d'attracteur (en haut) et de répulseur (en bas) obtenu par la dérivée d'une gaussienne positive pour l'attracteur et négative pour le répulseur. Lorsque l'on se décale à gauche du point attracteur $\frac{d\phi}{dt}$ est positif donc ϕ augmente et se décale vers la droite sur la courbe. Lorsque l'on se décale à droite de l'attracteur $\frac{d\phi}{dt}$ est négatif donc ϕ diminue et se décale vers la gauche. Il se passe le phénomène inverse pour le répulseur, un décalage à gauche par rapport au point répulseur entraîne un décalage vers la gauche ($\frac{d\phi}{dt}$ négatif), un décalage vers la droite entraîne un décalage vers la droite ($\frac{d\phi}{dt}$ positif) [Schöner <i>et al.</i> 1995].	98
5.7	Le contrôle moteur se basant sur la théorie des champs de neurones dynamiques. A - Les champs codant les buts (1) et les obstacles (2) sont sommés pour former le champ motivationnel (3) dont la dérivée donne la vitesse angulaire pour chaque orientation du robot (4) dont il suffit de lire la valeur analogique pour l'orientation courante du robot (5). B - Vue de haut représentant l'espace péricorporel du robot où chaque neurone encode un angle particulier. Une valeur de neurone positive correspond à un but tandis qu'une valeur négative réfère à un obstacle.	99
5.8	Un robot Robulab du laboratoire ETIS sur lequel est implémenté le modèle PerAc.	100
5.9	Schéma de la communication entre les machines. La partie véhicule centralise les données capteurs (image, orientation du véhicule et du volant) et les envoie via un nœud TCP. Sur le PC embarqué le modèle PerAct construit et reconnaît des lieux et envoie des consignes d'orientation du volant.	101
5.10	Le modèle du système différentiel utilisé lors de précédentes expériences.	102

5.11	Le modèle du tricycle permettant d'obtenir le modèle cinématique du véhicule.	103
5.12	La portion de piste utilisé pour l'expérience.	104
5.13	Le véhicule VEDECOM utilisé pour l'expérience.	105
5.14	La trajectoire des deux tests réalisés.	107
6.1	La chaîne de traitement visuel des contextes. L'image d'entrée est divisée en trois sous-images. Chaque sous-image est décrite par un descripteur GIST. Les trois descripteurs obtenus sont concaténés afin d'obtenir un descripteur qui sert d'entrée à la couche neuronale. . . .	113
6.2	A	115
6.3	B	115
6.4	Exemple de réponse d'un filtre de Gabor. La convolution d'une image (A) par un filtre de Gabor donne une réponse (B) sur laquelle les lignes ressortent. Le filtre utilisé est défini par un filtre de taille 32, un gamma de 1 et un angle θ de 0 degré.	115
6.5	Le modèle de contexte dans sa globalité. Les neurones de contextes viennent exercer une modulation sur les neurones de signature et les cellules de lieu. Chaque neurone de la couche de signature ou de la couche des cellules de lieu est connecté à l'ensemble des neurones de la couche des contextes.	117
6.6	L'activité des contextes pour les 200 premières images des bases de données KITTI 01 (autoroute) KITTI 06 (banlieue) et KITTI 07 (centre-ville).	119
6.7	Les erreurs de reconnaissances pour la base KITTI 06. A - L'indice de la cellule gagnante pour chaque image. La zone où a lieu l'essentiel des erreurs est encadrée. B - L'activité des contextes pour la plage d'images où a lieu le plus d'erreurs. On peut observer des variations dans l'ordre des gagnants. C - L'activité des cellules de lieu pour la plage d'images où a lieu le plus d'erreurs. On peut observer une corrélation entre certaines erreurs des cellules de lieu et des contextes.	121
6.8	La valeur moyenne du pixel pour chaque image convoluée par un filtre de Gabor. Les 1101 premières images correspondent à la base de données KITTI 01 (autoroute), les 1101 suivantes à la base KITTI 06 (banlieue) et les 1101 dernières à la base de données KITTI 07 (centre-ville). Le filtre utilisé est défini par un filtre de taille 16, un gamma de 1 et un angle θ de 0 degré. On peut observer que la valeur moyenne des pixels est plus faible pour la base de données KITTI 01.	126

-
- 7.1 Exemple de cellules encodant plusieurs champs de lieux. De haut en bas, pour trois neurones différents : la fréquence des impulsions sur un trajet de 48 m, les pics d'activations sont indiqués ; positions linéarisées de l'animal (en gris) avec les pics d'activité (en rouge) ; fréquence des impulsions neuronales linéarisée. Les pics d'activations lorsque la vitesse de l'animal est inférieure à $5m/s$ sont omis [Rich *et al.* 2014]. 130
- 7.2 Les différentes combinaisons pour l'encodage d'un lieu discutées. **A** - Une cellule encode un lieu dans le cas où une cellule n'a qu'une réponse unique (gauche) et le cas où elle montre des réponses multiples (droite). **B** - Un lieu est encodé par une réponse conjointe de plusieurs cellules dans le cas où une cellule n'a qu'une réponse unique (gauche) et le cas où elle montre des réponses multiples (droite) [Park *et al.* 2011]. 131

Liste des tableaux

2.1	Comparaison qualitative des technologies employées pour la localisation	16
3.1	Performances du modèle selon la valeur du seuil de vigilance	56
4.1	Les caractéristiques des jeux de données KITTI utilisés	73
4.2	Les caractéristiques des jeux de données constitués au sein de l’institut VEDECOM	76
4.3	Les paramètres du modèle utilisés lors des expériences.	77
4.4	Les métriques pour des valeurs de seuil s_{vigi} extrémales (0.3 et 0.8) et celle considérée optimale (0.42) pour l’ensemble des bases de données KITTI utilisées	82
4.5	Les métriques pour des valeurs de seuil s_{vigi} extrémales (0.3 et 0.8) et celle considérée optimale (0.42) pour les bases de données VEDECOM	82
4.6	le score F1 pour les modèles PerAc et FabMap	86
4.7	Description qualitative des différentes approches de l’état de l’art en fonction de critères concernant le type de capteurs utilisés les structures et encodages utilisés et les types de descripteurs utilisés.	88
4.8	Comparaison quantitative des différentes approches en fonction d’un certain nombres de critères	88
4.9	Les mesures de coût computationnel par des mesures de temps d’expérience, d’utilisation de mémoire vive ou de mémoire de stockage tout en indiquant le nombre de neurones alloués en début d’expérience	89
5.1	Les résultats de mesure de performances sur deux ordinateurs. La première machine (ordinateur 1) intègre un processeur Intel i5-6200U 2 coeurs 4 Threads @2.3 GHz, 2.8 GHz en Turbo avec 8Go de RAM. La seconde machine (ordinateur 2) intègre un processeur Intel i7-6950X 10 coeurs 20 Threads @3 GHz, 3.5 GHz en Turbo avec 128Go de RAM.	101
5.2	Les mesures, présentées au chapitre 4, de coût computationnel par des mesures de temps d’expérience, d’utilisation de mémoire vive ou de mémoire de stockage tout en indiquant le nombre de neurones alloués en début d’expérience.	101
5.3	L’ensemble des paramètres utilisés pour paramétrer la chaîne de traitement visuel.	106
5.4	Les différentes mesures des résultats obtenus pour chacune des expériences.	108
5.5	Les résultats de mesure sur les trajectoires pour chaque expérience. .	108
6.1	Les métriques pour la base de données KITTI 06 pour la modulation de l’activité des cellules de lieu avec binarisation de la réponse des contextes ou utilisation de leurs valeurs analogiques	118

6.2	Les métriques pour la base de données KITTI 06 pour la modulation de l'activité des cellules de lieu selon le nombre de contextes pris en compte.	120
6.3	Les métriques pour les 3 bases KITTI 01, KITTI 06 et KITTI 07 sans modulation et avec modulation de l'activité des cellules de lieu par les contextes générés par le GIST.	120
6.4	Le score F1 pour le modèles de cellules de lieu et le modèle de contexte	122
6.5	Les mesures de coût computationnel par des mesures de temps d'expérience, d'utilisation de mémoire vive et de mémoire de stockage pour le nombre de neurones nécessaires pour chaque expérience. Le processeur utilisé est un Intel Core i7-8550U à 4 cœurs cadencé à 1.80 Ghz	122
6.6	Les métriques pour la base KITTI 06 sans modulation et avec modulation de l'activité des cellules de lieu par une représentation contextuelle plus ou moins précise.	123
6.7	Les métriques pour la base KITTI 06 sans modulation et avec modulation de l'activité des cellules de lieu avec différent seuils pour le recrutement des cellules de lieu	123
6.8	Les métriques pour les bases KITTI 01 et KITTI 07 sans modulation avec un seuil de 0.42 pour le recrutement des cellules de lieu et avec modulation de l'activité des cellules de lieu pour un seuil de recrutement des cellules de lieu fixé à 0.5.	124
6.9	L'écart moyen entre le premier et le deuxième gagnant pour le modèles de cellules de lieu et le modèle de contexte	124

Introduction

La capacité de se mouvoir, la mobilité, sans que l'on s'en rende forcément compte, est un élément structurant majeur de nos vies. Au-delà de sa nécessité basique pour la survie (rejoindre une source de nourriture, fuir un danger) la mobilité influe grandement sur l'organisation de nos vies et de nos sociétés.

On peut voir son influence dans l'organisation de structures étatiques. Un exemple est le premier découpage des départements suite à la Révolution française. Un département définissait un espace où une préfecture était à moins d'une journée de voyage pour un cavalier.

Elle structure surtout fortement l'urbanisme. Les villes ne s'étendaient rarement au-delà d'un rayon de 5 kilomètres lorsque l'on utilisait principalement la marche. L'utilisation plus répandue de destriers (chevaux, mules, calèches) a conduit au doublement de la taille de ces villes. Le vingtième siècle a vu une révolution majeure de la mobilité avec la multiplication des moyens de transport (train, avion) et de leur facilité d'accès. Le plus important a été la démocratisation de l'accès à la voiture. Cette dernière a profondément changé nos sociétés. En plus de faire exploser la taille des villes qui peuvent maintenant s'étaler sur des rayons de plusieurs dizaines de kilomètres elle en a profondément changé l'organisation : les villes sont maintenant très segmentées, avec des zones d'habitation, des zones de bureaux, des zones d'activités commerciales et des zones d'activités industrielles. Le livre "Airvore ou la face obscure des transports" de Laurent Castaignède détaille ces bouleversements, dont certains politiques, apportés par la révolution des transports.

Dans la décennie à venir, la mobilité pourrait connaître un nouveau bouleversement d'ampleur apporté par les véhicules autonomes. Ce dernier est le résultat de l'hybridation de l'automobile et de la robotique. Ce sont des véhicules (voiture, moto, camion) équipés de différents capteurs pour percevoir l'environnement et de contrôleurs robotiques pour donner des ordres au véhicule (vitesse, angle de volant) permettant à ce dernier de se déplacer sans l'aide d'un humain.

Le terme robot provient du terme tchèque *robota* qui peut être traduit par : travail, besogne, corvée. Ce mot a été imaginé par Joseph Capek et introduit en 1920 par son frère Karel Capek auteur de la pièce de théâtre nommée *Rossum's Universal Robots*. Un robot peut prendre des formes diverses mais peut être généralement décrit comme étant un système, souvent munis de capteurs (tactiles, visuels, sonores) afin de percevoir son environnement, de divers systèmes de calculs afin de traiter de l'information et parfois d'actionneurs (moteurs, écrans, enceintes) afin d'agir sur l'environnement. La robotique autonome vise à créer des systèmes dotés

de suffisamment de capacité d'autonomie afin de pouvoir accomplir une ou plusieurs tâches sans intervention d'un humain. La robotique correspond au rêve très ancien de l'humanité de fabriquer une machine accomplissant les tâches ingrates à sa place. On en trouve des traces au moins depuis l'Antiquité avec le mythe de Talos dans la mythologie grecque. Dans certaines versions de cette légende, Talos était un automate forgé par Héphaïstos, le dieu des forgerons. Il était gardien de l'île de Crète et effectuait, chaque jour, trois rondes parcourant l'ensemble de l'île.

Ce rêve commence à se concrétiser depuis la deuxième partie du vingtième siècle grâce à la robotique. Les robots, après avoir envahi les laboratoires de recherche ont progressivement commencé à entrer dans les usines à partir des années 1980 en s'insérant dans les chaînes de production sous forme d'automates se cantonnant à répéter les mêmes tâches, donc sans la moindre autonomie. Depuis les années 2000 ils commencent à s'inviter dans nos vies quotidiennes sous forme d'aspirateurs ou de tondeuses autonomes ou bien d'assistants intelligents (Siri, Alexa, etc). Il est tout naturel qu'ils en viennent, à terme, à s'insérer dans nos transports, et en particulier l'automobile, qui est un pilier central de l'organisation de nos vies et de nos sociétés.

Au moment de l'écriture de ce manuscrit, si le niveau d'autonomie des véhicules s'améliorent sans cesse leur permettant d'effectuer certaines tâches de conduite, ils restent toutefois encore bien inférieurs aux conducteurs humains.

Ces derniers, comme les autres animaux, sont d'excellents systèmes de navigation biologiques qui se sont perfectionnés au travers de millions d'années de survie. L'observation de la nature et des animaux a permis à l'humanité de résoudre bon nombre de problèmes, que ce soit de survivre en adoptant les comportements de chasse des prédateurs vivant en meute, ou de parvenir à fabriquer des machines volantes à partir de l'observation de la morphologie des oiseaux. L'étude des systèmes de navigation des animaux et de leurs éléments communs, l'approche bio-inspirée est susceptible d'apporter des briques de solutions afin de contribuer à résoudre ce grand challenge de la navigation autonome des véhicules. Ce problème, extrêmement vaste, peut être découpé en tout un ensemble de sous problèmes. Un découpage possible assez classique est perception -> planification -> action : pour naviguer je dois être capable de réaliser les actions menant à un déplacement (action) mais pour savoir quelles actions réaliser il me faut savoir vers où aller (planification) mais pour savoir où aller il me faut savoir où je suis (perception). C'est sur cet aspect de la perception, la question de la localisation, qui vise à construire une représentation de l'environnement utilisable pour effectuer des tâches de navigation que se concentre cette thèse.

Ce manuscrit commence par aborder les notions de navigation en robotique en évoquant les technologies et méthodes courantes avant de se concentrer plus particulièrement sur les aspects de perception du monde et de sa représentation.

Il continue par présenter les notions de bio-inspiration et de neuromimétisme. Il propose aussi un socle de connaissances sur les structures biologiques impliquées dans la perception visuelle et la cognition spatiale qui serviront à la bonne compréhension des travaux qui seront présentés par la suite. Les premiers travaux détaillent un modèle de cellules de lieux visuelles, les tests réalisés sur ce modèle à partir de

données issues d'enregistrements acquis depuis un véhicule et discutent les résultats obtenus. Le chapitre suivant présente un modèle de navigation réactive se basant sur ce modèle de cellules de lieux visuelles : le modèle PerAc. Il décrit le principe du modèle et son implémentation initiale avant de détailler les changements apportés afin de l'adapter sur véhicule. Ce chapitre détaille aussi les expériences réalisées sur véhicules et en discute les résultats.

Le chapitre 5 présente une évolution du modèle de cellules de lieux visuelles, le modèle de contexte visuel. Il détaille le modèle avant de présenter et discuter les résultats obtenus par des simulations basées sur des données réelles.

Ce manuscrit se termine sur une discussion des différents modèles proposés et indique un ensemble de pistes, dont certaines, en partie déjà explorées dans d'autres travaux, permettraient d'améliorer les différentes parties de ces modèles.

La navigation en robotique

Sommaire

2.1	La perception	8
2.1.1	La proprioception	8
2.1.2	La géolocalisation	9
2.1.3	La reconnaissance de lieux visuelle	10
2.1.3.1	Les descripteurs locaux	12
2.1.3.2	Les descripteurs globaux	14
2.1.4	Les télémètres	15
2.1.5	Synthèse	16
2.2	La navigation sans carte	17
2.2.1	La navigation vers un but	17
2.2.2	Les systèmes réactifs	18
2.2.3	Les méthodes de suivi	18
2.2.4	Les modèles de subsumption	19
2.2.5	L'algorithme des fourmis	20
2.2.6	L'intégration de chemin	20
2.2.7	Les méthodes de bout-en-bout	21
2.2.8	Applications actuelles	22
2.3	La navigation avec carte	22
2.3.1	Les cartes métriques	23
2.3.2	Les cartes topologiques	24
2.3.3	Les cartes hybrides	25
2.3.4	Les cartes cognitives	26
2.4	Le SLAM	26
2.5	Discussion	27

La navigation est la capacité à se mouvoir dans un environnement donné. Elle peut être utilisée pour explorer l'environnement afin d'améliorer la capacité à y naviguer, mais sert, plus généralement, à se rendre à un endroit particulier de cet environnement, pour satisfaire un objectif donné, trouver de la nourriture dans le cas des animaux par exemple. En robotique, la navigation représente un champ de recherche majeur, l'un des premiers étudié dès le début des années 1900 [Miessner 1912]. Un robot capable de naviguer peut accomplir un certain nombre de tâches (livraison, inspection, surveillance, transport de passager) et acquérir une certaine autonomie,

en retournant vers une station de recharge par exemple. Les travaux en navigation robotique ont permis de résoudre partiellement ce problème en environnement intérieur.

Par contre, en environnement extérieur, la navigation reste un défi majeur à résoudre par les systèmes robotiques. Ce challenge est encore plus ardu quand il concerne la navigation de véhicules autonomes.

Les difficultés à naviguer dans un environnement extérieur proviennent du fait que ce type d'environnement est très différent d'un environnement intérieur. Un environnement intérieur est structuré, constitué principalement de murs, généralement droit, et de portes définissant des pièces. Un environnement extérieur peut être structuré en zone urbaine avec des bâtiments, des rues et des croisements, mais peut aussi être bien plus chaotique en zone rurale ou pire en pleine nature. En extérieurs, les environnements sont aussi moins contrôlés qu'en intérieurs. Ils subissent des variations de luminosité qui peuvent être plus importantes, des variations des conditions météorologiques qui peuvent radicalement altérer l'apparence d'un environnement (pluie, neige) et ont tendance à subir plus de changements du fait de la plus grande présence d'agents mobiles (voitures, humains, animaux). De plus, les environnements extérieurs sont généralement bien plus vastes que les milieux intérieurs.

Les véhicules autonomes circulent dans des environnements extérieurs donc doivent faire face à ces problèmes avec quelques difficultés supplémentaires. Ils naviguent généralement en utilisant le réseau routier, qui est structuré et régit par un ensemble de règles qui comporte son lot de contraintes. Un véhicule doit respecter le code de la route et est donc restreint dans ces déplacements (rester dans sa voie, avancer, prendre un rond-point dans le bon sens ...). Il doit aussi être capable d'extraire et d'interpréter un certain nombre d'informations supplémentaires de l'environnement (panneaux de signalisation, feux de croisements ...) pour respecter les règles de circulation. Pour finir, il doit naviguer tout en gérant les interactions avec d'autres agents (véhicules, cyclistes, piétons) afin de garantir la sécurité de chacun.

La question de la navigation en robotique est un problème à tiroir pouvant être découpé en toute une série de questions subsidiaires.

Il existe deux grandes familles de modèles de navigations, ceux qui reposent sur une carte du monde et ceux qui n'utilisent pas de carte. La formulation du problème de la navigation varie donc selon la solution envisagée. Pour un modèle sans carte le problème peut se scinder en trois questions :

- Comment percevoir l'environnement ?
- Comment associer perceptions et actions ?
- Comment réaliser l'action ?

Pour un modèle reposant sur une carte du monde le problème peut s'énoncer de la façon suivante :

- Comment percevoir l'environnement ?
- Comment positionner la perception dans la carte du monde ?
- Comment planifier l'action ?
- Comment réaliser l'action ?

La question de la perception de l'environnement, commune aux deux types de modèles, renvoie à la question du choix de l'information à extraire permettant de discriminer différentes zones de l'environnement. Une différence vient de la possibilité pour les modèles reposant sur des cartes d'utiliser ces dernières pour guider la recherche d'information. Cette question concerne aussi le choix de la technologie à utiliser pour réaliser cette extraction d'information. La perception peut être employée pour diverses tâches (détecter les obstacles, identifier les règles de circulation à suivre) mais ce manuscrit se limitera essentiellement à son utilisation pour la localisation dans le but de naviguer.

Le problème de l'association des perceptions aux actions des modèles sans carte englobe deux sous-questions. La première concerne le choix de la technique pour associer une action spécifique à une perception particulière. La deuxième est de concevoir des méthodes permettant de construire un répertoire d'associations suffisamment exhaustif pour que le modèle puisse gérer les différentes situations auxquelles il sera confronté.

La question du positionnement dans la carte est centrale dans les modèles basés sur des cartes. Cette opération peut être coûteuse en temps selon la représentation utilisée et la taille de cette carte. Elle est nécessaire pour pouvoir exploiter au mieux la carte.

La planification désigne l'élaboration du trajet à suivre en s'appuyant sur la représentation. Elle se fait, souvent par des algorithmes de recherches du plus court chemin, comme A^* [Hart *et al.* 1968] ou Dijkstra [Dijkstra 1959] par exemple, ou par des méthodes d'interpolation de courbe entre différents points de passages [Gonzalez Bautista *et al.* 2015]. Elle peut aussi désigner le découpage de la séquence d'action pour reproduire dans le monde réel la trajectoire envisagée sur la carte.

Le problème de la réalisation de l'action, elle aussi commune aux deux familles de modèles, concerne le découpage de l'action en tâches élémentaires et aux questions de contrôle moteur des actionneurs.

L'élaboration d'une représentation spatiale est une étape préalable à la planification et la réalisation des actions. Ce manuscrit se focalise sur cette première tâche de localisation en proposant un modèle neuronal de reconnaissance visuelle de lieux. Les questions qui ont trait à la planification et la réalisation des actions n'y seront donc que peu développées.

Ce chapitre porte donc sur les deux premières étapes du processus de navigation identifiées plus haut : la perception et la représentation de l'espace en vue de s'y localiser. Il commence par décrire des technologies et des méthodes généralement utilisées pour percevoir l'environnement. Il se poursuit par une présentation des différentes stratégies de navigations regroupées selon qu'elles reposent sur l'utilisation de cartes ou qu'elles s'en passent. Le chapitre terminera par décrire sommairement différents algorithmes, fréquemment employés pour la navigation des véhicules autonomes, qui vise à résoudre à la fois le problème de la cartographie de l'environnement et celui du positionnement dans cette carte : le SLAM (Simultaneous Localisation And Mapping).

2.1 La perception

La capacité à percevoir l'environnement et son propre état est primordiale pour pouvoir effectuer des tâches de navigation, qu'elles reposent sur l'utilisation de carte ou non. Elle est d'autant plus nécessaire pour les architectures se basant sur une carte, qu'elle soit fournie ou construite, car ces dernières doivent passer par une étape de mise en correspondance entre leur perception de l'environnement et leur représentation de cet environnement.

Pour percevoir son environnement il existe une myriade de technologies de capteurs qui extraient des données de types différents. La forme de représentations que l'on pourra construire (carte métrique, topologique, ...) et les méthodes de génération de cette représentation sont fortement influencées par le type d'information que le système sera capable d'extraire (distance, image,...). Dans le cadre de la navigation des véhicules autonomes, la perception se fait sur des environnements extérieurs qui peuvent être soumis à tout une série de perturbations comme les variations de luminosité ou les changements météorologiques qui peuvent dégrader la qualité de l'acquisition d'information suivant les technologies utilisées. Par exemple les variations de luminosité influenceront sur la prise d'image par une caméra mais n'auront pas d'incidence sur un radar.

Pour la conduite déléguée, quatre types d'informations sont généralement utilisés, souvent de concert : la proprioception, le positionnement géo-référencé, la vision et la télémétrie.

2.1.1 La proprioception

La proprioception désigne un ensemble de méthodes et de technologies permettant de mesurer les états propres à un agent. Par conséquent, elle ne récupère pas directement des informations sur l'environnement mais indirectement sur la manière dont un agent se meut dans cet environnement.

Les informations de proprioception, pour la robotique mobile, concernent essentiellement les déplacements et permettent d'obtenir, directement (accéléromètres,gyroscopes) ou indirectement (roues codeuses), des informations sur les mouvements de translations et de rotations. Pour les véhicules automobiles, une autre information importante est l'angle de braquage actuel des roues.

Il existe plusieurs méthodes et technologies pour acquérir ces mesures qui ont leurs avantages et inconvénients. Certaines se concentrent sur une seule information alors que d'autres permettent d'acquérir des informations liées aux translations et aux rotations.

Pour la mesure des translations il existe principalement deux technologies.

Une première méthode, pour les véhicules munis de roues, est l'utilisation de roues codeuses. Ce sont des capteurs qui mesurent la rotation des roues auxquelles ils sont liés et, connaissant la dimension des roues, permet d'en déduire la translation effectuée. L'inconvénient de cette méthode est que lorsque la roue subit un glissement (terrain boueux, aquaplaning, ...) et donc ne tourne pas, le capteur est incapable

de mesurer le mouvement.

Une seconde méthode se base sur les effets d'inertie par l'intermédiaire d'accéléromètres. Par une double intégration du signal obtenu, on peut estimer les translations. Cette méthode a l'avantage de ne pas être sensible aux mêmes limites que rencontrent les roues codeuses. Le défaut de cette technologie est la nécessité d'une phase de calibration du capteur pour limiter l'accumulation des erreurs liées à la double intégration d'informations bruitées, avec des réponses souvent non linéaires et des biais.

Du côté de la mesure des mouvements de rotation, il existe majoritairement deux solutions. La première est l'utilisation du champ magnétique terrestre par l'intermédiaire d'une boussole magnétique. Cette méthode a l'avantage de reposer sur une référence généralement fiable (sauf près des pôles), le nord magnétique, disponible n'importe où sur Terre. Le défaut de cette méthode vient du fait que le champ magnétique peut être perturbé localement par des champs magnétiques autres, venant de l'électronique par exemple, ou bien par des structures métalliques (béton armé, ponts).

La seconde technologie repose sur des effets d'inertie : le gyroscope. Ces capteurs permettent de mesurer les accélérations angulaires. Par un calcul de double intégration, on calcule les changements d'orientation. Cette technologie a les mêmes avantages et inconvénients que les accéléromètres, la nécessité d'une calibration et une dérive de la mesure au cours du temps.

Certaines technologies permettent de mesurer à la fois les translations et les rotations. C'est le cas notamment des centrales inertielles. Ce sont des systèmes qui sont généralement constitués à la fois d'accéléromètres et de gyroscopes. De ce fait, ils ont les mêmes avantages et inconvénients que ces capteurs pris séparément.

Les méthodes d'estimation des mouvements proprioceptifs donnent des mesures avec des erreurs qui ont le défaut d'être cumulatives. Les erreurs sur l'orientation sont plus délétères que celles sur les translations. Les premières provoquent un effet cumulatif sur les dernières. C'est un effet qui est particulièrement visible sur les algorithmes de SLAM, menant à des cartes de l'environnement de plus en plus distordues du fait de l'accumulation d'erreurs dans le positionnement.

Pour limiter les erreurs issues de la proprioception, cette dernière est généralement utilisée en conjonction d'autres méthodes de localisation pour faire une étape de recalage qui va permettre de retrouver une localisation fiable permettant de corriger les effets de dérive.

La proprioception est généralement utilisée pour fournir l'information de déplacement propre à l'agent. Elle est aussi utilisée pour la construction des cartes cognitives, métriques ou topologiques pour déterminer la transition à parcourir pour passer d'un lieu à un autre par exemple.

2.1.2 La géolocalisation

La géolocalisation consiste à donner une position selon le repère géocentrique. Cette position est obtenue par la triangulation des distances entre le véhicule que l'on

veut localiser et un certain nombre (au minimum quatre) de points de référence. Ces points de référence peuvent être soit, dans la grande majorité des cas, des satellites en orbite terrestre (GPS, Galiléo, GLONASS, Beidou, ...), soit des infrastructures terrestres géo-référencées ou bien une combinaison des deux.

Cette méthode de localisation est assez précise pour les puces GNSS (Géolocalisation et Navigation par un Système de Satellites) les plus performantes en contrepartie d'un surcoût. Alors que les puces dans les smartphones et assistants GPS n'ont qu'une précision de l'ordre de quelques mètres et reposent sur les mesures effectuées sur les réseaux cellulaires pour améliorer la précision, les puces les plus précises donnent une position avec une erreur de positionnement de l'ordre de quelques dizaines de centimètres et au pire du mètre. Cependant, il existe toute une série de cas où la géolocalisation est insuffisamment performante ou ne fonctionne pas tout simplement. L'information d'altitude n'est pas suffisamment précise. Pour la conduite de véhicule, cette information d'altitude n'est pas utile mais peut provoquer des erreurs de géolocalisation au niveau des ponts ou des routes aériennes nécessitant des méthodes d'intégration temporelle pour les corriger. Par ailleurs cette information de position ne peut être obtenue que si le véhicule est capable de communiquer avec au moins quatre références. Ainsi, sous terre (tunnels, parkings souterrains) ou dans des zones entourées de hautes structures (ruelles, cités résidentielles) le signal n'est pas toujours disponible et la géolocalisation est donc impossible. Ces structures environnantes peuvent aussi provoquer des phénomènes de multi-trajet altérant le signal provenant des satellites et donc perturbant la localisation. Le terme multi-trajet désigne un phénomène où un signal, par des effets de réflexion, de réfraction ou de diffraction, suit plusieurs chemins avant d'être reçu plusieurs fois.

Il existe une version plus précise de géolocalisation, le GNSS-RTK (Real Time Kinematic), qui utilise la phase de l'onde porteuse du signal ainsi qu'un système différentiel reposant à la fois sur des satellites et des infrastructures terrestres géo-référencées pour obtenir une plus grande précision, de l'ordre de quelques centimètres. Son usage reste cependant limité à cause du recours à des infrastructures terrestres dédiées, pour le moment, encore insuffisamment déployées à l'échelle mondiale.

2.1.3 La reconnaissance de lieux visuelle

La vision est une modalité qui, passivement, capte et enregistre la lumière issue de l'environnement. Elle permet d'obtenir une riche quantité d'informations sur l'environnement :

- Par l'analyse des couleurs, on peut discriminer des régions.
- Par des algorithmes de détection de contours, on peut obtenir des informations sur la structure de l'environnement (routes, bâtiments).
- Par des analyses de la variation des valeurs des pixels on peut extraire des textures.
- Par des algorithmes de flux optique, on peut obtenir des informations de

déplacement.

- Par des méthodes de stéréoscopie, on peut calculer des informations sur la profondeur et faire du positionnement en 3 dimensions.

Les caméras qui captent l'information visuelle sont des composants passifs, donc consommant peu d'énergie, et en général peu coûteux. Il existe des caméras plus onéreuses, soit parce qu'elles ont une très grande fréquence de prise de vues, une adaptation rapide aux changements de luminosité ou bien parce qu'elles font de la détection d'évènements temporels (caméras événementielles). Ces dernières, qui captent des informations similaires à celles des yeux de certains mammifères, ne sont, pour l'instant, que très peu utilisées dans l'automatisation des véhicules du fait, en plus de leur coût, de leur faible résolution. Une limite de la plupart des caméras est la sensibilité aux changements brutaux de luminosité (sortie de tunnel), nécessitant un contrôle rapide du gain, qui peuvent provoquer un phénomène de saturation des capteurs. De plus, comme ce sont des capteurs passifs, ils sont très dépendants de l'environnement, en particulier de la luminosité qui, si elle est trop faible ou altérée par les conditions météorologiques peut dégrader la qualité des perceptions. Certains usages, notamment en stéréoscopie, nécessitent de réaliser une étape de calibration afin de connaître précisément les paramètres intrinsèques de la caméra utilisée.

Les travaux bio-inspirés présentés dans la suite de ce manuscrit se basent sur la vision, qui, en plus de ces avantages évoqués plus tôt, est une modalité largement utilisée dans le règne animal. Par conséquent sa description sera plus développée que celle des autres modalités de perception.

La plupart des systèmes utilisant la vision reposent sur de la vision passive avec une ou plusieurs caméras qui sont fixes sur le véhicule ou le robot. Il existe aussi des architectures, moins fréquentes, qui utilisent de la vision dite active. Dans ces systèmes, les caméras sont montées sur un mécanisme leur fournissant des degrés de liberté supplémentaires afin de pouvoir s'orienter et ainsi acquérir à un instant donné des informations sur une zone particulière de l'environnement selon les besoins du modèle de perception. [Chen *et al.* 2011] fournit un état de l'art de l'utilisation de la vision active.

Dans le cadre de la conduite autonome, les caméras les plus couramment utilisées sont des fish-eyes, des caméras avec des grands angles d'ouverture pour percevoir l'environnement mais entraînant des déformations des images pouvant nécessiter une phase de correction de ces distorsions. Il existe des configurations permettant de balayer les 360 degrés de l'environnement, soit par l'utilisation d'une caméra omnidirectionnelle fixée sur le véhicule [Gandhi & Trivedi 2005], soit par l'utilisation d'une multitude de caméras synchronisées afin de reconstruire une image globale. [Jauffret *et al.* 2013] et [Belkaid *et al.* 2016] sont des exemples de modèles utilisant une caméra sur un axe motorisé permettant de faire des panoramas. Une autre configuration courante est d'avoir des caméras orientées vers l'avant, en général, pour des applications de détection d'obstacles ou de maintien dans une voie par la détection de la direction de la route ou de marquages.

La vision est principalement utilisée pour accomplir trois tâches. La première est la détection d'objets dans l'environnement. Ces objets peuvent être,

par exemple, des panneaux de signalisation, des marquages, d'autres véhicules ou bien des piétons. Pour ces derniers, des caméras thermiques sont parfois employées [Fernández-Caballero *et al.* 2010] afin de les détecter. La détection et la classification d'objet constitue un vaste champ de recherche qui s'éloigne du sujet de ce manuscrit et ne sera donc pas plus développé.

Une seconde utilisation de la vision est l'odométrie visuelle. Elle consiste à utiliser le flux optique afin d'estimer les déplacements, aussi bien en termes de translations que de rotations, de la caméra. Cette méthode, peut voir ses mesures perdre en précision du fait des perturbations causées par des objets mobiles dans l'environnement qui sont potentiellement nombreux dans un environnement routier. L'avantage de ces algorithmes est qu'ils se basent sur des capteurs servant aussi à caractériser des environnements.

Il existe aussi des capteurs optiques dédiés à l'estimation des translations, similaires à ceux que l'on retrouve sur les souris d'ordinateurs. Ces capteurs doivent être placés en direction du sol, sous le véhicule en général, ce qui les rend vulnérables en environnement extérieur, à une obstruction du capteur par de la saleté.

La troisième utilisation, qui va particulièrement nous intéresser dans la suite de ce manuscrit, est la reconnaissance de lieux visuelle. Elle est obtenue à partir de méthodes qui analysent la scène visuelle en vue de discriminer et de discrétiser l'environnement et permettent ainsi d'obtenir une représentation permettant de se localiser.

Pour ces algorithmes de reconnaissance de lieux, l'encodage des informations de manière compacte afin de fonctionner sur de grands environnements et l'identification des invariants permettant de décrire de manière robuste un environnement afin de n'extraire que les informations pertinentes constituent des questions activement étudiées.

L'encodage des images peut se faire selon deux stratégies, la description globale d'une image ou bien la description de parties de l'image : une description locale. Ces deux stratégies peuvent aussi être utilisées de manière conjointe.

2.1.3.1 Les descripteurs locaux

Les descripteurs locaux fonctionnent en deux étapes. La première consiste à identifier des points ou des zones d'intérêts desquels seront extraits les informations visuelles. Lors de la deuxième phase les informations visuelles au niveau des zones d'intérêts sont encodées sous forme de descripteurs.

Il existe une quantité de méthodes pour détecter les zones d'intérêts au sein d'une image. Parmi les modèles les plus performants, on peut citer SIFT, le détecteur de Harris, SURF ou FAST. Scale-Invariant Feature Transform (SIFT) [Lowe 2004a] utilise une différence de gaussienne afin de détecter les points d'intérêts qu'elle applique à l'image à des résolutions différentes. Speed Up Robust Feature (SURF) [Bay *et al.* 2006], quant à lui, est une amélioration en termes de coût computationnel de SIFT. Enfin, Features From Accelerated Segment Test (FAST) [Rosten & Drummond 2006] fait de la détection d'angle pouvant servir ensuite de

points d'intérêts. Cette dernière méthode est parmi les plus efficaces [Ali *et al.* 2016] en termes de compromis entre le temps de calcul et les performances.

Après cette phase de détection vient la phase d'extraction de l'information. En plus de la détection de zones d'intérêt, SIFT et SURF proposent des descripteurs qui ont parfois été utilisés en conjonction avec FAST pour la détection de points d'intérêts [Churchill & Newman 2013][Mei *et al.* 2009]. Ces descripteurs sont construits pour chaque zone d'intérêt en découpant la zone en sous-régions pour chacune desquelles est calculé l'histogramme encodant les amplitudes et orientations issues du gradient obtenu lors de la phase de détection.

On peut aussi citer le descripteur local Binary Robust Independent Elementary feature (BRISF) [Strecha *et al.* 2012] lui aussi utilisé conjointement avec FAST. Ce descripteur, pour une portion d'image donnée, se construit en trois étapes :

- Un lissage des valeurs des pixels est appliqué.
- Un ensemble de couples de pixels est sélectionné selon une méthode pseudo-aléatoire. Le premier point est sélectionné sur une distribution gaussienne centrée sur le pixel central de la portion d'image. Le second point est sélectionné sur une distribution gaussienne centrée sur le premier point sélectionné. Le nombre de couples testés dépend de la taille du descripteur.
- On attribue une valeur binaire pour chaque couple selon le résultat de la comparaison des valeurs des pixels.

Il existe aussi Oriented FAST and Rotated BRISF (ORB) [Rublee *et al.* 2011] qui combine une version de BRISF invariant aux rotations, rBRISF, ainsi que FAST. Cet algorithme de vision est fréquemment utilisé pour la localisation, comme dans [Mur-Artal *et al.* 2015] par exemple.

Comme les descripteurs locaux propres à une image peuvent être assez nombreux, entraînant un coût computationnel rapidement ingérable, un algorithme d'agrégation est souvent utilisé afin de représenter ces informations de manière compacte. Les principales méthodes sont VLAD [Arandjelović *et al.* 2016], les sacs de mots visuels [Sivic & Zisserman 2003] et le noyau de Fisher [Jaakkola & Haussler 1999].

La méthode de sac de mots visuels s'inspire des méthodes de construction de vocabulaire utilisées pour le déchiffrement de texte. A partir d'une base d'images de tests sont extraits un ensemble de descripteurs. Un classifieur, en général basé sur un k-mean, va segmenter ces descripteurs afin d'obtenir un certain nombre de classes. Ces classes forment le vocabulaire. Cette méthode est utilisée dans divers modèles de SLAM visuel [Jiachen *et al.* 2018].

Le noyau de Fisher combine les avantages des deux grandes familles de classifieurs, les méthodes génératives (traiter des données de tailles différentes) et les méthodes discriminantes (critères de classification plus flexibles). Les solutions discriminantes segmentent les données que l'on leur fournit afin de les classer tandis que les algorithmes génératifs cherchent à modéliser le processus permettant de générer ces données afin de pouvoir les classer. Le noyau de Fisher est, initialement, une fonction qui, à partir d'un modèle statistique et d'un ensemble de mesures pour chaque objet, calcule la similarité entre deux objets. Appliquée à la classification, une nouvelle classe peut être formée, en minimisant, entre les classes déjà existantes,

la moyenne de la distance du noyau de Fisher d'un nouvel objet à tous ceux déjà connus.

Nous clôturerons cette description des algorithmes de traitement local des images en citant la méthode VLAD (Vector Of Locally Aggregated Data) [Arandjelović *et al.* 2016]. Cette méthode commence par encoder les informations sous un descripteur SIFT de 128 valeurs. Chaque descripteur est assigné à la classe dont il est le plus proche, le modèle possédant un nombre de classes assez sommaire (de l'ordre de 64 à 256 éléments). Pour chaque classe, la distance de chaque élément au centre est calculée et agrégée. Pour finir, une norme L2 est utilisée pour obtenir un vecteur normalisé. [Leng *et al.* 2019] propose, en plus d'une présentation de VLAD, un état de l'art récent des méthodes de description locales.

2.1.3.2 Les descripteurs globaux

Les premières méthodes de description globale se basent sur l'utilisation d'histogrammes de couleurs ou sur l'analyse en composantes principales [Schölkopf *et al.* 1997]. Elles utilisent souvent des caméras omnidirectionnelles pour être plus robustes aux petites variations.

Les descripteurs globaux peuvent être aussi construits selon les mêmes méthodes initialement développées pour les descripteurs locaux. Pour ce faire il suffit de remplacer la phase de détection de zones d'intérêt en prédéfinissant des zones d'intérêt dans l'image en la découpant, avec un motif de grille par exemple. Il existe plusieurs exemples de méthodes de ce type. On peut citer WI-SURF [Badino *et al.* 2012] et BRIEF-GIST [Sünderhauf & Protzel 2011] pour les plus connues.

GIST [Oliva & Torralba 2001] est un descripteur global souvent utilisé pour faire de la reconnaissance de scène. Il se base sur l'utilisation d'un ensemble de filtres de Gabor ayant différentes orientations et fréquences spatiales afin de construire une description compacte de l'image. Les traitements appliqués par ce filtre se rapprochent, en une version simplifiée, de ceux observés dans le cortex visuel chez des mammifères dont l'être humain.

Pour terminer, l'utilisation de réseaux de neurones profonds, qui sera plus détaillée dans le chapitre suivant, peut être considérée comme une méthode de descripteurs globaux puisqu'un réseau construit son descripteur sur ses premières couches à partir de l'ensemble de l'image d'entrée. Une brève revue des méthodes globales pour la reconnaissance d'image peut se trouver dans [Wang & Zhang 2011]

Il existe enfin des méthodes combinant à la fois des descripteurs locaux et des descripteurs globaux [Belkaid *et al.* 2016] [Garcia-Fidalgo & Ortiz 2015]. Cette combinaison a l'avantage de minimiser les points faibles de chaque type de descripteurs. Les méthodes locales avec la phase de détection et le nombre de descripteurs peuvent entraîner un coût computationnel et combinatoire limitant la vitesse de traitement des images. De plus elles sont plus sensibles aux changements de condition météorologique et de luminosité. Les méthodes globales étant quant à elles plus robustes à ces variations mais moins robustes aux variations de position (translation et orientations) pour reconnaître un lieu appris.

2.1.4 Les télémètres

Les télémètres sont des capteurs de mesure de distance. Ils envoient une onde (laser, ultrason) qui va être réfléchiée par les obstacles rencontrés et retournée vers le capteur qui va pouvoir déterminer la distance du capteur à l'obstacle. Certains, de technologies plus complexes, projettent un motif et extraient des informations par l'analyse de la déformation de ce motif. Pour les tâches de localisation, les télémètres les plus utilisés sont les lidars et, dans une moindre mesure, les radars [Van Brummelen *et al.* 2018].

Les systèmes les plus simples ne permettent d'obtenir que des informations sur une seule nappe en deux dimensions d'une portion de l'environnement définie par l'angle d'ouverture du capteur. La plupart des lidars utilisés ont un angle d'ouverture d'une centaine de degrés mais il existe des modèles capables de couvrir les 360 degrés de l'environnement. Des versions plus complexes, en plus de couvrir les 360 degrés de l'environnement, scannent selon plusieurs nappes, permettant d'obtenir une description en plusieurs tranches (voir figure 2.1).

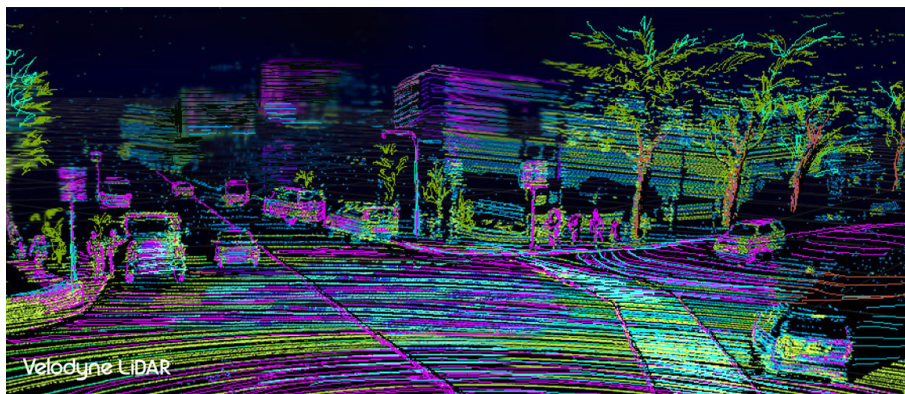


FIGURE 2.1 – Un exemple de représentation obtenue par un lidar multi-couches Velodyne.

Les informations obtenues sont des nuages de points. Ces nuages de points peuvent permettre de construire des représentations de l'espace sous forme de grilles d'occupation ou bien de faire du calcul d'odométrie 3D. Avec un nombre de points suffisamment important, il est aussi possible d'exploiter ces nuages de points par des algorithmes de traitement, que ce soit pour de l'odométrie ou de la reconnaissance de lieux.

Les télémètres sont des capteurs actifs, ce qui a l'avantage, contrairement à leurs cousins passifs (caméra), de les rendre, selon les fréquences utilisées par les capteurs, insensibles aux variations de luminosité. En contrepartie, comme ce sont des capteurs actifs, ils ont une consommation énergétique plus importante et sont susceptibles d'être des générateurs de perturbations pour d'autres capteurs ou pour eux-même (phénomène de diaphonie).

Dans le cas des radars, il existe des types d'environnements, comme les tunnels par exemple, où apparaissent des phénomènes d'écho qui entraînent des fausses dé-

Technologie	GPS	odométrie	télémétrie	vision
coût	plutôt faible	variable	élevé	généralement faible
richesse d'information	faible	plutôt faible	riche	très riche
besoins matériels	très élevés	faibles	moyens	moyens
traitements	faibles	plutôt faibles	très élevés	très élevés

TABLE 2.1 – Comparaison qualitative des technologies employées pour la localisation

tections rendant ces capteurs peu fiables pour ces environnements. Pour les lidars, il existe des conditions météorologiques (brouillard) qui provoquent des effets similaires. Par ailleurs, les lidars actuellement embarqués dans les véhicules intègrent de la mécanique de haute précision qui va être sensible au vieillissement et aux chocs. Pour finir ce sont des capteurs coûteux et relativement volumineux et se pose la question de leur incorporation sur véhicules, un lidar couvrant les 360 degrés de l'environnement placé au-dessus du véhicule ou une série de lidar, moins volumineux, synchronisés aux angles de couvertures plus réduits incorporés sur tout le tour de la carrosserie pour couvrir l'ensemble de l'environnement.

2.1.5 Synthèse

Les différentes technologies présentées qui peuvent être employées pour la localisation ont chacune des points forts et des lacunes. Ces dernières sont regroupées dans le tableau 2.1 qui indique, qualitativement, pour chaque technologie le coût, la richesse des informations obtenues ainsi que les besoins en termes matériels et de traitement pour leur exploitation.

Le GPS donne peu d'informations, juste une position géo-référencée et une orientation, mais qui fournit directement une position. Son coût est relativement faible et peu de traitements sont nécessaires pour obtenir l'information. En revanche, ils ont besoin de nombreuses infrastructures (satellites, stations de base, antennes cellulaires) pour fonctionner.

Les systèmes d'odométrie donnent juste des informations sur les déplacements. Par des calculs simples nécessitant peu de matériel, ils permettent d'obtenir une position relative. Le défaut de cette technologie est le phénomène d'accumulation d'erreurs faussant le positionnement. Il peut être limité en utilisant des capteurs de grande précision mais pour un coût qui devient élevé ou bien en le combinant avec des capteurs exteroceptifs.

Les télémètres permettent d'obtenir des informations de profondeurs et, pour les modèles capables de fournir suffisamment de points, l'équivalent d'une image très basse résolution. Leur utilisation nécessite du matériel de calcul afin de traiter les informations captées. Ces traitements sont assez importants, surtout dans le cadre de la localisation où les informations extraites doivent servir à construire une représentation du monde. De plus les télémètres sont des appareils avec un coût plutôt élevé.

La vision est une modalité qui permet d'obtenir une grande richesse d'informa-

tions (couleurs, formes, contrastes ...) pour un coût généralement modeste. Il existe des variantes, les caméras événementielles, qui apportent en plus une information temporelle mais pour un coût se rapprochant des télémètres. Cette technologie requiert de lourds traitements, afin de pouvoir profiter de la richesse des informations extraites. Par ailleurs, cette modalité est très largement utilisée dans le monde animal. Pour toutes ces raisons, elle est la modalité qui a été retenue pour les modèles bio-inspirés présentés dans la suite de ce manuscrit.

2.2 La navigation sans carte

Pour naviguer, il existe tout un ensemble de stratégies, que l'on observe chez différentes espèces d'animaux, dont on suppose qu'elles ne reposent pas sur une représentation du monde sous forme de carte. Dans cette section, nous allons présenter les principales méthodes qui rentrent dans cette catégorie en commençant par présenter la navigation vers un but. Nous décrirons ensuite comment des mécanismes simples inspirés des insectes peuvent permettre l'émergence de comportements de navigation, par exemple, des stratégies de suivi. L'organisation de ces différentes stratégies réactives simples a mené à l'élaboration de modèles dit de subsumption. Nous verrons ensuite d'autres stratégies directement inspirées du comportement des insectes : l'algorithme des fourmis et l'intégration de chemin. Nous aborderons ensuite les méthodes dites d'apprentissage de bout-en-bout avant de terminer par un tour d'horizon des applications actuelles sur véhicules.

2.2.1 La navigation vers un but

La navigation vers un but, ou homing, désigne la capacité à revenir à un lieu familier depuis des zones inconnues. C'est une capacité que l'on retrouve chez de nombreux oiseaux migrateurs [Walcott 1996], des tortues de mer [Avens & Lohmann 2004] et chez des insectes, notamment les abeilles [Wolf 1930] [Wilson 1968]. Les études de ces différents animaux suggèrent qu'ils se basent sur des amers visuels proches pour reconnaître le but lorsqu'ils en sont à proximité [Cartwright & Collett 1982]. Un amer visuel est un point de l'environnement particulièrement remarquable, comme le clocher d'une église dans un petit village ou bien un arbre dans une prairie. Avec un certain nombre d'amers visuels, il est possible de qualifier un lieu relativement précis de l'environnement. [Bernardino & Santos-Victor 1998], inspiré par l'abeille, est un exemple de système de homing basé sur le calcul de disparité entre l'image courante et l'image but afin d'inférer la direction à suivre pour rejoindre le but. Lorsqu'ils sont plus éloignés, les animaux peuvent se référer à des mécanismes d'orientations variés. Ces mécanismes peuvent être basés sur la détection d'amers distaux, des amers éloignés mais dont on connaît une relation, une distance par exemple, avec le but. L'utilisation de la voûte céleste [Dacke *et al.* 2013] ou du soleil [Waiker *et al.* 2019] en sont des exemples. Ces systèmes d'orientation peuvent aussi reposer sur une capacité à percevoir le champ magnétique [Wiltschko & Wiltschko 2012] [Rodgers & Hore 2009]

ou bien sur le sens de l'olfaction [Stabell 1984]. Ce comportement de homing peut être très efficace comme semble l'attester les abeilles qui peuvent parcourir des kilomètres lorsqu'elles butinent avant de revenir à leur ruche [Pahl *et al.* 2011] ou bien des espèces d'oiseaux migrateurs qui sont capables de parcourir des trajets de plusieurs milliers de kilomètres lors de leur migration et qui semblent se baser sur ce comportement.

2.2.2 Les systèmes réactifs

Avec des systèmes réactifs simples on peut réussir à obtenir une variété de comportements de navigation très différents. Une architecture réactive lie directement une perception à une action sans passer par une phase de planification. Les réflexes (fermer les yeux lorsque l'on éternue, retirer sa main d'une source chaude) sont des exemples biologiques de systèmes réactifs. Les véhicules de Braitenberg [Braitenberg 1986], inspirés par l'étude des insectes, sont un exemple robotique de ces systèmes. Ce sont des robots mobiles fonctionnant avec un système différentiel et munis de capteurs photosensibles fixés sur la gauche et la droite du véhicule. La valeur issue d'un des capteurs sert directement de consigne de vitesse à un des moteurs. Suivant la façon dont on connecte un capteur à un moteur (capteur gauche à moteur gauche ou capteur gauche à moteur droit), et la façon dont la consigne est utilisée (excitation ou inhibition) on obtient des comportements de navigation différents.

On peut obtenir un évitement d'obstacle (voir figure 2.2) en mettant un lien excitateur entre le capteur gauche au moteur gauche et faire le même montage sur le côté droit. Une application de cette stratégie est utilisée par les modèles les plus simples de robots aspirateurs qui, à force de se balader dans une enceinte délimitée juste en évitant les obstacles, vont parvenir à parcourir l'ensemble de la zone à nettoyer.

En inversant les connexions (capteur gauche connecté au moteur droit) on obtient un comportement de homing « agressif » (voir figure 2.2), le robot va accélérer de plus en plus quand il va se rapprocher de la source lumineuse.

2.2.3 Les méthodes de suivi

Il est possible d'obtenir des comportements complexes à partir de ces systèmes simples en jouant sur la source. Par exemple, en utilisant deux sources activées en alternance, on peut avoir un robot effectuant des allers-retours. Si la source est mobile le comportement de homing devient un suivi de source. En remplaçant la source par un véhicule on obtient un système de suivi de convoi dont la déclinaison possible, dans la conduite autonome de véhicule, est le platooning [Avanzini *et al.* 2009].

Des applications dans le domaine de la logistique consistent à avoir des robots qui vont tenter de garder un signal balise, un champ magnétique ou une couleur, au centre de leur capteur. Cela donne des stratégies de suivi de ligne ou de filoguidage permettant d'avoir un robot qui suit une trajectoire prédéfinie. Dans un contexte de

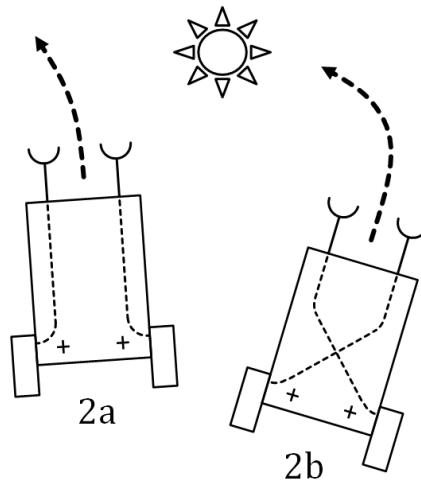


FIGURE 2.2 – L’architecture d’un véhicule de Braitenberg avec les comportements d’évitement d’obstacle (2a) et de homing (2b).

conduite autonome de véhicule, par des traitements visuels, il est possible d’extraire des informations faisant office de signal balise, comme la détection de marquage ou d’orientation de la route, permettant la mise en place d’une stratégie de suivi de ligne.

2.2.4 Les modèles de subsumption

Basé sur les observations qu’avec des règles simples on pouvait voir l’émergence de comportements complexes est née l’idée des modèles de subsumption [Prince 2002][Brooks 1986]. L’idée derrière ces modèles est de partir de réflexes et de comportements simples et d’obtenir des comportements plus complexes grâce à la combinaison de ces mécanismes simples. Les modèles de subsumption sont organisés sous forme de couches hiérarchiques. Les réflexes sont dans les couches les plus basses et plus on monte dans la hiérarchie des couches, plus on peut, grâce à des excitations et inhibitions, combiner différents comportements afin d’obtenir un comportement plus complexe. On peut ainsi obtenir un comportement de navigation robuste capable d’alterner entre plusieurs buts [Mo & Xu 2010]. Les différentes couches sont constituées de machines à état qui fonctionnent en parallèle et reçoivent et envoient des informations de manière asynchrone.

Ces modèles ont l’avantage de fonctionner en temps réel vu que cela fait partie de leur philosophie de conception et de pouvoir être améliorés de manière itérative ou d’être robustes à certaines défaillances du fait de leur architecture en parallèle. Le défaut de ces modèles est la difficulté à concevoir un mécanisme de sélection facilement adaptable à de nouvelles conditions du fait de son architecture distribuée [Arkin *et al.* 1998].

Les robots aspirateurs déjà évoqués en section 2.2.2 sont, pour certains, basés sur

une architecture de subsumption. Ce type de modèle a aussi été testé sur véhicule [Johnson *et al.* 1997].

2.2.5 L'algorithme des fourmis

Avec certaines méthodes de navigation sans carte il est possible de trouver le chemin le plus court pour atteindre une destination. C'est le cas avec l'algorithme des fourmis [Colorni *et al.* 1991]. Cet algorithme utilise un certain nombre d'agents qui fonctionnent avec des règles simples. Chaque agent dépose des phéromones, qui disparaissent progressivement avec le temps, là où il passe. Il choisit la direction dans laquelle il se dirige en fonction du niveau de phéromones qu'il détecte. S'il n'y a pas de phéromone il choisit une direction aléatoire. Quand un agent détecte le point d'arrivée ou que sa mémoire permettant de mémoriser son trajet est pleine, il fait de l'intégration de chemin pour retourner au point de départ avant de repartir. Par interaction entre les agents émerge un comportement permettant de trouver le chemin le plus court. Cette méthode nécessite d'avoir un certain nombre d'agents ainsi qu'un système faisant office de phéromones. De plus il y a une période de convergence. Dans le cadre de la navigation de véhicules, des variations de cet algorithme sont utilisées, par exemple, pour de la planification de trajectoires [Wang *et al.* 2019], de la détection de marquage ou de route [Arnay *et al.* 2008].

2.2.6 L'intégration de chemin

L'intégration de chemin est une méthode qui consiste à calculer la distance relative entre un point de départ servant de référence et le point courant [Cheng 1954]. Ce calcul se fait en sommant les vecteurs de distance et de direction depuis le point de référence (voir figure 2.3). Ce mécanisme a été observé chez plusieurs espèces d'animaux dont une espèce de fourmis du désert du Sahara [Müller & Wehner 1988] est un exemple de l'efficacité de ce système.

Les implémentations de systèmes d'intégration de chemin souffrent d'un problème d'imprécision, qui s'aggrave fortement avec le temps, causé par un phénomène d'accumulation d'erreurs. Ce phénomène est aussi observé chez différents espèces d'animaux, dont l'être humain [Harootonian *et al.* 2020], et provient de l'imprécision des capteurs de mesure, en particulier ceux concernant l'orientation. En effet, les erreurs d'orientations ont un plus fort impact sur l'erreur de positionnement que les erreurs de translation. Pour palier ce problème, les systèmes d'intégration de chemin s'appuient sur des méthodes de recalage. Ces méthodes consistent à utiliser une autre méthode de localisation, souvent du positionnement sur une carte ou de la reconnaissance de lieu, afin d'avoir un ensemble de positions fiables qui servent de jalon. Il suffit de réinitialiser l'intégration de chemin à chaque jalon afin de limiter ces erreurs.

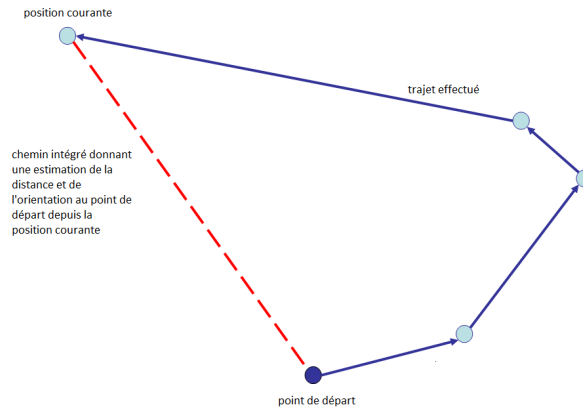


FIGURE 2.3 – Le principe de l’intégration de chemin. Les segments en bleu représentent le chemin parcouru. Le segment en rouge représente l’intégration de chemin qui donne une estimation de la position relative du point de départ au point courant.

2.2.7 Les méthodes de bout-en-bout

L’approche bout-en-bout est relativement récente, son essor durant ces dernières années vient de l’augmentation des capacités de calcul parallèle. Cette méthode consiste à utiliser des réseaux de neurones qui vont prendre en entrée les différentes sorties de capteurs à leur disposition (la plupart du temps des caméras) sans traitement et donner en sortie des tâches ou actions désirées. L’objectif est que le réseau, par apprentissage, arrive à construire les associations entre les perceptions de départ et les actions à l’arrivée. Dans le cas des véhicules autonomes la sortie est généralement soit des ordres d’accélération et d’angle volant ou bien un ensemble de tâches élémentaires (accélérer, tourner à gauche ...).

Il existe principalement trois méthodes pour construire des modèles dits de bout-en-bout. La première repose sur les réseaux de neurones profonds et un apprentissage supervisé [Pomerleau 1989][Bojarski *et al.* 2016]. Le résultat est un agent qui va apprendre à conduire comme un être humain. Cette méthode nécessite d’avoir une base de données labellisées.

La seconde méthode consiste à utiliser un réseau de neurones profond particulier qui utilise des mécanismes d’apprentissage par renforcement en plus de l’apprentissage profond appelés deep Q network (DQN) [Mnih *et al.* 2015]. Un apprentissage par renforcement est une méthode itérative où un agent effectue une action, un observateur évalue cette action et attribue une récompense à l’agent. Ce processus va se répéter avec un agent qui va tester les différentes actions possibles lui permettant de maximiser la récompense qu’il va obtenir. Un deep Q network va apprendre une façon de conduire dite optimale qui ne va pas forcément imiter un conducteur humain. Ce système a été testé avec succès sur une route de campagne [Kendall *et al.* 2018].

Une dernière solution repose sur l’utilisation d’algorithmes génétiques qui vont entraîner un réseau de neurones [Floreano *et al.* 2008]. Des résultats semblent mon-

trer que l'on obtient un meilleur apprentissage qu'avec des méthodes supervisées. [Baluja 1996].

L'inconvénient de la neuroévolution ou des DQN, comparé à l'apprentissage profond supervisé, est la nécessité d'interagir avec l'environnement pour faire les essais et erreurs permettant l'apprentissage. Un autre inconvénient, commun aux trois types de solutions, est la difficulté, inhérente aux réseaux profonds, à interpréter les réponses du modèle rendant difficile la mise en place de mesures de sécurité. Un dernier problème de ces modèles est de s'assurer de leur capacité à généraliser leur apprentissage et donc d'être capable de naviguer sur des routes différentes de celles utilisées lors de l'apprentissage.

2.2.8 Applications actuelles

Une partie du problème de la conduite des véhicules autonomes peut être résolue par des stratégies de navigation réactive. L'évitement d'obstacle peut se faire par des méthodes réactives comme celles décrites plus tôt. Les méthodes de suivi, de véhicule ou de ligne, permettent de faire naviguer des véhicules dans certaines circonstances, comme en conduite normale sur autoroute par exemple ou dans le cas de la circulation en convoi.

Par contre, pour des tâches de navigation plus complexes, aller d'un point de départ à un point d'arrivée, dans l'environnement complexe rempli de bifurcations qu'est le réseau routier, les méthodes de navigation réactives doivent encore faire leurs preuves. En effet pour pouvoir rejoindre sa destination, il faut savoir quelle direction choisir à chaque croisement et donc, soit avoir une idée de la direction générale dans laquelle aller grâce à des méthodes de homing ou d'intégration de chemin, soit de posséder une carte de l'environnement.

2.3 La navigation avec carte

Le principe de base de ce type de navigation est d'avoir une représentation de l'environnement, de pouvoir se situer dans cette représentation, de définir une trajectoire entre la position actuelle et la position cible dans la représentation, et enfin de la transposer au monde réel. La question centrale de ce type de modèles est celle du choix de la méthode de représentation de l'environnement. En effet, c'est de ce choix que dépend le type d'information à extraire de l'environnement et donc les types de capteurs les plus appropriés. La phase de planification est elle aussi dépendante de la structure de la carte utilisée. Une question supplémentaire est celle de l'obtention de la carte de l'environnement. Celle-ci peut être fournie a priori, mais se pose la question de sa fiabilité. Par ailleurs, si on se situe en dehors de l'espace décrit par cette carte, ou si on perd momentanément l'accès à la carte, on se retrouve avec un système incapable de fonctionner de manière optimale.

La carte de l'environnement peut être aussi construite par le modèle de navigation (SLAM voir section 2.4). Dans ce cas on a un système capable de s'adapter aux nouveaux environnements en les cartographiant à la volée. L'inconvénient étant

que la carte étant en cours de construction, elle peut être imprécise (phénomène de dérive).

Il existe plusieurs types de cartes, parmi lesquelles, les trois principalement utilisées sont les cartes métriques, les cartes topologiques et les cartes hybrides. Un autre type de carte, moins utilisé, va aussi nous intéresser dans ce manuscrit : les cartes cognitives.

2.3.1 Les cartes métriques

Les cartes métriques sont une forme de représentation de l'environnement qui conserve les relations spatiales entre les éléments de cette carte.

Il existe plusieurs types de carte métrique en fonction de l'information que l'on veut représenter. Dans le contexte de la navigation en extérieur, les cartes représentent généralement les zones praticables de l'environnement. Elles prennent généralement la forme de grilles d'occupations ou sont basées sur des amers [Jensfelt 2009].

Une carte basée sur des amers est simplement construite à partir d'une collection d'amers observés dans l'environnement en conservant leur relation spatiale. Ces amers peuvent être des angles, des murs, ou le marquage routier par exemple [Pink 2008].



FIGURE 2.4 – Un exemple de représentation sous forme de grille d'occupation avec, en rouge, le trajet parcouru par le robot [Bellekens *et al.* 2015].

Les grilles d'occupations sont un type de carte métrique (voir figure 2.4) très utilisé dans la navigation des robots et véhicules, par les algorithmes de SLAM notamment, qui indiquent l'espace utilisable par le véhicule ou le robot. Elles découpent l'environnement sous la forme d'une grille et indiquent si chaque case de la grille contient un obstacle ou non. La précision de la représentation dépend de la résolution de la grille. Cette forme de carte est généralement construite à partir de données lidar. Elle a l'avantage d'être lisible par un être humain et d'identifier les zones accessibles pour un agent. Une telle carte peut définir une représentation binaire [Elinas & Little 2007], il y a obstacle ou pas, qui a le défaut de ne pas permettre de prendre en compte des obstacles temporaires (véhicules, passants) qui vont fausser la carte. Il existe des variantes, souvent utilisées, qui pallient à ce problème en don-

nant à chaque case de la grille une probabilité d'occupation [Moravec & Elfes 1985] [Ozisk & Yavuz 2008]. Les changements de l'environnement, plus importants en milieu extérieur, peuvent rendre ce type de carte rapidement obsolète.

Enfin, il est possible de construire des cartes métriques à partir de méthodes basées sur des graphes [Grisetti *et al.* 2010] en projetant les scans laser selon la position du véhicule dans le graphe [Lu & Milios 1997].

Les cartes métriques ont l'avantage de refléter la réalité physique des lieux avec plus ou moins de précision, permettant donc de planifier un chemin optimal. La contrepartie de cette fidélité de la carte est un coût computationnel important pour planifier des trajectoires sur cette représentation détaillée. Le gros problème des cartes métriques est d'éviter l'obsolescence, c'est-à-dire de conserver la représentativité de la carte au cours du temps d'un environnement qui est amené à changer (saisons, météo, obstacles, nouveaux bâtiments construits ...).

2.3.2 Les cartes topologiques

Les cartes topologiques sont un type de carte minimaliste se concentrant uniquement sur les relations d'adjacence entre les éléments afin de décrire la structure de l'environnement. Contrairement aux cartes métriques, elles ne s'obligent pas à respecter les relations d'échelles dans la représentation. Ce type de carte peut donc représenter et organiser des informations assez diverses.

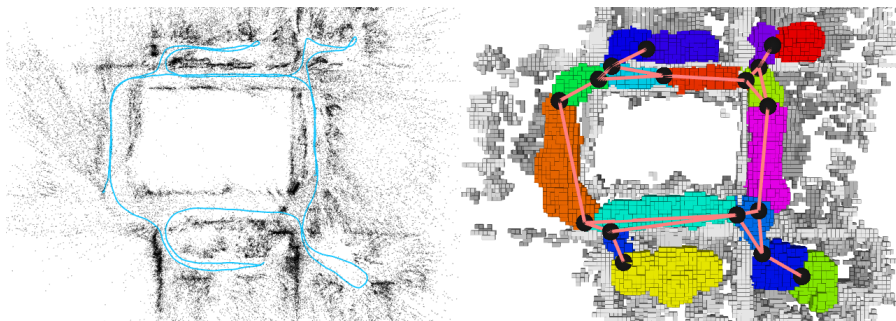


FIGURE 2.5 – Un exemple de carte topologique construite à partir d'une grille d'occupation [Blöchliger *et al.* 2017].

Dans le cadre restreint de la navigation, les cartes topologiques prennent le plus souvent la forme de graphes représentant un ensemble de lieux et les liens entre ces lieux (voir figure 2.5). Parmi les exemples les plus simples il existe les cartes des lignes de métro (voir figure 2.6) et de tramways donnant des lieux, les stations, et leurs liens (telle station est reliée à telle station).

Les cartes topologiques utilisées en navigation sont généralement constituées de lieux, définis de diverses manières, et d'informations spatiales (la distance entre les lieux, l'action à mener pour passer d'un lieu à l'autre, etc.) reliant les lieux entre eux. Elles sont parfois construites à partir de cartes métriques obtenues par des méthodes de SLAM [Blöchliger *et al.* 2017][Thrun 1998].

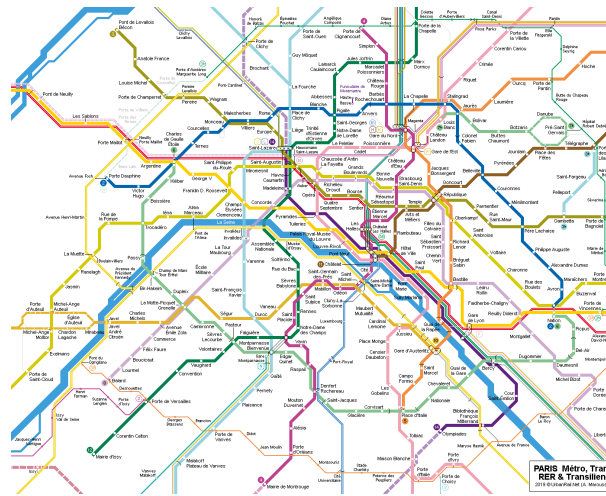


FIGURE 2.6 – Le plan du métro parisien, un exemple de carte topologique.

Les cartes topologiques ont l’avantage de ne pas nécessiter de localisation précise pour être construite et de bien fonctionner à grande échelle du fait de leur représentation minimaliste. De plus, comme elles sont souvent modélisées sous forme de graphes, elles peuvent facilement être utilisées par des algorithmes de recherche du plus court chemin lors des tâches de planification. En contrepartie, elles ont le défaut d’être imprécises et de ne pas garantir le chemin le plus optimal du fait de leur représentation minimaliste. Elles souffrent aussi, dans une moindre mesure du fait de leur représentation minimaliste, du même problème d’obsolescence rencontré par les cartes métriques.

2.3.3 Les cartes hybrides

Les cartes hybrides sont des cartes composées de plusieurs cartes superposées. Ces cartes peuvent être du même type comme dans [Alsayed *et al.* 2015] [Chen *et al.* 2015] qui utilise des grilles d’occupation de tailles différentes afin d’optimiser les temps de calcul. Plus généralement, elles sont composées de cartes métriques et topologiques afin de combiner la précision des cartes métriques avec la facilité à faire de la planification des cartes topologiques. Les méthodes qui construisent des cartes topologiques à partir des cartes métriques obtiennent des cartes hybrides. Un type de carte hybride, fréquemment utilisée, est constituée d’un ensemble de cartes métriques locales disjointes reliées par une carte logique globale [Kuipers *et al.* 2004][Konolige *et al.* 2011]. Elles ont l’avantage de pouvoir représenter de grands environnements en gardant une bonne précision sans occasionner un coût computationnel trop important.

Il existe aussi des modèles qui combinent d’autres types de cartes comme, c’est le cas par exemple de [Ko *et al.* 2013] qui combine une carte métrique avec une carte sémantique, un type de carte contenant des informations sur des objets (feux de circulations, panneaux) et l’autre leur relation avec l’environnement.

Un exemple usuel de l'utilisation d'une combinaison de cartes métriques et topologiques est l'assistant GPS. Ce dernier affiche une carte métrique du réseau routier et donne verbalement des indications, prendre telle sortie pour aller dans telle rue, qui repose sur une carte topologique.

2.3.4 Les cartes cognitives

Les cartes cognitives sont un concept de représentation mentale introduit initialement pour expliquer le comportement de rats qui semblaient apprendre la configuration d'un labyrinthe dans lequel ils étaient placés [Tolman 1948]. Le concept s'est par la suite étendu et désigne à présent l'ensemble des représentations utilisées par les animaux afin d'acquérir, de mémoriser et d'exploiter des informations concernant des lieux et la configuration spatiale de l'environnement. Les cartes cognitives peuvent être une forme neuronale de carte topologique comme dans l'exemple des rats dans un labyrinthe qui semblent mémoriser la structure de leur environnement afin de pouvoir s'y déplacer.

Une hypothèse, largement partagée, suppose que les cartes cognitives seraient en grande partie générées et utilisées par une zone du cerveau qui sera présentée dans le prochain chapitre de ce manuscrit : l'Hippocampe.

2.4 Le SLAM

Les méthodes de SLAM (Simultaneous Localization And Mapping) désignent un ensemble d'algorithmes qui effectuent simultanément la localisation et la cartographie de l'environnement. Ces méthodes ont obtenu d'excellents résultats lors d'expériences en environnements intérieurs menant même à des applications industrielles comme pour la logistique dans des entrepôts par exemple. Comme pour l'ensemble des modèles de navigation, le défi actuel du SLAM est de fonctionner dans de larges environnements extérieurs.

Les méthodes de SLAM reposent, pour la plupart, sur un même principe : utiliser les capteurs de proprioception pour estimer les déplacements relatifs de l'agent et les combiner avec les observations fournies par les capteurs extéroceptif afin de construire une carte. Il s'agit ensuite d'utiliser les observations pour se positionner sur la carte obtenue afin de se localiser. Cette localisation grâce aux observations permet de réaliser un recalage. Ce dernier permet d'estimer les erreurs de déplacements relatifs et de corriger la carte. Une grande partie des modèles de SLAM sont basés sur une implémentation d'un filtrage bayésien [Kalman & Bucy 1961] se décomposant en deux étapes. Lors de la première étape, un modèle prédictif donne une estimation de la position du véhicule. Ce modèle prédictif se base, la plupart du temps, sur des informations proprioceptives. Dans la seconde étape, grâce aux informations issues des capteurs une observation de l'environnement est obtenue. Cette observation est ensuite mise en relation avec la carte courante. Ainsi le système est capable, simultanément, de mettre à jour la carte ainsi que la position du véhicule

dans cette dernière. Le SLAM a deux gros défauts intrinsèques. Son défaut principal est le phénomène de dérive, entraînant des distorsions de la carte, provoqué par l'accumulation d'erreurs, en particulier dues à la proprioception. Cette dernière provient du fait que les algorithmes de SLAM reposent sur l'utilisation d'informations relatives. Le second défaut, qu'il partage avec toutes les architectures modulaires, est qu'il isole le problème de la construction de la représentation de son utilisation future pour naviguer.

Le principe du SLAM tient son origine de la conférence IEEE en automatique et robotique, en 1986, à San Francisco. A partir de la formulation du problème du SLAM lors de cette conférence un certain nombre d'implémentations ont été proposés. Les résultats encourageant obtenus ont entraîné une recherche intensive dans ce domaine qui a engendré une myriade de modèles et d'améliorations. Nous renvoyons les lecteurs intéressés par l'historique du SLAM vers [Durrant-whyte & Bailey]. Les différents algorithmes de SLAM peuvent être classés selon différents critères. Un premier critère peut être le type de capteurs utilisés pour la détection d'amers pour les phases de mise à jour. Le plus souvent les implémentations de SLAM utilisent des lidars (des télémètres lasers) qui donnent des nuages de points [Kim *et al.* 2018][Miyasaka *et al.* 2009][Pierzchała *et al.* 2018][Bellekens *et al.* 2015]. Les caméras sont aussi souvent utilisées par différents modèles de SLAM [Engelhard *et al.* 2011][Strasdat *et al.* 2012][Myung-Jin Jung *et al.* 2004], parfois en complément de lidar [Fremont 2009]. Il existe quelques occurrences d'utilisation de sonar [Li *et al.* 2018][Choi *et al.* 2005] comme capteur principal.

Un autre critère de discrimination est la méthode utilisée pour représenter les données et former une carte. Les principales représentations le plus souvent utilisées sont les cartes métriques (grilles d'occupation ou amers), les cartes topologiques ou les cartes hybrides combinant grilles d'occupation et carte topologique.

Une dernière classification possible se fait selon la méthode utilisée pour incorporer les nouvelles informations. Les algorithmes de SLAM se reposent le plus souvent sur des méthodes de filtrage ou des méthodes d'optimisation.

Un état de l'art récent des principaux algorithmes de SLAM peut être trouvé dans [Cadena *et al.* 2016] et [Bresson *et al.* 2017].

2.5 Discussion

Le problème de la navigation en extérieur, et des véhicules autonomes en particulier, est une question complexe composée d'un ensemble de sous-questions imbriquées. Pour pouvoir naviguer dans les environnements complexes que sont les réseaux routiers, il semble nécessaire de se construire une représentation du monde (carte, point de repère). La capacité à se localiser est indispensable afin de pouvoir exploiter cette représentation.

La méthode actuellement privilégiée pour résoudre le problème de la navigation du véhicule autonome est l'utilisation de la relocalisation dans des cartes. Cette méthode a permis d'obtenir de bons résultats pour de la navigation en intérieur,

notamment à travers des algorithmes de SLAM. Les stratégies reposant sur l'utilisation de cartes rencontrent des difficultés à obtenir des performances similaires en environnement extérieur. Le problème principal vient de la maintenance de la carte afin de garder une représentation fiable de l'environnement.

Il existe plusieurs types de cartes (métriques, topologiques, sémantiques) qui ont leurs avantages et leurs inconvénients. Les environnements extérieurs étant soumis à des changements fréquents (météo, saison, etc) et pouvant comporter de nombreux objets mobiles, la sélection des bonnes informations formant la représentation reste le problème majeur des cartes métriques. Ce problème est partagé par les méthodes de reconnaissance visuelle de lieux. De plus la phase de correspondance entre l'observation locale et sa place dans la représentation est plus ardue. Cette dernière augmente avec la taille de la carte et les obstacles mobiles peuvent fausser une vue locale empêchant la correspondance. Les cartes topologiques et sémantiques sont plus complexes à construire et sont moins précises.

D'un autre côté, on observe toute une série d'animaux (oiseaux migrateurs [Walcott 1996], tortues de mer fourmis ou abeilles) qui sont capables de naviguer efficacement sans nécessairement s'appuyer sur une carte de l'environnement. D'autres animaux, comme les rats, semblent construire des cartes cognitives leur permettant d'obtenir des comportements de navigation dont l'efficacité dépasse encore celles des systèmes robotiques.

Le problème de la navigation en extérieur pose toute une série de questions encore partiellement irrésolues à ce jour :

- Quelle représentation du monde adopter afin de pouvoir naviguer dans l'environnement et comment construire cette représentation ? Celles que l'on utilise actuellement (métriques, topologiques, sémantiques ...) ne permettent pas encore de naviguer aussi bien que les organismes biologiques.
- Comment parvenir à se localiser afin d'exploiter au mieux la représentation du monde ?
- Quelle modalité utiliser afin de percevoir le monde et quelle information en extraire ? Il existe une variété de méthodes qui ont toutes des avantages et des inconvénients. Parmi elles, l'une des plus prometteuses reste la vision largement utilisée par les animaux.

Les animaux qui sont capables de naviguer dans des environnements extérieurs de façon fiable ont apporté une réponse à chacune de ces questions. Par l'observation et la reproduction de mécanismes biologiques utilisés par certains de ces animaux, il doit pouvoir être possible de répondre au moins partiellement à ces questions.

Observations biologiques

Sommaire

3.1	La modélisation issue de la biologie	31
3.1.1	La bio-inspiration	31
3.1.2	La neurorobotique	32
3.2	Observations	33
3.2.1	La vision et l'œil	33
3.2.1.1	Description	34
3.2.1.2	Le champ visuel	36
3.2.1.3	La vision fovéale	37
3.2.1.4	La vision périphérique	37
3.2.2	Le cortex visuel et le corps géniculé latéral	38
3.2.2.1	Le corps géniculé latéral	38
3.2.2.2	V1	38
3.2.2.3	Les zones V2, V3, V4, V5 et V6	41
3.2.2.4	Le modèle dorso-ventral	41
3.2.3	La zone Hippocampique	42
3.2.3.1	Les cellules de lieux	45
3.2.3.2	Les cellules de direction de la tête	46
3.2.3.3	Les cellules de grille	48
3.2.3.4	Cellules de vitesse	49
3.3	Modèles bio-inspirés appliqués à la navigation de véhicules autonomes	50
3.3.1	Les réseaux de neurones profonds	50
3.3.2	RatSLAM	52
3.3.3	Modèle de mémoire visuelle	54
3.3.4	Modèle de carte cognitive de Zeng et Si	56
3.3.5	Beobot	57
3.4	Discussion	59

Les animaux sont de remarquables systèmes de navigation biologiques à la fois très adaptables et efficaces. Le fait que l'on trouve des espèces animales un peu partout sur le globe, donc dans des environnements variés, à l'instar du rat, que l'on retrouve aussi bien dans des mégapoles urbaines que dans des régions sauvages, dans des zones froides jusqu'aux zones tropicales en passant par les zones

tempérées, atteste de leur adaptabilité. Les animaux migrateurs, comme les baleines [Horton *et al.* 2011] ou les oiseaux migrateurs évoqués au chapitre précédent [Walcott 1996] en sont un autre exemple. Ils sont capables d'entreprendre des voyages de plusieurs dizaines de milliers de kilomètres et donc de rencontrer des environnements variés.

La seule existence de ces animaux justifie l'efficacité de leur système de navigation. En effet, pour une grande partie des organismes, la survie dépend entièrement de la capacité à naviguer. Cette dernière leur est indispensable pour rejoindre des sources de nourriture et d'eau, fuir des situations dangereuses ou trouver un partenaire pour la reproduction.

Les animaux ont développé une grande variété de solutions à ce problème de navigation. Elles se traduisent dans leur morphologie par la présence de toute une myriade d'organes sensoriels spécialisés servant à percevoir l'environnement. La vision est une modalité fréquemment utilisée par de nombreuses espèces [Mouritsen 2001] (insectes, oiseaux, primates). Chez certains cette modalité est secondaire et vient en complément d'autres sens comme l'olfaction [Stabell 1984] et, par l'intermédiaire des vibrisses [Kleinfeld & Deschênes 2011], du toucher chez les rongeurs. D'autres encore ont développé des organes leur permettant d'émettre des ultrasons et peuvent donc utiliser une forme d'écho-localisation comme les chauves souris [Jones & Holderied 2007] ou les baleines [Johnson *et al.* 2004] par exemple. Certains animaux pratiquant des migrations, comme des espèces d'oiseaux ou de tortues marines, possèdent un organe sensible aux variations du champ magnétique terrestre.

Les animaux utilisent, généralement, plusieurs modalités sensorielles différentes (vue, ouïe, toucher, odorat et goût chez l'être humain) qu'ils combinent afin de mieux percevoir leur environnement.

En plus de ces organes, ces stratégies de navigations reposent aussi sur tout un ensemble de fonctions cognitives. La perception de l'environnement ne peut se faire sans structures de traitement permettant d'extraire les informations à partir des réponses des organes sensoriels et de les mettre en forme, pour, par exemple, pouvoir fusionner les réponses de plusieurs modalités. Cette manipulation d'information ne peut se faire sans l'existence d'une mémoire de court terme, appelée mémoire de travail, qui stocke les informations le temps d'effectuer les différents traitements nécessaires.

Un mécanisme de mémoire à long terme capable de stocker une information et de la restituer ultérieurement est aussi généralement requis que ce soit pour mémoriser un lieu important ou bien construire une représentation de l'environnement plus complexe basée sur des cartes cognitives [Tolman 1948] par exemple. Ces dernières, en plus d'une mémoire pour les stocker nécessitent des structures qui se chargent de construire ces cartes. Enfin, pour naviguer il faut pouvoir élaborer des séquences d'actions et de mouvements et les traduire en consignes à envoyer aux différents muscles.

Ces fonctions cognitives se traduisent dans la morphologie des animaux par l'existence de tout un ensemble de substrats neuronaux, qui peuvent avoir des variations d'une espèce à une autre (cortex visuel plus développé chez les primates que chez

les rongeurs par exemple [Laramee & Boire 2015]) mais qui sont communes à ces différentes espèces.

Dans la suite de ce manuscrit on se restreindra à la description d'organes et de structures cérébrales présentes chez les mammifères et en particulier chez les primates et les rongeurs. Ces derniers sont le sujet d'étude le plus fréquent de la part des neurobiologistes du fait de leur grande adaptabilité à des environnements très différents, et de considérations pratiques (animaux de petite taille, cycle de reproduction et de maturation rapide) facilitant la mise en œuvre d'expériences. Les primates, quant à eux, ont un système visuel plus développé qui leur sert de modalité principale pour les tâches de navigation.

Les animaux sont capables de naviguer et de s'adapter à des environnements complexes tout en étant robustes à des changements comme ceux, temporaires, dus à des variations météorologiques ou d'autres, cycliques, comme l'alternance jour nuit ou le passage des saisons. Les systèmes robotiques comme le sont les véhicules autonomes sont encore incapables d'obtenir des mécanismes de navigation aussi efficaces.

Il pourrait donc être intéressant d'observer en détail le fonctionnement des différents mécanismes mis en œuvre par les animaux afin de naviguer et de s'en inspirer afin de concevoir des modèles robotiques capables d'exhiber des capacités de navigation similaires qui seraient donc plus performants. Cette démarche entre dans le cadre de ce que l'on nomme la bio-inspiration.

Ce chapitre commencera par détailler les notions de bio-inspiration et de neurorobotique avant de se lancer dans une description de l'œil et d'un ensemble de structures cérébrales et de neurones spécialisés impliqués dans les processus de perception et de cognition spatiale utilisés dans la navigation. Il donnera ensuite un état de l'art de modèles bio-inspirés ou neurorobotiques testés en environnement extérieur et généralement sur un véhicule autonome avant de discuter des apports potentiels d'un approche bio-inspirée.

3.1 La modélisation issue de la biologie

3.1.1 La bio-inspiration

La bio-inspiration est une démarche consistant à avoir une approche créative basée sur l'observation des systèmes biologiques comme les plantes ou les animaux. Ce concept est défini par la norme ISO TC 266 - ISO 18458. Il est entouré de tout un ensemble de notions. Par exemple, la coopération entre la biologie et d'autres disciplines (informatique, mécanique, architecture) afin de résoudre des problèmes par l'analyse des systèmes biologiques est de la biomimétique. L'amélioration ou le remplacement de fonctions biologiques par des systèmes artificiels (électronique et/ou mécanique) constitue la bionique. Enfin, l'approche conceptuelle visant à résoudre des problèmes liés au développement durable en prenant la nature pour modèle est nommée biomimétisme.

On retrouve des exemples de bio-inspiration dans de nombreux domaines variés parmi lesquels l'aéronautique, les matériaux, la médecine ou bien l'intelligence artificielle.

L'aviation, qui a contribué au bouleversement de la mobilité au vingtième siècle, est un exemple de locomotion, en partie, bio-inspirée . De nombreux essais infructueux de reproduction du vol des oiseaux ont été faits dans l'histoire, comme l'exemple de la machine volante de Leonardo da Vinci. A partir de l'étude de la forme des ailes des oiseaux on a pu déduire le principe physique de la portance leurs permettant de voler. La compréhension de ce principe ainsi que la prise en compte de contraintes techniques (impossibilité de reproduire des muscles souples, passage à une échelle plus grande) a abouti au modèle de l'avion qui vole grâce à ses ailes fixes et un déplacement linéaire qui doit toujours rester supérieur à un seuil plancher. Les ailes fixes auraient aussi l'avantage de requérir moins d'énergie pour pouvoir voler même si certains travaux [Pesavento & Wang 2009] semblent contester cette assertion.

Dans le cas de la localisation pour la navigation, on va se concentrer sur une partie de la bio-inspiration issue de l'hybridation avec la robotique qui se focalise sur le système nerveux et sur les aspects ayant trait à la cognition : la neurorobotique.

3.1.2 La neurorobotique

La neurorobotique est un champ de recherche s'inscrivant dans la bio-inspiration qui fait le lien entre les domaines de la robotique et des neurosciences. La neurorobotique consiste à élaborer, à partir des travaux menés par les neurosciences, des modèles computationnels qui reproduisent des mécanismes neuronaux de certaines structures cérébrales. La modélisation peut se faire au niveau macroscopique en considérant des fonctions cognitives ou des concepts (mémoire de travail, construction de carte cognitive, ...) et leurs interactions. Elle se fait aussi au niveau mésoscopique en reproduisant des activités résultantes de populations ou microscopique en incluant les propriétés particulières des neurones. Ces modèles computationnels sont implémentés sur robot afin de reproduire des expériences, issues des neurosciences, afin de valider ou d'invalider ces modèles en comparant leurs résultats aux observations des expériences initiales et de réaliser des prédictions falsifiables par la réalisation de nouvelles expériences chez l'animal.

Le but des neurosciences est de comprendre le fonctionnement du système nerveux et des processus entraînant l'émergence de la cognition. Elle étudie le cerveau à plusieurs niveaux. Au niveau microscopique, elle se concentre sur le neurone isolé, afin d'identifier les propriétés de différents types de cellules nerveuses et caractériser leurs différentes réponses. Au niveau mesoscopique, elle observe des populations de neurones afin de déterminer leurs interactions et leur organisation au sein du cerveau. Enfin, au niveau macroscopique les neurosciences analysent le fonctionnement global du cerveau en observant les réactions à un stimulus de différentes structures cérébrales ainsi que leurs interactions entre elles.

Du point de vue des neurosciences, l'intérêt de la neurorobotique est de pouvoir

tester les hypothèses et les modèles qui ont été élaborées afin de les valider ou de les invalider. Dans le cas d'une démarche de modélisation incrémentale, l'analyse des limites des modèles computationnels est susceptible d'indiquer l'absence d'une interaction avec une structure cérébrale non prise en compte par le modèle. Ceci peut mener à l'élaboration de nouvelles expériences de neurosciences afin de détecter cette possible interaction.

Du point de vue de la robotique le but est, à terme, d'obtenir un modèle computationnel reproduisant les capacités d'adaptation et de survie des animaux. Cela donnerait des robots capables d'accomplir des tâches (livraison, inspection) et d'évoluer dans l'environnement de manière robuste tout en étant vraiment autonomes, en pouvant, par exemple, retourner à une station de recharge ou s'adapter à des situations imprévues ou requérir de l'aide en cas d'échec.

3.2 Observations

Dans le cadre de ce manuscrit présentant des travaux sur la reconnaissance de lieux visuelle pour la localisation et la navigation, les observations vont se focaliser sur quatre structures que l'on retrouve chez la majorité des mammifères à savoir l'œil, un organe sensoriel, le cortex visuel et le corps géniculé latéral, des structures neuronales traitant les informations issues de l'œil, et la zone hippocampique, un ensemble de structures neuronales impliquées dans la cognition spatiale.

3.2.1 La vision et l'œil

La vision est une modalité riche permettant d'extraire bon nombre d'informations déjà évoquées au chapitre 2. On retrouve ce sens chez une grande partie du règne animal. Elle est aussi bien exploitée par certaines espèces d'insectes que par la quasi totalité des vertébrés, à quelques exceptions près comme le *Dibamus dalaiensis*. Chez les primates, elle est la modalité principale et se traduit dans leur morphologie par un ensemble d'organes capable d'extraire l'ensemble de cette richesse d'information (contraste, couleurs, estimation de profondeur, etc.).

L'œil est le premier organe du système visuel. Sa fonction est de capter la lumière issue de l'environnement, de la transformer en influx nerveux, de réaliser certains traitements de bas niveaux avant de la transmettre au cortex visuel.

Chez les primates, les informations perçues par les yeux peuvent être divisées en deux zones, la vision périphérique et la vision fovéale.

La vision périphérique fournit des informations globales sur une scène visuelle alors que la vision fovéale fait une focalisation plus détaillée sur une partie restreinte de cette scène visuelle. Cette section va commencer par la description de l'œil chez l'être humain avant de poursuivre par des considérations sur le champ visuel et de terminer par présenter les visions fovéale et périphérique.

3.2.1.1 Description

Les différents types d'yeux, fonctionnent comme une caméra. Ils ont une chambre dans laquelle est captée la lumière qui se projette ensuite sur une surface photoréceptrice par une projection conique. Les plus simples ne sont capables que de percevoir des variations de luminosités alors que d'autres, bien plus complexes, sont capables de capter une grande richesse d'information comme le font ceux des primates, dont les humains. Ce sont ces derniers qui vont être décrits.

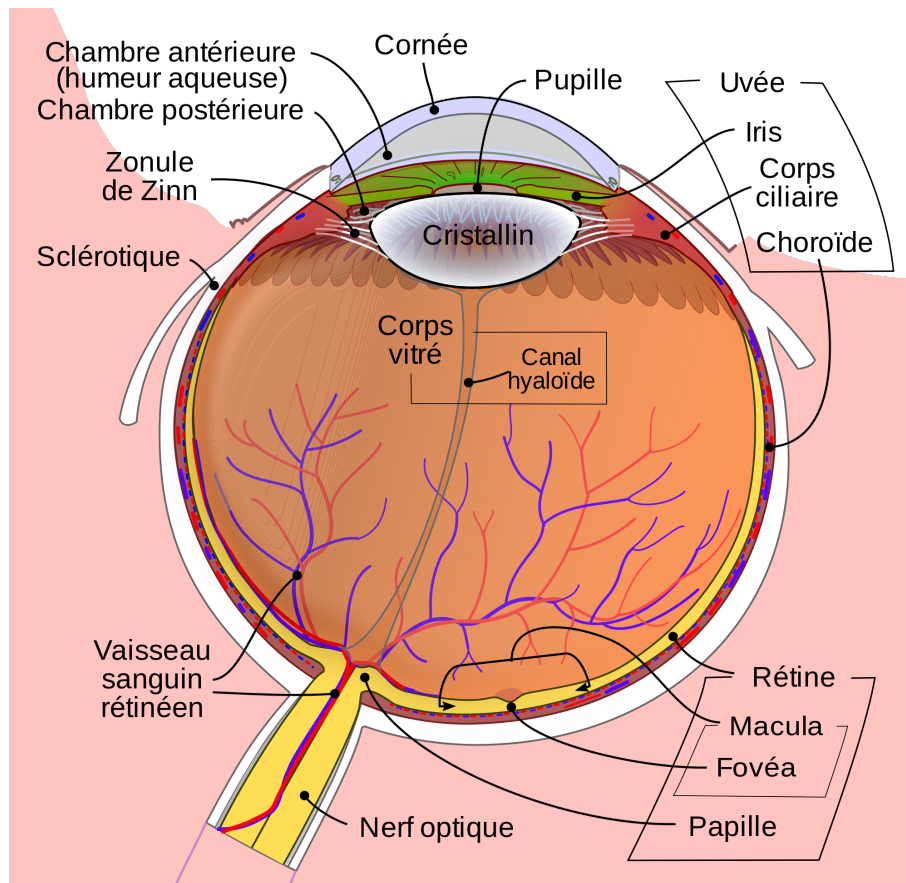


FIGURE 3.1 – Diagramme schématisique d'un oeil humain

L'oeil possède (voir figure 3.1) un iris lui permettant de s'adapter à la luminosité ambiante. Le cristallin fait la mise au point et sert donc d'autofocus. La rétine est la surface sur laquelle est projetée la lumière. La partie photosensible de la rétine comporte deux types de cellules photoréceptrices extrayant des informations spécifiques, les bâtonnets [Curcio *et al.* 1990], présents en forte densité en périphérie de la rétine, et les cônes [Curcio *et al.* 1990], majoritairement situés dans la zone fovéale.

La figure 3.2 donne la répartition des deux types de cellules photo-réceptrices. La zone fovéale comporte plus de cellules permettant d'avoir une perception plus fine.

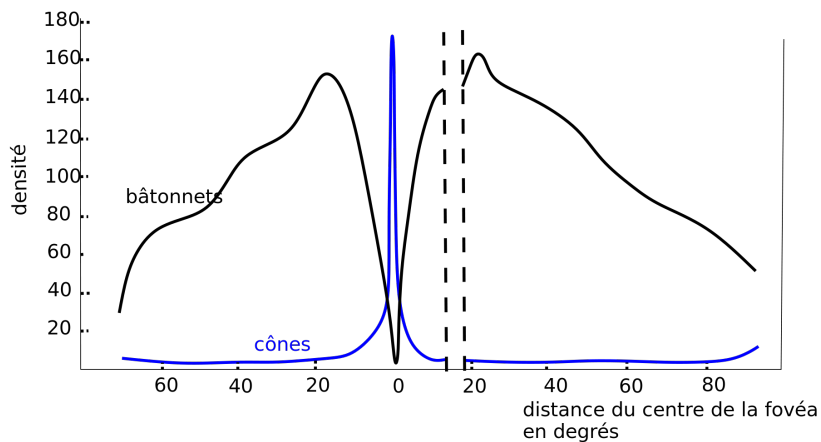


FIGURE 3.2 – Répartition des cônes (en bleu) et des bâtonnets (en noir) par rapport au centre de la fovéa. La zone sans photorécepteur correspond à la papille, la zone où commence le nerf optique.

En plus de cette couche de cellules photoréceptrices, la rétine est composée de quatre couches neuronales (voir figure 3.3) :

- la couche plexiforme externe composée de cellules horizontales [Masland 2012a]
- la couche nucléaire interne formée de cellules bipolaires [Masland 2001]
- la couche plexiforme interne constituée de cellules amacrines [Masland 2012b]
- la couche des cellules ganglionnaires [Erskine & Herrera 2014]

Les cellules horizontales [Masland 2012a] sont des neurones possédant de nombreuses synapses les reliant à un ensemble de cellules photoréceptrices de la rétine. Elles jouent un rôle régulateur auprès des cellules photoréceptrices en intégrant les réponses de ces cellules photoréceptrices afin de faire des opérations de moyennage ou de rehaussement de contraste.

Les cellules bipolaires sont elles aussi reliées aux cellules photoréceptrices. Leur connexion varie selon la position au sein de la rétine. Au niveau de la fovéa chaque cellule photoréceptrice est liée à une seule cellule bipolaire, résultant en une grande acuité visuelle dans cette zone. Plus on s'éloigne de la fovéa, plus le nombre de cellules photoréceptrices reliées à une même cellule bipolaire augmente entraînant une baisse de l'acuité visuelle. Ces cellules peuvent être de trois types :

- ON : la cellule bipolaire répond quand la cellule photoréceptrice reçoit de la lumière
- OFF : la cellule bipolaire répond quand la cellule photoréceptrice ne reçoit pas de la lumière
- ON/OFF : la cellule bipolaire répond quand intervient un changement de lumière au niveau de la cellule photoréceptrice

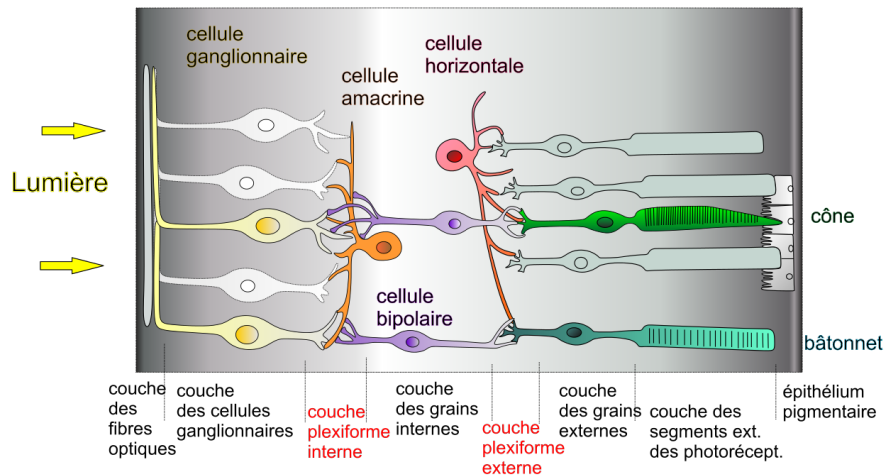


FIGURE 3.3 – Schéma détaillant les quatre couches constituant la rétine ainsi que les six types de neurones (batonnets, cônes, cellules horizontales, cellules bipolaires, cellules amacrines et cellules ganglionnaires) formant ces couches [Purves 2001]

Les cellules amacrines sont à l'interface des cellules bipolaires et ganglionnaires. Elles semblent exercer une forme de régulation des signaux provenant des cellules bipolaires en étant capables d'inhiber les cellules ganglionnaires.

Les cellules ganglionnaires font le lien entre la rétine et le reste du cerveau. Elles constituent le nerf optique. Elles héritent de la sensibilité (ON, OFF et ON/OFF) des cellules bipolaires auxquelles elles sont connectées dont elles traduisent les réponses en potentiels d'action (signaux électriques de certains neurones). Certaines cellules ganglionnaires sont intrinsèquement photosensibles même si elles réagissent lentement au stimuli lumineux [Do & Yau 2010]. D'autres sont orientées [Passaglia *et al.* 2002], elles ont des réponses variant selon la direction que suit un stimulus visuel.

L'ensemble de ces cellules permet à la rétine de procéder à l'extraction des contrastes.

3.2.1.2 Le champ visuel

Chez les mammifères les yeux vont par paire mais leur orientation sur la tête varie selon les espèces animales (voir figure 3.4). Chez l'être humain les axes optiques des yeux sont parallèles alors que chez la girafe ces axes sont de 120 degrés.

Selon l'orientation des yeux, on altère le ratio entre la taille du champ visuel et le recouvrement des champs visuels de chaque œil formant le champ visuel binoculaire. Les proies ont des axes optiques avec un grand angle pour maximiser la taille du champ visuel afin de détecter plus facilement les prédateurs. En contrepartie leur champ visuel binoculaire est très réduit. Au contraire, les prédateurs ont des yeux plus rapprochés leur donnant un champ visuel plus réduit mais augmentent la taille

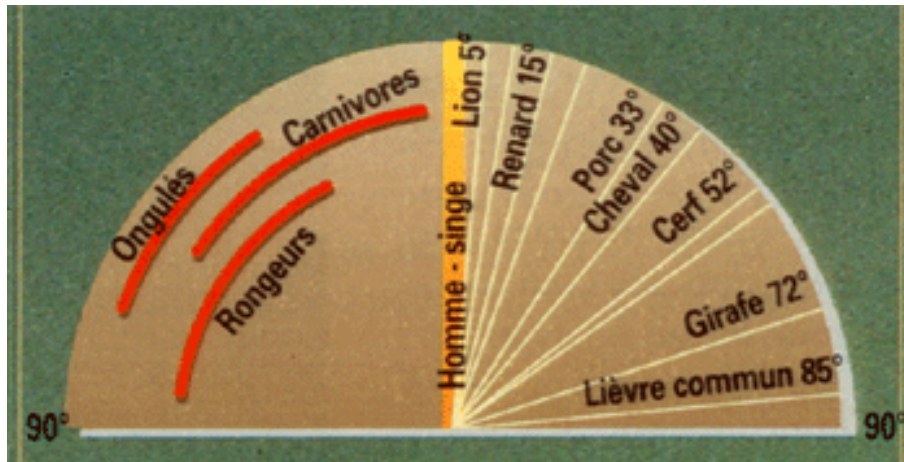


FIGURE 3.4 – Diagramme donnant l'angle d'orientation des yeux pour différents espèces de mammifères

du champ visuel binoculaire. Ce dernier leur permet de plus facilement traquer les proies grâce à l'obtention de l'information de profondeur. Cette orientation des yeux a une influence sur la représentation de l'espace qui sera décrite plus loin dans ce manuscrit (voir section 3.2.3.1).

3.2.1.3 La vision fovéale

La vision fovéale est la partie de la vision qui se base sur les informations captées par la fovéa. Le processus de vision fovéale est long (de l'ordre de 3 ou 4 images par secondes). Ceci est dû en partie au mécanisme de fixation de l'œil qui s'arrête pour faire une mise au point (200 ms [Krauzlis *et al.* 2017]) afin de capter des informations. La fovéa est munie de capteurs qui permet de percevoir les couleurs et d'avoir une haute résolution d'où le temps de fixation. La vision fovéale donne donc une image très détaillée et en couleurs d'une petite portion du champ visuel. Il existe cependant des tâche particulières, comme la reconnaissance de visage [Martin *et al.* 2018] où le processus de vision fovéale s'exécute plus rapidement.

3.2.1.4 La vision périphérique

La vision périphérique effectue un traitement rapide des informations (de l'ordre d'une centaine d'images par seconde). Elle utilise essentiellement les informations issues des bâtonnets. Elle extrait des informations basses résolutions en noir et blanc sur l'ensemble du champ visuel. Elle est aussi très performante dans la détection de mouvement. La vision périphérique donne une impression globale d'une scène visuelle et fournit des informations exploitées par la vision fovéale. Chez l'être humain, elle permet aussi de faire une reconnaissance précise, en détectant des mots dans le cadre de la lecture rapide [Yu *et al.* 2018], par exemple.

3.2.2 Le cortex visuel et le corps géniculé latéral

Les informations obtenues par la rétine sont transmises au reste du cerveau par l'intermédiaire du nerf optique, constitué de cellules ganglionnaires, qui mènent pour la plupart au corps géniculé latéral [Nieuwenhuys 2008]. C'est une structure latérale qui à son tour communique avec le cortex visuel.

Le cortex visuel, situé dans le lobe occipital, est une partie du cortex cérébral dont le rôle est de traiter les informations visuelles.

Chez les primates [Solomon & Rosa 2014] [Fattori *et al.* 2009], le cortex visuel peut être découpé en six zones respectivement appelées V1, V2, V3, V4, V5 et V6. Pour la suite on se concentre sur la description du cortex visuel primaire V1 des primates, qui est parmi ceux les plus développés. Cette partie ne donnera qu'une description sommaire du reste du cortex visuel car seul le cortex visuel primaire est pertinent pour les modèles présentés dans ce manuscrit.

3.2.2.1 Le corps géniculé latéral

Le corps géniculé latéral (CGL) est une structure cérébrale composée de six couches neuronales. Il est constitué par trois types de cellules réparties de manière ordonnée (voir figure 3.5) :

- les cellules magnocellulaires, de grande taille, sur les deux premières couches.
- les cellules parvocellulaires, plus petites, sur les quatre dernières couches.
- les cellules koniocellulaires, encore plus petites, sur les six couches intercalées.

La projection de la rétine sur le corps géniculé latéral se fait de manière structurée selon les voies magnocellulaire, parvocellulaires et koniocellulaires (voir figure 3.5). La voie parvocellulaire transmet des informations sur de fins détails spatiaux issues de la zone fovéale et sur l'opposition de couleur rouge/vert. La voie magnocellulaire relaie les signaux issus de la partie périphérique de la rétine et participe à la détection de mouvements. La voie koniocellulaire, encore très mal connue, participe à la vision des couleurs en fournissant la dimension jaune/bleu.

Cette projection conserve la topologie de la rétine sur le corps géniculé latéral, deux cellules proches sur la couche photoréceptrice vont exciter deux cellules proches dans le CGL. Cette projection est dite rétinotopique. Le corps géniculé latéral est fortement connecté au cortex visuel primaire. D'une part, le CGL se projette sur le cortex strié (V1) en conservant l'organisation structurée de la projection de la rétine au CGL. D'autre part, le cortex visuel primaire exerce, en retour, une forte rétroaction sur le corps géniculé latéral.

3.2.2.2 V1

Le cortex visuel primaire V1 (voir figure 3.6) ou cortex strié est la première aire visuelle qui reçoit les informations depuis le corps géniculé latéral. Du fait de cette connexion cette aire est aussi rétinotopique. C'est la zone du cortex visuel la plus connue et étudiée car, en plus d'être facile d'accès pour les différentes méthodes de mesures et d'imageries vu sa position proche dans le cerveau, les stimuli qui activent

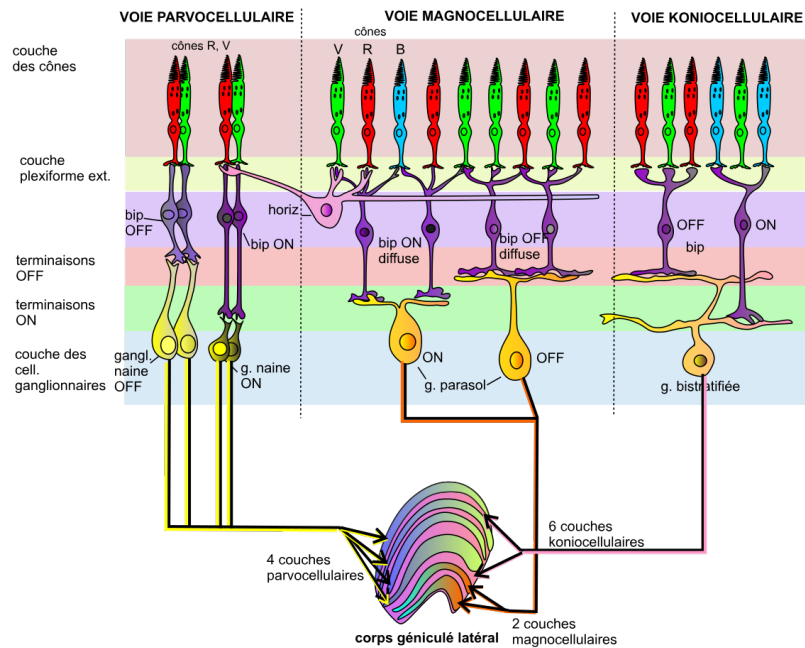


FIGURE 3.5 – Le corps géniculé latéral avec les trois voies (parvocellulaire, koniocellulaire et magnocellulaire) de projection depuis la rétine [Nieuwenhuys 2008].

les neurones de cette zone sont, en partie, connus et facilement générables avec des ordinateurs.

Le cortex visuel primaire est situé à l'arrière du néocortex sur les deux hémisphères, la partie droite traite les informations issues de l'œil gauche et inversement. Cette zone est constituée de 6 couches neuronales très denses distinctes. Le qualificatif strié vient de cette distinction visible à l'œil nu. Cette aire est composée de l'ordre de 140 millions de cellules chez un être humain.

Elle reçoit des signaux depuis le CGL qu'elle envoie après traitement aux autres aires du cortex visuel. Elle reçoit aussi des retours depuis les différentes aires visuelles et envoie un retour au Thalamus. Elle reçoit aussi des signaux depuis d'autres zones du cerveau qui selon les hypothèses actuelles auraient des effets de modulation, liés à l'attention par exemple.

Le cortex visuel primaire possède une organisation rétinotopique. Cela signifie que le champ neuronal reproduit l'organisation spatiale, à une transformation près, de la rétine et que deux neurones géographiquement proches vont réagir pour des portions de la rétine elles-mêmes proches. La zone V1 contient ainsi une projection complète du champ visuel capté par les deux yeux. Cette projection engendre relativement peu de distorsions locales même si il y a une distorsion globale qui est encore plus importante chez les primates. Cette distorsion effectue une magnification de la partie centrale du champ visuel. Ainsi, chez l'être humain la moitié du cortex visuel primaire sert à traiter 2 pourcents du centre du champ visuel. Une autre distorsion

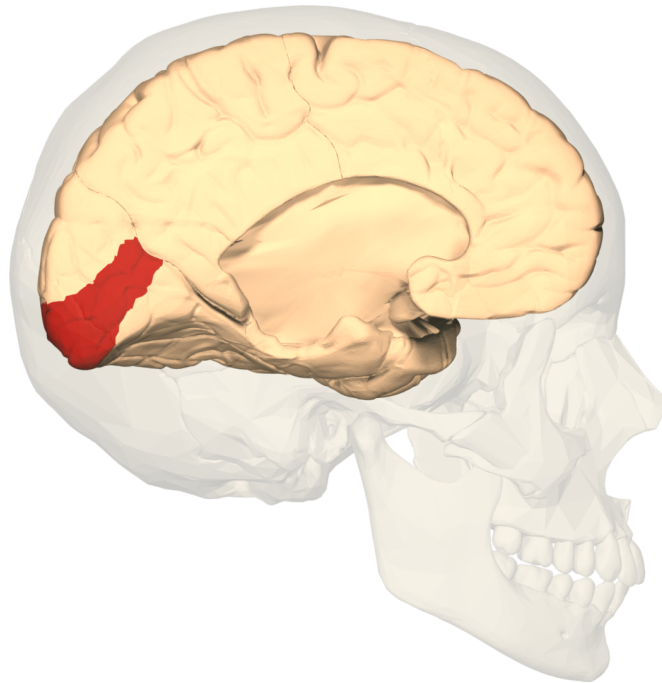


FIGURE 3.6 – La position du cortex visuel primaire, zone 17 de Broadman, dans le cerveau humain.

est similaire à une projection en log polaire qui transforme les cercles concentriques et les lignes radiales en lignes verticales et horizontales.

Le cortex visuel primaire est constitué principalement de deux types de neurones :

- **les cellules simples** [Hubel & Wiesel 1959] : ces cellules répondent à des stimuli linéaires et stationnaires. Leur réponse est sélective selon l'orientation et la position du stimulus dans le champ récepteur. Il existe une grande variété de ces cellules qui répondent à des orientations et des positions dans le champ visuel différentes. Une cellule simple peut être modélisée par un filtre linéaire suivi d'une fonction de seuil [Lindeberg 2010]. Par exemple, les cellules simples qui réagissent aux fréquences spatiales et à des orientations précises peuvent être modélisée sous forme de filtres de Gabor.
- **les cellules complexes** [Hubel & Wiesel 1962a] : ces cellules réagissent à des stimuli en mouvement. Comme les cellules simples, elles ont des réponses dépendant de l'orientation. En revanche, elles ne sont pas sélectives concernant la position du stimulus dans le champ visuel. De plus certaines cellules ont une activité qui est sélective selon la direction du mouvement du stimulus. Il existe deux hypothèses permettant de modéliser les cellules complexes. La première est que les cellules complexes forment leur réponse en effec-

tuant une sommation des réponses d'un ensemble de cellules simples ayant le même champ récepteur mais une répartition différentes des zones inhibitrices et excitatrices [Hubel & Wiesel 1959][Hubel & Wiesel 1962a]. L'autre hypothèse est qu'un ensemble de cellules simples partageant les mêmes préférences d'orientation possèdent des connexions latérales les reliant entre elles. Si les excitations mutuelles sont assez élevées, cet ensemble de cellules simples devient des cellules complexes [Chance *et al.* 1999].

De plus, les cellules simples ou complexes peuvent appartenir à une sous catégorie : les cellules de détection de fin de contour (end-stopped cells), parfois appelées cellules hypercomplexes [Yazdanbakhsh & Livingstone 2006]. Ces cellules, en plus de présenter les propriétés propres à leur type (simple ou complexe), sont aussi sensibles à la longueur du stimulus.

Par ailleurs, certains neurones du cortex visuel primaire sont aussi sensibles aux couleurs. Ils semblent être organisés selon trois directions cardinales correspondant aux trois bandes de longueurs d'onde auxquelles les cônes de la rétine sont sensibles. Ces trois axes sont l'axe noir-blanc, l'axe rouge-vert et enfin l'axe bleu-jaune.

Pour finir, des expériences [Haggerty *et al.* 2015] ont permis d'identifier des neurones dans la zone V1 qui auraient des propriétés d'encodage de l'espace similaires à celles de cellules présentées plus loin dans ce manuscrit : les cellules de lieu hippocampiques (voir partie 3.2.3.1). De plus la mesure des délais d'activation semblent indiquer que l'activation de ces cellules corticales dans V1 précèderaient l'activation des cellules de lieux hippocampiques.

3.2.2.3 Les zones V2, V3, V4, V5 et V6

Le reste du cortex visuel est constitué d'abord de V2 qui après des traitements similaires à ceux de V1 avec en plus des traitements liés à l'information de profondeur envoie des stimuli à V3, V4, V5 et en renvoie vers V1. V3 se divise en deux parties dont l'une participerait à la perception du mouvement et l'autre à la détection de couleur. V4 effectuerait des traitements similaires à V1 et V2 mais avec la détection de forme simples et une sensibilité à une modulation par les processus attentionnels [Nandy *et al.* 2017]. V5 alimenté par V1, V2 et V3 analyserait des informations de mouvements d'objets que ce soit leur vitesse ou leur direction. Enfin V6 stimulé par le cortex visuel primaire V1 se concentrerait sur les informations du mouvement propre à l'animal.

Il existe une modélisation de l'ensemble du fonctionnement du cortex visuel et des interactions entre ses différents zones communément admise : le modèle dorso-ventral.

3.2.2.4 Le modèle dorso-ventral

Le traitement neuronal des informations visuelles est généralement modélisé selon l'hypothèse des deux flux ou modèle dorso-ventral [Goodale & Milner 1992] (voir figure 3.7). Cette modélisation s'applique aussi aux informations sonores. Les informations perçues et pré-traitées par les yeux sont envoyées au cortex visuel primaire

V1. Une fois traitée par le cortex visuel primaire les informations sont transmises selon deux flux, le flux dorsal, le « où » est l'information, et le flux ventral, le « quelle » est l'information.

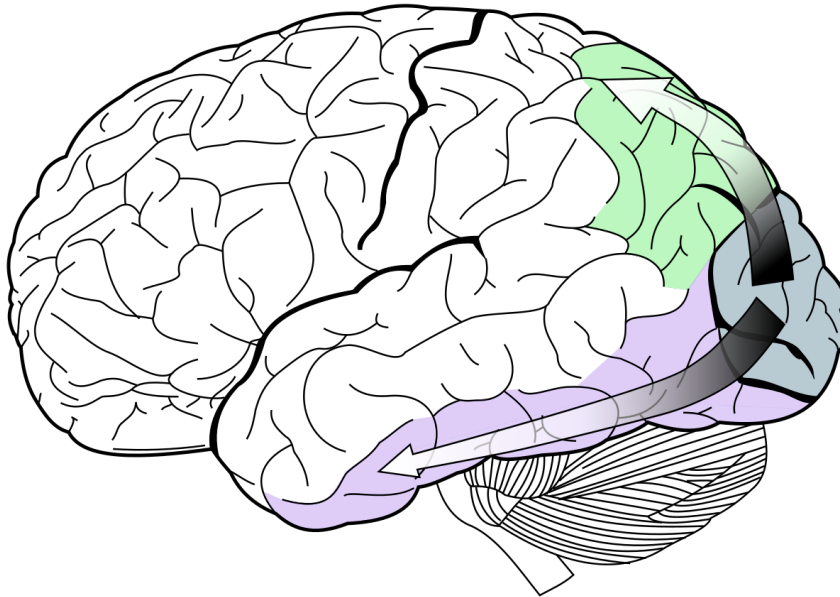


FIGURE 3.7 – Le modèle dorsoventral de transmission des informations depuis le cortex visuel primaire (en gris) vers les zones dorsales (en vert) et ventrales (en violet)

Le flux ventral part du cortex visuel primaire V1 et passe à travers les zones V2 et V4 avant de rejoindre le cortex temporal inférieur, une région dont la fonction semble être liée à la reconnaissance d'objets. Les transformations des données effectuées selon cette chaîne de traitement concernent la reconnaissance et la représentation des objets.

Le flux dorsal part du cortex visuel primaire, passe par les zones V2, V5 et V6 avant de rejoindre le cortex pariétal supérieur, une zone impliquée dans la planification de mouvement et l'attention. Les traitements selon ce flux concernent la représentation des positions des objets dans l'espace et le contrôle moteur des yeux et des bras [Norman 2002]. Une série d'études semble contredire ou tout du moins tempérer cette différenciation nette en deux voies de traitements [Ganel & Goodale 2003][Franz *et al.* 2000].

3.2.3 La zone Hippocampique

L'hippocampe est une zone cérébrale située profondément dans le cerveau. C'est une structure qui se retrouve chez tous les mammifères même s'il existe des variations dans la forme générale de cette zone en fonction des espèces. Chez l'être humain elle possède une forme similaire à l'animal marin d'où elle tire son nom.

Cette zone se décompose en trois structures : le gyrus dentelé, le subiculum ainsi que la corne d'Ammon. Cette dernière structure est elle même divisée en trois aires : CA1 CA2 et CA3. Avec le cortex enthorinal et le subiculum, elle forme ce que l'on appelle la zone hippocampique (voir figure 3.8).

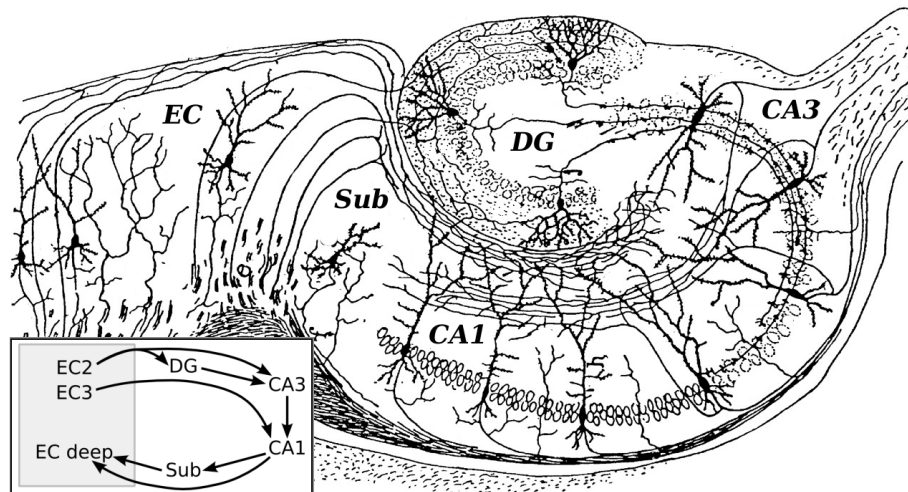


FIGURE 3.8 – La formation hippocampique dessinée par Santiago Ramon y Cajal. EC : Cortex Entorhinal, DG : Gyrus Dentelé , CA1-CA3 : Cornes d'Ammons (hippocampe), sub : Subiculum [Ramon 1909].

Cette région est impliquée dans le processus de mémorisation, en particulier la mémoire épisodique, ou autobiographique, comme l'a montré le cas d'Henry Gustav Molaison. C'était un homme souffrant d'une forme d'épilepsie résistant aux traitements médicamenteux et qui a subi une opération chirurgicale, en 1953, consistant en une ablation partielle de l'hippocampe. Suite à cette opération il souffre d'amnésie antérograde [Scoville & Milner 2000].

La zone hippocampique serait aussi capable de réaliser une compression et un encodage des informations. Par l'intermédiaire du cortex entorhinal, l'hippocampe est relié à l'ensemble des zones du néocortex avec lesquelles il échange des informations [Andersen 2007]. Du fait de cet échange d'information bilatéral, du néocortex vers l'hippocampe lors des phases d'exploration, et de l'hippocampe vers le néocortex lors des phases de repos, on peut en déduire que l'ensemble du néocortex est représenté d'une manière compressée dans l'hippocampe ce qui mène à cette hypothèse sur la capacité de cette région à compresser les informations.

Cette zone est aussi impliquée dans les processus de navigation. Ce fait est attesté par la présence dans l'hippocampe de neurones qui encodent des portions de l'espace [O'Keefe & Dostrovsky 1971] et de cellules servant à l'encodage de la proprioception dans le cortex entorhinal. De ces observations est née l'hypothèse que dans cette zone se mettrait en place les processus menant à la formation de cartes cognitives [Tolman 1948]. Pour rappel, ce serait une forme de représentation utilisée par les animaux afin d'encoder des configurations spatiales et des relations

entre objets. Pour les tâches de navigation, ces représentations pourraient prendre une forme se rapprochant des cartes topologiques.

Comme la zone hippocampique est à la fois employée pour la cognition spatiale et liée à la mémoire épisodique, a été émise l'hypothèse que les processus neuronaux impliqués dans les mémoires épisodique et spatiale ne seraient que des utilisations particulières de mécanismes d'encodage d'évènement ou de séquences d'évènements plus génériques [Buzsaki 2006].

Dans cette formation hippocampique, on observe un certain nombre de neurones liés à la navigation. Parmi ces cellules, trois types qui sont intensivement étudiés et utilisés vont particulièrement nous intéresser : les cellules de lieux, les cellules de direction de la tête et les cellules de grille (voir figure 3.9). Un dernier type de cellules découvert récemment, les cellules de vitesse, sera lui aussi décrit brièvement.

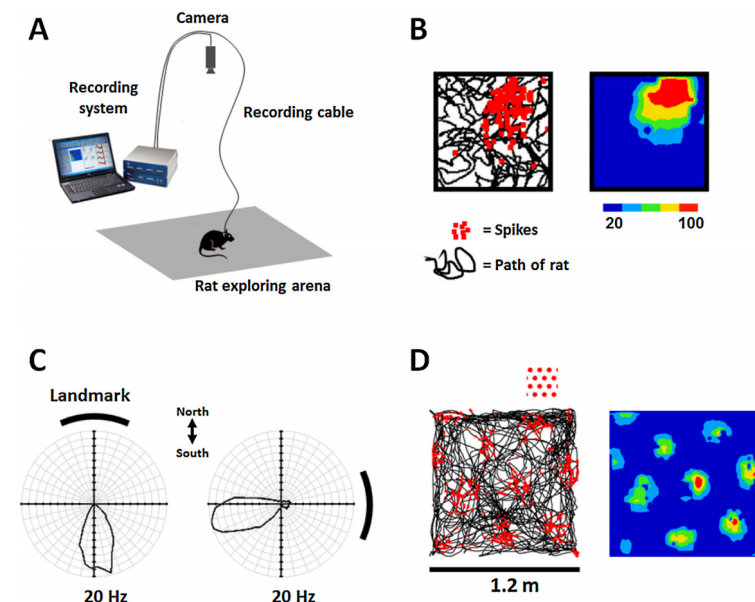


FIGURE 3.9 – Un exemple de dispositif de mesure ainsi que les principaux types de neurones identifiés impliqués dans la cognition spatiale **A** - Exemple de dispositif utilisé pour mesurer des activités neuronales. Une caméra zénitale enregistre les déplacements du rat sur lequel est implanté une électrode. **B** - Un enregistrement de l'activité d'une cellule de lieu dans une arène. L'image de gauche représente le trajet parcouru par le rat en noir avec les pics d'activation dessinés en rouge. L'image de droite représente l'activité moyenne du neurone dans cette même arène. Les zones en bleu correspondent à une activité quasi nulle et plus on se rapproche du rouge plus l'activité est élevée. **C** - Enregistrement d'activités d'une cellule de direction de la tête. A gauche, un amer situé au nord provoque une activation du neurone quand l'animal est orienté vers le sud. A droite, déplacement de l'amer vers l'est, le neurone répond quand l'animal est orienté vers l'ouest. **D** - Un enregistrement d'activité de cellules de grille selon la même méthode de représentation que pour la cellule de lieu (A) [Grieves & Jeffery 2016]

3.2.3.1 Les cellules de lieux

Les cellules de lieux sont un type de neurones découvert, chez le rat, dans l'hippocampe, par O'Keefe et Dostrovsky en 1971 [O'Keefe & Dostrovsky 1971]. Elles ont été depuis identifiées chez d'autres rongeurs [Las & Ulanovsky 2014] ainsi que chez les primates, dont l'être humain [Hazama & Tamura 2019].

Les cellules de lieux sont des neurones qui encodent des portions de l'espace que l'on appelle des lieux. L'activité d'une cellule de lieu est maximale à une position spécifique dans l'environnement et décroît progressivement lorsque l'animal s'en écarte. La zone de l'environnement pour laquelle une cellule décharge (émet des impulsions) s'appelle un champ de lieu (voir figure 3.10).

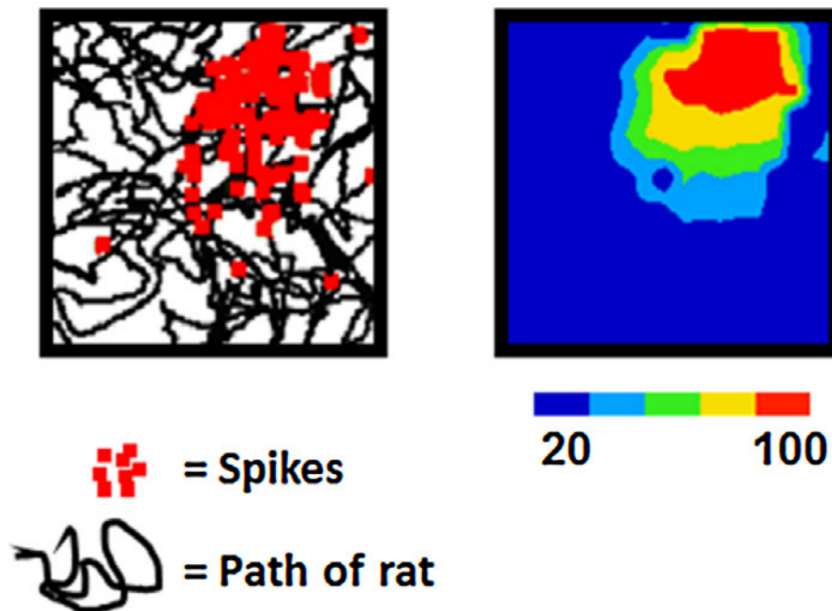


FIGURE 3.10 – Exemple d'activité d'une cellule de lieu. La zone de plus forte activité sur l'image de droite (en rouge) correspond au champ de lieu [Grieves & Jeffery 2016]

Il y a une grande variation dans la taille des champs de lieux. Les plus petits sont de l'ordre d'une dizaine de centimètres. Les plus grands observés sont de l'ordre de la dizaine de mètres [Jung *et al.* 1995].

Les cellules de lieux sont arrangées en fonction de la taille de leur champ de lieu [Kjelstrup *et al.* 2008]. Au niveau de la zone dorsale de l'hippocampe, on trouve des cellules de lieu avec des champs de lieux de petites tailles. Plus on se rapproche de la zone ventrale plus la taille des champs de lieux augmente. Cependant, il ne semble pas y avoir de topologie pour les cellules de lieu.

L'activité des cellules de lieux est fortement affectée par la modalité visuelle. Des expériences [Dudchenko *et al.* 1997] où, dans un environnement contrôlé, on

déplace des amers après apprentissage, permettent d'observer que les champs de lieux subissent un déplacement similaire indiquant ainsi le rôle fondamental des informations visuelles.

Dans des expériences [Smith *et al.* 2009][Jacob *et al.* 2014][Wiener *et al.* 1995] où l'animal est maintenu dans le noir après avoir appris des lieux on observe un maintien, temporaire, de l'activité des champs de lieux indiquant qu'en plus d'informations visuelles, les cellules de lieux utilisent des informations odométriques.

Dans le cas des rongeurs on observe aussi que l'olfaction [Zhang & Manahan-Vaughan 2013] et le toucher, par l'intermédiaire de leurs vibrisses [Gener *et al.* 2013], servent de stimuli importants pour la génération des réponses des cellules de lieux.

D'autres expériences venant à modifier l'environnement, que ce soit sa taille [Muller & Kubie 1987], sa couleur [Jeffery *et al.* 2004] ou bien son odeur [Anderson & Jeffery 2003] ont permis de mettre en lumière la capacité de recodage des cellules de lieux. Ces dernières encodent un lieu dans un environnement donné mais vont encoder un autre lieu, ou ne pas avoir d'activité significative dans un autre lieu.

Certains ont émis l'hypothèse que des cellules de lieux effectueraient de l'intégration des informations afin de faire de la correction d'erreur selon un processus semblable à de l'intégration bayésienne [Madl *et al.* 2014].

La mesure des activités de cellules de lieux hippocampiques a montré que ces activités ont tendance à suivre une certaine rythmicité commune que l'on appelle rythme theta [Bose & Recce 2001]. Ce signal serait en lien avec le processus d'encodage des lieux. En effet, il a été observé que l'activité des cellules est synchronisée sur ce rythme en temps normal, mais que, lorsque l'animal entre dans le champ de lieu d'une cellule, cette dernière voit son activité se désynchroniser jusqu'à être en anti-phase au centre du champ de lieu. Cette activité se resynchronise lorsque l'on quitte le lieu.

On a observé que, chez différentes espèces de mammifères, les cellules de lieux avaient des champs de lieux de formes différentes. Chez le rat, les champs de lieux sont omnidirectionnels alors que chez les primates les champs de lieux sont orientés, elles portent le nom de cellules de vue [Georges-François *et al.* 1999] et encodent les lieux que le singe regarde. Une série d'expériences [Gaussier *et al.* 2002a] semble montrer que la forme des champs de lieux dépend de l'implémentation des yeux sur la tête. Les primates ont des yeux proches pour la vision binoculaire qui ne couvrent qu'une portion restreinte de l'espace. Les rongeurs, au contraire, ont des yeux très écartés de part et d'autre de la tête qui couvrent une large portion du champ visuel d'où les champs de lieux omnidirectionnels.

3.2.3.2 Les cellules de direction de la tête

Les cellules de direction de la tête sont des neurones initialement découverts dans le presubiculum dorsal (aussi appelé postsubiculum) [Taube *et al.* 1990][Taube & Bassett 2003]. Depuis, ces neurones ont été trou-

vés dans un certain nombre de parties du cerveau. Ils ont été identifiés dans le striatum, les thalamus antérodorsal et dorsal latéral, les cortex entorhinal et rétrospécial et enfin le noyau mamillaire latéral.

Ce sont des cellules qui encodent des directions précises de la tête. Elles ont une activité qui est maximale selon une direction précise. Elles sont codées selon un référentiel absolu, à l'instar d'une boussole magnétique, mais propre à l'animal.

L'activité d'une cellule de direction de la tête se présente sous la forme d'un pic selon une orientation précise et une activité qui décroît de part et d'autre de ce pic.

Certaines cellules de direction de la tête sont conjonctives : elles ont des réponses qui sont aussi modulées par des informations de vitesse angulaire. Si l'animal tourne vite sa tête dans la direction préférentielle d'une de ces cellules, sa réponse sera plus forte que si l'animal tourne lentement.

Diverses expériences ont permis d'identifier les informations participant à la construction des activités des cellules de direction de la tête.

Un premier groupe d'expériences concerne le sens de la vision qui se révèle prépondérant dans la génération des réponses de cellules de direction de la tête.

Dans des expériences où on applique une rotation à un amer ayant servi à apprendre une direction on observe que la réponse de la cellule de direction de la tête se décale selon le même angle que l'amer (voir figure 3.11) indiquant que la vision est utilisée comme stimulus par les cellules de direction de la tête.

Certaines expériences ont mis en lumière que les amers en arrière plan étaient prépondérant sur ceux en premier plan.

Dans des expériences où on fait apprendre plusieurs directions à partir d'amers et que l'on enlève un de ces amers, on obtient un décalage, du même ordre de grandeur, des directions préférentielles pour l'ensemble des neurones de direction de la tête. Cela indique que les cellules de direction de la tête, dans certaines zones, mémorisent leur direction préférentielle relativement aux autres cellules et qu'elles fonctionnent en réseau.

Des expériences avec privation du sens de la vision [Brown *et al.* 2003] montrent que les cellules de direction de la tête sont capables de maintenir leur activité pendant un certain temps malgré un phénomène de dérive. Cela indique que les neurones de direction de la tête utilisent, en plus de la vision, des informations de proprioception pour faire de l'intégration de chemin afin de maintenir leur direction préférentielle.

Pour finir avec la vision, dans des expériences où les informations visuelles viennent contredire les informations proprioceptives [Taube & Burton 1995], la vision l'emporte.

Les autres expériences concernent les informations motrices et vestibulaires qui participent à la proprioception. Des expériences [Stackman Jr *et al.* 2003] où l'animal est soit transporté d'un environnement à un autre ou se déplace seul d'un environnement à un autre mettent en évidence l'importance des signaux moteurs. Ces signaux permettent aux cellules de direction de la tête de maintenir leur activité dans un nouvel environnement, ce qu'elles n'arrivent pas à faire quand l'animal est transporté.

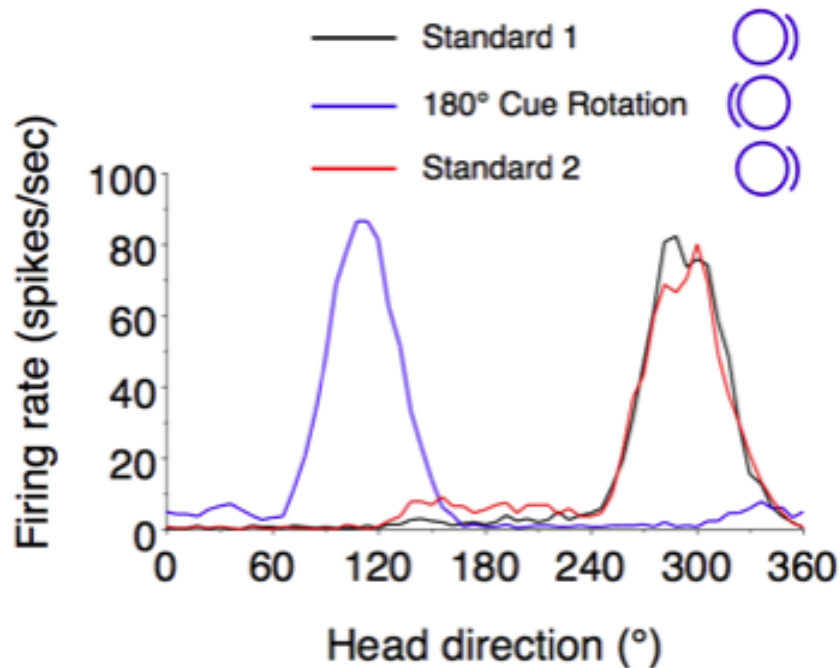


FIGURE 3.11 – Activité d'une cellule de direction de la tête pour une condition initiale (Standard 1), une rotation d'amer de 180 degrés (180 degree Cue Rotation) et enfin un retour aux conditions initiales (Standard 2) [Taube & Burton 1995].

Chez le rat, l'olfaction est un sens qui participe grandement à former la réponse des cellules de direction de la tête.

Les observations ont montré que quand l'animal restait immobile, les activités des cellules se maintenaient sans stimulus. L'hypothèse est que les cellules de direction de la tête possèdent un mécanisme de réseau d'attracteurs circulaire pour maintenir leur activité.

3.2.3.3 Les cellules de grille

Les cellules de grille sont un type de neurone, jouant un rôle dans la perception de l'espace, découvert en 2005 dans le cortex entorhinal [Hafting *et al.* 2005]. L'activité d'une de ces cellules définit un motif triangulaire qui se répète de manière périodique et recouvre l'ensemble de la surface de l'environnement (voir figure 3.12). Leur fonction est encore mal connue, elles pourraient jouer un rôle dans la formation des cellules de lieux [Solstad *et al.* 2006] ou bien être impliquées dans le processus d'intégration d'informations multimodales.

Les observations dans le cortex entorhinal montrent qu'il existe des cellules de grille avec des motifs de différentes tailles, des orientations différentes et des décalages dans leur champ d'activation.

Une cellule de grille est donc définie par un décalage, une orientation et une

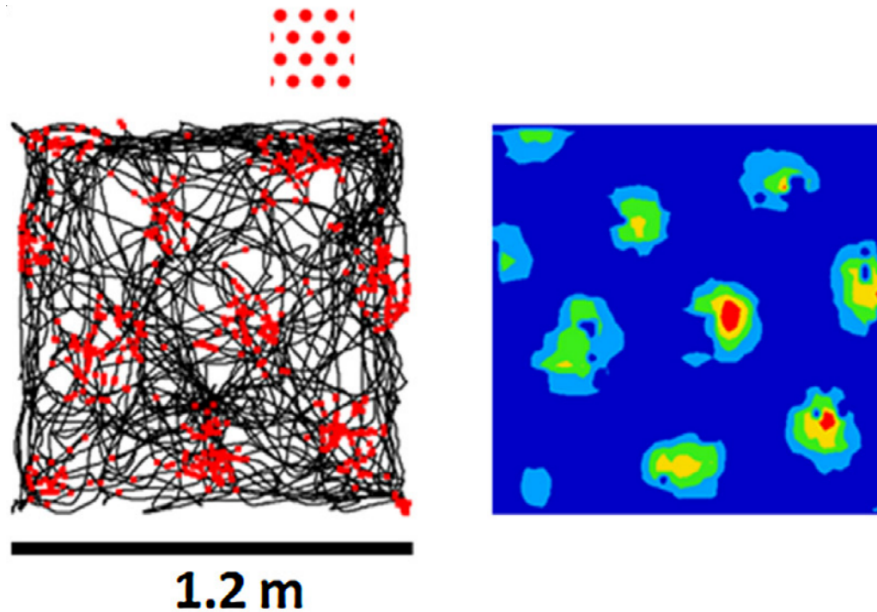


FIGURE 3.12 – Enregistrement d’une cellule de grille dans le cortex entorhinal. On peut voir le motif défini par cette cellule [Grieves & Jeffery 2016].

période spatiale. Les observations semblent indiquer que des cellules voisines ont une période spatiale similaire qui augmente quand l’on va de la portion dorsomédiane à la zone ventrolatérale du cortex entorhinal [Brun *et al.* 2008]. Il semblerait y avoir aussi une organisation selon l’orientation des cellules de grille. Les cellules de grille dans les cortex entorhinal médian droit et gauche auraient des orientations différentes [Fyhn *et al.* 2007]. Pour finir, il ne semble pas exister de distribution particulière des cellules de grille en fonction de leur décalage.

Les cellules de grille semblent utiliser des informations visuelles afin de conserver leur orientation comme l’atteste des expériences en milieu contrôlé où la rotation d’un amer entraîne un changement d’orientation similaire des cellules de grille. Elles utilisent aussi des informations proprioceptives car elles arrivent à maintenir, temporairement, leur activité dans les cas de privation du sens de la vision [Hafting *et al.* 2005]. Leur rôle reste encore très mal connu. Selon certaines hypothèses, les cellules de grille participeraient au processus de mémorisation en encodant les informations.

3.2.3.4 Cellules de vitesse

Les cellules de vitesse (speed cells) sont des neurones présents dans le cortex entorhinal [Kropff *et al.* 2015]. Comme leur nom l’indique, ces cellules sont sensibles à la vitesse propre à l’animal. Ce sont des neurones à réponse impulsionnelle dont la fréquence d’activation est corrélée à la vitesse de course de l’animal.

Ces différentes cellules aux caractéristiques variées sont utilisées par les neuro-

boticiens comme autant de briques élémentaires leur permettant de construire des modèles computationnels neuromimétique dont certains exemples vont être présentés par la suite.

3.3 Modèles bio-inspirés appliqués à la navigation de véhicules autonomes

Comme expliqué plus tôt, la robotique est, pour les neurosciences, un outil pour faire de la modélisation et valider ou infirmer des hypothèses. Pour les roboticiens qui s'intéressent aux neurosciences, ces dernières apportent des pistes pour créer des modèles robotiques dans des domaines divers : apprentissage, navigation, contrôle moteur. Cependant, dans le cadre auquel se restreint ce manuscrit, de localisation et de navigation bio-inspirées, il y a encore peu de modèles appliqués à la conduite autonome de véhicule et il existe encore moins de modèles neurorobotiques.

En plus des modèles neurorobotiques issus du laboratoire ETIS qui seront, en partie, présentés par la suite, cette section donne un état de l'art de modèles bio-inspirés ou neurorobotiques testés dans un contexte de conduite de véhicule ou, au moins, sur robot.

3.3.1 Les réseaux de neurones profonds

Les réseaux de neurones profonds (deep neural networks) sont un incontournable de la robotique bio-inspirée, et de la robotique tout court d'ailleurs. Ces réseaux ont permis de faire des progrès significatifs dans le traitement des images, la reconnaissance d'objets et le contrôle.

On ne peut guère les considérer comme neurorobotiques. Ils utilisent, certes, des couches neuronales d'inspiration biologique, c'est le cas des CNN, les réseaux neuronaux de convolution, qui sont similaires à certaines couches de neurones observées dans le cortex visuel du chat [Hubel & Wiesel 1962b]. Cependant la succession des couches et des types de couches ne correspondent généralement pas à des observations effectuées par les neurosciences mais plutôt à des retours d'expérimentations empiriques.

En plus de leur utilisation par des approches de bout-en-bout présentées au chapitre 2, les réseaux de neurones profonds sont intensivement utilisés pour faire du traitement visuel. Ils sont très efficaces pour reconnaître des objets dans une scène. Le modèle YOLO V3 (You Only Look Once [Redmon & Farhadi 2018]) en est un récent exemple. Ce réseau composé de 106 couches de convolutions est capable de détecter en temps réel 80 classes d'objets à des échelles de taille différentes.

Ces réseaux profonds sont parfois utilisés pour générer un encodage des images. Ce code sert ensuite à faire de la reconnaissance de lieux visuelle, notamment pour la navigation des véhicules.

Un exemple d'un de ces modèles est celui de [Chen *et al.* 2014]. Ce modèle utilise OVERFeat [Sermanet *et al.* 2014], un réseau profond conçu initialement pour la

détection d'objet, pour encoder les images sous forme d'un vecteur de features. Il utilise les vecteurs issus d'une base d'entraînement et d'une base de test afin de construire une matrice de confusion associant chaque vecteur de test avec un vecteur d'entraînement par minimisation de la distance euclidienne entre les caractéristiques extraites pour chaque couche de la partie neuronale. Le modèle utilise deux filtres appliqués sur la matrice de confusion pour s'assurer de la continuité spatiale des réponses (voir figure 3.13). Le premier filtre s'assure que deux réponses consécutives du modèle ont, dans la matrice de confusion, des indices proches, signe de continuité spatiale. Le second filtre part de l'hypothèse que, dans la matrice de confusion l'ensemble des réponses suit une fonction linéaire de référence. Le filtre prend, en plus de la réponse courante, une partie des réponses précédentes. Il effectue une régression linéaire de ces réponses et vérifie que cette régression correspond, avec une tolérance pour le coefficient directeur, à la fonction linéaire de référence. Ceci permet de s'assurer de la continuité spatiale.

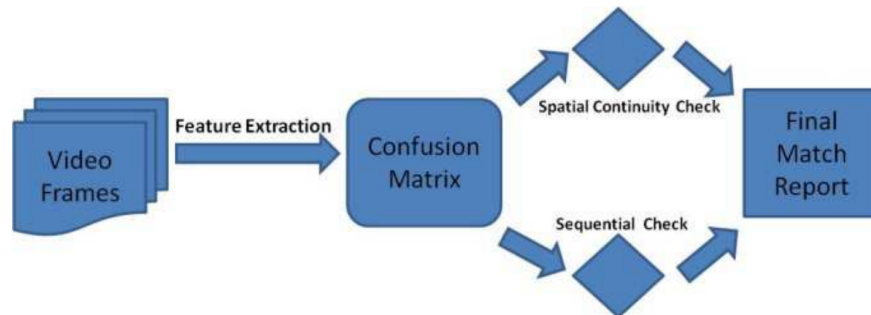


FIGURE 3.13 – Schéma de fonctionnement du modèle. Il extrait un vecteur de features à partir d'images d'entrée avant d'utiliser la matrice de confusion et deux filtrages avant de faire la correspondance finale [Chen *et al.* 2014].

Ce modèle a été testé sur les bases de données de Eynsham et de QUT et comparé aux modèles FAB-MAP [Cummins & Newman 2009] et SeqSLAM [Milford & Wyeth 2012]. Il obtient des performances meilleures en termes de précision et rappel que les modèles auxquels il se compare. Cela montre l'efficacité du codage obtenu par un réseau profond initialement entraîné pour une autre tâche. Les auteurs expliquent que, selon les environnements, certaines couches neurales donnent un codage plus pertinent que d'autres. Il n'existe cependant pas, comme ils le font remarquer, de mécanisme permettant de sélectionner la couche pertinente à la volée.

Un problème commun à de nombreuses approches reposant sur des apprentissages supervisés est la nécessité de posséder une grosse base de données labellisées pour procéder à l'entraînement du réseau. De plus, la phase d'apprentissage est généralement longue. Un problème plus général des réseaux de neurones profonds est l'aspect boîte noire de ces réseaux. Les réponses de ces réseaux sont complexes à analyser (la somme linéaire d'une somme linéaire d'une somme linéaire ...) et leur

complexité augmente pour chaque couche ajoutée. Cette difficulté d'analyse fait des réseaux de neurones profonds, pour l'instant, un outil d'ingénierie pour obtenir une solution à un problème donné, mais peu adapté pour comprendre la solution obtenue.

3.3.2 RatSLAM

RatSLAM [Milford *et al.* 2004] est un modèle neurorobotique qui se base sur la structure du cerveau des rongeurs. Il est composé d'une carte d'expérience et de ce qu'il nomme des cellules de pose et des cellules de vue (voir figure 3.14). Son aspect neurorobotique vient de son utilisation de cellules de lieux visuelles et de cellules de direction de la tête.

Les cellules de pose sont modélisées par un réseau d'attracteurs continus en trois dimensions. Les trois dimensions représentent les coordonnées x , y et l'orientation θ . Pour chaque dimension, il existe des connexions entre les bords opposés qui effectuent une forme de modulo. Lorsque le robot est dans une pose (x, y, θ) le neurone de pose correspondant à ces coordonnées va s'activer.

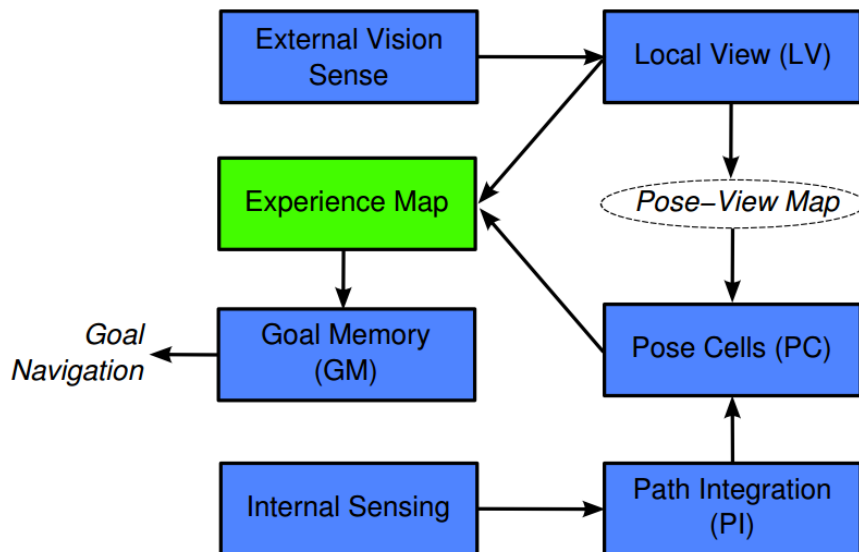


FIGURE 3.14 – Le schéma de fonctionnement du modèle RatSLAM. Des cellules de vue sont obtenues à partir de senseurs externes (en général une caméra). La proprioception, par intégration de chemin, sert à obtenir une cellule de pose. Ces deux types de cellules servent à construire une carte d'expérience utilisable pour des tâches de navigation [Milford & Wyeth 2010].

Les cellules de vue locale mémorisent des prises de vue de l'environnement encodées. Lorsque le système visuel reconnaît une de ces prises de vue, la cellule de vue locale associée est stimulée. Lorsqu'une cellule de pose et une cellule de vue

locale sont actives simultanément, une association est apprise ce qui permet par la suite à la cellule de vue locale d’entraîner l’activation de la cellule de pose et donc de permettre une recalibration. La recalibration de l’intégration de chemin est une méthode qui consiste à exploiter une autre source d’information reposant sur une modalité différente (souvent la vision) afin d’estimer le niveau d’erreur de positionnement de l’intégration de chemin, la dérive, et de la corriger.

La carte d’expérience est une carte topologique semi-métrique qui mémorise des expériences et les transitions entre les expériences . Une expérience est définie comme une association d’une cellule de vue locale et d’une cellule de pose. Les transitions sont les changements de position entre deux expériences. La première expérience est apprise au démarrage. Quand le système se déplace et que les cellules de pose et de vue locale changent suffisamment une nouvelle expérience est apprise. Le déplacement entre l’ancienne et la nouvelle position est mémorisé dans la transition. Cette carte est similaire à des représentations que l’on retrouve dans certains algorithmes de SLAM [Thrun & Montemerlo 2006].

Le modèle a été testé dans quantité d’expériences différentes ([Milford & Wyeth 2010], [Silveira *et al.* 2013], [Steckel & Peremans 2013], [He *et al.* 2015]). Les résultats présentés au prochain chapitre pour faire une comparaison entre les approches ne concernent que les résultats obtenus sur un trajet de soixante six kilomètres effectué dans la banlieue de Sainte Lucie à Brisbane, en Australie (voir figure 3.15). Le test a été mené avec une caméra d’ordinateur portable fixée sur un toit de voiture [Milford & Wyeth 2008]. Les images captées par la caméra sont découpées en trois parties. L’une sert pour la caractérisation des lieux, les deux autres servent à calculer la vitesse et l’orientation du véhicule par flux optique, une méthode consistant à faire une estimation du déplacement entre deux images successives.

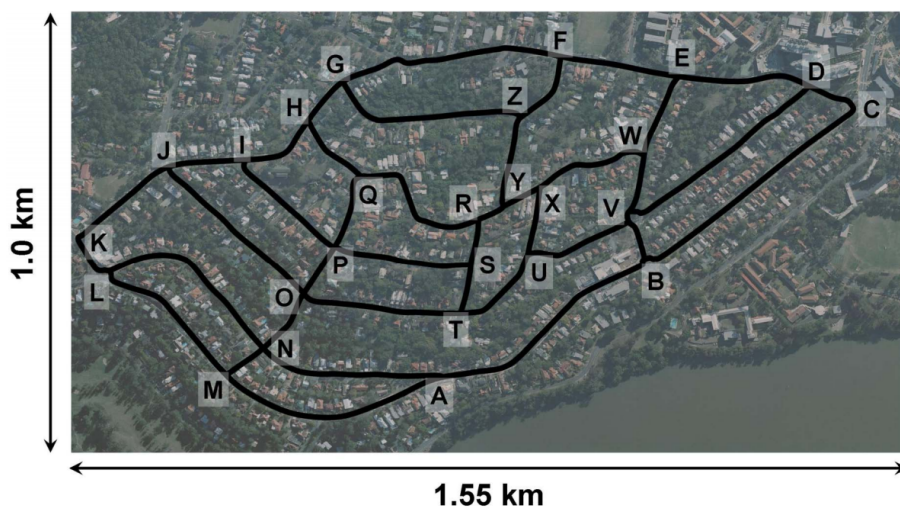


FIGURE 3.15 – La banlieue de Sainte Lucie où a été testée le modèle RatSLAM [Milford & Wyeth 2008].

Ce modèle est l'un des rares modèles neurorobotiques testés en temps réel sur véhicule. Les différentes expériences montrent la capacité à passer à des échelles plus grandes et l'adaptabilité du système à différents types d'environnements. Le modèle est capable de détecter des fermetures de boucles et de reconnaître des lieux avec un taux de rappel de l'ordre de 80 % pour l'expérience de Brisbane. La carte d'expérience obtient un écart moyen avec la vérité terrain de l'ordre de 1,2 mètres pour un trajet de 66 kilomètres. L'odométrie basée sur flux optique permet d'obtenir de bons résultats pour le calcul de l'orientation. En revanche pour le calcul de vitesse linéaire les résultats sont moins précis avec des surestimations pouvant atteindre un facteur de 1,6.

L'un des défauts de ce modèle est que les cellules de vue locales peuvent être sujettes à des problèmes de repliement perceptif : deux lieux distincts de l'environnement peuvent être perçus par le système comme étant un seul et même lieu. Un autre défaut est le fait que la zone couverte par les cellules de vue locales est une taille fixée par un paramètre durant l'expérience. Ce système ne permet pas d'assurer que chaque portion de l'environnement sera couverte par une cellule de vue locale. De plus, ce mécanisme ne permet pas d'optimiser le compromis entre le nombre de cellules recrutées et la précision requise en fonction des environnements traversés. Par exemple, sur autoroute, les vitesses étant plus grandes et les bifurcations moins nombreuses qu'en centre ville, une faible précision de la représentation est suffisante pour naviguer. Un autre problème est le temps nécessaire pour stabiliser la carte d'expérience (de l'ordre de 1700 secondes pour l'expérience présentée) afin de la rendre exploitable. D'un point de vue des neurosciences, ce modèle repose sur des hypothèses (structure du réseau d'attracteurs continus) biologiquement peu plausibles et le lien entre les différentes cartes exploitées et les structures modélisées n'est pas explicite.

3.3.3 Modèle de mémoire visuelle

Le modèle de mémoire visuelle [Rebai *et al.* 2012], [Azouaoui & Achour 2014] s'inspire de cellules de vue, des cellules de lieux visuelles orientées découvertes chez le primate. Il est composé d'un système visuel et d'une mémoire (voir figure 3.16). Le système visuel utilise un mécanisme attentionnel qui ne récupère l'image que dans quatre fenêtres situées au dessus de la ligne d'horizon. Ces quatre portions d'image sont transformées au format HSV (Hue Saturation Value). Les histogrammes de saturation et de teinte (hue) sont calculés pour les quatre portions d'images extraites et concaténées afin de former un vecteur de 128 éléments qui constitue la cellule de vue.

La mémoire est composée d'un système neural et d'un système d'orientation. Le système neural est un réseau fuzzy ART [G. Carpenter & Rosen 1991] constitué de deux couches neuronales. Un réseau ART (Adaptive Resonance Theory) est une structure neuronale permettant d'apprendre des catégories et de faire de la classification à partir d'un motif d'entrée. Un réseau fuzzy ART est un réseau ART utilisant de la logique floue. La première couche du réseau fuzzy ART est l'histo-

gramme normalisé de l'image courante. La deuxième couche correspond aux cellules de vue précédemment mémorisées. Le système propose une cellule de vue gagnante correspondant à l'histogramme courant. Le système d'orientation utilise un seuil de vigilance pour valider ou invalider la correspondance. Dans le deuxième cas une nouvelle cellule de vue est recrutée.

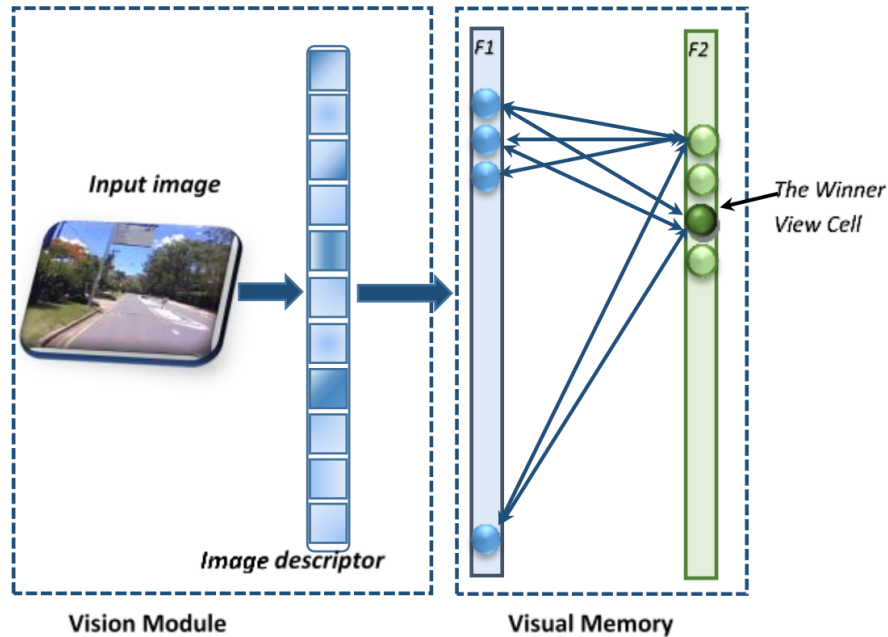


FIGURE 3.16 – Schéma du modèle de mémoire visuelle [Azouaoui & Achour 2014]. Le système visuel extrait un descripteur à partir d'une image d'entrée. Le système de mémoire visuelle compare le descripteur à ceux qu'il a en mémoire et apprend un nouveau lieu si il ne parvient pas à obtenir une correspondance [Azouaoui & Achour 2014]

Ce modèle a été testé hors ligne à partir d'un enregistrement vidéo pris sur un véhicule issu de l'expérience menée dans la banlieue de Sainte Lucie. Les résultats montrent que ce modèle est capable de détecter des fermetures de boucle même si le taux de précision obtenu est faible (40 %) du fait des fausses reconnaissances. Les expériences, faites avec différentes valeurs de vigilance, montrent que cette dernière a un effet significatif sur la précision, le rappel et l'utilisation de la mémoire et qu'il y a donc un compromis à trouver pour maximiser au mieux ces différents critères.

Ce modèle en utilisant la teinte et la saturation est robuste aux changements de luminosité. Il utilise des traitements visuels simples afin d'obtenir ses descripteurs. Le système de vigilance permet un apprentissage en ligne permettant de s'adapter à de nouveaux environnements.

Aucune information n'est donnée sur le coût computationnel. Il est donc impossible de dire si ce modèle peut fonctionner en temps réel. Ce dernier possède un nombre d'erreurs de reconnaissance significatif limitant la fiabilité de la repré-

TABLE 3.1 – Performances du modèle selon la valeur du seuil de vigilance

Vigilance	Nombre de cellules de vue	Précision	Rappel
0.98	567	100%	7.5%
0.95	236	73.3%	61.11%
0.8	34	40%	83.33%

sensation. Par ailleurs le système attentionnel est assez rudimentaire et empirique : les zones d'intérêt sont définies arbitrairement et sont fixes. Par conséquent, dans certains cas, l'information permettant de définir un lieu peut se retrouver en dehors de la zone d'intérêt définie par le modèle.

3.3.4 Modèle de carte cognitive de Zeng et Si

Ce modèle neurorobotique [Zeng & Si 2017] se base sur l'utilisation de cellules de grille et de cellules de direction de la tête afin de faire de la reconnaissance de lieux visuelle en environnement urbain. Le modèle se base sur des informations visuelles et utilise le flux optique pour calculer les vitesses linéaires et angulaires. Le système se décompose en quatre parties.

Le premier sous-système est celui qui modélise des cellules de direction de la tête encodant des orientations et des vitesses angulaires. Il est constitué d'un réseau de neurones formant un anneau attracteur. Chaque neurone encode une orientation et une vitesse angulaire. L'orientation est obtenue par intégration de la vitesse angulaire projetée sous un sous-ensemble du réseau. Le système utilise une transformation de Fourier pour convertir l'état du réseau neuronal en orientation et vitesse angulaire du robot.

Le second module de ce modèle est le réseau formant les cellules de grille. Il est alimenté par les estimations de vitesse linéaire et les informations des cellules de direction de la tête. Les cellules de grille conjonctives, des cellules de grille qui, en plus, ne réagissent que pour une orientation précise, forment un réseau sous forme d'un tore constitué d'attracteurs. Cette forme torique est due à la projection de la position qui est exprimée sous forme d'un modulo. Ce réseau représente la position et la vitesse linéaire qui sont chacune encodées sous forme de deux coordonnées dans le réseau neuronal. La phase du motif de grille encode les positions et son évolution sur la grille correspond à la vitesse.

Les deux systèmes précédents reposent sur des estimations de vitesse linéaire et angulaire. Il est donc nécessaire d'avoir un mécanisme de recalibration pour s'assurer de la fiabilité des estimations. Ce système de recalibration extrait des vues locales depuis une caméra. Ces vues sont comparées à celles précédemment mémorisées par le système. Une vue nouvelle déclenche l'apprentissage d'une cellule de vue locale. Chaque cellule de vue mémorisée est associée avec les configurations du réseau des cellules de grille et du réseau des cellules de direction de la tête au moment de l'apprentissage de la vue. Par conséquent, quand une vue est reconnue, la cellule

de grille et la cellule de direction de la tête associées sont activées permettant la recalibration.

Le dernier module est le système construisant la carte. Il génère une carte d'expérience similaire à celles créées par RatSLAM [Milford & Wyeth 2008]. La carte est générée à partir des activités des cellules de grille. Un algorithme de relaxation de graphe [Duckett *et al.* 2002] utilisant un mécanisme de fermeture de boucle est appliqué sur cette carte afin de la corriger.

Le modèle a été testé sur la base de données de Sainte Lucie utilisée par RatSLAM et les auteurs montrent qu'ils obtiennent des réponses qualitativement correctes par l'intermédiaire de la carte d'expérience mais aucune mesure quantitative n'a été fournie. De plus aucune information n'est fournie sur le seuil servant à différencier une vue nouvelle de celles déjà mémorisées. La capacité apparente du modèle à cartographier un environnement de 66 kilomètres semble montrer qu'il fonctionne sur de grands environnements. Sa structure respecte l'organisation du cortex entorhinal observé chez le rat bien qu'il n'y ait pas de modélisation de l'hippocampe ni des cellules de lieux qui semblent centrales pour faire de la localisation chez les animaux. De plus, la projection sous une forme torique semble a priori peu plausible biologiquement.

3.3.5 Beobot

Ce modèle [Christian *et al.* 2010] se concentre sur la localisation d'un robot au milieu d'environnements remplis de piétons en se basant principalement sur la vision monoculaire.

Il se compose d'un système de localisation et d'un système de navigation. Le modèle de navigation sort du cadre de ce manuscrit se concentrant sur la reconnaissance de lieux par la vision et ne sera donc pas décrit. Le système de localisation se base sur une représentation hiérarchique hybride : une carte topologique globale et une carte locale sous forme de grille d'occupation que l'on retrouve souvent dans les algorithmes de SLAM (voir figure 3.17).

Le système de localisation prend en entrée des images dont il extrait des informations d'orientation, de couleur et d'intensité. Ensuite les informations sont analysées selon deux chaînes de traitement en parallèle. La première utilise GIST [Oliva & Torralba 2001] pour obtenir un vecteur de seize éléments pour caractériser l'image courante. Ce traitement est le plus rapide. La seconde chaîne utilise une extraction de points saillants pour obtenir un ensemble d'amers définissant l'image. Ces amers subissent deux transformations, l'une utilisant la méthode SIFT [Lowe 2004b] et l'autre par un vecteur d'éléments saillants afin d'établir des correspondances. Les vecteurs d'éléments saillants sont ensuite comparés à la base de données des amers selon une distance euclidienne et une distance dans l'espace des vecteurs.

Le vecteur résultant du GIST et la carte de saillance sont utilisés en conjonction d'informations odométriques pour alimenter un algorithme de MonteCarlo [Fox *et al.* 1999]. La carte topologique est divisée en plusieurs segments. Les informations issues du GIST permettent d'identifier le segment courant. La localisation

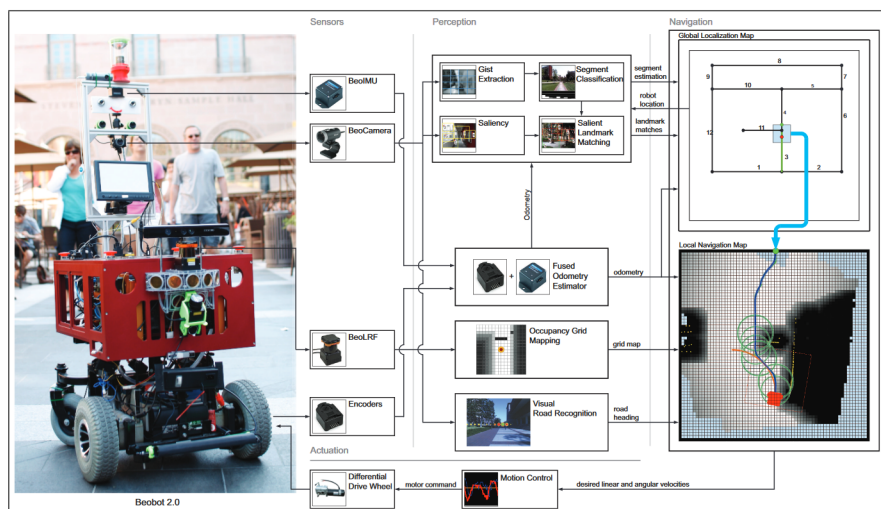


FIGURE 3.17 – Le robot Beobot et le schéma de son modèle de localisation et de navigation. Il utilise toute une série de capteurs afin de se localiser sur une carte en utilisant une stratégie d'extraction d'amers, une stratégie de description globale d'image et de l'intégration de chemin. Cette localisation est utilisée par plusieurs stratégies de navigation, voir [Christian *et al.* 2010] pour de plus amples détails.

précise se fait en classant les positions de la base de données de la plus plausible à la plus improbable en se servant du segment courant et de la position hypothétique selon la proprioception. Quand la mise en correspondance des amers est terminée, la localisation métrique est calculée. Comme l'étape de mise en correspondance des amers est longue (3 secondes en moyenne) et que le robot continue de se déplacer, il y a un écart entre la position calculée à partir de la perception et la position courante. Cet écart est corrigé par une intégration de chemin.

L'information de position obtenue est utilisée par l'algorithme de Monte Carlo pour estimer le mouvement du robot sur la carte topologique. Ce dernier fonctionne en temps réel car il n'utilise que 100 particules et travaille sur une carte topologique unidimensionnelle. Ces informations de localisation sont ensuite utilisées par un système de navigation qui sort du champ d'étude de ce manuscrit.

Ce modèle est bio-inspiré, pour son système de localisation, du fait de l'organisation de ces fonctions. Les premiers traitements visuels extrayant les orientations les intensités et les couleurs correspondent à des modèles biologiques, plus vraiment à jour, de la rétine, du nerf optique et du cortex visuel primaire. Les deux chaînes de traitement correspondent aux visions périphérique, rapide, et fovéale, plus lente.

Ce modèle a été testé lors de plusieurs expériences sur le campus de l'université de Californie sud où le robot a été capable de se localiser sur plusieurs trajets avec et sans foule piétonne. Sur les trois trajets, labellisés LIBR, PROF et ATHL, le système obtient un écart moyen de positionnement par rapport à la vérité terrain de l'ordre de $0,97\text{ m}$ (voir figure 3.18).

Les points forts de ce modèle sont sa redondance dans ses sources de percep-

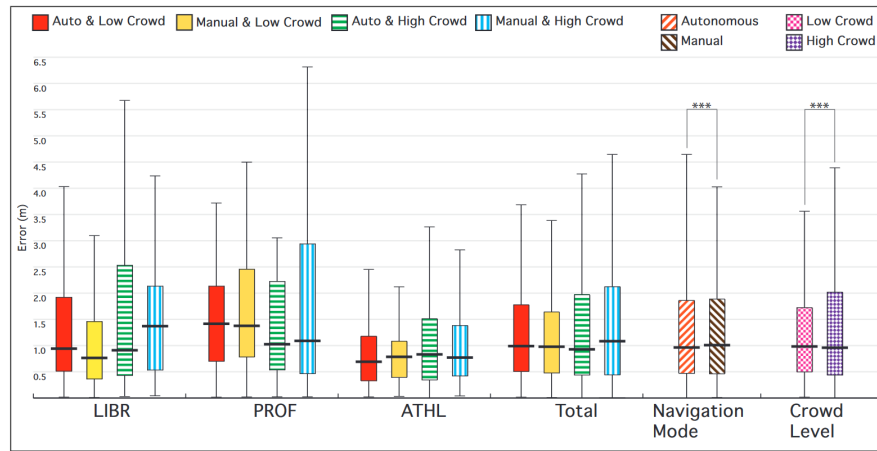


FIGURE 3.18 – Les erreurs de positionnements pour plusieurs trajets (LIBR, PROF et ATHL) selon que le modèle soit en mode automatique ou contrôlé par un utilisateur et en fonction de la présence ou non d’une foule sur le trajet [Christian *et al.* 2010].

tion (caméra, lidar pour l’évitement d’obstacle, centrale inertielle, roue codeuse) et ses stratégies de localisation et de navigation qui rendent le modèle robuste aux défaillances. Son fonctionnement asynchrone, en plus de la robustesse aux défaillances, permet d’avoir des performances de localisation compatibles avec le fonctionnement en temps réel. Enfin son système visuel de détection d’amers est en partie robuste aux changements de luminosité. Son point faible principal vient de sa carte topologique, construite à la main, qui rend le modèle peu adaptable à la nouveauté. Notons que ces bonnes performances de reconnaissance visuelle nécessitent de faire un trajet d’apprentissage plusieurs fois à des horaires différents afin de mémoriser les amers visuels.

3.4 Discussion

Les animaux fournissent des exemples de systèmes de navigation capables de résoudre le problème de la navigation des véhicules autonomes. Ces modèles biologiques restent encore mal connus malgré les travaux des neurosciences qui restent lacunaires dans leur compréhension de ces systèmes du fait de leur grande complexité et de dispositifs de mesures limités. Ces équipements ont, néanmoins, permis d’identifier certains éléments des mécanismes de localisation biologiques, que ce soit des neurones spécialisés ou bien des structures cérébrales ainsi que les interactions liant ces structures.

A partir de ces quelques briques élémentaires et d’hypothèses formulées par les chercheurs des neurosciences, les roboticiens ont imaginé et implémenté des modèles de localisation et de navigation dont certains fonctionnent sur robot et sur véhicule. Ces modèles sont capables de résoudre partiellement des problèmes de localisation

et de navigation.

Les algorithmes bio-inspirés ont l'avantage de partir d'un modèle de navigation fonctionnel, les animaux. Cette approche permet d'avoir tout un ensemble de pistes à explorer pour résoudre les différents challenges que pose la navigation autonome. Par ailleurs, par un dialogue avec les neurosciences, peut être mis en place un cercle vertueux pour résoudre ce problème. Les roboticiens testent des modèles de neurosciences permettant de les valider ou invalider en en donnant les limites. Ceci permet aux chercheurs en neurosciences d'élaborer de nouvelles expériences, d'en tirer de nouvelles observations et d'améliorer leurs modèles, offrant de nouvelles pistes à explorer aux roboticiens. La difficulté majeure de cette approche est de parvenir à rassembler les différents éléments de solutions existants, disséminés dans l'ensemble des travaux du vaste domaine des neurosciences, afin de concevoir un modèle cohérent à tester.

Le plus gros frein à l'utilisation des modèles bio-inspirés est d'ordre conceptuel. Dans les approches dites classiques, la précision métrique est primordiale afin de pouvoir effectuer des tâches de navigation. Par conséquent, les modèles qui découlent de ces approches visent à être les plus précis possible. L'objectif est d'obtenir une précision de l'ordre de la vingtaine de centimètres afin de naviguer en ville. De ce point de vue, les modèles bio-inspirés, en apparence peu performants selon ce critère de précision, semblent être à exclure. En réalité, on ne connaît pas la précision de localisation de ces modèles. Les études en neurosciences n'évaluent que de façon indirecte les systèmes de localisation des animaux par le biais du comportement spatial observé conduisant ou non au succès d'une tâche de navigation. Un exemple classique est le fait d'atteindre une zone où se trouve une récompense. Une fois dans cette zone, une grande précision n'est pas requise car la récompense est directement visible. La suite de ce manuscrit va présenter un modèle de navigation réactif neurorobotique reposant sur la reconnaissance de lieux par la vision et ne nécessitant pas une grande précision afin de naviguer.

Modèle de cellules de lieu visuelles

Sommaire

4.1 Description du modèle	61
4.1.1 La chaîne de traitement visuel	62
4.1.2 La partie neuronale	64
4.1.3 Les mécanismes d'apprentissage	69
4.2 Expérimentations	72
4.2.1 Jeux de données	73
4.2.2 Méthodologie	75
4.2.3 Métriques	76
4.2.4 Résultats	80
4.3 Discussion	86
4.3.1 Comparaison à l'état de l'art bio-inspiré	86
4.3.2 Perspectives	90

Notre architecture de localisation visuelle est basée sur un modèle de cellules de lieu hippocampiques. L'entrée de ce modèle est alimentée par un mécanisme de vision attentionnelle, un processus cognitif que l'on retrouve souvent chez les mammifères. Ce processus permet de ne sélectionner et de ne traiter qu'une partie des informations visuelles, ce qui a pour avantage de réduire la charge de calcul nécessaire pour les traitements [Tsotsos *et al.* 2018].

Le modèle émule la structure globale du cortex visuel des primates par l'utilisation de deux voies de traitement : la voie ventrale "what" et la voie dorsale "where", correspondant au modèle dorso-ventral [Goodale & Milner 1992], pour détecter et reconnaître des objets.

Ce chapitre commencera par présenter le modèle de cellules de lieu visuelles ainsi que les adaptations qui y ont été apportées. Il décrira ensuite en détail la méthodologie et les mesures permettant de tester si le modèle est capable de fonctionner pour décrire un environnement de véhicule.

4.1 Description du modèle

Le modèle est constitué de deux sous ensembles, une chaîne de traitement visuel qui alimente une partie neuronale. La chaîne de traitement visuelle extrait une liste de coordonnées (x, y) de points d'intérêt (PoI : Point Of Interest). La partie neuronale est composée d'une boucle attentionnelle qui va traiter chacun de ces

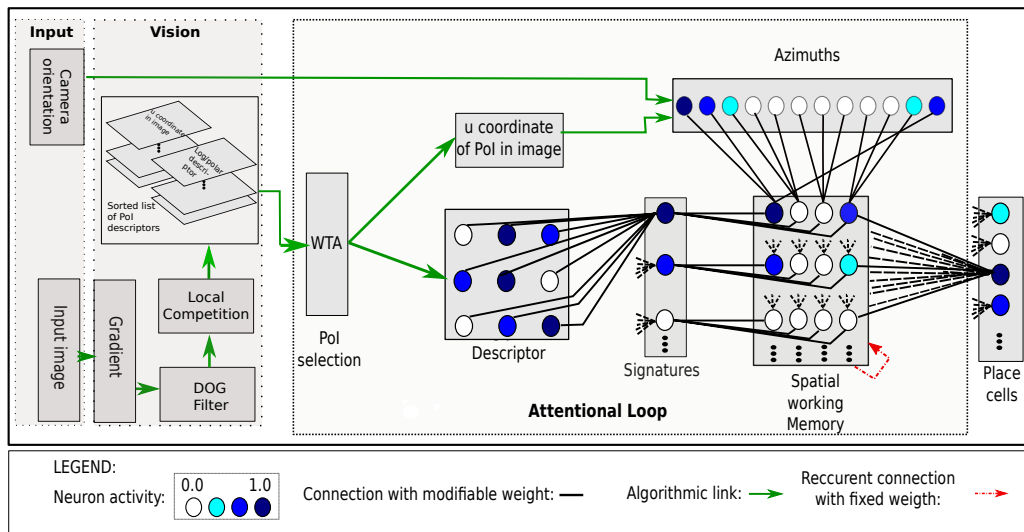


FIGURE 4.1 – L’architecture neuronale du modèle de reconnaissance visuelle de lieux. Il prend en entrée une image ainsi qu’une information d’orientation. Une chaîne de traitement visuel permet d’obtenir une carte de saillance qui est exploitée par un mécanisme attentionnel afin d’extraire des informations d’identification (what) et d’orientation (where). Un réseau neuronal stocke, combine et apprend ces données aboutissant à la construction de cellules de lieu visuelle. Pour plus de clarté l’ensemble des connexions neuronales sont cachées (lignes hachurées)[Espada *et al.* 2018]. Les connexions en verts représentent des liens algorithmiques (opérations mathématiques, etc), les connexions en noir représentent les liens synaptiques entre neurones dont les poids sont utilisés pour apprendre des informations.

points en extrayant une signature et un azimuth. Le résultat de ce traitement aboutit à la formation d’une mémoire spatiale caractéristique d’un lieu encodée sur une mémoire de travail (SWM : Spatial Working Memory) qui peut être apprise par la couche de cellules de lieu (PC : Place Cell). Un signal binaire d’apprentissage ($l(t)$) est utilisé pour synchroniser les apprentissages des réseaux de neurones.

4.1.1 La chaîne de traitement visuel

La chaîne de traitement visuel (voir figure 4.2) prend en entrée une image. Cette image subit une diminution de sa résolution afin de limiter les coûts computationnels liés aux traitements et ne faire ressortir que les POI liés aux plus grosses structures présentes dans l’image. Elle traduit aussi la faible acuité visuelle des rongeurs qui nous servent d’inspiration.

De cette image d’entrée on ne garde que l’image en niveau de gris sur laquelle est appliquée une série de traitements qui vont permettre d’extraire des points saillants (POI) de l’environnement. On commence par appliquer un filtre de gradient afin d’obtenir les contours de cette image. Ce contour est ensuite convolué avec un filtre

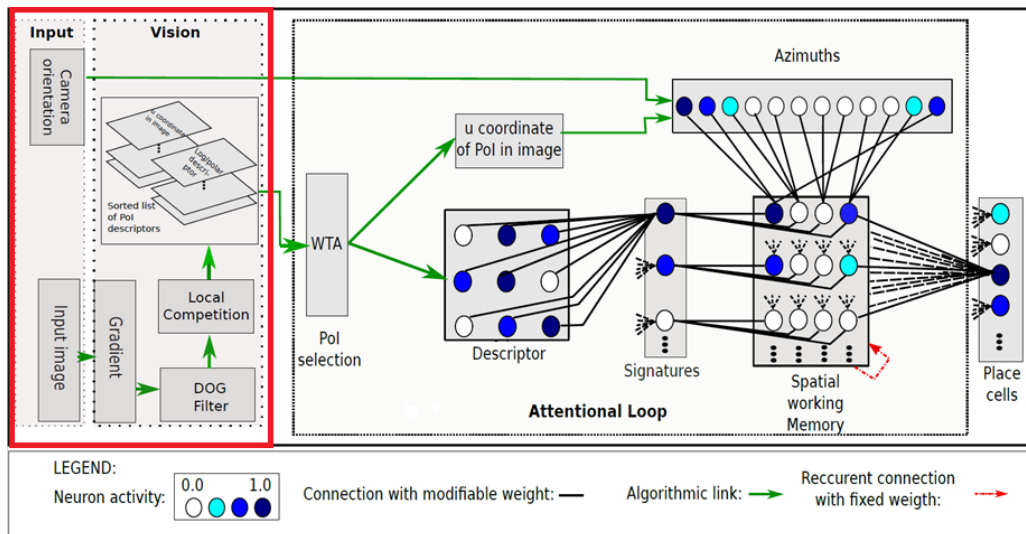


FIGURE 4.2 – La chaîne de traitement visuel encadrée en rouge, première partie du modèle de cellules de lieu visuelles [Espada *et al.* 2018].

de différence de gaussiennes (DoG) réhaussant les zones de fortes courbures caractéristiques des POI recherchés. Ce filtre est composé de deux gaussiennes dont les écart-types respectifs sont σ_{DoG_1} et σ_{DoG_2} . Grâce à ce filtrage couplé à un mécanisme de compétition locale permettant d’obtenir des maxima locaux, une carte de points saillants est générée.

Cette compétition est réalisée par une fenêtre glissante de taille r_{compet} permettant d’éviter d’avoir des points d’intérêt trop proches les uns des autres. Ces maxima locaux sont nos points d’intérêt PoI dont on en sélectionne N_{PoI} .

La valeur N_{PoI} est fixée empiriquement par retour d’expérience. Si on ne prend pas suffisamment de points d’intérêt on ne va garder que peu d’information pour décrire des lieux, ce qui réduit le coût computationnel mais diminue la capacité de discrimination des lieux et la robustesse aux occlusions partielles de l’environnement. Au contraire si on prend trop de points d’intérêt le coût computationnel et la robustesse aux occlusions vont augmenter mais les cellules de lieu risquent d’être trop discriminantes et provoquer du sur-apprentissage. Cette première série de traitements reproduit partiellement l’analyse des informations visuelles effectuée par la vision périphérique à partir des informations extraites par les bâtonnets de la rétine. La différence avec le modèle biologique vient essentiellement de la non-utilisation de la détection de mouvement qui augmenterait le coût computationnel pour un gain non encore évalué.

Pour chaque point d’intérêt on extrait une portion de l’image, que l’on appelle une vue locale, comprise entre deux disques de rayon r_{small} et r_{big} . Cette vue locale subit finalement une transformation en log polaire (voir figure 4.3).

Cette transformation a l’avantage d’être invariante aux petites rotations et petits changements d’échelle et donne de bons résultats en termes de reconnaissance

de lieux [Giovannangeli *et al.* 2006b][Belkaid *et al.* 2016]. Elle modélise de plus la structure de la rétine.

La liste des points d'intérêt et des vues locales projetées en log polaire sert d'entrée à la partie neuronale du système.

L'utilisation d'une carte de saillance pour se concentrer uniquement sur des régions localisées d'une image s'inspire du mécanisme attentionnel du système de vision de la plupart des mammifères qui fait que les yeux ne font pas un balayage systématique de l'environnement mais sautent d'un point d'intérêt à un autre [Koch & Ullman 1987].

Le cycle d'extraction de vue locale à partir de points d'intérêt, que l'on appelle la boucle d'exploration, simule la saccade oculaire d'un oeil virtuel qui parcourrait l'image. Si cet oeil virtuel ne garde qu'une portion de l'image projetée en log-polaire c'est parce qu'il reproduit partiellement la rétine : il ne garde que les informations captées par les récepteurs de la fovéa, la petite zone de la rétine qui contient beaucoup de récepteurs, et ces récepteurs ont une implantation similaire à la projection en log-polaire. r_{big} définit la taille de la fovéa. La projection en log polaire a pour effet secondaire de dilater l'information visuelle au centre de l'image initiale. Le rayon r_{small} sert donc à exclure la partie centrale de la vue locale fortement déformée par cet effet de dilatation.

4.1.2 La partie neuronale

Le réseau de neurones (voir figure 4.4) est basé sur un modèle de neurone à fréquence moyenne de décharge comprise entre 0 et 1. Ce n'est pas le modèle de neurones le plus biologiquement plausible mais il constitue un bon compromis entre fidélité et complexité calculatoire. Dans nos simulations, pour chaque pas de simulation, l'activation des neurones du réseau se fait séquentiellement de la gauche vers la droite sur l'ensemble des réseaux présents sur un script de code. A chaque pas de simulation, la fonction d'activation de chaque neurone est calculée en premier. Le calcul de la fonction d'apprentissage pour les neurones qui apprennent, dont les équations dépendent des fonctions d'activation d'autres neurones, se fait ensuite.

La partie neuronale du modèle est constituée de cinq couches de neurones : une couche d'entrée, le descripteur, qui est connecté à une couche de signature. Cette dernière et une couche d'azimut sont connectées à une mémoire de travail qui est elle même connectée à la couche de sortie qui correspond à nos cellules de lieu.

Le descripteur \mathbf{d} est l'entrée de la partie neuronale de notre modèle. C'est un champ neuronal de taille N_d sur lequel sont projetés les vues locales. Chaque neurone du descripteur encode la valeur, avec une remise à l'échelle entre 0 et 1, d'un pixel issu des vues locales transformées en log polaire avec le cas échéant, une réduction de la résolution. De ce fait, N_d , r_{small} , r_{big} et r_{compet} sont liés. N_d est proportionnel à la différence entre r_{big} et r_{small} . r_{compet} sert à garantir une distance minimum entre deux POI afin d'éviter au maximum que ceux-ci n'extraient une même information visuelle. Ce paramètre s'assure aussi que les POI sont pris sur l'ensemble de l'image d'entrée afin d'éviter de ne couvrir qu'une zone restreinte de l'image. r_{compet} est donc

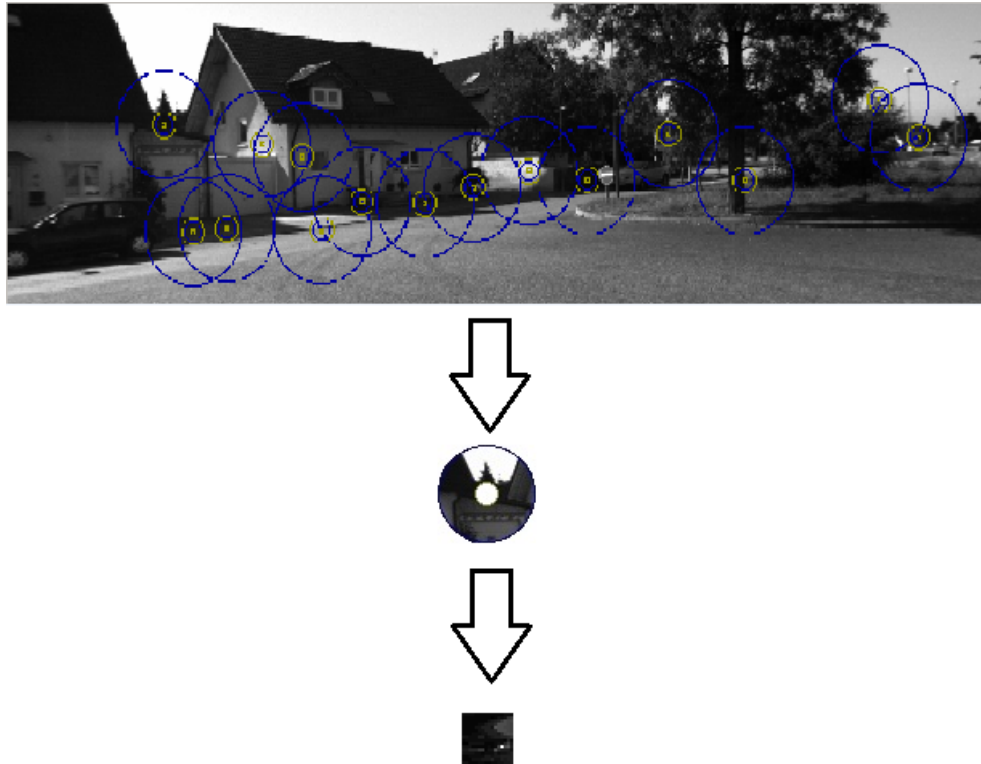


FIGURE 4.3 – Un exemple d'extraction d'une vue locale et de sa transformation en log polaire [Espada *et al.* 2018]. **Haut** - Sur l'image d'entrée on identifie des POI pour lesquels on va extraire une portion d'image définie par deux cercles concentriques (en bleu). **Milieu** - Représentation en log/polaire de la portion d'image associée à un point. **Bas** - Représentation de cette même signature du POI sous forme d'une carte neuronale.

proportionnel à r_{big} mais doit rester suffisamment faible pour que suffisamment de POI soient recrutés.

Chaque neurone de cette couche d'entrée est connecté à l'ensemble des neurones de la couche de signature par des liens dont les poids synaptiques sont modifiables et initialisés à une petite valeur aléatoire.

La couche de signature \mathbf{s} comporte N_s neurones. Elle a pour rôle d'encoder les signatures visuelles (le vecteur de taille N_d). Chaque neurone s_i mémorise une configuration de la couche du descripteur sur ses poids synaptiques. Chaque neurone voit son activité régie à chaque instant t par l'équation suivante :

$$s_i(t) = 1 - \frac{1}{N_d} \sum_{j=1}^{N_d} |w_{ij} - d_j(t)| \quad (4.1)$$

w_{ij} est le poids du lien synaptique reliant le i^{eme} neurone de la couche de signature

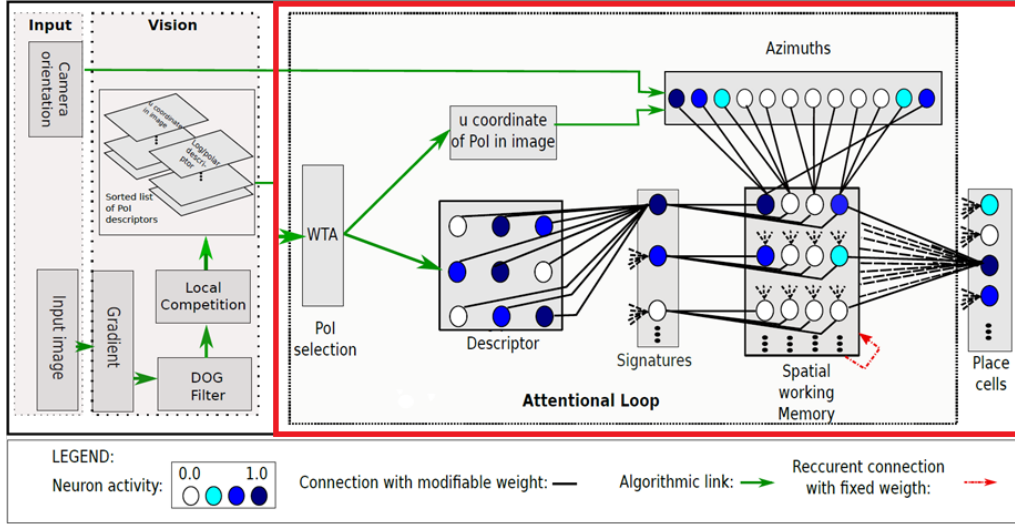


FIGURE 4.4 – La partie neuronale encadrée en rouge, seconde partie du modèle de cellules de lieu visuelles [Espada *et al.* 2018].

(s_i) au j^{eme} neurone appartenant au descripteur (d). $d_j(t)$ est l'activité du neurone j , appartenant au descripteur d , au temps t . Du fait de cette équation, plus la configuration des neurones de la couche du descripteur correspondra à la configuration mémorisée par le neurone i de la couche de signature, plus son activité $s_i(t)$ sera élevée. Cette activité est maximale, $s_i(t) = 1$, lors du recrutement du neurone i .

Le modèle utilise, en plus de l'information visuelle localisée autour d'un point d'intérêt (PoI), l'azimut, encodé par rapport à une référence fixe. Cette information est obtenue suite à un calcul en deux étapes. Il consiste, en premier lieu, à utiliser les coordonnées en pixel du PoI dans l'image et l'angle d'ouverture de la caméra pour calculer une orientation relative par rapport au centre de la caméra. Cette orientation est ensuite sommée avec l'orientation, modulo $2 \times \pi$, de la caméra. Dans notre cas l'orientation de la caméra est obtenue par une boussole magnétique. Notre référence est donc le nord magnétique. L'orientation de la caméra aurait aussi pu être obtenue par une boussole visuelle [Giovannangeli & Gaussier 2007][Delarboulas *et al.* 2014] comme le font certains mammifères sans sensibilité au champ, magnétique terrestre (primates, rats). La référence fixe aurait été un, ou plusieurs, amer visible depuis une grande distance comme dans le cas de la navigation maritime à l'estime.

Cette orientation est encodée sur une population de neurones de taille N_a [Georgopoulos *et al.* 1988] (voir figure 4.5). Le i^{eme} neurone de la couche d'azimut \mathbf{a} voit son activité $a_i(t)$ exprimée par un profil Gaussien non-normalisé défini par un écart type σ_{azim} selon l'équation 4.2.

$$a_i(t) = \exp \left[-\frac{(\theta^l - \frac{2\pi}{N_a}i) \bmod 2\pi)^2}{2\sigma_{azim}^2} \right] \quad (4.2)$$

θ^l est l'azimut selon lequel le l^{eme} PoI est observé. N_a est la taille de la population de neurones constituant la couche azimut \mathbf{a} . $\frac{2\pi}{N_a}i$ est la direction préférentielle propre au

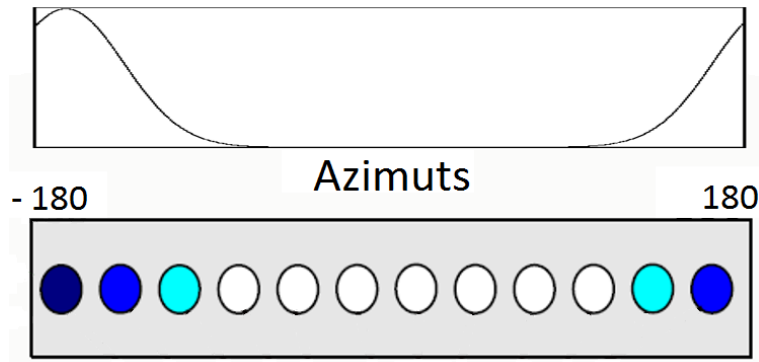


FIGURE 4.5 – L’encodage d’un azimut. L’orientation donnée par une boussole magnétique est encodée sous la forme d’une gaussienne centrée sur l’orientation. Cet encodage se fait par une population de neurones simulant ceux de directions de la tête définis par l’équation 4.2 [Espada *et al.* 2018].

neurone i . Grâce à la propriété de diffusion latérale des activations vers leurs voisins des neurones de cette couche, modélisée par le profil gaussien, il n’est pas nécessaire qu’un neurone de cette couche soit paramétré précisément selon l’orientation du PoI pour être activé. σ_{azim} est donc fixé pour obtenir une résolution de $\frac{360}{N_a}$ degré. Cette couche sert uniquement à projeter l’orientation en entrée sur la population et donc aucun apprentissage n’est effectué.

Cette couche d’azimut de notre modèle correspond à des cellules de direction de la tête (voir chapitre 3.2.3.2).

Les couches de signature \mathbf{s} et d’azimut \mathbf{a} sont connectées par des liens avec poids modifiables à un champ neuronal qui est une mémoire spatiale de travail (SWM) où sont combinées les projections des vues locales avec les orientations selon lesquelles leurs PoI respectifs ont été détectés.

Ensemble, ces trois couches neuronales forment une modélisation de la structure "what" et "where" du cortex visuel, la couche \mathbf{s} apportant la partie "what" et la couche \mathbf{a} le "where". Par conséquent, chaque neurone appartenant à la mémoire de travail (SWM) encode un amer visuel : il apprend à reconnaître une signature de taille N_d selon une plage d’orientation donnée.

Les neurones constituant la couche SWM sont organisés selon un champ neuronal découpé en deux dimensions.

La mémoire de travail est définie par les tailles $N_{SWM} = N_s \times N_l$. La couche SWN est constituée d’autant de lignes qu’il y a de neurones dans la couche de signature \mathbf{s} . Le nombre de colonnes spécifie selon combien d’orientations différentes il est possible de reconnaître un même vecteur de signature de taille N_d . Le codage d’une signature sur ces orientations différentes sera donc considéré comme appartenant à des amers différents.

Un neurone de la couche de signature s_i est connecté à l’ensemble de neurones de taille N_l formant la i_{eme} ligne de la mémoire de travail. Chacun des neurones

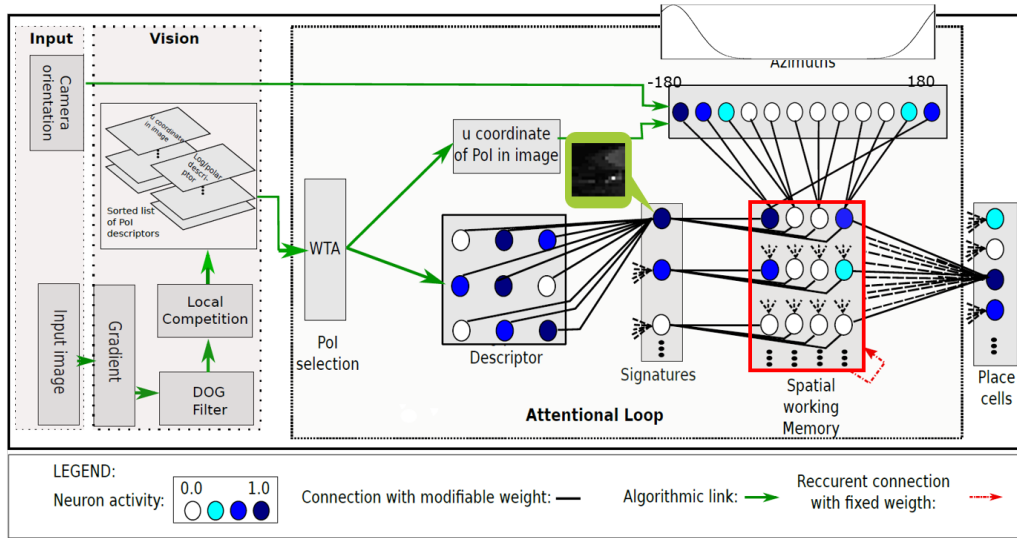


FIGURE 4.6 – La structure de la mémoire de travail, encadrée en rouge. Elle combine une vue locale sous forme d'un log polaire encodé sur un neurone de signature avec une orientation sous forme d'une gaussienne discrétisée sur une population de neurones [Espada *et al.* 2018].

de cet ensemble est aussi relié à un sous-ensemble exclusif de neurones appartenant à la couche **a**. Chacun de ces sous-ensembles est défini par une taille $l_\theta = \frac{N_a}{N_l}$. N_l paramètre la résolution de l'encodage de l'orientation dans la mémoire de travail. Sa valeur est un compromis entre la capacité à discriminer un même amer selon deux orientations différentes et le coût mémoire qu'entraîne une résolution d'encodage plus ou moins grande. Un schéma de cette structure est donnée en figure 4.6.

La diminution du nombre d'orientations encodées dans la SWM du fait de la structure neuronale permet d'obtenir une meilleure généralisation de la reconnaissance [Gaussier *et al.* 2002b] en n'ayant qu'une faible décroissance de l'activité d'un neurone de la couche SWM lorsque des petites variations d'azimut sont perçues. En effet, cette diffusion latérale permet de reconnaître un amer dans le cas où l'azimut selon lequel la signature est perçue reste proche de l'orientation selon laquelle a été appris l'amer.

Comme son nom l'indique, la mémoire de travail permet de former et d'accumuler les associations what and where pour chaque POI lors du parcours de la boucle attentionnelle. Ainsi l'activité du neurone x_{il} (sur la l_{eme} colonne et la i_{eme} ligne) de la mémoire de travail est défini par l'équation 4.3.

$$x_{il}(t) = f(x_{il}(t-1) + I_{il}(t) - x_{il}(t-1).In(t)) \quad (4.3)$$

In est un signal d'inhibition permettant d'effacer la mémoire de travail lorsque l'on passe d'une image à une autre. I_{il} est le signal d'entrée qui active le neurone x_{il} . Il

est calculé grâce à l'équation suivante :

$$I_{il}(t) = (s_i(t).w_{i,il}(t)).(\max_{j \in N_{al}}(a_j(t).w_{j,il}(t))) \quad (4.4)$$

Le i^{eme} neurone de la mémoire de travail est connecté à un sous-ensemble de neurones appartenant à la couche **a**. La liste des indices de chaque neurones de ce sous-ensemble est représentée par N_{al} dans l'équation précédente 4.4.

$f(x)$ est une simple fonction d'activation qui borne l'activité du neurone entre 0 et 1 :

$$\begin{aligned} f(x) &= 1 \text{ if } x > 1 , \\ &= 0 \text{ if } x < 0 , \\ &= x \text{ sinon.} \end{aligned} \quad (4.5)$$

Une fois que la boucle de traitement visuel pour l'image courante est terminée (l'extraction et la projection de vue locale pour chaque POI) la couche SWM est remplie. La configuration de la mémoire de travail que l'on obtient forme un vecteur de taille N_{SWM} qui caractérise le lieu courant. Chaque neurone de la couche SWM est connecté à l'ensemble des neurones de la couche de sortie, les cellules de lieu, par des liens de poids modifiables initialisés à de petites valeurs aléatoires.

La couche de sortie code les cellules de lieu (**p**) qui mémorisent les configurations de la SWM. Le fonctionnement de cette couche est identique à celui de la couche de signature (voir 4.1). Chaque neurone p_i de la couche de cellule de lieu a une activité définie par l'équation suivante :

$$p_i(t) = 1 - \frac{1}{N_{SWM}} \sum_{j=1}^{N_{SWM}} |w_{ij}(t) - x_j(t)| \quad (4.6)$$

x_j est le j^{eme} élément appartenant à la mémoire de travail. w_{ij} est le poids synaptique reliant le neurone p_i au neurone x_j de la couche SWM. N_{SWM} est la taille de la couche SWM.

Pour finir, sur cette dernière couche on utilise un mécanisme de type "winner-takes-all" [Grossberg 1973] pour ne garder que la cellule de lieu ayant l'activité la plus forte.

4.1.3 Les mécanismes d'apprentissage

Dans notre modèle tous les liens susceptibles d'apprendre ne le font pas par renforcement comme c'est le cas dans de nombreux modèles. Les poids sont plutôt appris avec un mécanisme d'apprentissage en un coup. Cette forme d'apprentissage a l'avantage d'être rapide permettant d'avoir un système qui apprend des lieux dans des délais compatibles avec l'utilisation en temps réel. Cette vitesse permet d'avoir un système qui peut s'adapter à la nouveauté en apprenant de nouvelles informations rapidement. Cette vitesse d'apprentissage est aussi plus proche du modèle de l'hippocampe dont une des caractéristiques est l'apprentissage rapide.

L'apprentissage est contrôlé par un signal de vigilance global, $l(t)$. Ce signal est binaire, valant 1 lors d'un apprentissage et 0 sinon. Chaque neurone i a un signal de vigilance propre ($l_i(t)$) issu de ce signal global suivant l'équation 4.7 :

$$l_i(t) = l(t).r_i(t) \quad (4.7)$$

Dans la version implémentée du modèle $r_i(t)$ est un signal binaire indiquant si le neurone i est en état d'apprendre. Le simulateur de neurones génère, en début de simulation, l'ensemble des neurones qui pourront être utilisés. Il garde, par la suite, une liste des neurones qui n'ont pas encore appris qu'il met à jour à chaque pas de simulation pour éviter d'écraser les précédents apprentissages. Ces neurones ont une activité résiduelle variant entre 0 et 0.2. Elle est issue d'un tirage aléatoire afin d'accélérer le temps d'exécution et de ne pas passer par le calcul de fonction d'activations de neurones dont les poids sont encore à leurs valeurs aléatoires initiales et qui donneraient des résultats équivalents. Dans un réseau donné à un instant t , parmi les neurones qui n'ont pas encore appris, le neurone avec l'activité résiduelle maximale sera celui qui pourra apprendre ($r_i(t) = 1$).

L'apprentissage d'une cellule de lieu est déclenché lorsque le signal de vigilance global est actif ($l(t) = 1$) et maintenu pour toute la durée d'un pas d'apprentissage. L'image d'entrée lors de ce pas de simulation est traitée par la chaîne de traitement visuel pour obtenir un ensemble de point d'intérêt et de vues locales. Chacune de ces vues locales passe par le descripteur \mathbf{d} et la couche de signature \mathbf{s} où elle est apprise, sous la forme du vecteur de taille N_d , par un neurone s_i sur ces poids synaptiques selon l'équation 4.8.

$$w_{ij}(t) = (1 - l_i(t)).w_{ij}(t - 1) + d_j(t).l_i(t) \quad (4.8)$$

$l_i(t)$ est le signal d'apprentissage du neurone s_i . w_{ij} est le poids du lien synaptique reliant le i^{eme} neurone de la couche de signature (s_i) au j^{eme} neurone appartenant au descripteur \mathbf{d} . Le signal $l_i(t)$ ne valant 1 qu'une fois pour chaque neurone lors d'une expérience, le poids n'est modifié qu'à cet instant et demeure inchangé pour le reste de la simulation.

La vue locale et son azimuth sont combinés dans la mémoire de travail (SWM). A cette occasion les poids synaptiques $w_{s_i, x_{il}}$ et $w_{a_j, x_{il}}$ liant le neurone x_{il} de la mémoire de travail respectivement au neurone actif s_i de la couche de signature \mathbf{s} et au neurone actif a_j de la couche d'azimut \mathbf{a} sont appris. L'ensemble des poids de x_{il} est initialement fixé selon des valeurs aléatoires faibles. Lorsque les liens entre x_{il} , s_i et a_j sont appris, la même règle que celle régissant l'apprentissage des neurones de la couche \mathbf{s} est employée.

$$w_{s_i, x_{il}}(t) = (1 - l_{i,l}(t)).w_{s_i, x_{il}}(t - 1) + l_{i,l}(t) \quad (4.9)$$

$$w_{a_j, x_{il}}(t) = (1 - l_{i,l}(t)).w_{a_j, x_{il}}(t - 1) + l_{i,l}(t) \quad (4.10)$$

Après l'application de ce mécanisme d'apprentissage, les poids synaptiques $w_{s_i, x_{il}}$ et $w_{a_j, x_{il}}$ valent chacun 1 alors que le reste des poids de x_{il} gardent leur valeur précédente.

Pour finir la configuration de cette mémoire de travail (SWM) est mémorisée par un neurone appartenant à la couche des cellules de lieu \mathbf{p} . Cet apprentissage se fait par l'intermédiaire des poids synaptiques du neurone p_i le reliant à la mémoire de travail (SWM). Sa règle d'apprentissage est la même que celle des neurones de signatures (4.8) à l'exception de l'activité des neurones de la couche \mathbf{d} qui est remplacée par celle des neurones de la mémoire de travail (SWM).

Dans notre modèle, l'apprentissage de cellules de lieu est régi par le principe suivant : on ne recrute une nouvelle cellule de lieu que si aucune des cellules de lieu déjà recrutée n'a de réponse (activité) supérieure à un seuil donné, nommé S_{vigi} . Le signal d'apprentissage global $l(t)$ est donc commandé par l'équation 4.11 :

$$l(t) = 1 \text{ if } \max_{\{i \in N_{pc}\}} |p_i(t-1)| < S_{vigi} , \quad (4.11)$$

$$= 0 \text{ sinon.}$$

$p_i(t-1)$ est l'activité du neurone i appartenant à la couche de cellules de lieu \mathbf{p} au pas de simulation précédent. N_{PC} est le nombre de cellules de lieu recrutées. S_{vigi} est un seuil d'activité compris entre]0 et 1[. Ce seuil commande le recrutement de cellule et donc son paramétrage est primordial. Un seuil trop faible mènera à un faible recrutement de cellules de lieu et donc une représentation peu précise. Au contraire un seuil trop élevé va entraîner un apprentissage trop important, donc une représentation segmentant beaucoup trop l'environnement pour être exploitée et avec de nombreux cas de confusion entre les lieux.

4.2 Expérimentations

Le modèle a été soumis à une série d'expériences afin de remplir un certain nombre d'objectifs :

- vérifier que le modèle est capable de fonctionner en environnement extérieur
 1. le passage à une plus grande échelle est possible
 2. le modèle est capable de fonctionner dans différents types d'environnement automobile
 3. le modèle a une certaine robustesse aux perturbations de l'environnement
- étudier l'influence du seuil de vigilance S_{vigi} dans différents environnements
- comparer deux mécanismes de recrutements
- obtenir des éléments de comparaison avec des modèles de la littérature

Le fonctionnement du modèle de cellules de lieu visuelles en environnement de conduite constitue l'objectif principal. Ce modèle a été initialement imaginé pour modéliser les cellules de lieu et reproduire le comportement de ces neurones enregistrés chez le rat. Il a été conçu pour fonctionner embarqué sur des petits robots mobiles afin de reproduire les expériences de navigation lors desquelles ont été effectuées les enregistrements de ces cellules. Ce modèle a donc été pensé pour fonctionner dans des environnements intérieurs de petites tailles. Au cours des expériences réalisées le modèle a été testé dans des environnements dans lesquels évoluent habituellement des véhicules. Ce sont des environnements extérieurs qui sont donc soumis à de nombreux changements, comme des variations de luminosité ou des modifications de l'environnement. De plus, ces environnements extérieurs sont bien plus variés, campagne, autoroute, centre ville, banlieue, mais aussi de bien plus grande taille. Les expériences sur les rats se déroulaient dans des environnements de l'ordre du mètre alors que les véhicules peuvent évoluer sur des environnements de plusieurs dizaines de kilomètres.

Soumettre notre modèle de cellules de lieu visuelles à ce type d'environnement va nous permettre d'observer un certain nombre de caractéristiques de notre modèle. La première, est de voir si notre modèle est capable de supporter le passage à l'échelle. La seconde est d'observer quelle est l'influence des différents types d'environnement sur nos cellules de lieu afin de déterminer l'adaptabilité de notre modèle. Une autre encore, est d'attester de la robustesse de nos cellules de lieu aux différentes perturbations que l'on peut rencontrer au sein de ces environnements.

Cela nous permet enfin de voir les limites du modèle actuel afin de déterminer quelles sont ses lacunes. Par exemple, une interaction avec une autre zone du cerveau qui n'a pas été encore modélisée. Cette réflexion est d'autant plus importante que les chercheurs en neurosciences, étant jusqu'à récemment limités par les outils de mesures à leur disposition, ont effectués peu d'expériences sur la navigation sur de grandes distances.

L'un des objectifs des expériences concerne la paramétrisation de la vigilance pour le mécanisme d'apprentissage. L'exploration de paramètre a pour objectif

Jeux de données KITTI	00	01	05	06	07	09
Nb d'images	4541	1101	2761	1101	1101	1501
Type	urbain	autoroute	banlieue	banlieue	urbain	banlieue
Taille (m)	3721	2450	2203	1232	695	1701
Fréquence d'échantillonnage (Hz)	10	10	10	10	10	10
Vitesse moyenne (m/s)	7.90	21.54	7.61	10.80	6.09	10.33

TABLE 4.1 – Les caractéristiques des jeux de données KITTI utilisés

d'identifier les plages de valeurs permettant d'obtenir une réponse de localisation à la fois robuste et suffisamment précise pour être exploitée. Les expériences permettent d'étudier l'influence de la valeur de son seuil S_{vigi} , sur la précision de reconnaissance, la capacité de généralisation et de détection de fermeture de boucle de notre modèle. Les tests dans des environnements variés (autoroute, banlieue, centre ville) va permettre aussi de savoir si l'environnement influe sur la robustesse de ce mécanisme d'apprentissage.

La comparaison du mécanisme d'apprentissage présenté dans la section 4.1.3, avec un autre système couramment employé dans d'autres approches (voir section 4.3.1) va permettre de vérifier l'hypothèse selon laquelle le mécanisme se basant sur l'activité des cellules de lieu déjà apprises devrait être plus adaptable que l'ancien système. Dans l'ancien modèle l'activation du signal de vigilance dépend d'un pas de distance parcourue. Ce pas peut être une distance ou un nombre d'image.

4.2.1 Jeux de données

Notre modèle prend en entrée des informations visuelles et une orientation absolue afin de faire de la reconnaissance de lieux.

Il existe peu de base de données comportant l'orientation. La plupart des modèles de reconnaissance visuelle de lieux ne reposent que sur la vision et utilisent des méthodes d'odométrie visuelle afin d'obtenir l'information d'orientation. Ils ont donc été testés sur des bases de données où l'information d'orientation est absente. De ce fait, il nous était impossible d'utiliser ces bases de données pour nous comparer. Le choix s'est donc porté sur des bases de données utilisées par des modèles SLAM et d'odométrie visuelles qui fournissent l'orientation et la vérité terrain. Le but de ces modèles et de faire du positionnement métrique qui est une tâche différente de la reconnaissance de lieu opérée par notre modèle. Il est donc, encore une fois, très difficile de faire une comparaison avec notre modèle.

Pour l'essentiel, les expériences se sont déroulées sur le jeu de données KITTI [Geiger *et al.* 2012]. C'est une série de bases de données d'images représentant des trajets de taille plus ou moins longue, avec ou sans rebouclage, dans des environnements variés allant des centres urbains à la campagne en passant par l'autoroute.

Pour les premiers trajets, une vérité terrain donnant la position de chaque image est fournie. Le détail des différents trajets peut être consulté sur le tableau 4.1.

Deux bases de données supplémentaires ont été constituées avec des véhicules

appartenant à l'institut VEDECOM (voir tableau 4.2). Chacun des deux véhicules utilisés, un Renault Scenic et une Renault Zoe, (voir figure 4.7) est muni d'une série de télémètres laser disposés afin de percevoir l'ensemble de l'entourage du véhicule, d'un GPS RTK, d'une centrale inertielle ainsi que de deux caméras orientées vers l'avant des véhicules pour faire de la stéréoscopie dont on n'utilise que celle de gauche.



(a) La renault Zoe.



(b) Le renault Scenic.

FIGURE 4.7 – Les véhicules de l'institut VEDECOM [Espada *et al.* 2018].

La première base de données a été prise de jour avec un Renault Scenic (voir figure 4.7b) sur un tronçon d'autoroute près de Versailles. Le second trajet a été enregistré de jour également avec une Renault Zoe (voir figure 4.7a) sur la piste d'essai de Satory (Versailles, France). L'environnement de cette piste d'essai peut être considéré comme étant un environnement rural. L'enregistrement a été effectué dans des conditions de luminosité défavorables (voir figure 4.8a).



(a) Une image prise sur la piste de Satory.



(b) Une image prise sur le tronçon d'autoroute.

FIGURE 4.8 – Exemple d’images acquises avec les véhicules de VEDECOM [Espada *et al.* 2018].

Pour l’ensemble des bases de données les orientations des caméras sont calculées à partir des informations GPS servant de vérité terrain.

4.2.2 Méthodologie

Les expériences ont été faites hors ligne en utilisant des données réelles issues d’enregistrements pris à bord de véhicules en mouvement.

Pour chaque trajet, une expérience se déroule en deux passes de la base d’image : un premier passage durant lequel s’effectue l’apprentissage des cellules de lieu et un second passage durant lequel est testé l’apprentissage précédent.

Le modèle de cellules de lieu visuelles est basé sur un apprentissage en un coup. Cela signifie qu’une cellule de lieu est apprise sur une seule image. Par ailleurs, le système de vigilance dépend de l’activité des cellules de lieu précédemment apprises. Il est donc impossible de déterminer combien de cellules de lieu seront recrutées pour un trajet donné et de prédire quelles images seront utilisées pour l’apprentissage, on ne peut donc pas scinder la base d’apprentissage et de test comme c’est le cas lorsque l’on effectue l’apprentissage avec un pas fixe de distance. De ce fait, seule une portion de la base de donnée est effectivement utilisée lors de l’apprentissage,

Jeu de donné	Piste de Satory	Autoroute en Scenic
Nb d'images	1366	2000
Type	rural	autoroute
Taille (m)	4064	4082
Fréquence d'échantillonnage (Hz)	4	12
Vitesse moyenne (m/s)	12.74	25.53

TABLE 4.2 – Les caractéristiques des jeux de données constitués au sein de l'institut VEDECOM

on ne risque pas donc d'avoir de sur apprentissage.

L'ensemble des paramètres invariants utilisés est donné dans la table 4.3.

Dans une première série d'expériences, on utilise le système d'apprentissage apprenant une cellule tous les 10 mètres. Pour le reste des expériences, on utilise le mécanisme de vigilance basé sur le seuil s_{vigi} . Pour chaque trajet, on effectue une série d'expériences entre lesquelles on fait varier la valeur du seuil de vigilance S_{vigi} . L'activité d'une cellule de lieu varie entre 0 et 1 mais comme expliqué précédemment en section (4.1.3), les neurones non recrutés ont une activité variant aléatoirement entre 0 et 0.2. De plus, par expérience, l'activité d'une cellules de lieu est généralement en dessous d'une valeur de 0.8 même lorsque l'on est proche du lieu d'apprentissage car les PoI ne sont pas systématiquement retrouvés sur des images proches, un amer est rarement parfaitement reconnu et donc un lieu est rarement parfaitement reconnu. Par conséquent S_{vigi} varie, au cours de la série d'expériences, d'une valeur de 0.3, le seuil à partir duquel la réponse devient significative par rapport au bruit, à 0.8, la valeur limite au dessus de laquelle une cellule de lieu serait recrutée toutes les deux images. La variation de S_{vigi} se fait avec un pas fixé arbitrairement de 0.02. Pour chaque passage de la base de données d'images, on enregistre l'activité de toutes les cellules de lieu.

Le fonctionnement du modèle à une échelle plus grande sera évalué par la précision de la représentation obtenue, par la fiabilité (nombre de fausses reconnaissances) des réponses de localisation et par le coût computationnel du modèle. L'adaptabilité du modèle aux différents types d'environnements sera évaluée en comparant la précision des représentations obtenues pour chaque environnement. La comparaison entre les deux mécanismes d'apprentissage se fera aussi par comparaison de la précision de leur représentation respective.

4.2.3 Métriques

Pour évaluer les performances du modèle et de son système de recrutement de cellules de lieu, on se concentre donc sur deux critères : le coût computationnel du modèle et le compromis entre précision et généralisation de la réponse des cellules de lieu créées. Pour estimer le coût computationnel du modèle, on mesure le nombre de neurones utilisés par la mémoire de travail (SWM), pour stocker les vues locales

Paramètre	Valeur
σ_{DoG_1}	8 pixels
σ_{DoG_2}	2 pixels
r_{small}	10 pixels
r_{big}	64 pixels
r_{compet}	32 pixels
N_{PoI}	30
N_d	2916
<i>Image width</i>	642
<i>Image height</i>	188
$N_{azimuth}$	361
N_a	60
σ_{azim}	0.5
N_i	3

TABLE 4.3 – Les paramètres du modèle utilisés lors des expériences.

ainsi que ceux pour recruter les cellules de lieu.

Pour qualifier la réponse des cellules de lieu, on se concentre sur la capacité du modèle à discriminer les lieux en termes de précision, de rappel et de capacité de généralisation. La précision de notre système est mesurée grâce à trois critères. Le premier critère est le ratio gagnant sur bruit (WNR). Il est calculé en faisant le ratio de l'activité moyenne des trois cellules de lieu ayant les activités les plus élevées sur l'activité moyenne des autres cellules de lieu. Cette mesure est similaire à un ratio signal sur bruit classique où l'on considère l'activité des cellules de lieu non gagnantes comme du bruit. Cette mesure permet de vérifier que la réponse de nos cellules de lieu est significative donc plus ce ratio est élevé meilleur est la performance du système.

Le second critère est le nombre d'erreurs de reconnaissance des cellules de lieu. Comme on effectue le même trajet quand on fait l'apprentissage et quand on teste ce dernier, on peut prédire l'ordre d'activation des cellules de lieu. Si cet ordre n'est pas respecté nous avons une erreur. Du fait de la méthode de calcul, une erreur de reconnaissance ayant lieu sur plusieurs images successives n'est comptée qu'une seule fois revenant à compter le nombre de champs de lieu mal positionnés. Plus ce nombre d'erreurs est faible, mieux c'est.

Le dernier critère servant à qualifier la précision des cellules est la capacité du modèle à reconnaître les rebouclages, le fait de repasser sur une même section de l'environnement lors d'un trajet. On utilise les courbes de précision-rappel ainsi que le score F1 afin de mesurer cette capacité. Le score F1 est calculé selon la formule 4.12.

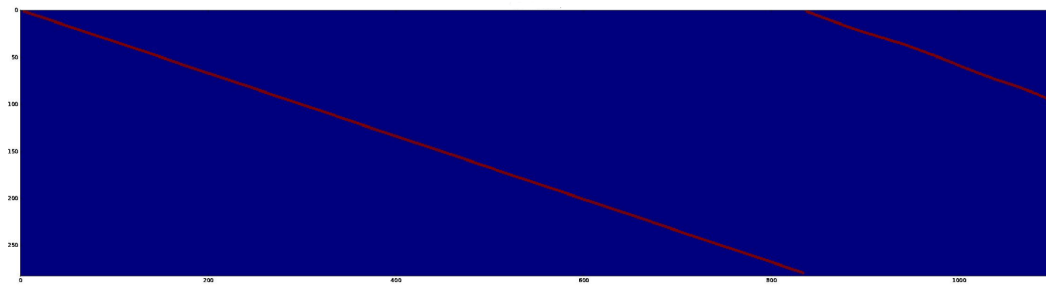
$$F1 = 2. \left(\frac{precision \times rappel}{precision + rappel} \right) \quad (4.12)$$

Où la précision est calculée en faisant le ratio des cellules qui répondent correctement sur l'ensemble des cellules qui répondent et le rappel est obtenu par le ratio des cellules qui répondent correctement sur l'ensemble des cellules qui devraient répondre.

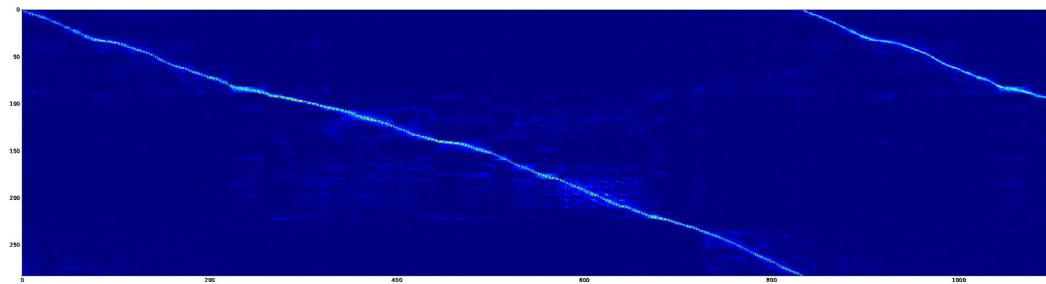
Ce sont des mesures classiquement utilisées pour évaluer les performances des approches de reconnaissance de lieu visuelle. Des scores proches de 1 indiquent une bonne capacité de reconnaissance des rebouclages. Ces scores sont calculés à partir de matrices de confusion. Ce sont généralement des matrices carrées qui donnent la correspondance spatiale entre chaque image d'abscisse avec chaque image d'ordonnée. Cette correspondance varie entre 0, les images ont été prises dans des lieux différents, et 1, les images ont été prises au même endroit. Dans le cadre de ce manuscrit, le modèle ne fait pas de la reconnaissance d'image mais un découpage de l'environnement en cellules de lieu. Dans ce cas les matrices de confusion font la correspondance entre chaque image et chaque cellule de lieu. Chaque image est associée à l'indice de la cellule de lieu qui répond le plus. Pour la vérité terrain (voir figure 4.9a), elle est construite en faisant l'hypothèse que les cellules de lieu découpent l'environnement de manière uniforme en termes d'image, sauf zone de rebouclage. Par exemple un environnement de 1000 images où sont recrutées 100 cellules de lieu, les images seront regroupées par groupe de 10. Les 10 premières images correspondant à la première cellule, les 10 suivantes à la deuxième cellule et ainsi de suite.

La capacité de généralisation est une propriété centrale du modèle. Cette dernière permet aux cellules de lieu de ne pas reconnaître uniquement les lieux là où elles ont été apprises mais aussi aux voisinages de ces lieux. Les cellules de lieu génèrent des champs de lieu grâce à cette propriété. On utilise deux critères pour estimer la capacité de généralisation.

Pour évaluer la réponse de localisation de notre modèle de cellules de lieu, on mesure la taille des champs de lieu. On restreint leur définition à la zone où une cellule de lieu est la seule à répondre le plus fortement. Pour chaque trajet, on mesure la taille moyenne des champs de lieu ainsi que l'écart type. Pour ce critère, on ne peut pas affirmer qu'il y a de bonnes ou de mauvaises valeurs, à part dans les cas extrêmes, indiquant soit l'absence de champs de lieu soit une cellule qui ne discrimine pas un lieu. L'arbitrage devra s'effectuer par la suite au niveau comportemental : les champs de lieu générés permettront-ils de contrôler correctement le véhicule ? La taille moyenne du champ de lieu doit être mise en rapport avec la précision souhaitée dans les cas, comme le nôtre, où l'on utilise que la réponse de la cellule la plus active. Pour l'écart type, il doit être mis en rapport avec le type d'environnement traversé lors du trajet. Cette mesure nous renseigne sur la régularité des champs de lieu au cours d'un trajet. Dans le cas d'une trajectoire où, par exemple, on passe d'une ville à une autoroute cette régularité n'est pas forcément souhaitable. On passe d'un environnement où une grande précision, donc de petits champs de lieu, est nécessaire (ville) à un environnement où une plus faible précision, donc de plus grands champs de lieu, est suffisante (autoroute). Ces mesures sont donc à interpréter en fonction des environnements traversés.



(a) une vérité terrain



(b) Exemple de matrices de confusion.

FIGURE 4.9 – Exemple de vérité terrain 4.9a et de matrices de confusion 4.9b pour la base KITTI 06. La vérité terrain, construite pour un nombre donné de cellules de lieu, indique à quelle cellule devrait correspondre chaque image. La matrice de confusion donne pour chaque image à quelle cellule de lieu correspond réellement chaque image. Pour un système avec une parfaite reconnaissance des fermetures de boucles avec des champs de lieu de taille régulière, la vérité terrain et la matrice de confusion seraient identiques.

Le second critère pour évaluer la capacité de généralisation est la distance aux trois gagnants (3WD) (voir figure 4.10). Cette mesure est une moyenne d'un ratio calculé pour chaque image i selon l'équation 4.1.3.

$$3WD_i = \frac{D_a + D_b + D_c}{\times D_{ac}} \quad (4.13)$$

Elle prend, au numérateur, la somme des distances entre la position réelle du véhicule, fournie par la vérité terrain, et les positions où ont été apprises les trois cellules de lieu ayant la plus forte activité ($D_a + D_b + D_c$). Le dénominateur est calculé grâce à la distance la plus grande séparant deux des trois cellules de lieu gagnantes (D_{ac}). Ce ratio nous indique si les lieux les plus proches de la position du véhicule sont ceux dont l'activité est la plus élevée. Par construction, un ratio dont la valeur est proche de 1 signifie que les trois cellules les plus proches sont celles qui répondent le plus indiquant la pertinence des réponses du système, et que dans le cas d'erreurs dans l'ordre de reconnaissance des cellules, cette confusion se fait avec des cellules apprises en des lieux proches de celle qui devrait répondre.

L'ensemble des tests concernant le paramétrage de la valeur de seuil s_{vigi} est

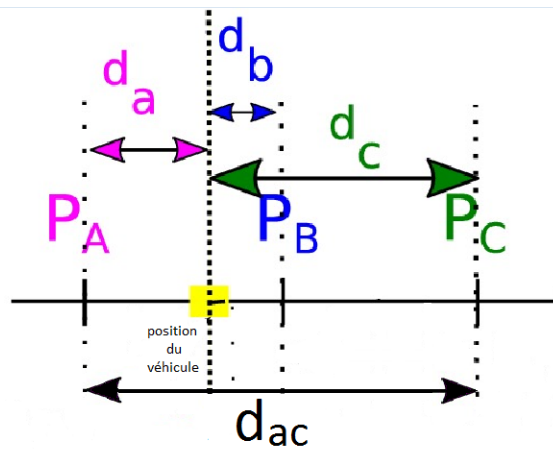


FIGURE 4.10 – schéma explicatif du calcul du 3WD

réalisé sur l'ensemble des bases de données présenté. Les bases de données KITTI 00 (centre ville) KITTI 06 (banlieue) et KITTI 07 (centre ville) servent aussi à effectuer la comparaison, par l'intermédiaire des mesures de score F1, avec un modèle de l'état de l'art, Fabmap 2.0 [Cummins & Newman 2009]. Cette comparaison, bien qu'imparfaite, permet d'avoir une intuition du bon fonctionnement de notre approche. La base de données KITTI 06 est en plus utilisée pour faire les comparaison entre les deux mécanismes d'apprentissage.

4.2.4 Résultats

Comme indiqué précédemment pour la base KITTI 06, un test complémentaire a été effectué avec le modèle de vigilance reposant sur un pas fixe. Ce modèle de vigilance est utilisé dans certains modèles notamment avec le modèle RatSLAM dans l'expérience de Sainte Lucie [Milford & Wyeth 2008]. Dans cette expérience, une cellule de lieu est apprise tous les 10 mètres. Avoir un pas de distance fixe pour le recrutement permet, pour un trajet d'une distance connue, de prédire le nombre de cellules de lieu qui sera recruté et d'avoir un découpage régulier de l'environnement. Dans notre expérience, la base d'apprentissage est composée de 124 images. Cependant, sans une connaissance a priori de l'environnement dans lequel le véhicule va évoluer permettant de régler finement ce paramètre, le risque d'une couverture lacunaire de l'environnement par les cellules de lieu augmente. Les résultats d'expérience (voir figure 4.11) montrent des exemples de couverture insuffisante d'environnement.

Ce problème provoque des erreurs de reconnaissance séquentielles supplémentaires et amène donc à une plus faible précision. De plus, le système d'apprentissage en un coup peut potentiellement encore aggraver ces erreurs. Dans le cas hypothétique où, à cause des limitations du système visuel, l'apprentissage se porterait sur des éléments non pertinents (des objets mobiles comme des voitures ou des passants par exemple), le système apprendrait une cellule de lieu impossible à reconnaître ag-

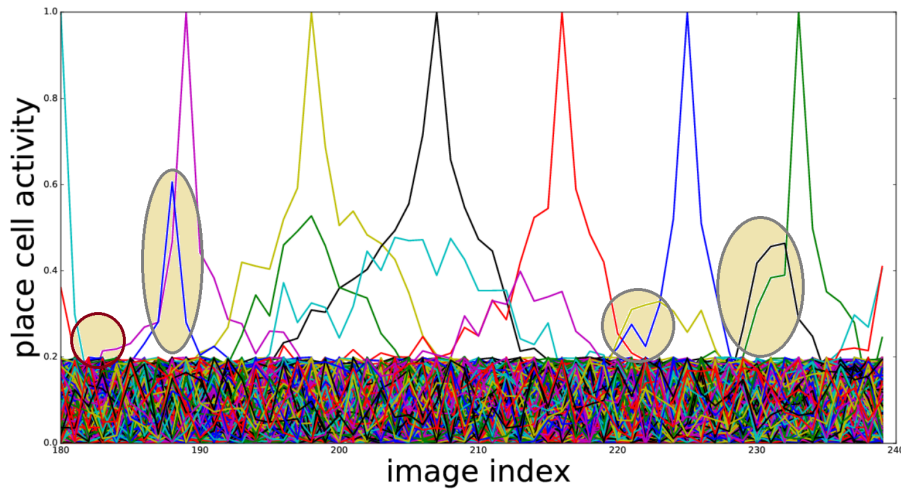


FIGURE 4.11 – Tracé des activités des cellules de lieu, sur la base de données KITTI 06, avec le mécanisme basé sur un pas fixe. Les zones mise en valeurs avec un contour gris correspondent à des erreurs dans l'ordre de reconnaissance des cellules de lieu. La zone mise en valeur avec un contour marron montre une absence de recouvrement par la représentation

gravant la couverture insuffisante de l'environnement. Le problème inverse peut aussi provoquer une augmentation du nombre d'erreurs de reconnaissance séquentielles. Un pas trop faible, dans un environnement qui varie peu, comme une autoroute par exemple, peut provoquer l'apprentissage de plusieurs cellules de lieu qui se baseraient sur les mêmes points d'intérêt. Elles auraient donc des activités proches provoquant des erreurs.

Par ailleurs, cette nécessité de régler le pas d'apprentissage rend le modèle peu adaptable aux différents types d'environnements.

Pour le reste des expériences, on utilise le système de recrutement basé sur un seuil de reconnaissance des cellules de lieu déjà recrutées.

Dans les tableaux 4.4, pour la base KITTI et 4.5 pour les bases de VEDECOM sont donnés les résultats des expériences avec ce système de recrutement.

Suite à une exploration de paramètre pour ce seuil de vigilance entre les valeurs 0.3 et 0.8 avec un pas de 0.02 on a déterminé qu'une valeur de seuil égale à 0.42 (voir figures 4.13, 4.14 et 4.15) donne généralement le meilleur compromis entre les différentes métriques pour les environnements urbains et péri-urbains. Pour cette valeur on obtient à la fois une représentation robuste (scores 3WD, WNR et F1 élevés et nombre d'erreurs faible) tout en gardant une représentation compacte (faible nombre de cellules de lieu recruté).

Pour la base de données KITTI 01, un trajet effectué sur autoroute, du fait d'un environnement plus pauvre en termes d'amer, la valeur 0.42 pour le seuil de vigilance donne un plus grand nombre d'erreurs pour la reconnaissance des lieux. En contrepartie on obtient une amélioration de la précision grâce à des champs de

Base de donnée	Vigilance	Nombre de PC	Score 3WD	WNR	Erreurs dans l'ordre des PC	Score F1	Taille moyenne des PC (variance) (m)
00	0.30	1089	0.794	23.2	11	0.9999	1.41 (1.17)
00	0.42	1520	0.801	25.8	15	0.9999	0.83 (0.87)
00	0.80	1909	1.94	25.2	24	0.9998	0.96 (0.62)
01	0.30	55	1.17	7.32	17	non pertinent	65.18 (269.63)
01	0.42	122	0.844	8.62	24	non pertinent	18.14 (179.43)
01	0.80	544	0.791	11.1	19	non pertinent	9.99 (84.73)
05	0.30	609	1.05	20.8	12	0.9999	1.21 (1.35)
05	0.42	885	1.30	23.3	9	0.9998	0.95 (0.87)
05	0.80	1357	1.51	22.6	20	0.9998	0.81 (0.55)
06	0.30	186	0.786	13.7	7	0.9998	4.71 (2.739)
06	0.42	283	0.779	16.1	3	0.9997	3.12 (1.13)
06	0.80	419	0.784	16.9	6	0.9995	2.17 (0.93)
06	fixed step (10m)	87	1.03	8,63	44	0.9986	7.61 (3.96)
07	0.30	233	0.790	19.6	4	0.9998	1.73 (0.94)
07	0.42	328	0.880	21.1	2	0.9997	1.07 (0.76)
07	0.80	533	0.928	21.0	4	0.9996	0.78 (0.42)
09	0.30	400	0.776	18.8	6	0.9998	3.34 (5.19)
09	0.42	562	0.772	21.9	3	0.9998	2.51 (1.87)
09	0.80	787	0.909	22.8	2	0.9996	1.23 (0.78)

TABLE 4.4 – Les métriques pour des valeurs de seuil s_{vigi} extrémales (0.3 et 0.8) et celle considérée optimale (0.42) pour l'ensemble des bases de données KITTI utilisées

Base de donnée	Vigilance	Nombre de PC	Score 3WD	WNR	Erreurs dans l'ordre des PC	Score F1	Taille moyenne des PC (variance) (m)
piste de Satory	0.30	2	non pertinent	non pertinent	5	0.6666	249.68 (1782.35)
piste de Satory	0.42	120	1.67	6.38	71	0.7739	11.96 (36.76)
piste de Satory	0.80	370	15.73	8.48	128	0.5835	5.81 (5.24)
Scenic autoroute	0.30	101	0.83	7.9	25	non pertinent	28.66 (84.40)
Scenic autoroute	0.42	221	0.80	12.6	15	non pertinent	15.52 (26.14)
Scenic autoroute	0.80	642	1.09	16.5	2	non pertinent	5.48 (4.49)

TABLE 4.5 – Les métriques pour des valeurs de seuil s_{vigi} extrémales (0.3 et 0.8) et celle considérée optimale (0.42) pour les bases de données VEDECOM

lieu de plus petites tailles et donc une meilleure résolution de la représentation.

Sur ces résultats, on peut observer qu'avec l'augmentation du seuil de vigilance le nombre de cellules de lieu recrutées augmente. Par conséquent on obtient une représentation d'une plus grande précision car le champ de lieu propre à chaque cellule de lieu recouvre une plus faible surface. Le nombre de cellules de lieu apprises est fortement affecté par le type d'environnement parcouru. Pour un seuil de 0.42, pour la base de données KITTI, en environnement urbain on obtient des champs de lieu dont la taille moyenne est de l'ordre du mètre alors que sur autoroute la taille moyenne des champs de lieu est de 18.14 mètres. De plus, comme le montre l'expérience sur la base de données Scenic, dans un environnement autoroutier, la présence de circulation influe aussi sur la taille des champs de lieu. Les tailles moyennes des champs de lieu de 18.14 m (179.43 m) sur une autoroute dégagée (KITTI 01) et 15.52 m (26.14 m) sur une autoroute avec de la circulation (Scenic), illustrent cet effet. Cette différence dans la taille des champs de lieu entre les environnements de type urbain ou péri-urbain et ruraux ou autoroutiers s'explique par la distance des

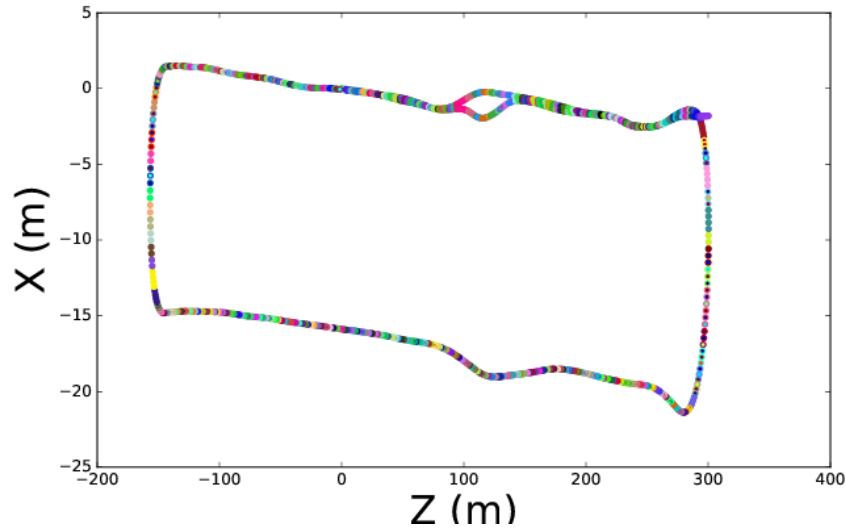


FIGURE 4.12 – Tracé de la trajectoire de la base de données KITTI 06 pour le mécanisme d'apprentissage basé sur le seuil de vigilance. Le champ de lieu associé à chaque cellules de lieu est dessiné avec une couleur différente.

amers qui servent à construire les cellules de lieu. Dans des environnements urbains ou péri-urbains la vue est très encombrée par un nombre importants d'objets (murs, véhicules, etc) et nombre de ces objets sont mobiles (passants, véhicules dans le trafic). Par conséquent, les amers que l'on peut percevoir dans ces environnements vont, la plupart du temps, se retrouver rapidement hors de vue. De plus, la configuration de ces environnements avec ces nombreux virages serrés amplifie ce phénomène. Les cellules de lieu créées par ces amers auront des activités qui décroissent rapidement du fait de la disparition de ces mêmes amers. Par conséquent, les champs de lieu définis par ces cellules seront de petites tailles et plus de cellules seront recrutées. Au contraire, sur autoroute ou en milieu rural, la ligne de vue est plus dégagée. Par ailleurs, ces environnements sont généralement plus pauvres en termes de points d'intérêt. De ce fait, les amers présents dans ces environnements sont peu nombreux, visibles sur de longues distances. Par conséquent les amers détectés par le modèle seront reconnus pendant plus longtemps et les cellules de lieu créées par ces amers auront une activité qui se maintiendra sur de grandes distances. Ces cellules généreront donc des champs de lieu plus larges et moins de cellules de lieu seront recrutées.

Dans l'ensemble des expériences, le ratio gagnant sur bruit (WNR) reste élevé malgré une légère influence du seuil de vigilance sur ce ratio.

Ce ratio élevé indique deux informations sur la réponse des cellules de lieu. La première est que les cellules gagnantes ont une activité significativement élevée par rapport aux autres cellules de lieu. La deuxième information est que, comme la réponse de l'ensemble des PC non gagnantes est faible, l'ensemble des cellules de

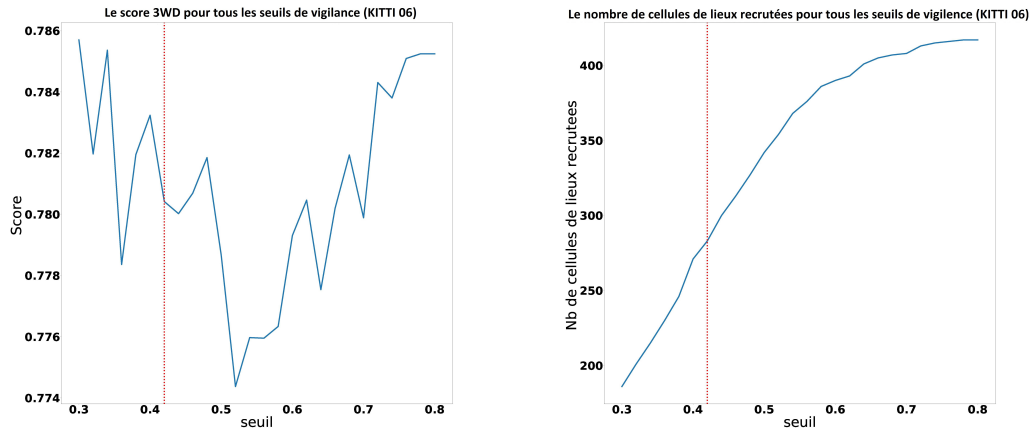


FIGURE 4.13 – Le score 3WD et le nombre de cellules de lieu recrutées pour l’ensemble des seuils de vigilance pour la base de données KITTI 06.

lieu ont bien des réponses localisées : leur activité est élevée dans le voisinage du lieu d’apprentissage et devient négligeable ailleurs. Ce ratio gagnant sur bruit augmente en même temps que la valeur du seuil de vigilance. Avec un seuil de vigilance élevé, un plus grand nombre de cellules de lieu sont recrutées. Même si ces dernières définissent des champs de lieu de plus petites tailles et que donc plus de cellules non gagnantes ont une activité élevée, le fait que leurs réponses restent localisées et qu’il y ait plus de cellules de lieu recrutées, a pour effet que plus de cellules avec de faibles activités servent au calcul du bruit moyen, qui diminue, et donc le ratio gagnant sur bruit augmente. La contrepartie est un coût computationnel plus élevé.

La valeur du score 3WD doit, comme expliqué lors de la description des métriques (section 4.2.3), idéalement être proche de 1. Cette valeur indique que la position actuelle du véhicule est bien encadrée par les trois cellules de lieu ayant les activités les plus élevées. Cette condition est globalement vérifiée indiquant que les erreurs de reconnaissances des cellules de lieu restent généralement circonscrites aux cellules de lieu proches de la position réelle du véhicule. Par conséquent, le nombre d’erreur dans la reconnaissance séquentielle des cellules de lieu ne donne une indication que sur une imprécision locale, la cellule $n+1$ répondant avant la cellule n par exemple, et qu’il y a très peu de réponses de cellules totalement incohérentes.

La précision de la reconnaissance est globalement bonne : le nombre d’erreurs de reconnaissances reste plutôt faible. Dans la plupart des cas la valeur de seuil de 0.42 pour le seuil de vigilance permet d’obtenir le plus faible nombre d’erreurs.

Le score F1 est élevé et assez proche de 1 sur l’ensemble des trajectoires comportant un rebouclage. La figure 4.16 présente les courbes de précision-rappel pour les trois principales valeurs de vigilance. Elle montre que le niveau de vigilance n’a que peu d’influence sur la reconnaissance des fermetures de boucles. Cela signifie que

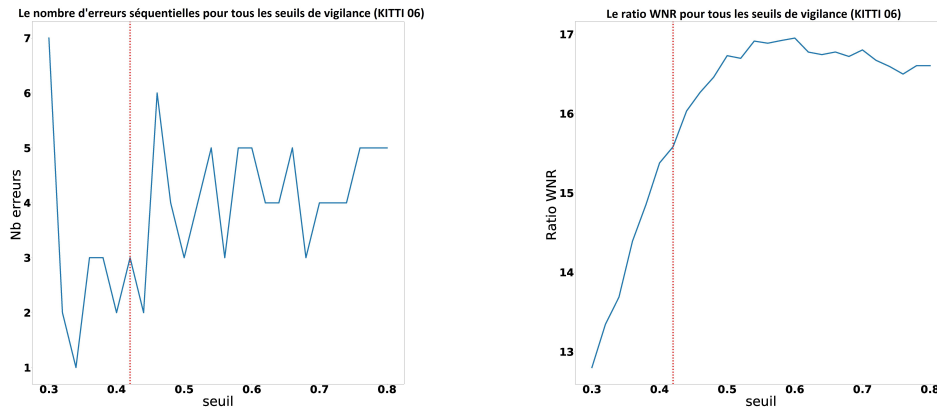


FIGURE 4.14 – Le nombre d’erreurs séquentielles et le ratio WNR pour l’ensemble des seuils de vigilance pour la base de données KITTI 06.

pour les différents seuils, on génère des cellules capables à la fois de discriminer des lieux mais avec une tolérance suffisante (capacité de généralisation) pour reconnaître des lieux soumis à de légères perturbations (léger déplacement latéral, changement de luminosité).

Pour avoir un point de comparaison avec des modèles « classiques », une implémentation de FabMap 2.0 [Cummins & Newman 2009] qui reste une référence en matière de reconnaissance de lieu visuelle a été utilisée. Cette implémentation existante sous OpenCV 2.4 a été testée sur les bases KITTI 00, KITTI 06 et KITTI 07 afin de comparer la capacité de précision-rappel. Cette comparaison n’est pas parfaite pour deux raisons. Le modèle a été testé avec le vocabulaire fourni par OpenCV pour faire tourner FabMap. Comme expliqué lors de la description du score F1 en partie 4.2.3, pour prendre en compte la capacité de généralisation du modèle lors du calcul de la précision et du rappel, les vérités terrains ont été construites en segmentant les trajectoires en lieux faisant le même nombre d’images. Cela peut entraîner une dégradation des performances du modèle dont la réponse se compose de lieux qui peuvent avoir des tailles différentes. Malgré cela, cette comparaison (voir tableau 4.6) permet de donner un ordre d’idée. Les résultats de score F1 montrent des performances similaires entre les deux modèles indiquant que notre modèle, sur la reconnaissance de fermeture de boucle est au niveau d’un modèle plus classique. Ceci, plus les résultats précédents montrant que l’on obtient une représentation de l’espace pertinente (peu d’erreurs de reconnaissance) permet d’affirmer que le modèle de cellules de lieu visuelles, initialement conçu pour de la localisation en intérieur, malgré un coût computationnel élevé, est capable d’avoir des performances honorables comparé à des modèles plus classiques.

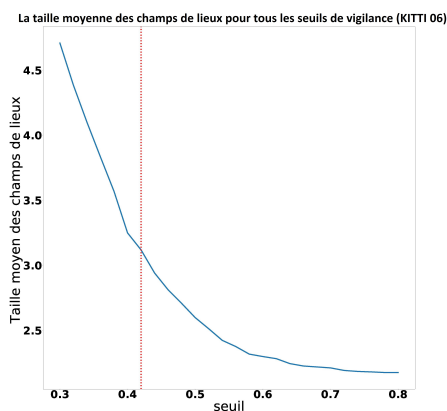


FIGURE 4.15 – La taille des champs de lieu pour l’ensemble des seuils de vigilance pour la base de données KITTI 06.

base	modèle ETIS	FabMap
KITTI 00	0.9999	0.9992
KITTI 06	0.9997	0.9977
KITTI 07	0.9997	0.9963

TABLE 4.6 – le score F1 pour les modèles PerAc et FabMap .

4.3 Discussion

Les résultats présentés au chapitre précédent permettent de mettre en lumière les point forts et points faibles de l’architecture. Une information manquante est une comparaison avec d’autres modèles bio-inspirés. Après une comparaison avec les modèles bio-inspirés présentés dans l’état de l’art du chapitre 2 cette section discutera des limites actuelles du modèles à dépasser en priorité en indiquant un certain nombre de pistes à suivre pour y parvenir. Elle proposera enfin des perspectives d’évolution de ce modèle à plus long termes.

4.3.1 Comparaison à l’état de l’art bio-inspiré

Une comparaison avec d’autres modèles est ardue car, entre les modèles « classiques » et les modèles bio-inspirés, les critères d’évaluation sont différents et même au sein des modèles bio-inspirés les critères de mesure ne sont pas unifiés. Cependant à partir des modèles bio-inspirés et neurorobotiques présentés précédemment dans l’état de l’art (section 3.3) on peut faire une comparaison qualitative (voir tableau 4.7) et une estimation des coûts computationnels (voir tableau 4.8) des modèles de l’état de l’art.

En termes qualitatif (table 4.7) on peut observer une certaine diversité même si

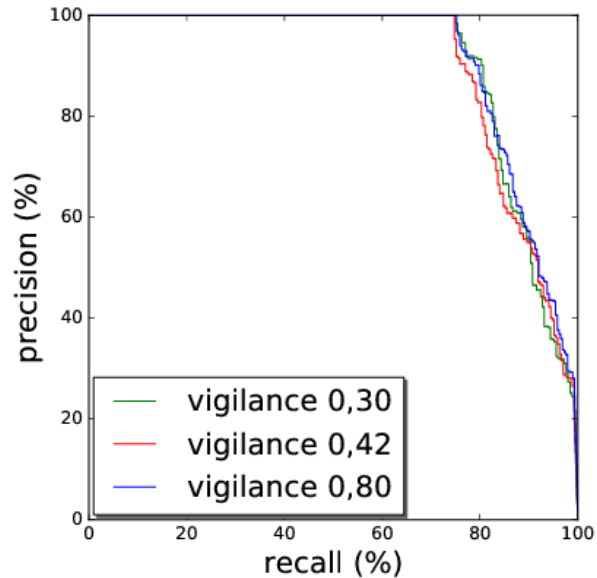


FIGURE 4.16 – Base de données KITTI 06 : la précision et le rappel pour les trois vigilances 0.30 0.42 et 0.80

ils ont des points communs. Les modèles utilisent peu de capteurs, à l'exception de Beobot 2.0 qui en utilise cinq types. Ils peuvent se baser sur des descripteurs d'image différents (globaux ou locaux) qui obtiennent des performances similaires (voir table 4.8) et utilisent des structures d'encodage différent dont la forme neuronale qui à l'avantage de pouvoir avoir une représentation unifiée et donc de pouvoir faire plus aisément de la fusion de données ou de la communication entre sous systèmes (pour de la planification de trajectoire en construisant des cartes cognitives basées sur la reconnaissance de lieu par exemple).

La description quantitative (voir table 4.8) montre que les différents modèles ont un coût mémoire similaire, à l'exception de Beobot 2.0, le robot se déplaçant parmi une foule de piétons, qui est plus léger en calcul. En contrepartie ce modèle utilise des informations a priori (une carte topologique lui est fournie) le rendant peu adaptable à de nouveaux environnements si on ne lui fournit pas une nouvelle carte.

Les autres obtiennent un ratio similaire (entre 0.18 et 0.42) avec un léger avantage pour les modèles de Rebai (0.195) et Zeng et Si (0.186). Le modèle de Rebai a pour limitation d'utiliser des fenêtres attentionnelles se concentrant sur des parties d'une image qui sont paramétrées de manière arbitraire. Le modèle par Zeng and Si reposent fortement sur l'intégration de chemin et nécessite donc des recalages fréquents. Notre modèle, PerAC, est dans la moyenne haute des modèles de l'état de l'art, mais il n'a pas été testé en ligne contrairement à RatSLAM et Beobot. Notre modèle a l'avantage de faire partie d'un modèle global de localisation et navigation n'ayant pas besoin d'avoir une représentation cartésienne. Il a l'inconvénient, dans son implémentation actuelle, de ne pas avoir de système d'allocation dynamique de

Modèle	Ccapteurs	Structure	Gestion des données	Encodage	Descripteurs visuels
Rebai <i>et al.</i> [Rebai <i>et al.</i> 2012]	Caméra	Hiéarchique	Sync.	Symbolique	Global
Beobot 2.0 [Christian <i>et al.</i> 2010]	Caméra, roue codeuse, capteur LRF, centrale inertielle	Publieur/ souscripteur hiéarchique	Async.	Symbolique	local et global
RatSLAM [Milford & Wyeth 2008]	Caméra	Distribué	Sync.	Symbolique	Global
Zeng and Si [Zeng & Si 2017]	Caméra	Distribué	Sync.	Réseaux de neurones	Global
PerAC [Espada <i>et al.</i> 2018]	Caméra, compas	Distribué	Sync.	Réseaux de neurones	Local

TABLE 4.7 – Description qualitative des différentes approches de l'état de l'art en fonction de critères concernant le type de capteurs utilisés les structures et encodages utilisés et les types de descripteurs utilisés.

Modèle	Informations a priori	Longueur du trajet	Type d'environnement	Type d'expérience	Coût mémoire (node/m)	Précision
Rebai <i>et al.</i> [Rebai <i>et al.</i> 2012]	Non	66 km	Banlieue	Hors ligne	0.195	10 m
Beobot 2.0 [Christian <i>et al.</i> 2010]	oui	511.25 m 615.48 m 401.16 m	allée piétonne	En ligne	0.013 0.008 0.015	60 cm
RatSLAM [Milford & Wyeth 2008]	Non	66 km	Banlieue	En ligne	0.414	10 m
Zeng and Si [Zeng & Si 2017]	Non	66 km	Banlieue	Hors ligne	0.186	Inconnu
PerAC [Espada <i>et al.</i> 2018]	Non	3721 m 2450 m 2203 m	Centre ville Autoroute banlieue	Hors ligne	0.40 0.05 0.40	0.83 m 18.14 m 0.95 m

TABLE 4.8 – Comparaison quantitative des différentes approches en fonction d'un certain nombre de critères .

neurones et donc de devoir garder en mémoire l'ensemble des neurones qui vont être potentiellement utilisés. Cela a pour effet de nécessiter un coût en mémoire vive important, qui augmente sur de longues distances, ralentissant le calcul et empêchant, pour l'instant, les tests en ligne. Le tableau 4.9 présente les mesures de coût computationnel qui illustrent ce problème de coût mémoire rencontré par le système.

Le coût computationnel est la limite la plus importante de ce modèle comme l'illustre le tableau 4.8. Il est en partie lié au fonctionnement du simulateur qui ne permet pas pour l'instant de faire de l'allocation dynamique. Ce choix correspond à la volonté de se rapprocher de la modélisation du cerveau dont le nombre de neurones varie peu et où chaque neurone conserve toujours une activité même très faible. Cela oblige donc à faire une estimation surévaluée du nombre de cellules de lieu et de signatures lors d'une expérience pour allouer suffisamment de neurones dans le modèle avant le lancement d'une simulation. Cela pose d'autant plus problème que,

Base de données	Nb d'images	Nb de PC allouéss	Nb d'amers recrutés	Nb total de neurones	Utilisation de la RAM (Go)	taille sur le disque (Go)	Durée de l'expérience
KITTI 00	4541	4541	45410	558131	42,5	2,8	12h 30
KITTI 01	1101	700	18000	221527	4,8	0,5	1h 30
KITTI 06	1101	600	11010	137447	4,4	0,5	1h 30
KITTI 07	1101	600	11010	137447	4,2	0,5	1h 30

TABLE 4.9 – Les mesures de coût computationnel par des mesures de temps d'expérience, d'utilisation de mémoire vive ou de mémoire de stockage tout en indiquant le nombre de neurones alloués en début d'expérience

pour l'instant encore, l'ensemble des neurones alloués lors d'une simulation, restent constamment chargés dans la mémoire vive.

Du point de vue du modèle ce coût mémoire est essentiellement dû aux signatures pour deux raisons. La première vient du nombre de signatures apprises. Pour l'instant l'apprentissage des cellules de lieu est couplé, ce qui a pour conséquence que lors de l'apprentissage d'une nouvelle cellule de lieu, l'apprentissage d'un ensemble de signatures est aussi réalisé. Il peut donc arriver que deux signatures très proches soient encodée plusieurs fois, ce qui, en plus d'augmenter le coût computationnel, peut altérer la reconnaissance de cellules de lieu ayant des signatures similaires.

Le poids des signatures pour le coût mémoire vient aussi de l'encodage des signatures. A l'heure actuelle, un neurone est utilisé pour encoder chaque signature. En utilisant une autre forme de codage, il peut être possible d'obtenir un encodage qui aura une meilleure combinatoire et donc utiliser moins de neurones pour encoder plus de signatures. Cela peut se faire en utilisant un code où peu de neurones parmi la population servent à encoder une entrée : un codage épars. D'après des études récentes [Beyeler *et al.* 2019], différentes structures cérébrales seraient susceptibles d'utiliser des stratégies de codage épars. Par exemple, les observations du cortex visuel primaire indiquent que les cellules simples (voir section 3.2.2.1) pourraient construire un tel codage. La construction d'un codage épars est une question activement recherchée du fait de ces possibles applications dans des domaines variés (traitement d'image, acoustique, traitement du signal). [Zhang *et al.* 2015] propose un état de l'art récent de la représentation épars, des stratégies de génération épars et de ces applications. L'anti-hebb [Földiák 1990] est un exemple d'algorithme de construction de code épars, basé sur une représentation neuronale qui pourrait être utilisée.

Par ailleurs les signatures enregistrent directement les vues locales en utilisant un lien par pixel. Ainsi une vue locale projetée en log polaire sur une image de 16x16 pixels sera encodée par 256 poids synaptiques codés en flottant. Il doit être possible d'encoder ces images de manière plus compacte à l'aide de descripteurs, comme ceux présentés au chapitre 1, ou conçu en s'inspirant des traitements effectués par le cortex visuel.

L'autre limite du modèle actuel réside dans le fait que la pertinence des points n'est déduite que de l'information de saillance. Cela peut poser des problèmes car ce critère de saillance n'est pas suffisant pour désigner un amer stable pour la construc-

tion de cellules de lieu. Comme l'illustre l'expérience Scenic, des points d'intérêt peuvent être pris sur des objets mobiles. Dans le pire des cas on pourrait se retrouver avec une cellule de lieu construite uniquement sur des amers non fiables et donc inexploitable. Une solution possible pourrait être d'incorporer la dimension temporelle, par utilisation du flux optique par exemple, afin d'exclure les points d'intérêt en mouvement. Une autre solution, plus centrée sur la gestion des objets mobiles, pourrait être d'exploiter des fonctions de plus haut niveau comme la reconnaissance et la classification d'objet afin d'exclure des régions de l'image lors de la recherche de points d'intérêt. Cela permettrait par exemple de ne pas apprendre d'amer sur un objet fixe lors de l'apprentissage mais potentiellement mobile comme une voiture garée par exemple [Lu *et al.* 2020].

Une dernière limite du modèle provient de l'apprentissage en un coup. Si il a le gros avantage de permettre un apprentissage rapide, il ne permet pas de remettre en cause les apprentissages réalisés. On peut donc se retrouver dans une situation où est appris un amer qui ne sera par la suite jamais reconnu.

4.3.2 Perspectives

Malgré ces contraintes, le modèle de cellules de lieu visuelle est capable de construire une représentation de l'environnement permettant de se localiser. Le mécanisme de vigilance basé sur un seuil minimum de réponse des cellules de lieu a montré, au cours des expériences présentées précédemment, sa capacité à recruter correctement des cellules de lieu et à s'adapter aux différents types d'environnements. Même si un seuil fixe permet d'obtenir un apprentissage suffisant dans la plupart des cas, l'expérience sur le trajet KITTI 01 montre que, dans certaines conditions, un seuil différent serait plus optimal.

De plus la précision requise pour la localisation dépend de la tâche. Dans un environnement urbain, il y a de nombreuses intersections. Par ailleurs, entre ces intersections, suivant si l'objectif est d'atteindre la prochaine intersection ou de rejoindre une adresse précise la précision requise n'est pas la même. Sur autoroute, hormis pour les bifurcations et sorties d'autoroutes, si elle est utilisée conjointement à des stratégies de suivi de route, la localisation devient secondaire et la représentation peut être plus grossière en considérant que les voies de circulation sont plus larges. La précision n'est utile qu'au niveau des bifurcations. Ces deux cas illustrent le gain potentiel en coût mémoire en modulant la précision en fonction de contexte de conduite.

La forme des champs de lieu est influencée par le système visuel et plus précisément par la largeur du champ visuel. Un champ visuel large donne des champs de lieu omnidirectionnels alors qu'un champ visuel plus réduit donne des champs de lieu orientés. Comme précédemment expliqué en section 3.2.3.1 cela correspond aux observations recueillies par les chercheurs en neurosciences. Dans le cas des expériences sur véhicules, avec une caméra fixe pointée vers l'avant, nos cellules de lieu produisent des champs de lieu orientés avec un angle d'ouverture limité. Elles sont donc similaires aux cellules de lieu orientées observées chez les primates. Dans de

précédents travaux, sur une itération précédente du modèle, des expériences réalisées en intérieurs avec des robots mobiles équipés, soit de caméras montées sur une base rotative pour effectuer des panoramas, soit de caméras omnidirectionnelles, ont obtenu des cellules de lieu qui formaient des champs de lieu moins orientés et donc plus proches des cellules de lieu observées chez le rat [Gaussier *et al.* 2002b]. Utiliser ce type de caméra sur un véhicule pourrait permettre d'obtenir des cellules de lieu avec des champs de lieu omnidirectionnels mais aussi plus larges, notamment dans des environnements urbains ou péri-urbains, et donc possiblement d'obtenir de meilleures performances de reconnaissance en contrepartie d'un coût de calcul plus important du fait du surplus de données à traiter. En effet, grâce à une caméra omnidirectionnelle, les amers proches sont reconnus pendant plus de temps, d'où la meilleure reconnaissance en milieu urbain. Par conséquent, en fonction de la taille du champ visuel utilisée par le système, une même valeur de seuil pour la vigilance pourrait donner des résultats bien différents au niveau de la granularité de la couverture de l'ensemble des champs de lieu. Ceci aurait pour conséquence d'affecter la précision de la localisation.

Dans la version actuelle du modèle, on n'utilise pas les réponses du système au maximum de leur potentiel. Le système nous donne une réponse analogique allant de 0 à 1 pour toutes les cellules de lieu recrutées. On ne garde que celle avec la plus grande activité pour se localiser. Par conséquent le système permet de nous indiquer dans quel champ de lieu l'on se trouve mais est incapable de nous situer dans ce champ de lieu. Notre précision dépend uniquement de la taille de nos champs de lieu. Il pourrait être intéressant de prendre les réponses des trois cellules de lieu avec les plus fortes activités, ou bien les réponses de la cellule gagnante et des cellules de lieu précédentes et suivantes. En utilisant ces réponses, on pourrait déterminer plus finement la position du véhicule en effectuant une forme de triangulation. Grâce à cette triangulation, on pourrait déterminer si le véhicule se situe à l'endroit de l'apprentissage de la cellule gagnante, avant ou après en fonction de l'activité des deux autres cellules de lieu.

Le modèle actuel n'utilise que des descripteurs locaux grâce à la chaîne d'attention visuelle qui extrait des vues locales. L'étude du système visuel des animaux montre qu'en plus de descripteurs locaux (la vision fovéale), des descripteurs globaux sont utilisés (la vision périphérique). Les descripteur globaux, en n'ayant pas de phase de recherche de points d'intérêt, peuvent permettre d'identifier des lieux plus rapidement, même si, en contrepartie, cette identification est potentiellement plus grossière. Une hypothèse courante est que des animaux utilisent ces deux traitements en parallèle pour faire une localisation grossière grâce aux descripteurs globaux, puis affiner cette localisation par l'information fournie par les descripteurs locaux. Ce principe a déjà été utilisé dans plusieurs modèles [Christian *et al.* 2010] [Belkaid *et al.* 2016]

Enfin le modèle pourrait bénéficier de l'exploitation de la dimension temporelle. Cette dernière pourrait permettre de faire un filtrage des amers qui, en plus d'exclure les objets mobiles évoqués précédemment, serait capable de différencier les amers proches, visibles sur une fenêtre de temps courte, des amers éloignés, visibles plus

longtemps. Ceci permettrait un plus grand contrôle de la précision des cellules de lieu obtenues. Une autre utilité de la dimension temporelle et de pouvoir travailler sur des séquences et donc de faire de la prédiction sur l'apparition de cellules et d'amers déjà appris et donc aussi faire de l'autoévaluation. Cette incorporation de la dimension temporelle pourrait être réalisée par l'intermédiaire de la mémoire de travail en changeant son mécanisme de réinitialisation (à chaque image pour l'instant).

Modèle PerAc

Sommaire

5.1	Description du modèle	93
5.1.1	Principe	94
5.1.2	Contrôle neuronal	95
5.1.3	Implémentation	98
5.1.4	Adaptation au véhicule	99
5.2	Expériences	103
5.2.1	Dispositif expérimental	104
5.2.2	Résultats	105
5.3	Discussion	108

Le chapitre 4 a montré qu'un modèle neurorobotique de reconnaissance de lieu visuelle initialement conçu pour fonctionner en environnement intérieur est capable de fonctionner avec des performances acceptables en environnement extérieur de grande taille.

Ce problème de la reconnaissance visuelle de lieu en environnement extérieur, même s'il est un vaste champ de recherche en soi, n'est qu'un sous problème de la question plus large de la navigation en extérieur.

Le modèle de cellules de lieu visuelles est en fait une partie d'un modèle de navigation conçu au sein du laboratoire ETIS, le modèle PerAc [Gaussier & Zrehen 1995]. Dans ce chapitre sera présenté dans son ensemble ce modèle de navigation et son test en conditions réelles. Il commencera par la description du modèle en détaillant son principe, son implémentation initiale et son adaptation au contrôle d'un véhicule automobile. Cette adaptation est nécessaire, car la cinématique d'une automobile, ayant une contrainte dans sa capacité de rotation, du fait de l'existence d'un rayon de braquage diffère de celle de robots à système différentiel, donc capable de tourner sur eux-mêmes. Ce chapitre décrira par la suite la série d'expériences servant à attester de la portabilité de ce modèle à la navigation de véhicule en milieu extérieur avant de terminer en exposant et critiquant les résultats.

5.1 Description du modèle

Le modèle PerAc (Perception-Action) est une architecture neuronale permettant l'apprentissage d'associations sensorimotrices composée de deux voies. La première,

dite réflexe, relie une perception de l'environnement à une action. La seconde voie se sert d'un mécanisme cognitif afin de reconnaître et de catégoriser les informations issues des capteurs. Un système de conditionnement lie les deux voies permettant d'obtenir la réalisation d'une action donnée lors de la reconnaissance d'une perception. Cette architecture générique peut se décliner pour faire apprendre des tâches à un robot manipulateur [de Rengerve *et al.* 2015].

Dans le cadre de la navigation, le modèle PerAc devient un modèle de navigation réactive reposant sur une représentation du monde qui n'est pas une carte. Cette représentation de l'environnement prend la forme d'une collection de lieux sans lien entre eux, mais où chaque lieu est associé à une action. La conséquence est un système réactif où la reconnaissance d'un lieu implique la réalisation de l'action associée.



FIGURE 5.1 – Architecture du modèle PerAc. Le modèle de cellules de lieu visuelles recrute des lieux qui sont ensuite associés à des actions par une seconde partie du système. Une troisième partie s'appuie sur cette représentation lieu/action pour sélectionner l'action à mener et génère la consigne associée.

Le modèle PerAc peut être découpé en trois parties (voir figure 5.1). Le modèle de reconnaissance de lieu visuelle présenté au chapitre 4 décompose l'environnement en une collection de lieux. Un second système s'appuie sur cette représentation en attribuant une action de déplacement à chaque lieu défini par la représentation. Enfin la dernière partie du système, à partir des différents lieux reconnus, fait une sélection de l'action parmi celles possibles et élabore les consignes motrices à l'aide d'un substrat neuronal. Ce chapitre va se concentrer sur ce second système en en décrivant tout d'abord le principe puis ensuite son application sur véhicule.

5.1.1 Principe

Le principe du système d'association lieu/action est, comme son nom l'indique, d'associer un lieu à une action. Cette association se fait par conditionnement Hebbien [Gaussier & Zrehen 1995]. La théorie de Hebb peut être résumée par la formule : des neurones qui s'excitent ensemble se lient entre eux. Dans notre cas, cela consiste à créer un lien synaptique entre le neurone encodant le lieu courant et le neurone encodant la direction courante lors de l'apprentissage afin qu'ensuite l'activation de la cellule de lieu provoque l'activation du neurone encodant la direction à suivre. Ce que l'on appelle un lieu est une portion de l'espace que l'on peut discriminer du reste de l'environnement. Il est construit par le modèle de cellules de lieu visuelles

présenté dans le chapitre précédent. L'action est généralement une direction que l'agent va suivre. Par un certain nombre d'associations il est possible d'apprendre des tâches de navigation différentes. Le principe est de considérer le comportement du robot comme un système dynamique dans lequel les associations lieu/action viennent déformer l'espace sensorimoteur selon l'action à obtenir.

En apprenant des lieux autour d'une position cible et en associant pour chacun de ces lieux la direction vers la position cible on obtient dans l'espace sensorimoteur un bassin d'attraction qui se traduit par un comportement de homing (voir figure 5.2).

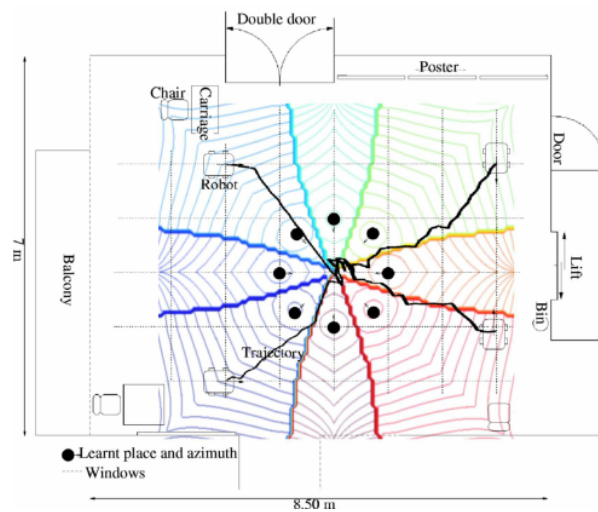


FIGURE 5.2 – Apprentissage de huit lieux autour d'une position but permettant de réaliser des tâches de homing. Dans cet exemple, les huit cellules de lieu permettent de couvrir l'ensemble de l'espace grâce à leur champs de lieu. Chaque lieu est associé à une direction pointant vers la position cible. En suivant la direction correspondant au lieu courant, le robot ira soit vers le centre soit entrera dans un autre champ de lieu et changera donc de direction à suivre. Par une succession de changements de directions, le robot convergera vers le centre, réalisant ainsi le homing. On obtient l'équivalent d'un puits dans un champ de potentiel.

Cette association de lieu et action permet aussi d'apprendre des trajectoires en apprenant une succession d'association de lieu et d'action de part et d'autre de la trajectoire qui forme un sillon dans l'espace sensorimoteur (voir figure 5.3).

Les actions que l'on associe aux lieux sont des directions qui sont encodées sur une couche neuronale particulière, un champ de neurones dynamique, décrivant les 360 degrés de l'environnement.

5.1.2 Contrôle neuronal

Le modèle de cellules de lieu donne en sortie la réponse analogique de chacune des cellules de lieu apprises. Si on appliquait un mécanisme de compétition stricte de

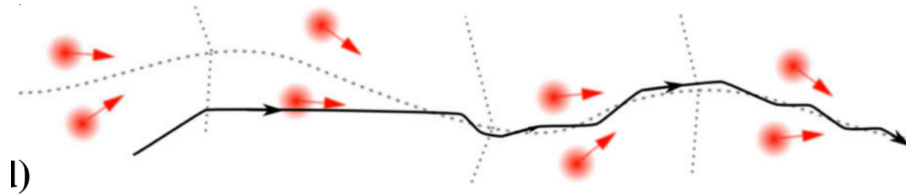


FIGURE 5.3 – Test de l'apprentissage d'une trajectoire cible. La trajectoire cible , en pointillée, est obtenue par un ensemble d'associations lieu/action, point et flèches rouge. La trajectoire de test est dessinée en noir.

type WTA on obtiendrait un comportement de navigation brusque avec de possibles changements brutaux et répétés de directions à suivre du fait de cellules de lieu dont les activités d'un niveau similaire entraînerait des alternances dans l'ordre des gagnants. Le modèle de champs de neurones dynamiques [Amari 1977] possède des propriétés de filtrage permettant d'obtenir une forme de compétition douce. Un champ de neurones dynamique est composé de neurones dont chacun dispose de connexions excitatrices le reliant à ses voisins les plus proches et de connexions inhibitrices le reliant aux neurones plus éloignés (voir figure 5.4) Ces connexions sont définies par un noyau d'interaction qui est construit par une différence de gaussienne. Chaque neurone comporte aussi un lien récurrent produisant un lissage temporel permettant un filtrage contre le bruit ou les oscillations en entrée du réseau.

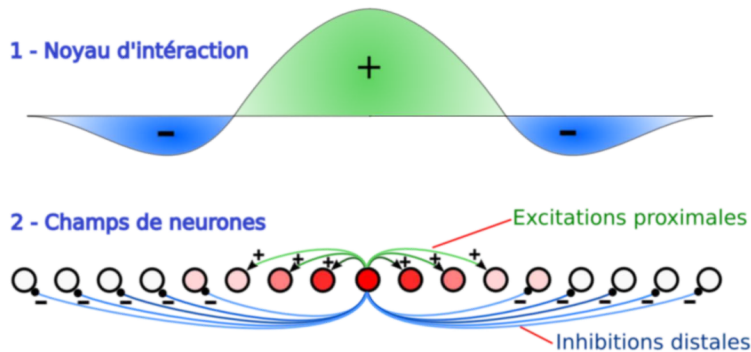


FIGURE 5.4 – 1 - Le noyau d'interaction définit à partir d'une différence de gaussienne avec le centre excitateur et la périphérie inhibitrice. 2 - Exemple de connexions latérales pour un neurone lui permettant d'exciter ces voisins proches et d'inhiber ses voisins plus éloignés. [Jauffret 2014]

La dynamique du champ neuronal est modélisée par l'équation 5.1.2.

$$\tau \times \frac{u(x, t)}{dt} = u(x, t) + I(x, t) + h + \int_{z \in V_x} \omega(z) \times f(x - z, t) dz \quad (5.1)$$

Où $u(x, t)$ est le potentiel d'un neurone x au temps t . $I(x, t)$ est l'entrée du système. h est une constante négative assurant la stabilité du champ. τ est le taux de relaxation du système. $\omega(z)$ est le noyau d'interaction définissant les connexions latérales des neurones. $f(x - z, t)$ est la fonction d'activation des neurones. V_x est l'intervalle d'interaction définissant le voisinage. Cet ensemble de connexions entre les neurones donne au champ neuronal des propriétés de compétition globale et de coopération locale (voir figure 5.5) résultant en une compétition douce.

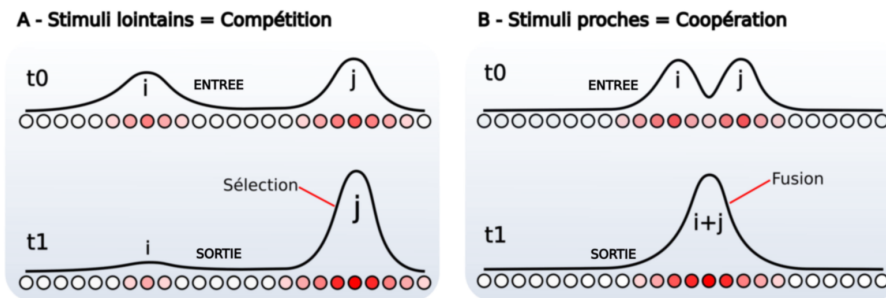


FIGURE 5.5 – Exemple de propriétés de coopération et de compétition d'un champ de neurones dynamiques. A - Compétition : deux stimuli sont éloignés et donc s'inhibent mutuellement jusqu'à ce que l'un d'eux disparaisse permettant au stimuli restant de se renforcer par son lien récurrent. B - Coopération : deux stimuli proches s'excitent mutuellement et finissent par converger vers une réponse correspondant à la moyenne des deux stimuli [Jauffret 2014].

En utilisant un champ neuronal d'Amari sur lequel on projette les directions associées aux cellules de lieu les plus actives, après convergence, on obtient une réponse unique. Cette réponse peut être, soit une des directions projetées, soit un compromis entre deux directions proches. La sélection de l'action à effectuer a été obtenue simplement grâce aux propriétés du champ neuronal.

Il est possible à partir des réponses de ce type de champs neuronaux de générer un contrôle moteur dynamique [Schöner *et al.* 1995]. Ce contrôle, issu de la théorie des champs dynamiques, proposé par Schöner, repose sur la construction d'attracteurs et de répulseurs (voir figure 5.6) en utilisant la continuité des bulles d'activités du champ neuronal comme un gradient que le système peut remonter. La dérivée locale du champ d'activité est considérée comme une sortie motrice.

Le modèle PerAc implémente une version de cette théorie des champs dynamiques dans le cas de robots évoluant dans un espace à 2 dimensions, donc commandés en vitesse linéaire et angulaire. La vitesse linéaire est soit fixe, soit modulée négativement par la proximité des obstacles. La vitesse angulaire dépend de motivations propres au système qui peuvent être attractives, un but à suivre, soit répulsives, un obstacle à éviter (voir figure 5.7). Le modèle utilise deux champs de neurones dynamiques dont chacun encode les 360 degrés autour du robot que l'on nomme l'espace péricorporel. Sur l'un d'eux sont projetés les obstacles à éviter tandis que l'autre encode les directions à suivre. Par sommation de ces deux champs

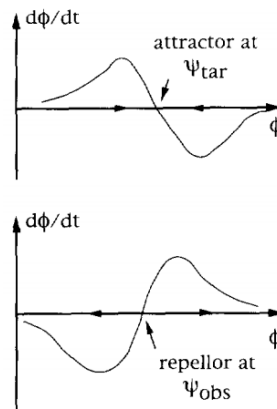


FIGURE 5.6 – Exemple d’attracteur (en haut) et de répulseur (en bas) obtenu par la dérivée d’une gaussienne positive pour l’attracteur et négative pour le répulseur. Lorsque l’on se décale à gauche du point attracteur $\frac{d\phi}{dt}$ est positif donc ϕ augmente et se décale vers la droite sur la courbe. Lorsque l’on se décale à droite de l’attracteur $\frac{d\phi}{dt}$ est négatif donc ϕ diminue et se décale vers la gauche. Il se passe le phénomène inverse pour le répulseur, un décalage à gauche par rapport au point répulseur entraîne un décalage vers la gauche ($\frac{d\phi}{dt}$ négatif), un décalage vers la droite entraîne un décalage vers la droite ($\frac{d\phi}{dt}$ positif) [Schöner *et al.* 1995].

neuronaux on obtient un troisième champ que l’on appelle champ motivationnel du robot. Par une dérivée du premier ordre on obtient un ensemble d’attracteurs et de répulseurs donnant la consigne de vitesse angulaire à appliquer pour chaque orientation possible du robot. Il suffit de lire sur ce tracé de dérivée la valeur analogique correspondant à l’orientation courante du robot afin d’obtenir la consigne de vitesse angulaire à appliquer.

L’utilisation des champs de neurones dynamiques pour le contrôle présente trois avantages. La propriété du champ, en plus de permettre un filtrage du bruit, permet très simplement de faire de la sélection d’action et le contrôle moteur est intuitif, correspondant à une simple dérivée de l’orientation cible.

5.1.3 Implémentation

En pratique le modèle PerAc a été implémenté sur des robots d’intérieurs (voir figure 5.8) à système différentiel et munis d’une série de capteurs (caméra, capteurs de proximité, compas magnétique) ainsi que d’une laisse. Cette laisse est utilisée pour réaliser un apprentissage proscriptif d’une trajectoire. Ce dernier consiste à laisser le robot naviguer librement et le ramener vers la trajectoire lorsqu’il s’en éloigne trop. Dans l’espace sensorimoteur, cela revient à ramener le robot dans le bassin d’attraction en formant une nouvelle pente menant vers la trajectoire via l’apprentissage. Un apprentissage se déroule de la manière suivante :

- le robot circule librement tant qu’il suit la trajectoire
- lorsque le robot s’éloigne de la trajectoire un utilisateur vient tirer sur la

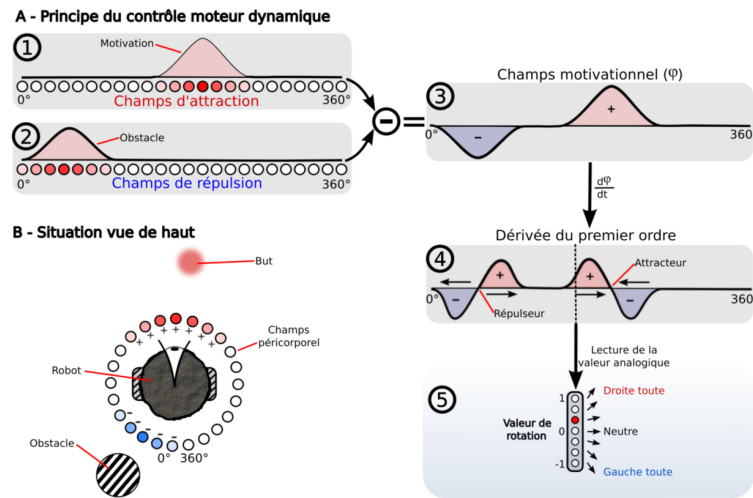


FIGURE 5.7 – Le contrôle moteur se basant sur la théorie des champs de neurones dynamiques. A - Les champs codant les buts (1) et les obstacles (2) sont sommés pour former le champ motivationnel (3) dont la dérivée donne la vitesse angulaire pour chaque orientation du robot (4) dont il suffit de lire la valeur analogique pour l'orientation courante du robot (5). B - Vue de haut représentant l'espace péricorporel du robot où chaque neurone encode un angle particulier. Une valeur de neurone positive correspond à un but tandis qu'une valeur négative réfère à un obstacle.

laisse. Ce geste indique au système qu'il doit apprendre ou corriger une association

- le robot s'arrête puis s'oriente dans la direction imposée par la laisse
- Si l'écart entre la direction désirée indiquée par la laisse et celle précédente est supérieure à un seuil alors un nouveau lieu est appris. Sinon le mouvement associé à la cellule gagnante est corrigé.
- en répétant ce processus on définit une trajectoire

On obtient une dynamique d'apprentissage proscriptive où le robot apprend des associations lieux actions de part et d'autre de la trajectoire souhaitée. Le résultat est un apprentissage de trajectoire robuste aux déplacements latéraux et aux contournements d'obstacles où quand le robot commence à s'éloigner de la trajectoire il se retrouve dans un lieu associé à une action redirigeant le robot vers la trajectoire.

5.1.4 Adaptation au véhicule

L'adaptation du modèle sur un véhicule automobile a entraîné un certain nombre de changements et de difficultés à résoudre.

Une première difficulté est d'ordre purement technique. Le véhicule utilisé est une Renault ZOE modifiée appartenant à l'institut VEDECOM, déjà présentée au chapitre 4. Les capteurs et actionneurs de ce véhicule sont contrôlés par un middle-



FIGURE 5.8 – Un robot Robulab du laboratoire ETIS sur lequel est implémenté le modèle PerAc.

ware appelé RTmaps. Ce dernier se présente sous la forme de diagrammes constitués de blocs, encapsulant des fonctions, et de liens entre les blocs. Le tout fonctionne sur un ordinateur embarqué sous système d'exploitation Windows. Le modèle PerAc, lui, est implémenté sous Promethe, un middleware qui se présente lui aussi sous forme de diagrammes mais modélise des neurones et fonctionne sous un système d'exploitation Linux.

Afin de pouvoir appliquer le modèle PerAc, il convenait de pouvoir faire communiquer les deux systèmes. Ce problème a été résolu en utilisant un script python qui fait l'interface entre le diagramme RTMaps et le script Promethe par l'intermédiaire d'un nœud TCP (voir figure 5.9).

L'inconvénient de cette méthode est qu'elle entraîne un délai supplémentaire dans la transmission des données à un système incapable de fonctionner en temps réel dans son implémentation actuelle (se référer au tableau 5.2). Le temps de calcul a pu être diminué en donnant des contraintes au modèle. En restreignant le nombre de cellules de lieu et d'amers mémorisables il a été possible d'obtenir des temps de calcul suffisamment faibles pour une première série de tests. Le tableau 5.1.4 donne les performances en termes de temps d'exécution et de fréquence d'échantillonnage obtenue par deux machines différentes pour deux paramètres du nombre d'amers

Machine	Nb de lieux	Nb d'amers	Temps d'exécution total (s)	Nb d'images par secondes
ordinateur 1	200	6000	287.66	2.14
ordinateur 1	100	1500	103.0	6.0
ordinateur 2	200	6000	199.58	3.09
ordinateur 2	100	1500	67.9	9.1

TABLE 5.1 – Les résultats de mesure de performances sur deux ordinateurs. La première machine (ordinateur 1) intègre un processeur Intel i5-6200U 2 coeurs 4 Threads @2.3 GHz, 2.8 GHz en Turbo avec 8Go de RAM. La seconde machine (ordinateur 2) intègre un processeur Intel i7-6950X 10 coeurs 20 Threads @3 GHz, 3.5 GHz en Turbo avec 128Go de RAM.

et de cellules de lieu testés sur une base de 617 images. Les résultats montrent que le gain est quasi linéaire (temps d'exécution divisé par 3 et nombre de fps multiplié par 3) peu importe le matériel utilisé.

Base de données	Nb d'images	Nb de PC allouéss	Nb d'amers recrutés	Nb total de neurones	Utilisation de la RAM (Go)	taille sur le disque (Go)	Durée de l'expérience
KITTI 00	4541	4541	45410	558131	42,5	2,8	12h 30
KITTI 01	1101	700	18000	221527	4,8	0,5	1h 30
KITTI 06	1101	600	11010	137447	4,4	0,5	1h 30
KITTI 07	1101	600	11010	137447	4,2	0,5	1h 30

TABLE 5.2 – Les mesures, présentées au chapitre 4, de coût computationnel par des mesures de temps d'expérience, d'utilisation de mémoire vive ou de mémoire de stockage tout en indiquant le nombre de neurones alloués en début d'expérience.

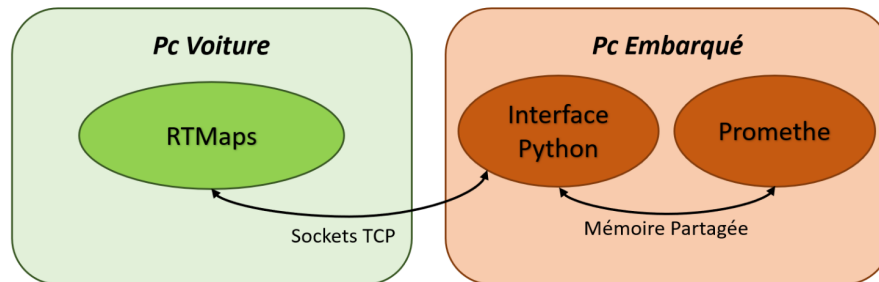


FIGURE 5.9 – Schéma de la communication entre les machines. La partie véhicule centralise les données capteurs (image, orientation du véhicule et du volant) et les envoie via un nœud TCP. Sur le PC embarqué le modèle PerAct construit et reconnaît des lieux et envoie des consignes d'orientation du volant.

Une autre difficulté vient du changement de loi de contrôle du véhicule. Les expériences précédentes étaient menées sur des robots fonctionnant sur un système différentiel avec un angle de braquage nul. Un système différentiel (voir figure 5.10) est capable d'effectuer des rotations sur lui-même et peut être modélisé selon l'équation 5.2.

$$\begin{cases} \dot{x} = u_T \times \cos(\theta) \\ \dot{y} = u_T \times \sin(\theta) \\ \dot{\theta} = \frac{r}{L} \times u_R \\ u_T = \frac{1}{2} \times (u_d + u_g) \\ u_R = u_d - u_g \end{cases} \quad (5.2)$$

où L est l'écartement entre les roues, r est le rayon des roues, θ est l'orientation du véhicule et u_g et u_d sont respectivement les commandes de vitesse angulaire des roues gauche et droite.

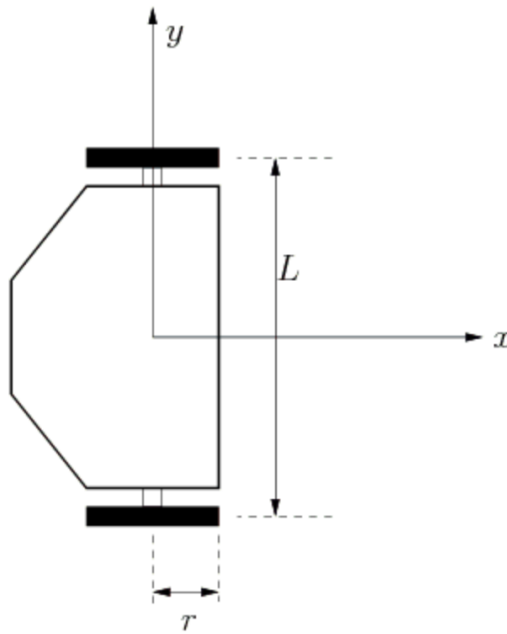


FIGURE 5.10 – Le modèle du système différentiel utilisé lors de précédentes expériences.

Un véhicule automobile a une cinématique différente, avec un angle de braquage non nul et donc incapable de rotation sans translation. Cela implique de changer de modèle cinématique inverse afin de pouvoir obtenir l'angle de volant à appliquer pour orienter le véhicule dans une direction voulue. Le modèle utilisé est celui du tricycle (voir figure 5.11) qui donne les équations suivantes :

$$\begin{cases} \dot{x} = v \times \cos(\theta) \\ \dot{y} = v \times \sin(\theta) \\ \dot{\theta} = \frac{v}{D} \times \tan(\psi) \\ \dot{v} = u_1 \\ \dot{\psi} = u_2 \end{cases} \quad (5.3)$$

où D est l'empattement du véhicule, ψ est l'angle des roues, θ est l'orientation du véhicule et u_1 et u_2 sont respectivement les commandes de vitesse linéaire et de vitesse angulaire.

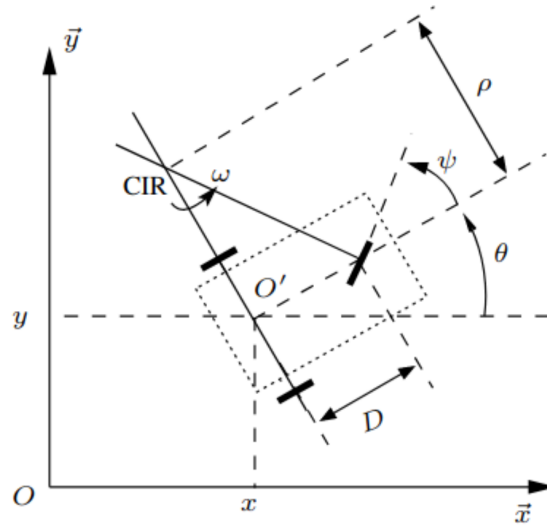


FIGURE 5.11 – Le modèle du tricycle permettant d’obtenir le modèle cinématique du véhicule.

Ceci entraîne aussi des changements dans la dynamique d’apprentissage. Sur robot, le modèle apprend un lieu puis s’oriente dans la direction indiquée par la laisse. Une fois bien orienté l’association lieu/action est apprise.

Sur véhicule, du fait de son angle de braquage et que la laisse, qui servait de mécanisme de déclenchement de l’apprentissage, est remplacé par le volant, il y a un délai entre le moment où le système apprend un lieu et le moment où le véhicule atteint l’orientation qui doit être associée à ce lieu. Pour les expériences, comme la vitesse linéaire est fixe, ce délai a été défini comme un parcours de distance de l’ordre de deux mètres. Une dernière difficulté provient du véhicule de test. Le modèle PerAc apprend des trajectoires par des interactions répétées avec un humain pour le guider. Ceci implique que le véhicule alterne fréquemment entre mode autonome où le modèle contrôle directement le véhicule et mode manuel lorsqu’un utilisateur vient corriger la trajectoire du modèle. Hors jusqu’à présent, les protocoles de sécurité implémentés sur le véhicule empêchaient les changements de mode lors d’une expérience, le passage en mode manuel impliquait l’arrêt de l’expérience. Il a fallu apporter des modifications afin de permettre l’alternance entre les modes automatique et manuel lors d’une expérience.

5.2 Expériences

Le modèle a déjà été testé avec succès en milieu intérieur [Gaussier & Zrehen 1995] et dans de petits environnements extérieurs [Giovannangeli *et al.* 2006a]. Implémenté sur robot à système différentiel et utilisé conjointement à un système d’évitement d’obstacle simple, il permet d’apprendre des trajectoires et de les reproduire de manière robuste. L’objectif

principal de cette série d'expériences préliminaires est d'attester de la portabilité de ce modèle sur un véhicule automobile. En effet, le portage sur véhicule implique des changements au niveau de la dynamique d'apprentissage, il y a un passage d'un apprentissage proscriptionnel sur place du lieu et de son action associée à un apprentissage proscriptionnel différé avec un apprentissage d'évènements décalés dans le temps. Ce portage induit également des modifications de la dynamique du véhicule, passage d'un robot à système différentiel à une voiture avec un angle de braquage, qui peut modifier la dynamique globale du modèle. Ces expériences sont donc plus une preuve de concept, qu'une analyse de performance.

5.2.1 Dispositif expérimental

L'expérience se déroule sur une portion de la piste d'essai du plateau de Satory à Versailles (voir figure 5.12). À l'ouest et au sud de cette portion de piste se trouve une zone boisée, à l'est et au nord l'environnement se compose de bâtiments répartis de manière éparse avec une autre zone boisée en arrière-plan. On peut considérer que l'on est dans une zone péri-urbaine peu dense, donc avec potentiellement peu de points d'intérêts, et donc une zone difficile pour le modèle de construction de lieux.

Le but de chaque expérience est d'apprendre une boucle que le véhicule reproduira de manière autonome. Le véhicule a une vitesse linéaire fixe durant toute l'expérience. Un conducteur utilise uniquement le volant pour définir la trajectoire au début et ensuite apporter des corrections quand le véhicule, en mode autonome, s'écarte de la trajectoire.



FIGURE 5.12 – La portion de piste utilisé pour l'expérience.

Comme cité précédemment, le véhicule utilisé est une Renault ZOE (voir figure 5.13) modifiée de l'institut VEDECOM présentée dans un chapitre précédent (voir section 4.3.1). Pour faire fonctionner le modèle, en plus du module de contrôle moteur, nous n'utilisons qu'une caméra ainsi que la centrale inertielle, pour obtenir l'information d'orientation, parmi l'ensemble des capteurs présents sur le véhicule.

Un GPS RTK est utilisé afin d'obtenir la position précise du véhicule à chaque instant. Ces données, avec l'information sur le mode de fonctionnement du véhicule (mode autonome ou prise en main par le conducteur), permettant de tracer la trajec-



FIGURE 5.13 – Le véhicule VEDECOM utilisé pour l’expérience.

toire cible, la trajectoire obtenue en mode autonome et les zones où un conducteur est intervenu pour apporter des corrections.

Le modèle de cellules de lieu visuelles utilisé par le modèle PerAc est réglé selon les paramètres décrits par le tableau 5.3. La vitesse linéaire du véhicule, du fait des limites imposées par les temps d’exécution de la chaîne de traitement actuelle est de l’ordre de 7km/h.

Pour évaluer les performances lors des expériences, en plus du tracé des trajectoires cibles et effectives, plusieurs mesures vont être utilisées. Certaines mesures s’intéressent aux caractéristiques de la trajectoire comme sa longueur, le nombre de corrections apportées ainsi que la moyenne et l’écart-type de l’écart à la trajectoire cible. D’autres mesures se concentrent sur les cellules de lieu utilisées que ce soit leur nombre lors de l’apprentissage ou le nombre total incluant les corrections ou les moyennes et écart-types du ratio gagnant sur bruit décrit plus en détail dans la partie métrique du chapitre 4.

5.2.2 Résultats

Une expérience se déroule en deux phases, une phase d’apprentissage d’une trajectoire cible suivi d’une tentative de reproduction de cette trajectoire. Le résultat d’une expérience est donc composé de deux trajectoires, la trajectoire apprise et celle reproduite. Deux expériences apprenant chacune une trajectoire différente ont été conduites.

Les résultats peuvent être observés en figure 5.14a et 5.14b. Pour chaque image, les trajets en vert correspondent aux trajectoires cibles, les trajectoires en bleu correspondent aux phases en mode autonome et les parties en rouge correspondent aux corrections par le conducteur.

Paramètre	Valeur
σ_{DoG_1}	5 pixels
σ_{DoG_2}	2 pixels
r_{small}	10 pixels
r_{big}	61 pixels
r_{compet}	30 pixels
N_{PoI}	15
N_d	256
<i>Image width</i>	640
<i>Image height</i>	480
$N_{azimuth}$	361
σ_{azim}	0.5
N_i	5

TABLE 5.3 – L’ensemble des paramètres utilisés pour paramétrer la chaîne de traitement visuel.

Les mesures pour les différents tests sont regroupées dans les tableaux 5.5 et 5.4. Les mesures sur la trajectoire du véhicule montrent que dans les deux expériences on obtient une précision similaire avec un écart aux trajectoires cibles inférieur au mètre.

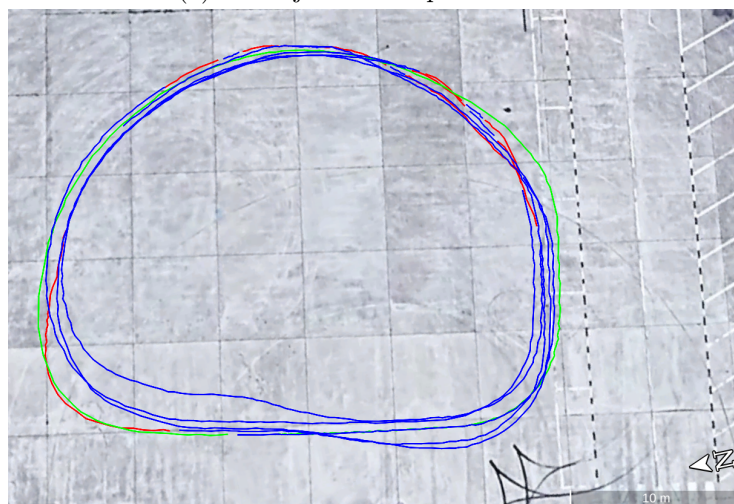
Les mesures sur la réponse des cellules de lieu utilisées illustrent que l’on obtient des cellules avec ratio WNR indiquant que la réponse des cellules gagnantes reste pertinente. Ce ratio est comparable à ce que l’on obtenait dans des environnements péri-urbains.

Dans les deux expériences on se retrouve avec un nombre de corrections équivalentes, aux alentours de 10, mais on peut observer que l’on recrute plus de lieux pour le second trajet alors que ce dernier est plus court. Cela provient du fait que cette trajectoire est essentiellement circulaire alors que l’autre possède des tronçons en ligne droite. Le modèle apprend des associations lors des changements de direction, une trajectoire circulaire implique, par définition, de nombreux changements de direction et donc un recrutement plus important de lieux afin d’apprendre ces changements de directions. De manière plus générale, sur les tracés des trajectoires on voit que la majorité des corrections se fait au niveau des virages. Cette observation correspond au comportement attendu du modèle.

La piste sur laquelle ont été apprises les trajectoires constitue un environnement assez difficile. Une partie significative des alentours de la piste est constituée d’une zone boisée avec peu de points discriminants. Le reste est constitué de bâtiments relativement éloignés et d’une zone dégagée. La conséquence est une présence quasi inexistante d’amers proches. Ceci implique une difficulté à obtenir des cellules précises. Comme expliqué au chapitre 4, la précision des cellules de lieu est influencée par la distance aux amers. Des amers proches, induisant des variations angulaires



(a) La trajectoire du premier test.



(b) La trajectoire du second test.

FIGURE 5.14 – La trajectoire des deux tests réalisés.

plus importantes que ceux distaux, construiront des cellules de lieu avec de petits champs de lieu donc précises. Au contraire, des amers éloignés, ne varient que peu angulairement et construiront des cellules avec des champs de lieu larges, donc peu précises. On peut donc en conclure que, malgré un environnement défavorable, le modèle permet au véhicule d'apprendre une trajectoire et de la reproduire de façon autonome sur plusieurs tours. Comme cela avait été déjà observé sur les tests conduits sur robot, les virages, notamment ceux avec un rayon de courbure assez grand, sont plus difficiles à apprendre. Cela est principalement dû au décalage entre le lieu précis où est appris la cellule de lieu ainsi que sa direction associée et la position à laquelle se trouve le véhicule lorsqu'il rentre à nouveau dans le champ de lieu lors du tour suivant. En effet, dès l'entrée dans le champ de lieu, le véhicule va prédire la direction associée. Ce décalage entre ces deux positions induit une ten-

Mesure	Test 1	Test 2
Nombre de cellules recrutées lors du 1 ^{er} apprentissage	50	64
Nombre total de cellules recrutées	78	122
Nombre de corrections	11	9
Moyenne du score WNR	5.19	4.41
Écart-type du score WNR	1.25	0.51

TABLE 5.4 – Les différentes mesures des résultats obtenus pour chacune des expériences.

Mesure	Test 1	Test 2
distance de parcours	105.35 m	86.06 m
moyenne de l'écart à la trajectoire	0.71	0.63
écart-type de l'écart à la trajectoire	0.53	0.51

TABLE 5.5 – Les résultats de mesure sur les trajectoires pour chaque expérience.

dance à couper les virages que l'on peut remarquer sur les figures 5.14a et 5.14b. Le véhicule tend à s'éloigner de la trajectoire apprise en coupant par l'intérieur. Enfin, la précision des trajectoires n'est cependant pas encore suffisante pour respecter les critères requis (erreur de l'ordre de la dizaine de centimètres) pour de la conduite autonome.

5.3 Discussion

Les résultats de cette expérience montrent que le modèle PerAc semble pouvoir être transposé dans un contexte de conduite autonome de véhicule. Le modèle arrive à apprendre une trajectoire avec une représentation minimaliste de l'environnement. De nombreux tests restent à réaliser afin de pouvoir tester différents aspects du modèle comme la robustesse des trajectoires en cas d'évitement d'obstacles et de tester des variétés différentes d'environnements (autoroutes, centres urbains). Par ailleurs, des modifications du modèle pour en obtenir une variation plus spécifiquement adaptée à la conduite de véhicule, en utilisant les contraintes fortes que sont le fait de rester sur la route ou de n'avoir qu'un sens de circulation, devraient permettre d'obtenir de meilleures performances.

Le modèle PerAc actuel rencontre cependant trois types de limitations qu'il convient de lever afin d'améliorer les performances du système. Le premier type de limite vient de l'implémentation actuelle du modèle sur véhicule. Le fait d'utiliser deux machines séparées avec deux systèmes d'exploitation et deux middlewares différents entraîne des délais dans la chaîne de traitement limitant le fonctionnement en mode temps réel. La chaîne de traitement générant des cellules de lieu, dans son implémentation actuelle, à un temps d'exécution empêchant le fonctionnement

en temps réel. Ce temps d'exécution implique que la version testée du modèle ne peut avoir qu'une faible fréquence d'échantillonnage et donc un temps de réaction long, qui est assez délétère pour les performances de n'importe quelle architecture réactive. La méthode utilisée pour déterminer le délai entre l'apprentissage d'un lieu et de l'action associée (parcourir une distance de deux mètres) peut induire le recrutement de plusieurs cellules de lieu pour un même virage. Une piste qui exploiterait la cinématique du véhicule pour limiter le nombre de recrutement pourrait être de détecter le moment où un conducteur termine la correction (quand il lâche le volant) d'apprendre la direction cible à ce moment-là et d'also enregistrer le temps mis pour parcourir le virage. Cette gestion de la cinématique du véhicule est un problème largement abordé et qui a donné des techniques capables de résoudre ce problème. On peut citer les méthodes basées sur de l'échantillonnage comme le Rapidly-exploring random Tree RRT [LaValle & Kuffner 2001], des techniques basées sur l'interpolation de courbes [Gonzalez Bautista *et al.* 2014], des solutions basées sur de l'optimisation numérique [Cardoso *et al.* 2017], des méthodes de path tracker [Paden & Frazzoli 2016] ou bien de modèles de contrôle prédictif [Guidolini *et al.* 2017].

Un second type de limitation vient du modèle de cellules de lieu visuelles. Les limites de ce modèle ont été abordées en détail dans le chapitre 4. Certaines d'entre elles peuvent affecter les performances du modèle PerAc. La première d'entre elles est aussi liée à l'implémentation sur véhicule. La forme des champs de lieu est liée à l'angle d'ouverture de la caméra fournissant les informations visuelles pour la construction des lieux. La caméra embarquée sur le véhicule permet de construire des cellules avec des champs de lieu orientés. Dans l'environnement de l'expérience, où le véhicule peut circuler sans contrainte, cela peut limiter la robustesse de la trajectoire apprise ou nécessiter plus d'associations lieu action pour obtenir une même robustesse. En effet, comme les cellules sont orientées, il est possible qu'avec un écart trop important, en particulier au niveau de l'orientation, le modèle soit incapable de reconnaître un lieu alors que si le champ de vision du véhicule était plus large, le champ de lieu étant omnidirectionnel, le lieu serait quand même reconnu. Il serait intéressant de voir dans de prochaines expériences si ce cas théorique se vérifie et si, le cas échéant, dans des expériences en environnement contraint (route où il faut rester sur sa voie), cette influence est plus limitée.

La troisième limite vient de l'impossibilité de contrôler la taille des champs de lieu des cellules de lieu. Cela peut poser problème s'il est nécessaire d'apprendre plusieurs cellules de lieu proches les unes des autres, lors d'un virage par exemple. Comme les cellules ont de larges champs de lieu qui se chevauchent, les risques de confusion sont plus importants. Une solution permettant de modérer cet effet pourrait être de combiner le mécanisme de recrutement basé sur une action de l'utilisateur avec celui basé sur un seuil de vigilance de l'activité des cellules de lieu présenté au chapitre 4. Par cette combinaison on pourrait envisager un mécanisme à même de garantir une taille minimale des cellules de lieu. Il consisterait à contraindre le modèle à attendre que l'activité des autres cellules baisse suffisamment avant de recruter une nouvelle cellule. Une autre solution est de construire un répertoire moteur pour

avoir des trajectoires types, comme dans la déclinaison de PerAc sur bras robotique [de Rengerve *et al.* 2015] afin de pouvoir faire de la prédiction de trajectoire issue de ce répertoire. Ceci permet de n'encoder que les cellules de départ et d'arrivée d'une de ces trajectoires type et donc de réduire le nombre total de cellules de lieu recrutées.

Le problème le plus important vient de la méthode d'encodage des cellules de lieu. Cet encodage entraîne un coût computationnel et mémoire empêchant l'utilisation à grande échelle de la reconnaissance de lieu et donc du modèle PerAc.

La dernière limitation vient du modèle PerAc lui-même. Comme vu précédemment, le modèle permet d'apprendre des trajectoires à partir d'une représentation minimaliste. Cependant cette représentation est limitée. Si on apprend plusieurs trajectoires qui possèdent une partie de trajet commune on obtient des intersections. Le modèle est incapable d'associer plusieurs actions à un même lieu est donc de gérer les intersections. Cette constatation a mené à l'élaboration, au sein de l'équipe Neurocyber, d'un modèle de cellules de transitions [Cuperlier *et al.* 2007]. Ce modèle construit, à partir du modèle de cellules de lieu visuelles, une forme de carte cognitive constituée de cellules qui encodent les transitions permettant de passer d'un lieu à un autre. Le problème est alors résolu par l'association d'une unique direction à chaque cellule de transition.

Modèle de contexte visuel

Sommaire

6.1	Description du modèle	113
6.1.1	La chaîne de contexte visuel	113
6.1.1.1	Les filtres de Gabor	114
6.1.1.2	La mémoire	114
6.1.1.3	Le mécanisme d'apprentissage	115
6.1.2	Interactions avec le modèle de cellules de lieu visuelles	116
6.2	Expérimentations	117
6.2.1	Dispositif expérimental	117
6.2.2	Résultats	118
6.3	Discussion	124

Les modèles computationnels de reconnaissance de scènes visuelles qui servent de base à nos modèles de localisation peuvent généralement se diviser en deux grandes familles (voir chapitre 2). La première, utilise des descripteurs locaux qui décrivent des scènes visuelles à partir d'informations de bas niveau. La seconde famille représente les informations spatiales d'une scène visuelle de façon globale à l'aide de descripteurs globaux. Il existe une troisième catégorie, moins étudiée, qui combine les deux types de descripteurs. Notre modèle de cellules de lieu visuelles peut être classé dans la catégorie des descripteurs locaux. Il repose sur une émulation de la vision fovéale afin de décrire l'environnement, la vision périphérique ne servant qu'à obtenir les points d'intérêts qui sont décrits par la chaîne de traitement de la vision fovéale.

Il semblerait que la vision, chez l'homme, utilise, en plus de la vision fovéale, la vision périphérique afin de décrire une scène visuelle. La vision périphérique apporterait une description globale grossière [Navon 1977][Bar 2004] qui participerait à la caractérisation d'un environnement.

Ces observations servent de base à l'élaboration de modèles de contexte visuel. L'idée est d'utiliser deux stratégies pour décrire un lieu : une première, celle de notre modèle de cellules de lieu, utilisant des descripteurs locaux émulant ainsi la vision fovéale et une seconde reposant sur des descripteurs globaux reproduisant la vision périphérique. Cette utilisation d'informations contextuelles fournies par la vision périphérique doit permettre d'améliorer la réponse des cellules de lieu hippocampiques. Cette amélioration serait une conséquence de la réduction du nombre de cellules participant à une compétition grâce à la modulation par les contextes. Cette amélioration devrait se traduire par des champs de lieu plus larges ou des réponses plus fortes des cellules gagnantes. Cette information contextuelle pourrait aussi être utilisée afin de limiter le coût computationnel et mémoire en permettant de découper les groupes neuronaux ainsi que leurs liens en différents blocs et de n'en charger qu'une partie en fonction des contextes.

Un premier modèle de contexte a été proposé dans [Belkaid *et al.* 2016] dans lequel on pouvait observer l'amélioration de la réponse du gagnant par rapport au bruit des autres cellules. Ce modèle, qui sera plus détaillé dans la partie discussion, prenait en entrée un large panorama découpé en plusieurs images qui étaient chacune décrites par 4 filtres de Gabor et associées à une direction. L'ensemble de ces associations formait un contexte. Le modèle de contexte proposé dans ce chapitre, n'utilise pas d'information directionnelle, prend en entrée une seule image, donc un champ de vision plus étroit, et utilise le descripteur GIST afin de qualifier les images.

Ce chapitre commencera par décrire le modèle de contexte visuel en se concentrant sur les ajouts en comparaison du modèle de cellules de lieu visuelles. Il présentera aussi les expériences permettant de le comparer à notre modèle de cellules de lieu avant de terminer par une discussion sur les différences avec l'ancien modèle de contexte et les possibilités de notre modèle.

6.1 Description du modèle

Le modèle de contexte visuel est une évolution du modèle de cellules de lieu visuelles qu'il reprend en totalité. Un système construisant des contextes visuels vient moduler l'activation et l'apprentissage des cellules de lieu visuelles. Il est donc constitué de deux chaînes de traitements visuels. La première est celle vue précédemment (chapitre 4) qui mène à la construction de cellules de lieu. La deuxième, la chaîne de contexte visuel est détaillée dans la suite de cette section. L'interaction entre les contextes et les cellules de lieu sera aussi décrite dans cette partie.

6.1.1 La chaîne de contexte visuel

Cette chaîne de traitement (voir figure 6.1) a pour but de construire des contextes qui vont permettre de segmenter grossièrement les environnements. Elle s'inspire des premières couches de traitement du cortex visuel primaire et en particulier de la voie magnocellulaire, liée à la vision périphérique. Elle prend en entrée une image qui est divisée en trois parties égales. Le but de ce découpage est d'émuler les mouvements de gauche à droite de la tête lors de l'exploration de l'environnement. Sur chaque sous-image, est appliqué un filtrage par GIST [Oliva & Torralba 2001] permettant d'obtenir un descripteur de faible dimension.

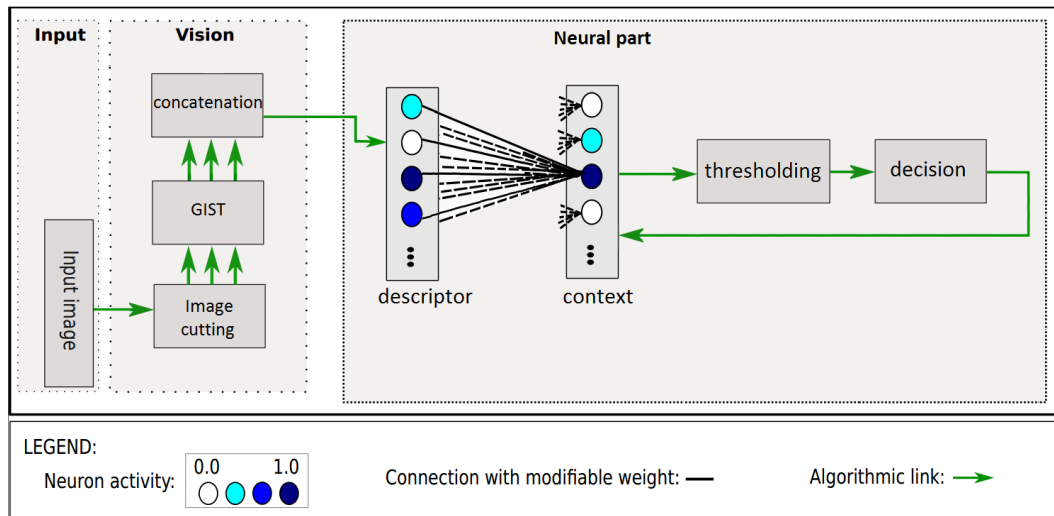


FIGURE 6.1 – La chaîne de traitement visuel des contextes. L'image d'entrée est divisée en trois sous-images. Chaque sous-image est décrite par un descripteur GIST. Les trois descripteurs obtenus sont concaténés afin d'obtenir un descripteur qui sert d'entrée à la couche neuronale.

GIST, déjà présenté succinctement au chapitre 2, est un descripteur d'image global qui fonctionne en trois étapes. Dans ce manuscrit, est utilisée l'implémentation de GIST employée dans [Douze *et al.* 2009]. L'image commence par être divisée selon une grille de 4 par 4 donnant 16 sous-images. Chaque sous-image est convoluée par

un ensemble de 32 filtres de Gabor pour 4 échelles et 8 orientations donnant un ensemble de cartes de caractéristiques. Une opération de moyennage est effectuée sur l'ensemble de ces cartes de caractéristiques afin de construire les histogrammes des orientations de l'image de départ. Les réponses de chacun de ces histogrammes sont agrégées afin d'obtenir un vecteur de 960 éléments qui forment notre descripteur.

Ces trois descripteurs sont ensuite concaténés pour former un descripteur de l'image qui est le contexte visuel. Ce descripteur est comparé à l'ensemble des contextes déjà appris par le modèle en utilisant une distance de Manhattan. Un mécanisme d'apprentissage, similaire à celui des cellules de lieu, permet d'apprendre de nouveaux contextes lorsque la réponse de ceux déjà appris devient inférieure à un certain seuil. La suite de cette section présente les filtres de Gabor et décrit la couche neuronale servant à mémoriser les contextes ainsi que le mécanisme d'apprentissage utilisé pour la mémorisation de contexte.

6.1.1.1 Les filtres de Gabor

Un filtre de Gabor est, dans le cadre du traitement de l'image, un filtre spatial qui répond à des changements de textures selon une direction préférentielle (voir figure 6.4). C'est un filtre linéaire dont la réponse impulsionnelle est composée d'une sinusoïde modulée par une fonction gaussienne. Ce type de filtre est décrit par l'équation 6.1.

$$\begin{cases} g(x, y, \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 \times y'^2}{2 \times \sigma^2}\right) \times \cos\left(2 \times \pi \times \frac{x'}{\lambda} + \psi\right) \\ x' = x \cos(\theta) + y \sin(\theta) \\ y' = -x \sin(\theta) + y \cos(\theta) \end{cases} \quad (6.1)$$

Où x et y sont les coordonnées du point sur lequel est appliqué le filtre, θ est l'orientation de la normale aux bandes parallèles de la fonction de Gabor, λ est la longueur d'onde de la sinusoïde, ψ est le décalage de la phase, σ est la variance de la gaussienne déterminant l'étendue de la gaussienne et γ est le ratio spatial définissant la forme elliptique du filtre. La figure 6.4 montre un exemple de réponse issue d'un filtre de Gabor appliqué sur une image.

6.1.1.2 La mémoire

La reconnaissance de contexte se fait sur une couche neuronale qui est similaire à celles utilisées par le modèle de cellules de lieu visuelles pour mémoriser des vues locales et des cellules de lieu. Elle est précédée par une couche (d) sur laquelle est projetée le descripteur, chaque neurone correspondant à un élément du descripteur. Chaque neurone de cette couche est connecté à l'ensemble des neurones constituant la couche contexte. Chaque neurone c_i de cette couche mémorise une configuration de la couche du descripteur sur ses poids synaptiques. L'activité des neurones de cette couche est régie à chaque instant t par l'équation 6.2.



FIGURE 6.2 – A



FIGURE 6.3 – B

FIGURE 6.4 – Exemple de réponse d'un filtre de Gabor. La convolution d'une image (A) par un filtre de Gabor donne une réponse (B) sur laquelle les lignes ressortent. Le filtre utilisé est défini par un filtre de taille 32, un gamma de 1 et un angle θ de 0 degré.

$$c_i(t) = 1 - \frac{1}{N_d} \sum_{j=1}^{N_d} |w_{ij} - d_j(t)| \quad (6.2)$$

Où w_{ij} est le poids du lien synaptique reliant le i^{eme} neurone de la couche de contexte (c_i) au j^{eme} neurone appartenant au descripteur (d). $d_j(t)$ est l'activité du neurone j , appartenant au descripteur d , au temps t . Du fait de cette équation, plus la configuration des neurones de la couche du descripteur correspondra à la configuration mémorisée par le neurone i de la couche de signature, plus son activité $c_i(t)$ sera élevée. Cette activité est maximale, $c_i(t) = 1$, lors du recrutement du neurone i .

6.1.1.3 Le mécanisme d'apprentissage

Le modèle utilise un apprentissage en un coup comme le modèle de cellules de lieu visuelles. Le mécanisme d'apprentissage est similaire à celui employé pour l'apprentissage des cellules de lieu visuelles ou des vues locales.

Il est contrôlé par un signal binaire $l(t)$ global qui commande le signal de vigilance ($l_i(t)$) propre à chaque neurone i . Ce dernier suit l'équation 6.3 :

$$l_i(t) = l(t) \cdot r_i(t) \quad (6.3)$$

Avec $r_i(t)$ un signal binaire signalant si le neurone i peut apprendre. Une description plus détaillée d'un signal similaire utilisé pour l'apprentissage des cellules de lieu a déjà été proposée au chapitre 3 (section 3.2.2).

L'apprentissage d'un contexte, une configuration du descripteur, sur les poids synaptiques d'un neurone c_i se fait selon l'équation 6.4.

$$w_{ij}(t) = (1 - l_i(t)) \cdot w_{ij}(t-1) + d_j(t) \cdot l_i(t) \quad (6.4)$$

$l_i(t)$ est le signal d'apprentissage du neurone c_i . w_{ij} est le poids du lien synaptique reliant le i^{eme} neurone de la couche de contexte (s_i) au j^{eme} neurone appartenant au descripteur (d). $d_j(t)$ est l'activité du neurone j de la couche du descripteur d .

Cet apprentissage est déclenché lorsque les réponses des contextes déjà appris chutent sous une valeur seuil. Le signal d'apprentissage global $l(t)$ suit le fonctionnement décrit par l'équation 6.5 :

$$\begin{aligned} l(t) &= 1 \text{ if } \max_{\{i \in N_C\}} |c_i(t-1)| < S_{contextvigi} , \\ &= 0 \text{ sinon.} \end{aligned} \quad (6.5)$$

$c_i(t-1)$ est l'activité du neurone i appartenant à la couche de cellules de contexte au pas de simulation précédent. N_C est le nombre de contextes recrutés. $S_{contextvigi}$ est un seuil d'activité compris entre 0 et 1. Ce seuil commande le recrutement de contexte.

6.1.2 Interactions avec le modèle de cellules de lieu visuelles

Le modèle de contexte visuel utilise les contextes qu'il génère afin de moduler l'activité, soit de la couche de signatures qui mémorisent des vues locales, soit de la couche des cellules de lieu qui mémorisent une configuration de la mémoire spatiale de travail sur leurs poids synaptiques (voir figure 6.5). Pour rappel, dans le modèle de cellules de lieu visuelles, l'apprentissage est réalisé sur cette couche par des neurones dont la fonction d'activation est similaire à l'équation 6.2.

La fonction d'activation des cellules de lieu p_i devient :

$$p_i(t) = \frac{1}{N_e} \sum_{q=1}^{N_e} \left(\max_{k=1..N_c}^q (w_{ik}^c \times c_k(t)) \times \left(1 - \frac{1}{N_{SWM}} \sum_{j=1}^{N_{SWM}} |w_{ij}^{swm} - SWM_j(t)| \right) \right) \quad (6.6)$$

Où $SWM_j(t)$ est l'activité du j^{eme} élément de la mémoire spatiale de travail, w_{ij}^{swm} est le poids synaptique liant s_i à SWM_j , N_{SWM} est le nombre d'éléments composant la mémoire spatiale de travail, $c_k(t)$ est l'activité du k^{eme} élément décrivant un contexte, w_{ik}^c est le poids synaptique liant s_i à c_k , N_c est la taille de la couche de contexte et N_e est le nombre de contextes sélectionnés pour réaliser la modulation. $\max_{k=1..N_c}^q$ est un opérateur qui donne le q^{eme} élément maximal parmi un vecteur de taille N_c .

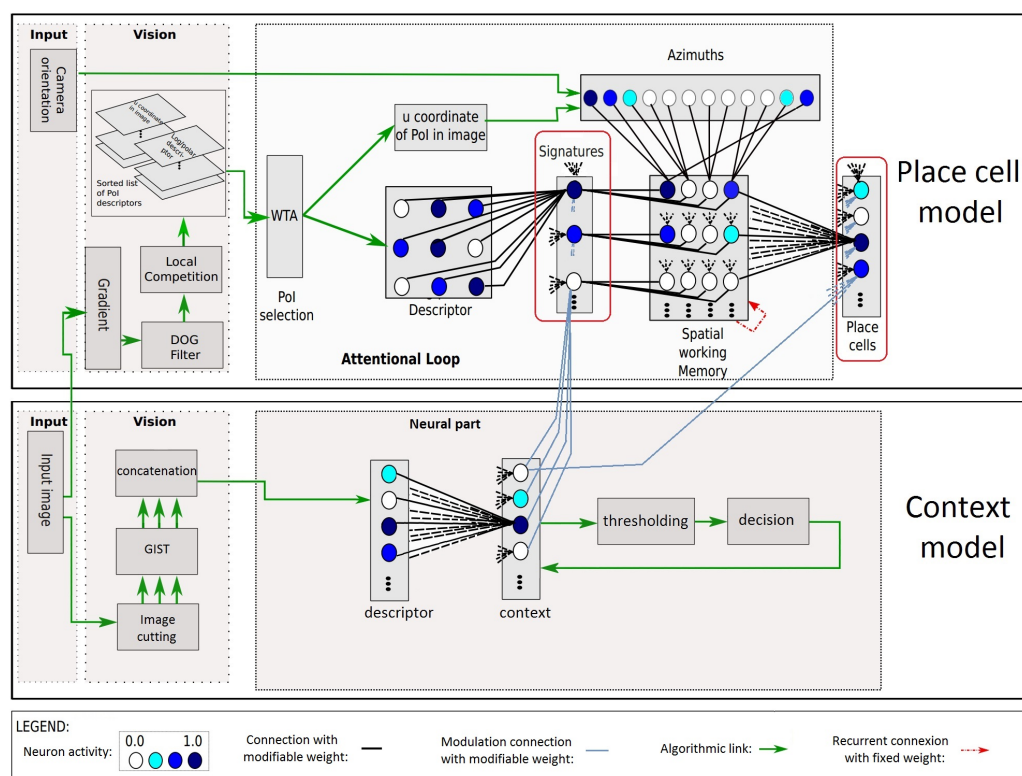


FIGURE 6.5 – Le modèle de contexte dans sa globalité. Les neurones de contextes viennent exercer une modulation sur les neurones de signature et les cellules de lieu. Chaque neurone de la couche de signature ou de la couche des cellules de lieu est connecté à l'ensemble des neurones de la couche des contextes.

6.2 Expérimentations

L'idée derrière le modèle de contexte visuel est que l'apport de cette information contextuelle doit améliorer la réponse des cellules de lieu comme observé dans [Belkaid *et al.* 2016]. Par conséquent, le modèle est évalué en le comparant au modèle de cellules de lieu présenté au chapitre 4. Il est donc testé dans les mêmes conditions et avec le même seuil de 0.42 pour le recrutement de nouvelles cellules de lieu.

6.2.1 Dispositif expérimental

Le modèle est testé sur les bases de données KITTI 01, un trajet sur autoroute, KITTI 06, un environnement de banlieue et KITTI 07, un circuit en centre-ville présentées au chapitre 3. Une expérience se déroule en deux temps. Un premier passage de la base de données lors duquel est réalisé l'apprentissage des contextes et des cellules de lieu avec une règle d'apprentissage en un coup. Un deuxième passage est effectué sur la base de données afin de tester l'apprentissage. Comme

pour le modèle de cellules de lieu, l'apprentissage se faisant un coup, seule une partie des images sert effectivement à l'apprentissage. Concernant les contextes, un peu moins d'une dizaine sont appris par trajectoire, donc, un peu moins de 1% de chaque base de données est utilisé pour l'apprentissage. Pour les cellules de lieu, selon les configurations, entre 10% et 30% de chaque base de données sert pour l'apprentissage. Les métriques utilisées sont les mêmes que celles utilisées au chapitre 4. Le coût computationnel est estimé au travers du nombre de cellules de lieu recrutées. La qualité des réponses des cellules de lieu est mesurée par l'intermédiaire de la mesure de la taille des champs de lieu, des ratios WNR et 3WD, du nombre d'erreurs de reconnaissance séquentielle et du score F1.

6.2.2 Résultats

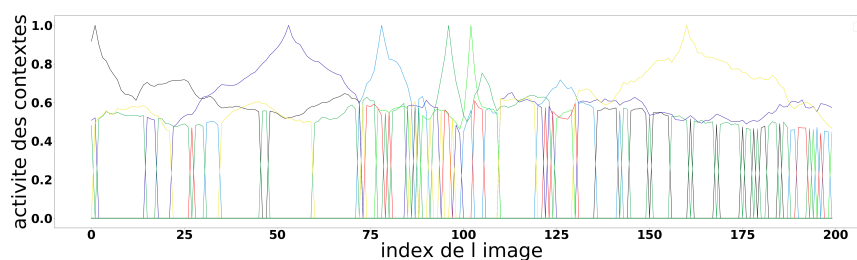
La chaîne de contextes visuels est capable de générer un découpage des trajectoires dans les trois types d'environnements (voir figure 6.6). On peut remarquer, par une simple observation, que la qualité des contextes diffère selon les environnements. Pour la base KITTI 07 (voir figure 6.6c), on peut voir que les contextes gagnants ont une activité ne se détachant pas clairement de celles des autres contextes et on observe de nombreuses inversions dans l'ordre des gagnants. Pour les deux autres bases de données, il est possible de constater que l'activité des gagnants se détache du bruit provoqué par les autres contextes et qu'il y a moins de confusion dans l'ordre des gagnants.

Une première série de tests s'est concentrée sur la manière d'utiliser les contextes. Dans un premier temps a été comparée une configuration utilisant les valeurs analogiques des contextes gagnants sélectionnés à une configuration où la réponse des gagnants est binarisée. Les résultats (voir figure 6.1) indiquent qu'avec les valeurs de contextes analogiques on obtient un peu moins d'erreurs mais avec les contextes binarisés on recrute significativement moins de cellules. Il n'y a pas de différences significatives entre les deux configurations pour les autres mesures. La configuration avec binarisation de l'activité des contextes gagnants sera utilisée pour la suite des expériences.

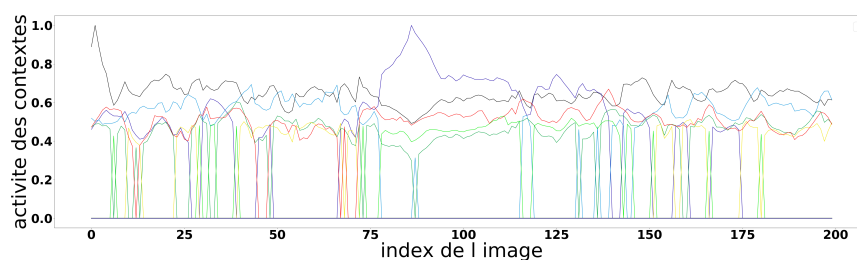
Réponse des contextes	Nombre de PC	Score 3WD	WNR	Erreurs dans l'ordre des PC	Score F1	Taille moyenne des PC (variance) (m)
analogique	216	0.997	3.80	14	0.9951	2.42 (10.81)
binarisée	196	0.992	3.71	17	0.9903	2.28 (11.41)

TABLE 6.1 – Les métriques pour la base de données KITTI 06 pour la modulation de l'activité des cellules de lieu avec binarisation de la réponse des contextes ou utilisation de leurs valeurs analogiques

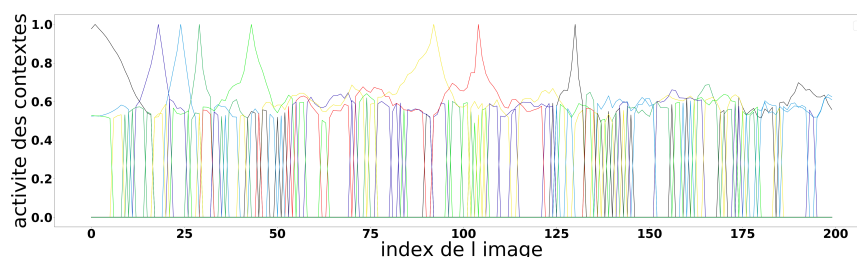
Dans un second test a été analysé l'influence du nombre de contextes gagnants pris en compte pour la modulation. Les résultats (voir tableau 6.2) excluent l'utilisation seule du contexte répondant le plus. Pour cette configuration on observe un nombre très important d'erreurs (59) qui, de plus, concernent des cellules éloignées comme l'atteste le score 3WD très éloigné de la valeur 1. Pour les deux autres confi-



(a) KITTI 01



(b) KITTI 06



(c) KITTI 07

FIGURE 6.6 – L’activité des contextes pour les 200 premières images des bases de données KITTI 01 (autoroute) KITTI 06 (banlieue) et KITTI 07 (centre-ville)

gurations on obtient des résultats similaires. L’utilisation des 5 contextes répondant le plus à l’avantage de générer un peu moins d’erreurs (15). La configuration utilisant les 3 contextes répondant le plus génère un peu plus d’erreurs (17) qui sont cependant légèrement moins significatives avec leur score 3WD plus proche de 1. De plus, le nombre de cellules recrutées est significativement plus faible. C’est donc cette dernière configuration utilisant 3 contextes gagnants qui a été retenue pour la suite.

Deux configurations de la modulation dans le modèle ont été testées. Dans la première configuration l’activité des neurones de signature et de cellules de lieu est modulée. Cette configuration correspond à la figure d’illustration 6.5. Dans la seconde configuration, seule l’activité des cellules de lieu est modulée par les contextes. Aucune différence n’a été constaté entre les deux configurations, il semble donc que

Nombre de contextes gagnants	Nombre de PC	Score 3WD	WNR	Erreurs dans l'ordre des PC	Score F1	Taille moyenne des PC (variance) (m)
1	230	1.26	3.77	59	0.9950	2.04 (8.53)
3	196	0.992	3.71	17	0.9903	2.28 (11.41)
5	216	1.084	3.89	16	0.9951	2.33 (10.92)

TABLE 6.2 – Les métriques pour la base de données KITTI 06 pour la modulation de l'activité des cellules de lieu selon le nombre de contextes pris en compte.

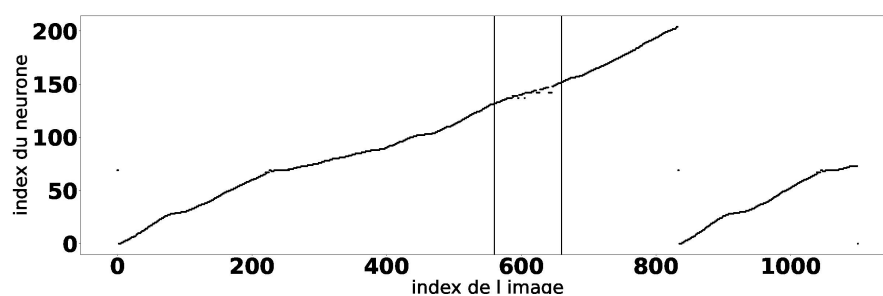
Base	configuration	Nombre de contextes	Nombre de PC	Score 3WD	WNR	Erreurs dans l'ordre des PC	Taille moyenne des PC (variance) (m)
KITTI 01	sans contexte	sans	122	0.844	8.62	24	18.14 (179.43)
KITTI 01	GIST	7	50	0.832	3.03	10	21.83 (98.25)
KITTI 06	sans contexte	sans	283	0.779	16.10	3	3.12 (1.13)
KITTI 06	GIST	7	196	0.992	3.71	17	2.28 (11.41)
KITTI 07	sans contexte	sans	328	0.880	21.1	2	1.07 (0.76)
KITTI 07	GIST	12	287	0.987	6.81	26	1.32 (5.76)

TABLE 6.3 – Les métriques pour les 3 bases KITTI 01, KITTI 06 et KITTI 07 sans modulation et avec modulation de l'activité des cellules de lieu par les contextes générés par le GIST.

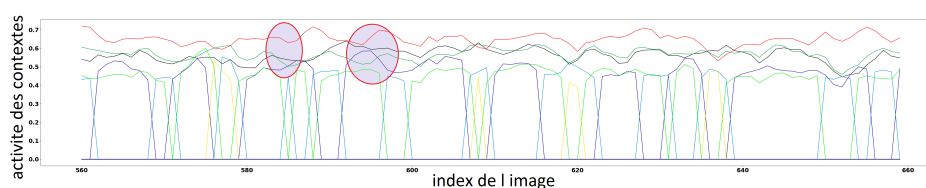
la modulation des vues locales par ces contextes n'apporte rien. Les résultats sur la modulation de l'activité des cellules de lieu montrent que cette dernière altère fortement la réponse des cellules de lieu. L'effet le plus visible de la modulation (voir tableau 6.3) est un nombre significativement plus faible de cellules de lieu recrutées. On peut observer que cet effet dépend du type d'environnement, il est faible en centre-ville (KITTI 07) avec une baisse de 12.5% du nombre de cellules recrutées, il est plus important en banlieue (KITTI 06) avec une baisse de l'ordre de 30% et très important sur autoroute (KITTI 01) avec 60% de moins de cellules recrutées. Les contextes sont construits à partir d'un vecteur de 2880 valeurs et seule une dizaine de contextes sont recrutés par trajectoire. Ceci allié au plus faible nombre de cellules de lieu recrutées réduit fortement le coût computationnel. Un autre effet de la modulation est la baisse du score WNR. Ce dernier est 3 à 6 fois plus faible qu'avec le modèle sans modulation. Il reste néanmoins suffisamment élevé pour indiquer que la réponse des cellules gagnantes reste significativement plus élevée que le bruit provoqué par les autres cellules.

Pour les bases KITTI 06 et KITTI 07 on observe une augmentation du nombre d'erreurs séquentielles. Cette augmentation est une conséquence des erreurs dans l'ordre de reconnaissance des contextes. Ces erreurs sont cependant localisées sur des portions de trajets et les confusions ont lieu avec des cellules proches comme l'indiquent les score 3WD proches de 1 pour les 3 bases de données testées. Cet effet est particulièrement visible sur la base KITTI 06 comme l'illustre la figure 6.7. L'essentiel des 17 erreurs apparaissent sur une plage d'images allant de l'image 560 à 660.

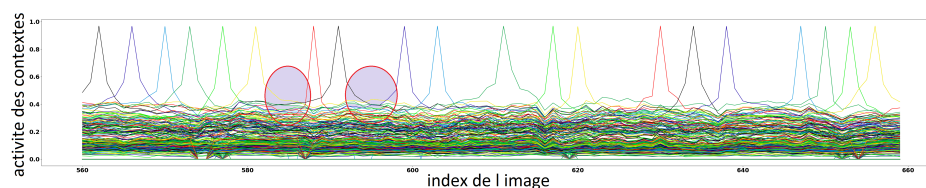
On observe aussi sur les bases de données KITTI 06 et KITTI 07 que l'on obtient une plus grande variabilité dans la taille des champs de lieu.



(a) Index de la cellule de lieu gagnante pour chaque image



(b) Activité des contextes



(c) Activité des cellules de lieu

FIGURE 6.7 – Les erreurs de reconnaissances pour la base KITTI 06. **A** - L'indice de la cellule gagnante pour chaque image. La zone où a lieu l'essentiel des erreurs est encadrée. **B** - L'activité des contextes pour la plage d'images où a lieu le plus d'erreurs. On peut observer des variations dans l'ordre des gagnants. **C** - L'activité des cellules de lieu pour la plage d'images où a lieu le plus d'erreurs. On peut observer une corrélation entre certaines erreurs des cellules de lieu et des contextes.

Par ailleurs le score F1 (voir tableau 6.4) indique que l'on obtient toujours des représentations qui détectent les rebouclages dans les trajectoires parcourues.

L'ensemble des mesures indiquent que la modulation par des contextes construits par GIST permet d'obtenir une plus grande généralisation des réponses des cellules de lieu qui ont des champs de lieu plus larges. Par conséquent, moins de cellules de lieu sont recrutées et on obtient donc des représentations de l'environnement plus grossières. Ces dernières ont, certes, le défaut d'avoir une moins grande précision de localisation, mais, en contrepartie, ont un coût computationnel plus faible. Ce dernier est quantifié dans le tableau 6.5. On observe que le nombre de cellules allouées influe fortement sur l'utilisation de la mémoire vive, de la mémoire de stockage et sur le temps d'exécution des simulations. Par ailleurs, ce tableau illustre l'une des limites actuelles du simulateur : l'absence d'allocation dynamique de neurones. Ce

Base de données	Modèle de cellules de lieu	Modèle de contexte
KITTI 06	0.9997	0.9903
KITTI 07	0.9997	0.9932

TABLE 6.4 – Le score F1 pour le modèles de cellules de lieu et le modèle de contexte

tableau n’a pu être réalisé que grâce à deux séries de simulations : une première servant à obtenir le nombre de cellules de lieu recrutées et une seconde pour obtenir les mesures du tableau pour le bon nombre de neurones alloués.

Base de données	Contexte	Nb de PC alloués	Nb d’amers PC recrutés	Nb total de neurones	Utilisation de la RAM (Go)	taille sur le disque (Go)	Durée de l’expérience
KITTI 01	sans	122	3503	52501	0.802	0.53	0h 57
KITTI 01	avec	50	1458	24986	0.324	0.20	0h 28
KITTI 06	sans	283	8477	113026	2.021	1.40	2h 20
KITTI 06	avec	196	5863	101831	1.682	1.20	1h 38
KITTI 07	sans	328	9751	126316	2.310	1.60	2h 57
KITTI 07	avec	287	8507	115031	2.021	1.40	2h 35

TABLE 6.5 – Les mesures de coût computationnel par des mesures de temps d’expérience, d’utilisation de mémoire vive et de mémoire de stockage pour le nombre de neurones nécessaires pour chaque expérience. Le processeur utilisé est un Intel Core i7-8550U à 4 cœurs cadencé à 1.80 Ghz

Sur la base KITTI 06 ont été effectués une série de tests afin d’obtenir une représentation plus fine de l’environnement en jouant sur le nombre de contextes recrutés par l’intermédiaire de son seuil de recrutement. Les résultats (voir tableau 6.6) montrent qu’il est impossible d’obtenir une représentation aussi fine que sans modulation en réglant le seuil de recrutement des contextes. La représentation obtenue s’en rapprochant le plus, pour le seuil de 0.83 est moins fiable car comportant plus d’erreurs. Par ailleurs, le nombre de contextes recrutés pour obtenir cette représentation est très élevé (263). L’un des buts des contextes est de catégoriser grossièrement l’environnement pour pouvoir en faire un découpage en vue, dans le futur, d’optimiser l’utilisation de la mémoire. Un découpage aussi fin n’a donc aucun sens.

En revanche, en changeant la valeur de seuil pour le recrutement des cellules de lieu il est possible d’obtenir des représentations plus fines. Le tableau 6.7 donne les mesures pour différentes valeurs de seuils pour le modèle de cellules de lieu et le modèle de contextes. On observe encore un tassement de la dynamique de la réponse des cellules de lieu qui se traduit par un score WNR plus faible et généralement un plus grand nombre d’erreurs. Les réponses restent, une fois encore, suffisamment significatives par rapport au bruit et le nombre d’erreurs reste limité. On obtient des représentations plus fines (plus de cellules de lieu) de l’environnement. Pour un seuil de 0.5 pour le modèle de contexte, on construit une représentation similaire à celle obtenue par le modèle de cellules de lieu avec un seuil de 0.42. Il est aussi possible de constater que pour les valeurs de seuils les plus élevées, Les seules mesures signi-

Configuration	Nombre de Contextes	Nombre de PC	Score 3WD	WNR	Erreurs dans l'ordre des PC	Taille moyenne des PC (variance) (m)
Sans contexte	sans	283	0.779	16.10	3	3.12 (1.13)
GIST seuil 0.52	7	196	0.992	3.71	17	2.28 (11.41)
GIST seuil 0.6	18	197	0.992	3.97	20	2.35 (11.37)
GIST seuil 0.7	81	221	0.931	3.78	30	2.28 (8.48)
GIST seuil 0.8	222	264	0.923	6.48	17	2.63 (7.60)
GIST seuil 0.82	251	271	0.870	6.57	14	2.52 (7.50)
GIST seuil 0.83	263	280	0.881	6.84	13	2.54 (5.46)
GIST seuil 0.84	282	290	0.881	4.69	10	2.14 (5.42)

TABLE 6.6 – Les métriques pour la base KITTI 06 sans modulation et avec modulation de l’activité des cellules de lieu par une représentation contextuelle plus ou moins précise.

ficativement différentes entre les réponses du modèle de cellules de lieu et le modèle de contexte sont le score WNR plus faible avec les contextes et le nombre d’erreur dans l’ordre de reconnaissance des gagnants plus important avec les contextes. Ceci s’explique par le fait que l’activité des cellules de lieu modulées résultants du produit de la reconnaissance des cellules de lieu et des contextes. Les contextes n’étant pas parfaitement reconnus eux-mêmes, l’activité finale des cellules modulées chute. Un plus grand nombre de contextes appris induit un plus grand risque de confusion dans la reconnaissance de ces derniers.

Seuil PC	Configuration	Nombre de PC	Score 3WD	WNR	Erreurs dans l'ordre des PC	Score F1	Taille moyenne des PC (variance) (m)
0.5	sans contexte	340	1.10	16.21	3	0.9949	2.31 (4.90)
0.5	GIST seuil 0.52	286	1.05	4.22	10	0.9933	2.40 (5.46)
0.6	sans contexte	387	1.09	16.15	10	0.9955	2.23 (2.95)
0.6	GIST seuil 0.52	366	1.03	4.41	4	0.9950	2.24 (4.71)
0.7	sans contexte	411	1.09	16.86	9	0.9953	2.15 (2.86)
0.7	GIST seuil 0.52	402	0.909	4.47	11	0.9952	2.18 (2.90)
0.8	sans contexte	419	0.784	16.9	6	0.9995	2.17 (0.93)
0.8	GIST seuil 0.52	415	1.09	4.51	12	0.9952	2.02 (2.86)

TABLE 6.7 – Les métriques pour la base KITTI 06 sans modulation et avec modulation de l’activité des cellules de lieu avec différent seuils pour le recrutement des cellules de lieu

L’utilisation de la valeur 0.5 pour le seuil de recrutement des cellules de lieu dans le modèle de contextes appliqué aux bases KITTI 01 et KITTI 07 donne les résultats du tableau 6.8. On obtient des résultats bien différents selon les bases de données. Pour la base KITTI 01 (autoroute) avec le modèle de contextes on obtient toujours une représentation plus grossière de l’environnement avec moins d’erreurs comparé à la réponse du modèle de cellules de lieu. En revanche le score WNR est bien plus faible.

Pour la base KITTI 07 (centre-ville), même si les représentations données par les deux modèles sont similaires en termes de granularité (nombre de cellules recrutées proche), les réponses du modèle de contextes sont moins fiables. Il y a 11 fois plus

d'erreurs, qui sont significatives (score 3WD très élevé). Ces moindres performances sont dues aux contextes qui, pour cette base, sont de piètre qualité avec des réponses des gagnants peu significatives et de nombreuses inversions dans l'ordre des gagnants (voir figure 6.6c).

Base	Configuration	Nombre de PC	Score 3WD	WNR	Erreurs dans l'ordre des PC	Score F1	Taille moyenne des PC (variance) (m)
KITTI 01	sans contexte	122	0.844	8.62	24	n/a	18.14 (179.43)
KITTI 01	GIST seuil 0.52	80	0.90	2.95	9	n/a	19.95 (56.42)
KITTI 07	sans contexte	328	0.880	21.1	2	0.9997	1.07 (0.76)
KITTI 07	GIST seuil 0.52	355	1.40	7.09	22	0.9944	1.10 (4.96)

TABLE 6.8 – Les métriques pour les bases KITTI 01 et KITTI 07 sans modulation avec un seuil de 0.42 pour le recrutement des cellules de lieu et avec modulation de l'activité des cellules de lieu pour un seuil de recrutement des cellules de lieu fixé à 0.5.

Comme évoqué précédemment, lors d'une expérience avec un autre modèle de contextes [Belkaid *et al.* 2016] on observait que l'activité du neurone gagnant était significativement plus élevée que celles des autres neurones grâce à l'apport de l'information contextuelle. On ne retrouve pas ce phénomène dans notre modèle de contexte visuel. Il est à noter que les précédentes expériences se sont déroulées dans des environnements très différents et de tailles bien plus faibles. Le tableau 6.9 donne l'écart moyen entre le premier et le second gagnant avec et sans modulation. On observe, au contraire, un écart moyen plus faible entre le premier et le deuxième gagnant pour l'utilisation de l'information contextuelle.

Base de données	Modèle de cellules de lieu	Modèle de contexte
KITTI 01	0.21	0.09
KITTI 06	0.29	0.20
KITTI 07	0.32	0.26

TABLE 6.9 – L'écart moyen entre le premier et le deuxième gagnant pour le modèles de cellules de lieu et le modèle de contexte .

6.3 Discussion

Les résultats obtenus par la modulation de contextes générés par GIST diffèrent de ceux obtenus dans une expérience précédente, menée sur un robot mobile, dans des environnements extérieurs autour de l'université de Cergy Pontoise [Belkaid *et al.* 2016]. Dans cette expérience, l'apport de la modulation par les contextes permettait de lever les ambiguïtés entre les réponses des cellules de lieu. On ne retrouve pas cet effet dans notre modèle. Cette différence de comportement est essentiellement due à la méthode de construction des contextes. Dans [Belkaid *et al.* 2016], une caméra montée sur pan-tilt effectuait des balayages de

l'environnement de gauche à droite. Ce balayage permet de construire un panorama de l'environnement avec un champ visuel large. Les contextes sont construits en trois étapes. La première consiste à appliquer sur chaque image servant à construire un panorama un ensemble de filtres de Gabor pour obtenir un descripteur par image. La seconde étape associe à chaque descripteur l'orientation de la caméra au moment de la prise d'image. Enfin, l'ensemble des descripteurs et de leurs orientations respectives décrivant un panorama sont associés sur une couche neuronale similaire à celle du modèle de cellules de lieu afin de former le contexte.

Ce modèle de génération de contextes ne pouvait être appliqué sur les bases de test KITTI du fait du plus faible champ visuel. Dans [Belkaid *et al.* 2016] une dizaine d'images permettent de constituer un panorama alors que dans les bases KITTI seule une image est disponible. Une tentative de portage a été faite en essayant de construire des contextes en appliquant les filtres de Gabor sur l'image disponible ou sur le découpage de cette image en trois parties. Cette adaptation ne permettait pas de construire des contextes exploitables. Une exploration de paramètres afin de trouver d'autres filtres de Gabor a été menée. Elle a permis de construire des contextes avec des réponses plus significatives, cependant les tests de modulation avec ces contextes amenaient à une dégradation générale pour l'ensemble des métriques comparé au modèle de cellules de lieu sans modulation. Il serait intéressant lors d'expériences sur véhicule d'avoir un dispositif de prise de vue, soit une caméra montée sur un axe, soit plusieurs caméras, afin de tester ce modèle afin de déterminer si le problème venait du manque d'information d'orientation et du plus faible angle de vue ou si cela venait des filtres qui sont inadaptés dans des environnements propres aux véhicules. Une autre piste à explorer serait de combiner les deux modèles de contextes afin de voir si on peut obtenir une représentation grossière de l'environnement tout en gardant une réponse significative des gagnants.

Lors de la phase de paramétrage de ces filtres de Gabor servant à construire le contexte visuel, certains filtres testés avaient des réponses très différentes selon les environnements. La figure 6.8 montre un exemple de filtre qui ne répond que faiblement dans un environnement peu structuré comme l'est l'autoroute de la base KITTI 01. En revanche, ce filtre répond plus fortement dans les environnements de banlieue (KITTI 06) et de centre-ville (KITTI 07) testés.

L'utilisation de ce filtre, en complément d'autres informations, comme l'information de vitesse obtenue par des cellules de vitesse par exemple, pourrait permettre de construire une forme de contexte plus général sur le type d'environnement (autoroute, banlieue, centre-ville, etc). Cette information contextuelle pourrait servir à adapter la stratégie de perception en fonction des types d'environnements, en modulant la valeur du seuil s_{vigi} par exemple. Elle pourrait aussi être employée afin d'adapter les comportements de navigation selon les environnements, en se reposant plus sur des méthodes de suivi de route ou de véhicule par exemple.

Le modèle de contexte visuel a pour but à moyen terme de diminuer le coût computationnel et surtout mémoire du modèle. L'information contextuelle servirait à découper les couches de neurones de signatures, de la mémoire de travail et des cellules de lieu en plusieurs sous-ensembles. Il serait ensuite possible de charger dy-

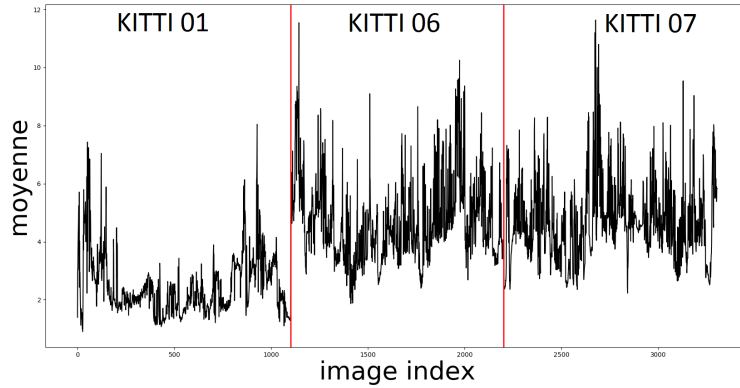


FIGURE 6.8 – La valeur moyenne du pixel pour chaque image convoluée par un filtre de Gabor. Les 1101 premières images correspondent à la base de données KITT1 01 (autoroute), les 1101 suivantes à la base KITT1 06 (banlieue) et les 1101 dernières à la base de données KITT1 07 (centre-ville). Le filtre utilisé est défini par un filtre de taille 16, un gamma de 1 et un angle θ de 0 degré. On peut observer que la valeur moyenne des pixels est plus faible pour la base de données KITT1 01.

namiquement certains de ces sous-ensembles en fonction des réponses contextuelles. Les résultats du tableau 6.7 montrent qu'il est possible d'obtenir une représentation similaire à celle du modèle de cellules de lieu en termes de précision et de fiabilité permettant la mise en œuvre de l'optimisation de la mémoire évoquée juste avant.

La différence principale du modèle vient du score WNR faible. Ce dernier traduit un tassement de la dynamique des réponses. Dans le cadre actuel où un lieu est encodé par une unique cellule et que cette dernière n'encode que ce lieu, la compétition entre les cellules reste capable de donner une réponse des gagnants pertinente. Ce tassement de la dynamique de l'activité des cellules de lieu n'est pas un réel souci à partir du moment où l'ordre relatif des réponses est préservé. En effet, notre modèle n'exploite pas directement la valeur de l'activité des cellules de lieu mais le résultat de la compétition, que ce soit pour la localisation ou la navigation sensorimotrice.

Par ailleurs, la représentation plus grossière mais néanmoins toujours fiable obtenue par modulation est intéressante. Elle pourrait être utilisée en combinaison de représentations plus fines pour construire des représentations multi-échelles. Comme mentionnées au chapitre 2, ces dernières sont utiles pour améliorer la localisation et surtout permettent de limiter fortement les coûts computationnels par un découpage de la mémoire. Par ailleurs, une telle représentation est biologiquement plausible comme l'atteste l'existence de cellules de lieu ayant des champs de lieu de tailles différentes [Brun *et al.* 2008].

Conclusion et perspectives

Le but de cette thèse était double. Pour l'institut VEDECOM l'objectif était d'aborder le problème de la navigation des véhicules autonomes selon un angle nouveau grâce à la bio-inspiration et de commencer à explorer ce champ de recherche. Du côté du laboratoire ETIS, le but de cette thèse était d'étendre son domaine de recherche en s'attaquant au problème de la navigation de véhicules autonomes et de commencer à tester ces modèles de navigation dans ce nouveau contexte.

Dans ce manuscrit, après avoir commencé par circonscrire le problème de la navigation de véhicules autonomes à celui de la reconnaissance de lieux visuels et exposé les limites des principaux modèles, j'ai présenté les approches bio-inspirées et surtout neuromimétiques et leur potentiel pour aider à résoudre ce problème. En complément j'ai donné de notions de neurobiologies utiles à la compréhension de mes travaux.

Par la suite j'ai présenté un modèle de reconnaissance de lieux par la vision, neurorobotique, basé sur le fonctionnement du cerveau des mammifères. Par toute une série de tests à partir d'enregistrements issus de véhicules, j'ai pu mesurer les performances de ce modèle selon un ensemble de critères. Ces résultats ont permis de montrer que, malgré une limitation en matière de coût computationnel, ce modèle, initialement pensé pour la reconnaissance de lieu en intérieur, s'il n'atteint pas le niveau de l'état de l'art, obtient des performances honorables.

Ces tests ont aussi permis d'identifier un des points bloquants dans le dialogue entre les communautés du véhicule autonome et de la navigation neurorobotique : la difficulté à avoir des critères de mesures et de comparaisons communs entre les approches dites classiques et celles neurorobotiques. Cela vient de différentes philosophies dans la stratégie de navigation mises en place à partir de la localisation. Dans les approches dites classiques, la navigation repose énormément sur la précision métrique de la localisation. Par conséquent, les critères liés à la précision sont cruciaux pour évaluer les performances des modèles de localisation. De ce point de vue les modèles bio-inspirés sont moins performants. Les différentes observations, à l'heure actuelle, des comportements de navigation des animaux semblent suggérer qu'ils se basent plus sur une localisation grossière complétée par des mécanismes réactifs afin de naviguer. La localisation précise, qui existe comme l'attestent certaines cellules de lieux avec un champ de lieu de l'ordre de la dizaine de centimètres, ne semble être utilisée que ponctuellement. Il en découle que les modèles de localisation, issus de la compréhension encore parcellaire des modèles biologiques, ont une faible précision métrique.

Avec l'aide d'un stagiaire, j'ai pu adapter sur un véhicule de l'institut VEDECOM le modèle de navigation réactive PerAc reposant sur le modèle de cellules de lieux visuelles précédemment évalué. J'ai pu réaliser des premières expériences permettant de prouver que ce modèle de navigation réactive est capable de fonctionner sur véhicule. D'autres expériences restent à réaliser afin de mesurer quantitativement les performances du modèle.

Pour finir j'ai proposé une itération du modèle de reconnaissance de lieu visuelle incluant une modélisation parcellaire du cortex visuel primaire : le modèle de contextes visuels. Il génère un découpage de l'environnement qui vient moduler l'activité des cellules de lieux. Cette modulation permet de construire une représentation de l'environnement avec des cellules de lieux plus larges tout en gardant des réponses pertinentes. Ce changement d'échelle, permis par cette modulation, pourrait, à plus long terme, servir à construire des représentations multi-échelles plus efficaces et plus légères en terme de coût computationnel.

Le gros point bloquant pour le fonctionnement du modèle de navigation PerAc ou du modèle de cellules de lieux visuelles en des conditions temps réel vient du coût computationnel et plus particulièrement du coût mémoire. Une partie de ce problème vient du fonctionnement du simulateur lui-même, qui n'a pas été pensé à l'origine pour manipuler autant de neurones et de connexions. Le plus grande partie vient de l'encodage des données. Pour pallier ce problème nous avons quatre pistes de réflexion qui peuvent être complémentaires.

La première piste est celle que j'ai partiellement commencé à explorer dans le chapitre 6. L'idée est d'avoir une représentation multi-échelle constituée d'une représentation grossière nécessitant peu de calculs, et d'une représentation plus fine, nos cellules de lieux visuelles, se reposant sur la représentation grossière. Le but de ce système est d'avoir une réponse de la représentation grossière permettant de ne mettre en mémoire vive et donc ne faire une compétition que sur les cellules de lieux apprises dans cette partie de la représentation grossière. Cette méthode, testée avec succès par un algorithme de SLAM [Alsayed *et al.* 2015], apporterait un progrès comparé à l'implémentation actuelle qui garde toute les cellules de lieux en mémoire vive. De plus, les observations au niveau du système visuel, avec une vision périphérique rapide influant sur la vision fovéale plus lente, et des tailles de champs de lieux multiples indiquent que l'existence d'un tel système chez les animaux est plausible.

La seconde solution consiste à découpler l'apprentissage des amers et des cellules de lieux. Actuellement l'apprentissage des cellules de lieux entraîne l'apprentissage de nouveaux amers. Il peut arriver une situation où un nouvel amer est appris alors que des amers précédemment appris avaient une réponse pertinente. On peut aussi se retrouver dans une configuration où un amer est appris plusieurs fois. Le découplage de l'apprentissage des amers et des cellules de lieux devrait donc également permettre de réduire le coût mémoire.

La troisième possibilité pour réduire ce coût est de modifier le mode d'apprentissage. Actuellement l'apprentissage se fait en un coup, ce qui signifie qu'un lieu et les amers qui lui sont associés sont appris sur une seule image. Ce mode d'apprentissage

a le mérite d'être rapide mais en contrepartie il peut donc arriver que les informations apprises ne soient pas pertinentes et que l'on apprenne un amer, ou pire, une cellule ne répondant que sur l'image d'apprentissage. Pour pallier ce problème tout en gardant l'avantage de la rapidité, une solution pourrait être d'avoir des phases de consolidation où le système rejouerait le trajet et éliminerait de la mémoire les cellules et amers dont le temps d'activation serait trop faible. Ce mécanisme de consolidation correspondant à une modélisation assez parcellaire du mécanisme de mémorisation à long terme que l'on observe chez les animaux lors des phases de sommeil.

La dernière piste pour diminuer le coût mémoire est d'améliorer l'encodage des informations et en particulier des informations visuelles des amers. Actuellement la projection de la portion d'image sur le champ neuronal de signature est apprise telle quelle. Cette projection représente un vecteur de deux cent cinquante six valeurs dans les expériences que j'ai présenté. Dans des expériences précédentes, en intérieur, cela représentait un vecteur de deux mille neuf cent seize valeurs. L'encodage pourrait être amélioré en obtenant un codage épars : un codage qui maximise le codage des informations tout en minimisant la taille de la base de codage. Un tel codage peut être obtenu de différentes manières. Une première méthode peut être de prendre les premières couches d'un réseau de neurones profond entraîné pour de la reconnaissance d'objets par exemple. Ces premières couches génèrent un code, une feature map, qui est un codage de l'information compact. Le défaut de cette méthode est d'avoir une phase d'apprentissage longue qui ne garantit pas d'avoir un code compact pertinent. Une autre solution, plus pertinente, est d'utiliser un algorithme pensé pour construire un codage épars. Il en existe un certain nombre, on peut citer par exemple l'algorithme d'anti-Hebb [Földiák 1990]. L'inconvénient de cette méthode est que les algorithmes ont un temps de convergence assez long avant d'obtenir une base de codage épars.

A plus long terme, j'ai identifié cinq pistes de recherche pour améliorer le modèle : changer l'encodage des lieux, améliorer le système visuel, améliorer la partie navigation du modèle, introduire de l'autoévaluation à plusieurs niveaux et enfin d'introduire des aspects d'information contextuelle.

Les modèles présentés dans ce manuscrit utilisent ce que l'on appelle des "cellules grand-mères" : une cellule encode un lieu. Or, on sait, grâce à des expériences de neurosciences sur des environnements de grande taille (plusieurs dizaines de mètres) [Rich *et al.* 2014], que les cellules de lieux encodent plusieurs lieux distincts (voir figure 7.1). Ceci transparait par le fait que les cellules exhibent plusieurs champs de lieux. Cette propriété des cellules de lieux remet en question certaines modélisations utilisées en neurorobotique. Ces dernières reposent souvent sur l'utilisation de réseaux d'attracteurs continus, des réseaux qui ne peuvent pas facilement générer ce phénomène de multiple champs de lieux. Notre modèle de cellules de lieux est potentiellement capable d'exhiber des champs de lieux multiples comme le suggère certaines erreurs dans l'ordre de reconnaissance des cellules de lieux. Il est d'ailleurs possible que certaines fausses réponses de nos cellules que l'on a constaté soient, en fait, des réponses multiples que notre façon d'exploiter les cellules (un lieu =

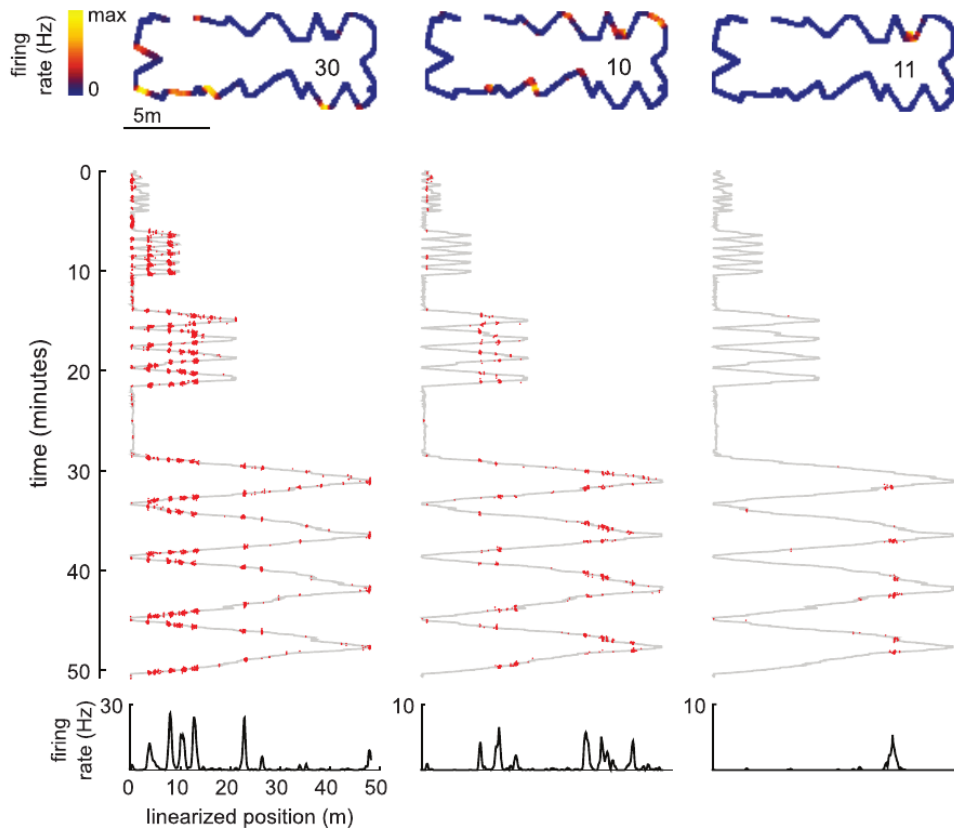


FIGURE 7.1 – Exemple de cellules encodant plusieurs champs de lieux. De haut en bas, pour trois neurones différents : la fréquence des impulsions sur un trajet de 48 m, les pics d’activations sont indiqués ; positions linéarisées de l’animal (en gris) avec les pics d’activité (en rouge) ; fréquence des impulsions neuronales linéarisée. Les pics d’activations lorsque la vitesse de l’animal est inférieure à $5m/s$ sont omis [Rich *et al.* 2014].

une cellule) considère comme des erreurs. Une hypothèse, renforcée par cette découverte de cellules à champs de lieux multiples, envisage qu’un lieu n’est pas encodé par une unique cellule mais qu’au contraire une population de neurones, par leur activité combinée, décrivent un lieu. La figure 7.2 illustre les deux méthodes d’encodage évoquées. Dans le cas où une cellule de lieux n’a qu’une réponse unique, l’encodage d’un lieu par un ensemble de neurones semble peu efficace en termes de coût mémoire. De son côté, le codage d’un lieu par une cellule unique se retrouve confronté au problème de la gestion des recouvrements de champs de lieux. Avec des cellules ayant des réponses multiples, pour l’encodage d’un lieu par une unique cellule on retrouve toujours le problème des recouvrements mais émerge aussi le problème d’identifier que les activations multiples d’une même cellule correspondent à des lieux différents. L’encodage de lieux par un ensemble de cellules devient plus intéressant, une même cellule peut participer à encoder plusieurs lieux définis par des

combinaisons de cellules différentes et donc moins peser sur le coût computationnel.

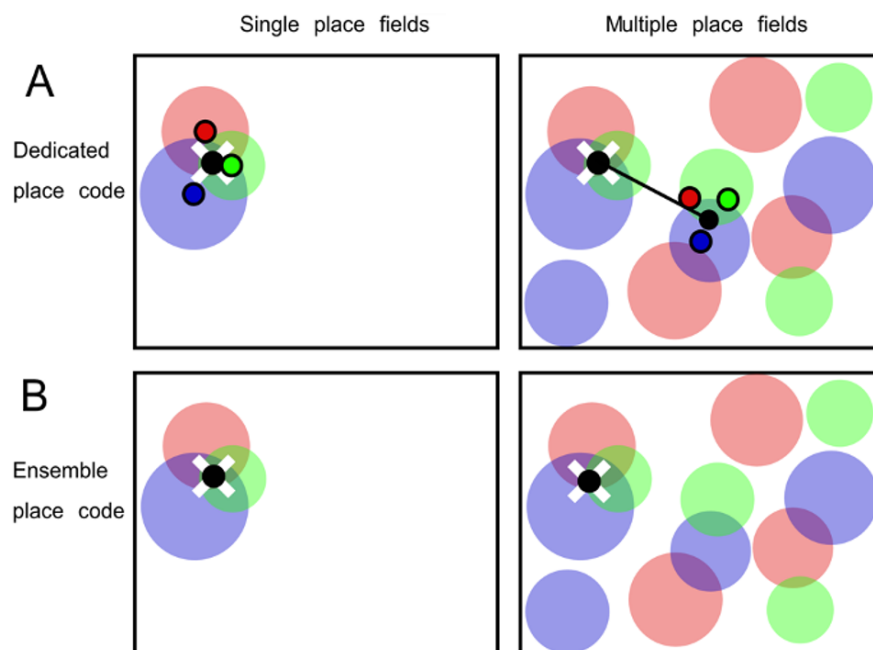


FIGURE 7.2 – Les différentes combinaisons pour l’encodage d’un lieu discutées. **A** - Une cellule encode un lieu dans le cas où une cellule n’a qu’une réponse unique (gauche) et le cas où elle montre des réponses multiples (droite). **B** - Un lieu est encodé par une réponse conjointe de plusieurs cellules dans le cas où une cellule n’a qu’une réponse unique (gauche) et le cas où elle montre des réponses multiples (droite) [Park *et al.* 2011].

Une piste d’évolution de nos modèles peut donc être de modéliser ses réponses de cellules multiples et de remplacer la compétition donnant une unique cellule gagnante par l’utilisation d’une population de neurones pour encoder un lieu. Ce changement de codage, en plus d’améliorer le coût computationnel comme expliqué précédemment, pourrait permettre d’obtenir une reconnaissance plus robuste des lieux. En effet si, pour un lieu encodé par un ensemble de neurones, seule une partie de cet ensemble est suffisant pour reconnaître le lieu, on peut envisager, par exemple, qu’une partie de cet ensemble de cellules repose sur des éléments visuels percevable de jour et d’autres sur des éléments percevable de nuit. On se retrouverait donc avec un encodage de lieu robuste au cycle jour/nuit.

Le système visuel de certaines espèces d’animaux est très complexe comme celui de l’être humain que j’ai décrit en détail au chapitre 3. Il est entre autre capable d’extraire des informations de profondeur. Doter le système visuel du modèle de cellules de lieu visuelles de cette capacité constituerait un progrès en particulier pour mieux contrôler la taille des champs de lieux. En effet, la taille des champs de lieux est directement liée à la proximité des amers. Des amers proches, donc rapidement

hors du champ de vision, construiront des cellules de lieux avec des champs de lieux de petites tailles alors que des amers éloignés, donc visibles plus longtemps, génèreront des champs de lieux plus larges. L'information de profondeur permettrait donc d'affiner la sélection des amers pour la construction des cellules de lieux et d'exercer un contrôle sur la taille des champs de lieux. Une modélisation plus profonde du cortex visuel pourrait aussi améliorer la sélection d'amer. Des observations indiquent que des parties de ce cortex sont capables d'identifier des formes simples ou de détecter les mouvements. Ces parties du cortex pourrait être grossièrement modélisées en utilisant un système de détection d'objets (véhicules, piétons) comme YOLO par exemple. L'information de ce détecteur pourrait permettre de définir des zones à ignorer pour la recherche de point d'intérêts. On obtiendrait ainsi des amers plus pertinents et robustes car ils ne seraient pris que sur des invariants de l'environnement.

La partie navigation du modèle pourrait être améliorée de plusieurs façons. Comme je l'ai expliqué en conclusion du chapitre 5 l'association lieu action permet de définir des attracteurs et des trajectoires stables mais trouve ses limites lorsqu'elle est confrontée à des bifurcations. Une solution simple pourrait être d'avoir une information contextuelle supplémentaire permettant de choisir une trajectoire plutôt qu'une autre. L'inconvénient de cette solution qui crée une carte par trajectoire est d'encoder plusieurs fois des mêmes portions de trajectoire et de nécessiter un apprentissage par trajet. Une solution plus efficace est de construire une carte cognitive plus exhaustive de la géométrie de l'environnement qui n'encode pas des associations lieux actions mais des associations entre des transitions d'un lieu à l'autre et l'action permettant d'effectuer cette transition. Un tel modèle existe déjà [Cuperlier *et al.* 2006] et a été testé avec succès en navigation intérieure. Ce modèle construit une carte cognitive basée sur les transitions qui peut ensuite être utilisée par un algorithme de recherche du plus court chemin, comme Dijkstra [Dijkstra 1959] par exemple, afin de définir une trajectoire. Il reste à tester ce modèle sur véhicule, ce qui nécessite, en plus de réduire le coût mémoire, de tester au préalable le modèle de cellules de lieux multimodales [Jauffret *et al.* 2015]. Ce modèle est une itération du modèle de cellules de lieux visuelles qui utilise, en plus de la vision, des informations de proprioception qui sont utilisées pour construire des cellules de lieux basées sur l'intégration de chemin. Ces informations de proprioception sont nécessaires pour la génération des cellules de transition. Le modèle de transition devrait être testé lors des prochains travaux du laboratoire sur la navigation de véhicules.

Le modèle de navigation peut aussi être amélioré en en le spécialisant pour la navigation des véhicules.

Les véhicules, dans une écrasante majorité des cas, circulent sur le réseau routier. Ce type d'environnement est régi par des contraintes qui peuvent être exploitées pour restreindre le problème de la navigation à un cas plus simple. En effet, sur une route, hors intersection, la mobilité est contrainte par le sens de circulation et la règle impliquant de rester sur sa voie. La contrainte sur l'orientation du véhicule a pour effet d'annuler les faiblesses des champs de lieux orientés. Pour rappel, le champ de lieux des cellules de lieux est directement lié à la largeur du champ de

vision, un champ de vision large générant des cellules de lieux avec des champs de lieux omnidirectionnels alors qu'un champ de vision plus resserré génère des cellules avec des champs de lieux orientés. Quand le véhicule ou le robot peut se déplacer librement, le champ de lieux doit être omnidirectionnel (donc le champ visuel aussi) pour que le lieu soit reconnaissable peu importe l'orientation selon laquelle l'agent atteint le lieu. C'est le cas pour les expériences de homing et d'apprentissage de trajectoire en intérieur réalisées avec le modèle PerAc. Quand les mouvements sont contraints, comme sur une route, un champ de lieu orienté (donc un champ visuel plus réduit) dont l'orientation respecte les contraintes de l'environnement est suffisant pour reconnaître un lieu.

Dans un tel environnement contraint, un simple algorithme de suivi de ligne (soit le bord de la route, soit le marquage au sol) est suffisant pour avoir une navigation réactive. Une version du modèle impliquant l'utilisation d'une stratégie de suivi de ligne en complément du modèle PerAc [Jauffret *et al.* 2013] à déjà été testée en simulation. Une version de ce modèle utilisant le modèle de cellules de transition devrait obtenir des résultats intéressants. Ce qui serait aussi intéressant de tester avec ce modèle est de vérifier si une version minimisant la carte cognitive, en limitant le recrutement des cellules de transition uniquement au niveau des intersections, est capable d'effectuer correctement des tâches de navigation.

Un modèle utilisant deux stratégies comme celui existant [Jauffret *et al.* 2013] ou celui que je viens d'ébaucher reposent sur une notion qui me semble primordiale d'explorer plus avant : l'autoévaluation. Cette dernière permet de faire la sélection entre les deux stratégies, suivi de ligne ou modèle PerAc. Les mécanismes d'autoévaluation peuvent être déclinés, en plus de la sélection de stratégie, dans plusieurs parties du modèle de reconnaissance de lieu et de navigation. Un système d'autoévaluation de la pertinence des différents capteurs (caméra, centrale inertielle) pourrait être utilisé, en conjonction de cellules de lieux multimodales pour améliorer la robustesse des réponses des cellules en donnant un poids plus ou moins important à l'une ou l'autre modalité, le modèle de cellules multimodales actuel ayant lui une pondération fixe. On peut aussi envisager, via ce système, de combiner le modèle bio-inspiré avec des approches classiques.

Les modèles de cellules de lieu et de cellules contextuelles utilisent un mécanisme ad-hoc de recrutement. Ce recrutement se fait par un contrôle des activités des cellules et si cette activité chute sous un certain seuil le signal d'apprentissage est envoyé. L'exploration de paramètre présentée au chapitre 4 illustre que cette valeur de seuil permet d'exercer un contrôle sur le nombre de cellules recrutées et donc sur la précision de la représentation. De plus j'ai observé que pour un même seuil le recrutement des cellules pouvait varier selon le type d'environnement traversé. Ainsi le mécanisme de recrutement pourrait être amélioré en modulant le seuil en fonction du type d'environnement traversé et de la précision requise pour accomplir telle ou telle tâche. Cette modulation du seuil de vigilance pourrait être opérée grâce à l'auto-évaluation du comportement de navigation du véhicule. Par exemple, en commençant avec un seuil relativement faible et en le relevant progressivement jusqu'à obtenir un comportement correct. La valeur ainsi obtenue pourrait alors être

apprise et associée aux neurones de contextes.

Cette question de l'autoévaluation mène à celles des contextes. Le terme contexte désigne ici un état de l'environnement extérieur ou bien un état interne au système.

Dans le dernier cas un contexte peut donc être la perception de l'efficacité d'une stratégie de navigation, comme dans le modèle d'autoévaluation [Jauffret *et al.* 2013], ou de son inefficacité menant à une adaptation du comportement, en lançant une phase d'exploration par exemple.

Le chapitre 6 illustre un exemple de contexte concernant l'environnement : en cherchant des filtres de Gabor pour construire un descripteur capable de découper grossièrement l'environnement j'ai trouvé un filtre qui ne réagissait pas à la végétation et donc est capable de discriminer entre des environnements urbains et plus ruraux.

Un premier exemple d'information contextuelle est le type d'environnement (urbain, rural, autoroutier). Cette information pourrait permettre de moduler le nombre et donc la taille des champs de lieux, comme je l'ai expliqué précédemment, ou bien d'adapter le comportement du véhicule en fonction des environnements. Sur autoroute, par exemple, une localisation précise, hormis au niveau des sorties et bifurcations, n'est pas vraiment utile et la détection d'un piéton, très inhabituelle, indique un problème et donc d'avoir une attitude prudente. Au contraire, en zone urbaine où il y a plus d'intersections et de croisements, une plus grande précision est requise et le fait de détecter un piéton est assez normal.

Pour de la navigation au long cours, pouvoir déterminer les informations contextuelles que constituent l'heure de la journée (jour ou nuit), les conditions météorologiques (ensoleillé, pluvieux, brumeux) ou la saison permettrait d'adapter la perception, que ce soit par un réglage de la sensibilité de la caméra, par une modulation de la confiance à tel ou tel capteur ou en modifiant la stratégie de prise d'information.

Bibliographie

- [Ali *et al.* 2016] Farman Ali, Sajid Ullah Khan, Muhammad Zarrar Mahmudi et Rahmat Ullah. *A comparison of FAST, SURF, Eigen, Harris, and MSER features*. International Journal of Computer Engineering and Information Technology, vol. 8, no. 6, page 100, 2016. (Cité en page 13.)
- [Alsayed *et al.* 2015] Zayed Alsayed, Guillaume Bresson, Fawzi Nashashibi et Anne Verroust-Blondet. *PML-SLAM : a solution for localization in large-scale urban environments*. In PPNIV - IROS 2015, Hambourg, Germany, Septembre 2015. (Cité en pages 25 et 128.)
- [Amari 1977] Shun-ichi Amari. *Dynamic of pattern formation in lateral-inhibition type neural fields*. Biological cybernetics, vol. 27, pages 77–87, 09 1977. (Cité en page 96.)
- [Andersen 2007] Per Andersen. The hippocampus book. Oxford University Press, Oxford New York, 2007. (Cité en page 43.)
- [Anderson & Jeffery 2003] Michael I. Anderson et Kathryn J. Jeffery. *Heterogeneous Modulation of Place Cell Firing by Changes in Context*. The Journal of Neuroscience, vol. 23, no. 26, page 8827–8835, Oct 2003. (Cité en page 46.)
- [Arandjelović *et al.* 2016] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla et J. Sivic. *NetVLAD : CNN architecture for weakly supervised place recognition*. In IEEE Conference on Computer Vision and Pattern Recognition, 2016. (Cité en pages 13 et 14.)
- [Arkin *et al.* 1998] Ronald C Arkin, Ronald C Arkin *et al.* Behavior-based robotics. MIT press, 1998. (Cité en page 19.)
- [Arnay *et al.* 2008] R. Arnay, L. Acosta, M. Sigut et J. Toledo. Applying an ant colony optimization algorithm to an artificial vision problem in a robotic vehicle, page 490–497. Springer Berlin Heidelberg, 2008. (Cité en page 20.)
- [Avanzini *et al.* 2009] Pierre Avanzini, Eric Royer, Benoit Thuilot et Philippe Martinet. *A Global Decentralized Control Strategy for Urban Vehicle Platooning using Monocular Vision and a Laser Rangefinder*. pages 291 – 296, 01 2009. (Cité en page 18.)
- [Avens & Lohmann 2004] Larisa Avens et Kenneth Lohmann. *Navigation and Seasonal Migratory Orientation in Juvenile Sea Turtles*. The Journal of experimental biology, vol. 207, pages 1771–8, 06 2004. (Cité en page 17.)
- [Azouaoui & Achour 2014] O Azouaoui et Nouara Achour. *HS Combined Histogram for Visual Memory Building and Scene Recognition in Outdoor Environments*, 05 2014. (Cité en pages x, 54 et 55.)
- [Badino *et al.* 2012] H. Badino, D. Huber et T. Kanade. *Real-time topometric localization*. In 2012 IEEE International Conference on Robotics and Automation, pages 1635–1642, May 2012. (Cité en page 14.)

- [Baluja 1996] S. Baluja. *Evolution of an artificial neural network based autonomous land vehicle controller*. IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics), vol. 26, no. 3, page 450–463, Jun 1996. (Cit  en page 22.)
- [Bar 2004] Moshe Bar. *Visual Objects in Context*. Nature reviews. Neuroscience, vol. 5, pages 617–29, 09 2004. (Cit  en page 112.)
- [Bay *et al.* 2006] Herbert Bay, Tinne Tuytelaars et Luc Van Gool. *SURF : Speeded up robust features*. volume 3951, pages 404–417, Graz, Austria, 07 2006. (Cit  en page 12.)
- [Belkaid *et al.* 2016] M. Belkaid, N. Cuperlier et P. Gaussier. *Combining local and global visual information in context-based neurorobotic navigation*. In 2016 International Joint Conference on Neural Networks (IJCNN), pages 4947–4954, July 2016. (Cit  en pages 11, 14, 64, 91, 112, 117, 124 et 125.)
- [Bellekens *et al.* 2015] Ben Bellekens, Vincent Spruyt, Raf Berkvens, Rudi Penne et Maarten Weyn. *A Benchmark Survey of Rigid 3D Point Cloud Registration Algorithms*. International Journal On Advances in Intelligent Systems, vol. 1, 06 2015. (Cit  en pages ix, 23 et 27.)
- [Bernardino & Santos-Victor 1998] Alexandre Bernardino et Jos  Santos-Victor. *Visual behaviours for binocular tracking*. Robotics and Autonomous Systems, vol. 25, pages 137–146, 11 1998. (Cit  en page 17.)
- [Beyeler *et al.* 2019] Michael Beyeler, Emily L. Rounds, Kristofor D. Carlson, Nikil Dutt et Jeffrey L. Krichmar. *Neural correlates of sparse coding and dimensionality reduction*. PLOS Computational Biology, vol. 15, no. 6, page e1006908, Jun 2019. (Cit  en page 89.)
- [Bl chliger *et al.* 2017] Fabian Bl chliger, Marius Fehr, Marcin Dymczyk, Thomas Schneider et Roland Siegwart. *Topomap : Topological Mapping and Navigation Based on Visual SLAM Maps*. 09 2017. (Cit  en pages ix et 24.)
- [Bojarski *et al.* 2016] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Praseon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, Xin Zhang, Jake Zhao et Karol Zieba. *End to End Learning for Self-Driving Cars*. CoRR, vol. abs/1604.07316, 2016. (Cit  en page 21.)
- [Bose & Recce 2001] Amitabha Bose et Michael Recce. *Phase precession and phase-locking of hippocampal pyramidal cells*. Hippocampus, vol. 11, pages 204 – 215, 01 2001. (Cit  en page 46.)
- [Braitenberg 1986] Valentino Braitenberg. *Vehicles : Experiments in Synthetic Psychology*. Philosophical Review, vol. 95, no. 1, pages 137–139, 1986. (Cit  en page 18.)
- [Bresson *et al.* 2017] Guillaume Bresson, Zayed Alsayed, Li Yu et S bastien Glaser. *Simultaneous Localization And Mapping A Survey of Current Trends in Autonomous Driving*. IEEE Transactions on Intelligent Vehicles, vol. 2, no. 3, 2017. (Cit  en page 27.)

- [Brooks 1986] R. Brooks. *A robust layered control system for a mobile robot*. IEEE Journal on Robotics and Automation, vol. 2, no. 1, page 14–23, 1986. (Cit  en page 19.)
- [Brown *et al.* 2003] J Brown, Bill Yates et Jeffrey Taube. *Does the vestibular system contribute to head direction cell activity in the rat ?* Physiology and behavior, vol. 77, pages 743–8, 01 2003. (Cit  en page 47.)
- [Brun *et al.* 2008] Vegard Heimly Brun, Trygve Solstad, Kirsten Brun Kjelstrup, Marianne Fyhn, Menno P. Witter, Edvard I. Moser et May-Britt Moser. *Progressive increase in grid scale from dorsal to ventral medial entorhinal cortex*. Hippocampus, vol. 18, no. 12, page 1200–1212, Dec 2008. (Cit  en pages 49 et 126.)
- [Buzsaki 2006] G Buzsaki. Rhythms of the brain. Oxford University Press, Oxford New York, 2006. (Cit  en page 44.)
- [Cadena *et al.* 2016] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid et J. J. Leonard. *Past, Present, and Future of Simultaneous Localization and Mapping : Toward the Robust-Perception Age*. IEEE Transactions on Robotics, vol. 32, no. 6, pages 1309–1332, 2016. (Cit  en page 27.)
- [Cardoso *et al.* 2017] Vinicius Cardoso, Josias Oliveira, Thomas Teixeira, Claudine Badue, Filipe Mutz, Thiago Oliveira-Santos, Lucas Veronese et Alberto De Souza. *A Model-Predictive Motion Planner for the IARA autonomous car*. pages 225–230, 05 2017. (Cit  en page 109.)
- [Cartwright & Collett 1982] B. A. Cartwright et T. S. Collett. *How honey bees use landmarks to guide their return to a food source*. Nature, vol. 295, no. 5850, page 560–564, Feb 1982. (Cit  en page 17.)
- [Chance *et al.* 1999] Frances S Chance, Sacha B Nelson et Larry F Abbott. *Complex cells as cortically amplified simple cells*. Nature neuroscience, vol. 2, no. 3, pages 277–282, 1999. (Cit  en page 41.)
- [Chen *et al.* 2011] Shengyong Chen, Youfu Li et Ngai Kwok. *Active Vision in Robotic Systems : A Survey of Recent Developments*. I. J. Robotic Res., vol. 30, pages 1343–1377, 10 2011. (Cit  en page 11.)
- [Chen *et al.* 2014] Zetao Chen, Obadiah Lam, Adam Jacobson et Michael Milford. *Convolutional Neural Network-based Place Recognition*. 11 2014. (Cit  en pages x, 50 et 51.)
- [Chen *et al.* 2015] Zetao Chen, Stephanie Lowry, Adam Jacobson, Michael E. Haselmo et Michael Milford. *Bio-inspired homogeneous multi-scale place recognition*. Neural Networks, vol. 72, pages 48 – 61, 2015. Neurobiologically Inspired Robotics : Enhanced Autonomy through Neuromorphic Cognition. (Cit  en page 25.)
- [Cheng 1954] Ken Cheng. *How to Navigate without Maps : The Power of Taxon-like Navigation in Ants*. Comparative Cognition and Behavior Reviews, vol. 7, 01 1954. (Cit  en page 20.)

- [Choi *et al.* 2005] Jinwoo Choi, Sunghwan Ahn et Wan Chung. *Robust sonar feature detection for the SLAM of mobile robot*. pages 3415 – 3420, 09 2005. (Cité en page 27.)
- [Christian *et al.* 2010] Siagian Christian, Chang Chin-Kai, Voorhies Randolph et Itti Laurent. *Beobot 2.0 : Cluster architecture for mobile robotics*. Journal of Field Robotics, vol. 28, no. 2, pages 278–302, 2010. (Cité en pages xi, 57, 58, 59, 88 et 91.)
- [Churchill & Newman 2013] Winston Churchill et Paul Newman. *Experience-based navigation for long-term localisation*. The International Journal of Robotics Research, vol. 32, no. 14, pages 1645–1661, 2013. (Cité en page 13.)
- [Colomi *et al.* 1991] Alberto Colomi, Marco Dorigo et Vittorio Maniezzo. *Distributed Optimization by Ant Colonies*. 01 1991. (Cité en page 20.)
- [Cummins & Newman 2009] Mark Cummins et Paul Newman. *Highly scalable appearance-only SLAM - FAB-MAP 2.0*. 06 2009. (Cité en pages 51, 80 et 85.)
- [Cuperlier *et al.* 2006] Nicolas Cuperlier, Mathias Quoy, Christophe Giovannangeli, Philippe Gaussier et Philippe Laroque. *Transition cells for navigation and planning in an unknown environment*. In International Conference on Simulation of Adaptive Behavior, pages 286–297. Springer Berlin Heidelberg, 2006. (Cité en page 132.)
- [Cuperlier *et al.* 2007] Nicolas Cuperlier, Mathias Quoy et Philippe Gaussier. *Neurobiologically inspired mobile robot navigation and planning*. Frontiers in Neurobotics, vol. 1, page 3, 2007. (Cité en page 110.)
- [Curcio *et al.* 1990] Christine A. Curcio, Kenneth R. Sloan, Robert E. Kalina et Anita E. Hendrickson. *Human photoreceptor topography*. The Journal of Comparative Neurology, vol. 292, no. 4, page 497–523, Feb 1990. (Cité en page 34.)
- [Dacke *et al.* 2013] Marie Dacke, Emily Baird, Marcus Byrne, Clarke H. Scholtz et Eric J. Warrant. *Dung Beetles Use the Milky Way for Orientation*. Current Biology, vol. 23, no. 4, page 298–300, Feb 2013. (Cité en page 17.)
- [de Rengerve *et al.* 2015] Antoine de Rengerve, Pierre Andry et Philippe Gaussier. *On-line Learning and Control of Attraction Basins for the Development of Sensorimotor Control Strategies*. Biological cybernetics, vol. 109, 01 2015. (Cité en pages 94 et 110.)
- [Delarboulas *et al.* 2014] Pierre Delarboulas, Philippe Gaussier, Ramesh Caussy et Mathias Quoy. *Robustness Study of a Multimodal Compass Inspired from HD-Cells and Dynamic Neural Fields*. In Angel P. del Pobil, Eris Chinellato, Ester Martinez-Martin, John Hallam, Enric Cervera et Antonio Morales, editeurs, From Animals to Animats 13, pages 132–143, Castellón, Spain, 2014. Springer International Publishing. (Cité en page 66.)

- [Dijkstra 1959] E. W. Dijkstra. *A note on two problems in connexion with graphs*. Numerische Mathematik, vol. 1, no. 1, page 269–271, Dec 1959. (Cité en pages 7 et 132.)
- [Do & Yau 2010] Michael Tri Hoang Do et King-Wai Yau. *Intrinsically Photosensitive Retinal Ganglion Cells*. Physiological Reviews, vol. 90, no. 4, page 1547–1581, Oct 2010. (Cité en page 36.)
- [Douze *et al.* 2009] Matthijs Douze, Hervé Jégou, Harsimrat Sandhawalia, Laurent Amsaleg et Cordelia Schmid. *Evaluation of GIST descriptors for web-scale image search*. International Conference on Image and Video Retrieval, 07 2009. (Cité en page 113.)
- [Duckett *et al.* 2002] Tom Duckett, Stephen Marsland et Jonathan Shapiro. *Fast, On-Line Learning of Globally Consistent Maps*, 2002. (Cité en page 57.)
- [Dudchenko *et al.* 1997] Paul Dudchenko, Jeremy Goodridge et Jeffrey Taube. *The effects of disorientation on visual landmark control of head direction cell orientation*. Experimental brain research. Experimentelle Hirnforschung. Experimentation cérébrale, vol. 115, pages 375–80, 07 1997. (Cité en page 45.)
- [Durrant-whyte & Bailey] Hugh Durrant-whyte et Tim Bailey. *1Simultaneous Localisation and Mapping (SLAM) : Part I The Essential Algorithms*. (Cité en page 27.)
- [Elinas & Little 2007] Pantelis Elinas et James J. Little. *Stereo vision SLAM : Near real-time learning of 3 D point-landmark and 2 D occupancy-grid maps using particle filters*. 2007. (Cité en page 23.)
- [Engelhard *et al.* 2011] Nikolas Engelhard, Felix Endres, Jürgen Hess, Jürgen Sturm et Wolfram Burgard. *Real-time 3D visual SLAM with a hand-held RGB-D camera*. In Proc. of the RGB-D Workshop on 3D Perception in Robotics at the European Robotics Forum, Vasteras, Sweden, volume 180, pages 1–15, 2011. (Cité en page 27.)
- [Erskine & Herrera 2014] Lynda Erskine et Eloisa Herrera. *Connecting the Retina to the Brain*. ASN Neuro, vol. 6, no. 6, page 175909141456210, Nov 2014. (Cité en page 35.)
- [Espada *et al.* 2018] Y. Espada, N. Cuperlier, G. Bresson et O. Romain. *Application of a Bio-inspired Localization Model to Autonomous Vehicles*. In 2018 International Conference on Control, Automation, Robotics and Vision (ICARCV), 2018. (Cité en pages xi, xii, 62, 63, 65, 66, 67, 68, 74, 75 et 88.)
- [Fattori *et al.* 2009] Patrizia Fattori, Sabrina Pitzalis et Claudio Galletti. *The cortical visual area V6 in macaque and human brains*. Journal of Physiology-Paris, vol. 103, no. 1–2, page 88–97, Jan 2009. (Cité en page 38.)
- [Fernández-Caballero *et al.* 2010] Antonio Fernández-Caballero, José Castillo, Javier Martínez-Cantos et Rafael Martínez-Tomás. *Optical flow or image subtraction in human detection from infrared camera on mobile robot*. Robo-

- tics and Autonomous Systems, vol. 58, pages 1273–1281, 12 2010. (Cit  en page 12.)
- [Floreano *et al.* 2008] Dario Floreano, Peter D rr et Claudio Mattiussi. *Neuroevolution : from architectures to learning*. Evolutionary Intelligence, vol. 1, pages 47–62, 2008. (Cit  en page 21.)
- [Fox *et al.* 1999] Dieter Fox, Wolfram Burgard, Frank Dellaert et Sebastian Thrun. *Monte Carlo Localization : Efficient Position Estimation for Mobile Robots*, 01 1999. (Cit  en page 57.)
- [Franz *et al.* 2000] V.H. Franz, K.R. Gegenfurtner, H.H. B lthoff et M. Fahle. *Grasping Visual Illusions : No Evidence for a Dissociation Between Perception and Action*. Psychological Science, vol. 11, no. 1, page 20–25, Jan 2000. (Cit  en page 42.)
- [Fremont 2009] Vincent Fremont. *Odom trie 3D vision/lidar pour les v hicules intelligents*. In Journ es Nationales de la Recherche en Robotique, JNRR’09, Neuvy-sur-Barangeon, France, Novembre 2009. (Cit  en page 27.)
- [Fyhn *et al.* 2007] Marianne Fyhn, Torkel Hafting, Alessandro Treves, May-Britt Moser et Edvard I. Moser. *Hippocampal remapping and grid realignment in entorhinal cortex*. Nature, vol. 446, no. 7132, page 190–194, Feb 2007. (Cit  en page 49.)
- [F ldi k 1990] P. F ldi k. *Forming sparse representations by local anti-Hebbian learning*. Biological Cybernetics, vol. 64, no. 2, page 165–170, Dec 1990. (Cit  en pages 89 et 129.)
- [G. Carpenter & Rosen 1991] S. Grossberg G. Carpenter et D. Rosen. *Fuzzy ART : Fast Stable Learning and Categorization of Analog Patterns by an Adaptive Resonance System*. In 1991 IEEE International Conference on Robotics and Biomimetics (ROBIO), volume 4, pages 759–771, 1991. (Cit  en page 54.)
- [Gandhi & Trivedi 2005] Tarak Gandhi et Mohan Trivedi. *Parametric ego-motion estimation for vehicle surround analysis using an omnidirectional camera*. Machine Vision and Applications, vol. 16, no. 2, page 85–95, Feb 2005. (Cit  en page 11.)
- [Ganel & Goodale 2003] Tzvi Ganel et Melvyn Goodale. *Visual control of action but not perception requires analytical processing of object shape*. Nature, vol. 426, pages 664–667, 12 2003. (Cit  en page 42.)
- [Garcia-Fidalgo & Ortiz 2015] Emilio Garcia-Fidalgo et Alberto Ortiz. *Vision-based topological mapping and localization methods : A survey*. Robotics and Autonomous Systems, vol. 64, page 1–20, Feb 2015. (Cit  en page 14.)
- [Gaussier & Zrehen 1995] Philippe Gaussier et St phane Zrehen. *PerAc : A neural architecture to control artificial animals*. Robotics and Autonomous Systems, vol. 16, no. 2, pages 291 – 320, 1995. Moving the Frontiers between Robotics and Biology. (Cit  en pages 93, 94 et 103.)

- [Gaussier *et al.* 2002a] Philippe Gaussier, Arnaud Revel, Jean Banquet et Vincent Babeau. *From view cells and place cells to cognitive map learning : Processing stages of the hippocampal system*. Biological cybernetics, vol. 86, pages 15–28, 02 2002. (Cité en page 46.)
- [Gaussier *et al.* 2002b] Philippe Gaussier, Arnaud Revel, Jean Banquet et Vincent Babeau. *From view cells and place cells to cognitive map learning : Processing stages of the hippocampal system*. vol. 86, pages 15–28, 02 2002. (Cité en pages 68 et 91.)
- [Geiger *et al.* 2012] Andreas Geiger, Philip Lenz et Raquel Urtasun. *Are we ready for Autonomous Driving ? The KITTI Vision Benchmark Suite*. In Conference on Computer Vision and Pattern Recognition (CVPR), 2012. (Cité en page 73.)
- [Gener *et al.* 2013] Thomas Gener, Lorena Perez-Mendez et Maria Sanchez-Vives. *Tactile Modulation of Hippocampal Place Fields*. Hippocampus, vol. 23, 12 2013. (Cité en page 46.)
- [Georges-François *et al.* 1999] Pierre Georges-François, Edmund T. Rolls et Robert G. Robertson. *Spatial View Cells in the Primate Hippocampus : Allocentric View not Head Direction or Eye Position or Place*. Cerebral Cortex, vol. 9, no. 3, pages 197–212, 04 1999. (Cité en page 46.)
- [Georgopoulos *et al.* 1988] AP Georgopoulos, RE Kettner et AB Schwartz. *Primate motor cortex and free arm movements to visual targets in three- dimensional space. II. Coding of the direction of movement by a neuronal population*. Journal of Neuroscience, vol. 8, no. 8, pages 2928–2937, 1988. (Cité en page 66.)
- [Giovannangeli & Gaussier 2007] Christophe Giovannangeli et Philippe Gaussier. *Orientation system in robots : Merging allothetic and idiothetic estimations*. 13th International Conference on Advanced Robotics (ICAR07), 01 2007. (Cité en page 66.)
- [Giovannangeli *et al.* 2006a] C. Giovannangeli, P. Gaussier et J.P. Banquet. *Robustness of Visual Place Cells in Dynamic Indoor and Outdoor Environment*. International Journal of Advanced Robotic Systems, vol. 3, no. 2, page 19, 2006. (Cité en page 103.)
- [Giovannangeli *et al.* 2006b] C. Giovannangeli, Ph. Gaussier et J. P. Banquet. *Robustness of Visual Place Cells in Dynamic Indoor and Outdoor Environment*. International Journal of Advanced Robotic Systems, vol. 3, no. 2, pages 115–124, Juin 2006. (Cité en page 64.)
- [Gonzalez Bautista *et al.* 2014] David Gonzalez Bautista, Joshué Pérez, Ray Lattarulo, Vicente Milanés et Fawzi Nashashibi. *Continuous curvature planning with obstacle avoidance capabilities in urban scenarios*. 2014 17th IEEE International Conference on Intelligent Transportation Systems, ITSC 2014, 10 2014. (Cité en page 109.)
- [Gonzalez Bautista *et al.* 2015] David Gonzalez Bautista, Joshué Pérez, Vicente Milanés et Fawzi Nashashibi. *A Review of Motion Planning Techniques for Au-*

- tomated Vehicles*. IEEE Transactions on Intelligent Transportation Systems, pages 1–11, 11 2015. (Cité en page 7.)
- [Goodale & Milner 1992] Melvyn A. Goodale et A. David Milner. *Separate visual pathways for perception and action*. Trends in Neurosciences, vol. 15, no. 1, pages 20 – 25, 1992. (Cité en pages 41 et 61.)
- [Grieves & Jeffery 2016] Roddy Grieves et Kate Jeffery. *The representation of space in the brain*. Behavioural Processes, pages –, 12 2016. (Cité en pages x, 44, 45 et 49.)
- [Grisetti *et al.* 2010] G Grisetti, R Kummerle, C Stachniss et W Burgard. *A Tutorial on Graph-Based SLAM*. IEEE Intelligent Transportation Systems Magazine, vol. 2, no. 4, page 31–43, 2010. (Cité en page 24.)
- [Grossberg 1973] Stephen Grossberg. *Contour enhancement, short-term memory, and constancies in reverberating neural networks*. In Studies in Applied Math, pages 213–257, 1973. (Cité en page 69.)
- [Guidolini *et al.* 2017] Ranik Guidolini, Alberto De Souza, Filipe Mutz et Claudine Badue. *Neural-based model predictive control for tackling steering delays of autonomous cars*. pages 4324–4331, 05 2017. (Cité en page 109.)
- [Hafting *et al.* 2005] Torkel Hafting, Marianne Fyhn, Sturla Molden, May-Britt Moser et Edvard I Moser. *Microstructure of a spatial map in the entorhinal cortex*. vol. 436, pages 801–6, 09 2005. (Cité en pages 48 et 49.)
- [Haggerty *et al.* 2015] Daniel Christopher Haggerty, Daoyun Ji et Howard Eichenbaum. *Activities of visual cortical and hippocampal neurons co-fluctuate in freely moving rats during spatial behavior*. In eLife, 2015. (Cité en page 41.)
- [Harootonian *et al.* 2020] Sevan K. Harootonian, Robert C. Wilson, Lukáš Hejtmánek, Eli M. Ziskin et Arne D. Ekstrom. *Path integration in large-scale space and with novel geometries : Comparing vector addition and encoding-error models*. PLOS Computational Biology, vol. 16, no. 5, page e1007489, May 2020. (Cité en page 20.)
- [Hart *et al.* 1968] P. E. Hart, N. J. Nilsson et B. Raphael. *A Formal Basis for the Heuristic Determination of Minimum Cost Paths*. IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pages 100–107, July 1968. (Cité en page 7.)
- [Hazama & Tamura 2019] Yutaro Hazama et Ryoji Tamura. *Effects of self-locomotion on the activity of place cells in the hippocampus of a freely behaving monkey*. Neuroscience Letters, vol. 701, page 32–37, May 2019. (Cité en page 45.)
- [He *et al.* 2015] Xf He, Lilian Zhang, Junxiang Lian et Xiaoping Hu. *Vision/Odometer Autonomous Navigation Based on RatSLAM for Land Vehicles*, 01 2015. (Cité en page 53.)
- [Horton *et al.* 2011] Travis W Horton, Richard N Holdaway, Alexandre N Zerbini, Nan Hauser, Claire Garrigue, Artur Andriolo et Phillip J Clapham. *Straight*

- as an arrow : humpback whales swim constant course tracks during long-distance migration*. *Biology letters*, vol. 7, no. 5, pages 674–679, 2011. (Cité en page 30.)
- [Hubel & Wiesel 1959] D. H. Hubel et T. N. Wiesel. *Receptive fields of single neurons in the cat's striate cortex*. *The Journal of Physiology*, vol. 148, no. 3, page 574–591, Oct 1959. (Cité en pages 40 et 41.)
- [Hubel & Wiesel 1962a] D. H. Hubel et T. N. Wiesel. *Receptive fields, binocular interaction and functional architecture in the cat's visual cortex*. *The Journal of Physiology*, vol. 160, no. 1, page 106–154, Jan 1962. (Cité en pages 40 et 41.)
- [Hubel & Wiesel 1962b] D. H. Hubel et T. N. Wiesel. *Receptive fields, binocular interaction and functional architecture in the cat's visual cortex*. *The Journal of Physiology*, vol. 160, no. 1, page 106–154, Jan 1962. (Cité en page 50.)
- [Jaakkola & Haussler 1999] Tommi Jaakkola et David Haussler. *Exploiting Generative Models in Discriminative Classifiers*. In M. J. Kearns, S. A. Solla et D. A. Cohn, éditeurs, *Advances in Neural Information Processing Systems 11*, pages 487–493. MIT Press, 1999. (Cité en page 13.)
- [Jacob *et al.* 2014] Pierre-Yves Jacob, Bruno Poucet, Martine Liberge, Etienne Save et Francesca Sargolini. *Vestibular control of entorhinal cortex activity in spatial navigation*. *Frontiers in integrative neuroscience*, vol. 8, page 38, 06 2014. (Cité en page 46.)
- [Jauffret *et al.* 2013] A. Jauffret, M. Belkaid, N. Cuperlier, P. Gaussier et P. Tarroux. *Frustration as a way toward autonomy and self-improvement in robotic navigation*. In 2013 IEEE Third Joint International Conference on Development and Learning and Epigenetic Robotics (ICDL), pages 1–7, Aug 2013. (Cité en pages 11, 133 et 134.)
- [Jauffret *et al.* 2015] Adrien Jauffret, Nicolas Cuperlier et Philippe Gaussier. *From grid cells and visual place cells to multimodal place cell : a new robotic architecture*. *Frontiers in Neurorobotics*, vol. 9, page 1, 2015. (Cité en page 132.)
- [Jauffret 2014] Adrien Jauffret. *From self-evaluation to emotions : neuromimetic and bayesian approaches for the learning of complex behavior involving multimodal informations*. Theses, Université Paris Sud - Paris XI, Juillet 2014. (Cité en pages xiii, 96 et 97.)
- [Jeffery *et al.* 2004] Kate Jeffery, Michael Anderson, R. Hayman et Subhojit Chakraborty. *A proposed architecture for the neural representation of spatial context*. *Neuroscience and biobehavioral reviews*, vol. 28, pages 201–18, 05 2004. (Cité en page 46.)
- [Jensfelt 2009] Patric Jensfelt. *Approaches to Mobile Robot Localization in Indoor Environments*. 04 2009. (Cité en page 23.)
- [Jiachen *et al.* 2018] Zhang Jiachen, Dahang Ai, Yi Xiang, Xin Chang, Yi Wang et Xiaodong Chen. *Bag-of-words based loop-closure detection in visual SLAM*.

- In Xiao-Cong Yuan, Kebin Shi et Michael G. Somekh, éditeurs, *Advanced Optical Imaging Technologies*, volume 10816, pages 104 – 111. International Society for Optics and Photonics, SPIE, 2018. (Cité en page 13.)
- [Johnson *et al.* 1997] Paul J. Johnson, Kevin L. Chapman et John S. Bay. *Naviga-tion of an autonomous ground vehicle using the subsumption architecture*. In Chase H. Kenyon et Pushkin Kachroo, éditeurs, *Mobile Robots XI and Au-tomated Vehicle Control Systems*, volume 2903, pages 54 – 62. International Society for Optics and Photonics, SPIE, 1997. (Cité en page 20.)
- [Johnson *et al.* 2004] Mark Johnson, Peter T Madsen, Walter MX Zimmer, Natacha Aguilar de Soto et Peter L Tyack. *Beaked whales echolocate on prey*. *Proceedings of the Royal Society of London. Series B : Biological Sciences*, vol. 271, no. suppl_6, pages S383–S386, 2004. (Cité en page 30.)
- [Jones & Holderied 2007] Gareth Jones et Marc W Holderied. *Bat echolocation calls : adaptation and convergent evolution*. *Proceedings of the Royal So-ciety B : Biological Sciences*, vol. 274, no. 1612, pages 905–912, 2007. (Cité en page 30.)
- [Jung *et al.* 1995] Min Whan Jung, Sidney Wiener et Bruce Mcnaughton. *Compari-son of spatial firing characteristics of units in dorsal and ventral hippocampus of the rat*. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 14, pages 7347–56, 01 1995. (Cité en page 45.)
- [Kalman & Bucy 1961] R. E. Kalman et R. S. Bucy. *New results in linear filtering and prediction theory*. *Transactions of the ASME. Series D, Journal of Basic Engineering*, vol. 83, pages 95–108, 1961. (Cité en page 26.)
- [Kendall *et al.* 2018] Alex Kendall, Jeffrey Hawke, David Janz, Przemyslaw Mazur, Daniele Reda, John-Mark Allen, Vinh-Dieu Lam, Alex Bewley et Amar Shah. *Learning to Drive in a Day*, 07 2018. (Cité en page 21.)
- [Kim *et al.* 2018] Pileun Kim, Jingdao Chen et Yong K. Cho. *SLAM-driven robotic mapping and registration of 3D point clouds*. *Automation in Construction*, vol. 89, pages 38 – 48, 2018. (Cité en page 27.)
- [Kjelstrup *et al.* 2008] Kirsten Brun Kjelstrup, Trygve Solstad, Vegard Heimly Brun, Torkel Hafting, Stefan Leutgeb, Menno P. Witter, Edvard I. Moser et May-Britt Moser. *Finite Scale of Spatial Representation in the Hippocam-pus*. *Science*, vol. 321, no. 5885, pages 140–143, 2008. (Cité en page 45.)
- [Kleinfeld & Deschênes 2011] David Kleinfeld et Martin Deschênes. *Neuronal basis for object location in the vibrissa scanning sensorimotor system*. *Neuron*, vol. 72, no. 3, pages 455–468, 2011. (Cité en page 30.)
- [Ko *et al.* 2013] D. W. Ko, C. Yi et I. H. Suh. *Semantic mapping and navigation : A Bayesian approach*. In 2013 IEEE/RSJ International Conference on Intel-ligent Robots and Systems, pages 2630–2636, Nov 2013. (Cité en page 25.)
- [Koch & Ullman 1987] Christof Koch et Shimon Ullman. *Shifts in selective visual attention : Towards the underlying neural circuitry*, pages 115–141. Springer Netherlands, Dordrecht, 1987. (Cité en page 64.)

- [Konolige *et al.* 2011] K. Konolige, E. Marder-Eppstein et B. Marthi. *Navigation in hybrid metric-topological maps*. In 2011 IEEE International Conference on Robotics and Automation, pages 3041–3047, 2011. (Cit  en page 25.)
- [Krauzlis *et al.* 2017] Richard Krauzlis, Laurent Goffart et Ziad Hafed. *Neuronal control of fixation and fixational eye movements*. Philosophical Transactions of the Royal Society B : Biological Sciences, vol. 372, page 20160205, 04 2017. (Cit  en page 37.)
- [Kropff *et al.* 2015] Emilio Kropff, James Carmichael, May-Britt Moser et Edvard I Moser. *Speed cells in the medial entorhinal cortex*. vol. 523, 07 2015. (Cit  en page 49.)
- [Kuipers *et al.* 2004] B. Kuipers, J. Modayil, P. Beeson, M. MacMahon et F. Savelli. *Local metrical and global topological maps in the hybrid spatial semantic hierarchy*. In IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004, volume 5, pages 4845–4851 Vol.5, April 2004. (Cit  en page 25.)
- [Laram e & Boire 2015] Marie-Eve Laram e et Denis Boire. *Visual cortical areas of the mouse : comparison of parcellation and network structure with primates*. Frontiers in Neural Circuits, vol. 8, Jan 2015. (Cit  en page 31.)
- [Las & Ulanovsky 2014] Liora Las et Nachum Ulanovsky. *Hippocampal neurophysiology across species*, page 431–461. Springer Vienna, 2014. (Cit  en page 45.)
- [LaValle & Kuffner 2001] Steven LaValle et James Kuffner. *Randomized Kinodynamic Planning*. I. J. Robotic Res., vol. 20, pages 378–400, 01 2001. (Cit  en page 109.)
- [Leng *et al.* 2019] Chengcai Leng, Hai Zhang, Bo Li, Guorong Cai, Zhao Pei et Li He. *Local Feature Descriptor for Image Matching : A Survey*. IEEE Access, vol. 7, pages 6424–6434, 2019. (Cit  en page 14.)
- [Li *et al.* 2018] Jie Li, Michael Kaess, Ryan Eustice et Matthew Johnson-Roberson. *Pose-graph SLAM using Forward-looking Sonar*. IEEE Robotics and Automation Letters, RA-L, vol. 3, no. 3, pages 2330–2337, July 2018. (Cit  en page 27.)
- [Lindeberg 2010] Tony Lindeberg. *Generalized Gaussian Scale-Space Axiomatics Comprising Linear Scale-Space, Affine Scale-Space and Spatio-Temporal Scale-Space*. Journal of Mathematical Imaging and Vision, vol. 40, no. 1, page 36–81, Dec 2010. (Cit  en page 40.)
- [Lowe 2004a] David G. Lowe. *Distinctive Image Features from Scale-Invariant Keypoints*. Int. J. Comput. Vision, vol. 60, no. 2, pages 91–110, Novembre 2004. (Cit  en page 12.)
- [Lowe 2004b] David G. Lowe. *Distinctive Image Features from Scale-Invariant Keypoints*. International Journal of Computer Vision, vol. 60, no. 2, pages 91–110, Nov 2004. (Cit  en page 57.)

- [Lu & Milios 1997] F. Lu et E. Milios. *Globally Consistent Range Scan Alignment for Environment Mapping*. AUTONOMOUS ROBOTS, vol. 4, pages 333–349, 1997. (Cit  en page 24.)
- [Lu et al. 2020] Jiaxin Lu, Mai Xu, Ren Yang et Zulin Wang. *Understanding and Predicting the Memorability of Outdoor Natural Scenes*. IEEE Transactions on Image Processing, vol. 29, page 4927–4941, 2020. (Cit  en page 90.)
- [Madl et al. 2014] Tamas Madl, Stan Franklin, Ke Chen, Daniela Montaldi et Robert Trapp. *Bayesian Integration of Information in Hippocampal Place Cells*. PLOS ONE, vol. 9, no. 3, pages 1–16, 03 2014. (Cit  en page 46.)
- [Martin et al. 2018] Jacob Martin, Charles Davis, Maximilian Riesenhuber et Simon Thorpe. *Zapping 500 faces in less than 100 seconds : Evidence for extremely fast and sustained continuous visual search*. Scientific Reports, vol. 8, 12 2018. (Cit  en page 37.)
- [Masland 2001] Richard H. Masland. *The fundamental plan of the retina*. Nature Neuroscience, vol. 4, no. 9, page 877–886, Sep 2001. (Cit  en page 35.)
- [Masland 2012a] Richard H. Masland. *The Neuronal Organization of the Retina*. Neuron, vol. 76, no. 2, page 266–280, Oct 2012. (Cit  en page 35.)
- [Masland 2012b] Richard H. Masland. *The tasks of amacrine cells*. Visual Neuroscience, vol. 29, no. 1, page 3–9, Jan 2012. (Cit  en page 35.)
- [Mei et al. 2009] Christopher Mei, Gabe Sibley, Mark Cummins, Paul Newman et Ian Reid. *A constant time efficient stereo SLAM system*. 01 2009. (Cit  en page 13.)
- [Miessner 1912] B.F. Miessner. *The wirelessly directed torpedo. Some new experiments in an old field*. Scientific American, 1912. (Cit  en page 5.)
- [Milford & Wyeth 2008] M. J. Milford et G. F. Wyeth. *Mapping a Suburb With a Single Camera Using a Biologically Inspired SLAM System*. IEEE Transactions on Robotics, vol. 24, no. 5, pages 1038–1053, Oct 2008. (Cit  en pages x, 53, 57, 80 et 88.)
- [Milford & Wyeth 2010] Michael Milford et Gordon Wyeth. *Persistent Navigation and Mapping using a Biologically Inspired SLAM System*. The International Journal of Robotics Research, vol. 29, no. 9, pages 1131–1153, 2010. (Cit  en pages x, 52 et 53.)
- [Milford & Wyeth 2012] M. J. Milford et G. F. Wyeth. *SeqSLAM : Visual route-based navigation for sunny summer days and stormy winter nights*. In 2012 IEEE International Conference on Robotics and Automation, pages 1643–1649, May 2012. (Cit  en page 51.)
- [Milford et al. 2004] M. J. Milford, G. F. Wyeth et D. Prasser. *RatSLAM : a hippocampal model for simultaneous localization and mapping*. In Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on, volume 1, pages 403–408 Vol.1, April 2004. (Cit  en page 52.)

- [Miyasaka *et al.* 2009] T. Miyasaka, Y. Ohama et Y. Ninomiya. *Ego-motion estimation and moving object tracking using multi-layer LIDAR*. In 2009 IEEE Intelligent Vehicles Symposium, pages 151–156, June 2009. (Cit  en page 27.)
- [Mnih *et al.* 2015] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei Rusu, Joel Veness, Marc Bellemare, Alex Graves, Martin Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dhharshan Kumaran, Daan Wierstra, Shane Legg et Demis Hassabis. *Human-level control through deep reinforcement learning*. Nature, vol. 518, pages 529–33, 02 2015. (Cit  en page 21.)
- [Mo & Xu 2010] Y. Mo et Y. Xu. *A Navigation System Based on Subsumption Architecture*. In 2010 International Conference on Measuring Technology and Mechatronics Automation, volume 1, pages 161–164, March 2010. (Cit  en page 19.)
- [Moravec & Elfes 1985] Hans Moravec et Alberto Elfes. *High resolution maps from wide angle sonar*. volume 2, pages 116 – 121, 04 1985. (Cit  en page 24.)
- [Mouritsen 2001] Henrik Mouritsen. *Navigation in birds and other animals*. Image and Vision Computing, vol. 19, no. 11, pages 713–731, 2001. (Cit  en page 30.)
- [Muller & Kubie 1987] RU Muller et JL Kubie. *The effects of changes in the environment on the spatial firing of hippocampal complex-spike cells*. The Journal of Neuroscience, vol. 7, no. 7, page 1951–1968, Jul 1987. (Cit  en page 46.)
- [M ller & Wehner 1988] Martin M ller et R diger Wehner. *Path integration in desert ants, *Cataglyphis fortis**. Proceedings of the National Academy of Sciences, vol. 85, no. 14, pages 5287–5290, 1988. (Cit  en page 20.)
- [Mur-Artal *et al.* 2015] Raul Mur-Artal, J. Montiel et Juan Tardos. *ORB-SLAM : a versatile and accurate monocular SLAM system*. IEEE Transactions on Robotics, vol. 31, pages 1147 – 1163, 10 2015. (Cit  en page 13.)
- [Myung-Jin Jung *et al.* 2004] Myung-Jin Jung, Hyun Myung, Sun-Gi Hong, Dong-Ryeol Park, Hyoung-Ki Lee et SeokWon Bang. *Structured light 2D range finder for simultaneous localization and map-building (SLAM) in home environments*. In Micro-Nanomechatronics and Human Science, 2004 and The Fourth Symposium Micro-Nanomechatronics for Information-Based Society, 2004., pages 371–376, Oct 2004. (Cit  en page 27.)
- [Nandy *et al.* 2017] Anirvan S. Nandy, Jonathan J. Nassi et John H. Reynolds. *Laminar Organization of Attentional Modulation in Macaque Visual Area V4*. Neuron, vol. 93, no. 1, page 235–246, Jan 2017. (Cit  en page 41.)
- [Navon 1977] David Navon. *Forest before trees : The precedence of global features in visual perception*. Cognitive Psychology, vol. 9, no. 3, pages 353 – 383, 1977. (Cit  en page 112.)
- [Nieuwenhuys 2008] R Nieuwenhuys. The human central nervous system. Springer, Berlin New York, 2008. (Cit  en pages ix, 38 et 39.)

- [Norman 2002] Joel Norman. *Two visual systems and two theories of perception : An attempt to reconcile the constructivist and ecological approaches*. Behavioral and Brain Sciences, vol. 25, no. 1, page 73–96, 2002. (Cité en page 42.)
- [O’Keefe & Dostrovsky 1971] J. O’Keefe et J. Dostrovsky. *The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat*. Brain Research, vol. 34, no. 1, pages 171 – 175, 1971. (Cité en pages 43 et 45.)
- [Oliva & Torralba 2001] Aude Oliva et Antonio Torralba. *Modeling the Shape of the Scene : A Holistic Representation of the Spatial Envelope*. Int. J. Comput. Vision, vol. 42, no. 3, pages 145–175, Mai 2001. (Cité en pages 14, 57 et 113.)
- [Ozisk & Yavuz 2008] O. Ozisk et S. Yavuz. *An Occupancy Grid Based SLAM Method*. In 2008 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, pages 117–119, July 2008. (Cité en page 24.)
- [Paden & Frazzoli 2016] Brian Paden et Emilio Frazzoli. *A Generalized Label Correcting Method for Optimal Kinodynamic Motion Planning*. 07 2016. (Cité en page 109.)
- [Pahl *et al.* 2011] Mario Pahl, Hong Zhu, Juergen Tautz et Shaowu Zhang. *Large Scale Homing in Honeybees*. PloS one, vol. 6, page e19669, 05 2011. (Cité en page 18.)
- [Park *et al.* 2011] Eun Hye Park, Dino Dvorak et André Fenton. *Ensemble Place Codes in Hippocampus : CA1, CA3, and Dentate Gyrus Place Cells Have Multiple Place Fields in Large Environments*. PloS one, vol. 6, page e22349, 07 2011. (Cité en pages xv et 131.)
- [Passaglia *et al.* 2002] Christopher L Passaglia, John B Troy, Lukas Rüttiger et Barry B Lee. *Orientation sensitivity of ganglion cells in primate retina*. Vision Research, vol. 42, no. 6, page 683–694, Mar 2002. (Cité en page 36.)
- [Pesavento & Wang 2009] Umberto Pesavento et Z. Wang. *Flapping Wing Flight Can Save Aerodynamic Power Compared to Steady Flight*. Physical review letters, vol. 103, page 118102, 09 2009. (Cité en page 32.)
- [Pierzchała *et al.* 2018] Marek Pierzchała, Philippe Giguère et Rasmus Astrup. *Mapping forests using an unmanned ground vehicle with 3D LiDAR and graph-SLAM*. Computers and Electronics in Agriculture, vol. 145, pages 217 – 225, 2018. (Cité en page 27.)
- [Pink 2008] Oliver Pink. *Visual map matching and localization using a global feature map*. pages 1 – 7, 07 2008. (Cité en page 23.)
- [Pomerleau 1989] Dean A. Pomerleau. *ALVINN : An Autonomous Land Vehicle in a Neural Network*. In D. S. Touretzky, éditeur, Advances in Neural Information Processing Systems 1, pages 305–313. Morgan-Kaufmann, 1989. (Cité en page 21.)

- [Prince 2002] Christopher Prince. *Rodney A. Brooks, Cambrian Intelligence : The Early History of the New AI*, Cambridge, MA : MIT Press, 1999, *xii + 199 pp.*, ISBN 0-262-52263-2. Minds and Machines, vol. 12, pages 145–151, 02 2002. (Cité en page 19.)
- [Purves 2001] Dale Purves. Neuroscience. Sinauer Associates, Sunderland, Mass, 2001. (Cité en pages ix et 36.)
- [Ramon 1909] . cajal Ramon. Histologie du systeme nerveux de l’homme et des vertebres, volume 1. 1909. (Cité en pages ix et 43.)
- [Rebai *et al.* 2012] K. Rebai, O. Azouaoui et N. Achour. *Bio-inspired visual memory for robot cognitive map building and scene recognition*. In 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 2985–2990, Oct 2012. (Cité en pages 54 et 88.)
- [Redmon & Farhadi 2018] Joseph Redmon et Ali Farhadi. *YOLOv3 : An Incremental Improvement*. arXiv, 2018. (Cité en page 50.)
- [Rich *et al.* 2014] Patrick Rich, Hua-Peng Liaw et Albert Lee. *Place cells. Large environments reveal the statistical structure governing hippocampal representations*. Science (New York, N.Y.), vol. 345, pages 814–7, 08 2014. (Cité en pages xv, 129 et 130.)
- [Rodgers & Hore 2009] C. T. Rodgers et P. J. Hore. *Chemical magnetoreception in birds : The radical pair mechanism*. Proceedings of the National Academy of Sciences, vol. 106, no. 2, page 353–360, Jan 2009. (Cité en page 17.)
- [Rosten & Drummond 2006] Edward Rosten et Tom Drummond. *Machine Learning for High-Speed Corner Detection*. volume 3951, Graz, Austria, 05 2006. (Cité en page 12.)
- [Ruble *et al.* 2011] Ethan Rublee, Vincent Rabaud, Kurt Konolige et Gary Bradski. *ORB : an efficient alternative to SIFT or SURF*. pages 2564–2571, 11 2011. (Cité en page 13.)
- [Schölkopf *et al.* 1997] Bernhard Schölkopf, Alexander Smola et Klaus-Robert Müller. *Kernel principal component analysis*. In Wulfram Gerstner, Alain Germond, Martin Hasler et Jean-Daniel Nicoud, editeurs, Artificial Neural Networks — ICANN’97, pages 583–588, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg. (Cité en page 14.)
- [Schöner *et al.* 1995] G. Schöner, M. Dose et C. Engels. *Dynamics of behavior : Theory and applications for autonomous robot architectures*. Robotics and Autonomous Systems, vol. 16, no. 2, pages 213 – 245, 1995. Moving the Frontiers between Robotics and Biology. (Cité en pages xiii, 97 et 98.)
- [Scoville & Milner 2000] W Scoville et B Milner. *Scoville WB, Milner B. Loss of recent memory after bilateral hippocampal lesions*. J Neurol Neurosurg Psychiatry 20 : 11-21. The Journal of neuropsychiatry and clinical neurosciences, vol. 12, pages 103–13, 02 2000. (Cité en page 43.)

- [Sermanet *et al.* 2014] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus et Yann LeCun. *OverFeat : Integrated Recognition, Localization and Detection using Convolutional Networks*. CoRR, vol. abs/1312.6229, 2014. (Cité en page 50.)
- [Silveira *et al.* 2013] L. Silveira, F. Guth, P. Drews et S. Botelho. *3D robotic mapping : A biologic approach*. In 2013 16th International Conference on Advanced Robotics (ICAR), pages 1–6, Nov 2013. (Cité en page 53.)
- [Sivic & Zisserman 2003] Josef Sivic et Andrew Zisserman. *Video Google : A Text Retrieval Approach to Object Matching in Videos*. In IEEE International Conference on Computer Vision, volume 2, pages 1470–1477, 2003. (Cité en page 13.)
- [Smith *et al.* 2009] Paul Smith, Cynthia Darlington et Yiwen Zheng. *Move It or Lose It-Is Stimulation of the Vestibular System Necessary for Normal Spatial Memory ?* Hippocampus, vol. 20, pages 36–43, 01 2009. (Cité en page 46.)
- [Solomon & Rosa 2014] Samuel G. Solomon et Marcello G. P. Rosa. *A simpler primate brain : the visual system of the marmoset monkey*. Frontiers in Neural Circuits, vol. 8, Aug 2014. (Cité en page 38.)
- [Solstad *et al.* 2006] Trygve Solstad, Edvard Moser et Gaute Einevoll. *From grid cells to place cells : A mathematical model*. Hippocampus, vol. 16, pages 1026–31, 12 2006. (Cité en page 48.)
- [Stabell 1984] Ole B. Stabell. *Homing and Olfaction in Salmonids : a Critical Review with Special Reference to the Atlantic Salmon*. Biological Reviews, vol. 59, no. 3, pages 333–388, 1984. (Cité en pages 18 et 30.)
- [Stackman Jr *et al.* 2003] Robert Stackman Jr, Edward Golob, Joshua Bassett et Jeffrey Taube. *Passive Transport Disrupts Directional Path Integration by Rat Head Direction Cells*. Journal of neurophysiology, vol. 90, pages 2862–74, 12 2003. (Cité en page 47.)
- [Steckel & Peremans 2013] Jan Steckel et Herbert Peremans. *BatSLAM : Simultaneous Localization and Mapping Using Biomimetic Sonar*. PLOS ONE, vol. 8, no. 1, pages 1–11, 01 2013. (Cité en page 53.)
- [Strasdat *et al.* 2012] Hauke Strasdat, J.M.M. Montiel et Andrew J. Davison. *Visual SLAM : Why filter ?* Image and Vision Computing, vol. 30, no. 2, pages 65 – 77, 2012. (Cité en page 27.)
- [Strecha *et al.* 2012] C. Strecha, V. Lepetit, T. Trzcinski, M. Calonder, M. Å-zuysal et P. Fua. *BRIEF : Computing a Local Binary Descriptor Very Fast*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 07, pages 1281–1298, jul 2012. (Cité en page 13.)
- [Sünderhauf & Protzel 2011] N. Sünderhauf et P. Protzel. *BRIEF-Gist - closing the loop by simple means*. In 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1234–1241, Sep. 2011. (Cité en page 14.)

- [Taube & Bassett 2003] Jeffrey S. Taube et Joshua P. Bassett. *Persistent Neural Activity in Head Direction Cells*. *Cerebral Cortex*, vol. 13, no. 11, pages 1162–1172, 2003. (Cité en page 46.)
- [Taube & Burton 1995] J. S. Taube et H. L. Burton. *Head direction cell activity monitored in a novel environment and during a cue conflict situation*. *Journal of Neurophysiology*, vol. 74, no. 5, page 1953–1971, Nov 1995. (Cité en pages x, 47 et 48.)
- [Taube *et al.* 1990] Jeffrey S. Taube, Robert U. Muller et James B. Ranck. *Head-direction cells recorded from the postsubiculum in freely moving rats. II. Effects of environmental manipulations*. *J. Neurosci. IO*, page 436447, 1990. (Cité en page 46.)
- [Thrun & Montemerlo 2006] Sebastian Thrun et Michael Montemerlo. *The Graph SLAM Algorithm with Applications to Large-Scale Mapping of Urban Structures*. *The International Journal of Robotics Research*, vol. 25, no. 5-6, pages 403–429, 2006. (Cité en page 53.)
- [Thrun 1998] Sebastian Thrun. *Learning Metric-Topological Maps for Indoor Mobile Robot Navigation*. *Artif. Intell.*, vol. 99, no. 1, page 21–71, Février 1998. (Cité en page 24.)
- [Tolman 1948] Edward C. Tolman. *Cognitive maps in rats and men*. *Psychological Review*, vol. 55, no. 4, page 189–208, 1948. (Cité en pages 26, 30 et 43.)
- [Tsotsos *et al.* 2018] John K. Tsotsos, Iuliia Kotseruba, Amir Rasouli et Markus D. Solbach. *Visual attention and its intimate links to spatial cognition*. *Cognitive Processing*, vol. 19, no. S1, page 121–130, Aug 2018. (Cité en page 61.)
- [Van Brummelen *et al.* 2018] Jessica Van Brummelen, Marie O’Brien, Dominique Gruyer et Homayoun Najjaran. *Autonomous vehicle perception : The technology of today and tomorrow*. *Transportation Research Part C : Emerging Technologies*, vol. 89, page 384–406, Apr 2018. (Cité en page 15.)
- [Waiker *et al.* 2019] Prashant Waiker, Saman Baral, Anissa Kennedy, Shilpi Bhatia, Anika Rueppell, Kevin Le, Esmaeil Amiri, Jennifer Tsuruda et Olav Rueppell. *Foraging and homing behavior of honey bees (*Apis mellifera*) during a total solar eclipse*. *The Science of Nature*, vol. 106, no. 1–2, Jan 2019. (Cité en page 17.)
- [Walcott 1996] Walcott. *Pigeon homing : observations, experiments and confusions*. *The Journal of experimental biology*, vol. 199 Pt 1, pages 21–7, 1996. (Cité en pages 17, 28 et 30.)
- [Wang & Zhang 2011] Hai Wang et Shuwu Zhang. *Evaluation of global descriptors for large scale image retrieval*, page 626–635. Springer Berlin Heidelberg, 2011. (Cité en page 14.)
- [Wang *et al.* 2019] Y. Wang, X. Lu et Z. Zuo. *Autonomous Vehicles Path Planning With Enhanced Ant Colony optimization*. In *2019 Chinese Control Conference (CCC)*, pages 6633–6638, 2019. (Cité en page 20.)

- [Wiener *et al.* 1995] Sidney I. Wiener, Victor A. Korshunov, René Garcia et Alain Berthoz. *Inertial, Substratal and Landmark Cue Control of Hippocampal CA1 Place Cell Activity*. European Journal of Neuroscience, vol. 7, no. 11, page 2206–2219, Nov 1995. (Cité en page 46.)
- [Wilson 1968] E. O. Wilson. *The Dance Language and Orientation of Bees*. Karl von Frisch. *Translated from the German edition (Berlin, 1965) by Leigh E. Chadwick*. Belknap Press (Harvard University Press), Cambridge, Mass., 1967. *xiv + 566 pp.* Science, vol. 159, no. 3817, page 864–865, Feb 1968. (Cité en page 17.)
- [Wiltschko & Wiltschko 2012] Roswitha Wiltschko et Wolfgang Wiltschko. Magnetoception, page 126–141. Springer US, 2012. (Cité en page 17.)
- [Wolf 1930] Ernst Wolf. *The Homing Behavior of Bees*. The Journal of Social Psychology, vol. 1, no. 2, pages 300–310, 1930. (Cité en page 17.)
- [Yazdanbakhsh & Livingstone 2006] Arash Yazdanbakhsh et Margaret S Livingstone. *End stopping in V1 is sensitive to contrast*. Nature Neuroscience, vol. 9, no. 5, page 697–702, Apr 2006. (Cité en page 41.)
- [Yu *et al.* 2018] Deyue Yu, Gordon E. Legge, Gunther Wagoner et Susana T.L. Chung. *Training peripheral vision to read : Boosting the speed of letter processing*. Vision Research, vol. 152, page 51–60, Nov 2018. (Cité en page 37.)
- [Zeng & Si 2017] Taiping Zeng et Bailu Si. *Cognitive Mapping Based on Conjunctive Representations of Space and Movement*. vol. 11, page 61, 11 2017. (Cité en pages 56 et 88.)
- [Zhang & Manahan-Vaughan 2013] Sijie Zhang et Denise Manahan-Vaughan. *Spatial Olfactory Learning Contributes to Place Field Formation in the Hippocampus*. Cerebral cortex (New York, N.Y. : 1991), vol. 25, 09 2013. (Cité en page 46.)
- [Zhang *et al.* 2015] Z. Zhang, Y. Xu, J. Yang, X. Li et D. Zhang. *A Survey of Sparse Representation : Algorithms and Applications*. IEEE Access, vol. 3, pages 490–530, 2015. (Cité en page 89.)