



**HAL**  
open science

# Extraction d'informations spatio-temporelles du mouvement avec un réseau de neurones à spike : application sur une tâche de prédiction de trajectoire de balles

Guillaume Debat

## ► To cite this version:

Guillaume Debat. Extraction d'informations spatio-temporelles du mouvement avec un réseau de neurones à spike : application sur une tâche de prédiction de trajectoire de balles. Neurosciences [q-bio.NC]. Université Paul Sabatier - Toulouse III, 2021. Français. NNT : 2021TOU30228 . tel-03684131

**HAL Id: tel-03684131**

**<https://theses.hal.science/tel-03684131>**

Submitted on 1 Jun 2022

**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.



# THÈSE

En vue de l'obtention du  
**DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE**  
Délivré par l'Université Toulouse 3 - Paul Sabatier

---

Présentée et soutenue par  
**Guillaume DEBAT**

Le 17 décembre 2021

**Extraction d'informations spatio-temporelles du mouvement avec  
un réseau de neurones à spike : application sur une tâche de  
prédiction de trajectoire de balles**

---

Ecole doctorale : **CLESCO - Comportement, Langage, Education, Socialisation,  
Cognition**

Spécialité : **Neurosciences**

Unité de recherche :

**CERCO - Centre de Recherche Cerveau et Cognition**

Thèse dirigée par

**Robin BAURÈS et Michel PAINDAVOINE**

Jury

**M. Laurent PERRINET**, Rapporteur

**M. Jean MARTINET**, Rapporteur

**Mme Julie GROLLIER**, Rapporteur

**M. Benoit MIRAMOND**, Examineur

**M. Robin BAURES**, Directeur de thèse

**M. Michel PAINDAVOINE**, Co-directeur de thèse

---

---

*« L'avenir n'est pas ce qui va arriver  
mais ce que nous allons faire. »  
Henri Bergson*



---

**Titre** : Extraction d'informations spatio-temporelles du mouvement avec un réseau de neurones à spike : application sur une tâche de prédiction de trajectoire de balles  
**Mots clés** : Réseau de neurones à spike, caméra événementielle, analyse de mouvement, prédiction

L'intelligence artificielle regroupe l'ensemble des techniques et théories permettant à un système artificiel d'agir de façon dite intelligente, c'est-à-dire comme un humain ou un animal le ferait. Ce domaine de recherche a fait récemment l'objet de grandes avancées qui ont permis de surpasser de nombreuses méthodes de traitement de l'information. Plus particulièrement pour la catégorie des réseaux de neurones qui ont fait preuve de performances impressionnantes et d'une forte capacité à s'adapter à différents cas d'usages. Ces réseaux de neurones cherchent à imiter le fonctionnement des neurones du cerveau humain et peuvent apprendre des données qui leur sont fournies pour répondre à différents problèmes. Cet attrait pour ces méthodes s'est notamment amplifié ces dernières années avec l'arrivée des réseaux de neurones artificiels profonds (DNNs) et notamment dans le domaine du traitement de l'information visuelle avec les réseaux de neurones convolutionnels (CNNs). Cependant, la quête de puissance des DNNs et CNNs les ont amenée à s'éloigner de leur inspiration biologique initiale. Ainsi, les réseaux de neurones artificiels (ANNs) ne ressemblent désormais plus au fonctionnement du cerveau humain, consomment beaucoup d'énergie et nécessitent de nombreuses données pour les entraîner.

Une réponse à ces différents problèmes a été proposée et celle-ci vient encore une fois du biomimétisme. En effet, celle-ci consiste à s'inspirer de la méthode de communication de l'information des neurones. Les réseaux de neurones à spike (SNNs) se sont alors développés. Considérés comme la troisième génération de réseaux de neurones (MAASS 1997), ils utilisent les spikes (potentiels d'action) pour transmettre l'information entre les neurones, qui sont des événements binaires et inscrits temporellement, contrairement aux ANNs qui utilisaient des valeurs analogiques et continues. Le développement des SNNs s'est accompagné de l'utilisation de la règle d'apprentissage Spike-Timing-Dependent-Plasticity (STDP), qui a pour particularité d'être bio-inspirée, d'utiliser l'information temporelle contenue dans les spikes et d'être non-supervisée. De nouveaux capteurs et cartes neuromorphiques ont ensuite émergé permettant de profiter pleinement des avantages présentés par les SNNs et par le traitement de l'information événementielle de façon plus générale.

De récentes études ont montré que les SNNs pouvaient extraire de l'information à partir d'information visuelle sur des images ou en utilisant des caméras événementielles. L'objectif de cette thèse est d'utiliser et d'adapter une nouvelle caméra événementielle développée par l'entreprise *Yumain* à Dijon pour effectuer de l'analyse et de la prédiction de trajectoires de balles dans un contexte sportif. En effet, la capacité de prédiction est primordiale dans le sport afin d'anticiper le comportement de l'adversaire ou la trajectoire d'un ballon. Les experts sportifs notamment, présentent de fortes capacités de prédiction dans leur sport de prédilection. Ainsi l'objectif est d'évaluer la capacité de notre solution à prédire la trajectoire d'un ballon et de comparer les performances avec des participants experts ou non en sport de balle, nécessitant une anticipation de la trajectoire de la balle.

Ce manuscrit de thèse reviendra donc dans un premier temps sur l'état de l'art actuel des traitements événementiels, dont notamment les SNNs et les caméras asynchrones événementielles. La caméra *NeuroSoc* sera ensuite présentée, ainsi que le SNN utilisé durant cette thèse. Puis nous montrerons que ce type de SNN permet d'extraire de l'information à partir de spikes générés par de l'information visuelle, que ce soit sur des simulations ou depuis des acquisitions réelles. Ainsi, nous montrerons que l'information extraite permet de prédire la trajectoire de la balle et que cette prédiction est plus précise que celles effectuées par des participants humains. Enfin, quelques limites de la caméra *NeuroSoc* utilisée durant cette thèse seront exposés, ainsi que des propositions pour améliorer cette dernière.



---

**Title :** Extraction of spatio-temporal motion information with a spiking neural network : application to a ball trajectory prediction task

**Keywords :** Spiking neural networks, event-based visual sensor, motion analysis, prediction

Artificial intelligence covers all the techniques and theories that enable an artificial system to act in a so-called intelligent way, i.e., as a human or an animal would. This field of research has recently made significant progress which have enabled it to surpass many information processing methods. Specifically, the category of artificial neural networks (ANN) has shown impressive capabilities and strong abilities to adapt to different use cases. These neural networks seek to mimic the processing of neurons in the human brain and can learn from provided data to solve various tasks. This attraction for these methods has notably increased in recent years with the arrival of deep neural networks (DNN), notably in visual information processing with convolutional neural networks (CNN). However, the methods currently used by DNNs and CNNs no longer mimic the human brain's processing, use a lot of energy, and require a lot of data to train them. An answer to these different limits has been proposed. This one comes from bio-mimicry, once again, as it consists of being inspired by the information communication method of neurons. Spiking neural networks (SNN) were then developed. Considered as the third generation of neural networks, they use spikes (action potentials) to transmit information between neurons, which are binary events with a temporal information, as opposed to ANNs that used analog and continuous values. The development of SNNs was accompanied by the spike timing dependant plasticity (STDP) learning rule. This one has the particularity of being bio-inspired, using the temporal information contained in the spikes and being unsupervised. New sensors and neuromorphic chips have emerged to take full advantage of the benefits offered by SNNs and event-driven information processing more generally.

Recent studies have shown that SNNs can extract information from visual stimuli on images or by using event-driven cameras. The objective of this thesis is to use and adapt a new event-driven camera developed by the company *Yumain*, located in Dijon, to perform analysis and prediction of ball trajectories in a sports context. Indeed, the ability of prediction is essential while practicing sport in order to anticipate the opponent's behavior or the trajectory of a ball. Sports experts, particularly, have strong predictive skills in their favorite sport. Thus the objective is to evaluate the capacity of our solution on the ball's trajectory prediction task and compare the performances with expert and/or non-expert participants.

First of all, this thesis manuscript will review the current state of the art of event-driven processing, including event-driven cameras and asynchronous cameras. Then, the NeuroSoc camera will be presented, along with the SNN used during this thesis. Afterward, we will show that this type of camera allows extracting information from spikes generated by visual information, either on simulations or from real acquisitions. Thus, we will show that the extracted information allows predicting the ball's trajectory and that this prediction is more accurate than those made by human participants. Finally, some limitations of the NeuroSoc camera used during this thesis will be presented, along with some proposals to improve it.





---

# Remerciements

Les résultats et analyses présentés dans ce manuscrit de thèse constituent les travaux réalisés durant ces trois années de doctorat. Ceux-ci sont le résultat de travaux et de collaborations entre différentes personnes à travers des implications directes dans la réflexion des études et l'obtention des résultats mais aussi des implications indirectes à travers des échanges, des propositions d'idées, des conseils, etc. Ainsi, avant de débiter ce manuscrit, je tiens à remercier autant que possible toute personne ayant participé aux différents travaux réalisés durant cette thèse.

Pour commencer, j'aimerais remercier mes deux co-directeurs de thèse, Robin Baurès et Michel Paindavoine. Le projet dans lequel s'inscrit cette thèse est transdisciplinaire mêlant les sciences humaines (perception du mouvement, prédiction, etc), les neurosciences et les sciences de l'ingénieur (développement logiciel, intelligence artificielle, développement embarqué, etc). Robin et Michel à travers leurs profils différents ont permis la réalisation de ce projet tout en prenant en compte les différentes contraintes de celui-ci. J'aimerais aussi remercier différents membres du laboratoire CerCo qui ont activement participé aux travaux et partagé leurs connaissances. Ainsi, je tiens dans un premier temps à particulièrement remercier Tushar Chauhan, post-doctorant au laboratoire, qui m'a très largement accompagné durant cette thèse. Je tiens aussi à remercier Benoît Cottureau et Timothée Masquelier, chargés de recherche au laboratoire, qui m'ont partagé leurs expertises durant ces trois années. De façon plus générale, je remercie les différents membres du laboratoire CerCo, notamment les membres de l'équipe SV3M ainsi que Simon Thorpe dont les recherches ont très largement impacté les travaux réalisés au cours de cette thèse.

Durant cette thèse j'ai eu l'occasion d'utiliser une nouvelle caméra événementielle appelée la caméra *NeuroSoc*. Cette caméra a été développée par l'entreprise *Yumain* à Dijon dont Michel Paindavoine est le directeur technique et l'un des co-fondateurs. Ainsi, je tiens à remercier tous les membres de *Yumain* et plus particulièrement Clément Joly avec qui j'ai très largement collaboré durant les premiers mois de la thèse pour prendre en main et terminer le développement de cette caméra.

Avant de conclure cette partie, je voudrais remercier les différents membres du jury qui ont fait preuve d'une réelle implication et d'un professionnalisme sans égal dans la relecture de cette thèse et dans les rapports, questions et remarques apportées à ces travaux. Ainsi, je remercie Laurent Perrinet et Jean Martinet en qualité de rapporteurs ainsi que Benoît Miramond et Julier Grollier en qualité d'examineurs.

Pour conclure, je tiens à remercier toutes les personnes qui n'étaient pas directement impliquées dans ces études, mais qui m'ont accompagnées durant ces trois années. Je remercie ainsi toute ma famille, plus particulièrement mes parents Pascal et Ghislaine et mon frère Johan de m'avoir soutenus durant ces trois années (et de façon plus générale depuis ma naissance), mes amis (le QG et les Fuchs) qui étaient là pour me rappeler qu'il y a plus important que les travaux de thèse et qui ont toujours témoigné de l'intérêt pour mes sujets de recherche. Je tiens particulièrement à remercier ma compagne Clara Giraudat, qui m'a soutenu durant ces trois années tant moralement que professionnellement, en relisant maintes et maintes fois mes présentations et mes différents rapports. Je remercie aussi mes collègues de bureau, Mario Hervault, Pauline Audurier et Paul Fricker qui ont toujours été de bons conseils tant sur le côté scientifique que sur des sujets plus généraux relatifs à la thèse.

Et pour terminer, je tiens à remercier tout ceux que j'ai pu oublier dans cette page et qui ont de près ou de loin participé à cette étape de ma vie.



# Table des matières

<b>1</b>	<b>Introduction</b>	<b>21</b>
1.1	Motivations et objectifs . . . . .	23
1.2	Architecture de ce manuscrit de thèse . . . . .	24
<b>2</b>	<b>Etat de l'art</b>	<b>27</b>
2.1	Traitement visuel . . . . .	27
2.1.1	Neurones . . . . .	27
2.1.2	Rétine . . . . .	28
2.1.3	Cortex visuel . . . . .	29
2.2	Traitement visuel artificiel . . . . .	31
2.2.1	Capteurs visuels . . . . .	31
2.2.2	ANN/SNN . . . . .	34
2.2.3	Modèles de neurones . . . . .	35
2.2.3.1	Hodgkin-Huxley . . . . .	35
2.2.3.2	Izhikevich . . . . .	35
2.2.3.3	Neurone intégrateur à fuite (LIF) . . . . .	36
2.2.3.4	Spike Response Model (SRM) . . . . .	36
2.2.3.5	Neurone stochastique . . . . .	37
2.2.4	Connections . . . . .	37
2.2.4.1	<i>Feed-forward</i> (FF) . . . . .	37
2.2.4.2	Récurrent . . . . .	37
2.2.5	Architectures . . . . .	38
2.2.5.1	Rétinotopique . . . . .	38
2.2.5.2	Convolutionnel . . . . .	39
2.2.5.3	Dense . . . . .	39
2.2.5.4	Reservoir computing . . . . .	39
2.2.6	Stratégie de codage . . . . .	39
2.2.6.1	Codage fréquentiel . . . . .	40
2.2.6.2	Codage par latence . . . . .	40
2.2.6.3	Codage par rang . . . . .	41
2.2.6.4	Codage parcimonieux . . . . .	41
2.3	Apprentissage . . . . .	42
2.4	Extraction d'informations spatio-temporelles . . . . .	45
2.4.1	Forme . . . . .	45
2.4.2	Profondeur . . . . .	46
2.4.3	Mouvement . . . . .	48
2.4.3.1	Délais . . . . .	49
2.5	Neuromorphic chip . . . . .	53
2.6	Prédictions dans le domaine du sport . . . . .	54
2.7	Prédictions visuelles par des SNN . . . . .	56

---

2.8	Problématique	57
<b>3</b>	<b>Méthodologie</b>	<b>59</b>
3.1	Méthodes	59
3.1.1	NeuroSoc	59
3.1.2	SNN	62
3.1.2.1	Architecture/topologie	62
3.1.2.2	Modèle du neurone	63
3.1.2.3	Adaptation de seuil	64
3.1.2.4	Délais	66
3.1.2.5	STDP	66
3.1.2.6	Sélection des paramètres	67
3.2	Travaux préliminaires	67
3.2.1	Catégorisation de direction de voitures	68
3.2.2	Simulations	69
3.2.2.1	Simulation 2D	69
3.2.2.2	Simulations 3D	71
3.2.2.3	Analyse de trajectoires et catégorisation du point de chute de la balle	72
3.2.2.4	Prédiction du point de chute de la balle	74
3.2.3	Conclusion	77
<b>4</b>	<b>Etude 1 : Prédiction d'échanges de passes</b>	<b>79</b>
4.1	Objectifs de l'étude	79
4.2	Méthodes	79
4.2.1	SNN	80
4.2.2	Acquisition de données	80
4.2.3	Vicon	80
4.2.4	Prédictions	81
4.2.5	Entraînement	82
4.2.6	Performances chez l'humain	82
4.3	Résultats	84
4.3.1	Sélectivités	84
4.3.1.1	Direction	84
4.3.1.2	Vitesse	85
4.3.2	Motion Tracking	86
4.3.3	Prédictions	88
4.3.4	Performances chez l'humain	90
4.3.5	Comparaisons	91
4.4	Discussion	93
<b>5</b>	<b>Etude 2 : Basket</b>	<b>95</b>
5.1	Motivations et objectifs	95
5.2	Acquisition de données	95
5.3	Méthodes	96
5.3.1	Architecture	96
5.3.2	Apprentissage et prédictions	97
5.3.3	Performances chez l'humain	98
5.4	Résultats	99
5.4.1	Prédictions (SNN)	99
5.4.2	Impact du <i>shooter</i>	101

---

5.4.3	Prédictions (Humains) . . . . .	103
5.4.4	Comparaison des résultats . . . . .	105
5.5	Discussion . . . . .	107
<b>6</b>	<b>Optimisation de la caméra NeuroSoC</b>	<b>109</b>
6.1	Événements entourés de polarités opposées . . . . .	109
6.2	Détection de mouvements lents . . . . .	112
6.3	Résultats . . . . .	113
6.3.1	Impact sur la génération d'événements . . . . .	113
6.3.2	Précision temporelle de la caméra NeuroSoc . . . . .	116
6.4	Discussion . . . . .	118
<b>7</b>	<b>Discussion et perspectives</b>	<b>121</b>
7.1	Discussion générale des résultats . . . . .	121
7.2	Perspectives . . . . .	124
7.3	Maturité des SNNs . . . . .	126
<b>8</b>	<b>Références</b>	<b>131</b>
<b>9</b>	<b>Annexe</b>	<b>157</b>
9.1	DMA . . . . .	157



# Table des figures

2.1	Panneau A : Schéma de deux neurones (REZAEI-MAZINANI 2017). Panneau B : Évolution du potentiel de membrane d'un neurone <sup>1</sup> . . . . .	28
2.2	Connectivité de la rétine . . . . .	29
2.3	Panneau A : modèle proposé permettant d'expliquer la formation d'un champ récepteur de type Gabor à partir des champs récepteurs des neurones du Corps Géniculé Latéral (LGN) (image provenant de (HUBEL et WIESEL 1962). Panneau B : Hiérarchie des connexions entre les différentes aires visuelles (FELLEMAN et VAN ESSEN 1991). Panneau C : Simplification des différentes hiérarchies entre les aires (images provenant de (PASTERNAK, BISLEY et CALKINS 2003). . . . .	30
2.4	Photographie animée de James Muybridge . . . . .	31
2.5	Différences entre des caméras standard ( <i>frame-based</i> , graphique du haut) et asynchrones ( <i>event-based</i> , graphique du bas) (H. KIM, LEUTENEGGER et DAVISON 2016) . . . . .	32
2.6	A : Circuit d'une <i>Dynamic Vision Sensor</i> (DVS) et leurs comparaisons avec les cellules de la rétine. B : Génération de spikes en fonction de la variation de luminosité. C : représentation visuelle des spikes générés. Figure provenant de (Christoph POSCH, Teresa SERRANO-GOTARREDONA et al. 2014) . . . . .	33
2.7	Différents types de connexions . . . . .	38
2.8	Codage temporel, figure provenant de (S. THORPE, DELORME et VAN RULLEN 2001) . . . . .	41
2.9	Différentes méthodes de codage . . . . .	41
2.10	Codage parcimonieux, figure provenant de (Bruno A OLSHAUSEN et David J FIELD 2004) . . . . .	42
2.11	Règle d'apprentissage STDP observée par (BI et POO 2001) . . . . .	43
2.12	Règle d'apprentissage STDP inspirée par (BICHLER et al. 2012) . . . . .	44
2.13	Figure provenant de (CHAUHAN et al. 2018) . . . . .	47
2.14	Panel A : Délais fixes de (G. ORCHARD et al. 2013). Panel B : Multi-synapses de (PAREDES-VALLES, SCHEPER et CROON 2019) . . . . .	49
2.15	Différentes méthodes de délais. . . . .	50
2.16	Impacts des délais proportionnels à la distance entre deux neurones. . . . .	51
2.17	Impact des différentes méthodes d'inhibitions avec délais . . . . .	52
2.18	Comparaison du nombre d'opérations nécessaire pour atteindre une performance de 98% de précision sur MNIST entre des SNNs et des ANNs. . . . .	53
2.19	Différences de trajectoires selon que le panier est réussi (jaune) ou raté (bleu) . . . . .	56
3.1	Architecture de la NeuroSoc . . . . .	60
3.2	Panneau A : Kernel utilisé pour la différence de gaussiennes ( <i>Difference of Gaussians</i> ) (DoG). Panneau B : Valeurs complémentées . . . . .	61
3.3	Filtrage spatio-temporel effectué par la caméra . . . . .	62
3.4	Architecture du SNN utilisé . . . . .	63



3.5	Méthode d'adaptation de seuil utilisée. . . . .	65
3.6	Exemple des spikes générés lors du passage d'une voiture. . . . .	68
3.7	Nombre de spikes générés pour des voitures allant de la gauche vers la droite (jaune) ou de la droite vers la gauche. Certaines voitures n'étaient pas labellisées car de nombreuses voitures se croisaient au même moment sur l'image (couleur bleu). . . . .	69
3.8	Exemple des spikes générés pour différentes trajectoires simulées. . . . .	70
3.9	Matrice de poids après entraînement des 64 filtres. La couleur rouge identifie les poids des synapses sans délais, le vert celles avec une délais 2/240 s, le bleu les délais de 3/240 s. Chaque filtre est caractérisé par deux matrices, celle de gauche représente les poids ON et celle de droite les poids OFF. . . . .	70
3.10	Panneau A : Histogramme polaires des spikes émis par direction pour chaque filtre. Panneau B : Histogramme des spikes émis par vitesse pour chaque filtre. . . . .	71
3.11	Panneau A : Exemple de la simulation Unity utilisée. Panneau B : Spikes générés par les trajectoires simulées. Panneau C : Architecture de la simulation et emplacement des deux points de vues. . . . .	72
3.12	Panneau A, Dimensions X et Y des vecteurs initiaux permettant de générer les trajectoires de balles. Les deux couleurs représentent les deux classes possibles. Panneau B, C et D : Les points bleus représentent tous les couples (X, Y) générés. Les cercles rouges, ceux pour qui un neurone décharge. Le panneau B, représente les couples (X,Y) pour lesquels 16 différents neurones entraînés avec la STDP déchargent. Le panneau C, représente pour 16 neurones entraînés avec la Reward-STDP (RSTDP) et de la première classe (zone bleue). Le panneau D, représente pour 16 neurones entraînés avec la RSTDP et de la seconde classe (zone rouge). . . . .	73
3.13	Nombre de spikes émis pour les 64 neurones de la dernière couche en fonction de la classe de trajectoire. . . . .	74
3.14	Positions des points de chute pour lesquels les neurones ont déchargé (points bleus). Le polygone jaune représente la zone de sélectivité de ces neurones. Panneau B : prédiction effectuée par les neurones (croix rouge) en fonction du point réel de chute de la balle (cercle bleu), pour les deux angles de vues différents (45° et orthogonal). Les traces blanches représentent la somme des zones de sélectivités de chaque neurone qui a déchargé au cours de cette trajectoire. . . . .	75
3.15	Prédiction effectuée par les neurones (croix rouge) en fonction du point réel de chute de la balle (cercle bleu), pour les deux angles de vues différents (45° et orthogonal). Les traces blanches représentent la somme des zones de sélectivités de chaque neurone qui a déchargé au cours de cette trajectoire. . . . .	76
3.16	Évolution de l'erreur de prédiction en fonction du temps de présentation de la trajectoire de la balle et de l'angle de vue. . . . .	76
4.1	Architecture utilisée . . . . .	79
4.2	Positions spatiales de quelques trajectoires du ballon générées par le Vicon . . . . .	81
4.3	Panneau A : Processus de décodage. Panneau B : Positions horizontales utilisées comme points de réception. Panneau C : Exemple de mise à jour du score . . . . .	83
4.4	Sélectivité à la direction. Panneau A : Sélectivité normalisée $\theta_{f_n}$ de quatre filtres différents. Panneau B : Sélectivité moyenne de tous les filtres centrée sur leur direction favorite $\theta_{c_n}$ . Panneau C : Distribution des directions favorite $\theta_{c_n}$ et du nombre d'événements générés par la caméra NeuroSoc par direction. . . . .	85
4.5	Processus de sélection de vitesses aléatoires basé sur la sélectivité de direction du filtre . . . . .	86

4.6	Sélectivité à la vitesse. . . . .	87
4.7	Panneau A : Histogramme de la distance moyenne $D_n$ de tous les filtres. Panneau B : Position des spikes pour quatre filtres différents . . . . .	88
4.8	Panneau A : Évolution de l'erreur absolue moyenne en fonction du temps de présentation pour des régressions polynomiales (RPs) appliqués à différentes entrées. Panneau B : Erreur absolue moyenne pour des RPs appliqués à différentes entrées. . . . .	89
4.9	Panneau A : Évolution de l'erreur moyenne absolue en fonction du temps de présentation de la trajectoire pour un nombre différents de trajectoires présentées durant l'apprentissage. Panneau B : Erreur moyenne absolue pour un nombre différents de trajectoires présentées durant la phase d'apprentissage. . . . .	90
4.10	Panneau A : Distribution de l'erreur absolue moyenne en fonction de la phase d'apprentissage. Panneau B : Evolution de l'erreur moyenne absolue en fonction du temps de présentation de la trajectoire. . . . .	91
4.11	Évolution de la moyenne (panneau A) et de l'écart type (panneau B) de l'erreur absolue en fonction du temps de présentation pour les participants humains (rouge) et les SNNs (bleu). . . . .	92
4.12	Évolution de la moyenne (panneau A) et de l'écart type (panneau B) de la valeur prédite en fonction du temps de présentation pour les participants humains (rouge) et les SNNs (bleu). . . . .	93
5.1	Variations de luminosités . . . . .	96
5.2	Architecture de la solution utilisée pour la prédiction de lancer franc de basket . . . . .	97
5.3	Trajectoire de lancer franc à différents instants . . . . .	99
5.4	Panneau A : événements générés par la caméra pour les trajectoires réussies (rouge) et ratées (vert). Panneau B : Somme des poids synaptiques des neurones de la classe "réussie" (rouge) et de la classe "ratée" (vert) . . . . .	100
5.5	Évolution de la prédiction effectuée par le SNN selon le temps de présentation. Panneau A : Précision de la prédiction. Panneau B : Valeur prédite, plus la valeur est proche de zéro, plus le SNN a prédit que le lancé est raté. . . . .	100
5.6	Matrice de confusion de prédiction des <i>shooters</i> , à différents temps de présentation	101
5.7	Matrice de confusion de prédiction des <i>shooters</i> , sans le mouvement de lancer du <i>shooter</i> . . . . .	103
5.8	Panneau A : Boxplot de l'erreur moyenne selon que le sujet est expert ou non. Panneau B : Évolution de l'erreur moyenne en fonction du temps de présentation de la trajectoire. . . . .	104
5.9	Panneau A : Évolution de la tendance de prédiction en fonction du temps de présentation de la trajectoire. Panneau B : Comparaison entre les résultats pendant et après apprentissage de l'évolution de la tendance de prédiction en fonction du temps de présentation de la trajectoire. . . . .	105
5.10	Panneau A : Précision moyenne des différents groupes testés (SNN ;Expert ;Non-experts). Panneau B : Évolution de la précision de la prédiction indépendamment du groupe testé. . . . .	105
5.11	Panneau A : Prédiction moyenne des différents groupes testés (SNN, Expert, Non-experts). Panneau B : Évolution de la prédiction indépendamment du groupe testé. Panneau C : Évolution de la prédiction au cours du temps en fonction du groupe testé. . . . .	106
5.12	Comparaison des trajectoires entre l'étude de (AGLIOTI et al. 2008) (gauche) et nos trajectoires. Vert = lancer raté; Rouge = lancer réussi . . . . .	108
6.1	Filtrage par convolution d'un DoG . . . . .	110
6.2	Sortie du filtrage spatio-temporel lors d'un mouvement rapide . . . . .	111

---

6.3	Traitement spatio-temporel proposé . . . . .	111
6.4	Sortie du filtrage spatio-temporel lors d'un mouvement lent. L'intensité des valeurs retournée par le filtrage est proportionnelle à la vitesse du stimulus. . . . .	112
6.5	Traitement spatio-temporel proposé avec trace du mouvement . . . . .	113
6.6	Exemple des différents processus sur un stimulus de barre en rotation . . . . .	114
6.7	Exemple des différents processus appliqués à la vidéo extraite par la caméra NeuroSoc d'une balle se déplaçant rapidement . . . . .	115
6.8	Exemple des différents processus appliqués à la vidéo extraite par la caméra NeuroSoc d'une balle se déplaçant lentement . . . . .	115
6.9	Précision temporelle des événements générés en fonction de la valeur de seuil et de la méthode d'extraction utilisée. . . . .	117
6.10	Impact des différentes méthodes sur la précision temporelle des événements générés. . . . .	118
7.1	Panneau A, nombre de publications autour du thème "systèmes événementiels visuels, olfactifs ou auditifs", figure provenant de (TAYARANI-NAJARAN et SCHMUKER 2021). Panneaux B et C, nombre de publications contenant les mots "spiking neural network", "artificial neural network", "convolutional neural network" ou "recurrent neural network", données récoltées avec " <a href="https://app.dimensions.ai/discover/public">https://app.dimensions.ai/discover/public</a> " et " <a href="https://books.google.com/ngrams">https://books.google.com/ngrams</a> " . . . . .	127
7.2	Panneau A, évolution de l'intérêt pour les modèles neuromorphiques au cours du temps. Panneaux B, proportion des méthodes d'apprentissage utilisée pour des modèles embarqués. Figures provenant de (SCHUMAN et al. 2017). . . . .	128
9.1	Architecture du Zynq7020 . . . . .	158

# Acronymes

- AE** erreur absolue moyenne. 88, 90, 91
- AG** algorithme génétique. 67
- ANN** réseau de neurones artificiels. 5, 15, 22, 23, 31, 34, 37, 39, 40, 44–46, 48, 53, 54, 56, 57, 67, 68, 126–130
- AON** réseau d'action-observation (*action-observation network*). 55, 58
- ATIS** *Asynchronous Time Based Image Sensor*. 33
- CNN** réseau de neurones convolutionnels. 5, 34, 45, 48, 57, 58, 62, 126, 129
- CREPS** Centre de Ressources d'Expertise et de Performance Sportive. 95, 98
- CSNN** réseau de neurones convolutionnels à spike. 39, 46, 62, 80
- DAVIS** *Dynamic and Active Pixel Vision Sensor*. 33
- DMA** accès mémoire direct (*Direct Memory Access*). 157
- DNN** réseau de neurones artificiels profond. 5, 22, 53
- DoG** différence de gaussiennes (*Difference of Gaussians*). 15, 17, 42, 60, 61, 69, 110, 111, 123
- DVS** *Dynamic Vision Sensor*. 15, 32, 33, 46, 48, 58, 116, 122–124, 128, 129
- FB** *feed-back*. 43, 58, 125
- FF** *feed-forward*. 11, 37, 43, 44, 48, 50, 51, 58, 62, 64, 66, 125
- FIFO** *First In First Out*. 157
- FPGA** circuit logique programmable (*Field Programmable Gate Arrays*). 23, 24, 54, 124, 157
- FPS** images par seconde (*Frames Per Second*). 95, 110, 116, 118, 119, 122–124
- GPU** processeur graphique (*Graphics Processing Unit*). 53
- LGN** Corps Géniculé Latéral. 15, 29, 30, 34, 59, 122
- LIF** neurone intégrateur à fuite. 11, 36, 47, 112
- LSM** *liquide state machine*. 39, 57, 94, 126
- LSTM** Long Short Term Memory. 129
- LTD** dépressions à long terme. 42, 43, 98
- LTP** potentialisation à long terme. 42, 43, 67, 98, 130
- PL** circuit logique programmable *Programmable Logic*. 157
- PS** processeur (*Processing System*). 157

---

**ReSuMe** *Remote Supervised Learning*. 44

**RF** champs recepneur (*Receptive Field*). 62

**RMSE** erreur quadratique moyenne. 82

**RNN** *Recurrent Neural Network*. 56, 57, 129

**RP** régression polynomiale. 17, 81–83, 88–90, 96, 121, 122

**RSTDP** Reward-STDP. 16, 44, 45, 72–74, 96, 97, 99, 101, 121

**SD AE** écart type de l'AE. 88, 90, 91

**SNN** réseau de neurones à spike. 5, 11, 13, 15, 17, 22–25, 34, 35, 37, 39, 40, 42, 44–49, 53, 54, 56–59, 62, 63, 65–72, 74, 77, 79–82, 84, 86–103, 105–107, 121, 122, 124–130

**SPAN** *spike pattern association neuron*. 44

**SRM** Spike Response Model. 11, 36

**STDP** Spike-Timing-Dependent-Plasticity. 5, 12, 15, 16, 20, 42–47, 50, 51, 57, 58, 64, 66–68, 70, 72–74, 77, 80, 82, 87, 97–99, 101, 121, 125, 127, 128, 130

**TMS** Stimulation magnétique transcrânienne. 54

**VLSI** intégration à très grande échelle. 53

**WS** *weight sharing* (partage de poids). 39

**WTA** winner-take-all. 37

# Chapitre 1

## Introduction

Une des caractéristiques principales des êtres vivants est leur capacité à s'adapter afin de survivre. Chaque être vivant utilise ses propres moyens afin de faire perdurer son espèce, en fonction notamment de ses capacités sensori-motrices. Pour survivre les êtres vivants perçoivent leur environnement à travers leurs capteurs sensoriels (NINIO 1989). Ces informations représentées par différents types de signaux (électriques, chimiques, etc) sont ensuite traitées et leur permettent d'agir sur leur environnement. Chez les humains et les animaux on parle alors de boucle sensori-motrice, dans laquelle des capteurs sensoriels permettent de percevoir l'environnement. Les signaux perçus sont ensuite traités et transmis à différentes parties du corps par les neurones et le système nerveux, afin de pouvoir ensuite agir sur notre environnement perçu en utilisant notre corps, nos muscles et nos capteurs sensoriels (O'REGAN et NOË 2001 ; DI PAOLO et al. 2014). Cependant ce type de mécanisme ne s'applique pas qu'aux êtres vivants possédant un cerveau, des neurones et pouvant se déplacer, mais se généralise à tous les êtres vivants. En effet, les plantes par exemple intègrent différents signaux extérieurs (stress salique, luminosité, qualité du sol, etc) afin d'agir ensuite sur leur environnement (croissance de la racine ou d'une tige, floraison, etc). Ainsi, les capacités sensorielles et motrices propres à chaque individu lui permettent de percevoir le monde qui l'entoure afin d'agir sur celui-ci pour garantir sa survie ou effectuer d'autres tâches plus communes. Par exemple, un aigle, grâce à sa vision, va pouvoir distinguer une proie depuis une très grande distance et voler jusqu'à elle afin de l'attaquer. Certaines plantes vont suivre la position du soleil afin de profiter au maximum de son énergie. Enfin, l'humain va par exemple profiter de l'habileté de sa main et de ses capacités cognitives pour écrire sur le clavier d'un ordinateur afin de rédiger une thèse.

Afin d'interagir avec leur environnement et garantir leur survie, les espèces ont constamment besoin de prédire (POLI 2010 ; BUTZ, SIGAUD et GÉRARD 2003). En effet, agir en fonction de l'état actuel des choses n'est pas efficace et le traitement de l'information perçue et l'initiation d'une action prend du temps, il faut donc anticiper pour prendre en compte ces latences (NIJHAWAN 1994 ; SENOT et al. 2005 ; HOGENDOORN et BURKITT 2019). Chez l'homme par exemple, le délai visuo-moteur est de l'ordre de 150-200ms (NIJHAWAN 2008). Le sport, bien qu'il n'a pas d'enjeu de survie, est un bon exemple pour mettre en avant les capacités d'intégration et d'analyse du monde externe à travers les différents sens, afin de pouvoir ensuite anticiper et réagir (LOFFING et CAÑAL-BRULAND 2017). En effet, quel que soit le sport l'anticipation est constamment nécessaire, le joueur de football doit anticiper le déplacement du ballon afin de pouvoir l'intercepter, un quarterback doit anticiper la vitesse et la trajectoire de son receveur et des défenseurs afin de pouvoir lancer le ballon sans qu'il soit intercepté. Le joueur de tennis ne peut pas se permettre d'attendre le dernier moment pour se déplacer afin de taper dans la balle, il doit l'anticiper avant même que le joueur adverse l'ait renvoyée. Cette anticipation s'apprend. Dans une premier temps grâce à notre expérience de tous les jours, à travers notre apprentissage

---

naturel. Puis dans un second temps, cette anticipation nécessite aussi un apprentissage spécialisé afin de s'améliorer dans des tâches précises comme dans le sport. Afin de jouer à un meilleur niveau, il est donc nécessaire d'apprendre à mieux prédire les différentes situations sportives. Cet apprentissage se fait en pratiquant ce même sport. Ainsi, les joueurs de sports experts présentent des capacités de prédictions supérieures à des personnes novices lorsqu'ils sont testés sur des tâches proches de leur sport d'expertise (LOFFING et CAÑAL-BRULAND 2017 ; BALSER et al. 2014).

La capacité de traitement de l'information faite par les humains et les animaux, et plus particulièrement par leur cerveau, a depuis longtemps intéressé les humains. En effet, celui-ci semble pouvoir traiter des problèmes complexes, divers et variés, être résistant au bruit, rapide, apprendre de son expérience, etc, et tout cela en ne consommant approximativement que 30 Watts (MAASS 2015). De nombreuses personnes se sont alors intéressées à son fonctionnement afin de le reproduire dans le but de créer des modèles permettant de surpasser l'état de l'art du traitement de l'information. Les premiers ANNs sont alors nés (ROSENBLATT 1958). Ceux-ci ont par la suite évolué au court du temps avec de nouveaux modèles (CNN, LSTM, etc). L'arrivée des DNNs et de nouvelles méthodes d'apprentissages ont permis de poursuivre l'amélioration de leurs performances (RAWAT et Z. WANG 2017). Ces avancées ont ainsi permis aux ANNs de dépasser les performances d'humains dans des tâches complexes comme le jeu de GO (J. X. CHEN 2016), et ont ouvert de grandes opportunités dans de nombreuses activités industrielles comme le traitement d'image (RAWAT et Z. WANG 2017). Cependant quelques limites persistent. En effet, contrairement au cerveau ces modèles consomment beaucoup d'énergie et sont très spécialisés pour des tâches spécifiques, celles pour lesquelles on les a entraînés. Les méthodes d'apprentissage utilisées sont pour le plus souvent supervisées ou semi-supervisées et utilisent des méthodes d'apprentissage globales dans laquelle l'erreur globale est utilisée pour définir des paramètres locaux (RAWAT et Z. WANG 2017 ; NØKLAND et EIDNES 2019). Ces méthodes nécessitent alors de grandes bases de données labellisées, ou pire, de passer un temps considérable à générer des bases de données et à les labelliser afin d'entraîner le réseau de neurones. Alors que de son côté le cerveau utilise des règles d'apprentissage locales non-supervisées (BI et POO 2001), une nouvelle alternative aux ANNs a été mise en avant, les SNNs<sup>1</sup>. Ceux-ci ressemblent aux ANNs dans leurs architectures : des neurones connectés entre eux (bien souvent organisés sous forme de couches) et qui se transmettent l'information hiérarchiquement ; l'impact qu'a un neurone sur un autre dépend du poids synaptique entre ces deux neurones, etc. Mais ils diffèrent par leurs moyens de transmettre l'information. Les SNNs utilisent des spikes (potentiels d'actions) comme signal transmis entre les neurones. Ces signaux sont binaires, asynchrones, événementiels, parcimonieux et temporellement définis. Ils sont à l'opposé des signaux utilisés par les ANNs qui sont mis-à-jour de façon périodique et ont des valeurs analogiques (sensées représenter la fréquence de décharge d'un neurone). Les SNNs permettent donc de représenter de façon plus biologiquement plausible le transfert d'information au sein d'un réseau de neurones, tout en s'inspirant des mêmes signaux utilisés par les neurones biologiques. Ces SNNs considérés comme la troisième génération de réseaux de neurones (MAASS 1997), bien que moins développés que l'état de l'art actuel des ANNs, possèdent de grandes capacités de computation et commencent déjà à concurrencer les performances de certains ANNs. De plus, la temporalité des spikes, l'évolution temporelle du potentiel de membrane des neurones ainsi que des mécanismes comme les délais synaptiques en font des candidats naturels pour traiter des stimuli temporels.

---

1. Durant toute cette thèse, une distinction sera faite entre ANNs (réseaux de neurones communs utilisant des valeurs analogiques) et SNNs (réseaux de neurones à spikes). Idem pour les DNNs qui seront considérés comme des ANNs profond.

---

De nombreuses études ont maintenant mis en avant les capacités des SNNs sur des tâches visuelles, comme de la reconnaissance d'images (DIEHL, ZARRELLA et al. 2016 ; MASQUELIER et S. J. THORPE 2007 ; S.-R. KHERADPISHEH et al. 2018), du tracking (BICHLER et al. 2012), du calcul de flux optique (PAREDES-VALLES, SCHEPER et CROON 2019), etc. ainsi que sur des tâches auditives (TAVANAEI et A. S. MAIDA 2017 ; J. WU, CHUA et Haizhou LI 2018) ou d'analyse de stimuli temporels (MASQUELIER, GUYONNEAU et S. J. THORPE 2008). Le développement des SNNs est grandement impacté par les connaissances déjà acquises avec les ANNs. Ainsi, des architectures similaires sont utilisées, les règles d'apprentissages sont adaptées, les jeux de données sont convertis en format événementiel ou des ANNs déjà entraînés sont transformés en SNNs. Cependant pour encore beaucoup d'études, les SNNs sont appliqués sur des stimuli contenant seulement des informations spatiales (des images) dans le but de pouvoir comparer les performances des SNNs avec ceux des ANNs. Avec le développement de nouvelles caméras événementielles, de nouvelles opportunités s'ouvrent aux SNNs car elles permettent de générer des stimuli spatio-temporels, qui sont des flux d'événements<sup>2</sup> parfaitement adaptés aux SNNs. Le développement des SNNs ainsi que de ces nouveaux capteurs asynchrones (LICHTSTEINER, C. POSCH et T. DELBRUCK 2008 ; Christoph POSCH, MATOLIN et WOHLGENANT 2011) s'accompagnent aussi d'un développement de nouvelles plateformes permettant de simuler ces SNNs (GOODMAN et BRETTE 2009 ; Milad MOZAFARI, GANJTABESH, NOWZARI-DALINI et MASQUELIER 2019 ; MO, Xinao CHEN et G. WANG 2021) et de cartes neuromorphiques permettant de traiter la sortie de ces capteurs en temps réel, tout en ne consommant que peu d'énergie. Différentes architectures sont disponibles (Loihi, TrueNorth, FPGA, etc) (MEROLLA et al. 2014 ; AKOPYAN et al. 2015 ; DAVIES, SRINIVASA et al. 2018), elles se caractérisent par un traitement asynchrone et peu consommateur d'énergie, car elles tirent partie des caractéristiques asynchrones et de la parcimonie des spikes générés. Ainsi, bien que les SNNs sont loin d'être aussi populaires et utilisés que les ANNs, il semblerait qu'entre les connaissances déjà acquises avec les ANNs, le développement des capteurs asynchrones, des cartes neuromorphiques et de tous les avantages que leur combinaison permet, ce type de système devrait pouvoir se développer rapidement et apporter des solutions aux différentes problématiques dont font face les ANNs.

## 1.1 Motivations et objectifs

Cette thèse s'inscrit dans une collaboration entre l'entreprise Yumain<sup>3</sup> et le laboratoire Centre de Recherche Cerveau et Cognition (CerCo). Les caméras événementielles déjà présentes sur le marché sont des caméras entièrement asynchrones, qui utilisent des informations purement temporelles pour générer des événements. Yumain a récemment développé une nouvelle caméra hybride composée d'une caméra synchrone (*frame-based*) et d'un circuit logique programmable (*Field Programmable Gate Arrays*) (FPGA) permettant d'effectuer un filtrage spatio-temporel et de générer des événements. Ceci permet ainsi de mimer le fonctionnement de la rétine, de réduire le bruit et de générer des événements qui dépendent de différentes propriétés spatiales. Cette thèse a donc deux motivations principales :

- La première est d'essayer de modéliser l'apprentissage visuel effectué par un expert sportif dans une tâche qui est propre à son sport. Le but initial étant premièrement de comprendre comment cette expertise se forge au cours du temps et d'étudier si un

---

2. Dans ce manuscrit nous distinguons les événements (signaux en sortie d'un capteur) des spikes (signaux en sortie d'un neurone). Cependant, ces deux signaux sont extrêmement similaires (binaires et situés temporellement).

3. <https://yumain.fr/>



---

modèle bio-inspiré permettrait d'apporter des réponses à cette question. Secondement, il s'agit d'évaluer les performances du SNN sur une tâche de prédiction face à celle d'un humain expert ou non.

- La seconde motivation est de finaliser et d'adapter pour les objectifs de la thèse la caméra événementielle développée par Yumain et de la tester à travers un SNN. Le but étant par la suite pour Yumain de développer un SNN embarqué dans une carte FPGA.

La tâche de prédiction effectuée par le SNN sera une tâche de prédiction de trajectoire de balle. Comme il a été présenté ci-dessus, un joueur sportif a constamment besoin de prédire pour pouvoir performer dans son sport. Dans le cas d'un joueur recevant un service de tennis par exemple, celui-ci doit prédire la trajectoire de la balle afin de commencer à se déplacer dans la bonne direction de celle-ci et pouvoir la renvoyer de façon optimale. Ce même type de prédiction sera effectué avec des trajectoires de balles qui seront présentées et le SNN devra prédire la position spatiale finale de la balle. Des participants devront effectuer cette même tâche afin d'évaluer les différences de performances entre le SNN et ces participants.

L'objectif sera alors de voir si un SNN bio-inspiré peut apprendre des caractéristiques spatio-temporelles des trajectoires de balles afin de pouvoir par la suite les prédire. Ensuite, il s'agit d'évaluer les performances d'un système utilisant une seule caméra (vision monoculaire) ou de deux caméras (binoculaire), afin de permettre une estimation des profondeurs et d'affiner l'analyse et la prédiction faite par le SNN. Ces différentes motivations et objectifs seront par la suite révisés et affinés à la fin de l'état de l'art afin de mettre en avant ce qui a été accompli durant cette thèse en fonction de l'état actuel des connaissances, du projet et du temps disponible au cours d'une thèse de trois ans.

Cette thèse a été financée par le projet ANR "*Beating Roger Federer* : Modéliser l'apprentissage et l'expertise visuelle grâce à un réseau neuronal bio-inspiré" (ANR-16-CE28-0017-01) initié en 2017. Les objectifs de ce projet reprennent les motivations énumérées ci-dessus. Avant le début de cette thèse, trois stagiaires ont travaillé sur ce projet. Ceux-ci ont participé au développement de la caméra NeuroSoc.

## 1.2 Architecture de ce manuscrit de thèse

Ce manuscrit sera le résumé des réflexions eues durant ces trois années de doctorat. Bien que tous les travaux ne peuvent pas être résumés en un manuscrit de thèse, il essaiera d'en résumer les principaux et les plus pertinents. Pour cela, le prochain chapitre 2 fera un résumé de l'état actuel des connaissances des différentes thématiques abordées durant cette thèse, à savoir, la vision chez l'humain, les caméras événementielles, les SNNs, etc. Comme précisé ci-dessus, l'état de l'art se conclura par une problématique finale reprenant les différents objectifs et motivations énumérés ci-dessus.

S'en suivront alors les différents travaux effectués durant cette thèse, avec dans un premier temps un chapitre 3 de méthodes. Ce chapitre permet de décrire les choix et détails techniques permettant de simuler le SNN utilisé durant cette thèse ainsi que les méthodes communes aux différentes études qui suivront.

Trois différents chapitres (3.2, 4, 5) viendront à la suite de la partie méthode. Ceux-ci sont chronologiquement organisés et représentent donc l'évolution des études effectuées durant cette thèse. Le premier de ces trois chapitres (chapitre 3.2) présente les premiers travaux effectués permettant de tester le SNN sur des stimuli simulés, ainsi qu'une première expérience permet-

---

tant de tester le SNN et la caméra NeuroSoc. Cette première partie a pour but d'installer les bases de cette thèse et faire une preuve de concept qu'un SNN recevant des spikes générés par la NeuroSoc peut analyser et prédire des trajectoires de balles.

Ensuite, le chapitre 4 présente une étude réalisée durant cette thèse et publiée dans le journal *Frontiers in computational neuroscience* (DEBAT et al. 2021). Dans cet article, nous avons étudié la sélectivité des neurones et la capacité du SNN à prédire des trajectoires (réelles) de balles. Les performances de notre système sont ensuite comparées aux performances d'humains auxquels il a été demandé d'effectuer la même tâche.

Le chapitre 5 vient valider ces résultats dans une tâche plus appliquée au sport. En effet, cette fois-ci les trajectoires enregistrées sont cette fois-ci des trajectoires de lancer-franc de basket. Le même protocole est ensuite appliqué, les performances de prédiction du SNN sont évaluées puis confrontées à celles d'humains experts et non-experts.

Le temps passé au développement et au test de la caméra en début de thèse, puis à son utilisation au cours des différentes expériences, m'a permis de mettre en avant quelques limites de son processus de génération de spikes. Le chapitre 6 souligne certaines de ces limites et propose une alternative afin de pallier celles-ci et ainsi d'améliorer le processus de génération de spikes. Une analyse de la précision temporelle des spikes générés est également effectuée.

Enfin, ce manuscrit se clôture sur le chapitre 7, qui conclue ces trois années de thèse et discute des résultats obtenus ainsi que de l'état de l'art actuel des SNNs et de ses perspectives.



# Chapitre 2

## Etat de l'art

### 2.1 Traitement visuel

#### 2.1.1 Neurones

Le cerveau humain est composé d'environ 85 milliards de neurones (R. W. WILLIAMS et HERRUP 1988), qui permettent de traiter et transmettre l'information au sein des différentes aires cérébrales et différents organes du corps. Le signal utilisé par le neurone pour transmettre l'information est le potentiel d'action (aussi appelé "spike"), qui est un évènement bref (de l'ordre de la milliseconde), de type tout ou rien. L'amplitude du pic d'un spike est constante quelque soit l'intensité du signal reçu par le neurone. (cf. figure 2.1).

Un neurone est composé de différentes parties. Les dendrites, qui sont les parties réceptrices du neurone, reçoivent les spikes émis par d'autres neurones (dits afférents ou pré-synaptiques), puis transmettent l'information au sein du soma. Le soma intègre les différents spikes reçus et transmet à son tour un spike lorsqu'il reçoit suffisamment de spikes. Le soma peut être caractérisé par son potentiel de membrane. Lorsqu'il reçoit un spike, son potentiel de membrane augmente ou diminue, selon que le neurone pré-synaptique qui envoie le spike est excitateur ou inhibiteur. Puis le potentiel de membrane revient progressivement à sa valeur de potentiel de repos ( $\approx -70mV$ ) en l'absence de spike. Lorsqu'il reçoit plusieurs spikes dans une courte fenêtre temporelle, ceux-ci se cumulent. Si le potentiel de membrane dépasse une valeur de seuil ( $\approx -55mV$ ), s'en suit une forte dé-polarisation, suivie d'une rapide re-polarisation du potentiel de membrane, qui génère ainsi un spike qui sera transmis par l'axone aux autres neurones. Le neurone rentre ensuite en période réfractaire, période durant laquelle les spikes reçus n'auront que peu d'effets sur son potentiel de membrane. La dernière partie nécessaire au transfert de l'information est la synapse. Elle est la connexion entre deux neurones, entre l'axone d'un neurone pré-synaptique et une dendrite d'un neurone post-synaptique, qui permet de transmettre l'information entre ces deux neurones.

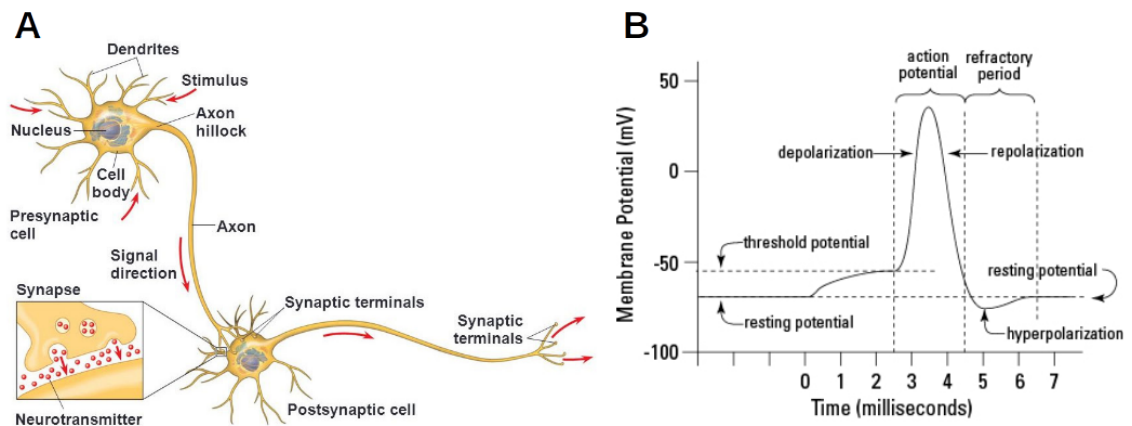


FIGURE 2.1 – Panneau A : Schéma de deux neurones (REZAEI-MAZINANI 2017). Panneau B : Évolution du potentiel de membrane d'un neurone <sup>1</sup>.

## 2.1.2 Rétine

La rétine est l'organe humain/non-humain permettant de capter les rayons lumineux émis par l'environnement extérieur et ensuite la retranscrire en information visuelle. Ces rayons traversent dans un premier temps différents niveaux comme la cornée et le cristallin puis viennent ensuite se projeter sur la rétine et plus particulièrement sur les photorécepteurs. D'une épaisseur de 300-400  $\mu\text{m}$  et d'une surface de 520  $\text{mm}^2$  (PASTERNAK, BISLEY et CALKINS 2003), la rétine est divisée en deux parties distinctes. Une première partie centrale constituée de la fovéa ( $5^\circ$  du champs visuel) principalement constituée de cônes et qui permet une perception fine, précise de l'information et de percevoir la couleur. Pour cela, trois différents types de cônes existent et codent pour des longueurs d'ondes différentes (Short=bleu, Medium=vert et Long=rouge). La seconde partie est la vision périphérique, beaucoup moins précise, qui ne permet pas de traitement de la couleur et est principalement constituée de bâtonnets. La rétine est composée de  $\simeq 90$  millions de bâtonnets et 5 millions de cônes. Ces photorécepteurs transduisent la luminosité en activité électrique. Cette information est ensuite transmise aux cellules bipolaires, horizontales et amacrines puis aux cellules ganglionnaires qui codent cette activité en spikes. L'information est divisée selon plusieurs voies, dont les deux principales sont (Ali YOONESSI et Ahmad YOONESSI 2011) (figure 2.2) :

- La voie parvocellulaire : recevant principalement l'information provenant du centre de la rétine. Les neurones de cette voie possèdent de petits champs récepteurs et sont plus sensibles aux couleurs et aux détails
- La voie magnocellulaire : recevant principalement l'information provenant de la périphérie de la rétine. Cette voie dont les neurones possèdent de plus grands champs récepteurs, permet une propagation plus rapide des potentiels d'actions le long du nerf optique, et qui est plus sensible aux changements de luminosités et donc aux mouvements.

1. <https://www.cliffsnotes.com/study-guides/anatomy-and-physiology/nervous-tissue/transmission-of-nerve-impulses>

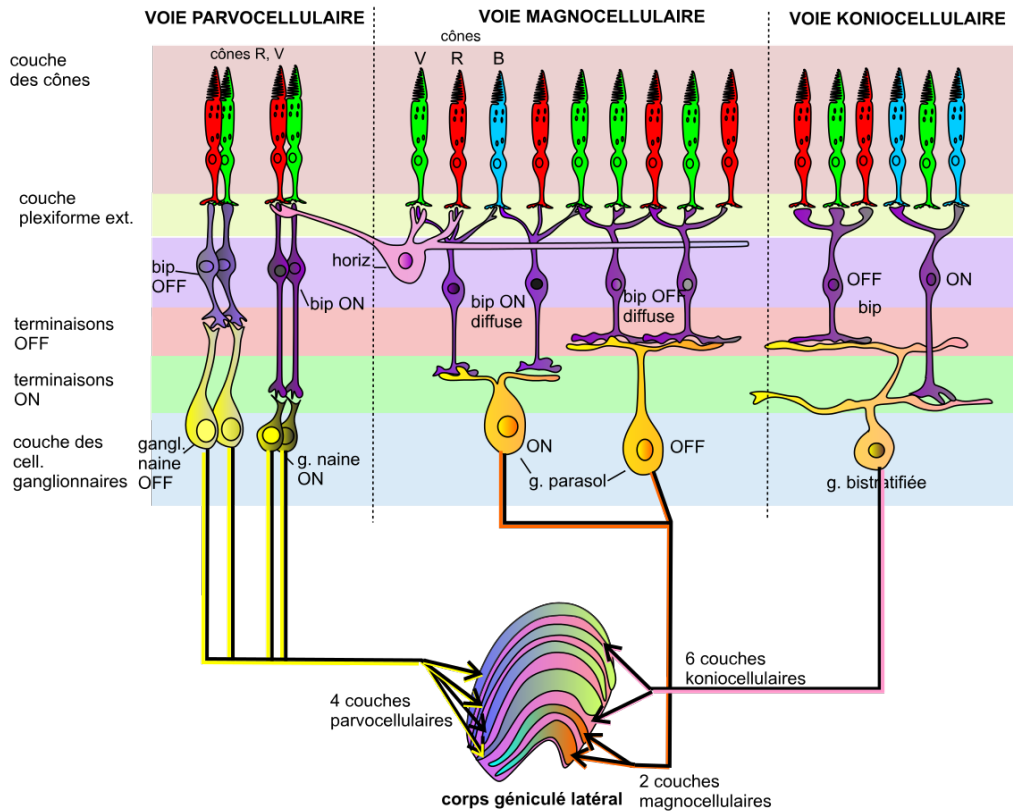


FIGURE 2.2 – Connectivité de la rétine

L'information visuelle domine les autres informations sensorielles. En effet, lors d'informations non-concordantes entre différents capteurs sensoriels, l'information visuelle sera privilégiée (POSNER, NISSEN et KLEIN 1976). La rétine et donc la vision jouent ainsi un rôle majeur dans la perception (HOON et al. 2014 ; HILDEBRAND et FIELDER 2011).

### 2.1.3 Cortex visuel

Le cortex visuel rassemble différentes aires visuelles qui ont pour objectifs de traiter l'information visuelle provenant du LGN, lui-même intégrant les informations provenant de la rétine. Le cortex visuel est composé de différentes aires visuelles, hiérarchiquement organisées. Ainsi, pour résumer l'organisation des différentes (figure 2.3, panneau B et C), le cortex visuel primaire (aire V1) reçoit principalement l'information provenant du LGN, qui traite et transmet l'information à l'aire V2, V3, puis est ensuite divisé en deux voies distinctes. La voie ventrale principalement impliquée dans l'identification et la catégorisation de formes et d'objets. La voie dorsale impliquée dans le contrôle visuo-moteur, en traitant des propriétés comme la position, l'orientation ou la taille d'un objet, bien que cette affirmation est de plus en plus contestée, car de très fortes interactions et connexions sont présentes entre ces deux voies (HEBART et HESSELMANN 2012). Les neurones des différentes aires ont des sélectivités qui évoluent et se complexifient au fur et à mesure que l'on avance dans les aires visuelles. En intégrant les neurones des précédentes couches, les champs récepteurs des neurones se complexifient et se spécialisent, comme par exemple les neurones de V1 qui intègrent la sélectivité des cellules ganglionnaires (de type centre-pourtour), pour obtenir des champs récepteurs de types Gabor orientés (HUBEL et WIESEL 1962) (figure 2.3, panneau A). Ainsi, les neurones de la première aire visuelle (V1) peuvent décoder des in-

formations locales comme reconnaître des orientations (grâce au champs récepteurs de types Gabor), des directions (tout en étant sujet au problème d'ouverture (PACK, BEREZOVSKII et BORN 2001)), des disparités binoculaires (OHZAWA, DEANGELIS et FREEMAN 1990).

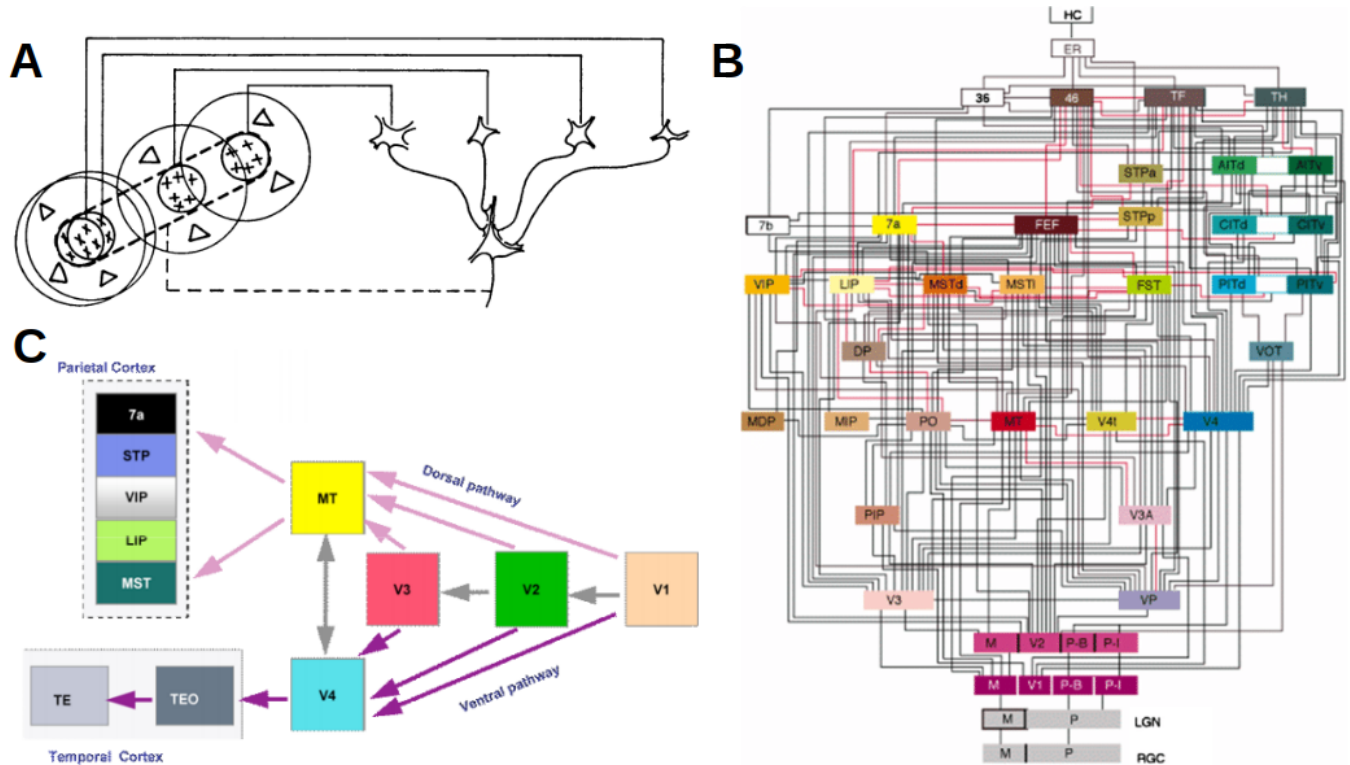


FIGURE 2.3 – Panneau A : modèle proposé permettant d’expliquer la formation d’un champ récepteur de type Gabor à partir des champs récepteurs des neurones du LGN (image provenant de (HUBEL et WIESEL 1962)). Panneau B : Hiérarchie des connexions entre les différentes aires visuelles (FELLEMAN et VAN ESSEN 1991). Panneau C : Simplification des différentes hiérarchies entre les aires (images provenant de (PASTERNAK, BISLEY et CALKINS 2003)).

L’aire V2, aura une sélectivité proche avec des champs récepteurs plus grands. Puis, l’aire MT, faisant partie de la voie dorsale et spécialisée dans le traitement du mouvement, est aussi sélective à la direction, à la profondeur, à la disparité binoculaire ainsi qu’à la vitesse, avec des sélectivités plus complexes (PASTERNAK, BISLEY et CALKINS 2003), comme une sélectivité à la disparité binoculaire relative (PARKER 2007). Puis encore plus haut dans la hiérarchie, l’aire MST permet un traitement du flux optique (PASTERNAK, BISLEY et CALKINS 2003). Une des caractéristiques des premières aires visuelles est qu’elles sont rétinotopiques. C’est à dire, qu’il y a une organisation spatiale de l’information traitée par le cortex visuel similaire à l’organisation spatiale perçue. Ainsi deux points proches dans la scène perçue par la rétine seront aussi proches et organisés spatialement de la même façon dans le cortex visuel (ENGEL, GLOVER et WANDELL 1997).

---

## 2.2 Traitement visuel artificiel

Depuis longtemps, l'humain s'est inspiré du vivant pour innover. Léonard de Vinci par exemple, s'était inspiré des oiseaux et des chauve-souris pour inventer des machines volantes. Depuis la fin des années 1990, s'inspirer du vivant pour l'innovation, aussi appelé bio-mimétisme, a gagné en popularité (KENNEDY et al. 2015). L'acquisition d'informations visuelles et plus particulièrement le traitement automatique de l'information en sont des exemples concrets, avec l'apparition des ANNs qui ont permis de grandes avancées dans le traitement de l'information visuelle. Plus récemment, l'arrivée de nouvelles caméras événementielles cherchant à mimer le fonctionnement de la rétine a ouvert de nouvelles opportunités dans le domaine du traitement visuel.

### 2.2.1 Capteurs visuels

Afin d'analyser un mouvement ou une image, la méthode la plus commune est d'utiliser une caméra de type "frame-based". Celle-ci agit comme un appareil photo qui prendrait des photos successives de manière périodique (figure 2.4). Cette suite d'image est ensuite utilisée comme vidéo.

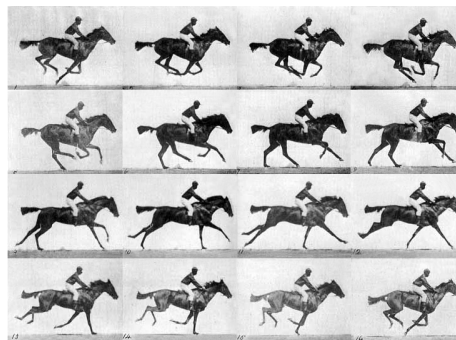


FIGURE 2.4 – Photographie animée de James Muybridge

Les premiers systèmes de caméras à base de capteurs d'images CMOS apparurent peu avant 1990. Depuis, ceux-ci ont très largement évolué et se sont démocratisés, couvrant différentes applications (films, appels vidéos, analyse vidéo, etc). Leurs évolutions ont permis d'améliorer la qualité vidéo, la résolution ainsi que la fréquence d'acquisition des images. Cependant, ces caméras présentent quelques limites. A l'opposé de la rétine, celles-ci fonctionnent de manière synchrone et peuvent entraîner des latences dues à la nécessité d'attendre la prochaine image (FARABET et al. 2012). Elles peuvent également présenter des bruits liés aux mouvements (B. MA et al. 2016) et nécessitent que tous les pixels soient encodés (et donc ensuite décodés lorsqu'ils sont analysés). Ceci peut représenter une consommation non-nécessaire dans certains cas d'utilisation lorsqu'il n'y a pas ou peu de mouvements/variations, il ne semble alors pas nécessaire d'encoder/décoder un pixel qui ne varie pas, car il n'apporte pas d'information supplémentaire.



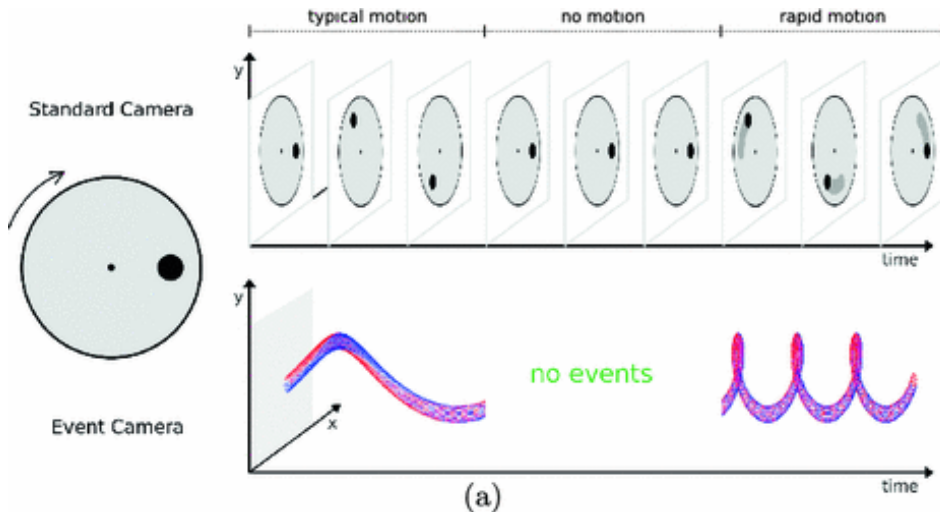


FIGURE 2.5 – Différences entre des caméras standard (*frame-based*, graphique du haut) et asynchrones (*event-based*, graphique du bas) (H. KIM, LEUTENEGGER ET DAVISON 2016)

Une caméra dite *frame-based* encode de façon périodique la scène visuelle qui lui est présentée. La rétine quant à elle et plus particulièrement la voie magnocellulaire, fonctionne de manière asynchrone et est sensible aux variations rapides de luminosités (Ali YOONESSI et Ahmad YOONESSI 2011). De nouveaux capteurs (LICHTSTEINER, C. POSCH et T. DELBRUCK 2008 ; SON et al. 2017 ; BRANDLI et al. 2014 ; Christoph POSCH, MATOLIN et WOHLGENANNT 2011) ont alors cherché à mimer ce mode de fonctionnement (Christoph POSCH, Teresa SERRANO-GOTARREDONA et al. 2014 ; STEFFEN, REICHARD et al. 2019) afin de pallier les problèmes des caméras "frame-based" cités ci-dessus (figure 2.5). Ces capteurs n'encodent pas une image complète à une fréquence déterminée, mais évaluent continuellement l'intensité de la valeur de chaque pixel. Lorsque la variation d'intensité logarithmique entre pixels à l'instant  $t$  ( $I_{x,y,t}$ ) et celle de son intensité ( $I_{x,y,t_{last}}$ ) à l'instant  $t_{last}$  (lors du précédent événement) dépasse une valeur de seuil, un événement est alors généré (voir équation 2.1. Les événements sont polarisés : si cette variation est positive, l'événement transmis est de type ON (augmentation de l'intensité lumineuse), sinon de type OFF.

$$|\log(I_{x,y,t}) - \log(I_{x,y,t_{last}})| \geq I_{thresh} \quad (2.1)$$

Pour cela, les DVS peuvent être divisées en trois parties (figure 2.6, panneau A). La première mime le principe des photorécepteurs et fait l'acquisition de la luminosité de la scène. De plus, les pixels logarithmiques permettent d'effectuer un contrôle de gain. La seconde partie détecte et amplifie les variations de luminosité, puis la troisième génère les spikes ON/OFF selon la polarité de la variation d'intensité (figure 2.6, panneau B).

Ainsi, comme on peut le voir sur la figure 2.5, la caméra événementielle, ne générera des événements que lorsqu'il y a du mouvement. Le reste de l'information qui ne varie pas n'est pas transmise, permettant ainsi de réduire la quantité d'information redondante. De plus, le traitement asynchrone de ces caméras permet de résoudre les problèmes de bruits liés aux mouvements rapides.

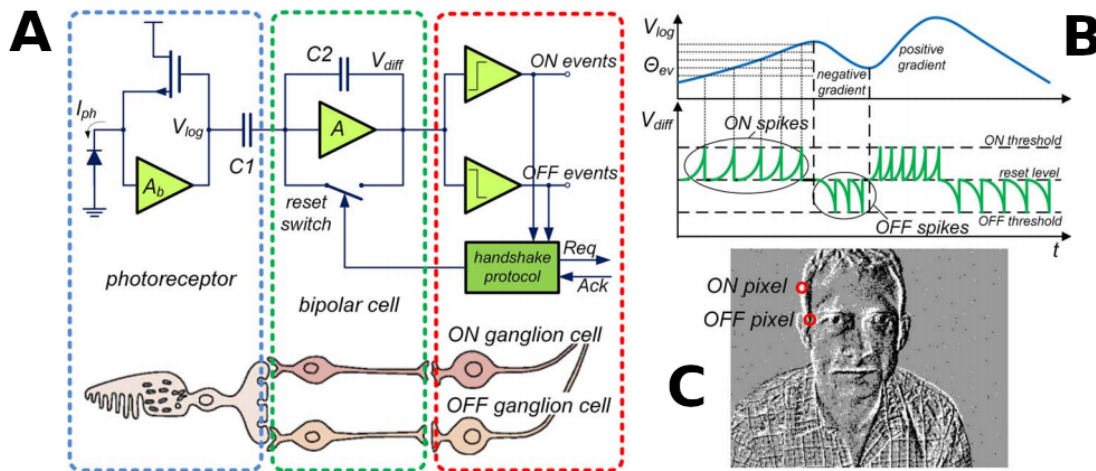


FIGURE 2.6 – A : Circuit d'une DVS et leurs comparaisons avec les cellules de la rétine. B : Génération de spikes en fonction de la variation de luminosité. C : représentation visuelle des spikes générés. Figure provenant de (Christoph POSCH, Teresa SERRANO-GOTARREDONA et al. 2014)

Ce mode de fonctionnement permet alors de n'encoder seulement les variations de luminosité (principalement dues aux mouvements d'un objet de la scène ou de la caméra), de profiter d'une grande gamme dynamique (120dB), de consommer peu d'énergie (24mW), d'avoir une très grande résolution temporelle (1 $\mu$ s) et de transmettre les spikes et donc les informations avec très peu de latences (de l'ordre de quelques  $\mu$ s).

Trois principales caméras asynchrones - également appelées caméras événementielles - existent actuellement. Tout d'abord la DVS (LICHTSTEINER, C. POSCH et T. DELBRUCK 2008) qui reprend le fonctionnement présenté ci-dessus. Une autre caméra asynchrone est la *Asynchronous Time Based Image Sensor* (ATIS) (Christoph POSCH, MATOLIN et WOHLGENANT 2011), qui génère un événement supplémentaire afin d'envoyer une information sur la valeur absolue du pixel qui génère un événement. Ainsi, deux événements sont générés, le premier similaire à ceux générés par une caméra DVS, puis le second avec une latence proportionnelle à la valeur de luminosité. Si la valeur du pixel est faible (sombre), la latence entre le premier et second pixel sera forte, sinon elle sera faible. Le troisième type de caméra est la *Dynamic and Active Pixel Vision Sensor* (DAVIS) (BRANDLI et al. 2014; BERNER et al. 2013), qui intègre une caméra synchrone (*frame-based*) avec le capteur visuel asynchrone, permettant ainsi d'utiliser les deux types d'informations (TEDALDI et al. 2016). Ces types de capteurs visuels souffrent cependant d'un manque de résolution spatiale (GALLEGO et al. 2020), qui est largement inférieur aux résolutions des capteurs de types *frame-based*. De plus, les capteurs *frame-based* profitent d'une certaine stabilité sur le marché, du fait qu'ils existent depuis plus longtemps et sont très largement démocratisés (webcam, téléphone, etc...). Les applications et les algorithmes les utilisant sont donc bien souvent adaptés à leurs caractéristiques et tirent profit de leurs avantages (tâche de détection/catégorisation sur des images) ou des méthodes existent pour pallier leurs défauts (PORTZ, L. ZHANG et JIANG 2012; Z. MA et al. 2015; B. MA et al. 2016). De plus les algorithmes pour traiter ces capteurs sont souvent eux aussi plus performants que ceux utilisant des spikes, bien que les algorithmes utilisant des spikes continuent à s'améliorer et commencent à les concurrencer. Les caméras événementielles sont cependant très intéressantes lors d'applications utilisant des stimuli avec des mouvements très rapides (HOWELL et al. 2020) ou sur des architectures nécessitant une consommation de ressources limitée et un traitement en temps réel comme en robotique ou pour les drones par exemple.

---

Les capteurs asynchrones ne se sont pas restreints aux capteurs visuels, mais il se sont aussi étendus aux capteurs audio (S.-C. LIU et al. 2014) et tactiles (TAUNYAZOV et al. 2020).

D'autres méthodes cherchent à mimer beaucoup plus précisément le fonctionnement de la rétine (WOHRER et KORNPBST 2009; MAHOWALD et MEAD 1991; HERAULT 1996; CARANDINI et al. 2005), comme par exemple *Virtual Retina* (WOHRER et KORNPBST 2009), nouvellement appelé *PRANAS* (CESSAC et al. 2017), qui divise le traitement selon trois étapes (couche plexiforme externe, couche plexiforme interne et les cellules ganglionnaires). La première couche effectue un filtrage spatio-temporel, la seconde contrôle le gain du contraste et augmente la parcimonie des spikes émis par les cellules ganglionnaires, la dernière génère les spikes de sortie en utilisant un modèle de neurones intégrateurs. Cependant, ce type de méthode utilise des vidéos "frame-based" et nécessite de nombreux traitements. Ils ne sont donc pas adaptés à une extraction de spike en temps réel, mais permettent une modélisation plus détaillée de la rétine pouvant être utilisée pour ensuite modéliser le LGN et la V1 plus fidèlement (MASQUELIER 2012).

## 2.2.2 ANN/SNN

A l'origine, les réseaux de neurones artificiels tiraient leurs inspirations des systèmes neuronaux humains/non-humains. Cela a commencé en 1943 avec le neurophysiologiste Warren McCulloch et le mathématicien Walter Pitts, qui ont modélisé le fonctionnement de neurones pour expliquer comment leur comportement. La recherche dans ce domaine s'est ensuite poursuivie en 1949 avec les travaux de Donald Hebb, mettant en avant que des neurones qui déchargent ensemble ont des connexions plus fortes, souvent résumés par : "*neurons that fire together wire together*" (LOWEL et SINGER 1992). Apparaît donc la règle de Hebb et les premiers réseaux de neurones qui apprennent par expérience. Depuis, les méthodes d'apprentissage et les différentes architectures et applications des réseaux de neurones se sont très largement développées. Le domaine de la vision par ordinateur est l'un des domaines dans lequel l'intelligence artificielle s'est le plus développé avec des performances impressionnantes dans des tâches de catégorisation et de détection par exemple (GU et al. 2018; RAWAT et Z. WANG 2017). Cependant, dans la course à l'optimisation et à la performance, les réseaux de neurones artificiels se sont bien souvent écartés des processus neuronaux biologiques (back-propagation, architecture, etc...) et sont très spécialisés dans une tâche précise pour laquelle ils ont été entraînés. C'est pourquoi l'intelligence artificielle est souvent qualifiée de verticale (très spécialisée dans un domaine) en opposition à l'intelligence naturelle horizontale (qui n'est pas optimale, mais permet de répondre à toute sorte de problèmes ou presque). Néanmoins, il y a une corrélation entre l'activité des neurones artificiels d'un CNN et celle des aires visuelles humaines. Et, plus le CNN est performant, plus cette corrélation est forte (SCHRIMPF, KUBILIUS, HONG et al. 2018; SCHRIMPF, KUBILIUS, M. J. LEE et al. 2020), excepté pour les CNN les plus performants pour lesquels cette corrélation semble diminuer. La plupart des réseaux de neurones artificiels actuels utilisent des valeurs analogiques et traitent l'information de manière périodique. Dans le cas de l'analyse vidéo, chaque image est analysée périodiquement (en fonction du FPS de la caméra). Comme expliqué dans les sections 2.2.3 et 2.1.2, les neurones biologiques traitent l'information de manière asynchrone et n'utilisent pas de signaux analogiques, mais utilisent les spikes qui sont des événements binaires. La valeur analogique utilisée dans les ANNs peut cependant correspondre à la fréquence d'une décharge d'un neurone (SCHRIMPF, KUBILIUS, HONG et al. 2018).

L'un des principaux inconvénients des ANNs est leur grande consommation de ressources qui peut nécessiter plusieurs centaines de Watts pour certaines architectures. A l'opposé, le cerveau humain utilise approximativement 30W pour toutes les tâches qui lui sont demandées (vision, audition, attention, mouvement, etc) (MAASS 2015). Comme pour la rétine (Section 2.1.2), en s'inspirant du cerveau, les SNNs essaient de répondre à ce problème en proposant des systèmes

---

moins gourmands en énergie et profitant du caractère éparé de la transmission d'informations par des spikes.

### 2.2.3 Modèles de neurones

Les SNNs et plus largement les réseaux de neurones se caractérisent par un ensemble de neurones artificiels connectés entre eux. Il existe plusieurs modèles de neurones dits bio-inspirés, qui se différencient par leur similitude avec les neurones biologiques et par leur complexité de calculs et de paramètres (IZHIKEVICH 2004 ; PAUGAM-MOISY et S. BOHTE 2012).

Afin de reproduire le comportement des neurones de manière artificielle et simulée, différents modèles de neurones à spikes ont été proposés. Ceux-ci diffèrent principalement sur la méthode utilisée pour calculer l'évolution temporelle du potentiel de membrane. Cependant, la fonction globale du neurone reste proche : lorsqu'il reçoit un signal en entrée (spike), le potentiel de membrane augmente et diminue proportionnellement au poids de la connexion synaptique entre le neurone pré et post-synaptique. Puis, le potentiel de membrane tend à revenir progressivement à son potentiel de repos, sauf si le potentiel de membrane dépasse une valeur de seuil. Alors, le neurone post-synaptique émet un spike.

Des modèles précis existent permettant de simuler avec précision les comportements des neurones biologiques, mais nécessitent en contre-partie des calculs longs et coûteux les rendant difficilement utilisables dans des SNNs à grandes échelles. D'autres modèles utilisent des méthodes plus simplifiées afin de pouvoir les utiliser dans des SNNs plus grands. Toutefois, ces derniers s'écartent davantage des modèles biologiques.

#### 2.2.3.1 Hodgkin-Huxley

Un des principaux modèles de neurone biologique est le modèle à conductance, dont celui de *Hodgkin-Huxley* (HODGKIN et HUXLEY 1952). En se basant sur le fonctionnement axonal des calmars, les auteurs ont proposé un modèle très proche du fonctionnement biologique d'un neurone qui permet de décrire certains de ses comportements avec précision. Ce modèle est le premier à décrire l'évolution de l'état du neurone avec des résistances et des condensateurs. Le modèle utilise plusieurs équations différentielles, ainsi que de nombreux paramètres, qui vont permettre de reproduire l'évolution des différents canaux ioniques. Cependant, sa complexité fait qu'il est principalement utilisé pour modéliser avec précision le comportement de neurones mais peu adapté pour l'utilisation de réseaux de neurones à grandes échelles (IZHIKEVICH 2004 ; PAUGAM-MOISY et S. BOHTE 2012).

#### 2.2.3.2 Izhikevich

Le modèle Izhikevich (IZHIKEVICH 2003), fait partie des modèles développés à la suite du modèle *Hodgkin-Huxley*. Il est défini par deux équations différentielles (2.2). Il est ainsi un bon compromis entre le biologiquement plausible et l'efficacité computationnelle.

$$\begin{aligned} \frac{dv}{dt} &= 0.04v(t)^2 + 5v(t) + 140 - w(t) + I(t) \\ \frac{dw}{dt} &= a(bv(t) - w(t)) \end{aligned} \tag{2.2}$$

si  $v \geq v_{Thresh}$  alors :  $v \leftarrow c$  et  $w \leftarrow w + d$

Avec a, b, c et d des paramètres qui influent sur le comportement du neurone.

### 2.2.3.3 Neurone intégrateur à fuite (LIF)

Le modèle neurone intégrateur à fuite (LIF), est un modèle simplifié d'un neurone biologique. Il est défini par une équation différentielle linéaire du premier ordre (équation 2.3). Lorsque le neurone dépasse une valeur de seuil  $v_{Thresh}$ , le neurone émet un spike. Son potentiel de membrane retombe instantanément à son potentiel de repos et le neurone entre en période réfractaire. Bien que ce neurone ne reflète pas fidèlement un neurone biologique (difficulté de reproduire un train de spike mesuré expérimentalement lors d'un courant continu en entrée (FUORTES et MANTEGAZZINI 1962) par exemple), il a pour avantage de reproduire de façon générale le fonctionnement d'un neurone biologique tout en étant simple à modéliser. Il nécessite également peu de calculs et de paramètres, lui permettant ainsi d'être utilisé dans des réseaux de neurones à grandes échelles (BICHLER et al. 2012; PAREDES-VALLES, SCHEPER et CROON 2019; DIEHL et COOK 2015; DAVIES, SRINIVASA et al. 2018). De plus, il peut être mis à jour de façon événementielle (BICHLER et al. 2012) et donc tirer pleinement partie du caractère parcimonieux des spikes qu'il reçoit en entrée.

$$\begin{aligned} \lambda_{Leak} \frac{dv_i(t)}{dt} &= -R(v_i(t) - v_{Rest}) + I_i(t) \\ v_i(t) &= v_{Rest} \quad si \quad v_i(t) \geq v_{Thresh} \end{aligned} \quad (2.3)$$

Avec :

- $v_i$  : potentiel de membrane du neurone  $i$
- $v_{Rest}$  : potentiel de repos
- $v_{Thresh}$  : valeur de seuil
- $\lambda_{Leak}$  : facteur de fuite du neurone
- $I_i(t)$  : courant d'entrée

### 2.2.3.4 SRM

Un autre modèle a été proposé par (KISTLER, GERSTNER et HEMMEN 1997). Celui-ci se différencie du modèle LIF par le fait qu'il n'utilise pas d'équations différentielles. Le potentiel de membrane d'un neurone  $i$  est déterminé par l'équation (2.4) :

$$v_i(t) = \eta(t - t_i) + \sum_j \sum_{t_j^{(f)}} w_{i,j} \epsilon_{i,j}(t - t_i, t - t_j^{(f)}) + \int_0^\infty \kappa(t - t_i, s) I^{ext}(t - s) ds \quad (2.4)$$

Avec :

- $v_i$  : potentiel de membrane du neurone post-synaptique  $i$
- $w_{i,j}$  : poids de la synapse qui connecte le neurone pré-synaptique  $j$  au neurone post-synaptique  $i$
- $t_i$  : instant du dernier spike du neurone  $i$
- $t_j^{(f)}$  : instant des spikes émis par le neurone  $j$

L'évolution de ce modèle de neurone est ainsi déterminé par trois filtres  $\eta$ ,  $\epsilon(i, j)$  et  $\kappa$ . Le premier,  $\eta$  décrit l'évolution du potentiel de membrane du neurone  $i$ , lorsque celui-ci dépasse le seuil  $v_{Thresh}$ . Le second,  $\epsilon(i, j)$  définit l'évolution temporelle du potentiel de membrane lorsqu'il reçoit des spikes pré-synaptiques et  $\kappa$  lorsqu'il reçoit des courants externes (nul s'il n'en reçoit pas).

Ce neurone, bien que simplement défini, est plus général que le modèle LIF. L'équation différentielle définie en (2.3) peut être dérivée du modèle SRM (KISTLER, GERSTNER et HEMMEN 1997).

---

### 2.2.3.5 Neurone stochastique

Un dernier modèle fréquemment utilisé est un modèle stochastique. Parce que les intervalles de temps entre les spikes émis par un même neurone semblent aléatoires, il est donc proposé que la génération des spikes se fasse par un processus de Poisson et qu'ainsi la fréquence de décharge du neurone importe le plus (D. HEEGER et P. D. HEEGER 2000). Les spikes sont ainsi générés de façon à conserver une fréquence de décharge du neurone proportionnelle au signal reçu en entrée et avec des intervalles de temps aléatoires entre les différents spikes émis.

## 2.2.4 Connections

L'activité d'un neurone dépend du signal qu'il reçoit provenant d'autres neurones pré-synaptiques. Comme expliqué dans la section 2.1.3, le cortex visuel est hiérarchiquement organisé, l'information part de la rétine puis avance dans les différentes aires visuelles qui vont traiter l'information. Au fur et à mesure que l'on avance dans la hiérarchie des aires, les neurones déchargent pour des informations de plus en plus complexes et précises. Cependant, les neurones ne sont pas seulement connectés dans le sens de la hiérarchie (connexions *bottom-up* ou *feed-forward* (FF) : neurones V1, connectés aux neurones de V2, eux-même connectés à V3, etc). De nombreuses connexions vont aussi des aires supérieures aux aires inférieures ou intra-aires. Dans les SNNs et les ANNs de façon générale, cette organisation est respectée. Ainsi, les réseaux de neurones sont organisés en différentes couches de neurones, qui vont avoir une sélectivité qui va se complexifier avec la profondeur de ces couches. Et, les neurones peuvent se connecter aux couches supérieures, inférieures ou égales à celles dont ils font partie (figure 2.7).

### 2.2.4.1 *Feed-forward* (FF)

Les connexions sont faites hiérarchiquement, un neurone d'une couche  $l - 1$  est connecté et envoie des spikes aux neurones de la couche  $l$ , qui envoie ensuite des spikes aux neurones de la couche  $l + 1$ , ainsi de suite. Ce type de connexions est très souvent utilisé, le neurone n'a alors connaissance d'informations provenant seulement de la couche précédente. Il permet d'avoir des neurones qui deviennent sélectifs à des patterns de plus en plus spécifiques et globaux au fur et à mesure des couches. Ce type de connexion est le plus fréquemment utilisé et obligatoirement nécessaire dans les ANNs et SNNs afin de transmettre l'information aux couches supérieures. Bien que ce type de connexion ne soit pas majoritaire dans le cerveau (WYATTE, CURRAN et O'REILLY 2012), elles semblent représenter les connexions qui permettent de traiter l'information de façon rapide et "efficace". Les connexions récurrentes prendraient plus de temps (le temps que l'information avance jusqu'aux couches supérieures (FF), puis qu'elle revienne aux couches inférieures), ce qui semble peu compatible avec le traitement rapide de l'information visuelle. Par ailleurs, ces connexions récurrentes permettraient d'affiner la réponse des couches inférieures en utilisant l'information décodée par les couches supérieures. Le traitement serait donc dans un premier temps rapide et efficace (FF) (S. THORPE, FIZE et MARLOT 1996; HUNG et al. 2005; VANRULLEN 2008), puis l'information serait affinée par les connexions récurrentes (KAR et al. 2019; WYATTE, CURRAN et O'REILLY 2012).

### 2.2.4.2 Récurrent

#### Latéral

Un neurone d'une couche  $l$  est connecté à d'autres neurones de cette même couche  $l$ . Ce type de connexions est souvent utilisé pour inhiber les neurones proches d'un neurone spikant. Ce type de connexion est souvent utilisé sous forme de winner-take-all (WTA) (S.-R. KHERADPISHEH et al. 2018; CHAUHAN et al. 2018), lorsqu'un neurone décharge, il va interdire d'autres neurones

(souvent ceux qui sont connectés au même champs récepteur) de décharger. Une méthode plus biologiquement plausible, est de soustraire une valeur au potentiel de membrane de ces neurones (MASQUELIER, GUYONNEAU et S. J. THORPE 2008 ; DELORME, PERRINET et S. J. THORPE 2001 ; DIEHL et COOK 2015), afin d'éviter qu'ils déchargent. Ces mécanismes permettent ainsi d'éviter que des neurones n'apprennent des patterns similaires et participent à la sélectivité des neurones (PAREDES-VALLES, SCHEPER et CROON 2019).

Ces connexions peuvent aussi être excitatrices ou sous forme de chapeau mexicain (connexions excitatrices pour les neurones proches et inhibitrices pour ceux plus éloignés) (ADAMS et HARRIS 2014), qui permet d'exciter les neurones proches et d'inhiber les neurones spatialement plus éloignés.

## Feedback

Un neurone d'une couche  $l$  est connecté à des neurones d'une couche précédente  $l - 1$ . Ce type de connexion permet d'apporter des informations provenant de couches supérieures (et donc potentiellement plus spécialisées) à des couches inférieures. Elles sont aussi souvent utilisées dans des architectures de type *predictive coding*, qui vont chercher à prédire l'activité des neurones sur de courtes périodes. Ceci peut être fait en excitant les neurones voisins avant même qu'ils aient reçus leurs signaux d'entrée (couche  $l - 1$ ) (SHON, RAO et SEJNOWSKI 2004) ou à l'opposé de prédire l'activité des neurones afin de les inhiber. Ceci permet ainsi d'avoir des neurones qui codent et déchargent pour des erreurs. C'est à dire, lorsque le signal d'entrée reçu par le neurone est différent de celui attendu (ORORBIA 2020 ; MASUMORI, SINAPAYEN et IKEGAMI 2019 ; WACONGNE, CHANGEUX et DEHAENE 2012).

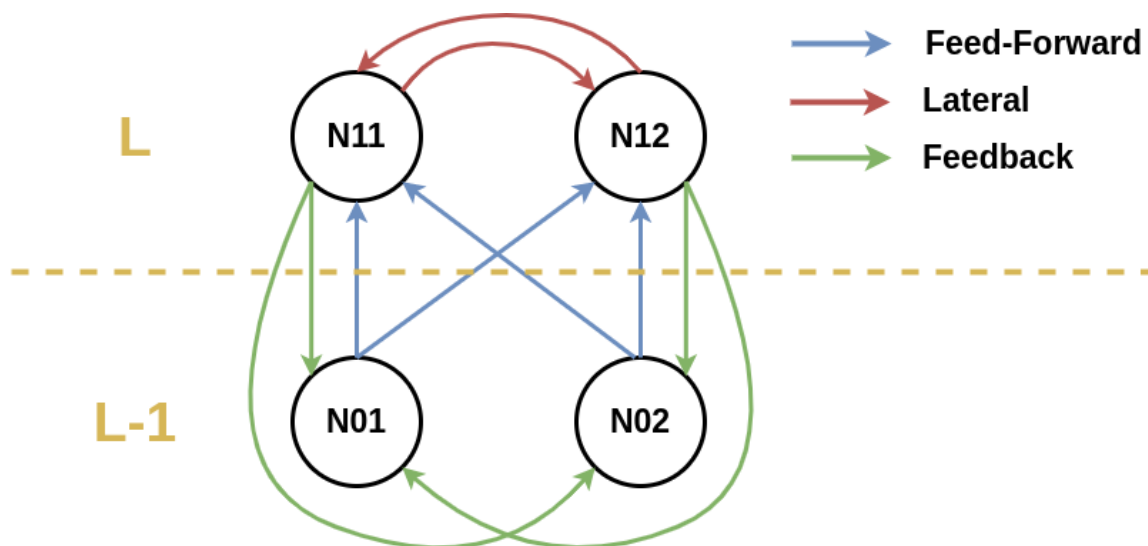


FIGURE 2.7 – Différents types de connexions

## 2.2.5 Architectures

### 2.2.5.1 Rétinotopique

Les neurones de la couche  $l$  sont organisés rétinotopiquement. Ils sont en effet connectés à une zone restreinte des neurones de la couche  $l - 1$  et leur organisation spatiale est similaire à celle du signal d'entrée. Un neurone a donc accès à une information locale et située spatialement. Plus on avance dans la hiérarchie du neurone, plus il sera connecté à une grande zone spatiale du signal d'entrée et donc aura accès à une information plus globale.

---

### 2.2.5.2 Convolutionnel

L'architecture convolutionnelle est une des architectures les plus utilisées dans les ANNs, qui a permis de nombreuses avancées dans le traitement d'image (KRIZHEVSKY, SUTSKEVER et HINTON 2017; RAWAT et Z. WANG 2017). Pour une couche, un certain nombre  $N_f$  de filtres convolutionnels sont définis/appris, puis sont appliqués sur l'image, donnant ensuite  $N_f$  cartes neuronales en sortie, représentant la sortie de la convolution de chaque filtre pour chaque position.

Un réseau de neurones convolutionnels à spike (CSNN) est un SNN avec une architecture rétinotopique présentée ci-dessus, à laquelle on va ajouter un système de *weight sharing* (partage de poids) (WS) (MASQUELIER et S. J. THORPE 2007; DIEHL et COOK 2015; PAREDES-VALLES, SCHEPER et CROON 2019; THIELE, BICHLER et DUPRET 2018; Milad MOZAFARI, GANJTABESH, NOWZARI-DALINI, S. J. THORPE et al. 2019). Lorsqu'on applique la règle d'apprentissage suite au spike d'un neurone  $j$ , tous les neurones liés au même filtre et donc partageant les mêmes poids synaptiques, mettent à jour leurs poids similairement au neurone  $j$ . Ainsi lorsqu'un pattern est appris à une position spatiale donnée, cet apprentissage est généralisé sur toute "l'image" et pourra donc être détecté à n'importe quelle position.

### 2.2.5.3 Dense

Tous les neurones de la couche  $l-1$  sont connectés à tous les neurones de la couche  $l$ . Ce type de couche permet ainsi de traiter l'information globale qui lui est présentée, contrairement aux architectures convolutionnelles qui ne prennent en compte qu'une partie locale de l'information.

### 2.2.5.4 Reservoir computing

Le *reservoir computing* est un type d'architecture neuronale qui se compose d'un réservoir, qui met en correspondance les entrées dans un espace à haute dimension, et d'un décodeur, qui analysera la sortie du réservoir. Le réservoir est fixe et seul le décodeur est entraîné avec une méthode simple telle que la régression linéaire.

Il existe deux principales architectures de *reservoir computing*, les *echo state network* (JAEGER 2001; JAEGER 2002; LAGORCE et al. 2015) et les *liquide state machine* (LSM), qui ont pour particularité d'utiliser des SNNs. Le LSM est une architecture très fortement récurrente de SNN, proposée par Maass (MAASS, LEGENSTEIN et al. 2002; MAASS, NATSCHLÄGER et MARKRAM 2002; BURGSTEINER et al. 2007; KAISER et al. 2017), avec comme constat que si on prend différents objets (les stimuli) et qu'on les jette dans un liquide donné, en regardant les ondes émises au niveau de l'eau (la perturbation) on peut déterminer le stimulus utilisé (forme de l'objet, position de contact avec l'eau, vitesse, ...)(FERNANDO et SOJAKKA 2003). Le LSM s'inspire donc de ce constat. Le liquide va être représenté par un ensemble de neurones fortement récurrents en appliquant un stimulus en entrée. En regardant les perturbations du liquide (spike émis par les neurones), on peut décoder la sortie de ce SNN pour obtenir des informations sur le stimulus en entrée.

## 2.2.6 Stratégie de codage

L'information est transmise entre les neurones grâce aux spikes (potentiels d'actions). Ceux-ci sont des événements binaires. Ainsi, individuellement, ils ne peuvent pas coder une grande quantité d'information hormis la présence ou non d'un stimulus. Différentes stratégies de codages ont ainsi été mises en avant afin d'expliquer la stratégie de transmission de l'information au sein du cortex et adapter les stratégies de codages pour les SNNs (AUGE et al. 2021). En effet, selon que l'intensité du stimulus augmente ou que celui-ci est plus ou moins proche du champs



---

récepteur du neurone, la réponse de celui-ci varie pour coder au mieux ces variations. Ces différentes méthodes de codage ne s'excluent pas entre elles. Au contraire, chacune présente ses propres avantages et inconvénients mais elles sont complémentaires. Le codage de l'information dans le cortex ne s'explique pas selon une seule méthode. Ces méthodes sont complémentaires et permettent d'encoder différents types d'information (BRETTE 2015). La figure 2.9 permet de donner un exemple de ces différents codages.

### 2.2.6.1 Codage fréquentiel

La première et la plus fréquemment utilisée est le codage par fréquence. Selon que l'intensité du stimulus augmente ou qu'il se rapproche du champs récepteur du neurone, celui-ci décharge à une fréquence qui augmente. Cette fréquence est donc proportionnelle au stimulus présenté. Cette méthode est souvent utilisée pour analyser la réponse de neurones à différents stimuli. Elle justifie les valeurs analogiques d'un ANN (la valeur analogique de la réponse d'un neurone correspond à sa fréquence de décharge) et peut aussi être utilisée comme modèle de neurone pour les SNN (voir section 2.2.3.5). Cependant, ce type de codage ne permet pas d'expliquer la rapidité du traitement visuel par exemple. En effet, le cortex visuel met seulement 100-150 ms pour traiter l'information provenant de la rétine, en traversant de nombreuses couches de neurones depuis les photorecepteurs de la rétine. Sachant qu'un neurone ne décharge que très rarement à une fréquence supérieure à 100Hz, cela voudrait dire que chaque neurone n'a le temps de décharger qu'une seule fois ou pas du tout afin de traiter l'information (S. THORPE, DELORME et VAN RULLEN 2001). Afin de déterminer la fréquence de décharge d'un neurone, il est nécessaire d'intégrer plusieurs spikes. Cela implique donc un délai pouvant aller jusqu'à plusieurs centaines de ms ou nécessite d'estimer cette fréquence de décharge en prenant en compte de nombreux neurones ayant la même fréquence et qui seraient donc redondants (GAUTRAIS et S. THORPE 1998).

### 2.2.6.2 Codage par latence

La seconde méthode proposée suite à la conclusion qu'un codage fréquentiel ne pourrait pas expliquer la rapidité du traitement visuel est le codage par latence. L'information du stimulus en entrée peut être convertie en latence : plus l'intensité du signal est forte et plus le neurone se chargera rapidement et donc déclenchera un spike tôt (S. THORPE et GAUTRAIS 1998) (voir figure 2.8). Cette méthode de codage a été mise en avant dans plusieurs aires du cerveau dont notamment les aires visuelles (FABRE-THORPE, RICHARD et S. J. THORPE 1998 ; SHRIKI, KOHN et SHAMIR 2012 ; REICH, MECHLER et VICTOR 2001 ; S. THORPE, FIZE et MARLOT 1996 ; MAINEN et SEJNOWSKI 1995), et la rétine (GOLLISCH et MEISTER 2008). Des études montrent que la précision temporelle des spikes est grande (de l'ordre de la milliseconde) (GOLLISCH et MEISTER 2008 ; BERRY, WARLAND et MEISTER 1997) et que celle-ci est donc viable pour décoder de l'information. Cette stratégie permet donc d'expliquer la rapidité du traitement visuel, car un seul spike par neurone est nécessaire pour décoder l'information portée par la latence.

Une règle d'intensité à latence est donc utilisée pour déterminer la latence du spike émis, pour convertir une image en train de spikes. La latence est définie de manière inversement proportionnelle à l'intensité du signal d'entrée (voir figure 2.8, panneau B). Dans le cadre d'une image, l'intensité de la valeur du pixel est utilisée pour définir la latence du spike généré. Ainsi, les pixels d'intensités plus élevées seront transmis en premier.

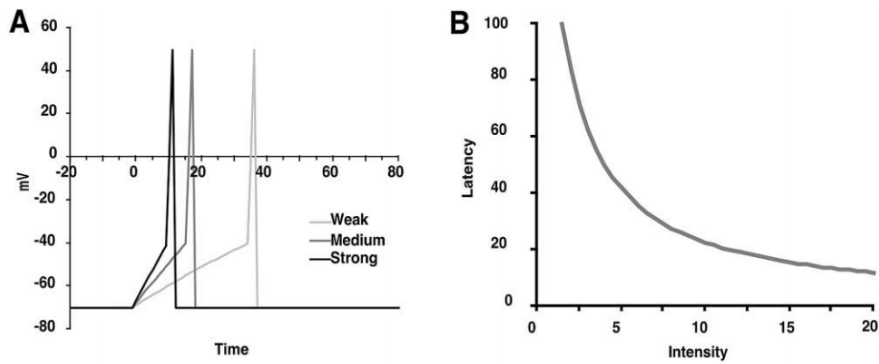


FIGURE 2.8 – Codage temporel, figure provenant de (S. THORPE, DELORME et VAN RULLEN 2001)

### 2.2.6.3 Codage par rang

Le codage par rang est une version simplifiée du codage temporel. La latence donnée au spike dépend du rang de son intensité reçue par rapport aux autres neurones, plutôt que de la valeur de cette intensité. Si un neurone A reçoit une intensité de 3 mV, un neurone B une intensité de 4 mV, et un neurone C une intensité de 1 mV, les spikes émis par les neurones A, B et C seront donc dans l'ordre  $B \rightarrow A \rightarrow C$  et également espacés dans le temps.

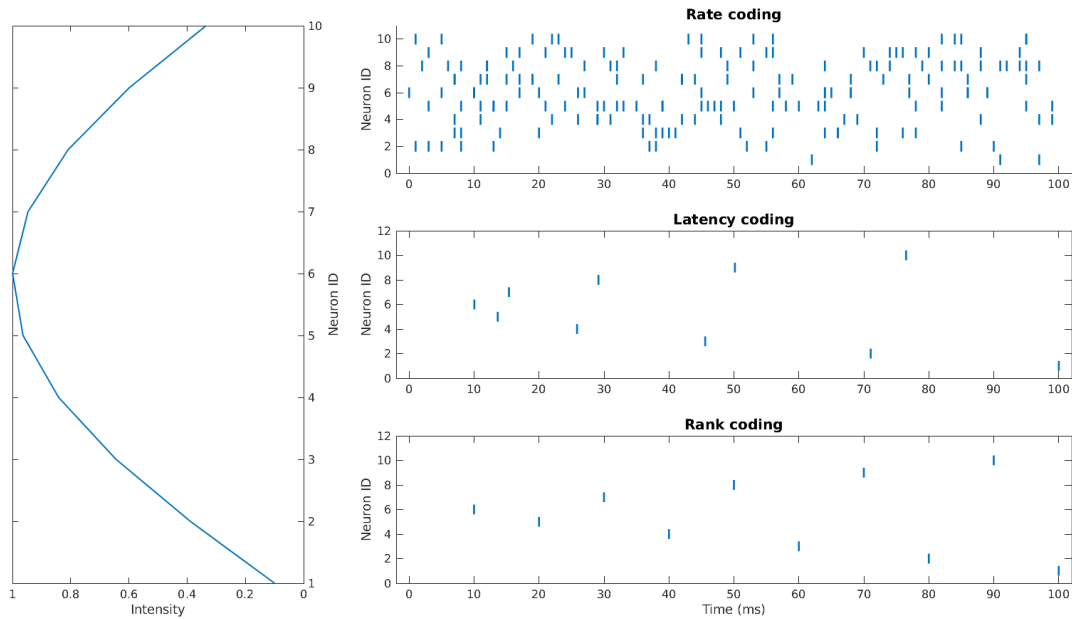


FIGURE 2.9 – Différentes méthodes de codage

### 2.2.6.4 Codage parcimonieux

Une des caractéristique des traitements de l'information sensorielle réalisés par les neurones est qu'ils effectuent un codage parcimonieux de celle-ci. Plutôt qu'avoir un grand nombre de neurones qui répondent de manière continue aux différents stimuli, seule une certaine partie de ces neurones vont répondre assez fortement à des stimuli bien spécifiques (Bruno A OLSHAUSEN et David J FIELD 2004). Les neurones sont donc la plupart du temps au "repos" et rarement

actifs. Ce type de stratégie, notamment utilisé dans V1, permet ainsi de représenter l'information visuelle selon un nombre restreint de structures primitives (comme les contours qui peuvent être extraits avec des DoG ou des Gabors orientés, voir figure 2.10) (Bruno A. OLSHAUSEN et David J. FIELD 1997). L'efficacité du codage parcimonieux a été montré par (GAUTRAIS et S. THORPE 1998). En convoluant une image avec des filtres DoG de types ON/OFF et de différentes échelles, ils ont montré qu'il est possible de reconstruire précisément l'image initiale avec un nombre restreint de neurones qui déchargent.

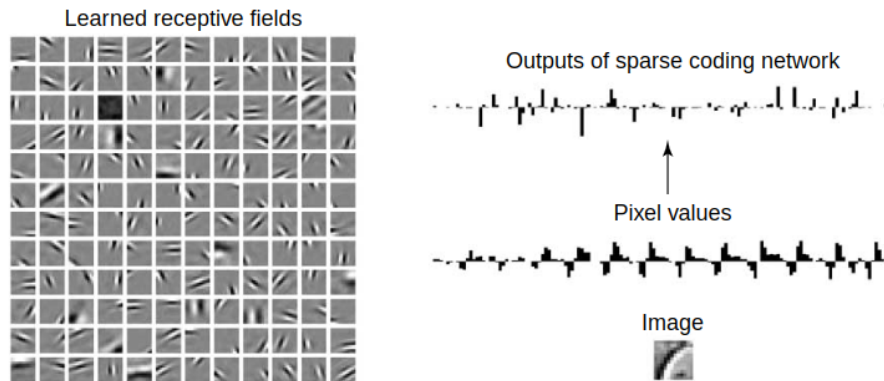


FIGURE 2.10 – Codage parcimonieux, figure provenant de (Bruno A OLSHAUSEN et David J FIELD 2004)

D'autres méthodes de codages existent comme le codage en rafale (PARK, S. KIM, CHOE et al. 2019) ou le codage de phase (KAYSER et al. 2009). Chaque méthode a ses avantages et inconvénients et peut répondre à différents besoins. Comme mis en avant par (GUO et al. 2021 ; PARK, S. KIM, NA et al. 2020), le codage par latence, expliqué plus haut, permet de réduire considérablement le nombre de spikes émis et ainsi la consommation de ressources et le nombre d'opérations nécessaires pour simuler le SNN tout en garantissant de bonnes performances.

## 2.3 Apprentissage

Comme expliqué dans la section 2.2.2, Donald Hebb fut le premier (sinon l'un des premiers) à proposer une règle d'apprentissage permettant ainsi au réseaux de neurones d'apprendre grâce à leurs expériences. Cette méthode relativement simple consistait à augmenter les connexions entre les neurones qui déchargent ensemble. Cette règle était inspirée du fonctionnement de la plasticité synaptique des neurones. Par la suite, une nouvelle règle fut proposée, la Spike-Timing-Dependent-Plasticity (STDP). Cette règle est basée sur des relevés d'étude électrophysiologique, montrant que cette règle d'apprentissage est divisée en deux cas spécifiques (BI et POO 2001 ; CAPORALE et DAN 2008) :

- Potentialisation à long terme (LTP) qui augmente les connexions avec les neurones pré-synaptiques ayant déchargés peu de temps avant l'instant de décharge du neurone post-synaptique
- Dépressions à long terme (LTD) qui réduit les connexions avec les neurones pré-synaptiques qui déchargent peu de temps après le neurone post-synaptique

La temporalité exacte des spikes pré et post-synaptique impacte fortement la modulation des poids synaptiques (voir figure 2.11). En effet, l'amplitude de la modification synaptique décroît de façon exponentielle avec la différence temporelle entre les spikes des neurones pré et post-synaptiques. Cela permet ainsi d'augmenter significativement les connexions avec les neurones ayant déchargé durant une courte fenêtre temporelle auparavant ( $\tau_{LTP}$ ) et réduire ceux ayant

déchargé peu de temps après ( $\tau_{LTD}$ ), avec  $\tau_{LTP} > \tau_{LTD}$ . La modification des poids, comme proposé par (MASQUELIER 2012), peut s'exprimer comme ceci (2.5) :

$$\Delta W_{i,j} = a_{LTP} \times \exp \frac{t_j - t_i}{\tau_{LTP}} \quad \text{si } t_j \leq t_i$$

$$\Delta W_{i,j} = -a_{LTD} \times \exp -\frac{t_j - t_i}{\tau_{LTD}} \quad \text{si } t_j > t_i$$
(2.5)

Avec :

- $\Delta W_{i,j}$  : variation apportée à la connexion synaptique entre le neurone post-synaptique  $j$  et le neurone pré-synaptique  $i$
- $a_{LTP}$  et  $a_{LTD}$  : intensité de la modification synaptique
- $t_i$  et  $t_j$  : instant du dernier spikes des neurones  $i$  et  $j$
- $\tau_{LTP}$  et  $\tau_{LTD}$  : fenêtre temporelle du potentialisation à long terme (LTP) et du dépressions à long terme (LTD)

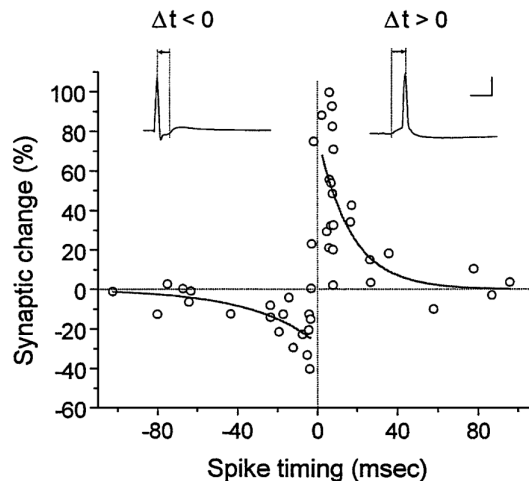


FIGURE 2.11 – Règle d'apprentissage STDP observée par (BI et POO 2001)

La modélisation de la règle STDP proposée par l'équation (2.5), ne permet pas de prendre en compte l'intégralité des spikes émis, mais seulement une paire de spike, c'est à dire, le dernier spike émis par le neurone pré et post-synaptique est pris en compte. Des variantes de cette règle STDP ont été proposées pour répondre à certains comportements étudiés au niveaux des modifications synaptiques des neurones ou selon le type de connexion (FF ou *Feed-back* (FB)). C'est le cas par exemple de la règle anti-STDP qui est une version inversée de la règle STDP, la triplet-STDP (PFISTER et GERSTNER 2006) qui prend en compte un triplet de spike (deux spike pré et un spike post) ou encore la m-STDP (BURBANK 2015) qui propose de varier la règle STDP selon le type de connexion. Une version simplifiée de la règle STDP a aussi été proposée dans laquelle l'amplitude du LTP et LTD est fixe et la fenêtre temporelle du LTD est supprimée, toutes les connexions ne recevant pas le LTP reçoivent alors le LTD (BICHLER et al. 2012) (figure : 2.12). Chacune des règles présente ses spécificités (VIGNERON et MARTINET 2020) et diffèrent aussi selon leur simplicité de calcul et de paramétrisation.

Comme montré par (GUYONNEAU, VANRULLEN et S. J. THORPE 2005 ; MASQUELIER, GUYONNEAU et S. J. THORPE 2008) la règle d'apprentissage STDP permet aux neurones de devenir sélectifs au début du stimulus présenté et donc de réduire de la latence des spikes émis par les neurones post-synaptiques permettant ainsi de justifier la rapidité de traitement de l'information. Une des caractéristiques de la règle STDP est qu'elle est non-supervisée, elle permet

aux neurones d'apprendre leurs propre expériences sans leur préciser quelle est la réponse attendue, cela permet ainsi à cette règle de permettre un apprentissage *online*, en même temps que le signal d'entrée (la scène visuel) est acquis (BAUMGARTNER et al. 2020 ; THIELE, BICHLER et DUPRET 2018).

Des règles d'apprentissage supervisées ou semi-supervisées ont été proposé en alternative à la règle STDP afin d'améliorer les performances des SNNs. (M. MOZAFARI et al. 2018) propose une version semi-supervisée de la STDP qui s'inspire de l'apprentissage par renforcement, la RSTDP. Une classe est attribuée à chaque neurone et selon que celui-ci décharge pour la bonne classe, une règle STDP classique est appliqué, sinon une anti-STDP lorsque le neurone décharge pour la mauvaise classe. Cette règle permet de conserver les caractéristique de l'apprentissage STDP (apprentissage local, dépendance de la temporalité des spikes, etc) tout en permettant d'améliorer les performances de la règle STDP (M. MOZAFARI et al. 2018).

Concernant les ANNs, une règle d'apprentissage supervisée est devenue prévalente depuis quelques temps, il s'agit de l'apprentissage par back-propagation (BISHOP 1995) (expliquer). Depuis récemment des versions adaptées aux SNNs ont été proposées (S. M. BOHTE, KOK et LA POUTRÉ 2002 ; Yujie WU et al. 2018 ; S. R. KHERADPISHEH et MASQUELIER 2020) permettant ainsi de s'approcher encore un peu plus des performances des ANN. Cependant, la possibilité qu'une telle méthode soit utilisée par le cerveau semble très faible, car elle nécessite de dériver des spikes, qui peuvent être assimilés à des fonctions de Dirac et donc non dérivables, et les poids des connexions FF sont mis-à-jour de façon top-down (voir (TAVANAIEI, GHODRATI et al. 2019) pour plus de détails). D'autres méthodes d'apprentissage supervisées sont aussi proposées comme *spike pattern association neuron* (SPAN) (MOHEMMED et al. 2012), le Chronotron (FLORIAN 2012), ou la méthode *Remote Supervised Learning* (ReSuMe) (PONULAK et KASIŃSKI 2010).

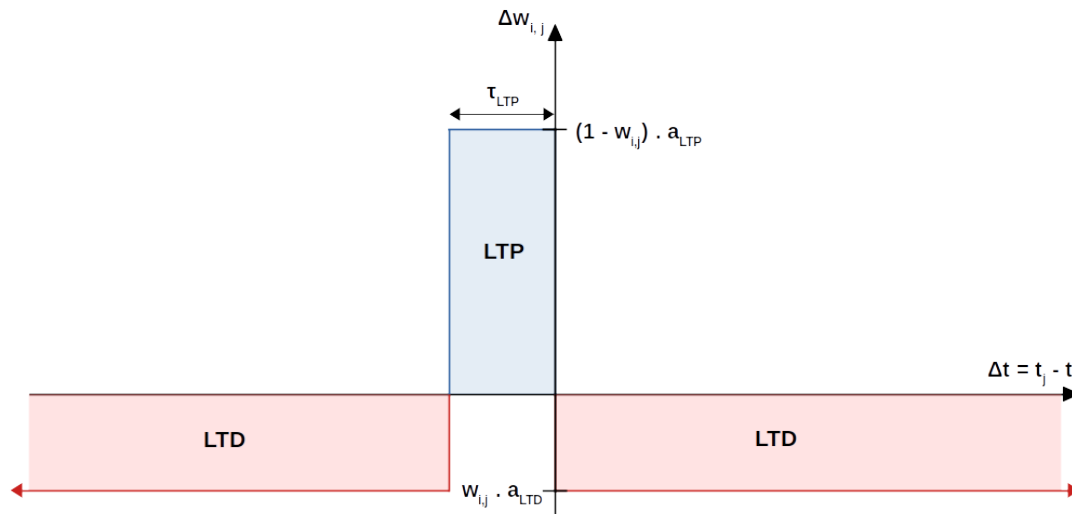


FIGURE 2.12 – Règle d'apprentissage STDP inspirée par (BICHLER et al. 2012)

---

## 2.4 Extraction d'informations spatio-temporelles

Un mouvement peut être caractérisé par différentes caractéristiques (forme, orientation, direction, vitesse, ...). De nombreuses études ont mis en avant le pertinence d'utiliser une SNN avec une règle d'apprentissage de type STDP afin d'avoir des neurones qui deviennent sélectifs à des contours orientés, puis à des formes en avançant dans la hiérarchie du SNN (MASQUELIER et S. J. THORPE 2007). Ces informations relèvent principalement de caractéristiques spatiales du stimulus, une sélectivité à l'orientation peut être utilisée pour discriminer un mouvement latéral d'un mouvement horizontal. Toutefois, elle ne permet pas de discriminer deux directions opposées (droite/gauche). Cette information se repose sur des caractéristiques spatiales et temporelles. En effet, il faut pour le neurone savoir l'ordre des spikes qu'il reçoit. Pour un mouvement qui va de la droite vers la gauche, le neurone est excité en premier par des neurones situés à gauche de son champ récepteur, suivi par ceux situés à droite (et inversement pour un mouvement allant de la droite vers la gauche).

### 2.4.1 Forme

Les ANNs présentent de très bonnes performances dans des tâches de détection de formes et catégorisations d'images. L'arrivée notamment des CNNs, de l'apprentissage en profondeur (KRIZHEVSKY, SUTSKEVER et HINTON 2017; RAWAT et Z. WANG 2017), de nouvelles méthodes d'apprentissage et de méthodes permettant d'éviter le sur-entraînement ont permis aux ANN d'apprendre des patterns visuels, de les généraliser et de pouvoir les réutiliser pour reconnaître des formes spécifiques. Le développement des jeux de données (MNIST, CIFAR-10, etc) ont offert la possibilité de d'évaluer et de comparer différents types d'ANNs et de pouvoir comparer leurs performances sur des jeux de données similaires. Le deep learning a permis d'avoir un apprentissage hiérarchique, avec les filtres des premières couches qui apprennent et détectent des informations très locales (bordure, orientations), puis ces informations locales se cumulent au fur et à mesure que l'on avance dans les couches pour obtenir des filtres qui détectent des patterns globaux et plus spécifiques (voitures, visage, oeil, arbre, chiffre, etc... ). Les SNNs se sont donc eux aussi inspirés des méthodes utilisés par les ANNs et ont ré-utilisés les jeux de données présentés ci-dessus afin de pouvoir comparer leurs résultats avec les ANNs et les autres SNNs. Les ANNs restent généralement meilleurs que les SNNs sur ces types de tâches (voir tableau 2.1).

Les études de catégorisation d'images utilisant des SNNs ont dans un premier temps majoritairement utilisé la règle d'apprentissage STDP dont notamment dans l'étude de (MASQUELIER et S. J. THORPE 2007) qui est une des premières études démontrant l'efficacité d'un SNN multicouche pour la catégorisation d'images, en utilisant le jeu d'images de *Caltech*. De nombreuses études ont ensuite validé ces résultats en utilisant des méthodes similaires avec des jeux de données différents (S.-R. KHERADPISHEH et al. 2018; C. LEE et al. 2018; DIEHL et COOK 2015; THIELE, BICHLER et DUPRET 2018). Des alternatives s'inspirant de la STDP ont été ensuite utilisées afin d'utiliser des méthodes d'apprentissage semi-supervisées, avec la RSTDP (M. MOZAFARI et al. 2018) et supervisées (TAVANAIEI et A. MAIDA 2019; HAO et al. 2020), afin d'améliorer les résultats. Finalement dans une quête de performance et de volonté de mimer et réduire l'écart avec les ANNs, une méthode d'apprentissage qui mime le fonctionnement de la back-propagation adaptée aux SNNs a été proposé et des méthodes de conversion d'ANNs en SNNs.

Afin de s'approcher de situations réelles, une alternative à la base de données MNIST a été proposée par (Garrick ORCHARD et al. 2015; IYER, CHUA et Haizhou LI 2021), le N-MNIST,

Model	Architecture	Learning method	Dataset	Accuracy
ESSER et al., 2015	Deep SNN	Offline learning, Conversion	MNIST	99.42
J. H. LEE et al., 2016	CSNN	Backpropagation	MNIST	99.31
DIEHL, NEIL et al., 2015	CSNN	Offline learning, Conversion	MNIST	99.10
T. LIU et al., 2018	SNN	Temporal backpropagation (3-layer)	MNIST	99.10
S.-R. KHERADPISHEH et al., 2018	CSNN	Layer wise STDP	MNIST	98.40
TAVANA EI et al., 2019	CSNN	Layer-wise and end-to-end STDP rules	MNIST	98.60
QUERLIOZ et al., 2013	SNN	STDP, Hardware implementation	MNIST	93.50
J. H. LEE et al., 2016	Deep SNN	Backpropagation	MNIST	98.88
J. H. LEE et al., 2016	Deep SNN	Backpropagation	N-MNIST	98.74
J. H. LEE et al., 2016	CSNN	Backpropagation	N-MNIST	98.30
RUECKAUER et al., 2017	CSNN	Offline learning, Conversion	CIFAR-10	90.85

TABLE 2.1 – Résumé des récents modèles de SNNs et leur précision sur différents jeux de données. Résultats tirés de (TAVANA EI, GHODRATI et al. 2019).

qui utilise une DVS filmant un écran qui affiche les différentes images de MNIST. Des saccades oculaires sont ensuite mimées afin de générer un mouvement de la DVS et donc des spikes. Une méthode similaire a été utilisée sur la base de données CIFAR-10 (Hongmin LI et al. 2017). (MORAITIS, SEBASTIAN et ELEFThERIOU 2020) a utilisé une version alternative de la base de données MNIST. Une occlusion spatiale qui évolue avec le temps est appliquée aux chiffres présentés. Ainsi, (MORAITIS, SEBASTIAN et ELEFThERIOU 2020) a montré que sur ce type de tâche un SNN pouvait surpasser les ANNs. De plus, une base de données de reconnaissance de cartes présentées extrêmement rapidement devant une caméra DVS (52 cartes en 0.65 secondes) est aussi utilisée pour tester les SNNs (Teresa SERRANO-GOTARREDONA et B. LINARES-BARRANCO 2015). Ce type de stimuli permet de tester la capacité de reconnaissance de formes d'un SNN sur des mouvements rapides, mettant ainsi en avant les avantages de ce type d'architectures.

## 2.4.2 Profondeur

Il existe différentes méthodes utilisées par l'humain pour obtenir des informations de profondeur, comme la parallaxe, l'occlusion, la disparité binoculaire, la taille apparente de l'objet, l'accommodation, etc... Différentes méthodes ont été utilisées pour mimer ces processus et décoder l'information de profondeur avec des caméras événementielles (STEFFEN, REICHARD et al. 2019). Une des méthodes la plus utilisée est la disparité binoculaire (OHZAWA, DEANGELIS et FREEMAN 1990). Nos yeux sont alignés verticalement et approximativement espacés d'environ 6 cm chez l'adulte sur l'horizontal. Chaque œil reçoit donc une information différente et légèrement décalée par rapport à l'autre œil. Un même point sur une scène n'aura pas la même position selon qu'il soit vu par l'œil gauche ou droit, ce décalage peut alors être utilisé pour estimer la profondeur de l'objet. Plus ce décalage sera important, plus l'objet sera proche de la personne qui perçoit la scène.

Cette méthode implique donc que deux points ou patterns similaires sur les deux images perçues soient identifiés afin de pouvoir calculer la disparité binoculaire. Ce problème de détection de patterns similaires est appelé le *problème de correspondance* (STEFFEN, REICHARD et al. 2019; OSSWALD et al. 2017; DIKOV et al. 2017). Une détection de deux patterns qui n'appartiennent pas en réalité aux mêmes points induirait ainsi une erreur de perception de la profondeur car la disparité serait biaisée. L'information temporelle portée par des spikes permet de répondre à ce problème, un point précis d'une scène déclenchera des spikes synchronisés par

les deux caméras, qui peuvent être facilement décodés avec des neurones LIF. Cela permet ainsi de pouvoir ensuite calculer la disparité à partir de ces détecteurs de coïncidences (OSSWALD et al. 2017 ; DIKOV et al. 2017).

(CHAUHAN et al. 2018) montre dans une étude que des neurones d'un SNN qui apprennent avec une règle d'apprentissage STDP et auxquels on présente des images binoculaires provenant de scènes naturelles, deviennent sélectifs à des contours avec des disparités différentes. Ces neurones ont des champs récepteurs proches de ceux que l'on trouve dans V1 (Gabor orientés, voir figure 2.13, panneau A). Leurs réponses et leurs répartitions sont proches de ceux que l'on trouve dans le cortex visuel primaire (voir figure 2.13, panneau B et C). Les neurones à qui on présente des portions situées dans la moitié basse de l'image ont plus de chances d'être sélectifs à des disparités croisées (positives) alors que ceux à qui on présente des portions situées dans la moitié haute de l'image sont sélectifs à des disparités décroisées (négatives).

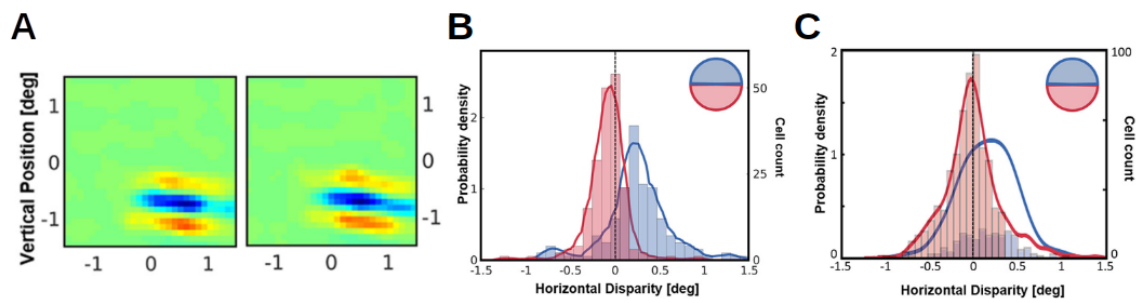


FIGURE 2.13 – Figure provenant de (CHAUHAN et al. 2018). Panneau A : Champs récepteur d'un neurone entraîné avec un stimulus binoculaire. Panneau B : histogramme des disparités horizontales de 300 neurones lorsqu'ils sont entraînés avec la partie supérieure (bleu) ou inférieure (rouge) du champs visuel. Panneau C : Similaire au panneau B, mais utilisant des neurones V1 de singes. Résultats provenant de (SPRAGUE et al. 2015).

Des études ont aussi utilisé des informations monoculaires pour décoder l'information de profondeur. Par exemple, (YANG et al. 2006) l'évalue lors d'un mouvement. Pour cela, il utilise des neurones radialement organisés ainsi que la règle d'apprentissage STDP pour détecter un stimulus passant par deux neurones placés sur le même axe radial et ensuite utilise l'intervalle de temps séparant la décharge de ces deux neurones pour décoder l'information de profondeur. Une méthode différente a été appliquée dans (HAESSIG, BERTHELON et al. 2019) qui s'inspire de la méthode d'estimation de la profondeur par defocus, utilisée par l'araignée (NOLTE et al. 2017) ainsi que par l'humain (V. A. NGUYEN, HOWARD et ALLISON 2005 ; MATHER et SMITH 2002 ; MATHER 1996), qui consiste à ajuster le focus fait par la rétine. En effet, afin de percevoir nettement un stimulus à une profondeur donnée, un focus spécifique est nécessaire. Lorsque l'on modifie la distance focale de la caméra, le stimulus apparaît dans un premier temps hors du focus (flou et de taille élevé), en focus (net et de taille minimale), puis revient hors de focus. Cette variation de la distance focale va ainsi générer des variations de luminosités et donc générer des spikes, qui permettront de déterminer la distance focale pour avoir le stimulus en focus et ainsi sa profondeur.



---

### 2.4.3 Mouvement

Comme vu précédemment, de nombreuses études utilisent des images afin d'évaluer les performances d'un SNN. Cette méthode permet souvent de profiter de bases de données d'images, bien souvent développées pour tester des ANN et pouvoir ainsi se comparer à eux. Les CNN ont permis de fortes avancées dans le traitement d'images et la détection et reconnaissance de formes. Il est donc normal que de nombreuses architectures de SNN s'inspirent de ces précédentes études et utilisent les mêmes types de stimuli. Cependant, à la différence de la plupart des ANN, les SNN possèdent une dimension temporelle importante. Les stimuli de types images étant statiques, ils ne possèdent pas de dimensions temporelles. Il est donc nécessaire d'en inférer une à partir de la valeur des pixels et en utilisant un des codages présentés section 2.2.6. De nombreuses études utilisent ainsi les bases de données (MNIST, etc...) pour tester leur SNN. Cependant, l'utilisation d'un stimulus avec une dimension temporelle permet de profiter pleinement des avantages d'un SNN, à savoir l'utilisation de données plus parcimonieuses avec des spikes lors de variations de luminosités, le traitement asynchrone et *event-based* de l'information et donc la réduction de la consommation d'énergie et le traitement en temps réel.

Des alternatives à MNIST (Garrick ORCHARD et al. 2015) sont maintenant disponibles en proposant une base de données similaire, mais filmée avec une caméra DVS qui mime les microsaccades de l'oeil et permet ainsi de s'approcher d'une situation plus réelle. Moraitis surpasse même les performances des ANNs dans (MORAITIS, SEBASTIAN et ELEFTHERIOU 2020), dans une expérience utilisant la base de données MNIST, mais avec une occlusion spatiale du stimulus qui varie avec le temps. Cette étude profite des caractéristiques temporelles d'un SNN et d'une règle bio-inspirée qui prend elle aussi en compte cette dimension et lui permet ainsi de dépasser l'état de l'art sur ce type de stimuli, qui utilise des images MNIST mais avec une dimension temporelle, à savoir l'occlusion.

Le mouvement se définit par un changement de position dans l'espace. Par défaut, un SNN qui prend en entrée le flux d'événements d'une caméra événementielle aura des neurones qui répondront aux mouvements. En effet, les événements générés par une caméra événementielle sont principalement générés par du mouvement. Excepté dans le cas d'une forte variation de luminosité, comme par exemple une lumière que l'on éteint et rallume par exemple. Même la méthode de (HAESSIG, BERTHELON et al. 2019), qui filme un stimulus immobile et une caméra immobile, induit un mouvement par le defocus. Ainsi les événements générés par la caméra événementielle proviendront principalement du mouvement (excepté du bruit ou des variations de luminosité rapides), les neurones ne pourront alors répondre principalement qu'à du mouvement

Un SNN de type *feed-forward* (FF) (avec des inhibitions latérales instantanées) permet de discriminer des orientations ou des profondeurs. Il ne pourra cependant pas discriminer des directions (balle qui se dirige vers la gauche/droite, s'approche ou recule). (BICHLER et al. 2012) propose un SNN de type FF auquel il présente des balles simulées qui vont dans différentes directions et montre qu'il obtient des neurones qui deviennent sélectifs à ces directions. En effet, lorsque une balle sombre se dirige de la gauche vers la droite avec un arrière plan clair, les spikes de types ON seront positionnés sur la partie gauche de la balle et les spikes OFF sur la partie droite (et inversement pour une balle allant dans la direction opposée). Ainsi, lors de cette étude ils utilisent la position spatiale des spikes ON/OFF pour déterminer la direction du stimulus. Cette méthode possède cependant deux inconvénients majeurs :

- Le champs récepteur du neurone doit être suffisamment grand pour voir toute (ou presque) la balle. Une information locale (la partie ON des spikes, par exemple) ne devrait pas suffire pour extraire la direction de la balle.
- Si on inverse le contraste de la balle et de l'arrière plan (balle claire et arrière plan sombre), les neurones déchargeront pour les directions opposées, car les spikes ON/OFF seront diamétralement opposés par rapport à la balle sombre et fond clair.

Afin de pouvoir déterminer la direction du mouvement indépendamment du contraste et en utilisant des informations locales, il est nécessaire d'ajouter de nouvelles caractéristiques au SNN afin d'améliorer sa sélectivité temporelle.

### 2.4.3.1 Délais

L'ajout de délais de transmission synaptiques entre les neurones permet aux neurones de devenir sélectifs à la temporalité des spikes qu'ils reçoivent en entrée. Ces délais vont ajouter un retard entre l'instant auquel le spike est émis par le neurone pré-synaptique et l'instant de réception de celui-ci par le neurone post-synaptique. Ces délais vont permettre de synchroniser (HÜNING, GLÜNDER et PALM 1998) les spikes reçus (en retardant les premiers spikes transmis afin qu'ils arrivent en même temps que ceux qui arrivent ultérieurement) lorsqu'un certain stimulus lui est présenté (direction allant vers la gauche) et les désynchroniser lors d'un stimulus ayant des "caractéristiques temporelles opposées" (vers la droite). La "spécificité" de fuite du potentiel de membrane d'un neurone va lui permettre de décharger lorsque les spikes sont reçus dans un court espace de temps et donc synchronisés (cf. figure 2.15). Il existe deux différentes méthodes afin de régler ces délais de transmission (C. W. EURICH et al. 2000; HÜNING, GLÜNDER et PALM 1998) (cf. figure 2.15).

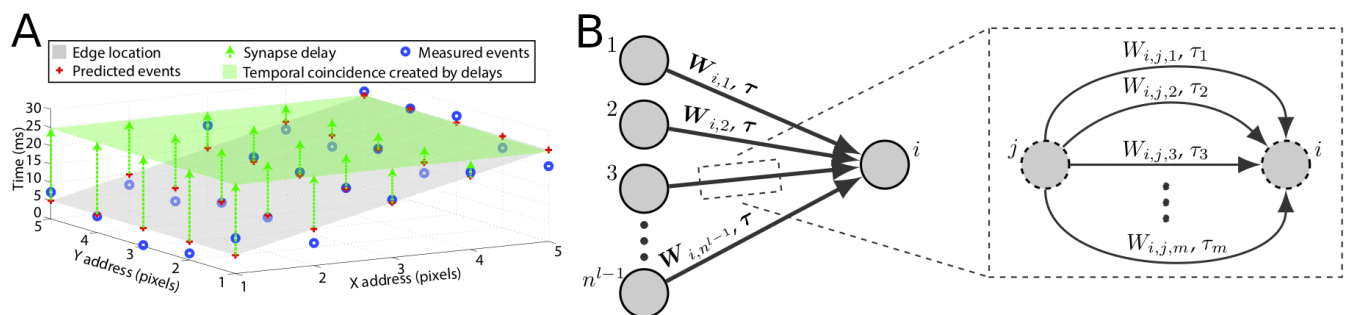


FIGURE 2.14 – Panel A : Délais fixes de (G. ORCHARD et al. 2013). Panel B : Multi-synapses de (PAREDES-VALLES, SCHEPER et CROON 2019)

La première, la *Delay shift* (figures à droite de 2.15) utilise des règles d'apprentissage afin de régler les délais de transmission. La règle d'apprentissage cherchera à minimiser le temps entre les spikes reçus et le spike émis. (C. EURICH et al. 1999; C. W. EURICH et al. 2000; TVERSKY et MIKKULAINEN 2002) utilisent une règle d'apprentissage hebbienne similaire à la STDP pour régler les délais. Les neurones ayant déchargés juste avant le spike du neurone post-synaptique verront alors leurs délais augmenter et ceux ayant déchargé peu de temps après verront leurs délais diminuer. (GIBSON, HENDERSON et WILES 2014) règle ses délais ainsi que d'autres paramètres en utilisant la divergence de Kullback–Leibler.

La seconde, *Delay selection* (figures centrales de 2.15) définit plusieurs synapses entre un même neurone pré-synaptique et un neurone post-synaptique. Chaque synapse aura un délai de transmission différent (Panneau B, figure 2.14). Lorsque un neurone pré-synaptique déchargera, son spike sera alors transmis plusieurs fois (un par synapse) et sera reçu à des temporalités différentes par le neurone post-synaptique. La règle de plasticité synaptique (STDP ou autre) sélectionnera ensuite les synapses avec les délais permettant de synchroniser les spikes reçus (PAREDES-VALLES, SCHEPER et CROON 2019; OUDJAIL et MARTINET 2021a).

(G. ORCHARD et al. 2013) utilise des délais fixes pré-définis (Panneau A, figure 2.14) et arrive à déterminer la vitesse et la direction du mouvement, tout en utilisant peu de neurones.

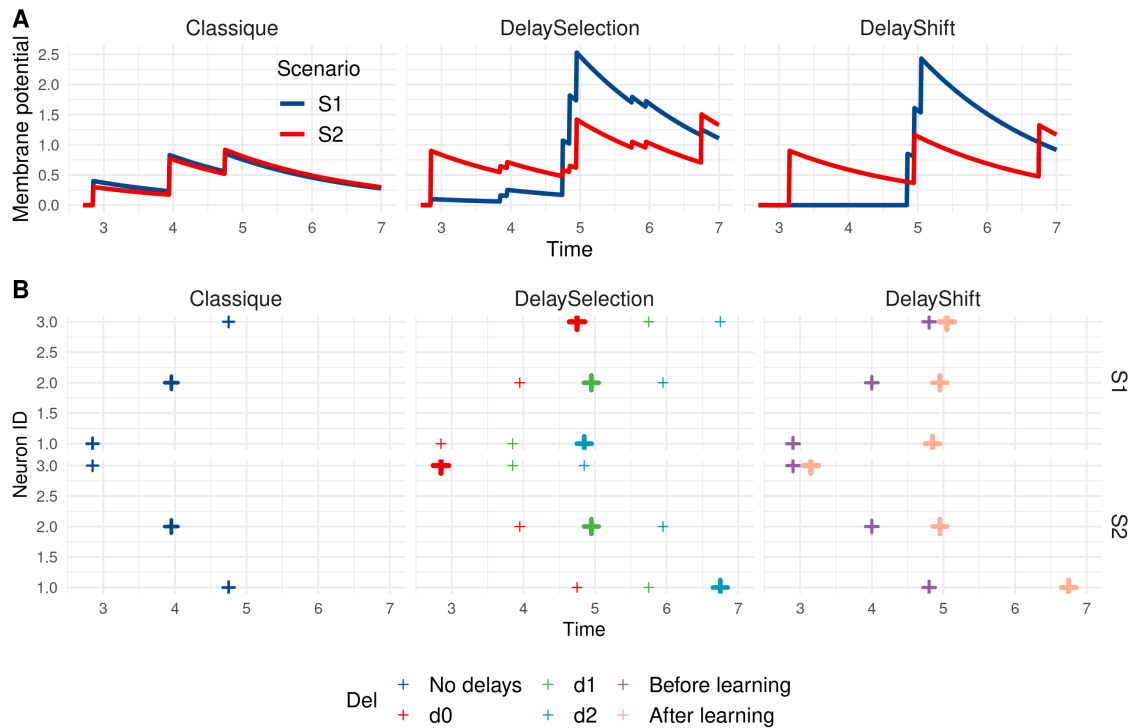


FIGURE 2.15 – Différentes méthodes de délais. Le panel A représente l'évolution du potentiel de membrane d'un neurone au cours du temps pour deux scénarios avec des conditions temporelles opposées (Scénario 1 : Neurone 1  $\rightarrow$  2  $\rightarrow$  3; Scénario 2 : Neurone 3  $\rightarrow$  2  $\rightarrow$  1, similaire à deux directions opposées). Le panel B représente les spikes reçus par le neurone lors de ces deux scénarios. L'épaisseur de la croix représente "l'intensité" de la connexion. Les deux figures à gauche représentent le cas sans utilisation de délais. Les deux figures centrales représentent le cas de sélection des délais. Les synapses sont dupliquées, les croix rouge, verte, bleue représentent le même spike transmis par des synapses différentes avec des délais qui augmentent. Les deux figures de droite représentent le cas de *delay-shift*. Les croix violettes représentent l'instant auquel le neurone post-synaptique décharge. Les croix beige les instants (après délais) auxquelles elles sont reçues par les neurones de la couche supérieure.

Ces délais peuvent être utilisés avec différentes architectures et sur des neurones excitateurs ou inhibiteurs. Une première méthode et la plus communément utilisée est d'appliquer des délais sur les connexions excitatrices FF (PAREDES-VALLES, SCHEPER et CROON 2019; OUDJAIL et MARTINET 2021a; HÜNING, GLÜNDER et PALM 1998). Pour cela, comme vu au-dessus les délais sont soit appris (*delay shift*) soit les synapses sont dupliquées et différents délais sont utilisés, puis sélectionnés par la règles d'apprentissage (*delay selection*, figures centrales de 2.15). Une autre architecture proposée par (WENISCH, NOLL et HEMMEN 2005; ADAMS et HARRIS 2014) utilise des connexions latérales excitatrices et inhibitrices avec des délais fixes et proportionnels à la distance entre les neurones pré et post-synaptiques. La règle STDP est ensuite utilisée pour sélectionner les synapses. Les connexions latérales avec des neurones proches topographiquement sont excitatrices et celles plus éloignées sont inhibitrices et permettent d'avoir des neurones regroupés selon leurs sélectivités (à la direction dans ce cas). Ces regroupements topographiques auto-organisés générés par les connexions latérales excitatrices/inhibitrices dépendantes de la distance avec les autres neurones, aussi appelées connexions de type chapeau mexicain, permettent ainsi de générer des cartes de sélectivité à la direction aussi présentes dans le cortex visuel (WELIKY, BOSKING et FITZPATRICK 1996; SHMUEL et GRINVALD 1996).

Les délais étant proportionnels à la distance entre les neurones pré et post-synaptiques, ils permettent ainsi d'augmenter la synchronie des spikes lors d'un mouvement. Les neurones éloignés du centre du champs récepteur du neurone post-synaptique déchargeront en premier mais avec un délais plus longs. Puis, au fur et à mesure que le stimulus s'approche du milieu du champs récepteur, les délais synaptiques se réduisent. Ainsi les spikes arrivent en quasi-synchronie lorsque le stimulus arrive dans le centre du champs récepteur du neurone post-synaptique, voir figure 2.16. La règle STDP sélectionne ensuite les synapses, qui permettront la sélectivité à la direction. Ainsi, dans le cas de la figure 2.16, pour un stimulus opposé ( $6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1$ ), les spikes des neurones 6 à 4 arriveraient en synchronie, mais leurs poids seraient faibles et n'engendreraient pas de décharge du neurone post-synaptique. Puis les spikes des neurones 3 à 1 auraient un impact fort sur le potentiel de membrane car leurs poids sont forts, mais ils seraient désynchronisés et n'engendreraient pas de spikes non plus.

Ce type d'architecture est intéressant car elle permet d'avoir une sélectivité à la direction, elle est bio-inspirée et permet la génération de carte de sélectivité à la direction. Cependant, elle nécessite de gérer les poids des connexions FF et latérales. De plus, les délais étant fixes, ils sont donc optimaux pour une vitesse spécifique. Afin d'avoir des neurones sélectifs à des vitesses différentes, selon le panel de vitesse présent dans les stimuli, il faudrait donc choisir différentes valeurs de délais ou les paramétrer comme pour les connexions FF avec une méthode de *delay selection* ou de *delay shift*.

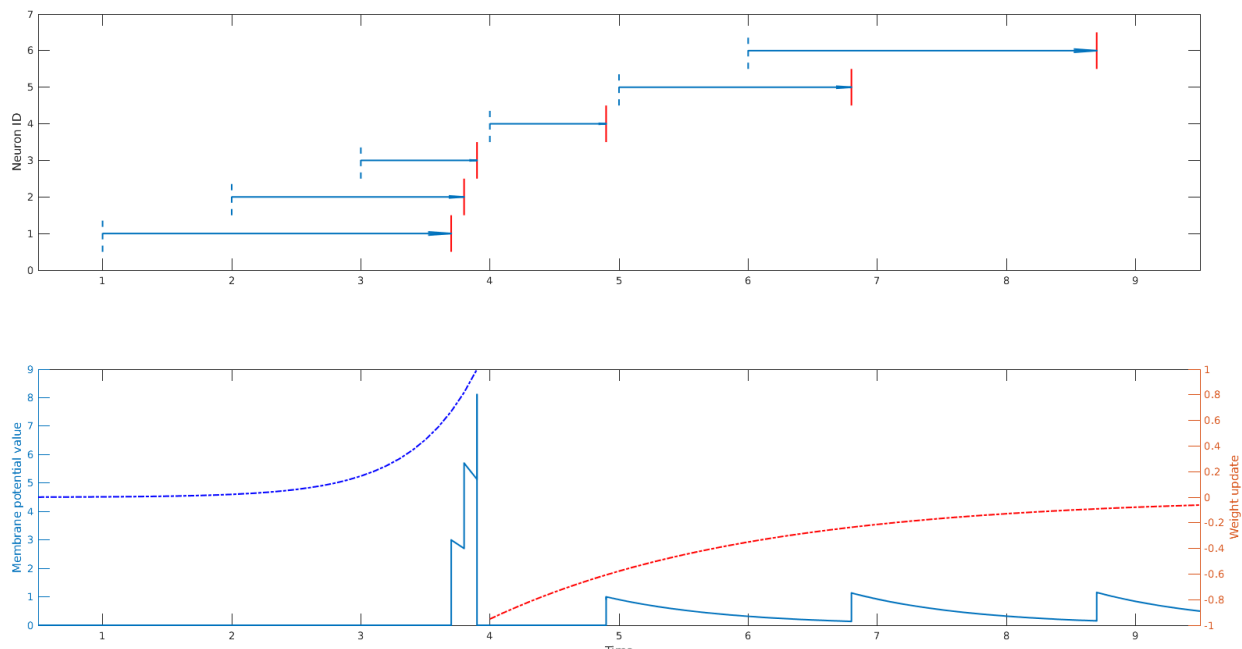


FIGURE 2.16 – Impacts des délais proportionnels à la distance entre les neurones pré-synaptiques et le neurone post-synaptique. Les neurones 1 et 6 sont éloignés du neurone post-synaptique, les neurones 2 et 5, sont à une distance moyenne et les neurones 3 et 4 en sont proches. Les délais de transmission synaptiques sont proportionnels à cette distance, comme illustré dans le panneau du haut. Lorsque le neurone post-synaptique décharge (panneau du bas), la règle STDP renforce les connexions des neurones synchronisés (1, 2 et 3) et diminue les autres (4, 5 et 6).

L'utilisation de connexions récurrentes inhibitrices nécessite cependant une synchronie assez précise, plus particulièrement lorsque peu de neurones déchargent et ne déchargent qu'une seule fois lors de la présentation du stimulus. Les spikes émis par les connexions récurrentes

doivent arriver suffisamment tard afin que le potentiel de membrane du neurone ait suffisamment commencé à augmenter pour que l'inhibition puisse impacter le potentiel de membrane suffisamment tôt, avant que le neurone post-synaptique ne décharge. Cette intervalle de temps est bien souvent faible, de l'ordre de quelques millisecondes et réduit par l'utilisation de délais. En effet, les délais augmentent la synchronie entre les spikes et réduisent ainsi le temps entre le début de l'excitation du neurone et l'instant où il décharge (figure 2.15. L'intervalle de temps entre le début et la fin de l'excitation est bien plus grand pour le cas "Classique" que pour les deux autres). Si le neurone a une fréquence de décharge proportionnelle au signal d'entrée (neurone Poisson par exemple, figure 2.17, panneau A), l'inhibition sera effective durant une plus longue fenêtre temporelle réduisant ainsi plus facilement la réponse de l'autre neurone. Si les neurones ne déchargent que peu de fois, voire une seule fois par présentation d'un stimulus (S. J. THORPE et al. 2004 ; S. R. KHERADPISHEH et MASQUELIER 2020), il est alors nécessaire d'avoir plusieurs neurones qui déchargent pour un stimulus très similaire, avec un décalage temporel entre les spikes des différents neurones. Les différents spikes du premier groupe de neurones (neurones 1 à 4, figure 2.17, panneau B) étant temporellement décalés car les neurones qui déchargent ont des champs récepteurs légèrement différents. Les spikes inhibiteurs reçus par le second groupe de neurones (5 à 8) couvrent une plage temporelle suffisamment grande pour inhiber les neurones. Il est donc nécessaire d'avoir de nombreux neurones qui codent pour le même stimulus. Si le neurone ne décharge qu'une seule fois et qu'il est le seul ou presque à décharger, afin de minimiser le nombre de spikes, il est donc plus difficile d'inhiber le second neurone. En effet, comme on le voit dans la figure 2.17, panneau C, une faible variation dans le délai de transmission peut faire que l'inhibition est reçue trop tôt ( $\Delta = 40 \text{ ms}$ ) ou trop tard ( $\Delta = 53/60 \text{ ms}$ ). Ainsi dans le cas d'un mouvement latéral, le neurone 1 sélectif à un mouvement allant de gauche de droite, ne pourrait inhiber le second neurone et donc l'empêcher de décharger que pour une plage de vitesse du stimulus faible. Une alternative à cela serait alors d'utiliser des inhibitions plus longues pour couvrir un large panel de vitesses (GIULIONI et al. 2016 ; HAESSIG, CASSIDY et al. 2018).

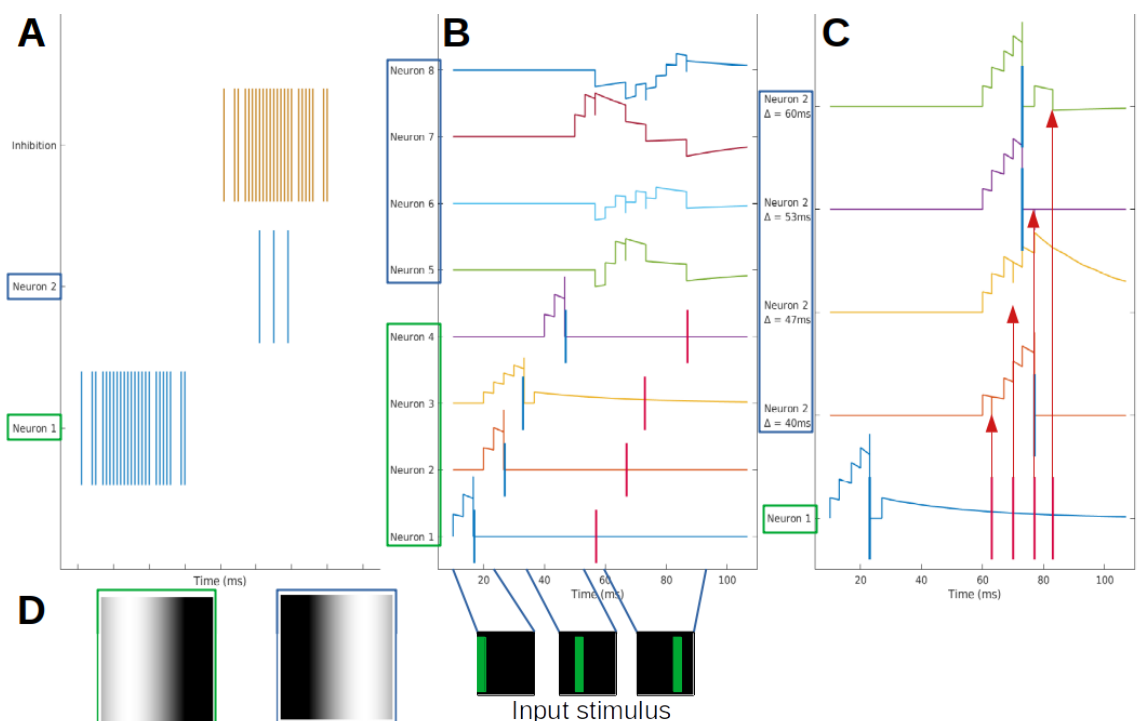


FIGURE 2.17 – Impact des différentes méthodes d'inhibitions avec délais.

## 2.5 Neuromorphic chip

Un des principaux inconvénients des ANNs est leur consommation d'énergie et leur capacité à traiter l'information rapidement. Les modèles se complexifiant et utilisant de plus en plus de couches, ils nécessitent ainsi de plus en plus d'opérations par inférences et d'énergie. Au contraire, comme montré par (TAVANA EI, GHODRATI et al. 2019; NEIL, PFEIFFER et S.-C. LIU 2016), et par la figure 2.18, les SNNs nécessitent beaucoup moins d'opérations qu'un ANN pour une même performance.

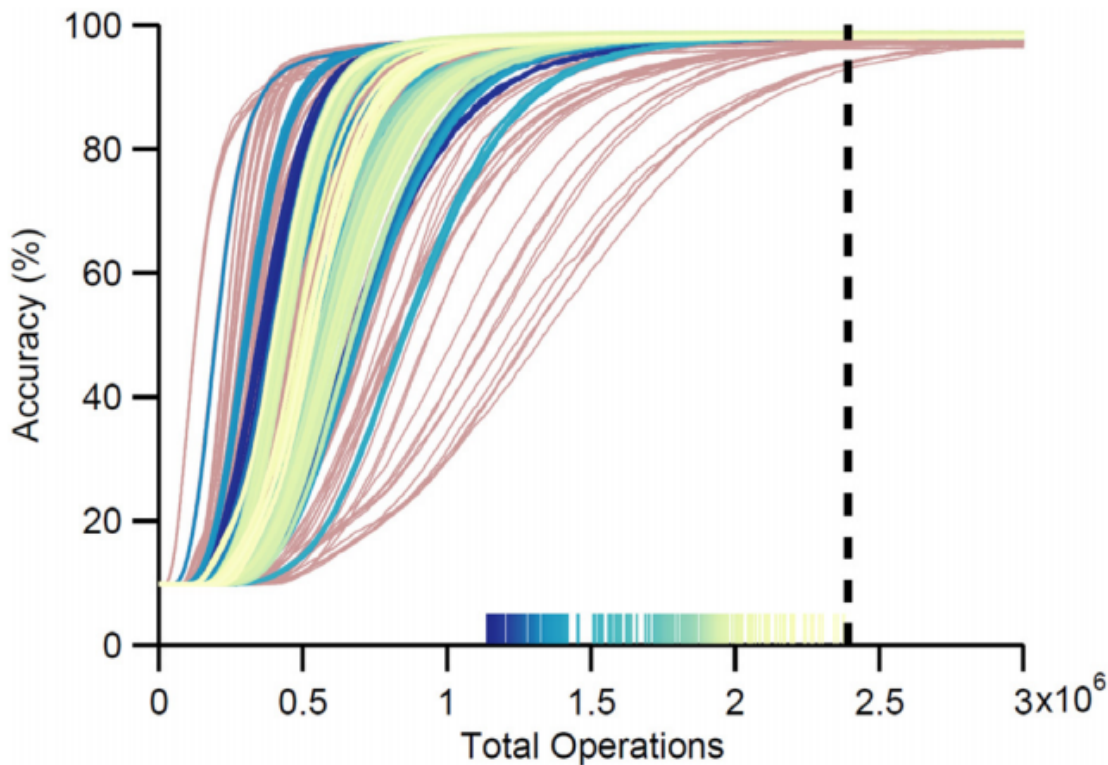


FIGURE 2.18 – Comparaison du nombre d'opérations nécessaire pour atteindre une performance de 98% de précision sur MNIST entre des SNNs et des ANNs. La ligne verticale en pointillés indique le nombre d'opérations nécessaires pour atteindre 98% pour un DNN (non-spikant). Les autres lignes montrent la précision de 522 SNNs en fonction du nombre d'opérations nécessaires. Les courbes roses représentent des SNNs ne parvenant pas à 98% de précision. Figure provenant de (TAVANA EI, GHODRATI et al. 2019).

L'arrivée des processeurs graphiques (*Graphics Processing Unit*) a permis aux ANNs et plus particulièrement aux DNNs de se développer en permettant de paralléliser leurs opérations et donc d'augmenter leur rapidité. Cependant ce type d'architecture reste consommatrice d'énergie. Un autre type d'architecture a vu le jour, les *neuromorphic chip* et les intégrations à très grande échelle (VLSI) (MEAD 1990), des composants analogiques qui imitent les systèmes neuronaux biologiques. Les ANNs utilisant des valeurs flottantes, ils sont plus difficilement implémentables sur ce type d'architecture, alors que les SNNs, utilisant des spikes comme moyen de transmission de l'information, semblent être des candidats naturels pour l'intégration sur ces architectures. Il existe cependant encore un écart entre les promesses faites par le *neuromorphique computing* et l'état de l'art actuel (PFEIFFER et PFEIL 2018).

---

Différentes puces neuromorphiques existent pour l'intégration de SNNs. Celles ci peuvent être divisées en deux catégories (D.-A. NGUYEN, TRAN et IACOPI 2021) : les accélérateurs neuromorphiques à grandes échelles, comme par exemple le SpiNNaker (FURBER et al. 2014), TrueNorth (MEROLLA et al. 2014; AKOPYAN et al. 2015), NeuroGrid (BENJAMIN et al. 2014), Loihi (DAVIES, SRINIVASA et al. 2018; DAVIES, WILD et al. 2021) et BrainScaleS (SCHEMMELE et al. 2010). Ces modèles ont pour objectif de pouvoir simuler un grand nombre de neurones, tout en permettant des latences et une consommation d'énergie faible (DIEHL, ZARRELLA et al. 2016). Ces différents modèles diffèrent en plusieurs points, leur architecture, le nombre de neurones et synapses qu'ils peuvent simuler, leur vitesse d'exécution, leur consommation d'énergie, la possibilité de faire l'apprentissage en ligne, etc. (PFEIFFER et PFEIL 2018). Les systèmes neuromorphiques embarqués, comme par exemple les architectures FPGA (MAGUIRE et al. 2007; KHODAMORADI, DENOLF et KASTNER 2021; A. LINARES-BARRANCO et al. 2019; ZAMARRENO-RAMOS et al. 2013), qui permettent d'optimiser la consommation d'énergie, mais avec plus de contraintes (nombre de neurones, synapses, etc.).

Différentes architectures de SNNs ont déjà été implémentées sur ce type de plateforme, dont notamment des SNNs récurrents (SHRESTHA et al. 2017; DIEHL, ZARRELLA et al. 2016), du traitement visuel avec des architectures convolutionnelles (T. SERRANO-GOTARREDONA et al. 2015; KHODAMORADI, DENOLF et KASTNER 2021; PADALA, BASU et Garrick ORCHARD 2018; ZAMARRENO-RAMOS et al. 2013), denses (HOPKINS et al. 2018). Bien que des solutions permettant de faire de l'apprentissage en ligne directement sur la carte neuromorphique sont proposées (BAUMGARTNER et al. 2020; THIELE, BICHLER et DUPRET 2018), de nombreuses études préfèrent entraîner un ANN en utilisant les méthodes classiques d'apprentissage, avant de le convertir en SNN pour pouvoir l'implémenter sur la carte neuromorphique (DIEHL, ZARRELLA et al. 2016; T. SERRANO-GOTARREDONA et al. 2015; MASSA et al. 2021).

## 2.6 Prédiction dans le domaine du sport

Dans le sport et plus particulièrement à haut niveau, une capacité majeure de l'expertise consiste à anticiper les événements (déplacements des joueurs, trajectoire de la balle ou du ballon, etc.), afin de pouvoir adapter ses décisions et comportement, et pouvoir prendre le dessus sur l'adversaire ou mieux collaborer avec ses coéquipiers. Comme présenté dans la section 2.1.2, l'information visuelle joue un rôle majeur dans la perception et cela n'échappe pas au domaine du sport. Les capacités de prédiction et d'anticipation des experts sportifs basés sur des informations visuelles sont examinées en présentant des vidéos de situations réelles sportives aux joueurs experts/non-experts. Ces vidéos sont occultées spatialement (une partie de l'image est cachée) et/ou temporellement (une partie de l'action est présentée, seulement les premières secondes par exemple). Les participants ont ensuite à donner leur prédiction pour indiquer la meilleure action à accomplir (CAUSER, SMEETON et A. M. WILLIAMS 2017). Ce type d'expérience a été appliquée dans différents sports, comme par exemple, le tennis (TRIOLET et al. 2013; BALSER et al. 2014), le football (CAUSER, SMEETON et A. M. WILLIAMS 2017), le volleyball (BALSER et al. 2014), le basket (AGLIOTI et al. 2008; Y. WU et al. 2013; Y. LI et FENG 2020).

À notre connaissance, il existe trois études qui ont cherché à évaluer les performances sur une tâche de prédiction de trajectoire de lancer franc de basket. La première, (AGLIOTI et al. 2008) a présenté des trajectoires coupées à différents instants après le début du lancer franc (moment où le joueur commence son mouvement de lancer). Cette expérience a été effectuée sur 3 différents groupes de participants, des joueurs novices (sans expérience), des observateurs experts (entraîneurs et journalistes sportifs) et des joueurs de basket professionnels. Il a été montré que les joueurs professionnels arrivaient à mieux prédire la trajectoire de la balle et plus tôt. (AGLIOTI et al. 2008) a aussi utilisé une méthode de Stimulation magnétique transcrâ-

---

nienne (TMS) pour mettre en avant que les joueurs professionnels ont une excitation des aires motrices lorsqu'ils devaient prédire la trajectoire de la balle. Cette excitation n'était pas présente lorsqu'ils regardaient un joueur de foot taper dans un ballon. Celle-ci est donc sélective aux mouvements longuement appris et pratiqués, permettant ainsi aux joueurs d'effectuer de meilleures prédictions dès le début de la vidéo, en utilisant les mouvements du lancer du joueur. Les prédictions des novices et observateurs experts quant à elles, reposaient principalement sur la trajectoire du ballon. De façon générale, les performances de prédictions des trois différents groupes s'amélioraient grandement au fur et à mesure de la présentation de la trajectoire. (Y. WU et al. 2013) a obtenu des résultats assez similaires avec une méthode proche.

Une autre étude (Y. LI et FENG 2020) a utilisé une méthode similaire pour évaluer les différences de performances entre différents niveaux d'expertises (experts, loisirs et novices). La conclusion générale que les résultats dépendent de l'expertise de la personne visionnant les vidéos est maintenue. En effet, les joueurs experts sont meilleurs que ceux qui pratiquent par loisir, qui sont meilleurs que les novices. Cependant, les résultats diffèrent avec l'étude (AGLIOTI et al. 2008) au niveau des taux de réussite. En effet, les personnes testées présentent des taux de réussite proches du niveau de chance (50% d'erreurs), sauf en fin de trajectoire (partie descendante de la trajectoire de balle) où les joueurs experts ont de meilleurs résultats que les joueurs loisirs, qui sont meilleurs que les novices. Les participants montraient aussi de meilleurs résultats pour les lancers réussis que pour les lancers ratés, avec une différence significative entre les performances des différents groupes pour les lancers réussis ( $\approx 70\%$  de réussite pour les athlètes,  $\approx 65\%$  pour les joueurs loisirs et  $\approx 60\%$  pour les novices).

Les résultats des deux premières études diffèrent assez fortement avec la troisième, notamment car les deux premières études montrent que les participants ont de bons résultats dès le début de la trajectoire. (Y. LI et FENG 2020) justifie cela en précisant que dans les deux premières études les participants ont la possibilité de répondre "Je ne sais pas" et beaucoup utilisent cette réponse en début de trajectoire, réponse qui n'est pas possible dans (Y. LI et FENG 2020). De plus (AGLIOTI et al. 2008) montre une différence significative entre les différents groupes testés en début de trajectoire (car les joueurs athlètes se basent sur le mouvement de lancer), alors que (Y. LI et FENG 2020) montrent des différences significatives entre les différents groupes en fin de trajectoire (phase descendante du ballon). (Y. LI et FENG 2020) justifient cela avec des différences probables dans le protocole de test et des différences individuelles entre les participants.

En analysant les vidéos présentées aux participants (AGLIOTI et al. 2008), on peut voir sur la figure 2.19 que les lancers de types réussis (en jaune) diffèrent très fortement des lancers ratés (en bleus). En effet, les lancers ratés tombent assez loin du panier et leurs trajectoires sont plus basses que les autres. En d'autres termes, il est possible que les lancers ratés aient été volontairement ratés dans le cadre de l'acquisition des vidéos, et non des tentatives infructueuses ou involontairement manquées, de lancers réels. Cette différence marquée entre les deux types de trajectoires peut ainsi expliquer les différences de résultats et les bons résultats de prédiction des participants même avec peu d'information.

Bien qu'il ait été montré que les personnes expertes étaient plus performantes pour prédire des situations proches de leurs domaines d'expertises (leur sport), il n'existe que peu de connaissances sur le processus permettant d'effectuer ces prédictions et sur leurs expertises. Toutefois, (BALSER et al. 2014) a montré par exemple, que le réseau d'action-observation (*action-observation network* (AON)) est activé durant les tâches de prédictions et cette activation est plus forte pour les joueurs experts du domaine. L'AON est une aire cérébrale sensorimotrice, l'apprentissage et l'expérience motrice semblent jouer alors un rôle important dans la prédiction et l'expertise. La perception étant un processus actif, les capteurs sensoriels n'attendent pas sagement de recevoir l'information. La prédiction visuelle semble donc être aussi un processus



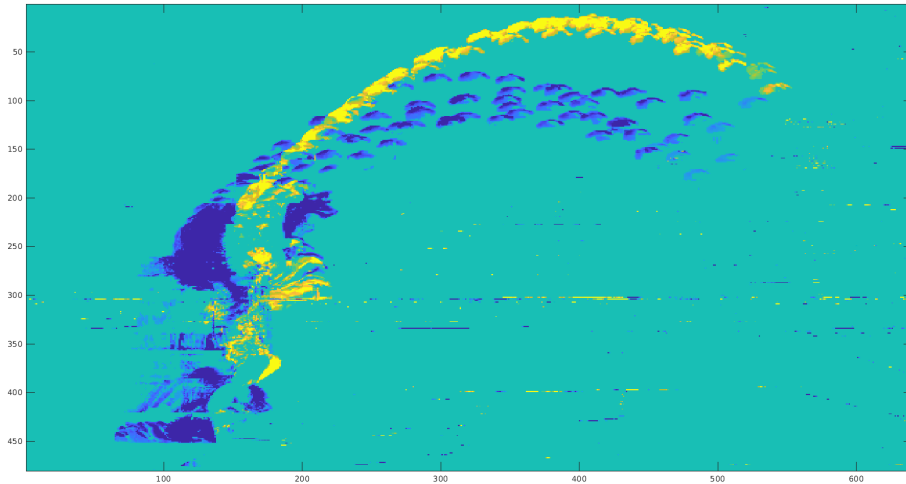


FIGURE 2.19 – Différences de trajectoires selon que le panier est réussi (jaune) ou raté (bleu)

actif, comme le montrent (DIAZ et al. 2013 ; SPERING et al. 2011). En effet, ces études mettent en évidence les caractéristiques anticipatoires des saccades et poursuites oculaires pour suivre le stimulus. Celles-ci permettent aussi de corriger les erreurs de prédictions.

## 2.7 Prédiction visuelle par des SNN

Prédire le futur à long terme a été étudié, à travers l'analyse dans un premier temps de séries temporelles (MAKRIDAKIS 1976), puis avec l'apparition des ANNs (OLAWOYIN et Y. CHEN 2018), les modèles se sont complexifiés et ont été appliqués sur différentes tâches notamment visuelles, comme la prédiction de mouvements humains (SUDA, MAKINO et SHINODA 2019 ; RUDENKO et al. 2020 ; MARTINEZ, BLACK et ROMERO 2017), de scènes naturelles (BABAEIZADEH et al. 2018 ; Z. LIU et al. 2017 ; A. X. LEE et al. 2018 ; FRAGKIADAKI et al. 2016), de trajectoires (MÜLLING, KOBER et PETERS 2010 ; NIKHIL et MORRIS 2019 ; GOMEZ-GONZALEZ et al. 2020 ; Xiaopeng CHEN et al. 2015), etc. L'analyse et la prédiction de trajectoires de balles en utilisant des systèmes artificiels est un domaine déjà développé. Il existe deux principales méthodes pour prédire une trajectoire de balle. La première méthode consiste à diviser le traitement en deux parties distinctes. Premièrement, l'extraction des caractéristiques spatio-temporelles de la balle (position, direction, vitesse, accélération, spin de la balle, ...). Puis l'envoi de ces informations dans un modèle physique de trajectoire de balle, basé sur des équations différentielles, qui fera la prédiction (MÜLLING, KOBER et PETERS 2010 ; Xiaopeng CHEN et al. 2015 ; Z. ZHANG, XU et TAN 2010 ; Y. ZHANG et al. 2014). Cette méthode nécessite donc de connaître le modèle physique en avance et quelques uns de ses paramètres comme le poids et la taille de la balle. De plus, certains paramètres comme le spin de la balle sont difficiles à extraire, ainsi que des problèmes de distorsion d'images peuvent amener des erreurs variables selon la position de la balle sur l'image.

La seconde méthode utilise des ANNs, dont principalement des *Recurrent Neural Network* (RNNs) (ZHAO et al. 2018) et des modèles auto-régressifs, afin d'apprendre par expérience la trajectoire de la balle et de pouvoir ensuite la prédire. Cependant, ces modèles sont souvent plus lourds, plus difficiles à utiliser en temps réel et présentent des erreurs cumulatives (plus

---

on prédit dans le futur, plus l'erreur est forte) (GOMEZ-GONZALEZ et al. 2020). (NIKHIL et MORRIS 2019) ont montré que des CNNs pouvaient concurrencer les RNNs tout en nécessitant moins de calculs, dans le cadre de trajectoire continue et sans "états compliqués", comme par exemple pour des trajectoires pédestres ou des trajectoires de balles.

Comme montré dans la section précédente (2.6), les humains et plus particulièrement les experts sont capables de prédire les trajectoires de balles avant que le joueur adverse l'ait renvoyé, selon son mouvement. (SUDA, MAKINO et SHINODA 2019) ont testé la capacité d'un ANNs à prédire une trajectoire selon seulement le mouvement du joueur, 0.3 secondes avant que celui-ci ne la renvoie et ont montré qu'il était ainsi possible de prédire la trajectoire future du ballon en utilisant le mouvement du joueur.

Du côté des SNNs, certaines études ont été faites afin d'effectuer des prédictions depuis des événements générés par une caméra événementielle. Certaines d'entre elles utilisent des LSMs, mais leurs prédictions restent limitées à des durées courtes (de l'ordre de quelques ms) (BURGSTEINER et al. 2007 ; KAISER et al. 2017). Une autre étude a utilisé des connexions récurrentes anisotropes avec latences qui permettent ainsi de continuer à propager l'activité électrique induite par le mouvement initial du stimulus, même lorsque celui-ci est stoppé (KAPLAN et al. 2013). Ce type de prédiction est intéressant pour des prédictions locales, durant lesquelles un mouvement peut paraître "rectiligne et uniforme", mais plus difficilement applicables à des trajectoires plus complexes, comme des trajectoires de ballons. (GIBSON, HENDERSON et WILES 2014 ; SHON, RAO et SEJNOWSKI 2004) utilisent aussi des connexions récurrentes avec latences pour prédire les suites des trajectoires. (SHON, RAO et SEJNOWSKI 2004) utilise une règle STDP pour adapter ces connexions, sur des stimuli proches de (KAPLAN et al. 2013). (GIBSON, HENDERSON et WILES 2014) utilise une règle d'apprentissage pour gérer les poids et latences des connexions synaptiques permettant ainsi des prédictions sur des fenêtres temporelles plus longues et des mouvements plus complexes que (KAPLAN et al. 2013). Cependant, les prédictions sont aussi effectuées sur des stimuli très contrôlés et répétitifs (un point effectuant un mouvement oscillatoire).

## 2.8 Problématique

Les SNNs, considérés comme la troisième génération de réseaux de neurones (MAASS 1997) ont permis d'ouvrir de nouvelles opportunités au traitement de l'information et à la compréhension du codage neural à travers des simulations bio-inspirées. Ceux-ci sont dans la continuité des ANNs. Ils permettent un traitement automatique de l'information permettant de répondre à différents besoins traitement du signal audio, visuel, classification, prédiction, etc. Ainsi le fonctionnement général des ANNs est très similaire à celui des SNNs, à la différence que ceux-ci nécessitent une dimension temporelle pour coder l'information. Ainsi, plus particulièrement dans le domaine du traitement visuel automatique, les SNNs peuvent effectuer des tâches similaires aux ANNs, avec pour le moment des performances généralement plus faibles, mais avec un écart qui tend à se réduire. Cependant, les SNNs et plus particulièrement le traitement événementiel, permettent de réduire la consommation d'énergie et assurent un traitement en temps réel (FARABET et al. 2012), grâce notamment au développement de puces neuromorphiques.

Cependant, les caméras événementielles actuellement existantes utilisent un traitement limité à un filtrage temporel. Or, comme présenté dans la section 2.1.2, la rétine effectue un filtrage spatio-temporel. C'est pourquoi durant cette thèse nous utilisons une caméra événementielle permettant d'effectuer un filtrage spatio-temporel avec différents filtres spatiaux implémentables.

L'utilisation de SNNs a permis de montrer qu'ils sont de bons candidats pour effectuer du

---

traitement visuel, similairement à ce que ferait un CNN par exemple. De plus, les SNNs profitent d'une règle d'apprentissage non-supervisée, la STDP, qui a pour particularité d'être inspirée de l'apprentissage effectué dans le cortex visuel et qui est donc très naturellement utilisée et développée dans les SNNs. Ce type d'apprentissage non-supervisé a notamment pour avantage de ne pas nécessiter de phase de labellisation et de jeux de données moins grands (LÄNGKVIST, KARLSSON et LOUTFI 2014; TUSTISON, AVANTS et GEE 2019). Cette règle permet d'identifier des corrélations temporelles dans le signal d'entrée et de renforcer les synapses corrélées, permettant ainsi aux neurones de devenir sélectifs à différents patterns.

Les SNNs, avec une règle d'apprentissage STDP, ont déjà été utilisés pour faire de l'analyse et du traitement visuel. Il est cependant nécessaire d'ajouter des délais synaptiques afin d'avoir des neurones sélectifs à certaines propriétés du mouvement, comme la direction et la vitesse de celui-ci par exemple. Ceux-ci peuvent être implémentés de différentes manières (connexions FF/FB/latérales, excitatrices/inhibitrices, apprentissage ou sélection par STDP des délais).

Différentes méthodes de codage de l'information en spikes existent, comme expliqué section 2.2.6. Deux méthodes principales dominent, le codage fréquentielle et le codage par latence, bien qu'elles ne soient pas incompatibles. Le codage par latence permet d'expliquer la rapidité du traitement visuel et de réduire le nombre de spikes générés pour coder l'information, tout en conservant de très bonnes performances. En effet, en prenant l'exemple d'une caméra DVS et d'une forte et rapide variation de luminosité due à un mouvement, les événements seront générés plus tôt et plus nombreux que lors d'une faible et lente variation de luminosité. La caméra utilisée lors de cette thèse, présentée dans la section 3.1.1, nous permet de choisir la méthode de codage pour générer les événements et donc de n'utiliser qu'un codage temporel par exemple.

Ainsi, les caméras événementielles (notamment la caméra NeuroSoc) et les SNNs semblent être de bons candidats pour analyser des scènes visuelles en mouvement. L'analyse et la prédiction de mouvement et de trajectoires de balles sont des tâches de haut niveau, les premières aires visuelles permettent d'extraire de l'information locale, mais seulement les aires MT, MST (corrélations d'activités) (SPERING et al. 2011) et supérieures, dont notamment des aires multimodales et sensorimotrices qui sont activées pour des prédictions de poursuites et saccades oculaires (prédictions très court terme)(THIER et ILG 2005). Les régions de l'AON spécialisées dans le sensorimoteur semblent elles aussi très importantes dans la prédiction de scène dans le domaine du sport et jouent un rôle important dans l'expertise. Il semble ainsi hors de portée de cette thèse de chercher à répliquer le fonctionnement de telles aires cérébrales. En effet, celles-ci sont fortement impactées par l'apprentissage moteur et nécessitent une perception visuelle active. De plus, celles-ci nécessiteraient des architectures complexes. En effet, augmenter le nombre de couches et utiliser des connexions principalement FF ne permettraient pas de justifier une modélisation d'aires visuelles telles que MT ou MST. Ainsi, dans les travaux réalisés durant cette thèse, le SNN modélisera principalement les réponses des premières aires visuelles. Nous verrons ensuite si l'ajout de couches supplémentaires permet d'affiner la précision du SNN et si cela permet ensuite de prédire la trajectoire de la balle.

De plus, lors de cette thèse nous avons pour objectif d'évaluer la pertinence d'utiliser une nouvelle caméra événementielle "hybride", la NeuroSoc, qui a pour particularité d'utiliser une caméra *frame-based*, permettant ainsi d'effectuer du filtrage spatio-temporel. L'objectif était ensuite d'évaluer les performances d'un SNN entraîné avec une règle STDP et recevant les événements générés par la NeuroSoc, pour effectuer de l'analyse de mouvement et voir si cette solution permet d'effectuer des prédictions de trajectoires de balles avec précision. Enfin, le dernier objectif était d'évaluer et de comparer les performances d'humains experts ou non-experts sur des tâches similaires.

# Chapitre 3

## Méthodologie

### 3.1 Méthodes

Ce chapitre a pour objectif de présenter les différents détails techniques ainsi que les méthodes utilisés dans les différentes études de cette thèse. La première partie de ces méthodes présentera brièvement la caméra NeuroSoc développée par *Yumain*. Trois stagiaires du Cerco avaient déjà participé au développement de ce capteur conjointement avec *Yumain* avant le début de cette thèse. Les premiers mois de celle-ci ont principalement été axés sur la finalisation du développement et la mise au point d'une première version stable. Pour cela, un développement pour permettre la récupération en mémoire et l'envoi des événements via une connexion Ethernet a été nécessaire (cette partie est résumée en annexe, chapitre 9). Puis une phase de tests et un travail de debuggage conjointement avec Clément Joly de *Yumain*.

La seconde partie comprendra les détails techniques du SNN implémenté et utilisé dans les études qui suivent.

#### 3.1.1 NeuroSoc

Comme présenté dans la section 2.2.1, les caméras événementielles présentent de nombreux avantages, tant sur le plan de la plausibilité biologique que sur le plan de l'ingénierie (consommation réduite de ressources, processus asynchrone, etc). Il existe actuellement différentes caméras événementielles disponibles (Prophesee, iniVation, Insightness, Samsung, CelePixel). Chacune présente ses spécificités mais utilise un mode de fonctionnement similaire basé sur l'évolution temporelle de l'intensité de chaque pixel (section 2.2.1). Bien que présentant des performances remarquables en termes de fréquence temporelle et de gamme dynamique, elles n'intègrent pas de filtrage spatial en amont de la génération des événements. Or, comme décrit dans la section 2.1.2, la rétine ainsi que les premières couches du cortex visuel (LGN, V1, ...) effectuent un filtrage spatio-temporel. Le filtrage spatial réalisé au sein de la caméra NeuroSoc est similaire à celui d'une différence de gaussienne ou d'un filtre de Gabor orienté. De plus, durant les dernières années, de nouvelles solutions ont été proposées, permettant de générer des événements à partir de capteurs de type CMOS (ABDERRAHMANE et MIRAMOND 2019 ; AKIDA 2021).

De nouveaux systèmes utilisent des caméra standard CMOS afin d'extraire des événements, permettant ainsi de pouvoir utiliser des filtres spatiaux avant la génération de événements. Ces systèmes permettent ainsi de profiter du large panel de caméras CMOS et de leurs avantages, permettant par exemple d'utiliser des caméras avec des résolutions allant jusqu'à 2M pixels (CAIMAN CAMERA 2021), alors que la résolution est limitée dans les caméras événementielles actuelles (LICHTSTEINER, C. POSCH et T. DELBRUCK 2008 ; SON et al. 2017 ; BRANDLI et al. 2014 ; Christoph POSCH, MATOLIN et WOHLGENANT 2011).

La NeuroSoc développée par Yumain s'inscrit donc dans cette catégorie de solution utilisant une caméra standard synchrone, à laquelle on ajoute une phase de filtrage spatio-temporel afin d'extraire et de générer les événements.

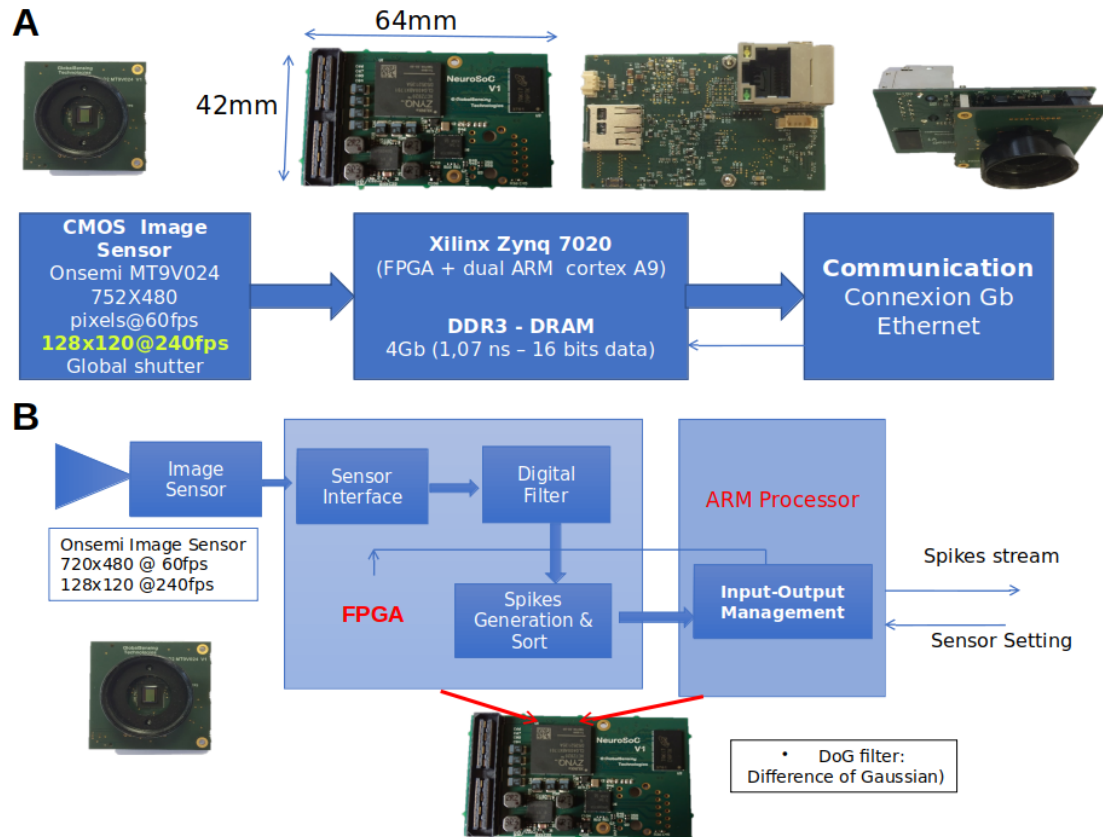


FIGURE 3.1 – Architecture de la NeuroSoc

La caméra NeuroSoc utilisée durant cette thèse est composée d'un capteur visuel CMOS MT9024 de *On Semiconductor* fonctionnant en mode global-shutter et pouvant faire des acquisitions à 240 FPS avec une résolution de  $128 \times 120$  pixels. Elle est également équipée d'une carte électronique composée d'un MPSoc Zynq 7020 (Xilinx) et d'une mémoire DDRAM de 4 GBits (figure 3.1 panneau A). Le MPSoc Zynq est composé d'une partie FPGA connectée à deux coeurs de processeurs ARM. Le traitement du flux vidéo et de la génération des événements est ensuite divisé en plusieurs parties (figure 3.1 panneau B). Dans un premier temps, le capteur vidéo CMOS fait l'acquisition successive des images. Celles-ci sont ensuite transmises au Zynq, qui effectuera le filtrage spatio-temporel. Le FPGA au sein du MPSoc Zynq fait en premier lieu un filtrage temporel en soustrayant l'image à l'instant  $t$  générée par le capteur CMOS avec l'image précédente, à l'instant  $t - 1$ . Ce premier filtrage s'apparente très fortement au filtrage effectué par les caméras événementielles. L'image générée est ensuite convoluée avec un filtre spatial. Dans le cadre de cette thèse, nous avons utilisé une différence de gaussiennes (*Différence of Gaussians*) (DoG) de taille  $5 \times 5$ , avec pour valeur maximale 16 au centre de la DoG et -1 comme valeur minimale (cf. figure 3.2).

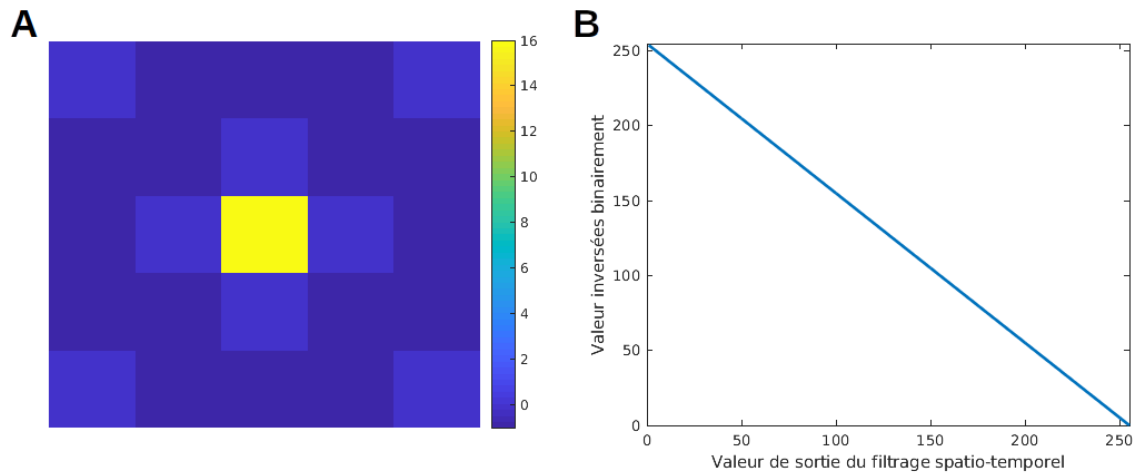


FIGURE 3.2 – Panneau A : Kernel utilisé pour la DoG. Panneau B : Valeurs complémentées

D'autres filtres de différents types ou tailles peuvent être utilisés. On obtient alors une matrice/image constituée de valeurs  $I_{dog}(x, y)$  positives, représentant les spikes de type ON et de valeurs  $I_{dog}(x, y)$  négatives, représentant les spikes de type OFF. Un seuillage est ensuite appliqué. Seuls les pixels dont la valeur absolue est supérieure à ce seuil sont conservés, puis triés selon la valeur absolue de leur intensité. Ces valeurs qui sont codées sur 8 bits sont ensuite complémentées (0 devient 255 et inversement) générant ainsi  $I_{inv}(x, y)$ , permettant d'avoir une temporalité du spike inversement proportionnelle à l'intensité du filtrage  $I_{dog}(x, y)$  (S. THORPE, DELORME et VAN RULLEN 2001; CHAUHAN et al. 2018; VANRULLEN, GUYONNEAU et S. J. THORPE 2005; MASQUELIER et S. J. THORPE 2007; S. THORPE, DELORME et VAN RULLEN 2001). Les valeurs  $I_{inv}(x, y)$ , la position spatiale du spike ainsi que sa polarité sont ensuite transmises au processeur ARM au sein du MPSoc Zynq. L'envoi des spikes et le contrôle du capteur (gain, temps d'exposition, FPS, ...) se fait ensuite via une connexion Ethernet. La valeur du seuil est déterminée manuellement en fonction du contexte et du stimulus (luminosité, contraste stimulus/arrière plan, vitesse du stimulus, etc), afin de garantir l'extraction du maximum d'informations possibles et de conserver un bruit de niveau faible.

Afin de garantir un bon fonctionnement du filtrage, le capteur vidéo CMOS doit fonctionner en mode "global-shutter" (acquisition instantanée d'images), et un temps d'exposition faible (de l'ordre de quelques millisecondes), afin de garantir un FPS élevé.

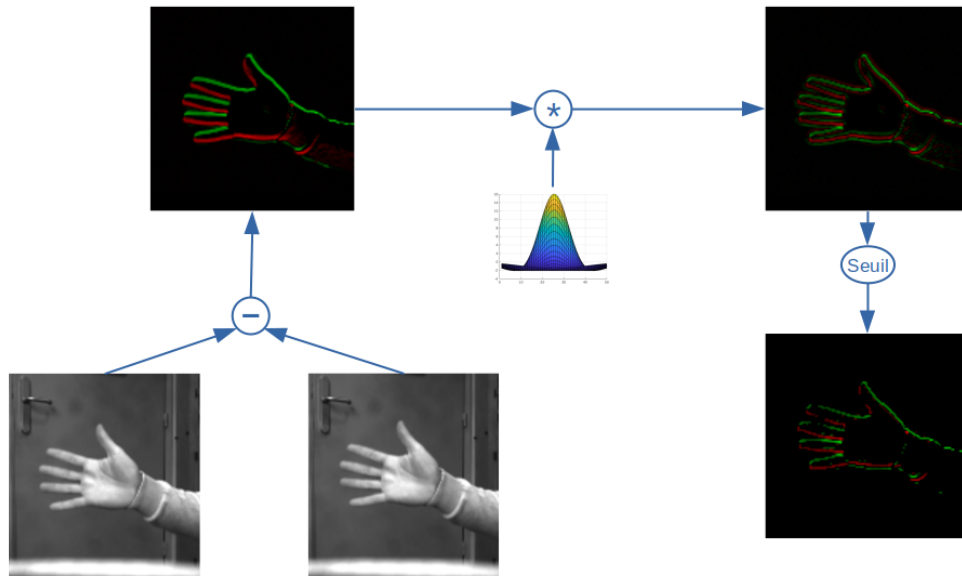


FIGURE 3.3 – Filtrage spatio-temporel effectué par la caméra

## 3.1.2 SNN

### 3.1.2.1 Architecture/topologie

Dans les études qui suivent, nous utilisons un SNN multi-couches avec des connexions FF et des connexions latérales inhibitrices. Le SNN que nous utilisons est de type convolutionnel (CSNN), similaire à (S.-R. KHERADPISHEH et al. 2018; THIELE, BICHLER et DUPRET 2018; MASQUELIER et S. J. THORPE 2007; M. MOZAFARI et al. 2018; TAVANAIE et A. S. MAIDA 2017; PAREDES-VALLES, SCHEPER et CROON 2019). Ainsi, comme illustré par la figure 3.4, les neurones sont organisés de manière rétinotopique, c'est à dire qu'un groupe de  $N_{RF,l}$  neurones sont situés spatialement dans leurs couches et reçoivent des spikes de  $s \times s \times d$  neurones centrés sur la même position (avec  $s$  : largeur/hauteur et  $d$  : profondeur de la couche précédente ( $N_{RF,l-1}$ ). Il existe un chevauchement important entre les différents champs récepteurs (*Receptive Field*), chacun étant connecté à la même zone que le précédent, décalé d'un ou deux pixels (stride = 1 ou 2). Une méthode de partage de poids permettant à un neurone  $n$  à une position donnée permet de généraliser la modification de ses poids synaptiques à tous les autres neurones partageant les mêmes poids, de façon similaire aux CNNs. Il y a donc  $N_{RF,l}$  filtres par couche  $l$  définis. Il y aura ainsi autant de cartes neuronales qui contiendront tous les neurones répartis sur le champs visuel et partageant les même poids synaptiques et donc le même filtre. Ainsi, chaque neurone a une position spatiale, est connecté à une partie restreinte de la couche précédente (définie par la position de ce même neurone), possède son propre potentiel de membrane, etc. Certaines architectures, notamment lors des premières simulations, n'utilisent pas cette méthode de partage de poids. Le fonctionnement du SNN reste similaire, sauf que chaque neurone a sa propre matrice de connexion. Les connexions latérales inhibitrices utilisées impactent les neurones connectés au même RF que le neurone qui vient de décharger. Le code permettant de simuler les SNNs a été développé durant cette thèse et est codé en langage C++.

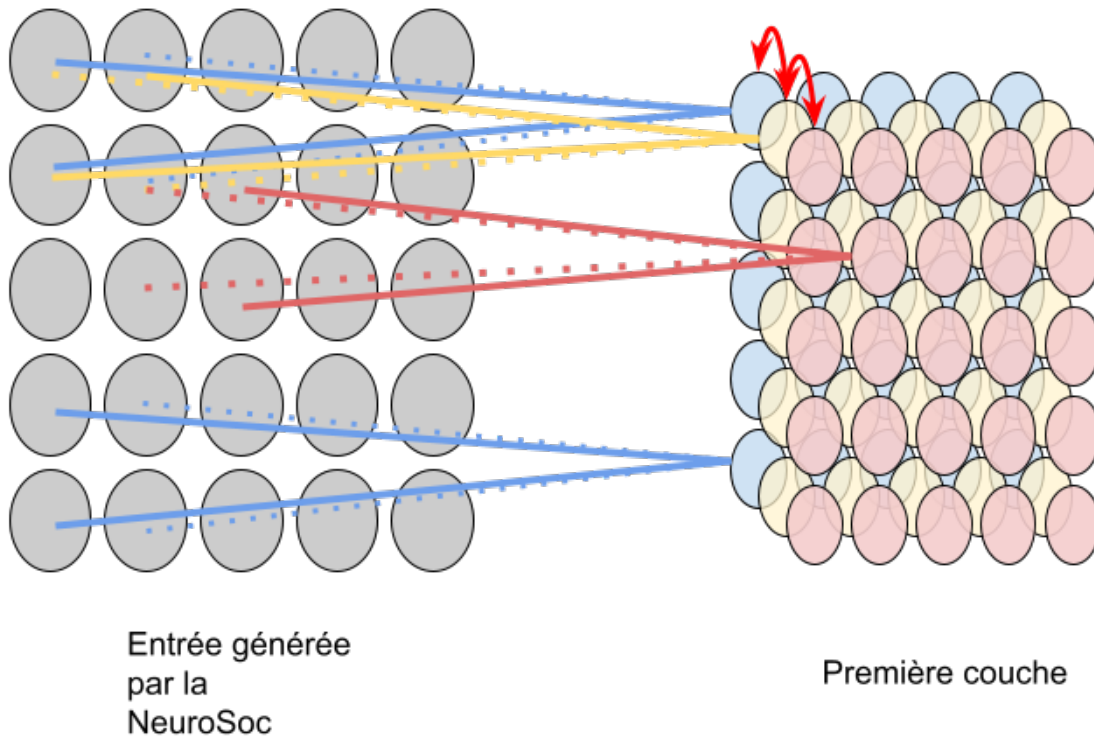


FIGURE 3.4 – Architecture du SNN utilisé. Les neurones de la première couche sont rétinotopiquement organisés et connectés à une zone restreinte de la couche précédente (l’entrée dans cet exemple). Les connexions fortes entre les neurones et l’entrée sont caractérisées par des traits pleins et les faibles par des traits en pointillés. Les neurones en bleu représentent la première carte neuronale caractérisée par un filtre partagé par tous ces neurones (fortes connexions avec les deux neurones sur la gauche et faible pour ceux sur la droite). Les neurones en jaune et bleu représentent les cartes neuronales caractérisées par deux autres filtres. Les flèches en rouge représentent les inhibitions latérales.

### 3.1.2.2 Modèle du neurone

Lors de ces études, nous utilisons un modèle de neurone intégrateur à fuite. Ce modèle de neurone permet d’être mis-à-jour de manière événementielle, lorsque cela est nécessaire et donc lorsque un spike est transmis au neurone (équation (3.1)). Cela permet de profiter au maximum de la caractéristique éparsée de la sortie de la caméra, et réduire la consommation de ressource. L’intégration d’un spike d’entrée se fait de façon instantanée, en suivant l’équation (3.1) permettant de mettre à jour le potentiel de membrane lorsqu’un neurone reçoit un spike :

$$v_j(t) = v_j(t_{Last}) \times \exp(-(t - t_{Last})/\tau_L) + w_{i,j} \times Wmax_L \quad (3.1)$$

Avec :

- $v_j$  : le potentiel de membrane du neurone  $j$
- $t$  : instant du spike reçu par le neurone  $j$
- $t_{Last}$  : instant du spike de la précédente mise-à-jour du neurone  $j$
- $\tau_L$  : constante de temps du potentiel de membrane du neurone
- $w_{i,j}$  : poids de la connexion synaptique entre le neurone afférent  $i$  et le neurone post-synaptique  $j$  (entre 0 et 1)



---

—  $W_{max_L}$  : facteur fixe multiplié aux poids synaptiques. Différent pour chaque couche  $L$

Habituellement une période réfractaire est ensuite utilisée empêchant le neurone de recevoir de nouvelles entrées et donc de décharger durant une certaine période. La parcimonie des spikes émis par la NeuroSoc et des sorties des couches de neurones font que les neurones ne déchargent qu'une seule fois lorsqu'ils détectent un stimulus particulier, hormis quelques rares cas dans la première couche. La période réfractaire n'a donc ici pas d'utilité et ne serait qu'un mécanisme rajoutant des opérations non-nécessaires.

### 3.1.2.3 Adaptation de seuil

Comme exposé dans la section 2.4.3.1, l'utilisation de connexions latérales inhibitrices avec délais ne fonctionne que sous certaines conditions. Le neurone doit émettre plusieurs décharges sur une relativement longue fenêtre temporelle et/ou plusieurs neurones doivent coder pour la même information avec un décalage temporel, afin de pouvoir inhiber les neurones non-sélectifs à ce stimulus (qui codent pour une direction opposée) efficacement. Dans le cas où seulement un seul neurone décharge et ne décharge qu'une seule fois, il est difficile d'utiliser une connexion latérale inhibitrice qui fonctionne dans toutes les configurations (différentes vitesses de stimulus par exemple). Une solution serait alors que plusieurs spikes inhibiteurs soient envoyés avec des délais différents afin de simuler un des deux cas de la figure 2.17 (Panneau A et B).

Nous avons donc testé durant cette thèse une méthode cherchant à inhiber durant une longue période les neurones lorsqu'un premier décharge (figure 3.5). Afin de répliquer un cas où plusieurs connexions latérales inhibitrices envoient des spikes à différentes temporalités, nous avons remplacé ces inhibitions par une variation du seuil du potentiel de membrane. Une des conditions pour que cette méthode fonctionne est que le neurone dit sélectif au stimulus présenté décharge en premier afin de pouvoir inhiber les autres. Un processus similaire est utilisé dans (PAREDES-VALLES, SCHEPER et CROON 2019). Cependant, à la différence de notre méthode il n'y a ni inhibition ou variation de seuil, mais lorsqu'un neurone décharge, il rentre en période réfractaire ainsi que tous les mêmes neurones rattachés au même champ récepteur, les empêchant ainsi de décharger durant une certaine période. Notre méthode permet cependant à d'autres neurones de décharger si leur potentiel de membrane monte suffisamment haut pour dépasser le seuil adaptatif. Ainsi cette méthode s'approche aussi de celle utilisée par (GIULIONI et al. 2016; HAESSIG, CASSIDY et al. 2018; SHON, RAO et SEJNOWSKI 2004), durant laquelle ils utilisent des neurones inhibiteurs qui restent actifs durant une certaine période pour inhiber les autres neurones et permettre une sélectivité à la direction. Notre méthode diffère par le fait que nos neurones recevant des connexions FF sont seulement excitateurs, et seules les connexions latérales sont inhibitrices. Alors que dans (GIULIONI et al. 2016; HAESSIG, CASSIDY et al. 2018), il existe des neurones excitateurs et inhibiteurs dans les deux configurations. Ainsi, un neurone qui décharge pour un mouvement allant de la gauche vers la droite, devrait avoir des poids forts positionnés sur la partie gauche de son champ récepteur, afin qu'il décharge plus tôt qu'un neurone appartenant au même champ récepteur, sélectif au mouvement opposé qui aurait des poids forts sur la partie droite. Cette condition est aussi un des principes du codage par latence d'un neurone. Un neurone très sélectif à un stimulus sera plus rapidement excité que les autres et déchargera plus rapidement. De plus, la règle d'apprentissage STDP permet de remplir cette condition, car elle permet aux neurones de décharger au début du stimulus présenté (MASQUELIER, GUYONNEAU et S. J. THORPE 2008; BURKITT et HOGENDOORN 2021) et donc de renforcer les poids situés à gauche du champ récepteur lors d'un mouvement allant de la gauche vers la droite, permettant ainsi au neurone de décharger en premier. Lors de la présentation de stimuli de directions opposées (gauche/droite), certains neurones auront donc des poids de connexion synaptique forts d'un côté ou de l'autre du champ récepteur du neurone. Avec le mouvement d'un objet dont la direction va vers la gauche, les neurones ayant

des poids forts du côté gauche du champ récepteur déchargeront donc naturellement avant ceux ayant des poids forts sur la droite. En ajoutant une longue augmentation de seuil, lorsque le premier neurone décharge, il empêchera ensuite le second neurone de décharger (et inversement dans la direction opposée), similairement à l'utilisation de connexions latérales inhibitrices qui auraient réduit le potentiel de membrane du second neurone.

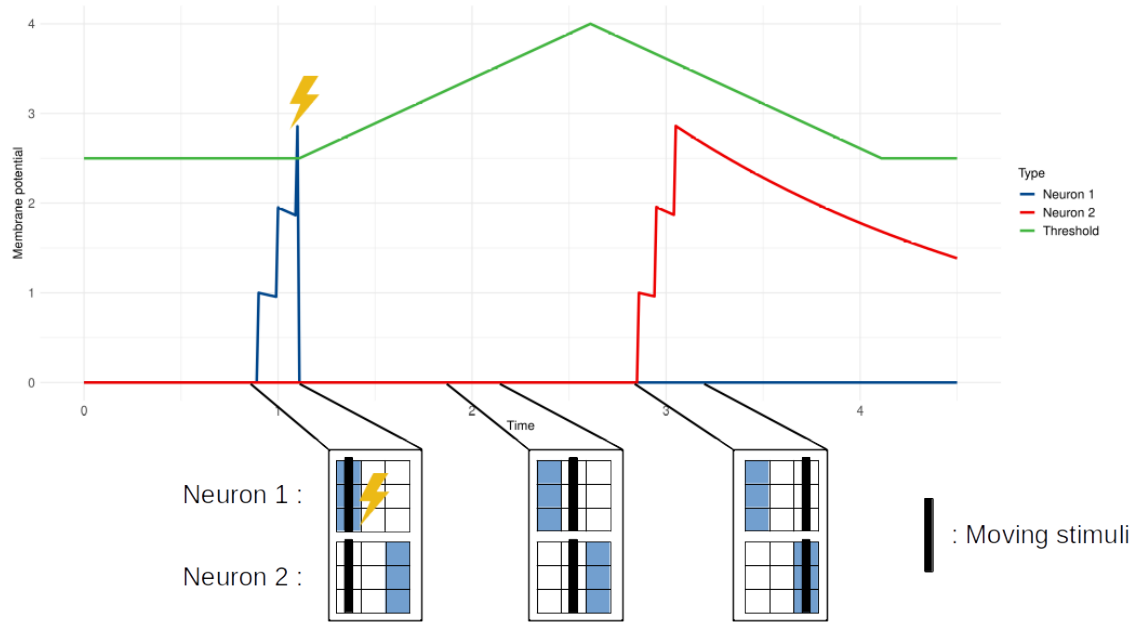


FIGURE 3.5 – Méthode d'adaptation de seuil utilisée. Lorsque le neurone 1 décharge ( $t=1$ ), un mécanisme d'adaptation de seuil est initié, durant lequel le seuil, va augmenter puis revenir à sa valeur de défaut. Lorsque le neurone 2 est excité par le stimulus ( $t=3$ ), le seuil est trop élevé, il ne peut donc pas décharger.

Cette variation de seuil doit donc évoluer temporellement. Afin de la simplifier, l'évolution de cette variation de seuil est de forme en "V" inversée (équation (3.2)). Le seuil augmente linéairement durant une première période  $T_{Thresh}$ , jusqu'à atteindre une valeur maximale  $U_{MaxThresh}$ . Puis décroît linéairement durant la même période jusqu'à revenir à une valeur égale à 0.

$$\begin{aligned}
 U_{Long,n}(t) &= U_{MaxThresh} \times \frac{t - t_n}{T_{Thresh}} \quad \text{si } t - t_n \leq T_{Thresh} \\
 U_{Long,n}(t) &= U_{MaxThresh} \times \left(1 - \frac{(t - t_n) - T_{Thresh}}{T_{Thresh}}\right) \quad \text{si } T_{Thresh} < t - t_n < 2T_{Thresh} \\
 U_{Long,n}(t) &= 0 \quad \text{sinon}
 \end{aligned}
 \tag{3.2}$$

Avec  $t_n$  : instant du dernier spike émis par le neurone  $n$ .

L'intensité de l'inhibition est proportionnelle à l'excitation des neurones. Comme on le voit dans la figure 2.17, si l'intensité du signal d'entrée est forte, plus de neurones déchargeront et avec une plus grande fréquence, nécessitant ainsi une plus grande inhibition. Conserver une balance entre excitation et inhibition est important pour un SNN, cela agit comme un processus homéostatique et permet de garder une activité proportionnelle au signal d'entrée (DENÈVE et MACHENS 2016). Dans notre cas, il est plus difficile d'avoir une inhibition proportionnelle au signal d'entrée, les neurones ne déchargent qu'une seule fois lors de la présentation d'un

stimulus et peu de neurones (parmi ceux connectés au même champ récepteur) ne déchargent. L'inhibition instantanée et l'augmentation du seuil participent à ce codage parcimonieux ayant l'avantage de ne propager que peu de spikes. Une méthode serait d'avoir une inhibition qui prend en compte les neurones spatialement proches mais qui n'appartiennent pas au même champ récepteur. Une seconde méthode, celle que nous avons adoptée, est de considérer que l'intensité moyenne du potentiel de membrane des neurones rattachés au même champ récepteur serait un bon indicateur (comme le neurone Poisson qui décharge avec une fréquence proportionnelle au signal reçu). Ainsi la valeur  $U_{MaxThresh}$  est calculée de façon à être proportionnelle à cette moyenne des potentiels de membrane. Plus le nombre de neurones ayant un potentiel de membrane élevé au moment où le neurone décharge est important, plus l'inhibition est forte. Cette méthode a aussi pour avantage de définir une inhibition forte au début de l'apprentissage car les neurones étant peu sélectifs, ils sont tous susceptibles de décharger pour un même stimulus et ont donc des potentiels de membranes élevés lors de la décharge du premier neurone. Cette tendance s'amointrit avec l'entraînement. Les neurones ont des matrices de poids différentes, seuls quelques neurones ont des potentiels de membrane élevés lors de la décharge d'un neurone. Les autres n'étant pas sélectifs à ce stimulus sont peu activés.

$$U_{PropInh} = \sqrt{\frac{1}{N_{RF}} \sum_{n=i}^{i+N_{RF}} (v_n(t) - U_{AT,n}(t))} \quad (3.3)$$

$$U_{MaxThresh} = f_{Long} \times U_{PropInh}$$

$$U_{Inst} = f_{inst} \times U_{PropInh}$$

Avec  $v_n$ , le potentiel de membrane du neurone  $n$ .

### 3.1.2.4 Délais

Comme montré par (SHON, RAO et SEJNOWSKI 2004; WENISCH, NOLL et HEMMEN 2005) des connexions simplement FF permettent une faible sélectivité à la direction. Afin d'améliorer la sélectivité spatio-temporelle des neurones, nous avons choisi d'ajouter des délais dans les connexions synaptiques. Différentes méthodes peuvent être utilisées pour ajouter des délais (section 2.4.3.1), nous avons fait le choix de la sélection de délais. Cette méthode consiste à dupliquer des synapses entre deux neurones et d'affecter des délais différents à chacune d'entre elles (PAREDES-VALLES, SCHEPER et CROON 2019). Lorsque le neurone pré-synaptique émet un spike, celui-ci sera envoyé au neurone post-synaptique plusieurs fois à des instants  $t$  différents et via des synapses différentes. La règle d'apprentissage STDP sélectionne ensuite les synapses avec des délais spécifiques permettant ainsi d'améliorer la sélectivité du SNN.

Nous avons sélectionné cette méthode, car elle ne nécessite pas d'apprentissage supplémentaire pour sélectionner les valeurs des délais et permet de propager un spike émis par un neurone pré-synaptique à tous les neurones post-synaptiques auquel il est connecté simultanément. Ceci permet donc de réduire les opérations et d'accélérer le traitement des spikes.

Afin de simplifier le SNN, les délais n'ont été appliqués que sur la première couche, les couches suivantes intègrent les délais de la première couche via les spikes émis par celle-ci.

### 3.1.2.5 STDP

La règle d'apprentissage utilisée est une version multiplicative de la règle de (BICHLER et al. 2012), permettant ainsi une limite des valeurs des poids entre 0 et 1 (équation (3.4)).

$$\begin{aligned} \Delta W_{i,j} &= a_{LTP} \times (1 - W_{i,j}) \quad \text{si } t_i - t_j < \tau_{LTP} \\ \Delta W_{i,j} &= -a_{LTD} \times W_{i,j} \quad \text{sinon} \end{aligned} \tag{3.4}$$

Avec :

- $\Delta W_{i,j}$  : variation apportée à la connexion synaptique entre le neurone pré-synaptique  $j$  et le neurone post-synaptique  $i$
- $a_{LTP}$  et  $a_{LTD}$  : intensité de la modification synaptique
- $t_i$  et  $t_j$  : instant du dernier spike des neurones  $i$  et  $j$
- $\tau_{LTP}$  : fenêtre temporelle du LTP

### 3.1.2.6 Sélection des paramètres

La règle d'apprentissage STDP permet de modifier les poids des connexions synaptiques de manière non-supervisée afin d'obtenir des neurones qui déchargent pour des stimuli spécifiques. Cependant, l'utilisation de la règle d'apprentissage ne garantit pas une convergence des poids synaptiques et une bonne performance du SNN, certains paramètres sont à définir (constante de temps, fenêtre d'apprentissage STDP, seuil, etc). La plupart du temps ces paramètres sont définis afin de respecter les valeurs retrouvées dans la biologie, en accord avec l'état de l'art, soit selon l'expertise de la personne qui définit le SNN ou l'ANN ou soit en définissant ces paramètres après une longue période de test et d'optimisation afin d'avoir de meilleurs résultats. En effet, de mauvais paramètres peuvent engendrer de mauvais résultats (OUDJAIL et MARTINET 2021b). (SAXE et al. 2011) montre qu'un ANN non-entraîné avec des paramètres bien sélectionnés peut concurrencer les performances d'un ANN entraîné. Ils proposent ainsi de définir dans un premier temps ces paramètres sur un réseau de neurones non-entraîné, permettant de les tester plus rapidement, puis d'effectuer ensuite l'apprentissage. Cette méthode a seulement été validée sur des ANN avec une seule couche, la non-linéarité augmentant avec l'ajout de couches pouvant limiter la corrélation des performances entre un réseau entraîné et non-entraîné. Nous avons dans un premier temps testé cette méthode. La corrélation semblait bien maintenue avec un SNN mais ne permettait pas d'avoir de bons résultats avec un SNN multi-couches. Par exemple, le flux de spikes en sortie de la couche  $l-1$  (entrée de la couche  $l$ ) varie beaucoup entre le début et la fin de l'apprentissage. Selon le ratio  $a_{LTP}/a_{LTD}$  le nombre de spikes émis par une même couche peut être grandement réduit, le flux de spikes transmis à la couche suivante ayant fortement varié (nombre de spikes, corrélation entre les spikes, etc). Les paramètres de la couche suivante  $l$  ne sont ainsi plus adaptés à cette entrée.

La solution d'utilisation d'un SNN non-entraîné pour définir plus rapidement les paramètres ne fonctionnant pas dans le cas d'un SNN multi-couches, nous avons alors opté pour l'utilisation d'un algorithme génétique (AG). Ce type d'algorithme a déjà fréquemment été utilisé pour définir les paramètres d'algorithme ou d'ANN/SNN (BICHLER et al. 2012 ; LEUNG et al. 2003 ; BASHIRI et FARSHBAF GERANMAYEH 2011). Nous avons ainsi utilisé l'AG proposé par (MOHAMMADI et al. 2017) pour optimiser nos paramètres pour les études présentées dans les chapitres 4 et 5.

## 3.2 Travaux préliminaires

Afin d'évaluer le SNN utilisé, différentes phases de tests ont été mises en place. L'objectif était de voir si le SNN était capable d'extraire des informations liées aux mouvements en utilisant des événements générés par la caméra NeuroSoc ou par une simulation qui mime son fonctionnement. Puis, d'évaluer la pertinence de cette solution pour faire de l'analyse et de la prédiction de trajectoires de balles. Pour cela, deux méthodes ont été utilisées. La première

---

était de tester le fonctionnement de la caméra NeuroSoc et du SNN avec des acquisitions de séquences d'images de voitures, afin de voir si on arrivait à extraire de l'information provenant de la caméra. La seconde était de simuler le fonctionnement effectué par la caméra afin de générer un mouvement contrôlé et évaluer les performances du SNN plus précisément.

### 3.2.1 Catégorisation de direction de voitures

La détection et l'analyse de données visuelles de voitures (direction, catégorisation, etc) est une des tâches fréquemment accomplies par des ANNs ou SNNs. Ainsi nous avons voulu tester notre SNN sur ce type de stimulus, car cela permettait de l'évaluer sur des patterns irréguliers (les voitures ne se ressemblent pas) et proche de ce qu'on peut demander à un SNN dans des situations réelles, représentant un bon cas d'application pour Yumain.

En début de thèse, Yumain avait fait des acquisitions séquences d'images de voitures se déplaçant sur une voie à double sens, afin de tester leur caméra (figure 3.6). Nous avons ainsi utilisé ces données afin de tester notre SNN sur une tâche de discrimination de direction de voitures. Un SNN avec une seule couche a été entraîné sur ce jeu de données. Cependant, le mécanisme de partage de poids n'était pas encore implémenté lors de ces premières phases de tests. Ainsi chaque neurone avait sa propre matrice de poids synaptiques. Afin de réduire la taille du SNN et accélérer le traitement, seule une partie des spikes générés étaient utilisés (carré rouge de taille 40x40 pixels sur la figure 3.6). Les neurones étaient connectés à une sous-partie de l'entrée de taille 10x10 pixels. Seize différents neurones étaient connectés pour chaque champ récepteur. Les mécanismes d'inhibition et de délais étaient activés. Le réseau de neurones était entraîné avec la règle STDP. Puis, après l'entraînement, nous avons regardé si certains neurones déchargeaient pour certaines directions. La voie étant à double sens, certaines voitures allaient de la gauche vers la droite ou inversement.

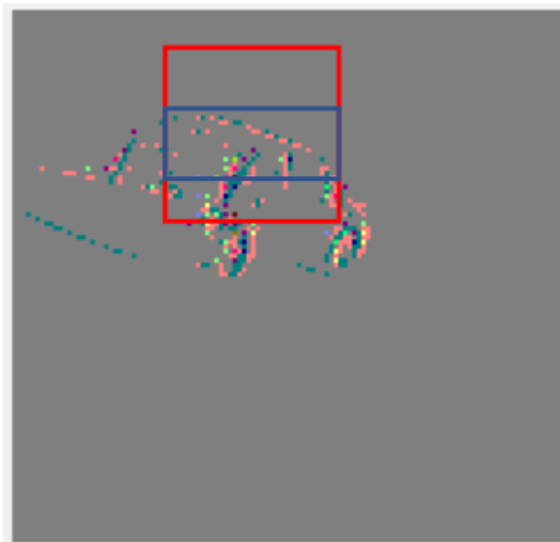


FIGURE 3.6 – Exemple des spikes générés lors du passage d'une voiture.

Ainsi, comme illustré sur la figure 3.7, les neurones déchargent majoritairement pour une direction particulière (illustrée par la couleur de la barre). Il apparaît clairement que les neurones sont devenus sélectifs à la direction du mouvement des véhicules. Les neurones ont une tendance à plus décharger pour une direction que pour son opposé. D'avantage de neurones déchargent pour des directions allant de la gauche vers la droite (jaune). Cela s'explique car les deux différentes directions ne partagent pas exactement les mêmes positions spatiales, les voitures

allant vers la gauche sont situées plus haut que celles allant vers la droite. Ainsi, les neurones qui sont affichés sont davantage sur la partie des voitures se dirigeant vers la gauche (rectangle bleu sur la figure 3.6).

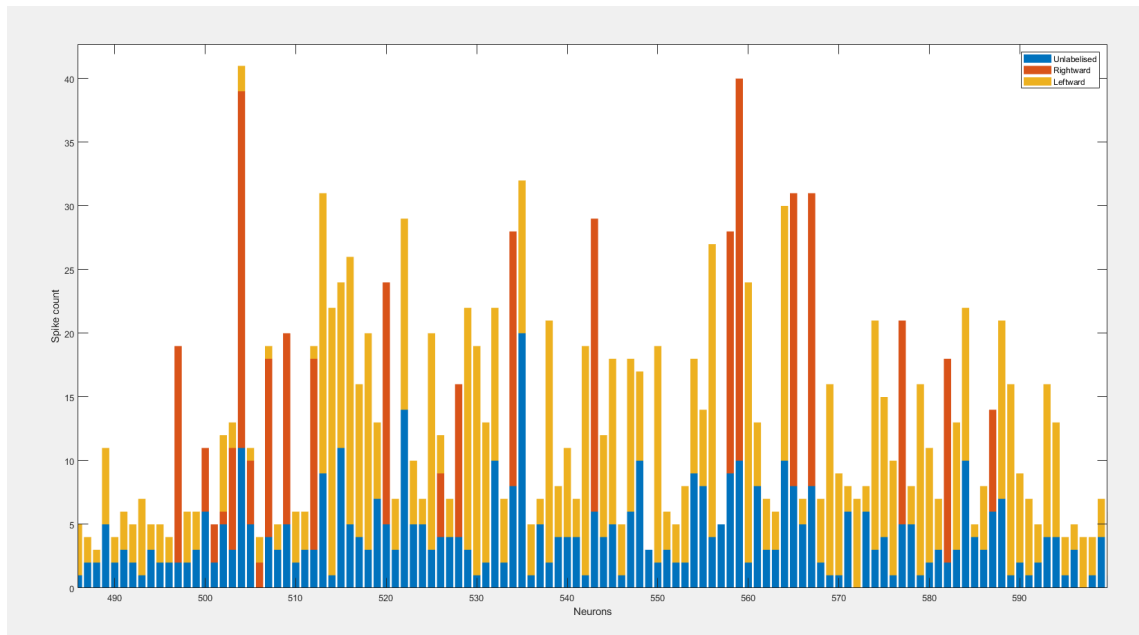


FIGURE 3.7 – Nombre de spikes générés pour des voitures allant de la gauche vers la droite (jaune) ou de la droite vers la gauche. Certaines voitures n'étaient pas labellisées car de nombreuses voitures se croisaient au même moment sur l'image (couleur bleu).

## 3.2.2 Simulations

Afin de tester notre SNN, nous avons utilisé des simulations qui copiaient le fonctionnement de la caméra NeuroSoc, mais avec des jeux de données contrôlés. Cela nous a permis de tester notre SNN sur des trajectoires 2D uniformes (à vitesses et directions constantes), puis sur des simulations de trajectoires de balles (avec effet de la gravité, vitesse et directions évolutives), nous permettant ainsi de se rapprocher de situations plus réelles. Pour cela, nous avons d'abord développé une solution permettant de générer des mouvements uniformes, avec des vitesses, directions et formes (cercle, barre) paramétrables. Cette solution nous permettait de facilement tester le SNN dans des situations contrôlées. Puis nous avons développé un programme en Python permettant de simuler le processus de la caméra et donc de générer des événements à partir de fichier vidéos ou de la sortie du générateur de mouvements. Cette solution reprenait le fonctionnement de la NeuroSoc, c'est à dire, un premier filtrage temporel (différence d'image), suivi d'un filtrage spatial (DoG similaire au filtre utilisé par la NeuroSoc). Les valeurs de sorties supérieures à une valeur de seuil étaient conservées, puis les événements étaient générés avec des latences inversement proportionnelles aux valeurs de sorties.

### 3.2.2.1 Simulation 2D

#### Objectifs et méthodes

Le SNN a été testé sur des simulations de stimuli à deux dimensions, cela permet ainsi de le tester sur une tâche simplifiée dans laquelle on peut contrôler les différents paramètres (taille

de l'objet, vitesse, direction, etc.). Nous avons simulé 500 trajectoires de barres 2D (barre de longueur de 30 pixels et de largeur de 15 pixels, figure 3.8). Leurs directions et vitesses étaient sélectionnées aléatoirement (entre 0 et 359° et une vitesse allant de 0.7 à 2 pixels par image) et constantes au cours d'une trajectoire. Un SNN avec une couche convolutionnelle contenant 128 filtres de taille 9x9 était utilisé. Les filtres étaient appris grâce à la règle STDP. Le SNN a été testé dans des conditions parfaites (balle et fond de couleur uniforme, luminosité contrôlée, pas de bruit, etc.). Mais la précédente expérience ainsi que celles présentées dans les prochains chapitres (4 et 5) nous ont permis de confirmer la viabilité du SNN dans des contextes réels.

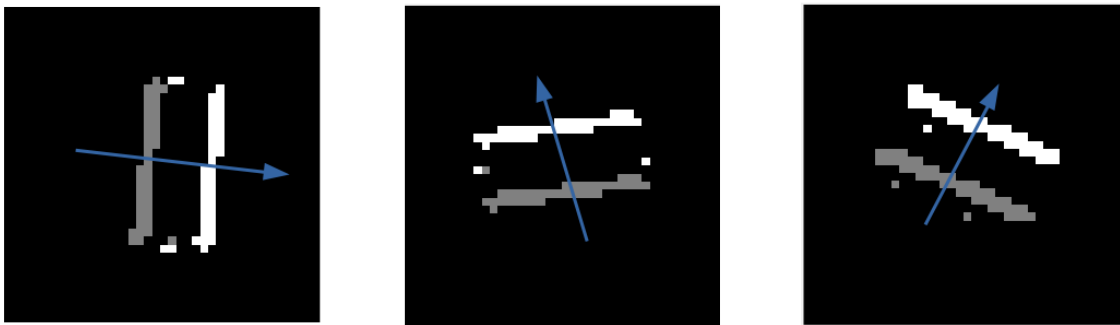


FIGURE 3.8 – Exemple des spikes générés pour différentes trajectoires simulées.

## Résultats

En regardant les matrices de poids de différents neurones affichés sur la figure 3.9, on voit que ceux-ci ont appris différents patterns et que les poids forts sont disposés et orientés différemment.

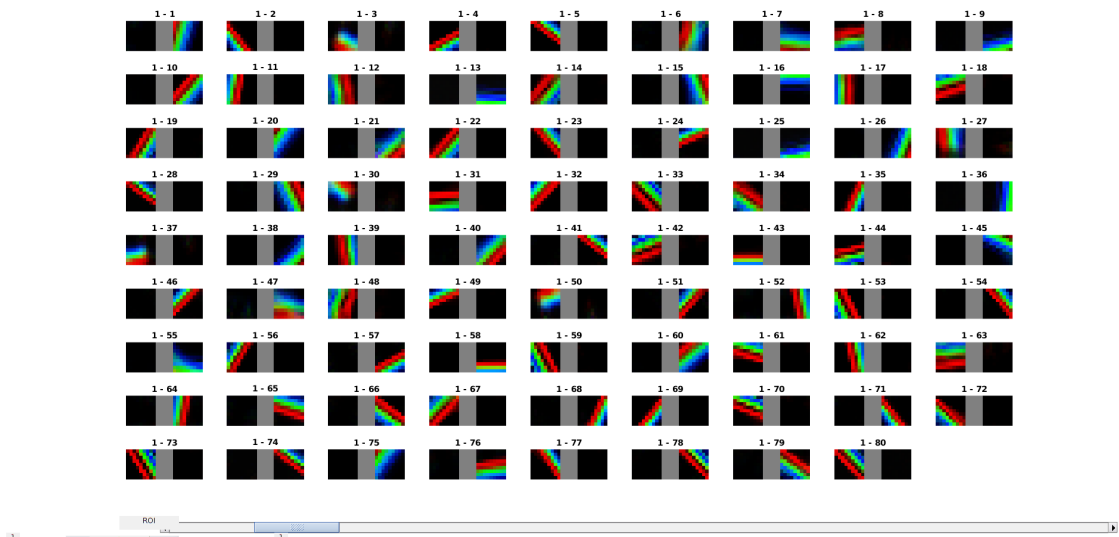


FIGURE 3.9 – Matrice de poids après entraînement des 64 filtres. La couleur rouge identifie les poids des synapses sans délais, le vert celles avec un délai de 2/240 s, le bleu les délais de 3/240 s. Chaque filtre est caractérisé par deux matrices, celle de gauche représente les poids ON et celle de droite les poids OFF.

En regardant la figure 3.10, panneaux A et B, on voit que les neurones répondent à des directions et vitesses spécifiques. La sélectivité à la vitesse semble cependant moins précise que

celle à la direction. En comparant les matrices de poids de la figure 3.9 et la sélectivité des neurones de la figure 3.10, on voit qu'on peut ainsi décoder la sélectivité des neurones grâce à leurs matrices de poids. Par exemple, les neurones 2 et 5 ont des sélectivités à des directions opposées. Ceci se retranscrit au niveau des matrices de poids, par des poids spatialement opposés. De même pour les neurones 5 et 14, le neurone 14 décharge pour des vitesses plutôt fortes (autour de 1.5-2 pixels), alors que le 5 pour des vitesses faibles (inférieur à 1). En regardant l'écart spatial entre les connexions sans délais (pixels rouges) et celles avec délais (vert), on voit que pour le neurone 14 l'écart est supérieur à celui du 5.

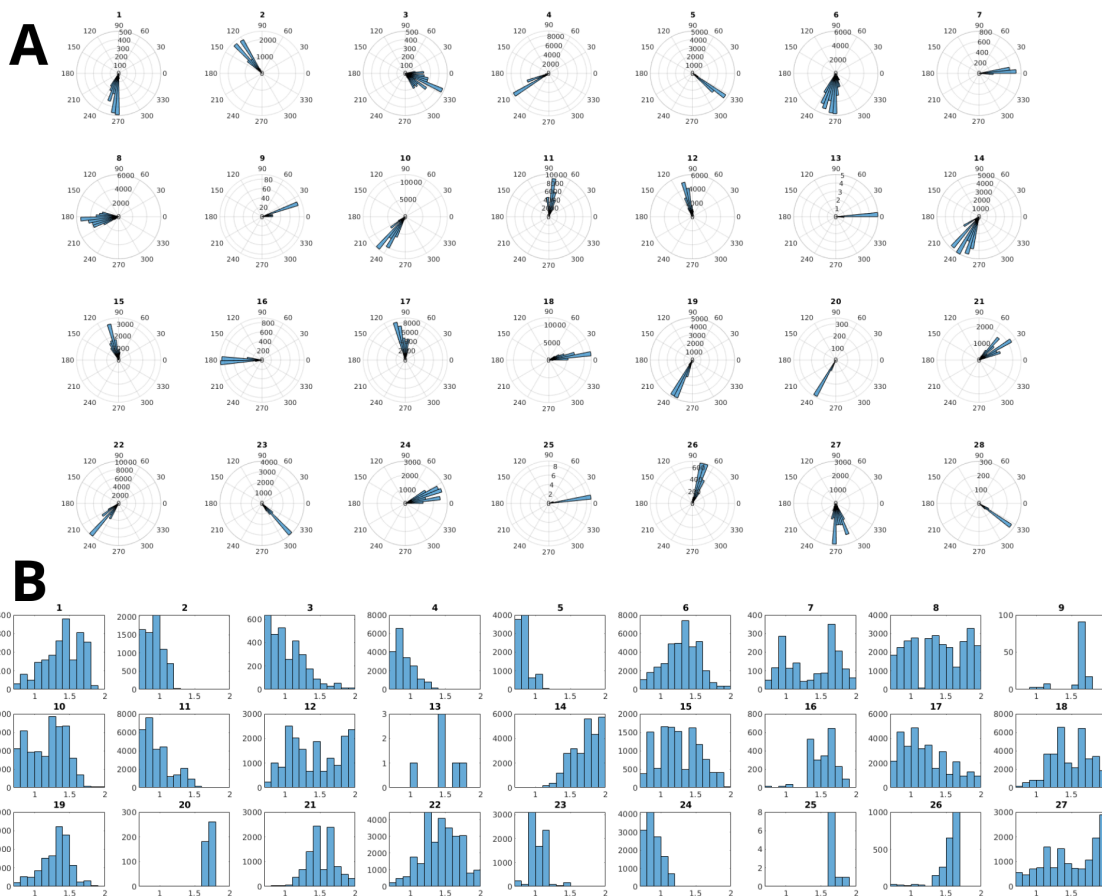


FIGURE 3.10 – Panneau A : Histogramme polaires des spikes émis par direction pour chaque filtre. Panneau B : Histogramme des spikes émis par vitesse pour chaque filtre.

### 3.2.2.2 Simulations 3D

Après avoir montré que le SNN pouvait extraire de l'information liée au mouvement, nous avons voulu appliquer cette méthode sur des stimuli plus complexes et plus proches de ceux que l'on voulait étudier, les trajectoires de balles. L'objectif étant de voir si les informations sur le mouvement extraites par le SNN permettent d'effectuer une prédiction de trajectoire de balle. Pour cela, nous avons utilisé le moteur de jeu graphique Unity<sup>1</sup>, nous permettant de facilement générer des trajectoires de balles 3D, avec des points de vues paramétrables et différents (figure 3.11, panneau A et C). Puis, de simuler ensuite le fonctionnement de la caméra NeuroSoc pour générer les événements à partir des fichiers vidéos générés (figure 3.11, panneau B).

1. <https://unity.com/fr>



Différentes simulations ont été réalisées avec différentes architectures, afin de tester le SNN selon différentes configurations. Deux différents scénarii ont été effectués. Le premier, plus simple, permettait de tester si des neurones pouvaient devenir sélectifs à des trajectoires de balles qui tombent dans une certaine zone. Le second scénario, plus complexe que le premier cherchait à tester s'il était possible de définir avec précision le point de chute (position X et Y) du ballon.

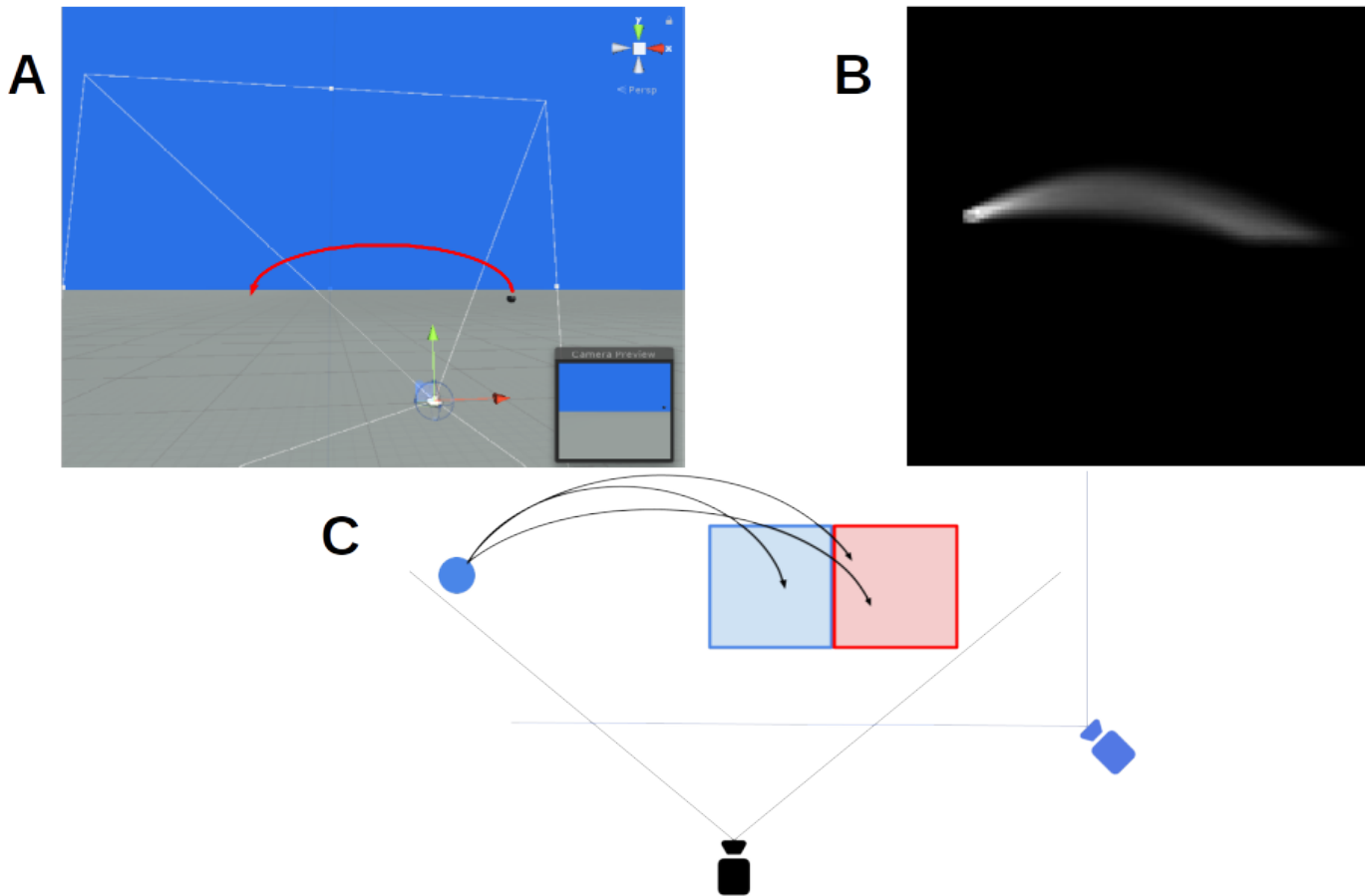


FIGURE 3.11 – Panneau A : Exemple de la simulation Unity utilisée. Panneau B : Spikes générés par les trajectoires simulées. Panneau C : Architecture de la simulation et emplacement des deux points de vues.

### 3.2.2.3 Analyse de trajectoires et catégorisation du point de chute de la balle

#### Objectifs et méthodes

Afin d'évaluer la capacité du SNN à extraire de l'information spatio-temporelle à partir des spikes générés par les trajectoires, nous avons testé le SNN dans une tâche de catégorisation du point de chute. Ainsi, les balles pouvaient atterrir dans deux zones distinctes (trajectoires courtes : zone bleue et trajectoires longues : zone rouge, figure 3.11), l'objectif était d'évaluer la capacité des neurones à devenir sélectifs aux différentes propriétés des trajectoires et si celles-ci permettaient de déterminer dans laquelle de ces deux différentes zones la balle atterrissait. Pour cela, nous avons utilisé un SNN à deux couches, la première étant une couche convolutionnelle, avec 64 filtres de taille 6x6, puis une seconde couche de type dense avec 64 neurones de sortie. La première couche convolutionnelle était entraînée avec la règle STDP, puis la seconde (dense) était entraînée avec la règle STDP ou RSTDP, permettant de classifier entre les deux

classes (trajectoires courtes et longues).

Les trajectoires sont initialisées en utilisant un vecteur initial permettant de générer le mouvement, figure 3.12. Ce vecteur permet de définir la direction et la force du mouvement. Si les neurones deviennent sélectifs à certaines propriétés du mouvement, ils devraient donc décharger pour certains vecteurs initiaux. Comme le montre la figure 3.12, une seule dimension ne permet pas de prédire la zone de chute de la balle. Il faut donc qu'un neurone devienne sélectif aux deux caractéristiques (direction et force).

## Résultats

Deux simulations ont été faites afin de déterminer si la RSTDP permettait d'avoir une sélectivité plus fine et spécialisée sur certaines propriétés des trajectoires de balles. Pour cela, nous avons regardé pour quels vecteurs initiaux chaque neurone décharge selon qu'ils soient entraînés avec la STDP ou RSTDP, afin de déterminer si la RSTDP permet une sélectivité différente de la STDP.

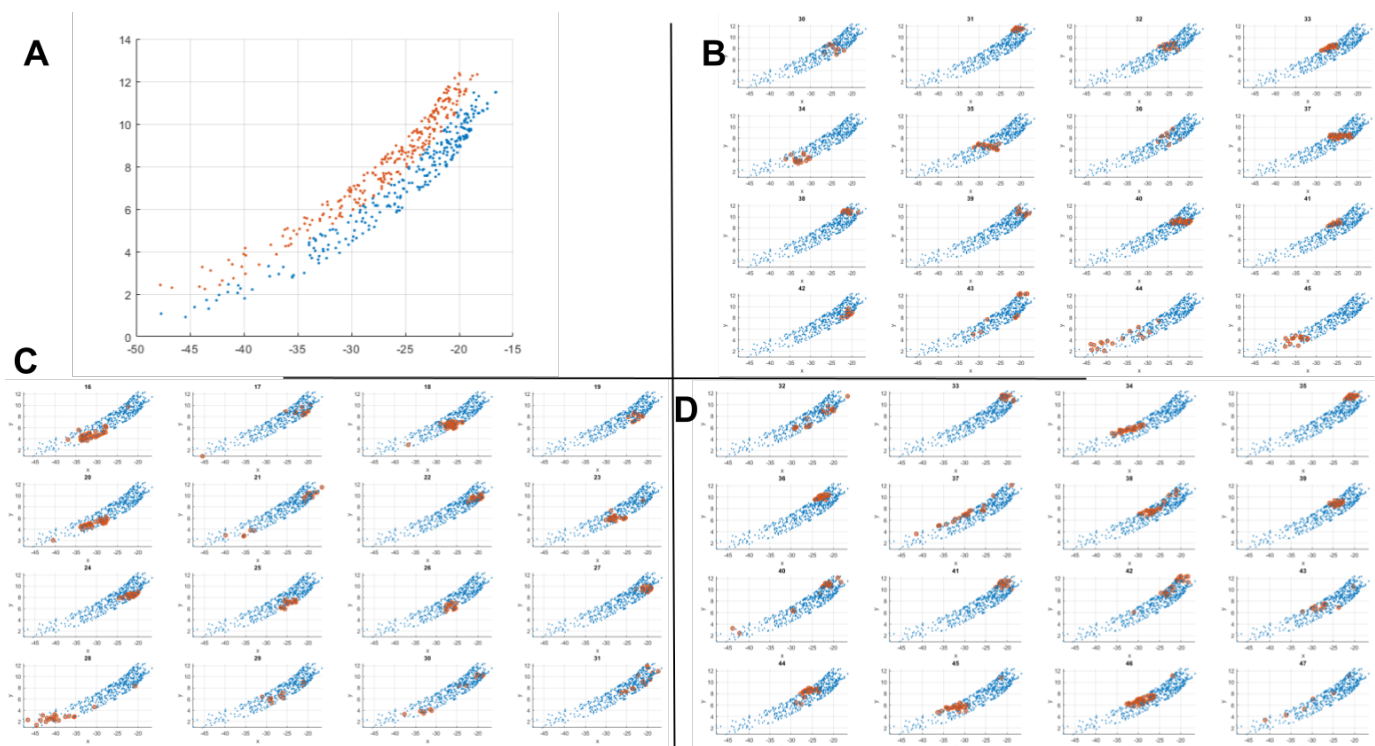


FIGURE 3.12 – Panneau A, Dimensions X et Y des vecteurs initiaux permettant de générer les trajectoires de balles. Les deux couleurs représentent les deux classes possibles. Panneau B, C et D : Les points bleus représentent tous les couples (X, Y) générés. Les cercles rouges, ceux pour qui un neurone décharge. Le panneau B, représente les couples (X,Y) pour lesquels 16 différents neurones entraînés avec la STDP déchargent. Le panneau C, représente pour 16 neurones entraînés avec la RSTDP et de la première classe (zone bleue). Le panneau D, représente pour 16 neurones entraînés avec la RSTDP et de la seconde classe (zone rouge).

Comme on peut le voir sur la figure 3.12, la règle STDP (panneau A) permet une sélectivité principalement sur la dimension Y, mais une faible sélectivité à la dimension X. Cela aboutit à ce que chaque neurone décharge pour des points de chutes très différents. Alors que la règle RSTDP (panneau B et C), conserve une bonne sélectivité sur la dimension Y, et une meilleure précision

sur la dimension X. On voit que les neurones de la première classe déchargent principalement pour les points bleus (panneau A), alors que ceux de la seconde classe déchargent pour les points rouges (panneau B). Ainsi, la règle RSTDP permet une meilleure sélectivité permettant d'avoir des neurones qui apprennent des propriétés spécifiques à leur classe. Cette sélectivité à l'une des deux classes de trajectoires est confirmée sur la figure 3.13, dans laquelle on voit que les 32 premiers neurones déchargent en grande majorité pour les trajectoires atterrissant dans la zone bleue (classe 1) et que les 32 autres (33 à 64) pour l'autre classe.

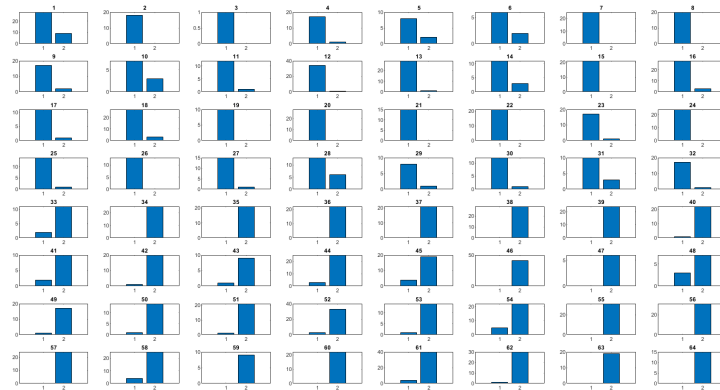


FIGURE 3.13 – Nombre de spikes émis pour les 64 neurones de la dernière couche en fonction de la classe de trajectoire.

### 3.2.2.4 Prédiction du point de chute de la balle

#### Objectifs et méthodes

L'objectif de cette simulation était de déterminer le point de chute de la balle simulée par Unity (position X et Y). Pour cela, nous avons utilisé un SNN à deux couches, la première étant une couche convolutionnelle, avec 50 filtres de taille 6x6, puis une seconde couche rétinotopique (sans partage de poids), avec 27 filtres de taille 7x7. Les deux couches étaient entraînées avec la règle STDP.

La sortie de la deuxième couche était par la suite utilisée pour prédire le point de chute de la balle simulée. Deux différents angles de vue étaient utilisés, un premier orthogonal à la trajectoire de la balle (caméra noire sur la figure 3.11, panneau C), un second placé à 45° en face des trajectoires générées (caméra bleue), afin d'évaluer l'impact du point de vue sur les performances de prédiction du SNN.

Après l'entraînement, la sélectivité des neurones de la seconde couche était évaluée en identifiant les points de chute des trajectoires pour lesquels chaque neurone déchargeait (voir figure 3.14, panneau A). Une "zone de sélectivité" était ensuite déterminée à partir de ceux-ci (polygones jaunes, fonction *boundary* de Matlab) pour effectuer les prédictions des points de chute. Une prédiction était effectuée tous les 10% de la trajectoire, afin de permettre d'avoir une évolution de la prédiction au cours du temps. Pour prédire le point de chute de la balle, les zones de sélectivité de chaque neurone ayant déchargé depuis le début de la trajectoire sont sommées et pondérées par le nombre de spikes émis depuis le début de la trajectoire afin de donner plus de poids aux derniers spikes émis (ainsi le premier neurone qui décharge a 100 fois moins d'impact que le centième). Puis, le barycentre était utilisé comme valeur prédite (figure 3.15).

## Résultats

Comme on peut le voir sur la figure 3.14, les zones de sélectivités de 27 neurones connectés à un même champ récepteur sont affichées. On y voit que les neurones déchargent pour des points de chute particuliers et précis, l'aire de la zone de sélectivité étant assez réduite pour la plupart des neurones. Ceux-ci peuvent ensuite être utilisés pour prédire le point de chute de la balle, comme on peut le voir sur la figure 3.15.

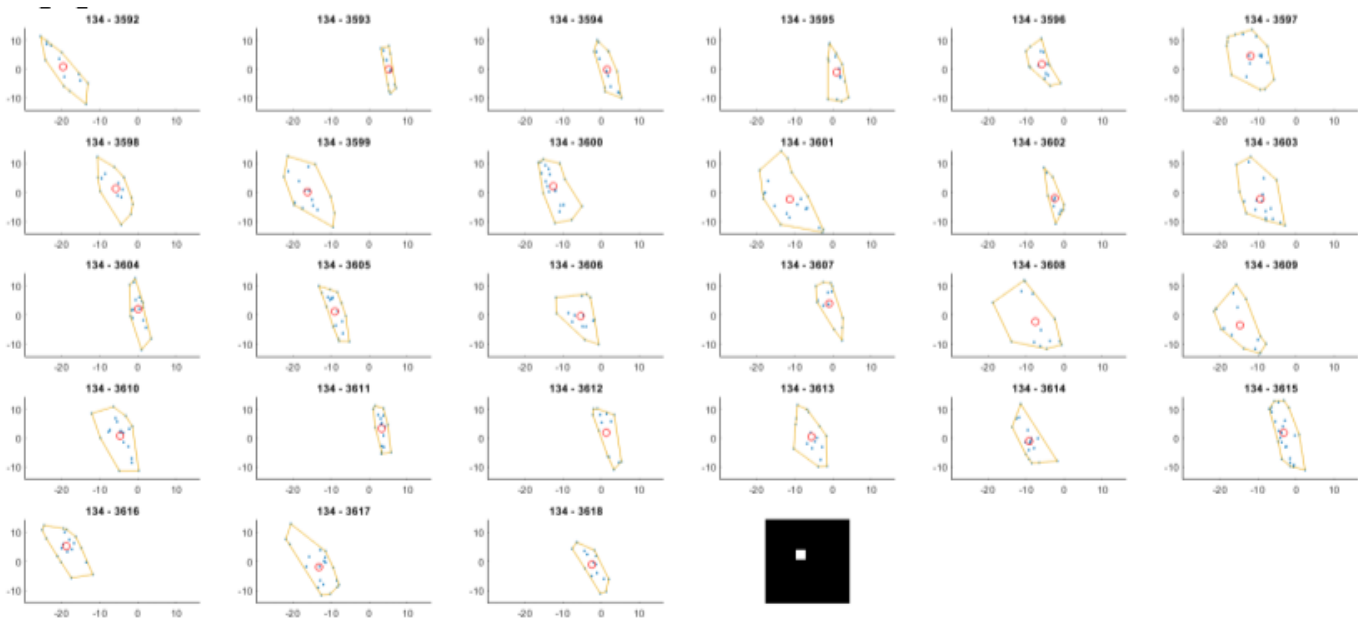


FIGURE 3.14 – Positions des points de chute pour lesquels les neurones ont déchargé (points bleus). Le polygone jaune représente la zone de sélectivité de ces neurones. Panneau B : prédiction effectuée par les neurones (croix rouge) en fonction du point réel de chute de la balle (cercle bleu), pour les deux angles de vues différents ( $45^\circ$  et orthogonal). Les traces blanches représentent la somme des zones de sélectivités de chaque neurone qui a déchargé au cours de cette trajectoire.

En utilisant la sélectivité des neurones pour prédire le point de réception, on voit que la position du barycentre s'approche du point réel de chute et donc que cette prédiction est précise et s'affine au cours du temps (figure 3.16). Selon que le point de vue soit positionné à  $45^\circ$  ou  $90^\circ$ , l'erreur de prédiction diffère. La moyenne d'erreur pour le point de vue à  $45^\circ$  est égale à 15.4 en Y et 19.1 en X. Pour le point de vue à  $90^\circ$ , l'erreur moyenne est de 10.2 en X et 21.5 en Y. On voit donc que lorsque le point de vue est de  $90^\circ$ , l'erreur en Y (information de profondeur) est bien plus grande (double) à l'erreur en X. Lorsque le point de vue est à  $45^\circ$ , l'erreur en X est supérieure à l'erreur en Y en début de trajectoire, puis légèrement inférieure en fin de trajectoire. L'erreur reste cependant supérieure à l'erreur de l'axe X, lorsque le point de vue est à  $90^\circ$ .

Lors de la simulation avec l'angle de vue à  $45^\circ$ , il arrive que des trajectoires ne déchargent pas au début des trajectoires. En utilisant ce point de vue, il y a beaucoup plus de mouvement de profondeur, la balle s'approche de la caméra. Ce type de mouvement génère beaucoup moins d'événements, car entre deux images la variation de luminosité par pixel liée au mouvement est très faible, plus particulièrement lorsque la balle est éloignée de la caméra (proche de sa position initiale).

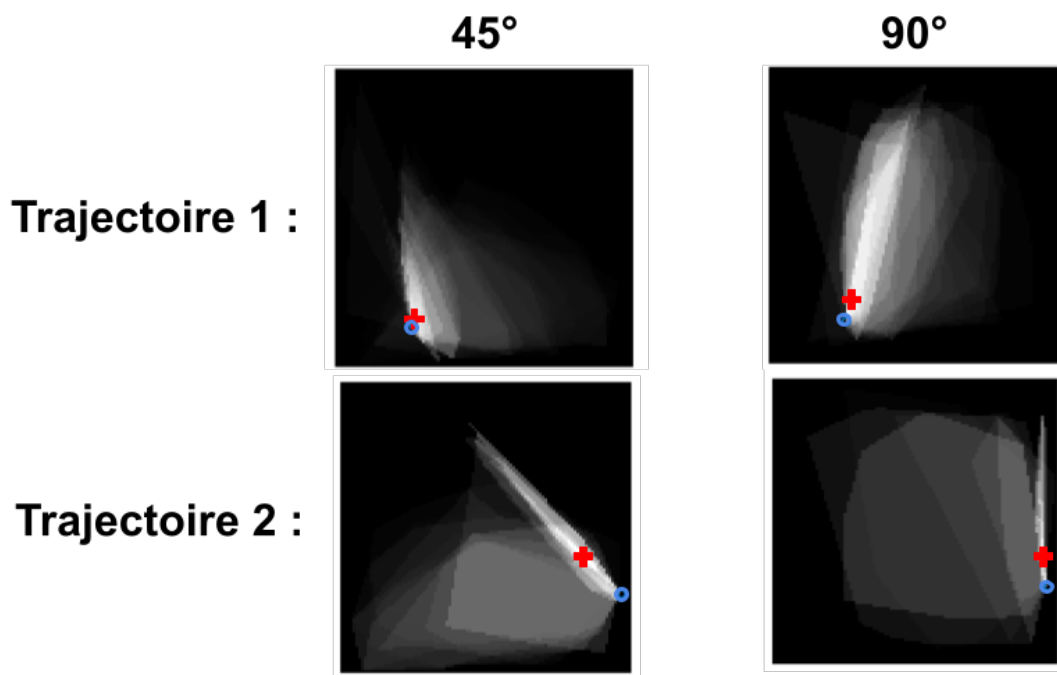


FIGURE 3.15 – Prédiction effectuée par les neurones (croix rouge) en fonction du point réel de chute de la balle (cercle bleu), pour les deux angles de vues différents ( $45^\circ$  et orthogonal). Les traces blanches représentent la somme des zones de sélectivités de chaque neurone qui a déchargé au cours de cette trajectoire.

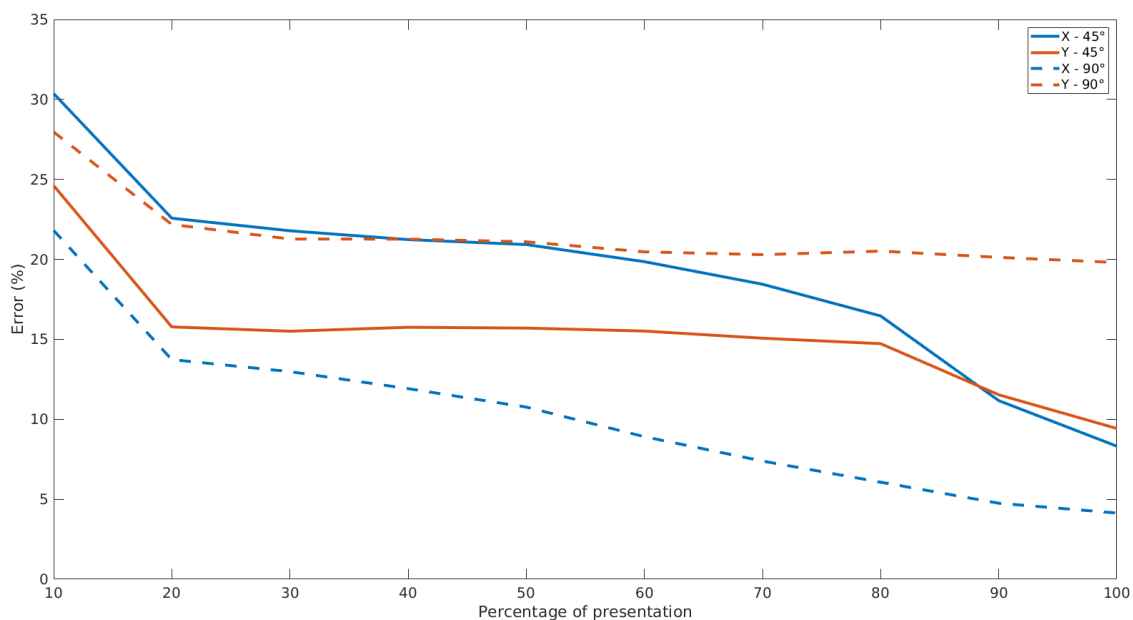


FIGURE 3.16 – Évolution de l'erreur de prédiction en fonction du temps de présentation de la trajectoire de la balle et de l'angle de vue.

---

### 3.2.3 Conclusion

Ces travaux avaient pour but d'effectuer de premiers tests d'un SNN pour faire de l'analyse de mouvement et de la prédiction de trajectoires, ainsi que pour tester s'il était possible d'extraire de l'information provenant de la caméra NeuroSoc. Nous avons ainsi pu montrer que le flux de sortie de la caméra NeuroSoc pouvait être utilisé pour extraire de l'information visuelle sur des données complexes (voitures de diverses formes, camions, etc.) et qu'un SNN avec des délais, des inhibitions latérales et une règle STDP permettait aux neurones de devenir sélectifs à des patterns spatio-temporels de spikes permettant de discriminer la direction des véhicules. Ces résultats ont ensuite été confirmés avec les simulations 2D et 3D, qui ont montré dans un premier temps que les neurones d'un SNN pouvaient avoir une sélectivité fine à la direction et une sélectivité plus grossière à la direction. Puis, les simulations 3D ont étendu ces résultats à des trajectoires plus complexes et ont montré qu'un SNN permettait de prédire le point de chute d'une balle avec précision. Le mouvement en profondeur (balle qui s'approche ou s'éloigne d'une caméra) ne génère que peu d'événements, notamment lorsque la balle est encore éloignée de la caméra et beaucoup plus lorsque celle-ci est proche. Cette haute variabilité du nombre de spikes et le peu de spikes générés en début de trajectoire entraîne ainsi une difficulté pour les neurones du SNN de décharger et donc de décoder de l'information. Ainsi, une caméra placée à  $90^\circ$  semble plus adaptée comparée à une caméra placée à  $45^\circ$ , de plus elle permet une meilleure prédiction du point de chute sur l'axe X, mais une plus mauvaise précision sur l'axe Y (information de profondeur) qui nécessiterait une information binoculaire pour être mieux évaluée. Ces travaux nous ont permis ainsi de tester et valider notre méthode. Il reste maintenant à la tester sur des acquisitions réelles afin de la valider dans des situations plus contraignantes.



# Chapitre 4

## Etude 1 : Predictions d'échanges de passes

### 4.1 Objectifs de l'étude

Après avoir montré que notre SNN pouvait extraire des informations sur le mouvement en utilisant des stimuli simulés et contrôlés et que la sortie de la caméra NeuroSoc permettait d'extraire de l'information sur le mouvement, nous voulions tester notre solution sur des situations plus proches de notre objectif final. Cette étude a donc pour ambition de valider les résultats précédents en situation réelle et de faire l'acquisition de trajectoires proches de celles que l'on pourrait trouver dans le domaine du sport. Avant de tester notre solution sur des scènes sportives réelles (services de tennis, lancer franc de basket, penalty au foot, par exemple), nous avons décidé d'effectuer des tests sur des trajectoires plus contrôlées. De plus, ceci nous a permis aussi de mieux évaluer la précision de notre solution en quantifiant la sélectivité au mouvement des filtres, grâce à la technologie Vicon (cf. Section 4.2.3). Ce chapitre a donné lieu à une publication en Mai 2021 dans le journal *Frontiers in computational neuroscience* (DEBAT et al. 2021).

### 4.2 Méthodes

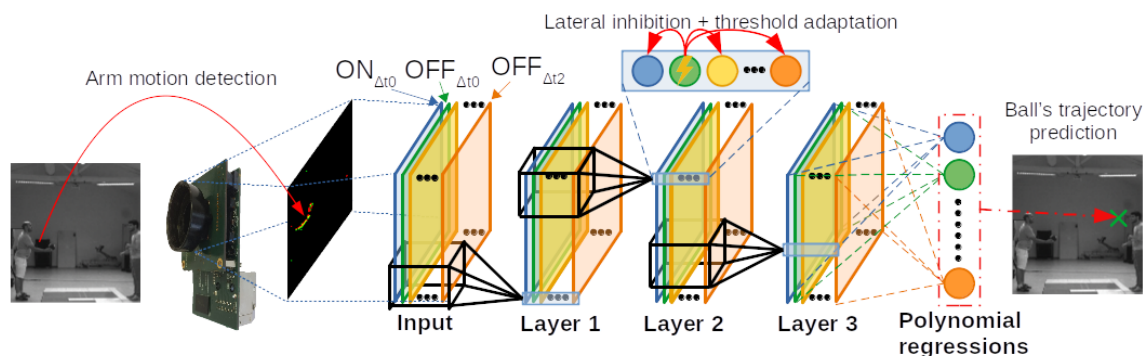


FIGURE 4.1 – Architecture utilisée



---

### 4.2.1 SNN

Lors de cette étude nous avons utilisé l'architecture présentée dans la figure 4.1 composée d'un CSNN à trois couches composé de neurones intégrateurs à fuite (3.1). Le SNN reçoit les événements générés par la caméra NeuroSoc, ceux-ci sont dupliqués et des délais sont ajoutés afin d'améliorer la sélectivité du SNN. Les mécanismes d'inhibition et d'adaptation de seuil étaient activés. Un système de partage de poids était mis en place, chaque neurone était connecté à  $5 \times 5 \times d$  neurones de la couche précédente (section 3.1.2). Les poids synaptiques étaient entraînés en utilisant la règle STDP décrite par l'équation (3.4).

### 4.2.2 Acquisition de données

Nous avons enregistré avec la caméra NeuroSoc un échange de 297 passes entre deux personnes espacées de 2,30 mètres. La caméra était placée perpendiculairement aux trajectoires et éloignées de manière à avoir les deux participants à chaque extrémité de l'image. Ces échanges ont ensuite été labellisés à la main afin d'en déterminer le début, la fin, la direction (droite ou gauche) et la position finale de la balle pour chaque échange. Ce jeu de données a ensuite été divisé en deux jeux de données entraînement/test sélectionnés aléatoirement, avec 208 (70%) trajectoires pour le jeu d'entraînement et 89 (30%) pour le test.

### 4.2.3 Vicon

Afin d'évaluer les performances et la précision de notre réseau de neurones sur une tâche d'analyse du mouvement, nous avons utilisé la technologie Vicon<sup>1</sup> disponible au CREPS de Toulouse. Le Vicon est utilisé pour faire de la capture du mouvement, il utilise des flash lumineux infra-rouge pour illuminer des marqueurs réfléchissants et détecter leurs positions 3D avec précision, à une fréquence de 200Hz. Nous avons donc disposé 4 capteurs diamétralement opposés sur le ballon, nous permettant ainsi d'obtenir sa position centrale en moyennant ces 4 positions (figure 4.2), puis de déterminer la direction et la vitesse de la balle. Un filtre passe-bas avec une fréquence de coupure à 6Hz a été appliqué sur les données générées par le Vicon afin de réduire le bruit.

Cependant les flash lumineux infra-rouge utilisés par le Vicon étaient perçus par la caméra générant ainsi de fortes variations de luminosités entre deux images et donc de nombreux événements. Nous avons donc ajouté un filtre anti-IR à la caméra (*SP730 Near-IR/Colorless Dichroic Block Shortpass Filter* 2015) afin de couper et atténuer les bruits émis par le Vicon.

---

1. <https://www.vicon.com/>

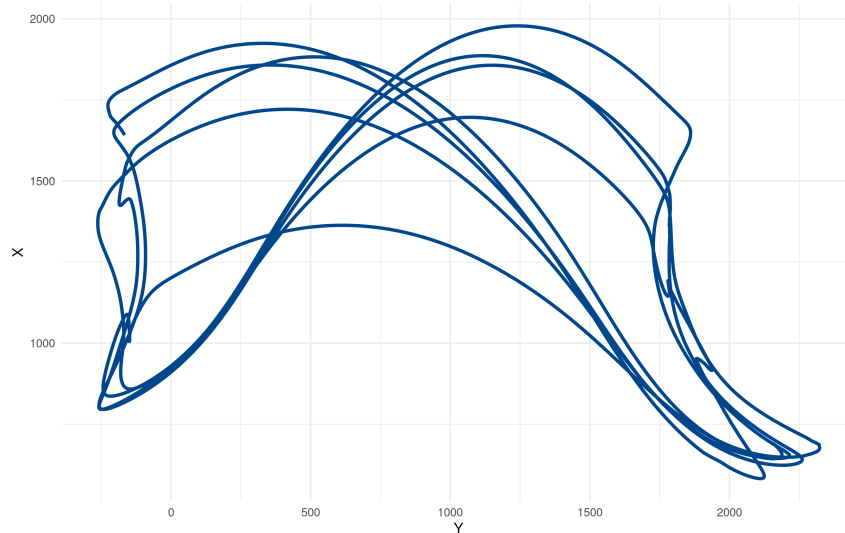


FIGURE 4.2 – Positions spatiales de quelques trajectoires du ballon générées par le Vicon

#### 4.2.4 Prédictions

L'objectif de cette étude est de prédire le point de réception du ballon en utilisant les spikes en sortie de la dernière couche du SNN. La prédiction est divisée en deux valeurs distinctes : la première est la direction, la valeur  $X_{Pred}$ , est-ce une trajectoire qui va de la droite vers la gauche ou l'opposé ; la seconde est la hauteur du point de réception, la valeur  $Y_{Pred}$ . Cette deuxième valeur pouvant être fortement impactée par la décision de la personne recevant le ballon (est-ce que cette personne avance ses mains pour aller chercher le ballon ou l'attend ?), nous avons donc choisi de la déterminer lorsque le ballon passe une certaine position (lorsque le ballon traverse la ligne bleu/marron pour une direction qui va vers la gauche/droite, figure 4.3, panel B). Les prédictions étaient faites à différents moments de la trajectoire (15, 30, 45, 60, 75 et 90%) afin d'évaluer l'évolution de cette prédiction en fonction de la quantité d'information présentée.

La valeur  $X_{Pred}$  est beaucoup plus simplifiée que la valeur  $Y_{Pred}$ , seulement deux valeurs sont possibles (0 ou 1 pour gauche ou droite). Une valeur  $X_{Pred_f}$  est définie en calculant le ratio de spike émis par direction pour un filtre  $n$ . Ainsi, un filtre qui a une valeur  $X_{Pred_f}$  proche de 1 encode pour des trajectoires allant de la gauche vers la droite. La direction prédite est ensuite calculée en faisant la moyenne des  $X_{Pred_f}$  pour chaque spike émis.

La valeur  $Y$  correspond à la hauteur lors de la réception. Pour cela nous définissons pour chaque filtre  $n$  une RP du second degré pour décoder la sortie du SNN et faire la prédiction (équation 4.1). Ce décodeur est suffisamment simple pour s'assurer que le traitement de l'information est principalement fait par le SNN. En effet, si le SNN n'extrait aucune information sur le mouvement (SNN non-entraîné par exemple) ou si le RP est appliqué directement en sortie, il ne pourra pas prédire avec précision le point de réception du ballon (cf figure 4.8). Nous souhaitons ainsi nous assurer que la RP ne fait qu'interpréter l'information de sortie du réseau de neurones et que sans le réseau une prédiction précise ne serait pas possible.

$$Y_{Pred_f} = a_{00} + a_{10} \times X + a_{01} \times Y + a_{20} \times X^2 + a_{02} \times Y^2 + a_{11} \times X \times Y \quad (4.1)$$

Avec :

- $X$  et  $Y$  : la position spatiale du neurone qui décharge (cf, figure 4.8)

---

—  $a_{xx}$  : poids de la régression polynomiale

La direction prédite de la balle  $X_{Pred}$  n'est pas prise en compte lors de la RP. Les filtres encodent principalement pour des directions spécifiques, elle est donc indépendante de la RP. L'erreur quadratique moyenne (RMSE) de chaque RP est ensuite calculée afin d'évaluer la fiabilité de chaque RP.

Deux vecteurs  $Sc_{Pred}$  d'une longueur égale à 120 (hauteur des images) sont définis et contiennent les scores, un pour chaque direction. Les prédictions effectuées à la fin de la trajectoire sont plus fiables que les prédictions précédentes. En effet, il est plus difficile de faire une prédiction au début de la trajectoire, lorsque le ballon est encore dans la main du lanceur, que quelques millisecondes avant l'arrivée de la balle sur la ligne d'estimation. Nous donnons alors plus d'impact aux dernières prédictions en ajoutant une fuite (similaire au neurones intégrateur à fuite) au vecteur de score  $Sc_{Pred}$ , comme le montre l'équation 4.2. Les scores sont mis à jour pour chaque spike. La prédiction est faite en fonction du filtre et de la position du neurone qui décharge. Le filtre  $f$  détermine quel RP est utilisé, puis une fonction triangulaire centrée en  $Y_{Pred_f}$  (équation 4.1) est utilisée pour déterminer le score ajouté. La pente de cette fonction et donc la plage de valeur impactée est déterminée par la RMSE de la RP, l'aire sous la courbe reste toujours égale à 1, comme on peut le voir sur la figure 4.3, panneaux A et C. Ces valeurs sont ensuite divisées par la distance moyenne à la balle du filtre  $D_n$ , afin de donner plus de poids aux filtres codant pour le mouvement du ballon qu'aux autres codant pour le bras du receveur, par exemple. Ces valeurs sont ensuite ajoutées au vecteur correspondant à la direction prédite. L'indice de la valeur maximale est finalement sélectionné comme valeur prédite.

$$Sc_{Pred_d} = Sc_{Pred_d} \times \exp(-\Delta t/\tau) \quad (4.2)$$

## 4.2.5 Entraînement

Comme pour les autres tests effectués, le réseau de neurones a été entraîné couche par couche de manière non-supervisée avec la règle d'apprentissage STDP, en lui présentant le jeu de données d'entraînement. Chaque trajectoire est espacée d'une durée de deux secondes de la précédente permettant ainsi au réseau de neurones de revenir à son état initial entre deux trajectoires. La sortie de la troisième couche a ensuite été utilisée pour effectuer les prédictions, comme détaillé dans la section 4.2.4. Pour chaque filtre convolutif ( $N_f = 100$  pour la troisième couche), une RP est effectuée avec la position spatiale (x et y) du neurone qui décharge comme entrée de la régression polynomiale et la position verticale de la balle comme variable prédite. Un SNN avec une règle d'apprentissage STDP a de nombreux paramètres qui doivent être réglés (constante de temps du potentiel de membrane, ratio LTP/LTD, taux d'apprentissage, ...). Nous avons estimé ces paramètres à l'aide d'un algorithme génétique (MOHAMMADI et al. 2017). Une version simplifiée du processus de prédiction présenté dans la section 4.2.4 a été utilisée. La fonction de coût pour la minimisation était la valeur moyenne de l'erreur de prédiction moyenne à 15, 30, 45, ..., 90% de la présentation de la trajectoire.

## 4.2.6 Performances chez l'humain

Afin de comparer les performances de notre système aux capacités humaines, nous avons réalisé une expérience au cours de laquelle 12 participants (âge moyen =  $27 \pm 12,9$ ) devaient prédire le point de réception du ballon à différents pas de temps, comme ce qui est demandé au SNN. La caméra NeuroSoc permet d'enregistrer la vidéo classique (frame-based) à une fréquence de 240 FPS en parallèle de l'extraction et la transmission des spikes. Les trajectoires présentées à ces participants étaient donc les mêmes que celles utilisées pour tester notre SNN. Tous les participants avaient une vision normale ou corrigée à la normale, et étaient en bonne

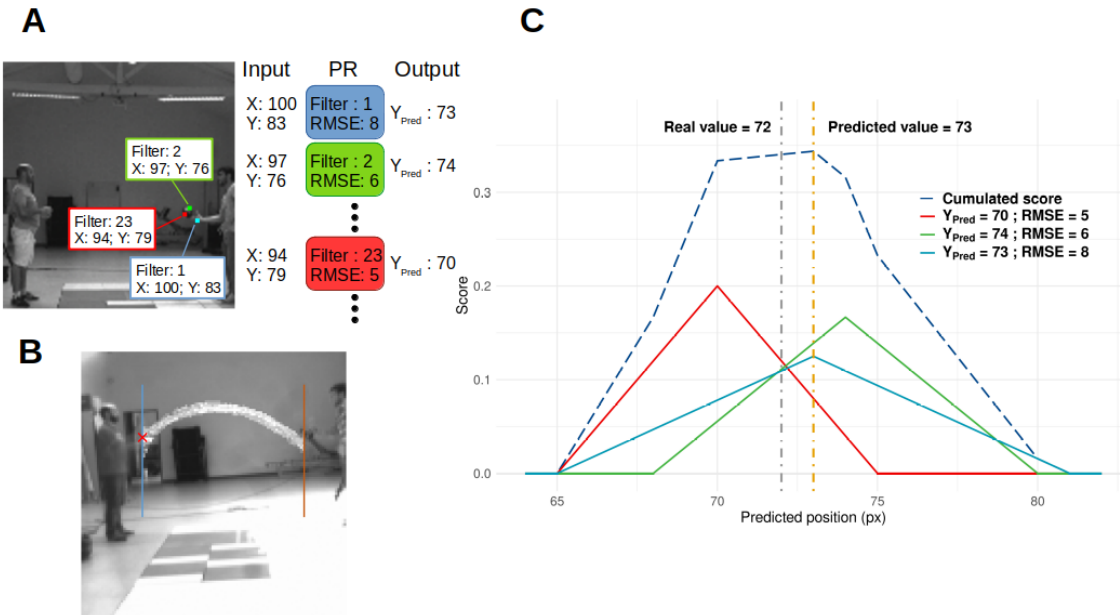


FIGURE 4.3 – Panneau A : Processus de décodage. Lorsqu'un neurone spike, sa position spatiale est utilisée comme entrée pour la RP liée au filtre du neurone en question. Dans cette figure, trois différents neurones de positions et de filtres différents déchargent et génèrent trois prédictions. Celles-ci sont ensuite intégrées pour calculer une valeur de score, comme présenté dans le panneau C. Panneau B : Positions horizontales utilisées comme points de réception. Panneau C : Exemple de mise à jour du score. Les lignes épaisses représentent le score ajouté par trois neurones provenant de différents filtres, sur la base de l'équation (4.1). On suppose que ces trois neurones déchargent en même temps et qu'il n'y a donc pas de fuite du score. La ligne bleue pointillée représente le score cumulé. Les lignes pointillées grises et orange représentent respectivement la valeur réelle attendue et la valeur prédite.

---

santé et sans aucune anomalie oculomotrice connue. Les participants étaient naïfs quant à l'objectif de l'expérience, qui a reçu l'autorisation éthique appropriée du "Comité d'éthique de la Recherche" de l'Université Fédérale de Toulouse (accord 2020-279). La taille de l'échantillon a été déterminée à l'aide de G\*Power (FAUL et al. 2007) après avoir analysé les résultats d'une expérience précédente portant sur l'influence de la durée de présentation sur les performances d'anticipation. Les résultats ont montré que pour une puissance souhaitée de .90, une taille d'échantillon totale de 12 participants était nécessaire. Seule une partie des trajectoires (c'est-à-dire les premiers 15, 30, 45, 60, 75 et 90%) était présentée aux participants qui devaient prédire le point de réception en cliquant avec la souris sur le point final anticipé, sans aucune contrainte temporelle. L'expérience était divisée en trois parties :

- Pré-apprentissage : les vidéos sont présentées aux participants pour la première fois.
- Apprentissage actif : après chaque prédiction faite par le participant, le point d'arrivée exact était affiché à l'écran pour fournir l'erreur de prédiction à l'utilisateur.
- Post-apprentissage : même procédure que le pré-apprentissage (aucune erreur n'est affichée) mais les sujets ont bénéficié de l'expérience de la phase d'apprentissage actif.

Pour chaque condition, 8 trajectoires ont été montrées pour chaque pourcentage de temps de présentation, pour un total de 48 essais dans chaque condition. Chaque vidéo n'est présentée que dans une seule catégorie. Les vidéos d'une dimension de  $384 \times 360$  pixels étaient présentées sur un écran de 13.3 pouces (60Hz, pleine résolution  $1366 \times 768$ , dimension  $29.5 \times 17$  cm en horizontal par vertical). Contrairement à notre SNN, les participants humains avaient déjà une expérience du mouvement de la balle ou du mouvement en général. Nous avons néanmoins inclus une phase d'apprentissage actif dans l'expérience afin que les participants puissent s'adapter à ses spécificités (résolution d'image, taille d'écran, etc.).

## 4.3 Résultats

Le premier objectif de cette étude est de confirmer les résultats présentés dans la section 3.2 et donc d'évaluer la pertinence d'utiliser un SNN pour extraire des informations sur le mouvement à partir des spikes générés par la NeuroSoc. Pour cela, nous avons utilisé l'information retournée par le Vicon pour analyser si une fois entraînés, les filtres codent pour certaines spécificités du mouvement du ballon (directions, vitesse). Le second objectif était d'utiliser cette information extraite par les filtres pour prédire la trajectoire du ballon.

Le SNN est entraîné sur le jeu de d'entraînement puis évalué sur le jeu de test. Six différentes simulations sont effectuées, avec des poids initiaux aléatoires différents, puis les résultats sont ensuite moyennés sur ces six simulations.

### 4.3.1 Sélectivités

#### 4.3.1.1 Direction

La sélectivité à la direction est évaluée pour un filtre  $n$  en comptant le nombre de spike émis pour chaque direction générée par le Vicon (arrondi à l'entier). Ce nombre de spikes est ensuite divisé par le nombre d'occurrences de ces directions durant la présentation des trajectoires, générant un vecteur  $\theta f_n$ . Un exemple de ces valeurs pour 4 différents filtres est présenté dans la figure 4.4, panneau A. On peut ainsi y voir que les filtres codent pour des directions spécifiques et différentes les unes des autres, démontrant une forte sélectivité à la direction. Le panneau B permet d'avoir une vision globale de cette sélectivité, les  $\theta f_n$  sont cumulés, tout en étant centrés sur leur direction favorite  $\theta c_n$  (la valeur moyenne de  $\theta f_n$ ). On peut ensuite voir dans le panneau C que les filtres codent pour des directions favorites  $\theta c_n$  différentes. On peut voir qu'il y a plus de filtres qui codent pour des trajectoires montantes (autour de 40 et 320 degrés). En

effet, ces directions sont plus présentes statistiquement en entrée, car les phases de trajectoires redescendantes sont mineures et les phases de lancers (lorsque le ballon est encore dans les mains du lanceur) sont des trajectoires montantes qui génèrent de nombreux spikes.

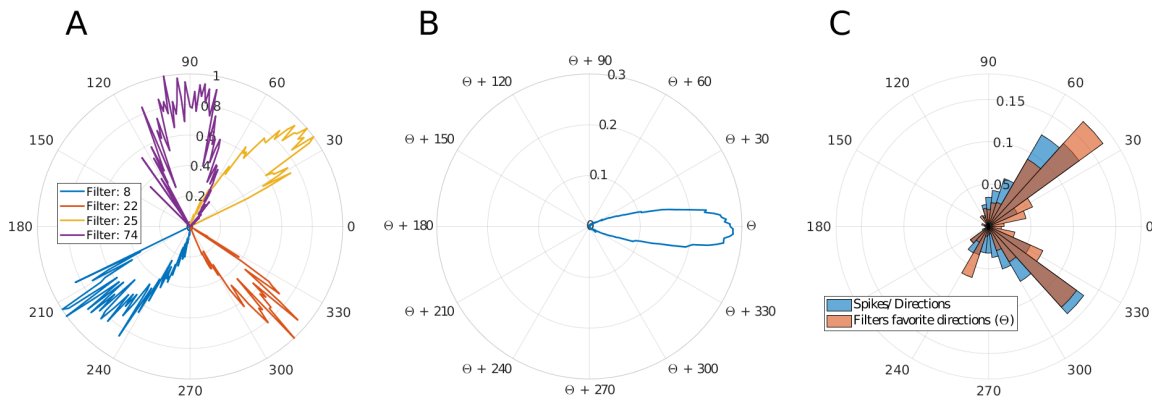


FIGURE 4.4 – Sélectivité à la direction. Panneau A : Sélectivité normalisée  $\theta f_n$  de quatre filtres différents. Panneau B : Sélectivité moyenne de tous les filtres centrée sur leur direction favorite  $\theta_{c_n}$ . Panneau C : Distribution des directions favorite  $\theta_{c_n}$  et du nombre d'événements générés par la caméra NeuroSoc par direction.

#### 4.3.1.2 Vitesse

Une méthode similaire est utilisée pour évaluer la sélectivité à la vitesse.  $Sf_n$  est déterminé en comptant le nombre de spikes pour chaque vitesse. La vitesse du ballon est très fortement liée à sa direction (figure 4.6, panneau A). En effet, les vitesses maximales sont atteintes lors des directions ascendantes car ce sont des directions présentes en initiation de trajectoires, par exemple.

Ainsi, pour évaluer la sélectivité à la vitesse indépendamment de la sélectivité à la direction du filtre,  $Sf_n$  est comparé avec une distribution de vitesse tirée aléatoirement  $Sf_{Rand_n}$ .

Pour chaque direction pour laquelle un filtre  $n$  émet un spike, nous avons sélectionné un nombre de valeurs de vitesse proportionnel au nombre de spikes pour cette direction. Ces valeurs sont tirées aléatoirement parmi toutes les vitesses de balle ayant des directions similaires à la sélectivité du filtre  $n$ . Cela donne ainsi une distribution de vitesses corrélée à la sélectivité de la direction de ce filtre, comme illustré dans la figure 4.5 ci-dessous. Pour chaque filtre  $n$ , nous avons également évalué  $\omega_n$  (écart-type de  $Sf_n$ ) et  $\omega_{Rand_n}$  (écart-type de  $Sf_{Rand_n}$ ). Nous calculons ensuite un score de sélectivité de vitesse  $\Omega_n$ , le ratio entre  $\omega_n$  et  $\omega_{Rand_n}$  (équation 4.3). Puis, nous évaluons  $\Omega_{All}$ , la moyenne de  $\Omega_n$  pour tous les filtres  $n$ , pondérée par la quantité de spikes émis par filtre (équation 4.3). Nous avons obtenu un  $\Omega_{All}$  de 0.38, significativement en dessous d'une valeur de 1. La valeur de  $\omega_n$  est donc inférieure à celle de  $\omega_{Rand_n}$ , mettant en évidence une sélectivité pour la vitesse.

$$\Omega_n = \frac{\omega_n}{\omega_{Rand_n}} \tag{4.3}$$

$$\Omega_{All} = \frac{\sum \Omega_n \times C_n}{\sum C_n}$$

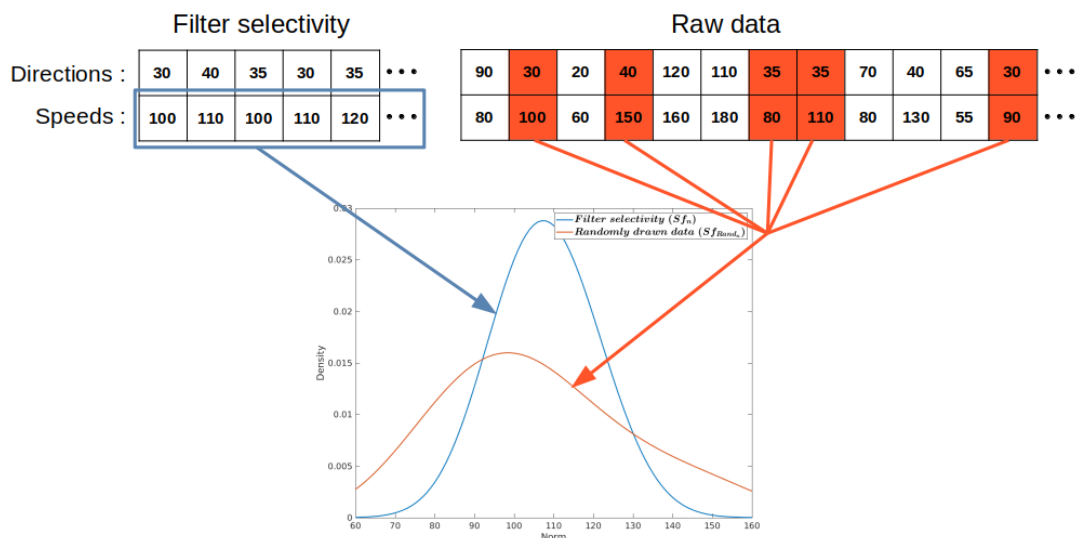


FIGURE 4.5 – Processus de sélection de vitesses aléatoires basé sur la sélectivité de direction du filtre

Si le filtre n'est pas sélectif à la direction il devrait alors avoir une distribution  $Sf_n$  proche de  $Sf_{Rand_n}$ . La figure 4.6, panneau B, montre la sélectivité à la vitesse de trois filtres différents avec des sélectivités à la direction  $\theta f_n$  proches. On peut dans un premier temps voir que la variabilité de  $Sf_n$  est bien inférieure à celle de  $Sf_{Rand_n}$  démontrant que ces filtres codent pour des vitesses spécifiques, indépendamment de leur sélectivité à la vitesse. De plus, ces filtres codent pour des vitesses différentes. En effet, les trois filtres ont des  $Sf_n$  centrés en des valeurs différentes.

Ces résultats confirment donc de précédents résultats obtenus avec un SNN proche de celui utilisé et une DVS (PAREDES-VALLES, SCHEPER et CROON 2019). Le SNN entraîné de manière non-supervisée réussit à trouver des coïncidences dans le stimulus qui lui est présenté et permet d'avoir des filtres sélectifs à des caractéristiques spatio-temporelles du mouvement. Cette sélectivité devrait ensuite nous permettre de prédire la trajectoire de la balle avec précision.

### 4.3.2 Motion Tracking

La sélectivité aux mouvements des filtres mis en-avant ci-dessus devrait nous permettre de discriminer et donc suivre des mouvements spécifiques en fonction de leur vitesse, de leur direction, de leur forme, etc... Il y a différents types de mouvements dans les stimuli enregistrés avec la NeuroSoc. Principalement la balle et les bras qui peuvent être divisés en plusieurs parties (mains, avant-bras, ...). Une des méthodes pour évaluer la capacité de notre réseau à suivre des mouvements spécifiques serait de les labelliser tous un par un, mais ceci serait chronophage. Nous nous sommes donc dans un premier temps intéressés si des filtres déchargent pour des mouvements liés à la balle. Pour cela, nous calculons la distance entre le centre du ballon et la position du neurone qui décharge. Comme la caméra faisait face au plan de la trajectoire de la balle, il existe une relation linéaire entre la position de la balle dans le référentiel de la caméra et celui du Vicon. Ainsi, étant donnée la position de la balle dans le cadre de référence du Vicon, nous pouvons facilement calculer sa position sur le cadre et calculer la distance entre un pic donné et le centre de la balle.

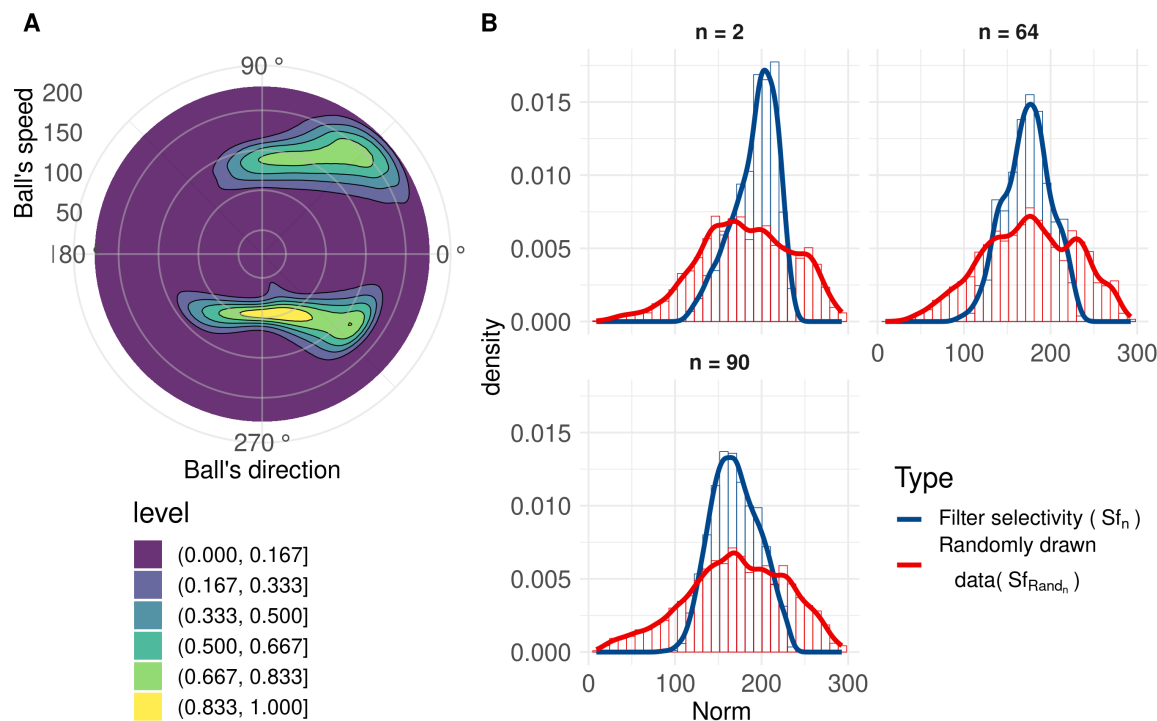


FIGURE 4.6 – Sélectivité à la vitesse. Panneau A : Densité 2D normalisée de la vitesse de la balle par direction. Les vitesses et les directions sont corrélées : par exemple, les mouvements ascendants sont en moyenne plus rapides (initiation du lancer, direction proche de 45 ou 315 degrés) que les mouvements horizontaux (haut de la trajectoire, direction proche de 90 ou 270 degrés). Panneau B : Gamme de vitesses qui génèrent un spike (ligne bleue) pour trois filtres avec une sélectivité similaire à la direction du mouvement. Cette gamme de vitesses est comparée à la gamme de vitesses sélectionnées en fonction de la sélectivité à la direction de ces mêmes filtres (ligne rouge). Ces filtres ont des sélectivités de direction similaires, ce qui explique la similarité des lignes rouges. Nous pouvons observer que ces filtres produisent des spikes pour une plus petite gamme de vitesses ( $Sf_n$ ) par rapport à la distribution tirée aléatoirement ( $Sf_{Rand_n}$ ).

Ensuite, en regardant la distance moyenne  $D_n$  entre la balle et tous les spikes émis par le filtre  $n$ , nous pouvons déterminer si ce filtre code pour le mouvement de la balle (si  $D_n$  est très faible) ou une autre caractéristique comme une partie d'un bras. Puisqu'un neurone est déterminé par sa position et son filtre, qui code ou non le mouvement de la balle (en fonction de  $D_n$ ), nous devrions être en mesure de suivre le mouvement de la balle sur la base de la sortie de ce réseau. Par exemple, dans le contexte de la figure 4.3, panneau A, nous pouvons nous attendre à ce que les filtres 2 et 23 pourraient être utilisés pour suivre le mouvement de la balle et le filtre 1 pour le mouvement de la main.

De nombreuses études ont montré que les SNNs équipés de STDP développent une sélectivité à la forme évoluant au fur et à mesure que l'on avance dans la hiérarchie de couches (comme les réseaux neuronaux plus conventionnels) (S.-R. KHERADPISHEH et al. 2018 ; THIELE, BICHLER et DUPRET 2018 ; MASQUELIER et S. J. THORPE 2007). Les neurones des premières couches sont sélectifs aux bords tandis que les neurones des couches plus profondes sont sélectifs aux caractéristiques plus complexes. Dans le contexte de notre SNN, les filtres dont la distance moyenne par rapport au ballon est inférieure à 6 pixels (barres bleues de la figure 4.7, panneau



A) encodent principalement des caractéristiques liées au ballon (y compris l'avant-bras du lanceur lorsque le ballon est encore dans ses mains). Les filtres dont la valeur est plus élevée (barres rouges de la figure 4.7, panneau A) codent pour des caractéristiques différentes, comme d'autres parties du bras, le receveur, etc. La distribution de  $D_n$  met en évidence la capacité des filtres à coder pour de mouvements spécifiques, tels que le mouvement de la balle (voir figure 4.7, panneau A). Comme le montre la figure 4.7, panneau B, nous pouvons voir la position des spikes pour 4 filtres différents. Chaque filtre génère des spikes pour des positions spécifiques et des motifs de mouvement différents. Le filtre numéro 37 (en rouge) est sélectif pour des mouvements de balle ascendants et allant vers la gauche de la balle, à l'opposé du filtre 5 (en jaune) qui est sélectif pour des mouvements descendants et vers la droite de la balle. Le filtre 87 (en bleu) est sélectif pour le bras droit du lanceur durant la phase de lancer (c'est-à-dire, quand le bras monte) et le filtre 10 (en vert) pour le bras gauche du lanceur pour des mouvements après le lancer (c'est-à-dire, quand le bras revient à sa position initiale). Cette sélectivité non supervisée pourrait alors être utilisée pour suivre le mouvement d'objets spécifiques tels que la balle, la main du lanceur, etc.

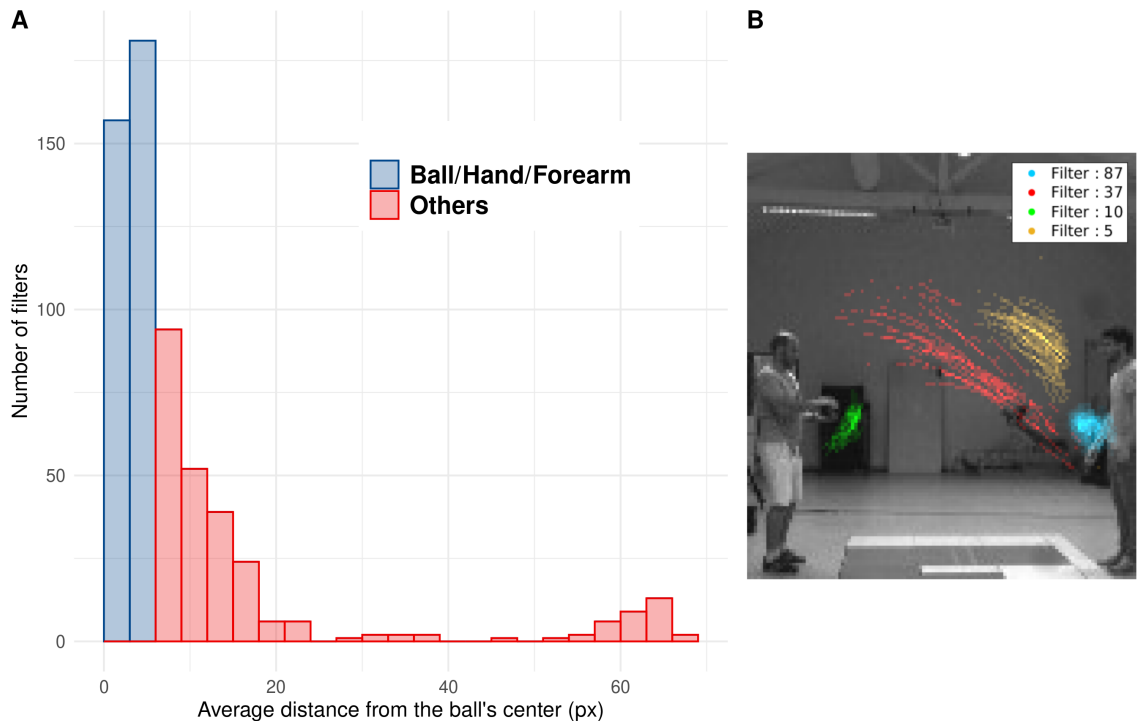


FIGURE 4.7 – Panneau A : Histogramme de la distance moyenne  $D_n$  de tous les filtres. Panneau B : Position des spikes pour quatre filtres différents

### 4.3.3 Prédictions

Après l'entraînement du SNN et des RPs, le jeu de test est ensuite présenté au SNN. Pour chaque spike généré par la dernière couche, la prédiction est mise à jour. Nous avons analysé l'erreur absolue moyenne (AE) et la écart type de l'AE (SD AE) obtenues pendant la phase de test, à travers des ANOVAs séparés, avec la visibilité (proportion de la trajectoire présentée, allant 15 à 90%) comme variable intra-sujet. L'ANOVA sur l'AE a montré que la visibilité influençait l'AE moyenne,  $F(5, 25) = 1156.38$ ,  $p < .001$ . Les tests post-hoc utilisant les corrections de Bonferroni ont démontré que l'AE moyenne était toujours réduite lorsque la visibilité augmentait (voir la couche 3 de la figure 4.8). Notre solution a de bonnes performances même avec 15 % de

la trajectoire avec une erreur de 7,7 pixels. Cette erreur diminue avec le temps de présentation et descend à 2,2 pixels pour 90 % de la présentation de la trajectoire. Aucune erreur de direction (lancer vers la droite/vers la gauche) n'a été faite, quel que soit le pourcentage de présentation. Il est important de noter que pour la condition de faible visibilité (15%), la vidéo est arrêtée en moyenne 0,114s après l'initiation du lancer, la balle est alors encore dans les mains du lanceur.

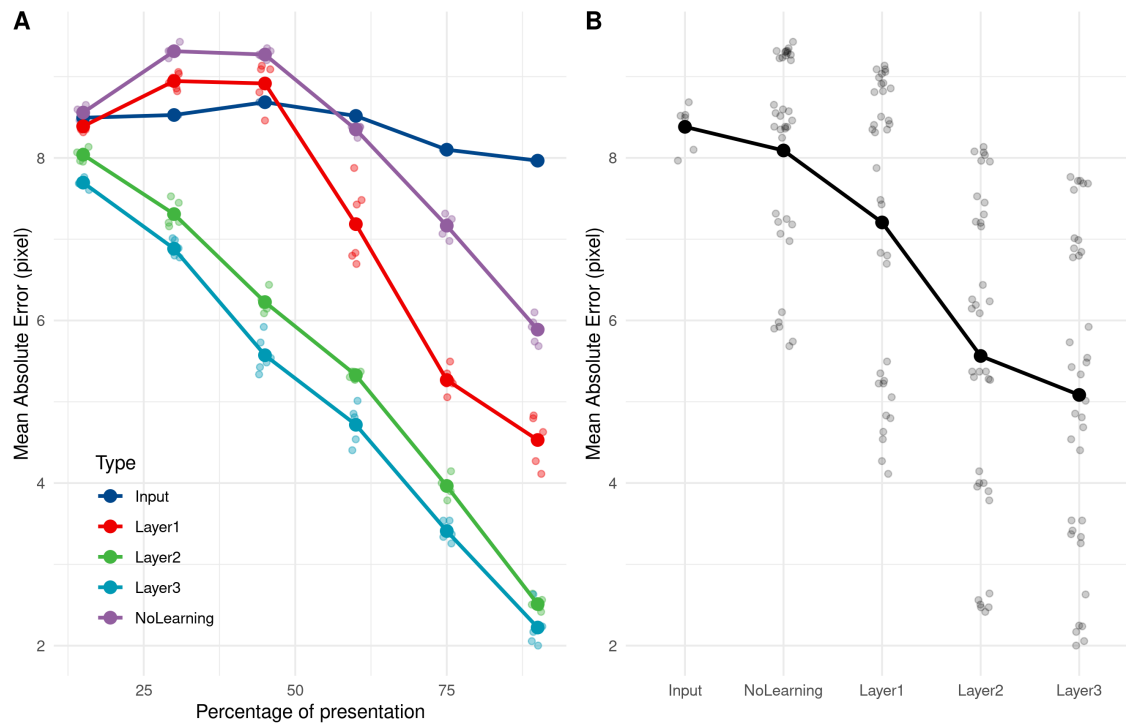


FIGURE 4.8 – Panneau A : Évolution de l'erreur absolue moyenne en fonction du temps de présentation pour des RPs appliqués à différentes entrées. Panneau B : Erreur absolue moyenne pour des RPs appliqués à différentes entrées.

A titre de comparaison, si nous calculons la moyenne de toutes les valeurs d'entraînement, égale à 68,2 pixels, et l'utilisons comme méthode de prédiction naïve, nous obtenons une erreur de 9,4 pixels. Même avec 15 % de la présentation de la trajectoire, le SNN peut prédire la direction du lancer et est 16,7 % meilleur que cette méthode "naïve". Le ballon est encore dans la main du lanceur pour toutes les trajectoires à 15% de la présentation et aussi à 30% pour 35 des 89 trajectoires (le ballon venait de quitter la main pour les autres).

L'analyse du SD AE a également montré une influence significative de la visibilité ( $F(5, 25) = 143.60, p < .001$ ), avec le SD AE diminuant au fur et à mesure que la visibilité augmente. Le SD passe de 5,605 pixels à 15% à 1,996 à 90%, sans différence significative entre les conditions 15 et 30%, et 75 et 90%. L'évolution de l'AE SD en fonction de la visibilité est visible sur la figure 4.11, panneau B.

Nous avons comparé l'erreur de prédiction sur différentes couches, avec un réseau non entraîné et avec des entrées directes pour évaluer la performance de l'apprentissage et l'impact de l'ajout de couches au réseau. Comme le montre la figure 4.8, en utilisant l'entrée directe (c'est-à-dire la sortie de la caméra), nous ne pouvons pas prédire le point de réception. Comme prévu, les RP ne sont pas précis avec les seules informations de position, et ont besoin de

caractéristiques plus complexes, telles que la vitesse et la direction encodées dans les filtres après entraînement. L'erreur de prédiction diminue au fil des couches successives. La première couche fait encore quelques erreurs, mais ses performances sont bien meilleures que celles du réseau non entraîné, plus particulièrement pour la fin de la trajectoire.

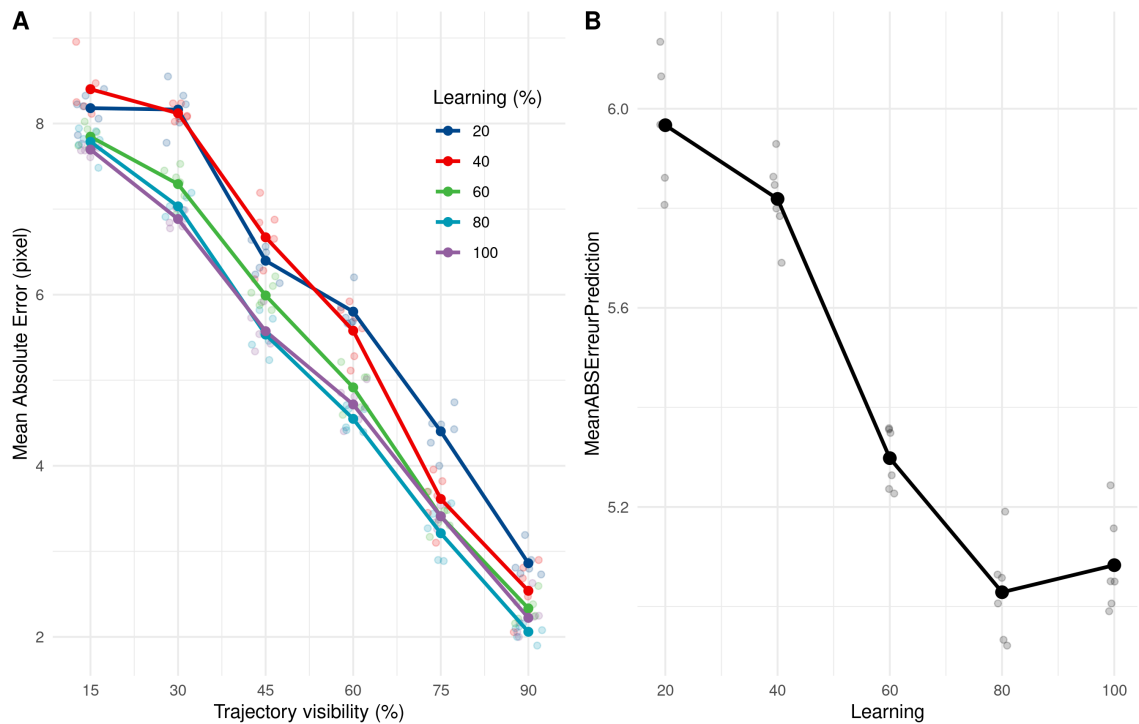


FIGURE 4.9 – Panneau A : Évolution de l'erreur moyenne absolue en fonction du temps de présentation de la trajectoire pour un nombre différents de trajectoires présentées durant l'apprentissage. Panneau B : Erreur moyenne absolue pour un nombre différents de trajectoires présentées durant la phase d'apprentissage.

Nous avons ensuite étudié la vitesse d'apprentissage de notre solution (SNN + RP) et le nombre de présentations nécessaires pour fournir des estimations correctes. Nous avons appliqué le même processus d'apprentissage et de prédiction que précédemment, mais l'apprentissage a été effectué avec seulement un sous-ensemble (20, 40, 60, 80 et 100 %) des 211 trajectoires d'entraînement. Notre approche a conduit à de très bonnes performances même en n'apprenant qu'à partir de 20 % des trajectoires. Les performances ont augmenté et ont ensuite atteint un plafond à 80 %, soit environ 168 trajectoires, comme le montre la figure 4.9.

#### 4.3.4 Performances chez l'humain

Nous avons analysé les performances des participants humains (AE et SD AE) à l'aide de deux ANOVA distinctes, avec l'apprentissage (pré-test, actif et post-test) et la visibilité (de 15 à 90%) comme variables intra-sujet. Les résultats montrent que la visibilité ( $F(5, 55) = 31,85$ ,  $p < .001$ ) et l'apprentissage ( $F(2, 22) = 10,78$ ,  $p < .001$ ) ont une influence sur l'erreur de prédiction et qu'il y a une interaction entre ces deux facteurs ( $F(10, 110) = 2,32$ ,  $p = .016$ ). Cependant, les post-tests de test-T par paire montrent qu'il n'y a pas de variations significatives entre les différents apprentissages pour des instants d'occlusions similaires. L'erreur de prédiction diminue avec le pourcentage de la trajectoire qui est présentée aux participants, comme le montre la figure 4.10, panneau B. D'après les tests post-hoc, nous n'avons pas observé de différences

significatives entre 15 et 45 % et entre 45 et 60 % de la présentation de la trajectoire. Il existe également des différences significatives entre la phase de pré-apprentissage et les autres phases d'apprentissage (apprentissage et post-apprentissage) et non entre l'apprentissage et le post-apprentissage, comme le montre la figure 4.10, panneau A. Ces résultats mettent en évidence l'effet de la phase d'apprentissage qui améliore les résultats de prédiction qui restent stables durant la phase de post-apprentissage.

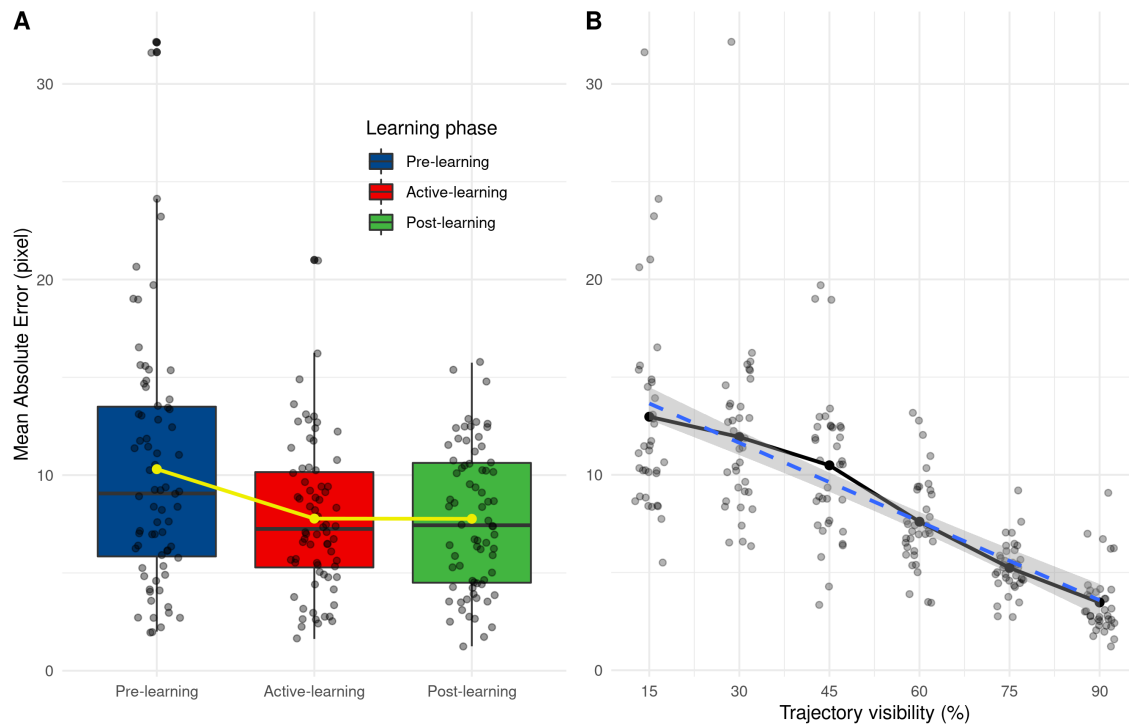


FIGURE 4.10 – Panneau A : Distribution de l’erreur absolue moyenne en fonction de la phase d’apprentissage. Panneau B : Evolution de l’erreur moyenne absolue en fonction du temps de présentation de la trajectoire.

### 4.3.5 Comparaisons

Pour comparer les performances entre les participants humains de notre expérience et notre SNN, nous avons effectué une ANOVA sur la moyenne de l’AE et l’écart-type de l’AE pour la condition entraînée uniquement (c’est-à-dire la condition post-test pour les humains et l’ensemble des 6 simulations du SNN). Nous avons utilisé le type de participant (humains vs SNN) comme variable intermédiaire, et la visibilité comme variable interne. En comparant les résultats entre les humains et les SNNs, une grande différence entre eux peut facilement être constatée au début en comparant leur erreur absolue moyenne ( $F(1, 16) = 47.09, p < .001$ ) et leur SD AE ( $F(1, 16) = 20.16, p < .001$ ) pour chaque participant/simulation. Cela suggère que les prédictions faites par le SNN sont plus exactes et plus précises que les prédictions humaines. La moyenne et l’écart-type de l’AE diminuent de manière linéaire lorsque la visibilité augmente (pour la moyenne :  $F(5, 80) = 38,21, p < .001$ , pour l’écart-type :  $F(5, 80) = 22,88, p < .001$ ), comme le montre la figure 4.11. Enfin, il y a une interaction significative,  $F(5, 80) = 2,53, p = .036$  pour AE seulement, indiquant que le SNN a toujours surpassé les participants humains, sauf lorsque la trajectoire était présentée pour 60 ou 90% de la trajectoire. L’absence d’interaction concernant SD AE ( $F(5, 80) = 1,19, p = 0,32$ ) indique que SD AE est généralement plus faible pour le SNN, indépendamment de la visibilité.

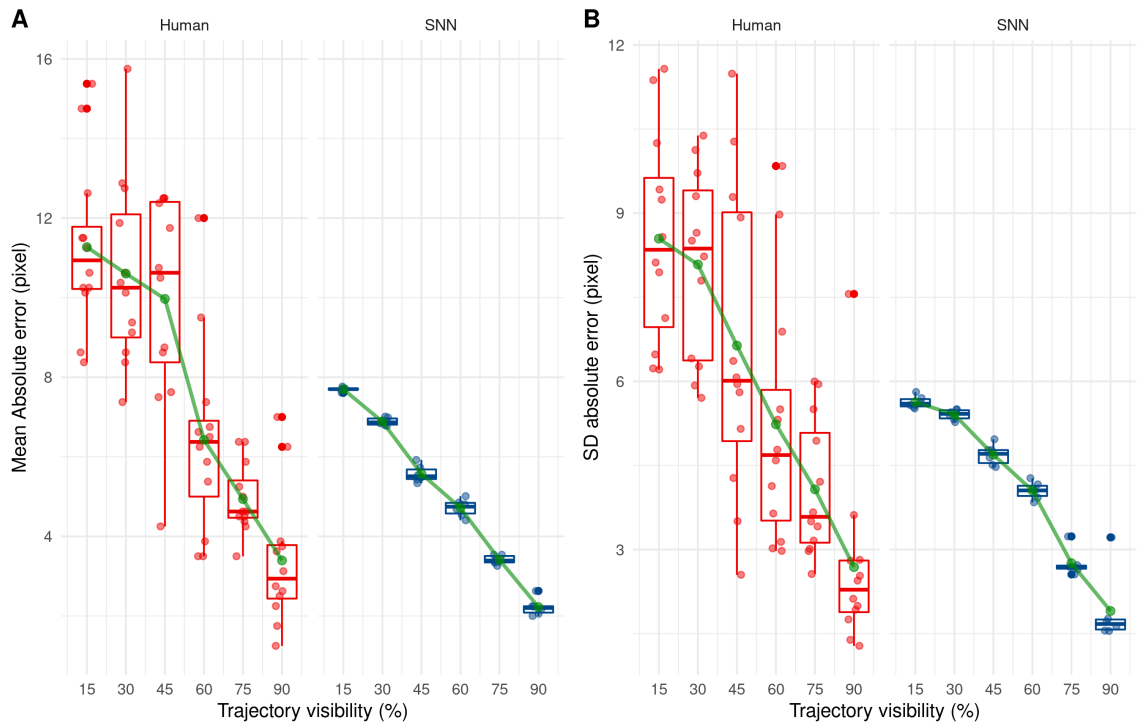


FIGURE 4.11 – Évolution de la moyenne (panneau A) et de l'écart type (panneau B) de l'erreur absolue en fonction du temps de présentation pour les participants humains (rouge) et les SNNs (bleu).

Ensuite, nous avons étudié comment le SNN ou les participants humains prédisent le point final de la trajectoire de la balle. La stratégie de prédiction utilisée par les humains et le SNN est différente, comme le montre la figure 4.12. En effet, le SNN a des prédictions moyennes proches de 68 pixels, ce qui est proche de la valeur moyenne à prédire de l'ensemble d'entraînement (égale à 68,2 pixels) et reste stable dans le temps. Au contraire, la prédiction moyenne des participants humains varie au cours du temps de présentation. En moyenne, les participants humains sous-estiment la position finale de la balle (prédiction inférieure à 68,2 pixels) pour le début de la trajectoire (durant les premiers 45%), ce qui se transforme en une légère surestimation au fur et à mesure que le temps de présentation de la trajectoire augmente.

La variabilité des valeurs prédites est également différente entre les humains et notre système. En effet, l'écart-type des valeurs prédites augmente avec le temps de présentation pour notre solution. Celui-ci est faible pour le début de la trajectoire et ainsi notre solution prédit des valeurs proches de 68px. Ensuite, la variabilité de la prédiction augmente avec plus de temps de présentation et donc d'informations. Cette tendance est également valable pour les humains mais de moindre ampleur qu'avec notre système.

Ces résultats montrent des stratégies différentes pour prédire le point de réception entre notre solution et les humains. D'une part, notre solution fait des prédictions prudentes et "statistiques" en ciblant en moyenne proche de 68.2px (la valeur moyenne de l'ensemble d'entraînement). Lorsque la visibilité augmente, le SNN est capable de s'écarter de cette valeur moyenne et de prédire une position différente, proche de la valeur correcte. En d'autres termes, le SNN fait des prédictions prudentes, basées sur la valeur moyenne de l'ensemble de données lorsque la visibilité est faible, mais est capable de faire des prédictions plus variées - mais précises - lorsque la visibilité augmente. Les humains semblent toutefois agir différemment, sans utiliser une moyenne statistique comme cible. Leur perception est variable même dans des conditions de faible visibilité, indiquant une décision au cas par cas, et semble passer d'une sous-estimation

à une légère surestimation lorsque la visibilité augmente.

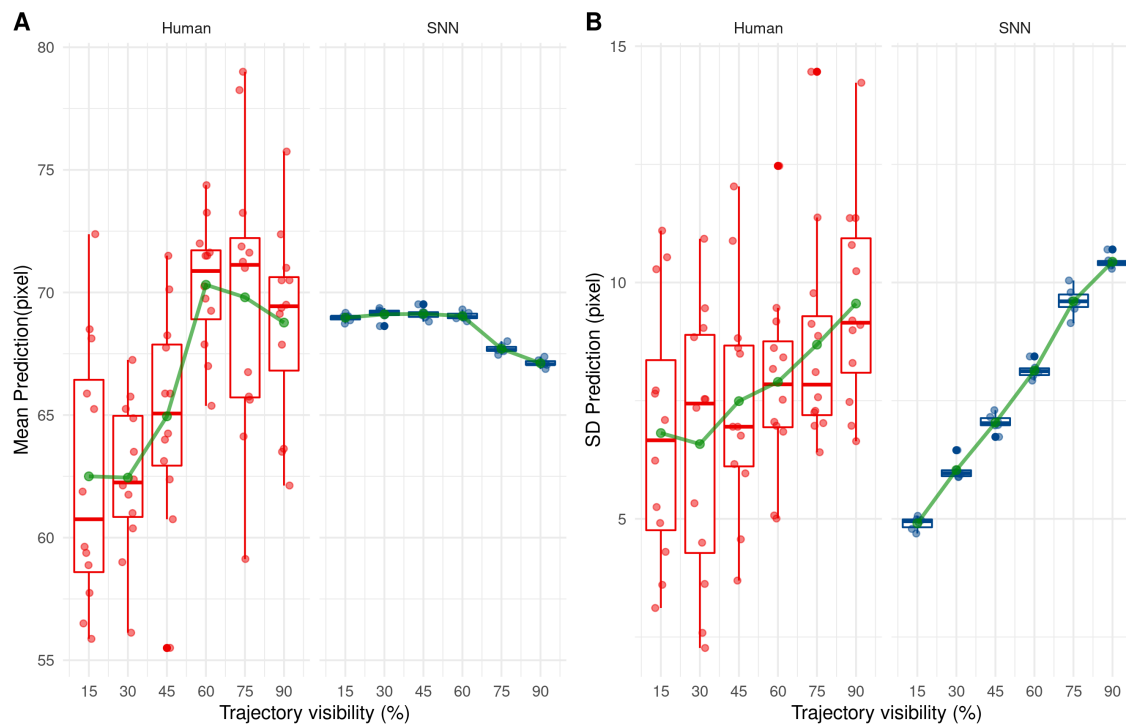


FIGURE 4.12 – Évolution de la moyenne (panneau A) et de l'écart type (panneau B) de la valeur prédite en fonction du temps de présentation pour les participants humains (rouge) et les SNNs (bleu).

## 4.4 Discussion

Des études ont déjà démontré la capacité des SNNs à analyser et extraire des informations du mouvement à partir de spikes générés par une caméra asynchrone. Cette étude participe à montrer la pertinence d'utiliser des SNNs combinés avec une caméra asynchrone pour le traitement du mouvement et ainsi démontre l'efficacité de ces réseaux de neurones. L'une des principales contributions de cette étude a été de mettre en avant la caméra NeuroSoc utilisée pour générer les événements et analyser le mouvement. Habituellement, les SNNs sont alimentés par des caméras événementielles asynchrones (BICHLER et al. 2012 ; G. ORCHARD et al. 2013 ; PAREDES-VALLES, SCHEPER et CROON 2019), qui émettent des événements lorsqu'il y a une variation de luminosité locale, générant ainsi un grand nombre de spikes. Dans cette étude, la variation de luminosité est encodée dans la temporalité de l'événements en utilisant le filtrage spatio-temporel de la caméra NeuroSoc et une règle de conversion de l'intensité en latence pour répartir les événements entre deux frames consécutives. Le processus de filtrage utilisé par le capteur NeuroSoc fournit ainsi des informations moins bruitées générant ainsi des spikes plus épars.

Dans un premier temps, nous montrons que notre système est capable de traiter le flux de spikes généré par la NeuroSoc et d'extraire des informations relative au mouvement (vitesse, direction). Deuxièmement, nous montrons que ces informations encodées dans les filtres peuvent ensuite être utilisées pour prédire le point de réception de la balle avec précision et pour suivre des motifs de mouvement spécifiques (bras, balle, etc.). Comme la sélectivité de mouvement de notre réseau, la capacité de suivi est entièrement non supervisée. Bien que celle-ci n'était pas

---

l'objectif principal de cette étude, elle met en évidence la capacité des réseaux neuronaux non supervisés à résoudre des tâches multiples. On peut donc s'attendre à ce qu'il soit pertinent pour d'autres tâches visuelles proches, telles que la détection et reconnaissance de "gestes", des tâches de comptage d'objets, etc.

Enfin, nous montrons que notre système surpasse la prédiction humaine sur cette tâche. Des études précédentes ont montré qu'il est possible de faire des prédictions à partir des sorties de SNNs recevant des événements générés par une caméra événementielle. Certaines d'entre elles utilisent des LSMs, mais leurs prédictions restent limitées à de courtes durées (c'est-à-dire quelques ms) (BURGSTEINER et al. 2007 ; KAISER et al. 2017). Une autre étude a utilisé des retards dans les connexions latérales anisotropes (KAPLAN et al. 2013). D'autres utilisent une règle d'apprentissage pour anticiper les entrées sur des temps de prédiction plus longs mais avec des stimuli d'entrée plus simples et répétitifs (GIBSON, HENDERSON et WILES 2014). Notre système permet des prédictions sur des durées plus longues en analysant les caractéristiques du mouvement, car les trajectoires de balles sont limitées par les lois de la physique et donc prévisibles. Les capteurs asynchrones permettent d'effectuer un codage parcimonieux sur des scènes visuelles dynamiques. Dans le contexte de cette étude, seule une petite partie de la scène visuelle est pertinente à extraire. Seul le mouvement (balle et bras) est nécessaire pour effectuer la tâche demandée. En utilisant le jeu de données de test présenté comme le SNN, une moyenne d'environ 9000 événements par seconde a été générée par la caméra NeuroSoc. Cela représente seulement 0,26% des pixels pour chaque image ( $37 \pm 15,5$  événements ON/OFF par image) et montre ainsi que, par rapport à une caméra synchrone *frame-based*, la sortie de la NeuroSoC était plus parcimonieuse que celle d'une caméra classique. Ceci, à son tour, met en avant la possibilité de l'efficacité énergétique du système s'il était intégré dans une puce neuromorphique, car la consommation d'énergie des réseaux neuronaux à spike est déterminée par le nombre de spikes traités (FARABET et al. 2012).

Même si notre approche a été évaluée sur une tâche relativement simple, avec peu de variations dans les trajectoires et peu de mouvements pouvant gêner la prédiction, cette évaluation reste pertinente pour de nombreuses situations telles que les lancers francs au basket-ball, ou les objets se déplaçant le long d'une contrainte spécifique (par exemple, des voitures se déplaçant sur une route ou un piéton traversant un trottoir). Dans une situation plus complexe avec un mouvement d'arrière-plan, des spikes générés ne provenant pas de la trajectoire de la balle, la capacité de suivi du mouvement du SNN pourrait être utilisée pour distinguer le mouvement du ballon des autres objets.

A l'issue de cette étude, l'étape suivante consistait à évaluer notre système sur des trajectoires plus complexes (rebonds, collisions, etc.) ou sur des situations plus réalistes et proches de ce que l'on retrouve dans le domaine du sport. Par conséquent, ce travail a été le premier pas vers la démonstration de la fiabilité d'un SNN pour extraire des caractéristiques spatio-temporelles utiles en utilisant la caméra NeuroSoc.

# Chapitre 5

## Etude 2 : Basket

### 5.1 Motivations et objectifs

Après avoir validé la pertinence d'utiliser une caméra événementielle avec un SNN pour extraire de l'information sur le mouvement et ensuite l'analyser pour effectuer des prédictions de trajectoires, nous voulions évaluer ce système dans une tâche plus proche de ce qui se fait lors d'une phase de sport. Nous nous sommes donc orientés vers le domaine du basket et plus particulièrement, des phases de lancer-franc. Ceux-ci nous permettent d'être sur une tâche proche de la précédente expérience, avec un mouvement principalement latéral. Comme pour la précédente étude, l'objectif est d'évaluer la performance de notre solution avec des temps de présentation différents et de comparer ces résultats avec ceux d'humains. La tâche demandée au SNN et aux participants était de prédire si le lancer franc était réussi (ballon qui rentre dans le panier) ou raté.

### 5.2 Acquisition de données

Nous avons fait nos acquisitions de trajectoires lors d'un entraînement de basket du pôle espoir du Centre de Ressources d'Expertise et de Performance Sportive (CREPS) de Toulouse. Durant leur entraînement, des groupes de deux joueuses et joueurs tournaient et venaient s'entraîner aux lancer-francs. Nous avons pu enregistrer un total de 280 trajectoires de 8 tireurs différents. Un nombre de 27 trajectoires étaient fortement perturbées car d'autres joueuses et joueurs passaient devant la caméra et cachaient le joueur qui lançait ou la trajectoire de la balle. Ces trajectoires ont été supprimées des trajectoires utilisées pour tester les humains et le SNN, mais ont été conservées lors de la phase d'entraînement du SNN (les trajectoires n'étaient que partiellement perturbées et avec le peu de trajectoires que nous avons, nous avons préféré les conserver pour l'entraînement).

La caméra NeuroSoc était placée orthogonalement aux trajectoires de lancer-francs et suffisamment éloignée pour enregistrer la totalité de la trajectoire, du mouvement du joueur qui lance, jusqu'au panier. La caméra NeuroSoc fonctionnait en mode 100 images par seconde (*Frames Per Second*) (FPS). Les salles d'entraînement de basket utilisent des tubes néons, ceux-ci éclairent à une fréquence de 100Hz. Nous avons alors choisi un FPS multiple de leur fréquence afin d'atténuer les effets de variation de luminosité brutales. Il restait cependant des variations de luminosités entre les frames, mais qui n'entraînaient pas de génération d'événements lorsqu'il n'y avait pas de mouvements. Les murs de la salle de basket étant d'aspect métallique, ceux-ci reflétaient fortement la luminosité émise par les néons entraînant une forte variation de luminosité sur une partie de l'image (cf. figure 5.1).

Chaque trajectoire a comme point de départ le moment où le joueur qui lance a le ballon qui



---

quitte ses mains, nommé comme l'instant  $t=0\text{ms}$  (cf. figure 5.1). Les prédictions sont effectuées à 8 instants différents ( $t=[-150, 0, 150, 300, 450, 600, 750, 900]\text{ms}$ , figure 5.3), une prédiction est donc faite avant que le ballon quitte les mains du lanceur, la prédiction se base alors seulement sur le mouvement de la personne qui lance. Les sept autres sont le moment où le ballon quitte les mains du lanceur ou durant la phase aérienne de la trajectoire du ballon.

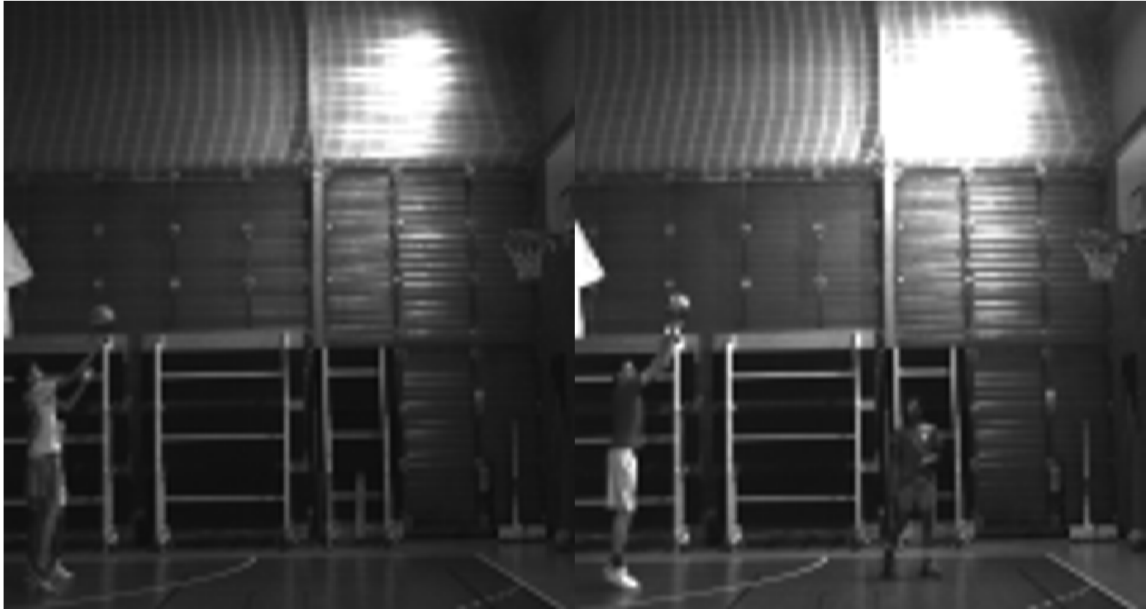


FIGURE 5.1 – Variations de luminosités

## 5.3 Méthodes

### 5.3.1 Architecture

Comme pour les précédentes expériences, les événements en sortie de la caméra NeuroSoc étaient utilisés comme entrée du SNN et dupliqués pour y ajouter des délais. Nous avons utilisé un SNN à trois couches (voir figure 5.2). Les deux premières étaient des couches convolutionnelles entraînées avec la règle STDP présentée dans la section 3.1.2.5. La troisième et dernière couche est une couche de type dense (cf. section 2.2.5.3) entraînée avec une règle de type RSTDP (M. MOZAFARI et al. 2018). Afin de réduire le nombre de connexions avec cette dernière couche, une phase *pooling* est effectué. Ainsi les dimensions de la dernière couche sont divisées par deux en longueur et en largeur. Les spikes en sortie de cette dernière couche sont ensuite utilisés pour faire la prédiction. La RSTDP attribue des classes aux neurones. On peut ensuite facilement décoder la sortie de cette couche en comptant le nombre de spikes émis par les neurones de chaque classe. Une méthode similaire à 4.2.4 est utilisée pour intégrer le score temporellement et donner plus de poids au derniers spikes émis. Ainsi, contrairement à la précédente étude (4.2.4) qui nécessitait l'utilisation de RPs pour décoder la dernière couche, aucun post-traitement nécessitant un entraînement n'est utilisé.

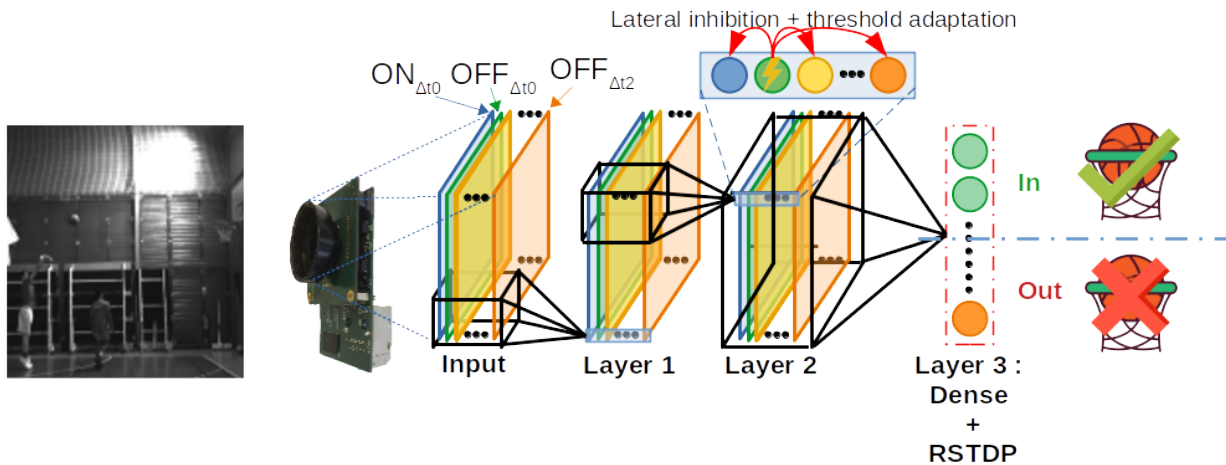


FIGURE 5.2 – Architecture de la solution utilisée pour la prédiction de lancer franc de basket

### 5.3.2 Apprentissage et prédictions

Parmi les objectifs de cette étude, nous voulions étudier si le SNN était capable de déterminer quel est le joueur qui lance. Huit différents joueurs ont participé à la phase d'acquisition, une division en deux jeux de données train/test (70/30%) n'était alors pas suffisant. En effet, avec seulement 30% de trajectoires, cela ne permettait pas de généraliser sur la sélectivité aux différents lanceurs, nous avons alors utilisé une méthode de validation croisée à 5 blocs. Ainsi, 5 jeux de train/test (80/20%) sont générés de façon à avoir toutes les trajectoires testées sur l'ensemble des jeux de test et pouvoir mieux généraliser la sélectivité des neurones. Dix différentes simulations avec des poids synaptiques initiaux différents (aléatoirement sélectionnés) sont effectuées, deux simulations sont faites par jeu de données. Les filtres convolutionnels des deux premières couches sont entraînés de façon similaire à la précédente étude 4.2.5. La dernière couche utilise la règle de RSTDTP (M. MOZAFARI et al. 2018), les neurones sont initialisés et séparés également en deux classes distinctes (lancer réussi, classe et label égale à 1 ou raté, classe et label égale à 0). Lors de la décharge d'un des neurones, selon qu'il décharge pour le bon type de trajectoire (neurone de la classe 0 décharge pour une trajectoire de type 0), un apprentissage de type STDP est utilisé, sinon (neurone de la classe 1 décharge pour une trajectoire de type 0) un apprentissage de type anti-STDP (voire section 2.3) est utilisé. Afin d'éviter le sur-apprentissage, comme proposé dans (M. MOZAFARI et al. 2018), une méthode de *dropout* est utilisée ainsi qu'une méthode de gestion du ratio d'apprentissage. La gestion du ratio d'apprentissage est composée de deux critères, le premier est similaire à la méthode utilisée pour adapter la vitesse d'apprentissage de (M. MOZAFARI et al. 2018), sauf que  $N = N_{hit} + N_{miss}$ . Un second terme est alors utilisé pour avoir un ratio d'apprentissage décroissant, il s'agit de  $a_L R$ , (5.1).

$$\begin{aligned}
W_{i,j} &= W_{i,j} + \Delta W_{i,j} \times \frac{N_{hit}}{N} \times a_{LR} \quad \text{si STDP} \\
W_{i,j} &= W_{i,j} + \Delta W_{i,j} \times \frac{N_{miss}}{N} \times a_{LR} \quad \text{si anti-STDP} \\
a_{LR} &= \exp\left(\frac{-N}{\tau_{LR}}\right)
\end{aligned} \tag{5.1}$$

La règle STDP de la dernière couche dense, a été adaptée afin que la temporalité des spikes ait plus d'impact sur la modification des poids. Dans la règle précédente, présentée dans la section 3.1.2.5, l'intensité de la variation du poids ne dépendait pas de la temporalité du spike pré-synaptique. Celle-ci servait seulement à définir si on appliquait le LTP ou le LTD à la connexion. La couche dense permet d'intégrer des spikes sur une plus grande durée temporelle, il nous paraissait alors nécessaire d'avoir une intensité de modification de poids proportionnelle à la latence entre le spike pré-synaptique et post-synaptique (équation (5.2)), comme le fait normalement la règle STDP.

$$\begin{aligned}
\Delta W_{i,j} &= (R_{i,j} \times a_{rLTP} \times \exp\left(\frac{-\Delta t}{\tau}\right) - (R_{i,j} \times a_{rLTD} \times (1 - \exp\left(\frac{-\Delta t}{\tau}\right))) \quad \text{si STDP} \\
\Delta W_{i,j} &= (R_{i,j} \times a_{pLTP} \times (1 - \exp\left(\frac{-\Delta t}{\tau}\right))) - (R_{i,j} \times a_{pLTD} \times \exp\left(\frac{-\Delta t}{\tau}\right)) \quad \text{si anti-STDP} \\
R_{i,j} &= 4 \times W_{i,j} \times (1 - W_{i,j})
\end{aligned} \tag{5.2}$$

La prédiction s'effectue en intégrant directement les spikes de la dernière couche du SNN. Aucune étape supplémentaire nécessitant un apprentissage supervisé n'est utilisée comme dans la précédente étude. Les spikes sont seulement comptés en incrémentant de un la valeur de  $Sc_{good}$  ou  $Sc_{miss}$ , selon que le neurone qui décharge fasse partie de la classe réussi ou raté. Comme pour la précédente étude, un facteur de fuite est appliqué aux deux scores, afin que les derniers spikes aient plus d'impact (5.3). La valeur maximale est ensuite choisie comme valeur prédite.

$$Sc_{good/miss}(t + \Delta t) = Sc_{good/miss}(t) \times \exp\left(\frac{-\Delta t}{\tau_{Sc}}\right) \tag{5.3}$$

### 5.3.3 Performances chez l'humain

Afin de comparer les résultats de notre solution avec ceux d'humains, nous avons comme pour la précédente étude (cf. section 4.3.4) fait passer une expérience similaire à des humains experts et non-experts. La tâche était similaire à celle que doit faire le SNN, c'est-à-dire prédire si le lancer est réussi ou non avec seulement une partie de la vidéo qui est présentée. Pour cela, 10 trajectoires étaient présentées pour chaque durée de présentation à chaque participant (5 lancer bons, 5 ratés). 8 différentes durées de présentation de la trajectoire de balle étaient possibles, la vidéo s'arrêtait à l'instant  $t = [-150, 0, 150, 300, 450, 600, 750, 900]$  ms, avec  $t=0$  ms l'instant où le joueur lâche le ballon de ses mains. Ainsi, à l'instant  $t = -150$  ms, le joueur est dans sa phase de lancer, balle encore dans les mains. Chaque vidéo démarrait à l'instant  $t = -700$  ms permettant ainsi d'avoir un minimum de 550 ms de présentation à  $t = -150$  ms.

Nous avons divisé les participants en deux catégories, les experts et non-experts. Les experts étaient des joueuses de l'équipe U18 France de basket du CREPS de Toulouse, qui pratiquent le basket en haut niveau. Les autres personnes étaient considérées comme non-expertes. Nous avons 12 participantes dans le groupe expert et 12 dans le groupe non-expert. Tout comme la

précédente expérience, les vidéos étaient présentées sur un écran de 13.3 pouces (60Hz, pleine résolution 1366 × 768, dimension 29.5 × 17 cm en horizontal par vertical). Les participants avaient ensuite à appuyer sur un bouton pour dire s'ils pensaient que le lancer était bon ou mauvais. L'expérience était divisée en deux parties, une première d'entraînement durant laquelle un feedback sur leur résultat était affiché après la présentation de chaque trajectoire, puis une seconde de test durant laquelle les participants réalisaient la même tâche, sans feedback cependant sur leur performance.

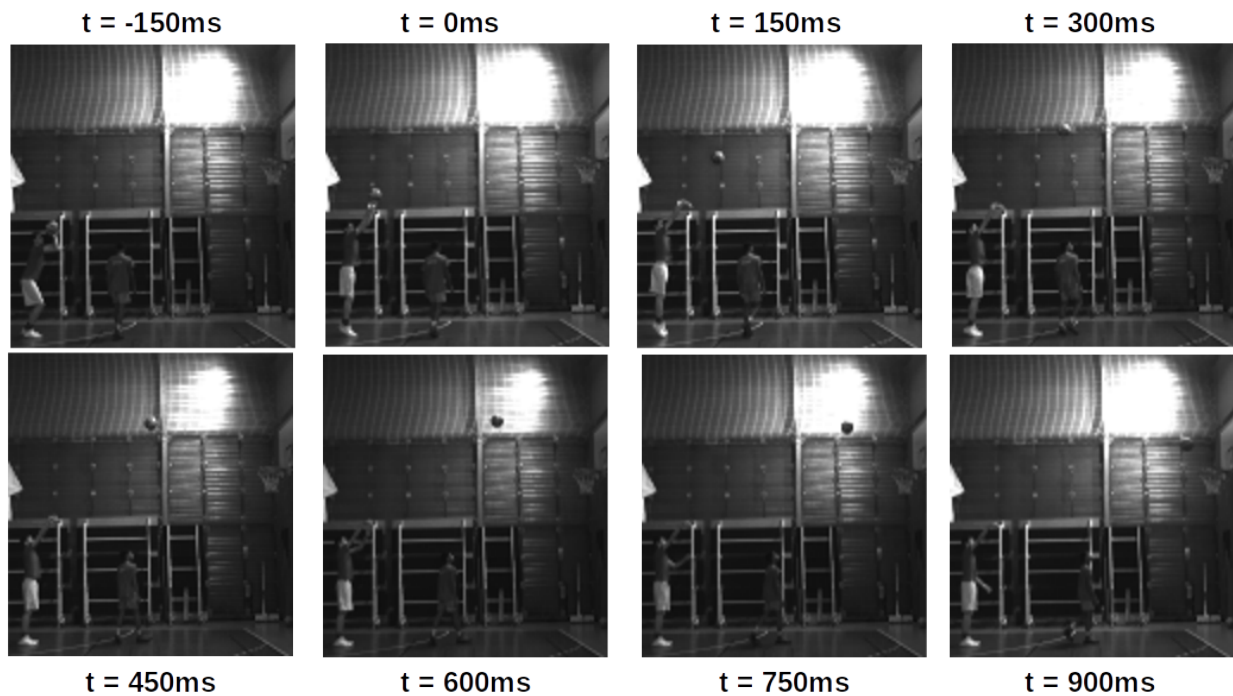


FIGURE 5.3 – Trajectoire de lancer franc à différents instants

## 5.4 Résultats

### 5.4.1 Prédictions (SNN)

Premièrement, nous avons évalué la qualité de la prédiction du SNN (sans comparaison avec les participants humains) avec différentes durées de présentation de la trajectoire, allant de -150ms (i.e., 150 ms avant que le joueur lâche le ballon), jusqu'à 900ms après l'avoir lâché. Afin que la prédiction soit faite par le SNN, contrairement à la précédente étude (Section 4), nous avons utilisé la règle d'apprentissage RSTDP. Celle-ci consiste à utiliser une règle STDP classique lorsque le neurone décharge pour la bonne classe et une règle anti-STDP sinon, pour que le neurone "oublie" le signal pour lequel il vient de décharger. En regardant la matrice de poids des neurones sur la figure 5.4, panneau B, on voit que la règle RSTDP a permis aux neurones de sélectionner les connexions spatiales correspondantes aux statistiques de l'entrée (figure 5.4, panneau A).

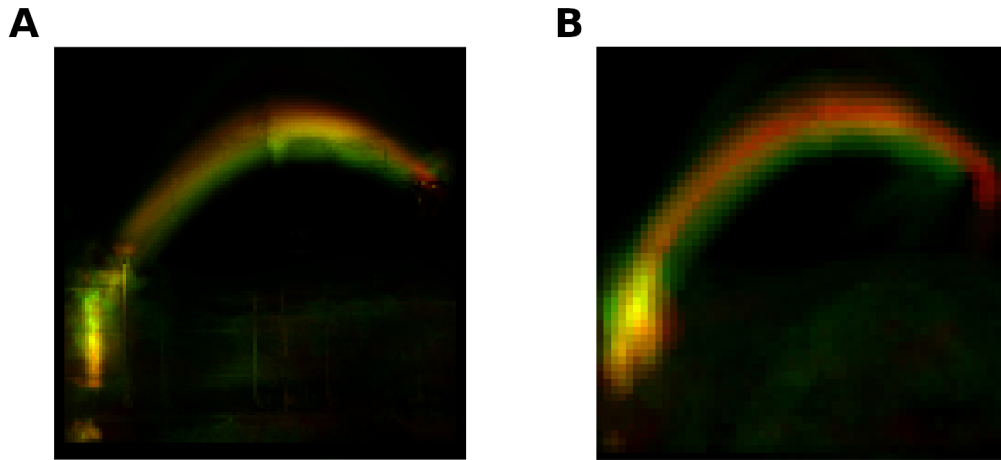


FIGURE 5.4 – Panneau A : événements générés par la caméra pour les trajectoires réussies (rouge) et ratées (vert). Panneau B : Somme des poids synaptiques des neurones de la classe "réussie" (rouge) et de la classe "ratée" (vert)

Nous avons analysé les performances des différentes simulations (les dix décrites dans la partie 5.3.2) du SNN selon deux critères : le pourcentage de bonnes réponses et la tendance de prédiction (le SNN prédit-il plus de lancers francs réussis ou non). Pour cela nous avons utilisé deux ANOVA distinctes, avec la visibilité (de -150 à +900ms) comme variables intra-sujet. Les résultats montrent que la visibilité ( $F(7, 63) = 2.28, p = .039$ ) a une influence sur le taux d'erreur. En effectuant des post-tests de test-T par paire, ainsi que grâce au panneau A de la figure 5.5, on voit que la visibilité a un impact à la fin de la trajectoire, à l'instant  $t=900ms$ . Cependant, la précision du SNN est constamment supérieure au facteur chance (= 50%) avec une moyenne de 60.01%. Toutefois, contrairement à l'étude précédente, le taux d'erreur varie peu avec le temps, excepté pour la dernière prédiction à  $t=900ms$  où le taux d'erreur diminue.

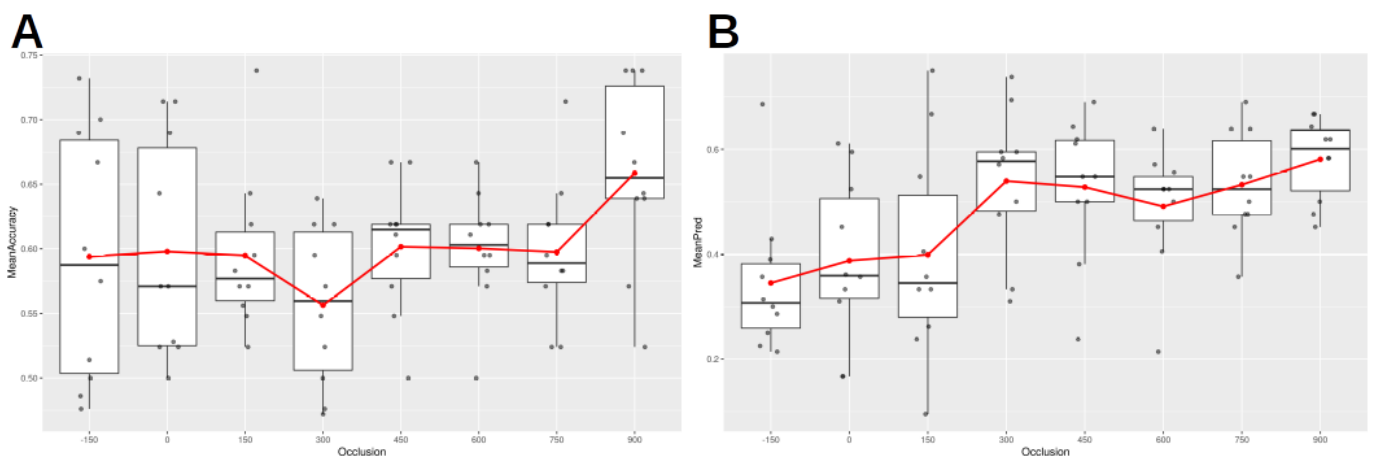


FIGURE 5.5 – Évolution de la prédiction effectuée par le SNN selon le temps de présentation. Panneau A : Précision de la prédiction. Panneau B : Valeur prédite, plus la valeur est proche de zéro, plus le SNN a prédit que le lancé est raté.

Concernant la tendance de prédiction, les résultats montrent que la visibilité ( $F(7, 63) = 5.42, p < 0.001$ ) a une influence sur celle-ci. Les post-tests ainsi que la panneau B de la figure 5.5, montrent que durant les trois premiers instants d'occlusions le SNN a plutôt tendance à prédire plus de lancers francs ratés, alors qu'à partir de  $t=300ms$ , cette tendance s'inverse.

## 5.4.2 Impact du *shooter*

Le SNN étant entraîné pour prédire si le lancer-franc est bon ou non, il peut utiliser les informations sur la cinématique du lanceur en plus de la trajectoire de la balle. Ainsi, il est possible qu'il ait aussi appris à reconnaître la personne qui lance le ballon. En effet, certains joueurs réussissent plus que d'autres leurs lancer-francs (41% de réussite pour le joueur numéro 4, 77% et 79% pour les joueurs 7 et 8). Il peut alors être avantageux de prédire qu'il y aura panier si c'est le joueur numéro 7 ou 8 qui lance, avant même que celui-ci n'ait lancé le ballon. Il paraît alors possible que le SNN ait appris à reconnaître les joueurs ou certains critères permettant de les reconnaître comme par exemple, la taille, le mouvement de lancer, les couleurs des habits (contraste avec l'arrière plan), etc. D'une part cette information participe à la prise de décision et la RSTDp bien qu'étant une règle semi-supervisée, s'appuie très fortement sur la STDP qui n'est pas supervisée et qui devrait être sensible au pattern permettant de différencier des joueurs. Pour cela, nous avons compté le nombre de spikes émis pour chaque joueur par chaque neurone de la couche de sortie du SNN lors de la présentation du stimulus d'entraînement, afin d'évaluer la sélectivité aux différents joueurs. Nous avons ensuite utilisé cette sélectivité pour prédire le joueur qui *shoot* lors de la présentation du stimulus de test. Pour cela, lors de la présentation du stimulus de test, un vecteur de taille 8 (nombre de joueurs) est défini et initialisé à zéro. Puis, lorsqu'un spike est émis par un neurone  $n$  ce vecteur est mis à jour, dans un premier temps en appliquant un facteur de fuite similaire à la méthode utilisée dans la précédente étude (section 4.2.4, équation (4.2)) ; dans un second temps, en ajoutant une valeur égale au nombre de spikes pour chaque joueur lors de la phase d'entraînement pour le neurone  $n$ , divisé par le nombre total de spikes émit par ce même neurone. Cela ajoute donc une valeur pour chaque joueur comprise entre 0 et 1, dont la somme est égale à 1. L'index de la valeur maximale est ensuite sélectionnée comme valeur prédite.

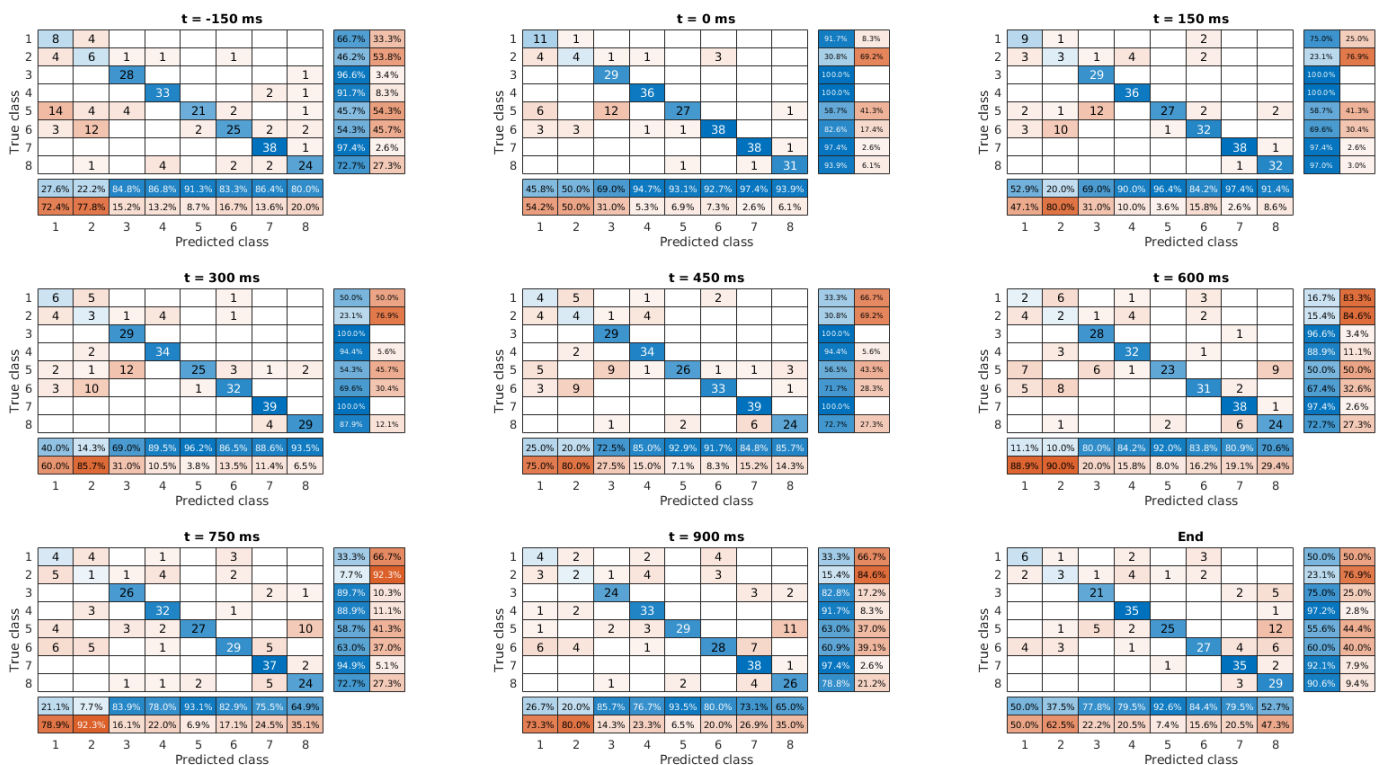


FIGURE 5.6 – Matrice de confusion de prédiction des *shooters*, à différents temps de présentation

---

Comme on peut le voir sur la figure 5.6, le SNN arrive très bien à discriminer quel est le *shooter* et plus particulièrement en début de trajectoire ( $t=-150\text{ms}$  et  $t=0\text{ms}$ ). Cela paraît totalement normal, car jusqu'à  $t=0\text{ms}$ , le joueur n'a pas encore lâché le ballon. Les spikes générés sont ainsi principalement générés par ce joueur et par son mouvement de lancer permettant de le reconnaître. Les instants qui suivent sont plus caractérisés par la trajectoire du ballon, qui permet moins de reconnaître le joueur qui lance et explique donc une augmentation des erreurs. Mais, les prédictions restent toutefois bonnes car on garde la trace des précédents spikes en mémoire. De plus, il est aussi possible que les trajectoires de balles soient aussi un moyen fiable pour discriminer les différents *shooters*. On peut voir que deux joueurs sont plus difficilement prédits par le SNN, les joueurs numéros deux et cinq. Le joueur numéro deux est beaucoup moins présent dans les acquisitions faites, il y a seulement 13 lancers pour le joueur deux et il y a par exemple, 45 trajectoires pour le joueur six. Le SNN a donc eu beaucoup moins de trajectoires pour s'entraîner sur le deuxième joueur (ainsi que sur le premier) et arrive donc beaucoup moins à le discriminer. Cela ne veut pas pour autant dire que le SNN n'a rien appris de leurs trajectoires et qu'il décharge aléatoirement lorsqu'elles sont présentées. Si on regarde la matrice de confusion à l'instant  $t=0\text{ms}$ , pour le joueur numéro cinq, on voit que 59% des prédictions sont bonnes, pour 26% des lancers il prédit que c'est le joueur numéro trois et pour 13% des lancers il prédit que c'est le joueur numéro un. Ainsi, bien que le SNN fasse plus d'erreurs pour certains joueurs, celles-ci restent en dessous du facteur de chance (allant de 5% pour le joueur numéro un à 18% pour le joueur numéro six). Le SNN a appris des patterns spécifiques à ces joueurs, ces patterns sont principalement partagés avec le joueur un et trois pour le joueur numéro cinq et avec les joueurs un et six pour le joueur numéro deux.

Afin de voir si le SNN pouvait discriminer le joueur qui *shoot* en se basant seulement sur la trajectoire du ballon, nous avons effectué la même analyse que la précédente, sauf que seuls les spikes générés 100ms après que le *shooter* ait lâché le ballon sont utilisés pour la prédiction. Ainsi, le SNN ne peut plus utiliser le mouvement de lancer du joueur, mais seulement la trajectoire du ballon. La figure 5.7 nous montre que le SNN commet beaucoup plus d'erreurs de prédiction, mais arrive tout de même à prédire convenablement le joueur qui lance. En prenant à l'instant  $t=900\text{ms}$  après que le *shooter* ait lâché le ballon, on voit que les joueurs 4 et 7 ont été prédits avec précision et en moyenne 51% des joueurs sont prédits avec succès.

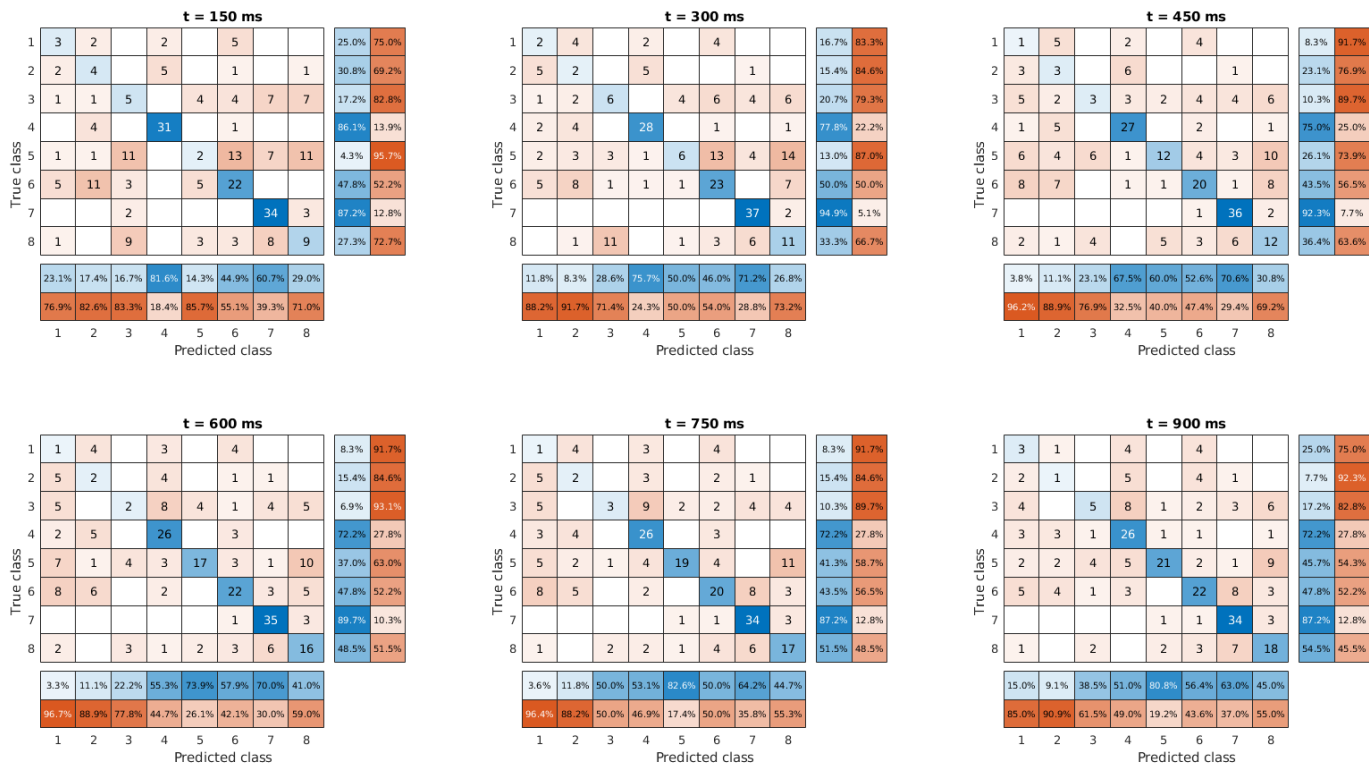


FIGURE 5.7 – Matrice de confusion de prédiction des *shooters*, sans le mouvement de lancer du *shooter*

### 5.4.3 Prédictions (Humains)

Afin d'évaluer la performance de notre SNN, nous avons fait passer une expérience similaire à des participants experts et non-experts. Les participantes considérées expertes étaient toutes de sexe féminin avec en moyenne  $9.25 (\pm 2.56)$  années de pratiques et jouant actuellement en U18 France, plus haut niveau français pour leur catégorie d'âge. Les participants non-experts étaient composés de 5 femmes et 7 hommes, n'ayant pas joué au basket avec un niveau avancé.

Nous avons analysé les performances des participants à l'aide de deux ANOVA distinctes (pourcentage de bonnes réponses et la tendance de prédiction), avec l'apprentissage (sans/avec feedback), la visibilité (de -150 à +900ms) comme variables intra-sujet et l'expertise comme variables inter-sujet. Les résultats montrent que la visibilité ( $F(7, 154) = 3.63, p = .001$ ) et l'expertise ( $F(1, 22) = 15.29, p < .001$ ) ont une influence sur la précision. Cependant, l'apprentissage n'a pas d'influence ( $F(1, 22) = 1.08, p = .31$ ). De plus, il n'y a pas d'interaction entre les différents facteurs : visibilité et apprentissage ( $F(7, 154) = 1.53, p = .16$ ), expertise et apprentissage ( $F(1, 22) = 0.59, p = .45$ ), expertise et visibilité ( $F(7, 154) = 0.76, p = .62$ ) et les trois facteurs ( $F(7, 154) = 1.28, p = .26$ ). Les joueuses expertes avaient un taux de réussite de 0.551 donc supérieur au seuil de chance égal à 0.5 ( $t(11) = 6.36, p < .001$ ). Alors que les participants non-experts avaient un taux de réussite égal à 0.497, équivalent au seuil de chance, ( $t(11) = 0.28, p = .79$ ) (figure 5.8, panneau A). Nous avons réalisé des tests de student paillés comme test post-hoc pour déterminer les différences significatives entre les différentes occlusions. Ces tests post-hoc ainsi que le panneau B de la figure 5.8 montrent qu'à l'instant  $t=-150$ ms la précision des participants est plus faible que pour les autres temporalités. Ainsi que, à l'instant  $t=0$ ms la précision des participants est supérieure que les deux temporalités d'occlusions suivantes et qu'à l'instant  $t=300$ ms, la prédiction est moins bonne qu'à  $t=0$  ou



900ms.

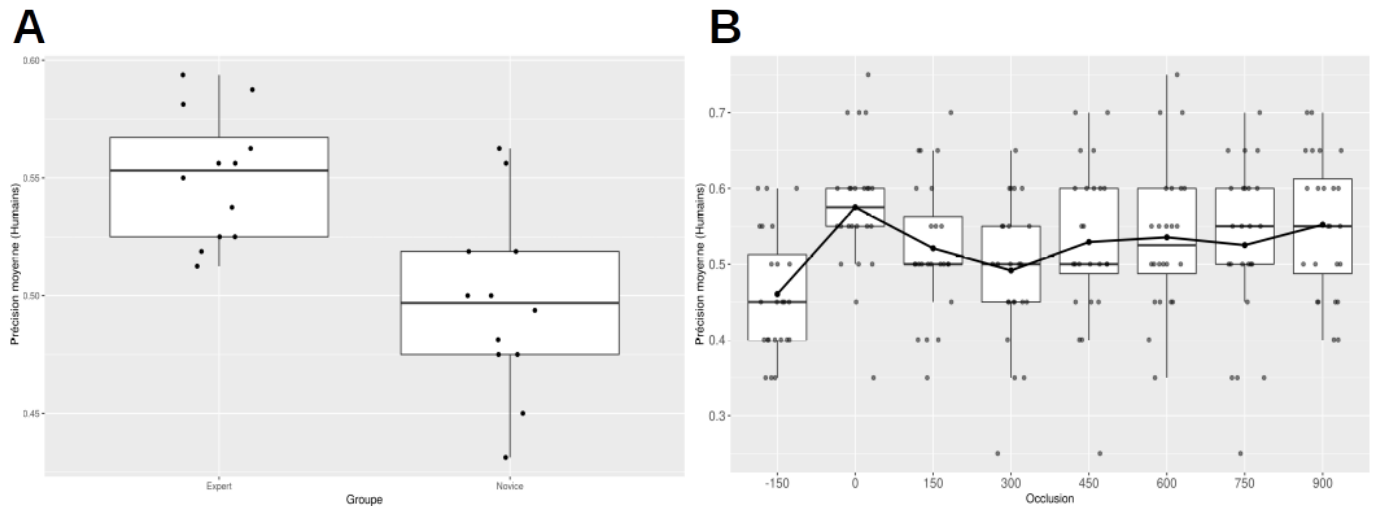


FIGURE 5.8 – Panneau A : Boxplot de l'erreur moyenne selon que le sujet est expert ou non. Panneau B : Évolution de l'erreur moyenne en fonction du temps de présentation de la trajectoire.

Nous avons effectué la même analyse concernant les prédictions faites par les participants. Nous avons ainsi regardé si ceux-ci ont plutôt tendance à prédire que les lancer-francs sont ratés ou réussis. Pour cela, comme pour l'analyse de l'erreur de prédiction, nous avons analysé la tendance de prédiction des participants à l'aide d'une ANOVA. Les résultats montrent que la visibilité ( $F(7, 154) = 22.49, p < .001$ ) a une influence sur la prédiction. L'expertise ( $F(1, 22) = 0.14, p = .71$ ) et l'apprentissage n'ont pas d'influence ( $F(1, 22) = 0.02, p = .88$ ). De plus, il y a une interaction entre les facteurs visibilité et apprentissage ( $F(7, 154) = 6.51, p < .001$ ). Il n'y a pas d'interaction entre les facteurs expertise et apprentissage ( $F(1, 22) = 1.91, p = .18$ ), expertise et visibilité ( $F(7, 154) = 0.76, p = .62$ ) et les trois facteurs ( $F(7, 154) = 0.62, p = .73$ ). Nous avons réalisé des tests de student pairés comme test post-hoc pour avoir plus de précisions sur l'impact de l'occlusion et l'interaction entre occlusion et apprentissage. Ces tests post-hoc ainsi que la figure 5.9 montrent que au début de la présentation de la trajectoire la tendance de prédiction diminue (les participants prédisent moins de lancers francs réussis), celle-ci remonte à partir de  $t=300$ ms, puis se stabilise à partir de  $t=600$ ms. De plus, lors de l'apprentissage les participants ont plus tendance de prédire de lancer-francs ratés pour les instants  $t=150$  et  $900$ ms et l'inverse pour l'instant  $t=600$ ms.

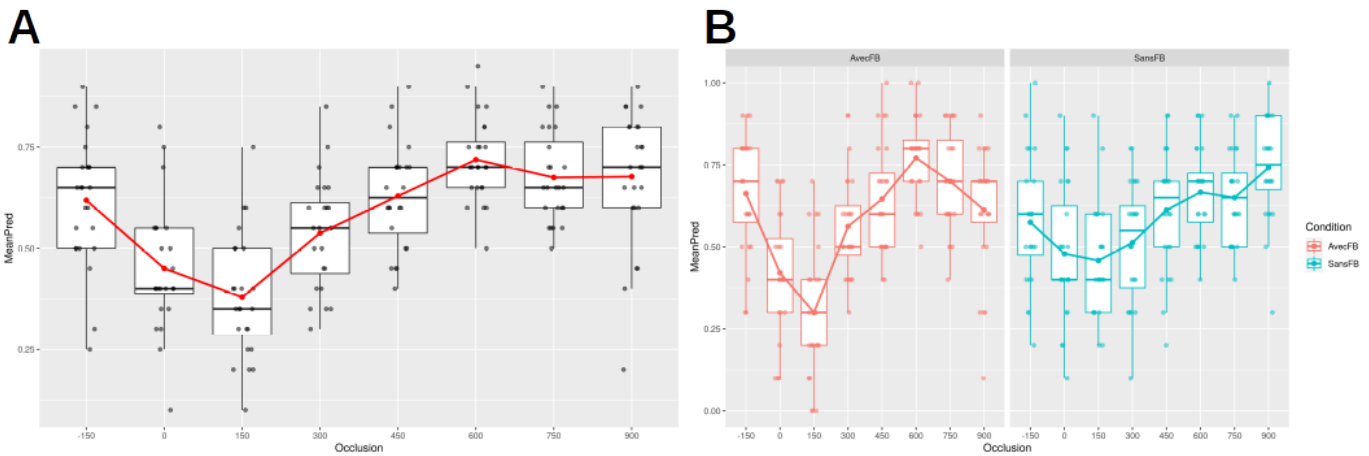


FIGURE 5.9 – Panneau A : Évolution de la tendance de prédiction en fonction du temps de présentation de la trajectoire. Panneau B : Comparaison entre les résultats pendant et après apprentissage de l'évolution de la tendance de prédiction en fonction du temps de présentation de la trajectoire.

#### 5.4.4 Comparaison des résultats

Comme pour la précédente étude, nous avons voulu comparer les résultats entre les participants experts, non-experts et SNN. Nous avons analysé les performances (erreur de prédiction et tendance de prédiction) à l'aide de deux ANOVA distinctes, avec la visibilité (de -150 à +900ms) comme variables intra-sujet et le type de participant (expert ou non, SNN) comme variables inter-sujet. Les résultats montrent que la visibilité ( $F(7, 217) = 4.22, p < .001$ ) et le type de participant ( $F(2, 31) = 22.70, p < .001$ ) ont une influence sur la précision. Il n'y a cependant pas d'interaction entre ces variables ( $F(14, 217) = 1., p = .45$ ). Les test post-hoc ainsi que la figure 5.10, montrent des résultats similaires à ceux des participants humains concernant l'impact de la visibilité et montrent que le SNN est meilleur que les experts qui sont eux-même meilleurs que les non-experts.

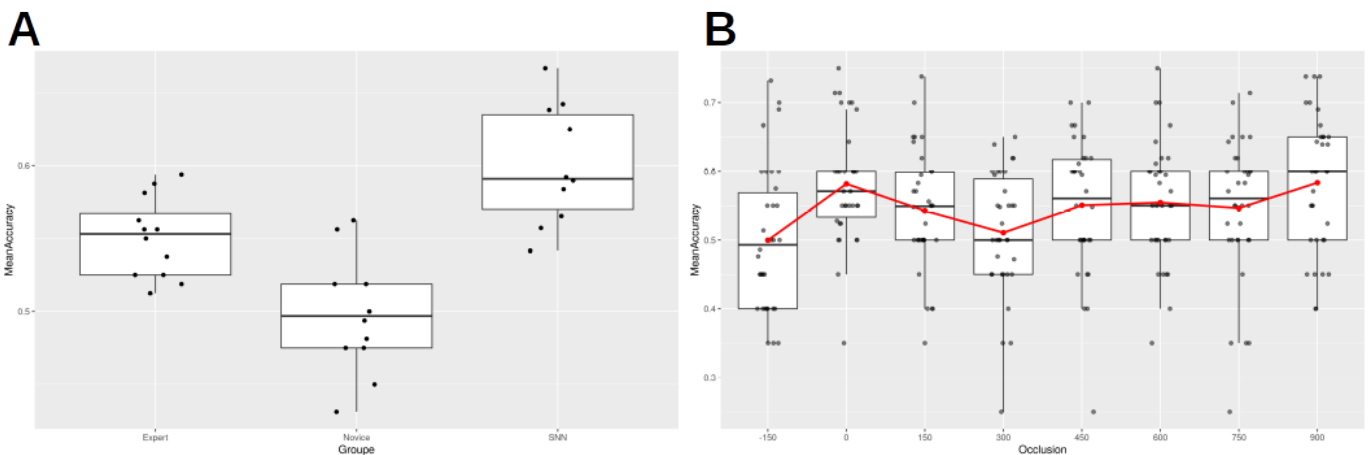


FIGURE 5.10 – Panneau A : Précision moyenne des différents groupes testés (SNN ;Expert ;Non-experts). Panneau B : Évolution de la précision de la prédiction indépendamment du groupe testé.

Concernant la tendance de prédiction, les résultats montrent que la visibilité ( $F(7, 217) =$

22.07,  $p < .001$ ) et le type de participant ( $F(2, 31) = 6.08, p = .006$ ) ont une influence sur la précision ainsi que leur interaction ( $F(14, 217) = 2.89, p < .001$ ). Les post-tests ainsi que la panneau A de la figure 5.11 montrent que le SNN a tendance à prédire moins de lancers francs réussis que les non-experts, qui eux-mêmes en prédisent moins que les experts. Concernant l'occlusion, on voit sur la figure B et grâce aux post-tests, que durant jusqu'à  $t=150$ ms, la tendance de prédiction diminue (de plus en plus de lancers francs ratés sont prédits), puis augmente jusqu'à  $t=450$ , et se stabilise à partir de  $t=600$ ms. En regardant l'interaction de ces deux variables (panneau C), on voit qu'à l'instant  $t=-150$ ms le SNN prédit beaucoup plus de lancers francs ratés que les autres participants, ainsi qu'à  $t=600$  et  $750$ ms.

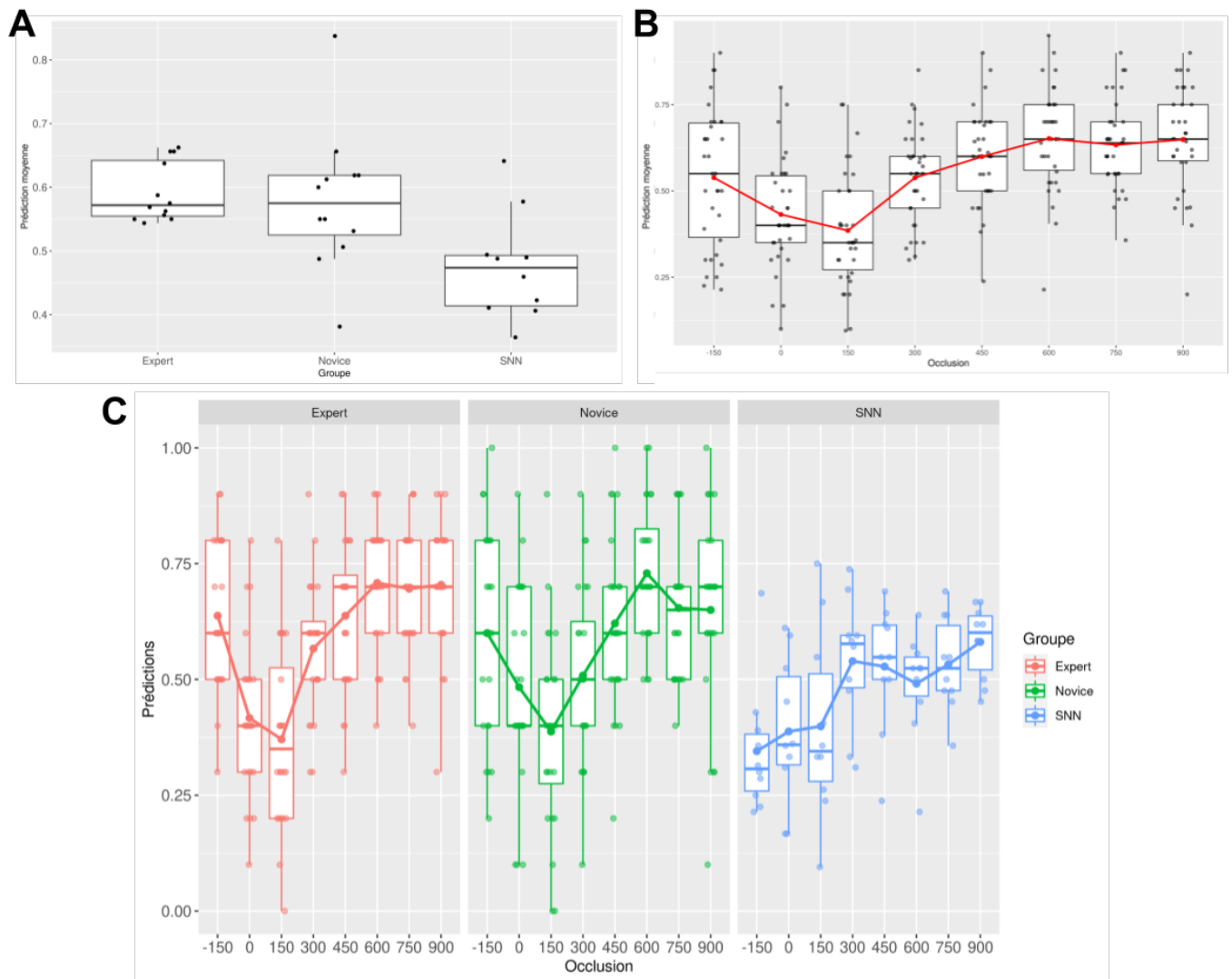


FIGURE 5.11 – Panneau A : Prédiction moyenne des différents groupes testés (SNN, Expert, Non-experts). Panneau B : Évolution de la prédiction indépendamment du groupe testé. Panneau C : Évolution de la prédiction au cours du temps en fonction du groupe testé.

---

## 5.5 Discussion

Les résultats de prédiction du SNN paraissent dans un premier temps peu convaincants. En effet, celui-ci effectue un résultat moyen de 60.01% de réussite, bien que cette valeur soit supérieure à 50%, on voit que le SNN a du mal à améliorer sa prédiction au cours du temps, contrairement à la précédente étude. Cependant, en comparant les résultats avec les participants experts et non-experts, on voit que le SNN est meilleur que ceux-ci et que l'évolution de l'erreur est similaire (peu de variation, puis une amélioration à  $t=900\text{ms}$ ). Cette différence d'évolution des performances entre les deux tâches peut s'expliquer par le type et la complexité de la tâche. Lors de la précédente étude (chapitre 4), l'objectif était de définir le point de réception de la balle, différents lancers avec différents profils étaient donc présentés. La position spatiale de la balle était donc une information pertinente permettant d'expliquer l'amélioration des performances. En effet, il n'y avait pas de cibles précises à viser, il fallait même au contraire chercher à diversifier les trajectoires (point de réception, paramètres initiaux, etc.). Lors de cette étude, la balle arrive à la même position ou presque et les différentes trajectoires se ressemblent énormément. En effet, les différents joueurs avaient pour objectif de marquer le panier. Ainsi, même les lancers qui échouent sont très ressemblants à ceux qui rentrent. La position spatiale du ballon n'apporte alors que très peu d'informations. De plus, cette tâche est d'autant plus complexe car un lancer franc réussi se caractérise par un ballon qui arrive à une bonne position spatiale (position 3D panier) avec un bon angle (pour ne pas taper le cercle). Contrairement à la précédente étude (chapitre 4) l'information de profondeur du ballon est importante car si le lanceur lance le ballon trop à droite ou à gauche (plus ou moins proche par rapport à la caméra), le ballon touchera le cercle et ne rentrera pas. De plus le ballon doit arriver selon un certain angle, paramètre non pris en compte lors de la précédente étude. Ainsi cette étude est considérablement plus difficile que la précédente, expliquant la différence des performances et de leurs évolutions au court du temps de présentation.

Quelques études ont déjà évalué les différences de performances entre joueurs experts et novices dans une tâche similaire à la notre (AGLIOTI et al. 2008 ; Y. WU et al. 2013 ; Y. LI et FENG 2020). Cependant, les résultats diffèrent selon les études comme expliqué dans la section 2.6. (AGLIOTI et al. 2008) par exemple montre un fort impact du temps de présentation de la trajectoire sur les performances, les participants arrivant à un taux de réussite proche de 100% en fin de trajectoire. (Y. LI et FENG 2020) montre des résultats plus proches des nôtres, avec peu d'évolution au cours du temps. Nous supposons que cela vient des stimuli utilisés. Alors que nos tireurs devaient marquer à chaque lancer, il nous semble que les tireurs dans l'étude de (AGLIOTI et al. 2008) avaient pour instruction de mettre ou de rater, volontairement, et visiblement, leurs tirs. Cette supposition est appuyée par la figure 5.12, montrant une très forte différence entre les lancers francs ratés et réussis dans l'étude de (AGLIOTI et al. 2008), alors que dans les trajectoires que nous présentons, cette différence est très minime. Ainsi, les tirs de l'étude de (AGLIOTI et al. 2008) sont très précocement et visiblement différents, distincts, entre les succès et échecs, au contraire des nôtres.

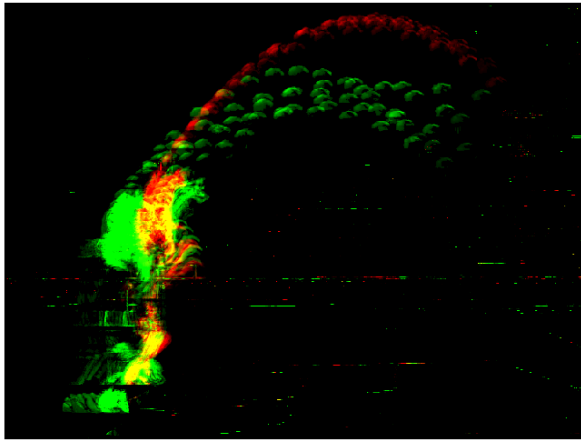


FIGURE 5.12 – Comparaison des trajectoires entre l'étude de (AGLIOTI et al. 2008) (gauche) et nos trajectoires. Vert = lancer raté; Rouge = lancer réussi

Nos résultats s'approchent davantage de ceux de (Y. LI et FENG 2020) en pourcentage moyen de réussite. Cependant, (Y. LI et FENG 2020) montre une amélioration significative des performances en fin de trajectoire, alors que dans nos résultats la précision à  $t=900\text{ms}$  est seulement supérieure de la précision à  $t=300\text{ms}$ . De plus, la précision à  $t=0\text{ms}$  est elle aussi supérieure, montrant un effet probable du mouvement du joueur qui lance. La faible résolution de la caméra NeuroSoc (120x128 pixels) peut aussi expliquer les différences de résultats avec les autres études (AGLIOTI et al. 2008; Y. WU et al. 2013; Y. LI et FENG 2020).

# Chapitre 6

## Optimisation de la caméra NeuroSoC

Comme expliqué dans la section 2.8, un des objectifs de cette thèse était de finaliser le développement de la caméra NeuroSoc déjà initié par Yumain et différents stagiaires pour l'analyse de trajectoires d'objets. Notre développement nous a permis de tester et valider le fonctionnement et la pertinence de cette caméra en l'évaluant sur une tâche d'analyse et de prédiction de trajectoire de balles en utilisant un réseau de neurones à spike. Une première partie de cette thèse a été consacrée à travailler avec Yumain sur le debuggage et la mise en place d'une première version stable de cette caméra pour notre cadre applicatif. Comme tout travail de recherche et développement, cette version n'est pas définitive et présente toujours quelques points d'améliorations qui ont pu être identifiés durant l'utilisation de la caméra au cours de cette période de thèse. Le premier objectif de ce chapitre est de revenir sur ces quelques points identifiés et de proposer une solution pour améliorer la caméra. Deux points sont principalement notés qui impactent la génération d'événements et vont être dépendants de la vitesse et du contraste du mouvement. Le second objectif est d'évaluer la précision temporelle des événements générés par la caméra NeuroSoc. Ces points semblent être critiques car ils impactent des points qui font partie des arguments principaux des caméras événementielles.

### 6.1 Événements entourés de polarités opposées

Comme expliqué dans la partie 3.1.1, la génération d'événements de la NeuroSoc se divise en deux parties. Une première, le filtrage temporel qui se caractérise par une différence d'image. La seconde, le filtrage spatial, qui se caractérise par une convolution, de différence de gaussienne (DoG) dans notre cas pour mimer le fonctionnement du LGN. Ce type de filtre centre/pourtour permet l'extraction de bordure quelle que soit son orientation. Cependant, ce filtrage peut amener des effets de bords plus ou moins forts selon la vitesse du mouvement enregistré. Lors de l'application du filtre spatial sur une image, les zones unies ou sans contour seront caractérisées par des valeurs proches de zéro. Alors que les contours seront caractérisés par un passage d'une valeur hautement positive à hautement négative (figure 6.1). Le contour peut ainsi être extrait en conservant les valeurs supérieures à une valeur de seuil.

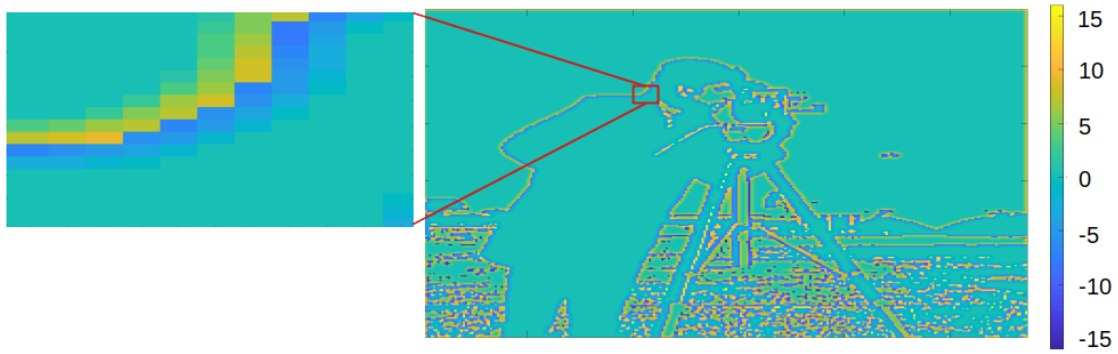


FIGURE 6.1 – Filtrage par convolution d'un DoG

Ainsi, si l'on devait extraire des événements de cette image, on obtiendrait des événements ON juxtaposés à des événements OFF. La première phase de filtrage temporel permet déjà d'effectuer une extraction de contour. Ainsi, lorsque appliqué sur la différence d'image, le filtrage spatial génère majoritairement des valeurs positives (et donc des événements ON) lorsqu'il y a une augmentation de luminosité détectée par le filtrage temporel et des valeurs négatives lors d'une diminution de luminosité. Ce comportement est celui attendu, en effet, si on prend un comme stimulus, une forme sombre se déplaçant de la gauche vers la droite, devant un arrière plan clair, on s'attend à obtenir des événements de type OFF sur le côté droit de l'objet en mouvement et des événements ON de l'autre côté.

Cependant des valeurs opposées à la polarité du pixel après filtrage temporel se trouvent autour de celui-ci lorsque l'on applique le filtrage spatial. Ces valeurs sont dues à la partie négative du filtre DoG lorsqu'il est appliqué sur les pixels avoisinants. Ces valeurs restent en valeur absolue relativement faibles comparées à la valeur du pixel sur lequel est détecté une variation de luminosité et ces événements à polarité inversée peuvent donc ne pas dépasser la valeur de seuil et donc ne pas être générés. De plus, ces valeurs sont proportionnelles à la variation de luminosité et donc au contraste entre l'objet en mouvement et l'arrière plan, ce qui peut être contraignant dans une scène avec de nombreux contrastes différents, une définition d'un seuil pour éviter les événements à polarité opposée serait donc difficile à déterminer. L'écart entre l'événement "attendu" et l'événement à polarité opposé "non attendu" se réduit plus la vitesse de l'objet s'agrandit. En effet, comme on le voit sur la figure 6.2, pour un déplacement d'un pixel par image, le ratio entre les valeurs absolues des événements "attendus" et les autres sont de 3 :1, pour une translation de 2 pixels par image, le ratio est 2 :1 et pour une translation de 3 pixels, les valeurs absolues des types de valeurs générées sont égales, il n'est donc plus possible de les séparer avec une phase de seuillage. Pour des mouvements plus lents, voir figure 6.2 et 6.4, les rapports sont beaucoup plus élevés, il est donc plus simple de les séparer et supprimer les événements à polarités opposées. Cet effet de bord est donc dépendant de la vitesse du stimuli. La caméra fonctionnant à 240 FPS pour une résolution de 128x120 pixels, il est donc rare d'avoir des objets qui génèrent des translations de plus de deux pixels par image. Lors de nos acquisitions, les mouvements étaient suffisamment lents et les variations de contraste suffisamment faibles pour pouvoir définir un seuil qui permettait de n'extraire que (ou presque) des événements "attendus" (c.a.d des événements ON/OFF lors d'une augmentation/diminution de luminosité).

Pour des objets se déplaçant à des vitesses encore plus rapides (translation supérieure à deux pixels par image), ce problème pourrait facilement être résolu en séparant le traitement des pixels à polarité positive des pixels à polarité négative à la suite du filtrage temporel. Le filtrage spatial serait ensuite appliqué sur chacune des deux images générées (une pour chaque polarité, voir figure 6.3). Seules les valeurs positives et dépassant le seuil généreraient des événements ON,

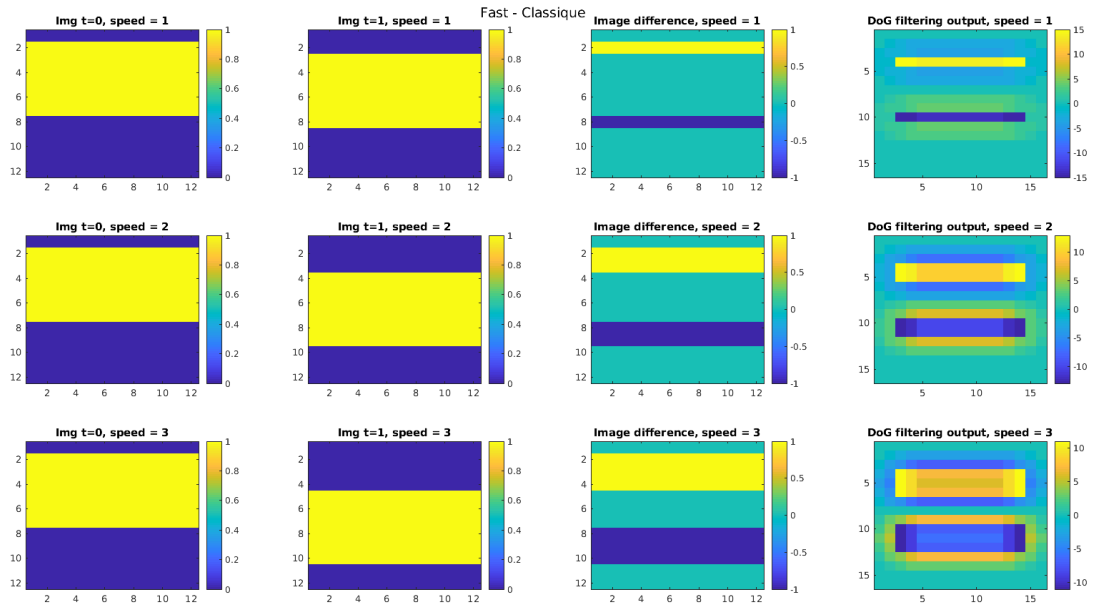


FIGURE 6.2 – Sortie du filtrage spatio-temporel lors d'un mouvement rapide

après application du filtre DoG sur l'image positive. Et inversement, seuls les valeurs négatives et dépassant le seuil en valeur absolue génèreraient des événements OFF, après application du filtre DoG sur l'image négative. En séparant les deux polarités à la suite du filtrage temporel, on s'assure de ne conserver que des événements ON provenant d'augmentation de luminosité en ne conservant que les valeurs positives et inversement pour les événements OFF. La figure 3.3 récapitulant le filtrage spatio-temporel actuellement implémenté sur la caméra deviendrait donc comme représenté sur la figure 6.3.

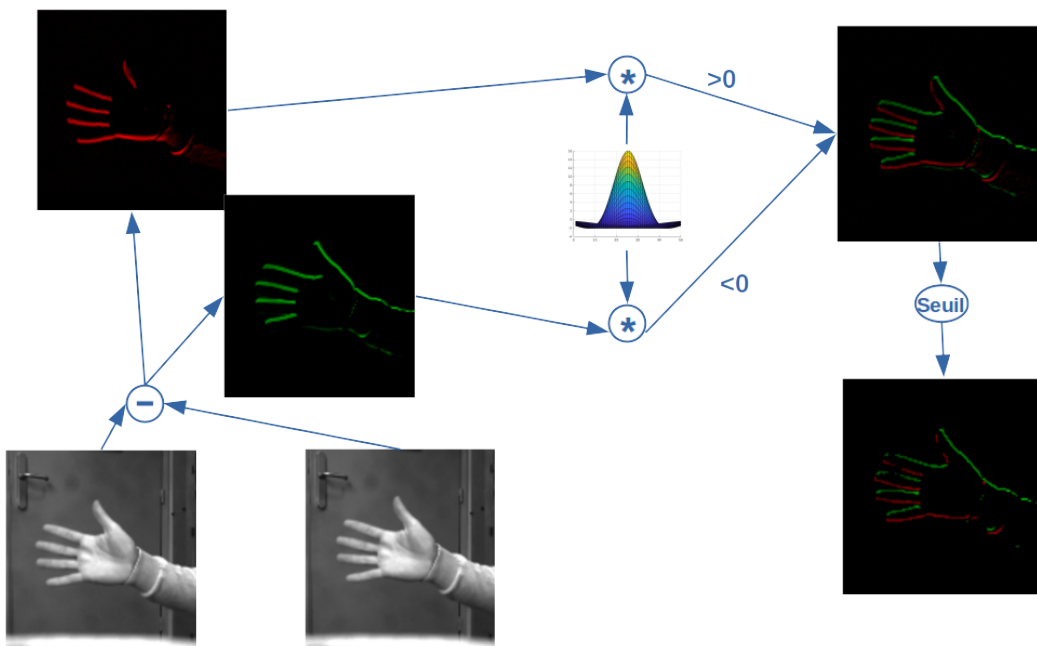


FIGURE 6.3 – Traitement spatio-temporel proposé



## 6.2 Détection de mouvements lents

A l'opposé de la précédente section, les mouvements lents peuvent être difficilement détectés par la caméra. Lorsqu'un mouvement est lent, la variation de luminosité entre pixels de deux images consécutives peut être faible. En effet, si on prend un objet avec un niveau de gris égal à 1 et un arrière plan égal à 0, pour une vitesse de translation d'un pixel par image, la variation de luminosité entre deux images consécutives est de un à l'endroit du mouvement. Pour une vitesse de translation d'un pixel toutes les deux images, la variation est de 0.5 et pour une vitesse d'un pixel toutes les cinq images, la variation devrait être de 0.2. Ainsi la variation de luminosité pour un pixel (à l'endroit du mouvement) entre deux images consécutives est de :

$$\Delta I(x, y) = (I_{Obj} - I_{Back}) \times \left( \sqrt{vx_{Obj}^2 + vy_{Obj}^2} \right) \quad (6.1)$$

- $I_{Obj}$  et  $I_{Back}$ , le niveau de gris de l'objet en mouvement et du l'arrière plan
- $vx_{Obj}$  et  $vy_{Obj}$  la vitesse de translation en pixel de l'objet entre deux images consécutives

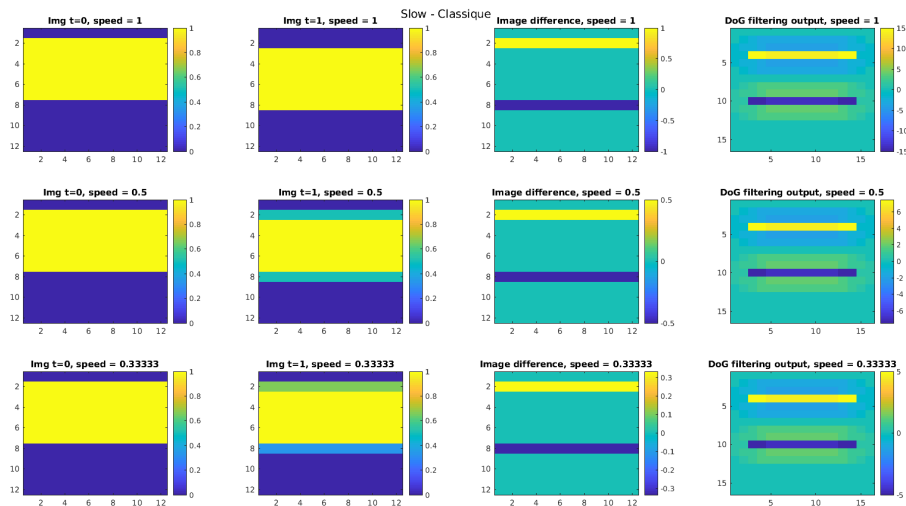


FIGURE 6.4 – Sortie du filtrage spatio-temporel lors d'un mouvement lent. L'intensité des valeurs retournée par le filtrage est proportionnelle à la vitesse du stimulus.

Encore un fois, on voit que cette situation est dépendante de la vitesse de translation du mouvement en pixels  $\underline{ET}$  du contraste entre l'arrière plan et l'objet en mouvement. Ainsi, comme le montre la figure 6.4, plus le mouvement est lent, plus l'intensité des valeurs retournées par le filtrage sont basses. Le principal problème de la solution actuellement utilisée est que le filtrage temporel se définit par une différence d'images entre deux d'images consécutives. Aucune information sur les variations de luminosité des images précédentes n'est conservé.

Une solution serait de garder une trace de l'activité du filtrage spatio-temporel, en mimant le comportement d'un neurone LIF (figure 6.5). Ainsi la sortie du filtrage spatio-temporel est additionnée aux potentiels de membrane des neurones. Si le potentiel de membrane d'une neurone dépasse la valeur de seuil, il décharge et son potentiel de membrane est remis à zéro. Sinon sa valeur est conservée et une fuite est appliquée à l'image suivante afin que lorsqu'il n'y a pas de mouvement, le neurone revient petit à petit à zéro. Si le mouvement est trop lent et donc ne permet pas de déclencher un événement, les variations de luminosités détectées par le filtrage vont s'accumuler au fur et à mesure des images, jusqu'à générer un événement.

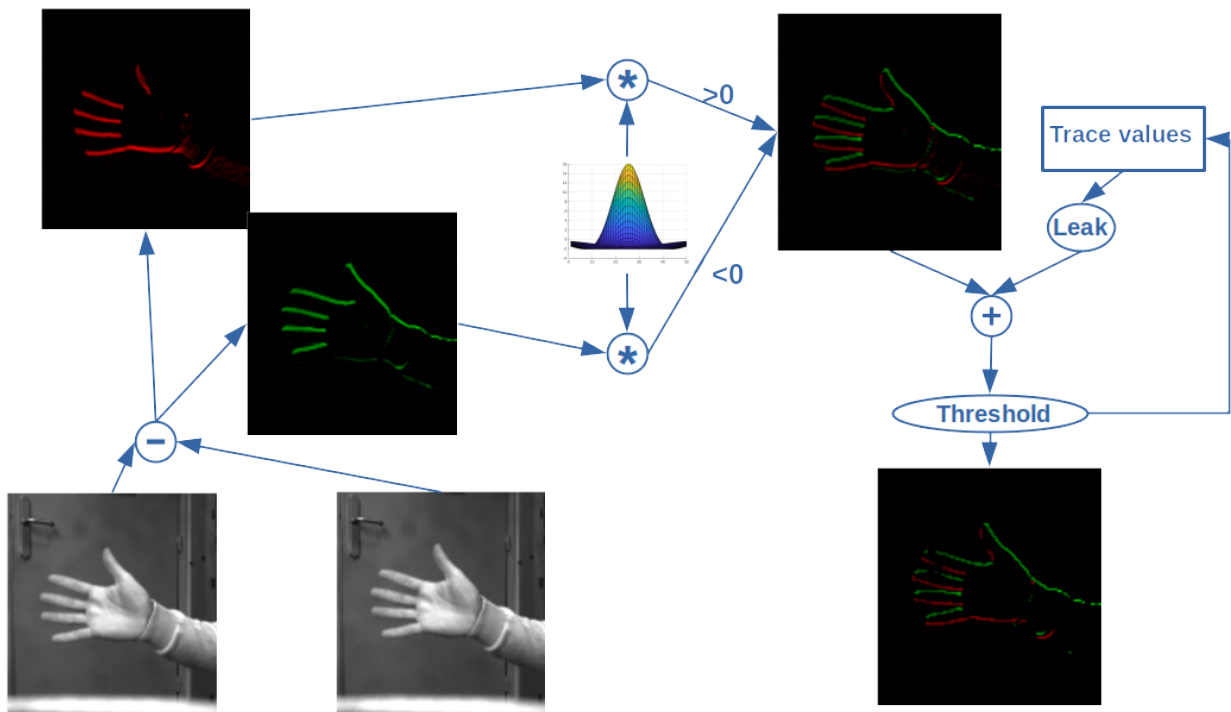


FIGURE 6.5 – Traitement spatio-temporel proposé avec trace du mouvement

## 6.3 Résultats

### 6.3.1 Impact sur la génération d'événements

Afin d'évaluer l'impact des ces différents processus, nous les avons dans un premier temps appliqués sur un stimulus simulé, en utilisant une barre en rotation. L'avantage du mouvement de rotation est qu'il permet d'avoir les deux problèmes exprimés ci-dessus dans un même mouvement, car la translation de pixels au centre de l'image (et donc de la barre en rotation) est faible et augmente au fur et à mesure qu'on s'écarte du centre. Ainsi ce stimulus permet de présenter un large panel de vitesses.

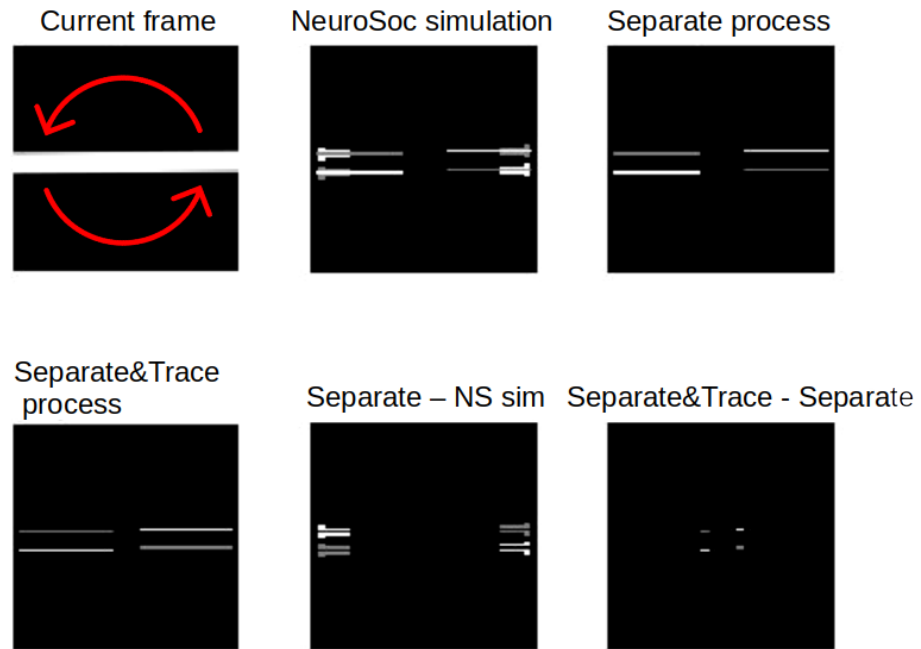


FIGURE 6.6 – Exemple des différents processus sur un stimulus de barre en rotation

On voit ainsi sur la figure 6.6 ci-dessus, les événements émis pour les 3 différents processus : une simulation du filtrage actuellement implémenté dans la caméra NeuroSoc (*NeuroSoc simulation*), la méthode exposée dans la section 6.1 (*Separate process*) et la méthode de la section 6.2 (*Separate&Trace process*). On voit ainsi que la méthode qui consiste à séparer les deux polarités à la suite du filtrage temporel permet de supprimer les événements à polarités opposées. Ce filtrage est représenté dans le panneau *Separate - NS sim*, qui affiche la différence des événements entre *NeuroSoc simulation* et *Separate process*. Ces événements se retrouvent principalement aux extrémités de la barre en rotation car c'est là que le mouvement y est le plus rapide. A l'opposé, au centre de l'image le mouvement y est beaucoup plus lent, on voit ainsi que sur *NeuroSoc simulation* et *Separate process* le mouvement n'est pas détecté et aucun événement n'est généré. En ajoutant le processus de trace (*Separate&Trace process*), on voit que plus d'événements sont générés au centre de la barre, comme indiqué sur le panneau *Separate&Trace - Trace* qui représente la différence entre *Separate&Trace process* et *Separate process*, sur lequel on y voit les événements générés par le processus de trace.

Les différents processus ont été testés sur des acquisitions faites par la caméra NeuroSoc, avec comme stimulus une balle noire qui passe devant la caméra à différentes vitesses. On voit ainsi sur la figure 6.7 qui correspond à un mouvement de balle assez rapide, que la méthode de séparation des polarités permet de supprimer les événements non attendus et que l'ajout de la trace permet d'ajouter quelques événements à l'ombre de la balle (qui a un contraste faible avec l'arrière plan). La figure 6.8, qui est appliquée sur un mouvement assez lent, montre que les événements sont principalement concentrés sur les parties latérales de la balle et peu ou pas sur le dessus/dessous de la balle ainsi que sur l'ombre de celle-ci. L'ajout de la méthode de trace permet de rétablir ces événements et d'obtenir un résultat proche de la figure 6.7 et d'avoir un traitement qui est donc plus indépendant du contraste et de la vitesse du mouvement.

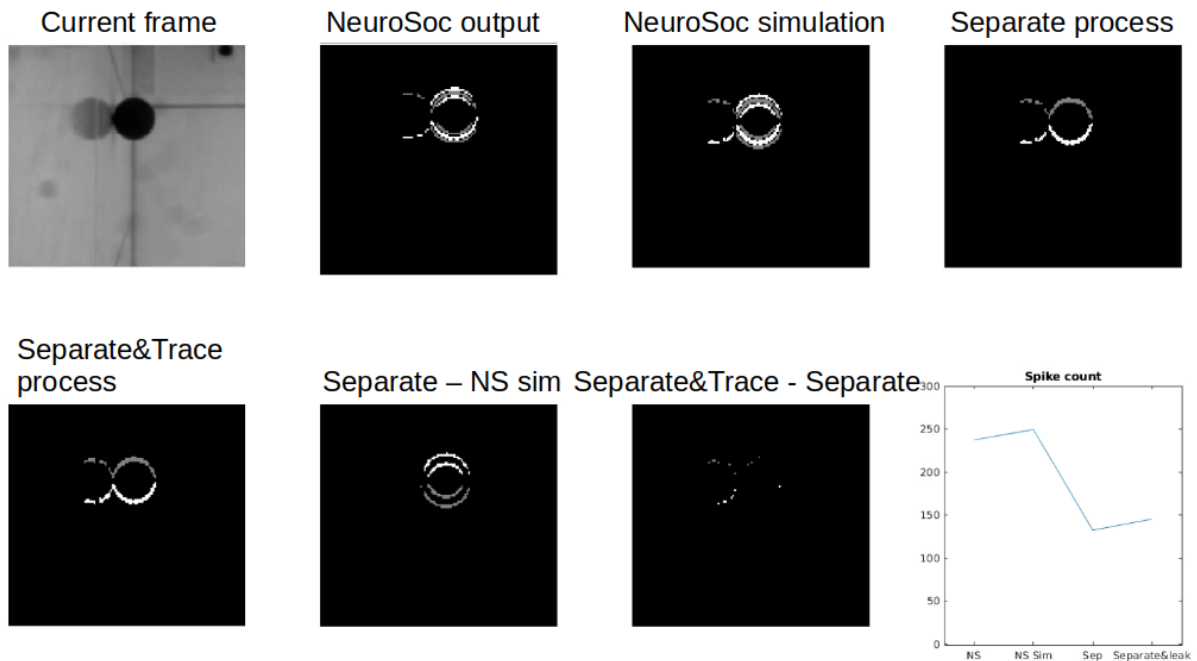


FIGURE 6.7 – Exemple des différents processus appliqués à la vidéo extraite par la caméra NeuroSoc d’une balle se déplaçant rapidement

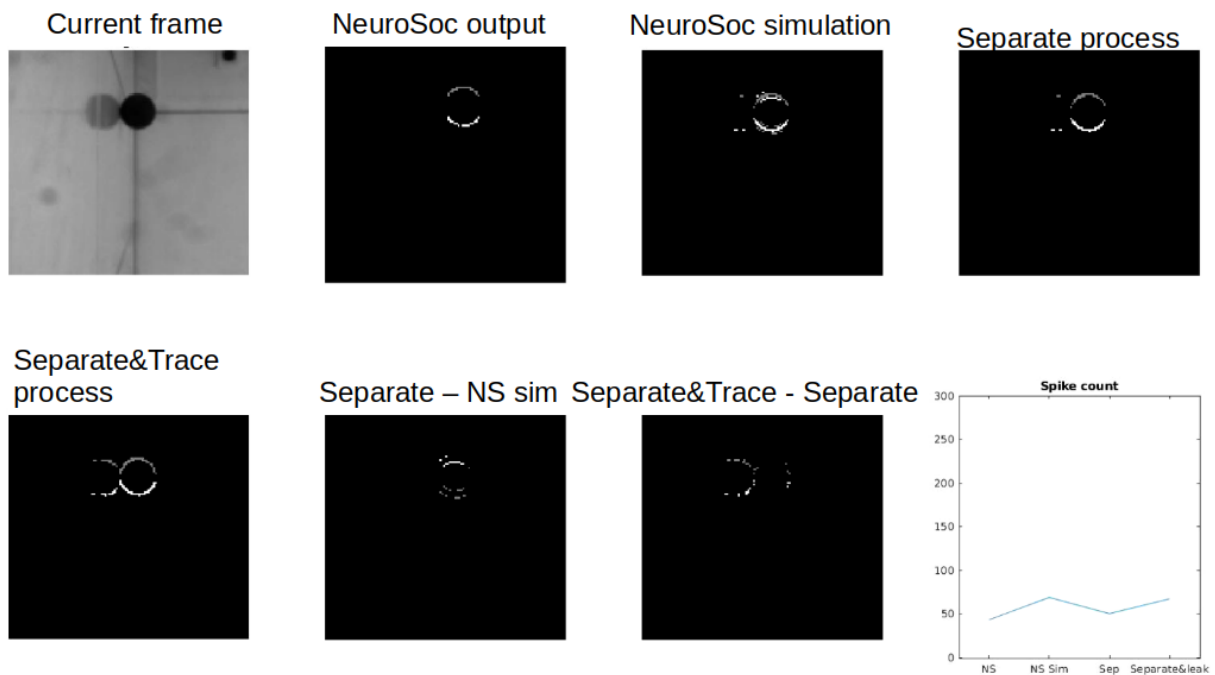


FIGURE 6.8 – Exemple des différents processus appliqués à la vidéo extraite par la caméra NeuroSoc d’une balle se déplaçant lentement

---

### 6.3.2 Précision temporelle de la caméra NeuroSoc

Une des principales différences de la caméra NeuroSoc par rapport aux autres caméras événementielles est le fait qu'elle utilise une caméra *frame-based*. Ainsi, la précision temporelle des événements devrait donc être dépendante du FPS de la caméra et moins précise qu'une caméra du type DVS qui peut émettre des événements avec une précision temporelle de l'ordre de la  $\mu\text{s}$ . Afin d'évaluer la précision temporelle des événements générés par la caméra, nous avons simulé une barre verticale, se déplaçant de la gauche vers la droite à différentes vitesses. Puis, nous avons simulé l'extraction d'événements effectuée par la caméra, avec différentes valeurs de seuil pour la génération des événements. Les événements étaient ensuite répartis entre les frames en utilisant une règle d'intensité à latence similaire à celle utilisée pour les précédentes études (section 3.1.1). Nous avons ensuite calculé l'erreur de précision temporelle pour évaluer la précision de la caméra NeuroSoc, en utilisant le processus de génération d'événements actuellement implémenté sur la caméra NeuroSoc ainsi que celui proposé ci-dessus, qui conserve une trace des précédentes valeurs.

L'information d'intensité de l'événement transmis par la caméra NeuroSoc, qui est ensuite utilisée pour répartir les événements entre deux frames successives, est codée sur 7 bits. Ainsi, il peut y avoir 128 latences temporelles possibles entre deux frames et la précision temporelle de la caméra NeuroSoc est de  $\frac{1}{128 \times FPS}$ , soit une précision temporelle de 0.03ms lors de l'utilisation de la caméra à 240 FPS.

La précision temporelle calculée ci-dessus ne suffit pas pour évaluer la qualité de l'information temporelle transmise par les événements générés. Pour cela, nous avons aussi calculé l'erreur temporelle entre des événements simulés (en utilisant le processus de la caméra), et la temporalité exacte attendue (*ground truth*), qui est l'espace de temps entre deux pixels latéraux égale à  $1/\text{vitesse}$  de la barre. L'erreur absolue moyenne entre les instants temporels de événements simulés et le *ground truth* est calculée, pour différentes vitesses de stimuli et différentes valeurs de seuils pour déclencher la génération d'événements. Seulement le premier événement émis par un neurone et ceux de types ON sont pris en compte.

La figure 6.9 nous permet de voir que la précision temporelle des événements émis est bien plus précise que  $1/\text{FPS}$ . En effet, on voit que quels que soient la vitesse du stimulus ou le seuil, cette erreur temporelle est inférieure à un. Une valeur égale à un voudrait dire une erreur égale à la période entre deux frames, soit  $1/\text{FPS}$ . Ainsi, on voit que l'erreur temporelle est inférieure au temps séparant deux frames.

La figure 6.9 ainsi que le panneau D de la figure 6.10 mettent en avant l'impact du seuil de déclenchement d'un événements sur l'erreur temporelle. Ainsi, plus ce seuil sera haut, plus l'erreur temporelle aura tendance à être élevée.

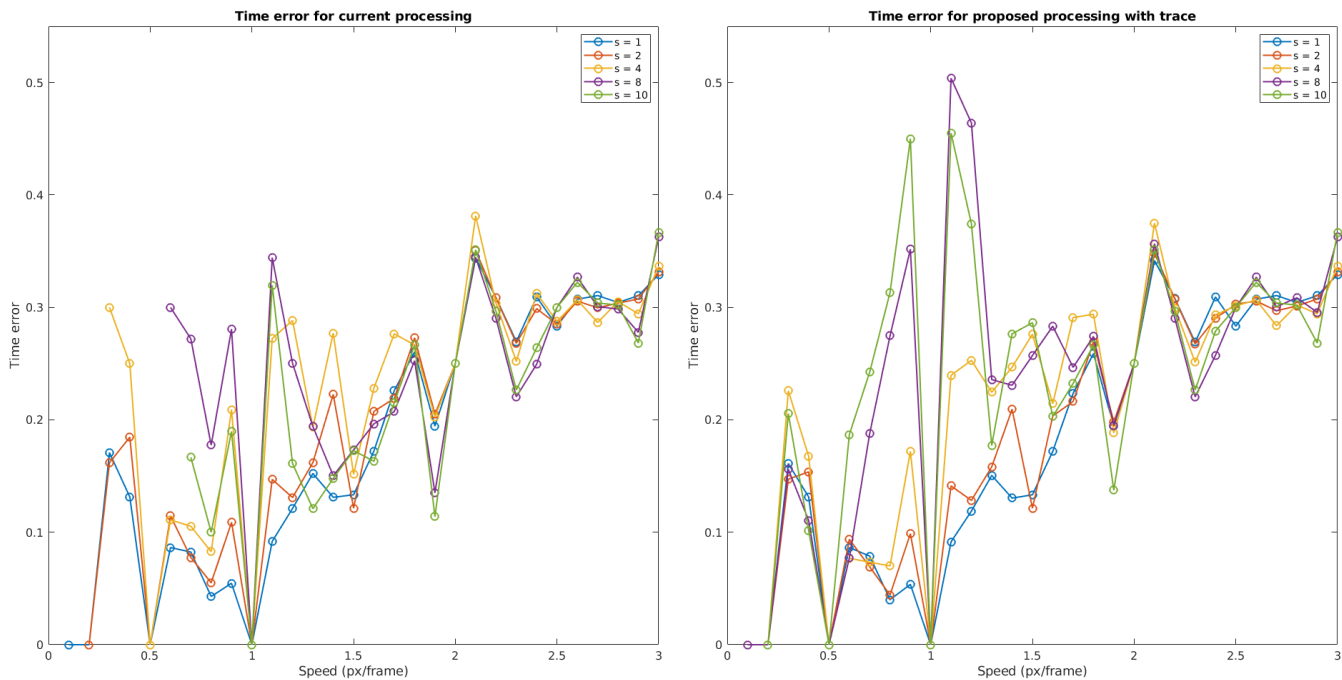


FIGURE 6.9 – Précision temporelle des événements générés en fonction de la valeur de seuil et de la méthode d'extraction utilisée. Le graphique de gauche indique l'erreur moyenne absolue temporelle calculé en fonction de la vitesse du stimulus et de la valeur de seuil utilisée pour déclencher les événements, en utilisant la méthode d'extraction actuelle. Les résultats pour la méthode utilisant les traces des frames précédentes sont présentés sur la figure de droite.

De plus, en comparant les deux graphiques de la figure 6.9 ainsi que le premier graphique de la figure 6.10, panneau A, on voit que selon le type de processus utilisé pour l'extraction d'événements, l'erreur temporelle varie. Cependant, en regardant le panneau C de la figure 6.10, on voit que tous les neurones ne déchargent pas selon la vitesse du stimulus et le seuil utilisé. En utilisant la méthode gardant la trace de l'activité (section 6.2) on voit que de nombreux neurones supplémentaires déchargent. Les panneaux A et B de la figure 6.10 nous montrent l'écart d'erreur temporelle entre les deux méthodes (celle actuellement utilisée et celle utilisant la trace). Cependant, le panneau B calcule l'erreur seulement pour les neurones ayant déchargé en utilisant la méthode actuelle d'extraction d'événements. Ainsi on voit qu'en prenant seulement ces événements spécifiques, l'erreur temporelle est inférieure en utilisant la méthode avec la trace. Les valeurs négatives du panneau A montrent des erreurs temporelles supérieures pour la méthode utilisant la trace. Ces erreurs proviennent des événements générés par cette méthode (qui n'étaient pas générés en utilisant la méthode actuelle). Ceux-ci sont générés plus tard, car il faut attendre que la trace se cumule pour qu'ils soient générés et sont donc générés en retard. Ainsi, la méthode utilisant la trace d'activité permet d'améliorer ou conserver l'erreur temporelle des événements générés par les deux méthodes et de générer des événements qui n'arrivaient pas à dépasser le seuil en utilisant la méthode de génération actuelle, avec cependant une erreur temporelle plus élevée.

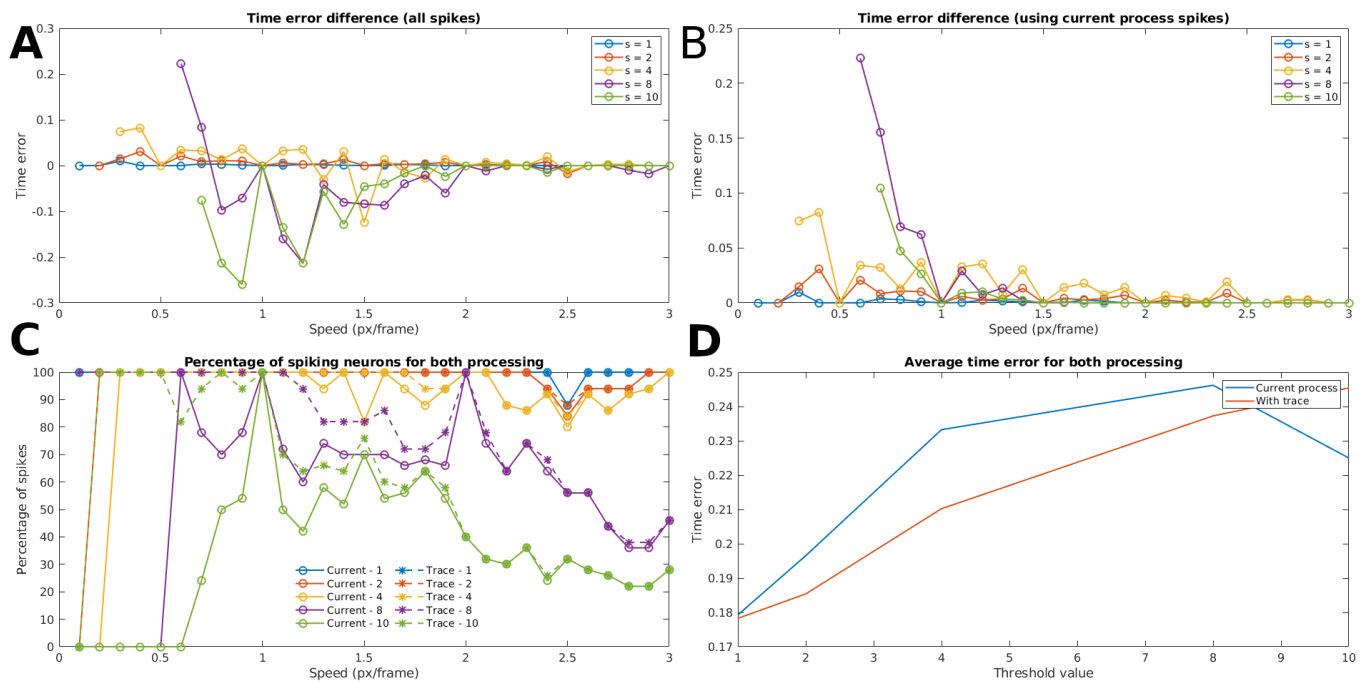


FIGURE 6.10 – Impact des différentes méthodes sur la précision temporelle des événements générés. Panneau A : Différence d’erreur temporelle entre la méthode actuelle et celle utilisant une trace de l’activité précédente. Une valeur positive indique que l’erreur est supérieure pour la méthode actuelle et inversement (valeurs du panneau de gauche de la figure 6.9 moins celles du panneau de droite). Panneau B : Similaire au panneau A, cependant, seulement les événements générés par la méthode actuelle sont pris en compte. Ainsi, les événements supplémentaires générés pas la méthode utilisant les traces ne sont pas comptés. Panneau C : Quantité d’événements générés par méthode d’extraction d’événements et par valeur de seuil. Panneau D : Erreur temporelle moyenne en fonction de la valeur de seuil et de la méthode d’extraction utilisée.

## 6.4 Discussion

Les deux solutions permettent de palier les problèmes exposés dont peut pâtir la caméra. Elles comportent cependant quelques points négatifs, notamment sur la quantité de mémoire supplémentaire nécessaire pour les appliquer. En effet pour la première solution, il est proposé de séparer en deux la phase de filtrage temporel, une pour chaque polarité, doublant ainsi le nombre d’opérations nécessaires. La seconde solution nécessite de garder constamment en mémoire une trace de l’activité de la sortie du filtrage et de le mettre à jour à chaque nouvelle image, impliquant ainsi des opérations et de la mémoire supplémentaire.

Les deux problèmes exposés dépendent principalement du contraste entre l’arrière plan et l’objet mouvant, et de la vitesse de celui-ci. Ce deuxième point dépend en réalité de la vitesse de translation de l’objet en pixel sur les images enregistrées. Celle-ci dépend donc de :

- la vitesse physique de l’objet
- sa distance de l’objectif : plus l’objet est loin, plus la translation en pixel est faible
- du FPS de la caméra : avec un faible fps, l’objet se déplace plus entre deux images générant ainsi une plus grande translation de pixels
- de la résolution d’image : en augmentant la résolution, on augmente la translation du

---

mouvement

La version utilisée de la caméra NeuroSoc est basée sur des images de 128x120 pixels avec un FPS de 240. Le premier problème était donc peu fréquent, car avec un fps élevé et une faible résolution, la translation d'un mouvement entre deux images était rarement élevée. Cependant, les vitesses faibles étaient plus difficilement détectées et généraient moins d'événements, comme par exemple lors des moments où la balle atteignait son niveau le plus haut et les débuts de phase de redescende lors des échanges de passes (chapitre 4).

Des capteurs avec de fortes résolutions spatiales et des FPS réduits, comme la nouvelle caméra proposée par Yumain (*CAIMAN CAMERA* 2021), devraient plus facilement détecter des mouvements plus lents, mais plus facilement observer le premier problème exposé.

On pourrait remettre en question l'utilité d'ajouter un filtrage spatial à la suite du filtrage temporel, alors que celui-ci permet déjà de détecter de façon générale des informations proches de la somme des deux filtrages (c.a.d. des bords en mouvement). Le filtrage spatial permet principalement de réduire le bruit, d'extraire les contours et donc de réduire le nombre d'événements générés par la caméra. De plus, la caméra NeuroSoc propose d'implémenter différents filtres de différentes tailles, sur des résolutions d'images différentes, permettant ainsi d'extraire des informations spatio-temporelles directement en sortie de la caméra (différentes orientations, fréquences spatiales, etc.).





# Chapitre 7

## Discussion et perspectives

Les différents travaux présentés dans ce manuscrit ont permis de faire les premiers pas d'une analyse et prédiction de trajectoires de balles tout en essayant de respecter des contraintes de bio-inspiration. Ainsi, nous avons montré que la caméra NeuroSoc permettait d'extraire des spikes à partir d'un filtrage spatio-temporel effectué sur la sortie d'une caméra *frame-based*, permettant une génération d'événements parcimonieuse. De ces événements et en utilisant un SNN entraîné avec une règle STDP, on peut ensuite extraire des propriétés spatio-temporelles du mouvement qui peuvent alors être utilisées pour prédire des trajectoires de balles. Ce dernier chapitre de discussion remettra en perspective les résultats obtenus et de façon plus générale les SNNs et leur développement.

### 7.1 Discussion générale des résultats

Les chapitres 3.2 et 4 ont permis de montrer que les neurones d'un SNN entraînés avec une règle d'apprentissage non-supervisée de type STDP et utilisant la sortie de la caméra NeuroSoc pouvaient devenir sélectifs à différentes caractéristiques du mouvement comme la direction et la vitesse. Cette sélectivité peut être utilisée par la suite pour prédire la trajectoire d'une balle. Ces travaux ont ensuite été étendus dans le chapitre 5 à un cas d'usage plus ancré dans le domaine du sport, les lancers francs de basket. Durant les deux chapitres 4 et 5, nous avons montré que les performances de notre solution dépassaient les performances de participants humains experts et non-experts sur des tâches similaires. Les études des chapitres 4 et 5 diffèrent aussi par leur méthode pour décoder le SNN. La première étude 4, dans laquelle nous cherchons à déterminer le point de réception de la balle utilise des RPs pour décoder la sortie de la troisième couche du SNN. Ainsi, la prédiction n'est pas réellement faite par le SNN, mais celui-ci permet d'extraire des informations sur le mouvement qui sont ensuite utilisées par les RPs pour effectuer la prédiction. Dans la seconde étude 5, dans laquelle les lancers francs sont prédits, la dernière couche utilise une règle RSTDP permettant ainsi de diviser les neurones en deux classes différentes (lancer franc réussi ou non). Le décodage se fait ainsi de façon plus directe en comptant les spikes émis par les neurones de cette dernière couche. La première méthode utilisant les RPs s'approche alors davantage des méthodes utilisant des modèles physiques pour définir la future trajectoire de balle, sauf qu'ici la direction et la vitesse de la balle ne sont pas directement données. En effet, elles sont déduites par la sélectivité du filtre qui décharge, les régressions polynomiales font alors office de modèle physique. La seconde méthode utilisant la couche dense entraînée avec la règle d'apprentissage RSTDP n'apprend pas de modèle physique, mais une succession de propriétés spatio-temporelles (la sélectivité des filtres). Ainsi, si un joueur fait un lancer différent des autres (très lobé, le joueur ne lance pas du même endroit, etc), il y a de fortes chances que

---

les neurones de la seconde couche déchargent à des endroits jamais vus par la troisième couche et dont les connexions synaptiques sont basses voire nulles. Ces positions spatiales n'étant alors jamais apprises, les neurones de la dernière couche ne déchargeraient pas ou de façon assez aléatoire. La méthode du chapitre 4 utilisant les RPs devrait pouvoir mieux généraliser lors de présentations de trajectoires différentes de celles utilisées durant l'entraînement.

L'un des principaux objectifs de cette thèse était de vérifier la pertinence de l'utilisation de la caméra NeuroSoc pour l'extraction et l'analyse de mouvement, et notamment de trajectoires de balles. Comme expliqué dans la section 3.1.1, cette caméra diffère grandement des autres caméras événementielles actuellement utilisées dans l'état de l'art et présentes sur le marché. Lors de cette thèse et de ses différentes études, nous avons montré que la caméra NeuroSoc était un bon candidat pour concurrencer les caméras asynchrones. En effet, il est tout à fait possible d'extraire des informations sur la scène à partir des événements générés par la caméra. Il serait maintenant intéressant de comparer les événements générés par la caméra NeuroSoc et ceux générés par une caméra de type DVS et de voir l'impact que cela a sur un SNN. Il est pour l'heure difficile de dire quel type de caméra est le mieux pour ce type de tâches. On peut cependant déduire quelques différences entre ces caméras.

### **Le filtrage spatio-temporel**

Comme expliqué, le processus d'extraction des événements diffère beaucoup entre les DVS et la caméra NeuroSoc. La DVS effectue un filtrage temporel, basé sur la variation de luminosité au cours du temps de chaque pixel. La caméra NeuroSoc effectue aussi un filtrage temporel. Celui-ci est bien plus simple car il s'agit d'une différence d'image, mais à ce filtrage temporel s'ajoute un filtrage spatial. Ce filtrage permet ainsi d'inclure une première étape de pré-traitement de l'information en mimant le filtrage fait par le LGN ou V1, permettant d'extraire des informations plus spécifiques (bord, orientation, etc) et de réduire le nombre d'événements émis par la caméra NeuroSoc. Cependant, en contrepartie l'information temporelle est plus pauvre comparée à une DVS. Comme montré dans la section 6, il y a une perte d'information proportionnelle au FPS de la caméra. Ainsi, plus le FPS sera élevé, plus l'information temporelle sera précise. L'utilisation d'un SNN multi-couches devrait de plus, permettre de pallier ce problème. Les dernières couches du SNN intégrant une partie plus globale de l'information, les faibles imprécisions temporelles locales (différence de temps entre deux événements consécutifs, figure 6.10 et 6.10) de la caméra NeuroSoc ne devraient pas être significatives pour les couches supérieures.

### **La parcimonie des événements générés**

Le nombre de spikes émis par la caméra NeuroSoc devrait être bien moindre que celui d'une DVS. L'intensité de la variation de luminosité étant codée par la latence du spike, il n'y a donc qu'un seul spike par pixel et par frame (et donc plus généralement par pixel) lors d'un mouvement. Alors que pour la DVS le nombre d'événements émis par pixel est proportionnel à la variation de luminosité. Cela permet ainsi de réduire la consommation de ressources du SNN, mais ne permet cependant pas de tirer parti des règles d'homéostasie basées sur la fréquence de décharge d'un neurone par exemple. Un comportement similaire à la DVS pourrait cependant être réalisé en utilisant un codage fréquentiel ou par burst plutôt qu'une règle d'intensité à latence à la sortie de la caméra NeuroSoc. De plus, le filtrage spatial permet d'extraire seulement les contours saillants ainsi qu'une réduction du bruit, ce qui participe à cette réduction du nombre d'événements générés.

### **Dépendance à la vitesse du mouvement**

Comme présenté dans le chapitre 6, la vitesse du mouvement ainsi que le contraste entre l'arrière-plan et l'objet en mouvement impacte fortement le processus de génération d'événements. Bien

---

que des pistes d'améliorations aient été proposées, la caméra NeuroSoc reste dépendante de la vitesse du mouvement. Notamment lors d'un mouvement très rapide (translation de plusieurs pixels par frame), la réponse de la caméra NeuroSoc sera peu précise (voir figure 6.2) avec des événements générés simultanément voire dans le mauvais ordre alors qu'une DVS pourra plus facilement appréhender ce type de mouvement sans effet de bord et donc sans perte de précision. Deux récentes études (HOLEŠOVSKÝ, HLAVÁČ et R. ŠKOVIERA 2020 ; HOLEŠOVSKÝ, Radoslav ŠKOVIERA et al. 2021) ont montré que des caméras DVS pouvaient avoir des comportements inattendus (bruit, latence, événements non générés) lors de conditions extrêmes (vitesses et luminosités). Cette dépendance aux mouvements et notamment aux mouvements rapides est un des gros désavantages de la caméra NeuroSoc. En effet, un des points avancés pour mettre en avant les avantages d'une caméra DVS est sa robustesse face aux mouvements rapides. Cependant la caméra NeuroSoc utilise un haut FPS (=240) comme lors de nos cas d'usage et plus généralement ce type de problème devrait être peu présent car les mouvements ne sont pas suffisamment rapides. Si on prend le cas d'une translation de trois pixels par frame, il faudrait alors 40 frames pour que l'objet traverse le champ visuel de la caméra, soit 167ms, ce qui serait un mouvement extrêmement rapide.

## Adaptation

L'un des gros potentiels d'une architecture telle que la caméra NeuroSoc est sa possibilité de diversification. Cette diversité peut se faire selon deux points différents. Le premier, la phase de pré-processing ou la phase du filtrage spatio-temporel. Lors de cette thèse nous n'avons utilisé qu'un filtrage de type DoG pour l'extraction d'événements. Il est cependant possible d'implémenter différentes variétés de filtres (DoG, Gabor, etc) avec des tailles différentes (5x5 lors de cette thèse) et appliqués sur des échelles différentes, y compris simultanément. Ainsi selon le cas d'usage, un seul ou plusieurs filtres différents peuvent être implémentés pour répondre au besoin. Il en va de même pour le filtrage temporel, comme proposé dans la section 6.2. Différentes méthodes avec différents paramètres pourraient être appliquées. Le deuxième point concerne l'utilisation du capteur d'entrée de la caméra NeuroSoc. Dans la version actuelle, celle-ci est composée d'un capteur APTINA en format natif de 752x480 pixels. Mais d'autres capteurs pourraient être utilisés comme pour la caméra Caiman (CAIMAN CAMERA 2021), actuellement développée par Yumain qui utilise un capteur de 2M pixels avec une cadence de 90 FPS. Cette infrastructure pourrait aussi pouvoir se généraliser aux vidéos couleurs ou à d'autres types de caméras (capteurs thermiques ou à grande gamme dynamique par exemple) et permettre ainsi de profiter du traitement événementiel pour un champ d'applications beaucoup plus large.

Ainsi, les principaux avantages d'une caméra événementielle (DVS, DAVIS, etc) asynchrone est sa capacité à générer des données parcimonieuses permettant ainsi de réduire la consommation d'énergie, leur fiabilité à acquérir des mouvements rapides et leur grande gamme dynamique. Concernant la génération parcimonieuse de données par ces capteurs, la caméra NeuroSoc répond tout aussi bien à ce problème voire dépasse les DVS sur ce sujet, car le filtrage spatial et le codage des événements par latence permettent de réduire considérablement le nombre d'événements générés. Concernant les mouvements rapides, le capteur visuel de la NeuroSoc utilisé était adapté aux mouvements analysés durant ces études. En effet, les mouvements humains ne sont pas des mouvements extrêmement rapides ainsi que les trajectoires de balles, qui sont autour de 7.3 m/s à l'initiation d'un lancer franc (HAMILTON et REINSCHMIDT 1997). La caméra NeuroSoc sera probablement moins performante pour des mouvements très rapides et plus dépendants des conditions de luminosités. Il est à noter que les caméras DVS sont elles aussi limitées par des vitesses et conditions de luminosités extrêmes (HOLEŠOVSKÝ, HLAVÁČ et R. ŠKOVIERA 2020 ; HOLEŠOVSKÝ, Radoslav ŠKOVIERA et al. 2021). Ces applications à très grandes vitesses ne représentent que certaines applications précises. La caméra NeuroSoc

---

permet donc de répondre à de nombreux cas d'usages. De plus, sa capacité d'adaptation, en utilisant d'autres capteurs visuels (plus grand FPS, résolution, gamme dynamique), lui permet de s'ouvrir à un nombre d'applications encore plus large.

## 7.2 Perspectives

Ainsi, bien que l'objectif de la NeuroSoc soit assez proche des DVS, c'est à dire, extraire des informations spatio-temporelles en générant des événements, ces deux types de caméras diffèrent grandement. D'un côté la DVS permet de profiter d'un traitement totalement asynchrone et commence à être largement démocratisée dans la communauté des SNNs et du traitement de l'information événementielle de façon plus générale. Elle peine toutefois à étendre son champ d'applications. D'un autre côté la caméra NeuroSoc (ainsi que la Caiman) se présente comme une caméra hybride. Elle permet plus de flexibilité et d'opportunité sur le pré-traitement, réduit le bruit et la génération d'événements. En contrepartie, cette caméra n'est pas asynchrone et est sensible à la vitesse et au contraste des stimuli. Cependant elle pourrait se présenter comme un entre-deux et sa flexibilité pourrait ouvrir à de nouveaux cas d'usages.

Comme expliqué dans la section 2.5, l'un des principaux avantages des SNNs est leur faible consommation d'énergie, notamment lorsqu'ils sont implémentés sur des cartes neuromorphiques. Embarquer le SNN dans une carte de type FPGA n'était pas l'objectif de cette thèse, mais reste un des objectifs futurs du Cerco et de l'entreprise *Yumain*. Lors de cette thèse nous avons profité des capacités que proposent un ordinateur de bureau classique. L'architecture, le nombre de neurones et de synapses n'a donc pas été considéré comme un facteur limitant. Afin d'implémenter ce type de SNN sur une carte neuromorphique, l'architecture devrait sûrement être revue à la baisse afin de réduire le nombre de neurones. Pour cela, différentes solutions existent. Premièrement, le pas (*stride*) qui sépare les différents neurones pourrait être augmenté. Lors de ces travaux le pas est égal à un. Le fait d'augmenter le pas à deux diviserait par quatre (par couche) le nombre de neurones. Une solution similaire et qui pourrait être utilisée en parallèle serait d'utiliser une méthode de pooling afin de réduire la dimension de la carte neuronale entre deux couches. Enfin, la dernière solution serait de diminuer le nombre de filtres convolutionnels utilisés. Ces solutions permettraient de réduire drastiquement le nombre de neurones et donc la quantité de mémoire utilisée par le SNN. Parmi les avantages de la caméra NeuroSoc, il y a la parcimonie des spikes générés. En effet, le filtrage spatial réduit grandement le bruit et le nombre de spikes générés. De plus, le filtrage temporel fait que pour une même position un spike ne sera généré que peu de fois (principalement une fois) lors du passage d'un mouvement. Comme il en est discuté dans la section 4.4 et ci-dessus, la DVS va générer un nombre de spikes proportionnel au contraste entre le fond et le stimulus, alors que ce contraste sera encodé dans la temporalité de l'événement avec la NeuroSoc. De plus, le SNN va augmenter cette parcimonie au fur et à mesure des couches. L'adaptation de seuil vient également augmenter cet effet. La consommation d'énergie d'un SNN étant proportionnelle au nombre de spikes reçus et générés, notre solution qui génère peu de spikes devrait permettre une faible consommation d'énergie. La méthode de partage de poids permet aussi de réduire considérablement la quantité de mémoire utilisée pour les poids synaptiques car ceux-ci sont partagés.

Un des derniers objectifs de cette thèse était d'utiliser deux caméras synchronisées afin de voir si notre SNN pouvait discriminer la profondeur comme cela a été montré dans différentes études (CHAUHAN et al. 2018; HAESSIG, BERTHELON et al. 2019) et voir si cela impactait les performances de prédictions sur des trajectoires latérales (avec peu d'informations de profondeur) et sur des trajectoires plus variées contenant des variations de profondeurs. Pour cela, il fallait mettre en place un système permettant de synchroniser deux caméras NeuroSoc. Dû à

---

un calendrier chargé du côté de Yumain, il leur était difficile de développer ce système rapidement, il fallait donc le faire du côté du CerCo. La durée de ce développement étant incertaine (instabilité de la connexion des deux caméras, implémentation d'un système basique de synchronisation, etc.), nous avons préféré abandonner ce sujet afin de nous concentrer sur les travaux constituant ce manuscrit de thèse (publication du premier article et validation des résultats avec l'expérience de basket). Cependant, ces travaux restent un des objectifs futurs de Yumain et du CerCo. En effet, de premières études ont déjà été effectuées par des chercheurs de l'équipe (CHAUHAN et al. 2018) sur la sélectivité binoculaire de neurones spikants entraînés avec de la STDP et n'attendent plus qu'à être étendus à des stimuli en mouvements. Cependant, un système binoculaire n'aurait que très peu impacté les performances sur les études présentées. En effet, lors de la première étude avec les échanges de passes, l'information de profondeur n'est pas présente et n'a pas d'impact sur le point de réception de la balle. Concernant la seconde étude sur le basket, la profondeur a son rôle car elle permettrait de déterminer si le ballon arrive trop à gauche ou à droite du panier et donc si le ballon touche le cercle et entre dans le panier ou non. Cependant cette information étant très précise (de l'ordre du centimètre), la disparité binoculaire entre un ballon qui rentre ou sort serait extrêmement faible, très difficile à discriminer et donc non-perceptible par le SNN. La binocularité aurait cependant pu affiner la sélectivité des neurones en doublant l'information fournie à ceux-ci et donner un effet moyenneur réduisant l'impact du bruit. Toutefois, elle aurait en contrepartie doublé le nombre de spikes traités.

Afin d'améliorer la sélectivité spatio-temporelle des neurones, nous utilisons des délais synaptiques qui permettent de synchroniser des spikes provenant d'instantanés différents. Lors de cette thèse, nous avons fait le choix d'utiliser une méthode de *delay selection* plutôt que *delay shift* (voire section 2.4.3.1). Cela signifie que les synapses sont dupliquées avec des délais différents puis sélectionnées avec la règle STDP, contrairement au *delays shift* qui n'utilise qu'une seule synapse entre deux neurones, puis apprend les délais. Cette méthode nécessiterait une seconde règle d'apprentissage et ajouterait de la complexité au SNN (paramètres supplémentaires, convergences des poids et des délais, etc). De plus, aucune méthode d'apprentissage biologique de délais n'est pour le moment connue. Les différentes études utilisent des méthodes assez différentes car peu de connaissances existent sur cet apprentissage au niveau du cortex. L'arrivée de la backpropagation permettra très probablement de résoudre ce problème lorsqu'elle permettra d'entraîner les délais de manière automatique et supervisée, mais ne permettra pas d'expliquer ou de reproduire la gestion des délais dans le cortex et de profiter des avantages de l'apprentissage non-supervisé et en ligne. Il existe actuellement différentes études utilisant des délais, selon différentes architectures et méthodes. Cependant, à ma connaissance, aucune étude n'a montré les avantages et inconvénients (performances, consommation de ressources, bio-inspiration, etc) des différentes méthodes utilisant ces délais (*delay shift/delay selection*, connexions FF/FB/latérales/excitatrices/inhibitrices). Une perspective d'évolution serait d'ajouter des délais dans les couches supérieures du SNN. Comme expliqué, ces délais ne sont actuellement implémentés que sur les entrées de la première couche. Il reste à démontrer quel intérêt apporterait l'ajout de délais dans les couches supérieures d'un SNN par rapport à une architecture comme la nôtre utilisant seulement des délais entre l'entrée et la première couche. Cependant, les couches suivantes incluent implicitement ces délais en intégrant les spikes émis par cette première couche et en sélectionnant les connexions avec certains neurones. L'ajout de délais se ferait au prix de mémoire supplémentaire utilisée. L'ajout sur la première couche n'impacte que peu la mémoire car la profondeur de l'entrée est faible (=2 pour ON/OFF). La taille de la matrice de poids étant donc faible pour la première couche, l'augmenter n'affecte que peu la mémoire. Cependant, pour les couches suivantes  $l$ , la profondeur des couches précédentes  $l - 1$  est bien plus élevée (de l'ordre de plusieurs dizaines) et impactera ainsi plus fortement la mémoire.

---

Durant la première étude 4, la prédiction des trajectoire de balles n'est pas faite par le SNN, celle-ci est faite par les régressions polynomiales qui utilisent les propriétés spatio-temporelles des neurones qui déchargent pour prédire le point de réception de la balle. Le SNN ne fait alors qu'extraire de l'information sur le mouvement. Lors de la seconde étude, tout le processus est fait par le SNN. La seule phase de décodage de la sortie consiste à compter les spikes, les neurones déchargent en priorité pour des positions et propriétés spatio-temporelles pertinentes afin de prédire si le lancer-franc est réussi ou non. Cette méthode ne reflète pas le processus de prédiction réalisé par les participants humains experts ou non. Comme précisé dans la section 2.6, il est difficile de déterminer comment les prédictions sont effectuées chez l'humain, d'autant plus que celles-ci utilisent des aires visuo-motrices. D'autres études se sont penchées sur la prédiction visuelle en utilisant des SNNs. (BURGSTEINER et al. 2007 ; KAISER et al. 2017) s'approchent de notre méthode car ils utilisent un décodeur linéaire pour prédire la trajectoire de la balle. Leur méthode diffère cependant par l'utilisation d'un LSMs. D'autres études comme (KAPLAN et al. 2013 ; SHON, RAO et SEJNOWSKI 2004 ; GIBSON, HENDERSON et WILES 2014) utilisent des connexions latérales qui viendront exciter les neurones au moment prédit même si le signal d'entrée est stoppé. Ce type d'architecture est intéressant car toute la prédiction se fait au sein du SNN. Une représentation de la scène est donc réalisée par le SNN qui prédit ensuite le mouvement à travers les connexions latérales. Cependant, ces études restent appliquées sur des stimuli simplifiés avec des mouvements qui ont des directions et vitesses continues. De plus, ce type de connexions récurrentes (principalement courtes) semble difficilement applicable sur nos types de stimuli. En effet, sur une période courte (quelques millisecondes) la vitesse et la direction de la balle varient peu. Il y aurait donc de grandes chances qu'il y ait de fortes connexions entre des filtres qui codent pour des propriétés similaires et que cela amène à de la récursivité. Ainsi, la prédiction continuerait infiniment (dans les limites de la taille de l'image) dans la direction de la balle sans considérer la gravité. Afin de prendre en compte cette dernière, des connexions récurrentes plus longues seraient donc nécessaires. L'étude de (GIBSON, HENDERSON et WILES 2014) utilise des connexions plus longues, il serait donc intéressant de voir à quel point cette méthode pourrait s'appliquer à notre solution, bien que celle-ci semble très dépendante du modèle de neurone et synapse utilisé dans leur étude.

### 7.3 Maturité des SNNs

Les SNNs considérés comme la troisième génération de réseaux de neurones artificiels (MAASS 1997), sont souvent présentés comme des candidats naturels pour effectuer de l'analyse spatio-temporelle. Ceux-ci, tout comme les ANNs, tendent à se développer avec le temps et à être de plus en plus utilisés, avec la vision par ordinateur comme l'une des principales applications.

Le traitement visuel événementiel offre de grandes opportunités dans le domaine de la vision par ordinateur en répondant à des besoins spécifiques (consommation de ressources, temps-réel, mouvements rapides, etc). Et, bien que les SNNs soient définis comme la troisième génération de réseaux de neurones, ils manquent toujours de maturité et peinent à s'affirmer dans le domaine de l'intelligence artificielle. En effet, comme affirmé par (TAYARANI-NAJARAN et SCHMUKER 2021), le nombre de publications autour du thème "systèmes événementiels visuels, olfactifs ou auditifs" est en constante augmentation depuis les années 2006 (figure 7.1, panneaux A). Toutefois, en comparant les résultats du nombre de publications contenant les mots "spiking neural network", "artificial neural network", "convolutional neural network" ou "recurrent neural network" (figure 7.1, panneaux B et C), il apparaît clairement que même si l'intérêt pour les SNNs semble en croissance continue, celle-ci reste bien inférieure aux autres termes. Pour comparaison, les CNNs, qui sont apparus bien plus tard que les SNNs, les dépassent majoritairement et ont une tendance bien plus croissante (presque exponentielle).

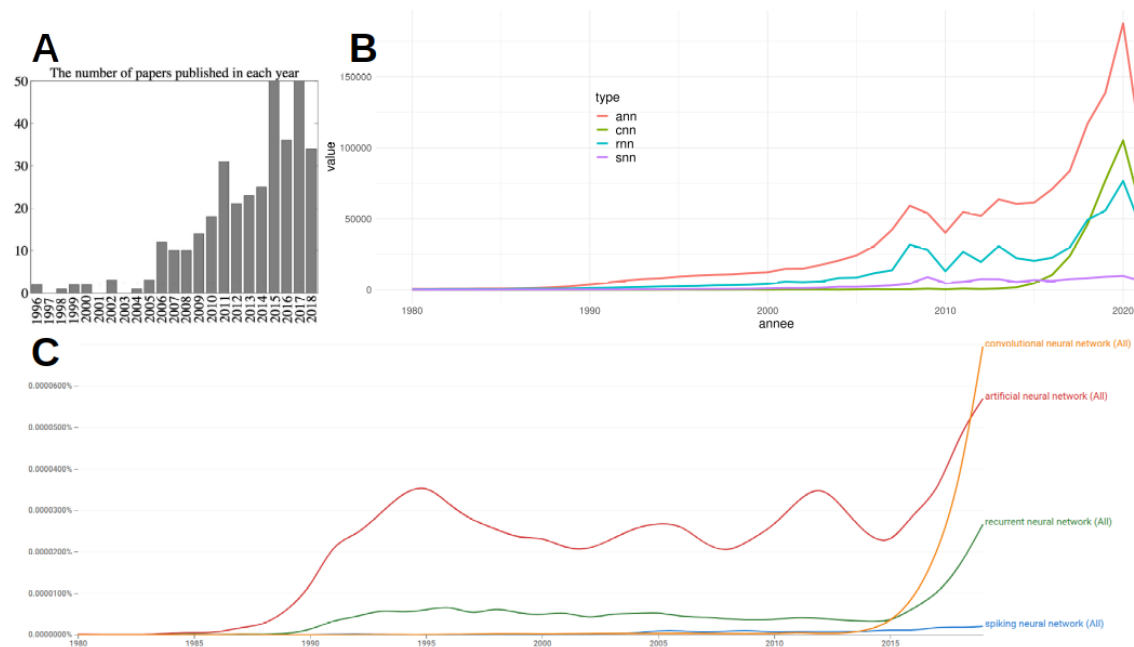


FIGURE 7.1 – Panneau A, nombre de publications autour du thème "systèmes événementiels visuels, olfactifs ou auditifs", figure provenant de (TAYARANI-NAJARAN et SCHMUKER 2021). Panneaux B et C, nombre de publications contenant les mots "spiking neural network", "artificial neural network", "convolutional neural network" ou "recurrent neural network", données récoltées avec "<https://app.dimensions.ai/discover/publication>" et "<https://books.google.com/ngrams>"

Cependant, dans le domaine des *neuromorphic chip* les SNNs se sont imposés et continuent à se développer. L'apprentissage STDP représente une part importante des méthodes d'apprentissages utilisées pour des modèles de traitement de l'information embarqués (voire figure 7.2). L'intérêt de plus en plus fort pour les systèmes consommant peu d'énergie (SCHUMAN et al. 2017) explique notamment l'intérêt des SNNs car ceux-ci permettent en effet d'apporter des réponses à ces contraintes. En effet, comme montré dans (TAVANA EI, GHODRATI et al. 2019; NEIL, PFEIFFER et S.-C. LIU 2016), et par la figure 2.18, les SNNs nécessitent beaucoup moins d'opérations qu'un ANN pour une même performance. Pour plus d'informations sur l'évolution des tendances du calcul neuromorphique, il est possible de se référer à (SCHUMAN et al. 2017) qui propose une grande revue de littérature.



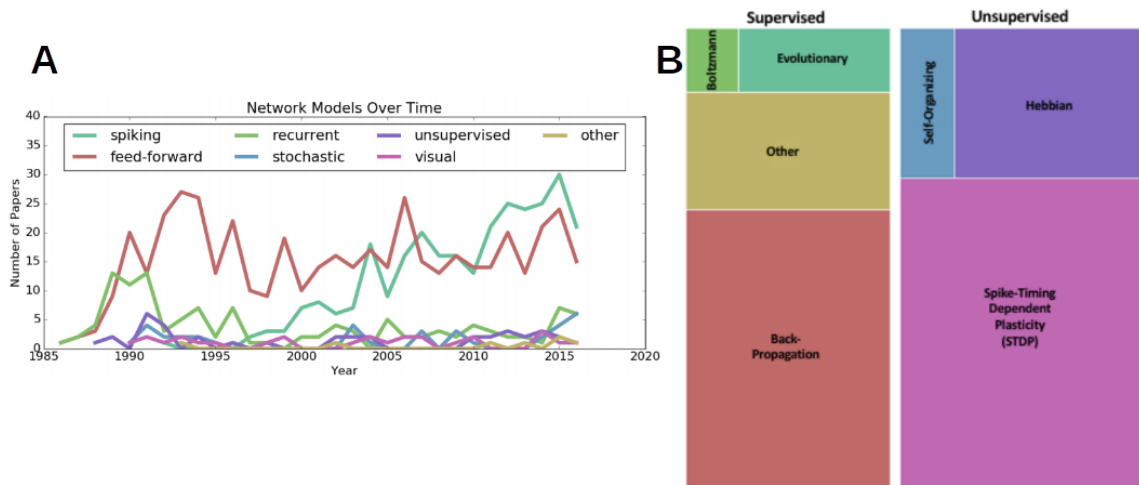


FIGURE 7.2 – Panneau A, évolution de l’intérêt pour les modèles neuromorphiques au cours du temps. Panneaux B, proportion des méthodes d’apprentissage utilisée pour des modèles embarqués. Figures provenant de (SCHUMAN et al. 2017).

Les SNNs traitent de l’information spatio-temporelle. En effet, un spike représente une information temporelle (il arrive à un certain instant  $t$ ) ainsi qu’une information spatiale détenue par le neurone émetteur de ce même spike. Un neurone agit comme un détecteur de coïncidences temporelles (HÜNING, GLÜNDER et PALM 1998), dont la fenêtre temporelle est déterminée par la constante de temps du neurone. La règle STDP sélectionne les connexions spatiales selon des informations temporelles et l’utilisation de délais synaptiques permettent d’améliorer la sélectivité des neurones en synchronisant ou désynchronisant temporellement les spikes reçus par le neurone post-synaptique.

Cependant, de nombreuses études utilisent encore des images (MNIST, CIFAR-10, etc) pour évaluer et tester les performances des SNNs sur des tâches de reconnaissance de formes. Par définition, une image ne possède pas d’information temporelle. Celle-ci est souvent déterminée en utilisant une règle d’intensité à latence (S. THORPE et GAUTRAIS 1998; CHAUHAN et al. 2018; S.-R. KHERADPISHEH et al. 2018) ou en définissant la fréquence des neurones en entrée proportionnelle à l’intensité du pixel (DIEHL, NEIL et al. 2015; THIELE, BICHLER et DUPRET 2018). Des versions davantage bio-inspirées ont été proposées en utilisant des caméras DVS et en mimant des saccades oculaires. Cependant, comme pointé par (IYER, CHUA et Haizhou LI 2021) cette méthode ne permet pas d’apporter de valeur ajoutée à la composante temporelle du stimulus. Des ANNs n’utilisant seulement des informations spatiales peuvent ainsi facilement dépasser les performances sur ce type de tâche.

Beaucoup de recherches sur les SNNs sont donc dans une stratégie de mimétisme des ANNs, en utilisant les mêmes jeux de données (MNIST, CIFAR), en implémentant des règles d’apprentissage similaires (back-propagation) ou en convertissant des ANNs en SNNs (DIEHL, ZARRELLA et al. 2016; T. SERRANO-GOTARREDONA et al. 2015; MASSA et al. 2021). Ces méthodes ont permis et permettent encore une évolution rapide des SNNs. Toutefois, cette stratégie pourrait aussi brider le développement des SNNs sur le long terme. En effet, en cherchant à utiliser des techniques et en utilisant des jeux de données adaptés et optimisés pour les ANNs (MNIST, CIFAR-10, etc), les SNNs resteront moins performants par rapport aux ANNs. Comme montré par (DENG et al. 2020; W. HE et al. 2020), les SNNs se démarquent des ANNs lorsque l’information temporelle apporte une information supplémentaire, comme par exemple dans la base de données *DVS-Gesture* dans laquelle l’information temporelle apporte une information sur le mouvement. En effet, lorsqu’ils sont appliqués sur ce type de stimuli, les SNNs dépassent les

---

CNNs et obtiennent des résultats proches voire meilleurs que des RNNs ou des Long Short Term Memory (LSTM) (DENG et al. 2020 ; W. HE et al. 2020 ; FANG, YU et al. 2021). L'étude de (KOOPMAN, LEEUWEN et VREEKEN 2003) a aussi comparé des SNNs avec des LSTM sur différentes tâches temporelles (mais non visuelles) et montré que les SNNs pouvaient dépasser les performances des LSTM sur certaines d'entre elles.

De nouvelles études proposent aussi des bases de données de stimuli provenant de DVS afin de pouvoir comparer les résultats des différentes architectures de SNNs (STEFFEN, ELFGEN et al. 2021 ; CALABRESE et al. 2019 ; MIAO et al. 2019 ; ZHU et al. 2018 ; HU et al. 2016 ; BINAS et al. 2017 ; Teresa SERRANO-GOTARREDONA et B. LINARES-BARRANCO 2015 ; Q. LIU et al. 2016), voir (GALLEGO et al. 2020 ; [uzh-rpg/event-based\\_vision\\_resources](https://github.com/uzh-rpg/event-based_vision_resources) 2021) pour un récapitulatif. Cela permet ainsi aux SNNs d'avoir des bases de données adaptées et tirant profit de leurs avantages (information spatio-temporelle, données parcimonieuses, asynchrones, etc). De plus, de nouvelles solutions alternatives existent permettant d'utiliser des bases de données de vidéos utilisées par des ANNs pour les convertir en flux de spikes (GEHRIG et al. 2020 ; DELBRÜCK, HU et Z. HE 2020) ou utilisent des simulateurs d'environnement comme unrealCV (QIU et al. 2017) pour ensuite générer un flux d'événements comme l'aurait fait une caméra DVS (REBECQ, GEHRIG et SCARAMUZZA 2018). Le processus de la caméra NeuroSoc peut lui aussi être facilement appliqué comme il est fait dans la section 3.2 pour extraire des événements de ce type de stimuli. Ce type de base de données permettrait d'évaluer les SNNs sur des données contrôlées, facilement comparables et ne plus être entièrement dépendant des bases de données utilisées par les ANNs. De plus, il reste pertinent de tester les SNNs sur des situations réelles ou proches du réel contenant du mouvement, afin d'éviter des pertes de performances lorsque le SNN sera ensuite appliqué sur des scènes naturelles (MORENO-TORRES et al. 2012 ; ALAIZ-RODRÍGUEZ et JAPKOWICZ 2008). (GEHRIG et al. 2020 ; DELBRÜCK, HU et Z. HE 2020) va dans ce sens en proposant une conversion de vidéo en événements. Pour cela, ils utilisent une méthode qui permet de mimer avec précision le fonctionnement d'une caméra DVS ainsi que les potentiels bruits (*Hot pixels*, *Leak noise events*, etc) présents lors de l'utilisation d'une DVS.

La principale méthode utilisée pour entraîner un SNN consiste (comme pour les ANNs) à modifier les poids des connexions synaptiques afin de moduler l'impact d'un spike sur le neurone post-synaptique en fonction de son expérience passée. Ensuite, afin d'augmenter la sélectivité des SNNs et plus particulièrement sur la dimension temporelle du stimulus, l'implémentation de délais synaptiques est de plus en plus utilisée. Comme expliqué dans la section 2.4.3.1, différentes architectures et méthodes existent pour implémenter des délais. Cependant, aucune étude comparative permet de sélectionner la meilleure architecture ainsi que la meilleure méthode d'apprentissage (*delay-shift/delay-selection*) permettant de mieux répondre au problème (performances, consommation de ressources, bio-inspiré, etc). De plus, les perspectives d'évolutions des SNNs ne se limitent pas à l'apprentissage des délais et des poids synaptiques. Comme mis en avant par (SAXE et al. 2011), l'impact des paramètres (pour un SNN : constante de temps, fenêtre de temps d'apprentissage, seuil, etc) a un très fort impact sur les performances d'un ANN même lorsque ses poids ne sont pas entraînés. Ainsi, des paramètres hétérogènes et évolutifs, à travers de l'adaptation à court terme ou de l'apprentissage sur du long terme, devraient permettre d'améliorer les performances des SNNs. Il devrait par exemple être nécessaire d'avoir une constante de temps élevée au début de l'apprentissage car les poids des connexions et les délais sont aléatoires. En effet, au début de l'apprentissage, il y a peu de synchronie dans les spikes reçus par le neurone post-synaptique. Il semble donc nécessaire d'avoir une constante de temps élevée afin de prendre en compte le plus de spikes possible, même ceux qui ne sont pas encore synchronisés. Au fur et à mesure de l'apprentissage, le neurone post-synaptique décharge pour des spikes synchronisés dans le temps, car il a sélectionné des synapses qui sont synchroni-

---

sées et/ou les a synchronisées avec de l'apprentissage de délais. Il ne semble donc plus nécessaire voire contre-intuitif de garder une constante de temps élevée. Il en va de même pour la fenêtre de temps du LTP. Ajouter à cela une hétérogénéité dans les paramètres afin que les neurones soient adaptés à des profils de patterns spatio-temporels différents (de nombreux spikes générés rapidement lors d'un mouvement brusque et rapide par exemple ou des spikes plus espacés dans le temps lors d'un mouvement lent) pourrait permettre aux SNNs de s'adapter aux différentes statistiques d'une scène. Ces types de processus existent dans le cerveau (WLODARCZYK et al. 2013; GARDNER 1986; PLATKIEWICZ et BRETTE 2010) et des premières initiatives commencent déjà apparaître dans les SNNs mais sont souvent restreints à de l'adaptation du seuil, afin de garantir une homéostasie (DIEHL et COOK 2015; FALEZ et al. 2019) (ou à des simulations bien souvent éloignées des patterns que l'on retrouve en situation réelle). De récentes études vont dans ce sens, comme par exemple (SHE et al. 2021), qui propose un SNN avec des constantes de temps différentes afin que différents patterns spatio-temporels soient appris, permettant ainsi de concurrencer certains ANNs. De la même manière l'étude (FANG, YU et al. 2021) utilise la backpropagation pour entraîner les constantes de temps et dépasse ainsi les performances de l'état de l'art sur de nombreux jeux de données. L'étude (MORAITIS, SEBASTIAN et ELEFThERIOU 2020) est aussi un bon exemple montrant le potentiel des SNNs. En ajoutant une règle de plasticité synaptique sur le court-terme en plus de la règle STDP, le SNN dépasse les performances de l'état de l'art des ANNs sur une tâche de reconnaissance d'image avec occlusion temporelle. L'architecture utilisée joue un rôle important, car l'utilisation de connexions récurrentes ou de connexions résiduelles améliorent par exemple les performances des SNNs (XING, DI CATERINA et SORAGHAN 2020; FANG, YU et al. 2021). On voit ainsi qu'en ajoutant de la dynamique dans les SNNs à travers de l'hétérogénéité dans les paramètres, en complexifiant l'architecture et/ou de l'adaptation à court terme par exemple, les SNNs commencent à concurrencer les ANNs. Les performances sont améliorées tout en conservant des processus non-supervisés (permettant ainsi d'être actifs "online") et biologiquement plausibles.

# Chapitre 8

## Références

- ABDERRAHMANE, Nassim et Benoît MIRAMOND (août 2019). “Information Coding and Hardware Architecture of Spiking Neural Networks”. In : *2019 22nd Euro-micro Conference on Digital System Design (DSD)*, p. 291-298. DOI : 10.1109/DSD.2019.00050.
- ADAMS, Samantha V. et Chris M. HARRIS (juil. 2014). “A Proto-Architecture for Innate Directionally Selective Visual Maps”. en. In : *PLOS ONE* 9.7, e102908. ISSN : 1932-6203. DOI : 10.1371/journal.pone.0102908. URL : <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0102908> (visité le 29/04/2019).
- AGLIOTI, Salvatore M. et al. (sept. 2008). “Action anticipation and motor resonance in elite basketball players”. en. In : *Nature Neuroscience* 11.9, p. 1109-1116. ISSN : 1546-1726. DOI : 10.1038/nn.2182. URL : <https://www.nature.com/articles/nn.2182> (visité le 27/04/2020).
- AKIDA (2021). *Akida Neural Processor IP*. en-US. URL : <https://brainchipinc.com/akida-neural-processor-ip/> (visité le 23/04/2021).
- AKOPYAN, Filipp et al. (oct. 2015). “TrueNorth : Design and Tool Flow of a 65 mW 1 Million Neuron Programmable Neurosynaptic Chip”. In : *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 34.10. Conference Name : IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, p. 1537-1557. ISSN : 1937-4151. DOI : 10.1109/TCAD.2015.2474396.
- ALAIZ-RODRÍGUEZ, Rocío et Nathalie JAPKOWICZ (2008). “Assessing the Impact of Changing Environments on Classifier Performance”. en. In : *Advances in Artificial Intelligence*. Sous la dir. de Sabine BERGLER. Lecture Notes in Computer Science. Berlin, Heidelberg : Springer, p. 13-24. ISBN : 978-3-540-68825-9. DOI : 10.1007/978-3-540-68825-9\_2.
- AUGE, Daniel et al. (juil. 2021). “A Survey of Encoding Techniques for Signal Processing in Spiking Neural Networks”. en. In : *Neural Processing Letters*. ISSN : 1573-773X. DOI : 10.1007/s11063-021-10562-2. URL : <https://doi.org/10.1007/s11063-021-10562-2> (visité le 26/07/2021).
- BABAEIZADEH, Mohammad et al. (mar. 2018). “Stochastic Variational Video Prediction”. In : *arXiv :1710.11252 [cs]*. arXiv : 1710.11252. URL : <http://arxiv.org/abs/1710.11252> (visité le 01/06/2021).
- BALSER, Nils et al. (2014). “The influence of expertise on brain activation of the action observation network during anticipation of tennis and volleyball serves”.

- 
- eng. In : *Frontiers in Human Neuroscience* 8, p. 568. ISSN : 1662-5161. DOI : 10.3389/fnhum.2014.00568.
- BASHIRI, M. et A. FARSHBAF GERANMAYEH (déc. 2011). “Tuning the parameters of an artificial neural network using central composite design and genetic algorithm”. en. In : *Scientia Iranica* 18.6, p. 1600-1608. ISSN : 1026-3098. DOI : 10.1016/j.scient.2011.08.031. URL : <https://www.sciencedirect.com/science/article/pii/S1026309811002136> (visité le 15/06/2021).
- BAUMGARTNER, Sandro et al. (oct. 2020). “Visual Pattern Recognition with on On-Chip Learning : Towards a Fully Neuromorphic Approach”. In : *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*. ISSN : 2158-1525, p. 1-5. DOI : 10.1109/ISCAS45731.2020.9180628.
- BENJAMIN, Ben Varkey et al. (mai 2014). “Neurogrid : A Mixed-Analog-Digital Multichip System for Large-Scale Neural Simulations”. In : *Proceedings of the IEEE* 102.5. Conference Name : Proceedings of the IEEE, p. 699-716. ISSN : 1558-2256. DOI : 10.1109/JPROC.2014.2313565.
- BERNER, Raphael et al. (juin 2013). “A 240×180 10mW 12us latency sparse-output vision sensor for mobile applications”. In : *2013 Symposium on VLSI Circuits*. ISSN : 2158-5636, p. C186-C187.
- BERRY, Michael J., David K. WARLAND et Markus MEISTER (mai 1997). “The structure and precision of retinal spike trains”. en. In : *Proceedings of the National Academy of Sciences* 94.10. Publisher : National Academy of Sciences Section : Biological Sciences, p. 5411-5416. ISSN : 0027-8424, 1091-6490. DOI : 10.1073/pnas.94.10.5411. URL : <https://www.pnas.org/content/94/10/5411> (visité le 09/06/2021).
- BI, Guo-qiang et Mu-ming POO (2001). “Synaptic Modification by Correlated Activity : Hebb’s Postulate Revisited”. In : *Annual Review of Neuroscience* 24.1, p. 139-166. DOI : 10.1146/annurev.neuro.24.1.139. URL : <https://doi.org/10.1146/annurev.neuro.24.1.139> (visité le 30/04/2020).
- BICHLER, Olivier et al. (août 2012). “Extraction of temporally correlated features from dynamic vision sensors with spike-timing-dependent plasticity”. In : *Neural Networks. Selected Papers from IJCNN 2011* 32, p. 339-348. ISSN : 0893-6080. DOI : 10.1016/j.neunet.2012.02.022. URL : <http://www.sciencedirect.com/science/article/pii/S0893608012000512> (visité le 15/05/2019).
- BINAS, Jonathan et al. (nov. 2017). “DDD17 : End-To-End DAVIS Driving Dataset”. In : *arXiv :1711.01458 [cs]*. arXiv : 1711.01458. URL : <http://arxiv.org/abs/1711.01458> (visité le 05/07/2021).
- BISHOP, Christopher M. (1995). *Neural Networks for Pattern Recognition*. USA : Oxford University Press, Inc. ISBN : 978-0-19-853864-6.
- BOHTE, Sander M., Joost N. KOK et Han LA POUTRÉ (oct. 2002). “Error-backpropagation in temporally encoded networks of spiking neurons”. en. In : *Neurocomputing* 48.1, p. 17-37. ISSN : 0925-2312. DOI : 10.1016/S0925-2312(01)00658-0. URL : <https://www.sciencedirect.com/science/article/pii/S0925231201006580> (visité le 28/06/2021).
- BRANDLI, Christian et al. (oct. 2014). “A 240 × 180 130 dB 3 μs Latency Global Shutter Spatiotemporal Vision Sensor”. In : *IEEE Journal of Solid-State Circuits* 49.10. Conference Name : IEEE Journal of Solid-State Circuits, p. 2333-2341. ISSN : 1558-173X. DOI : 10.1109/JSSC.2014.2342715.

- 
- BRETTE, Romain (2015). “Philosophy of the Spike : Rate-Based vs. Spike-Based Theories of the Brain”. English. In : *Frontiers in Systems Neuroscience* 9. Publisher : Frontiers. ISSN : 1662-5137. DOI : 10.3389/fnsys.2015.00151. URL : <https://www.frontiersin.org/articles/10.3389/fnsys.2015.00151/full> (visité le 27/06/2021).
- BURBANK, Kendra S. (déc. 2015). “Mirrored STDP Implements Autoencoder Learning in a Network of Spiking Neurons”. en. In : *PLOS Computational Biology* 11.12. Publisher : Public Library of Science, e1004566. ISSN : 1553-7358. DOI : 10.1371/journal.pcbi.1004566. URL : <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004566> (visité le 28/06/2021).
- BURGSTEINER, Harald et al. (avr. 2007). “Movement prediction from real-world images using a liquid state machine”. en. In : *Applied Intelligence* 26.2, p. 99-109. ISSN : 1573-7497. DOI : 10.1007/s10489-006-0007-1. URL : <https://doi.org/10.1007/s10489-006-0007-1> (visité le 15/05/2019).
- BURKITT, Anthony N. et Hinze HOGENDOORN (mai 2021). “Predictive Visual Motion Extrapolation Emerges Spontaneously and without Supervision at Each Layer of a Hierarchical Neural Network with Spike-Timing-Dependent Plasticity”. en. In : *Journal of Neuroscience* 41.20. Publisher : Society for Neuroscience Section : Research Articles, p. 4428-4438. ISSN : 0270-6474, 1529-2401. DOI : 10.1523/JNEUROSCI.2017-20.2021. URL : <https://www.jneurosci.org/content/41/20/4428> (visité le 10/09/2021).
- BUTZ, Martin V., Olivier SIGAUD et Pierre GÉRARD (2003). “Internal Models and Anticipations in Adaptive Learning Systems”. en. In : *Anticipatory Behavior in Adaptive Learning Systems : Foundations, Theories, and Systems*. Sous la dir. de Martin V. BUTZ, Olivier SIGAUD et Pierre GÉRARD. Lecture Notes in Computer Science. Berlin, Heidelberg : Springer, p. 86-109. ISBN : 978-3-540-45002-3. DOI : 10.1007/978-3-540-45002-3\_6. URL : [https://doi.org/10.1007/978-3-540-45002-3\\_6](https://doi.org/10.1007/978-3-540-45002-3_6) (visité le 20/09/2021).
- CAIMAN CAMERA (2021). en-GB. URL : <https://yumain.fr/en/products/caiman-camera/> (visité le 22/04/2021).
- CALABRESE, Enrico et al. (2019). “DHP19 : Dynamic Vision Sensor 3D Human Pose Dataset”. In : p. 0-0. URL : [https://openaccess.thecvf.com/content\\_CVPRW\\_2019/html/EventVision/Calabrese\\_DHP19\\_Dynamic\\_Vision\\_Sensor\\_3D\\_Human\\_Pose\\_Dataset\\_CVPRW\\_2019\\_paper.html](https://openaccess.thecvf.com/content_CVPRW_2019/html/EventVision/Calabrese_DHP19_Dynamic_Vision_Sensor_3D_Human_Pose_Dataset_CVPRW_2019_paper.html) (visité le 05/07/2021).
- CAPORALE, Natalia et Yang DAN (juil. 2008). “Spike Timing-Dependent Plasticity : A Hebbian Learning Rule”. In : *Annual Review of Neuroscience* 31.1. Publisher : Annual Reviews, p. 25-46. ISSN : 0147-006X. DOI : 10.1146/annurev.neuro.31.060407.125639. URL : <https://www.annualreviews.org/doi/10.1146/annurev.neuro.31.060407.125639> (visité le 07/07/2021).
- CARANDINI, Matteo et al. (nov. 2005). “Do We Know What the Early Visual System Does?” en. In : *Journal of Neuroscience* 25.46. Publisher : Society for Neuroscience Section : Symposia and Mini-Symposia, p. 10577-10597. ISSN : 0270-6474, 1529-2401. DOI : 10.1523/JNEUROSCI.3726-05.2005. URL : <https://www.jneurosci.org/content/25/46/10577> (visité le 11/05/2021).
- CAUSER, Joe, Nicholas J. SMEETON et A. Mark WILLIAMS (2017). “Expertise differences in anticipatory judgements during a temporally and spatially occluded

- 
- task”. eng. In : *PLoS One* 12.2, e0171330. ISSN : 1932-6203. DOI : 10.1371/journal.pone.0171330.
- CESSAC, Bruno et al. (2017). “PRANAS : A New Platform for Retinal Analysis and Simulation”. English. In : *Frontiers in Neuroinformatics* 11. Publisher : Frontiers. ISSN : 1662-5196. DOI : 10.3389/fninf.2017.00049. URL : <https://www.frontiersin.org/articles/10.3389/fninf.2017.00049/full> (visité le 22/06/2021).
- CHAUHAN, Tushar et al. (oct. 2018). “Emergence of Binocular Disparity Selectivity through Hebbian Learning”. en. In : *Journal of Neuroscience* 38.44, p. 9563-9578. ISSN : 0270-6474, 1529-2401. DOI : 10.1523/JNEUROSCI.1259-18.2018. URL : <http://www.jneurosci.org/content/38/44/9563> (visité le 15/05/2019).
- CHEN, Jim X. (juil. 2016). “The Evolution of Computing : AlphaGo”. In : *Computing in Science Engineering* 18.4. Conference Name : Computing in Science Engineering, p. 4-7. ISSN : 1558-366X. DOI : 10.1109/MCSE.2016.74.
- CHEN, Xiaopeng et al. (avr. 2015). “A Robust Vision Module for Humanoid Robotic Ping-Pong Game”. en. In : *International Journal of Advanced Robotic Systems* 12.4. Publisher : SAGE Publications, p. 35. ISSN : 1729-8814. DOI : 10.5772/60406. URL : <https://doi.org/10.5772/60406> (visité le 01/06/2021).
- DAVIES, Mike, Narayan SRINIVASA et al. (jan. 2018). “Loihi : A Neuromorphic Many-core Processor with On-Chip Learning”. In : *IEEE Micro* 38.1. Conference Name : IEEE Micro, p. 82-99. ISSN : 1937-4143. DOI : 10.1109/MM.2018.112130359.
- DAVIES, Mike, Andreas WILD et al. (mai 2021). “Advancing Neuromorphic Computing With Loihi : A Survey of Results and Outlook”. In : *Proceedings of the IEEE* 109.5. Conference Name : Proceedings of the IEEE, p. 911-934. ISSN : 1558-2256. DOI : 10.1109/JPROC.2021.3067593.
- DEBAT, Guillaume et al. (2021). “Event-Based Trajectory Prediction Using Spiking Neural Networks”. English. In : *Frontiers in Computational Neuroscience* 0. Publisher : Frontiers. ISSN : 1662-5188. DOI : 10.3389/fncom.2021.658764. URL : [https://www.frontiersin.org/articles/10.3389/fncom.2021.658764/full?utm\\_source=Email\\_to\\_authors&utm\\_medium=Email&utm\\_content=T1\\_11.5e1\\_author&utm\\_campaign=Email\\_publication&field=&journalName=Frontiers\\_in\\_Computational\\_Neuroscience&id=658764](https://www.frontiersin.org/articles/10.3389/fncom.2021.658764/full?utm_source=Email_to_authors&utm_medium=Email&utm_content=T1_11.5e1_author&utm_campaign=Email_publication&field=&journalName=Frontiers_in_Computational_Neuroscience&id=658764) (visité le 06/08/2021).
- DELBRÜCK, T., Yuhuang HU et Zhe HE (2020). “V2E : From video frames to realistic DVS event camera streams”. In : *ArXiv*.
- DELORME, Arnaud, Laurent PERRINET et Simon J. THORPE (juin 2001). “Networks of integrate-and-fire neurons using Rank Order Coding B : Spike timing dependent plasticity and emergence of orientation selectivity”. en. In : *Neurocomputing*. Computational Neuroscience : Trends in Research 2001 38-40, p. 539-545. ISSN : 0925-2312. DOI : 10.1016/S0925-2312(01)00403-9. URL : <http://www.sciencedirect.com/science/article/pii/S0925231201004039> (visité le 13/11/2020).
- DENÈVE, Sophie et Christian K. MACHENS (mar. 2016). “Efficient codes and balanced networks”. en. In : *Nature Neuroscience* 19.3. Number : 3 Publisher : Nature Publishing Group, p. 375-382. ISSN : 1546-1726. DOI : 10.1038/nn.4243. URL : <https://www.nature.com/articles/nn.4243> (visité le 15/03/2021).

- 
- DENG, Lei et al. (jan. 2020). “Rethinking the performance comparison between SNNs and ANNs”. en. In : *Neural Networks* 121, p. 294-307. ISSN : 0893-6080. DOI : 10.1016/j.neunet.2019.09.005. URL : <https://www.sciencedirect.com/science/article/pii/S0893608019302667> (visité le 04/08/2021).
- DI PAOLO, Ezequiel Alejandro et al. (2014). “Learning to perceive in the sensorimotor approach : Piaget’s theory of equilibration interpreted dynamically”. In : *Frontiers in Human Neuroscience* 8, p. 551. ISSN : 1662-5161. DOI : 10.3389/fnhum.2014.00551. URL : <https://www.frontiersin.org/article/10.3389/fnhum.2014.00551> (visité le 20/09/2021).
- DIAZ, Gabriel et al. (jan. 2013). “Saccades to future ball location reveal memory-based prediction in a virtual-reality interception task”. en. In : *Journal of Vision* 13.1. Publisher : The Association for Research in Vision and Ophthalmology, p. 20-20. ISSN : 1534-7362. DOI : 10.1167/13.1.20. URL : <https://jov.arvojournals.org/article.aspx?articleid=2121293> (visité le 18/08/2021).
- DIEHL, Peter U. et Matthew COOK (2015). “Unsupervised learning of digit recognition using spike-timing-dependent plasticity”. English. In : *Frontiers in Computational Neuroscience* 9. Publisher : Frontiers. ISSN : 1662-5188. DOI : 10.3389/fncom.2015.00099. URL : <https://www.frontiersin.org/articles/10.3389/fncom.2015.00099/full> (visité le 19/09/2020).
- DIEHL, Peter U., Daniel NEIL et al. (juil. 2015). “Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing”. In : *2015 International Joint Conference on Neural Networks (IJCNN)*. ISSN : 2161-4407, p. 1-8. DOI : 10.1109/IJCNN.2015.7280696.
- DIEHL, Peter U., Guido ZARRELLA et al. (jan. 2016). “Conversion of Artificial Recurrent Neural Networks to Spiking Neural Networks for Low-power Neuromorphic Hardware”. In : *arXiv :1601.04187 [cs]*. arXiv : 1601.04187. URL : <http://arxiv.org/abs/1601.04187> (visité le 09/07/2021).
- DIKOV, Georgi et al. (2017). “Spiking Cooperative Stereo-Matching at 2 ms Latency with Neuromorphic Hardware”. en. In : *Biomimetic and Biohybrid Systems*. Sous la dir. de Michael MANGAN et al. Lecture Notes in Computer Science. Cham : Springer International Publishing, p. 119-137. ISBN : 978-3-319-63537-8. DOI : 10.1007/978-3-319-63537-8\_11.
- ENGEL, S. A., G. H. GLOVER et B. A. WANDELL (mar. 1997). “Retinotopic organization in human visual cortex and the spatial precision of functional MRI”. eng. In : *Cerebral Cortex (New York, N.Y. : 1991)* 7.2, p. 181-192. ISSN : 1047-3211. DOI : 10.1093/cercor/7.2.181.
- ESSER, Steve K. et al. (2015). “Backpropagation for Energy-Efficient Neuromorphic Computing”. en. In : *Advances in Neural Information Processing Systems* 28. URL : <https://papers.nips.cc/paper/2015/hash/10a5ab2db37feedfdeaab192ead4ac0e-Abstract.html> (visité le 05/07/2021).
- EURICH, Christian et al. (fév. 1999). “Dynamics of Self-Organized Delay Adaptation”. In : *Physical Review Letters* 82. DOI : 10.1103/PhysRevLett.82.1594.
- EURICH, Christian W. et al. (juin 2000). “Delay adaptation in the nervous system”. en. In : *Neurocomputing* 32-33, p. 741-748. ISSN : 0925-2312. DOI : 10.1016/S0925-2312(00)00239-3. URL : <http://www.sciencedirect.com/science/article/pii/S0925231200002393> (visité le 18/12/2019).



- 
- FABRE-THORPE, Michèle, Ghislaine RICHARD et Simon J. THORPE (jan. 1998). “Rapid categorization of natural images by rhesus monkeys”. en-US. In : *Neuro-Report* 9.2, p. 303-308. ISSN : 0959-4965. URL : [https://journals.lww.com/neuroreport/Abstract/1998/01260/Rapid\\_categorization\\_of\\_natural\\_images\\_by\\_rhesus.23.aspx](https://journals.lww.com/neuroreport/Abstract/1998/01260/Rapid_categorization_of_natural_images_by_rhesus.23.aspx) (visité le 08/06/2021).
- FALEZ, Pierre et al. (juil. 2019). “Multi-layered Spiking Neural Network with Target Timestamp Threshold Adaptation and STDP”. In : *2019 International Joint Conference on Neural Networks (IJCNN)*. ISSN : 2161-4407, p. 1-8. DOI : 10.1109/IJCNN.2019.8852346.
- FANG, Wei, Zhaofei YU et al. (août 2021). “Incorporating Learnable Membrane Time Constant to Enhance Learning of Spiking Neural Networks”. In : *arXiv :2007.05785 [cs]*. arXiv : 2007.05785. URL : <http://arxiv.org/abs/2007.05785> (visité le 19/08/2021).
- FANG, Wei, Zhaofei YU et al. (juin 2021). *Deep Residual Learning in Spiking Neural Networks*.
- FARABET, Clément et al. (avr. 2012). “Comparison between Frame-Constrained Fix-Pixel-Value and Frame-Free Spiking-Dynamic-Pixel ConvNets for Visual Processing”. In : *Frontiers in Neuroscience* 6. ISSN : 1662-4548. DOI : 10.3389/fnins.2012.00032. URL : <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3324817/> (visité le 22/04/2021).
- FAUL, Franz et al. (mai 2007). “G\*Power 3 : A flexible statistical power analysis program for the social, behavioral, and biomedical sciences”. en. In : *Behavior Research Methods* 39.2, p. 175-191. ISSN : 1554-3528. DOI : 10.3758/BF03193146. URL : <https://doi.org/10.3758/BF03193146> (visité le 15/03/2021).
- FELLEMAN, D. J. et D. C. VAN ESSEN (fév. 1991). “Distributed hierarchical processing in the primate cerebral cortex”. eng. In : *Cerebral Cortex (New York, N.Y. : 1991)* 1.1, p. 1-47. ISSN : 1047-3211. DOI : 10.1093/cercor/1.1.1-a.
- FERNANDO, Chrisantha et Sampsa SOJAKKA (2003). “Pattern Recognition in a Bucket”. en. In : *Advances in Artificial Life*. Sous la dir. de Wolfgang BANTHAF et al. Lecture Notes in Computer Science. Berlin, Heidelberg : Springer, p. 588-597. ISBN : 978-3-540-39432-7. DOI : 10.1007/978-3-540-39432-7\_63.
- FLORIAN, Răzvan V. (août 2012). “The Chronotron : A Neuron That Learns to Fire Temporally Precise Spike Patterns”. en. In : *PLOS ONE* 7.8. Publisher : Public Library of Science, e40233. ISSN : 1932-6203. DOI : 10.1371/journal.pone.0040233. URL : <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0040233> (visité le 28/06/2021).
- FRAGKIADAKI, Katerina et al. (jan. 2016). “Learning Visual Predictive Models of Physics for Playing Billiards”. In : *arXiv :1511.07404 [cs]*. arXiv : 1511.07404. URL : <http://arxiv.org/abs/1511.07404> (visité le 31/05/2021).
- FUORTES, M. G. F. et Francoise MANTEGAZZINI (juil. 1962). “Interpretation of the Repetitive Firing of Nerve Cells”. In : *Journal of General Physiology* 45.6, p. 1163-1179. ISSN : 0022-1295. DOI : 10.1085/jgp.45.6.1163. URL : <https://doi.org/10.1085/jgp.45.6.1163> (visité le 07/05/2021).
- FURBER, Steve B. et al. (mai 2014). “The SpiNNaker Project”. In : *Proceedings of the IEEE* 102.5. Conference Name : Proceedings of the IEEE, p. 652-665. ISSN : 1558-2256. DOI : 10.1109/JPROC.2014.2304638.

- 
- GALLEGO, Guillermo et al. (2020). “Event-based Vision : A Survey”. In : *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Conference Name : IEEE Transactions on Pattern Analysis and Machine Intelligence, p. 1-1. ISSN : 1939-3539. DOI : 10.1109/TPAMI.2020.3008413.
- GARDNER, D. (nov. 1986). “Variations in amplitude and time course of inhibitory postsynaptic currents”. eng. In : *Journal of Neurophysiology* 56.5, p. 1424-1438. ISSN : 0022-3077. DOI : 10.1152/jn.1986.56.5.1424.
- GAUTRAIS, Jacques et Simon THORPE (nov. 1998). “Rate coding versus temporal order coding : a theoretical approach”. en. In : *Biosystems* 48.1, p. 57-65. ISSN : 0303-2647. DOI : 10.1016/S0303-2647(98)00050-1. URL : <https://www.sciencedirect.com/science/article/pii/S0303264798000501> (visité le 19/05/2021).
- GEHRIG, Daniel et al. (juin 2020). “Video to Events : Recycling Video Datasets for Event Cameras”. In : *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. ISSN : 2575-7075, p. 3583-3592. DOI : 10.1109/CVPR42600.2020.00364.
- GIBSON, Tingting Amy, James A. HENDERSON et Janet WILES (juil. 2014). “Predicting temporal sequences using an event-based spiking neural network incorporating learnable delays”. In : *2014 International Joint Conference on Neural Networks (IJCNN)*. ISSN : 2161-4407, p. 3213-3220. DOI : 10.1109/IJCNN.2014.6889850.
- GIULIONI, Massimiliano et al. (2016). “Event-Based Computation of Motion Flow on a Neuromorphic Analog Neural Platform”. eng. In : *Frontiers in Neuroscience* 10, p. 35. ISSN : 1662-4548. DOI : 10.3389/fnins.2016.00035.
- GOLLISCH, Tim et Markus MEISTER (fév. 2008). “Rapid Neural Coding in the Retina with Relative Spike Latencies”. en. In : *Science* 319.5866. Publisher : American Association for the Advancement of Science Section : Report, p. 1108-1111. ISSN : 0036-8075, 1095-9203. DOI : 10.1126/science.1149639. URL : <https://science.sciencemag.org/content/319/5866/1108> (visité le 08/06/2021).
- GOMEZ-GONZALEZ, Sebastian et al. (avr. 2020). “Real Time Trajectory Prediction Using Deep Conditional Generative Models”. In : *IEEE Robotics and Automation Letters* 5.2. Conference Name : IEEE Robotics and Automation Letters, p. 970-976. ISSN : 2377-3766. DOI : 10.1109/LRA.2020.2966390.
- GOODMAN, Dan et Romain BRETTE (2009). “The Brian simulator”. In : *Frontiers in Neuroscience* 3, p. 26. ISSN : 1662-453X. DOI : 10.3389/neuro.01.026.2009. URL : <https://www.frontiersin.org/article/10.3389/neuro.01.026.2009> (visité le 20/09/2021).
- GU, Jiuxiang et al. (mai 2018). “Recent advances in convolutional neural networks”. en. In : *Pattern Recognition* 77, p. 354-377. ISSN : 0031-3203. DOI : 10.1016/j.patcog.2017.10.013. URL : <https://www.sciencedirect.com/science/article/pii/S0031320317304120> (visité le 15/09/2021).
- GUO, Wenzhe et al. (2021). “Neural Coding in Spiking Neural Networks : A Comparative Study for Robust Neuromorphic Systems”. English. In : *Frontiers in Neuroscience* 15. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2021.638474. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2021.638474/full> (visité le 28/06/2021).

- 
- GUYONNEAU, Rudy, Rufin VANRULLEN et Simon J. THORPE (avr. 2005). “Neurons tune to the earliest spikes through STDP”. eng. In : *Neural Computation* 17.4, p. 859-879. ISSN : 0899-7667. DOI : 10.1162/0899766053429390.
- HAESSIG, Germain, Xavier BERTHELON et al. (mar. 2019). “A Spiking Neural Network Model of Depth from Defocus for Event-based Neuromorphic Vision”. en. In : *Scientific Reports* 9.1. Number : 1 Publisher : Nature Publishing Group, p. 3744. ISSN : 2045-2322. DOI : 10.1038/s41598-019-40064-0. URL : <https://www.nature.com/articles/s41598-019-40064-0> (visité le 20/05/2021).
- HAESSIG, Germain, Andrew CASSIDY et al. (août 2018). “Spiking Optical Flow for Event-Based Sensors Using IBM’s TrueNorth Neurosynaptic System”. In : *IEEE Transactions on Biomedical Circuits and Systems* 12.4. Conference Name : IEEE Transactions on Biomedical Circuits and Systems, p. 860-870. ISSN : 1940-9990. DOI : 10.1109/TBCAS.2018.2834558.
- HAMILTON, Gordon R. et Christoph REINSCHMIDT (jan. 1997). “Optimal trajectory for the basketball free throw”. In : *Journal of Sports Sciences* 15.5. Publisher : Routledge \_eprint : <https://doi.org/10.1080/026404197367137>, p. 491-504. ISSN : 0264-0414. DOI : 10.1080/026404197367137. URL : <https://doi.org/10.1080/026404197367137> (visité le 07/10/2021).
- HAO, Yunzhe et al. (jan. 2020). “A biologically plausible supervised learning method for spiking neural networks using the symmetric STDP rule”. en. In : *Neural Networks* 121, p. 387-395. ISSN : 0893-6080. DOI : 10.1016/j.neunet.2019.09.007. URL : <https://www.sciencedirect.com/science/article/pii/S0893608019302680> (visité le 28/05/2021).
- HE, Weihua et al. (déc. 2020). “Comparing SNNs and RNNs on neuromorphic vision datasets : Similarities and differences”. en. In : *Neural Networks* 132, p. 108-120. ISSN : 0893-6080. DOI : 10.1016/j.neunet.2020.08.001. URL : <https://www.sciencedirect.com/science/article/pii/S0893608020302902> (visité le 16/09/2021).
- HEBART, Martin N. et Guido HESSELMANN (juin 2012). “What Visual Information Is Processed in the Human Dorsal Stream?” en. In : *Journal of Neuroscience* 32.24. Publisher : Society for Neuroscience Section : Journal Club, p. 8107-8109. ISSN : 0270-6474, 1529-2401. DOI : 10.1523/JNEUROSCI.1462-12.2012. URL : <https://www.jneurosci.org/content/32/24/8107> (visité le 29/07/2021).
- HEEGER, David et Professor David HEEGER (2000). *Poisson Model of Spike Generation*.
- HERAULT, J. (juil. 1996). “A model of colour processing in the retina of vertebrates : From photoreceptors to colour opposition and colour constancy phenomena”. en. In : *Neurocomputing*. Current European Neurocomputing Research 12.2, p. 113-129. ISSN : 0925-2312. DOI : 10.1016/0925-2312(95)00114-X. URL : <https://www.sciencedirect.com/science/article/pii/092523129500114X> (visité le 11/05/2021).
- HILDEBRAND, G. D. et A. FIELDER (2011). “Anatomy and Physiology of the Retina”. In : DOI : 10.1007/978-3-642-12041-1\_2.
- HODGKIN, A. L. et A. F. HUXLEY (août 1952). “A quantitative description of membrane current and its application to conduction and excitation in nerve”. In : *The Journal of Physiology* 117.4, p. 500-544. ISSN : 0022-3751. URL : <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1392413/> (visité le 08/06/2021).

- 
- HOGENDOORN, Hinze et Anthony N. BURKITT (mai 2019). “Predictive Coding with Neural Transmission Delays : A Real-Time Temporal Alignment Hypothesis”. In : *eNeuro* 6.2, ENEURO.0412-18.2019. ISSN : 2373-2822. DOI : 10.1523/ENEURO.0412-18.2019. URL : <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6506824/> (visité le 20/09/2021).
- HOLEŠOVSKÝ, Ondřej, V. HLAVÁČ et R. ŠKOVIERA (2020). *Practical high-speed motion sensing : event cameras vs. global shutter*. en. URL : <https://www.semanticscholar.org/paper/Practical-high-speed-motion-sensing%3A-event-cameras-Hole%25A1ovsk%C3%BD-Hlav%C3%A1%C4%8D/20d64c0d9c219e07bc97f453fe12e34c3> (visité le 07/10/2021).
- HOLEŠOVSKÝ, Ondřej, Radoslav ŠKOVIERA et al. (jan. 2021). “Experimental Comparison between Event and Global Shutter Cameras”. en. In : *Sensors* 21.4. Number : 4 Publisher : Multidisciplinary Digital Publishing Institute, p. 1137. DOI : 10.3390/s21041137. URL : <https://www.mdpi.com/1424-8220/21/4/1137> (visité le 07/10/2021).
- HOON, Mrinalini et al. (sept. 2014). “Functional architecture of the retina : Development and disease”. en. In : *Progress in Retinal and Eye Research* 42, p. 44-84. ISSN : 1350-9462. DOI : 10.1016/j.preteyeres.2014.06.003. URL : <https://www.sciencedirect.com/science/article/pii/S135094621400038X> (visité le 12/05/2021).
- HOPKINS, Michael et al. (août 2018). “Spiking neural networks for computer vision”. In : *Interface Focus* 8.4. Publisher : Royal Society, p. 20180007. DOI : 10.1098/rsfs.2018.0007. URL : <https://royalsocietypublishing.org/doi/10.1098/rsfs.2018.0007> (visité le 09/07/2021).
- HOWELL, Jessie et al. (août 2020). “High-speed particle detection and tracking in microfluidic devices using event-based sensing”. en. In : *Lab on a Chip* 20.16. Publisher : The Royal Society of Chemistry, p. 3024-3035. ISSN : 1473-0189. DOI : 10.1039/D0LC00556H. URL : <https://pubs.rsc.org/en/content/articlelanding/2020/lc/d01c00556h> (visité le 14/09/2021).
- HU, Yuhuang et al. (2016). “DVS Benchmark Datasets for Object Tracking, Action Recognition, and Object Recognition”. English. In : *Frontiers in Neuroscience* 10. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2016.00405. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2016.00405/full> (visité le 05/07/2021).
- HUBEL, D. H. et T. N. WIESEL (jan. 1962). “Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex”. In : *The Journal of Physiology* 160.1, p. 106-154.2. ISSN : 0022-3751. URL : <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1359523/> (visité le 29/07/2021).
- HUNG, Chou P. et al. (nov. 2005). “Fast Readout of Object Identity from Macaque Inferior Temporal Cortex”. en. In : *Science* 310.5749. Publisher : American Association for the Advancement of Science Section : Report, p. 863-866. ISSN : 0036-8075, 1095-9203. DOI : 10.1126/science.1117593. URL : <https://science.sciencemag.org/content/310/5749/863> (visité le 07/06/2021).
- HÜNING, Harald, Helmut GLÜNDER et Günther PALM (avr. 1998). “Synaptic Delay Learning in Pulse-Coupled Neurons”. In : *Neural Computation* 10.3. Conference Name : Neural Computation, p. 555-565. ISSN : 0899-7667. DOI : 10.1162/089976698300017665.

- 
- IYER, Laxmi R., Yansong CHUA et Haizhou LI (2021). “Is Neuromorphic MNIST Neuromorphic? Analyzing the Discriminative Power of Neuromorphic Datasets in the Time Domain”. English. In : *Frontiers in Neuroscience* 15. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2021.608567. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2021.608567/full> (visité le 23/04/2021).
- IZHIKEVICH, E.M. (nov. 2003). “Simple model of spiking neurons”. In : *IEEE Transactions on Neural Networks* 14.6. Conference Name : IEEE Transactions on Neural Networks, p. 1569-1572. ISSN : 1941-0093. DOI : 10.1109/TNN.2003.820440.
- (sept. 2004). “Which model to use for cortical spiking neurons?” In : *IEEE Transactions on Neural Networks* 15.5. Conference Name : IEEE Transactions on Neural Networks, p. 1063-1070. ISSN : 1941-0093. DOI : 10.1109/TNN.2004.832719.
- JAEGER, Herbert (jan. 2001). “The "echo state" approach to analysing and training recurrent neural networks-with an erratum note”. In : *Bonn, Germany : German National Research Center for Information Technology GMD Technical Report* 148.
- (jan. 2002). “Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the echo state network approach”. In : *GMD-Forschungszentrum Informationstechnik, 2002*. 5.
- KAISER, Jacques et al. (sept. 2017). “Scaling up liquid state machines to predict over address events from dynamic vision sensors”. en. In : *Bioinspiration & Biomimetics* 12.5, p. 055001. ISSN : 1748-3190. DOI : 10.1088/1748-3190/aa7663. URL : <https://iopscience.iop.org/article/10.1088/1748-3190/aa7663> (visité le 19/09/2020).
- KAPLAN, Bernhard A. et al. (2013). “Anisotropic connectivity implements motion-based prediction in a spiking neural network”. English. In : *Frontiers in Computational Neuroscience* 7. ISSN : 1662-5188. DOI : 10.3389/fncom.2013.00112. URL : <https://www.frontiersin.org/articles/10.3389/fncom.2013.00112/full#B35> (visité le 07/06/2020).
- KAR, Kohitij et al. (juin 2019). “Evidence that recurrent circuits are critical to the ventral stream’s execution of core object recognition behavior”. en. In : *Nature Neuroscience* 22.6. Number : 6 Publisher : Nature Publishing Group, p. 974-983. ISSN : 1546-1726. DOI : 10.1038/s41593-019-0392-5. URL : <https://www.nature.com/articles/s41593-019-0392-5> (visité le 07/06/2021).
- KAYSER, Christoph et al. (fév. 2009). “Spike-Phase Coding Boosts and Stabilizes Information Carried by Spatial and Temporal Spike Patterns”. en. In : *Neuron* 61.4, p. 597-608. ISSN : 0896-6273. DOI : 10.1016/j.neuron.2009.01.008. URL : <https://www.sciencedirect.com/science/article/pii/S0896627309000750> (visité le 28/06/2021).
- KENNEDY, Emily et al. (juil. 2015). “Biomimicry : A Path to Sustainable Innovation”. In : *Design Issues* 31.3, p. 66-73. ISSN : 0747-9360. DOI : 10.1162/DESI\_a\_00339. URL : [https://doi.org/10.1162/DESI\\_a\\_00339](https://doi.org/10.1162/DESI_a_00339) (visité le 30/07/2021).
- KHERADPISHEH, Saeed Reza et Timothée MASQUELIER (mar. 2020). “Temporal Backpropagation for Spiking Neural Networks with One Spike per Neuron”. In : *International Journal of Neural Systems* 30.06. Publisher : World Scientific Publishing Co., p. 2050027. ISSN : 0129-0657. DOI : 10.1142/S0129065720500276.

---

URL : <https://www.worldscientific.com/doi/abs/10.1142/S0129065720500276>  
(visité le 10/06/2021).

- KHERADPISHEH, Saeed-Reza et al. (mar. 2018). “STDP-based spiking deep convolutional neural networks for object recognition”. en. In : *Neural Networks* 99, p. 56-67. ISSN : 0893-6080. DOI : 10.1016/j.neunet.2017.12.005. URL : <http://www.sciencedirect.com/science/article/pii/S0893608017302903> (visité le 20/10/2020).
- KHODAMORADI, Alireza, Kristof DENOLF et Ryan KASTNER (fév. 2021). “S2N2 : A FPGA Accelerator for Streaming Spiking Neural Networks”. In : *The 2021 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*. FPGA '21. New York, NY, USA : Association for Computing Machinery, p. 194-205. ISBN : 978-1-4503-8218-2. DOI : 10.1145/3431920.3439283. URL : <https://doi.org/10.1145/3431920.3439283> (visité le 09/07/2021).
- KIM, Hanme, Stefan LEUTENEGGER et Andrew J. DAVISON (2016). “Real-Time 3D Reconstruction and 6-DoF Tracking with an Event Camera”. en. In : *Computer Vision – ECCV 2016*. Sous la dir. de Bastian LEIBE et al. Lecture Notes in Computer Science. Cham : Springer International Publishing, p. 349-364. ISBN : 978-3-319-46466-4. DOI : 10.1007/978-3-319-46466-4\_21.
- KISTLER, Werner M., Wulfram GERSTNER et J. Leo van HEMMEN (juil. 1997). “Reduction of the Hodgkin-Huxley Equations to a Single-Variable Threshold Model”. In : *Neural Computation* 9.5. Conference Name : Neural Computation, p. 1015-1045. ISSN : 0899-7667. DOI : 10.1162/neco.1997.9.5.1015.
- KOOPMAN, Arne, Matthijs Van LEEUWEN et Jilles VREEKEN (2003). “Dynamic Neural Networks, Comparing Spiking Circuits and LSTM”. In : *Utrecht University*.
- KRIZHEVSKY, Alex, Ilya SUTSKEVER et Geoffrey E. HINTON (mai 2017). “ImageNet classification with deep convolutional neural networks”. In : *Communications of the ACM* 60.6, p. 84-90. ISSN : 0001-0782. DOI : 10.1145/3065386. URL : <https://doi.org/10.1145/3065386> (visité le 08/06/2021).
- LAGORCE, Xavier et al. (fév. 2015). “Spatiotemporal features for asynchronous event-based data”. In : *Frontiers in Neuroscience* 9. ISSN : 1662-4548. DOI : 10.3389/fnins.2015.00046. URL : <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4338664/> (visité le 07/05/2021).
- LÄNGKVIST, Martin, Lars KARLSSON et Amy LOUFI (juin 2014). “A review of unsupervised feature learning and deep learning for time-series modeling”. en. In : *Pattern Recognition Letters* 42, p. 11-24. ISSN : 0167-8655. DOI : 10.1016/j.patrec.2014.01.008. URL : <https://www.sciencedirect.com/science/article/pii/S0167865514000221> (visité le 19/07/2021).
- LEE, Alex X. et al. (avr. 2018). “Stochastic Adversarial Video Prediction”. In : *arXiv :1804.01523 [cs]*. arXiv : 1804.01523. URL : <http://arxiv.org/abs/1804.01523> (visité le 01/06/2021).
- LEE, Chankyu et al. (2018). “Training Deep Spiking Convolutional Neural Networks With STDP-Based Unsupervised Pre-training Followed by Supervised Fine-Tuning”. English. In : *Frontiers in Neuroscience* 12. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2018.00435. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2018.00435/full> (visité le 20/10/2020).

- 
- LEE, Jun Haeng, Tobi DELBRUCK et Michael PFEIFFER (2016). “Training Deep Spiking Neural Networks Using Backpropagation”. English. In : *Frontiers in Neuroscience* 10. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2016.00508. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2016.00508/full> (visité le 23/04/2021).
- LEUNG, F.H.F. et al. (jan. 2003). “Tuning of the structure and parameters of a neural network using an improved genetic algorithm”. In : *IEEE Transactions on Neural Networks* 14.1. Conference Name : IEEE Transactions on Neural Networks, p. 79-88. ISSN : 1941-0093. DOI : 10.1109/TNN.2002.804317.
- LI, Hongmin et al. (2017). “CIFAR10-DVS : An Event-Stream Dataset for Object Classification”. English. In : *Frontiers in Neuroscience* 11. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2017.00309. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2017.00309/full> (visité le 08/06/2021).
- LI, Yawei et Tian FENG (jan. 2020). “The effects of sport expertise and shot results on basketball players’ action anticipation”. en. In : *PLOS ONE* 15.1. Publisher : Public Library of Science, e0227521. ISSN : 1932-6203. DOI : 10.1371/journal.pone.0227521. URL : <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0227521> (visité le 01/06/2021).
- LICHTSTEINER, P., C. POSCH et T. DELBRUCK (fév. 2008). “A 128 x 128 120 dB 15 $\mu$ s Latency Asynchronous Temporal Contrast Vision Sensor”. In : *IEEE Journal of Solid-State Circuits* 43.2, p. 566-576. ISSN : 0018-9200. DOI : 10.1109/JSSC.2007.914337.
- LINARES-BARRANCO, Alejandro et al. (2019). “Low Latency Event-Based Filtering and Feature Extraction for Dynamic Vision Sensors in Real-Time FPGA Applications”. In : *IEEE Access* 7. Conference Name : IEEE Access, p. 134926-134942. ISSN : 2169-3536. DOI : 10.1109/ACCESS.2019.2941282.
- LIU, Qian et al. (2016). “Benchmarking Spike-Based Visual Recognition : A Dataset and Evaluation”. English. In : *Frontiers in Neuroscience* 10. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2016.00496. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2016.00496/full> (visité le 05/07/2021).
- LIU, Shih-Chii et al. (août 2014). “Asynchronous Binaural Spatial Audition Sensor With 2 $\times$  64 Channel Output”. In : *IEEE Transactions on Biomedical Circuits and Systems* 8.4. Conference Name : IEEE Transactions on Biomedical Circuits and Systems, p. 453-464. ISSN : 1940-9990. DOI : 10.1109/TBCAS.2013.2281834.
- LIU, Tao et al. (mar. 2018). “MT-Spike : A Multilayer Time-based Spiking Neuromorphic Architecture with Temporal Error Backpropagation”. In : *arXiv :1803.05117 [cs, q-bio]*. arXiv : 1803.05117. URL : <http://arxiv.org/abs/1803.05117> (visité le 05/07/2021).
- LIU, Ziwei et al. (août 2017). “Video Frame Synthesis using Deep Voxel Flow”. In : *arXiv :1702.02463 [cs]*. arXiv : 1702.02463. URL : <http://arxiv.org/abs/1702.02463> (visité le 01/06/2021).
- LOFFING, Florian et Rouwen CAÑAL-BRULAND (août 2017). “Anticipation in sport”. en. In : *Current Opinion in Psychology*. Sport psychology 16, p. 6-11. ISSN : 2352-250X. DOI : 10.1016/j.copsyc.2017.03.008. URL : <https://www>.

- 
- [sciencedirect.com/science/article/pii/S2352250X16301439](https://www.sciencedirect.com/science/article/pii/S2352250X16301439) (visité le 20/09/2021).
- LOWEL, S. et W. SINGER (jan. 1992). “Selection of intrinsic horizontal connections in the visual cortex by correlated neuronal activity”. en. In : *Science* 255.5041. Publisher : American Association for the Advancement of Science Section : Reports, p. 209-212. ISSN : 0036-8075, 1095-9203. DOI : 10.1126/science.1372754. URL : <https://science.sciencemag.org/content/255/5041/209> (visité le 07/07/2021).
- MA, Bo et al. (déc. 2016). “Visual Tracking Under Motion Blur”. In : *IEEE Transactions on Image Processing* 25.12. Conference Name : IEEE Transactions on Image Processing, p. 5867-5876. ISSN : 1941-0042. DOI : 10.1109/TIP.2016.2615812.
- MA, Ziyang et al. (juin 2015). “Handling motion blur in multi-frame super-resolution”. In : *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. ISSN : 1063-6919, p. 5224-5232. DOI : 10.1109/CVPR.2015.7299159.
- MAASS, Wolfgang (déc. 1997). “Networks of spiking neurons : The third generation of neural network models”. en. In : *Neural Networks* 10.9, p. 1659-1671. ISSN : 0893-6080. DOI : 10.1016/S0893-6080(97)00011-7. URL : <https://www.sciencedirect.com/science/article/pii/S0893608097000117> (visité le 06/07/2021).
- (déc. 2015). “To Spike or Not to Spike : That Is the Question”. In : *Proceedings of the IEEE* 103.12. Conference Name : Proceedings of the IEEE, p. 2219-2224. ISSN : 1558-2256. DOI : 10.1109/JPROC.2015.2496679.
- MAASS, Wolfgang, Robert LEGENSTEIN et al. (nov. 2002). “A New Approach towards Vision Suggested by Biologically Realistic Neural Microcircuit Models”. In : t. 2525. DOI : 10.1007/3-540-36181-2\_28.
- MAASS, Wolfgang, Thomas NATSCHLÄGER et Henry MARKRAM (nov. 2002). “Real-time computing without stable states : a new framework for neural computation based on perturbations”. eng. In : *Neural Computation* 14.11, p. 2531-2560. ISSN : 0899-7667. DOI : 10.1162/089976602760407955.
- MAGUIRE, L. P. et al. (déc. 2007). “Challenges for large-scale implementations of spiking neural networks on FPGAs”. en. In : *Neurocomputing. Dedicated Hardware Architectures for Intelligent Systems* 71.1, p. 13-29. ISSN : 0925-2312. DOI : 10.1016/j.neucom.2006.11.029. URL : <https://www.sciencedirect.com/science/article/pii/S0925231207002123> (visité le 09/07/2021).
- MAHOWALD, M. A. et C. MEAD (mai 1991). “The silicon retina”. eng. In : *Scientific American* 264.5, p. 76-82. ISSN : 0036-8733. DOI : 10.1038/scientificamerican0591-76.
- MAINEN, Z. F. et T. J. SEJNOWSKI (juin 1995). “Reliability of spike timing in neocortical neurons”. en. In : *Science* 268.5216. Publisher : American Association for the Advancement of Science Section : Reports, p. 1503-1506. ISSN : 0036-8075, 1095-9203. DOI : 10.1126/science.7770778. URL : <https://science.sciencemag.org/content/268/5216/1503> (visité le 09/06/2021).
- MAKRIDAKIS, Spyros (1976). “A Survey of Time Series”. In : *International Statistical Review / Revue Internationale de Statistique* 44.1. Publisher : [Wiley, International Statistical Institute (ISI)], p. 29-70. ISSN : 0306-7734. DOI : 10.2307/1402964. URL : <https://www.jstor.org/stable/1402964> (visité le 01/06/2021).



- 
- MARTINEZ, Julieta, Michael J. BLACK et Javier ROMERO (juil. 2017). “On Human Motion Prediction Using Recurrent Neural Networks”. In : *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. ISSN : 1063-6919, p. 4674-4683. DOI : 10.1109/CVPR.2017.497.
- MASQUELIER, Timothée (juin 2012). “Relative spike time coding and STDP-based orientation selectivity in the early visual system in natural continuous and saccadic vision : a computational model”. en. In : *Journal of Computational Neuroscience* 32.3, p. 425-441. ISSN : 1573-6873. DOI : 10.1007/s10827-011-0361-9. URL : <https://doi.org/10.1007/s10827-011-0361-9> (visité le 26/02/2020).
- MASQUELIER, Timothée, Rudy GUYONNEAU et Simon J. THORPE (jan. 2008). “Spike Timing Dependent Plasticity Finds the Start of Repeating Patterns in Continuous Spike Trains”. In : *PLoS ONE* 3.1. ISSN : 1932-6203. DOI : 10.1371/journal.pone.0001377. URL : <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2147052/> (visité le 22/08/2018).
- MASQUELIER, Timothée et Simon J. THORPE (fév. 2007). “Unsupervised Learning of Visual Features through Spike Timing Dependent Plasticity”. en. In : *PLOS Computational Biology* 3.2, e31. ISSN : 1553-7358. DOI : 10.1371/journal.pcbi.0030031. URL : <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.0030031> (visité le 15/05/2019).
- MASSA, Riccardo et al. (jan. 2021). “An Efficient Spiking Neural Network for Recognizing Gestures with a DVS Camera on the Loihi Neuromorphic Processor”. In : *arXiv :2006.09985 [cs, stat]*. arXiv : 2006.09985. URL : <http://arxiv.org/abs/2006.09985> (visité le 15/09/2021).
- MASUMORI, Atsushi, Lana SINAPAYEN et Takashi IKEGAMI (nov. 2019). “Predictive Coding as Stimulus Avoidance in Spiking Neural Networks”. In : *arXiv :1911.09230 [cs, q-bio]*. arXiv : 1911.09230. URL : <http://arxiv.org/abs/1911.09230> (visité le 01/05/2021).
- MATHER, George (fév. 1996). “Image blur as a pictorial depth cue”. In : *Proceedings of the Royal Society of London. Series B : Biological Sciences* 263.1367. Publisher : Royal Society, p. 169-172. DOI : 10.1098/rspb.1996.0027. URL : <https://royalsocietypublishing.org/doi/10.1098/rspb.1996.0027> (visité le 09/06/2021).
- MATHER, George et David R R SMITH (oct. 2002). “Blur Discrimination and its Relation to Blur-Mediated Depth Perception”. en. In : *Perception* 31.10. Publisher : SAGE Publications Ltd STM, p. 1211-1219. ISSN : 0301-0066. DOI : 10.1068/p3254. URL : <https://doi.org/10.1068/p3254> (visité le 09/06/2021).
- MEAD, C. (oct. 1990). “Neuromorphic electronic systems”. In : *Proceedings of the IEEE* 78.10. Conference Name : Proceedings of the IEEE, p. 1629-1636. ISSN : 1558-2256. DOI : 10.1109/5.58356.
- MEROLLA, Paul A. et al. (août 2014). “A million spiking-neuron integrated circuit with a scalable communication network and interface”. en. In : *Science* 345.6197. Publisher : American Association for the Advancement of Science Section : Report, p. 668-673. ISSN : 0036-8075, 1095-9203. DOI : 10.1126/science.1254642. URL : <https://science.sciencemag.org/content/345/6197/668> (visité le 08/07/2021).
- MIAO, Shu et al. (2019). “Neuromorphic Vision Datasets for Pedestrian Detection, Action Recognition, and Fall Detection”. English. In : *Frontiers in Neurorobotics*

- 
13. Publisher : Frontiers. ISSN : 1662-5218. DOI : 10.3389/fnbot.2019.00038. URL : <https://www.frontiersin.org/articles/10.3389/fnbot.2019.00038/full> (visité le 05/07/2021).
- MO, Lingfei, Xinao CHEN et Gang WANG (jan. 2021). “EDHA : Event-Driven High Accurate Simulator for Spike Neural Networks”. en. In : *Electronics* 10.18. Number : 18 Publisher : Multidisciplinary Digital Publishing Institute, p. 2281. DOI : 10.3390/electronics10182281. URL : <https://www.mdpi.com/2079-9292/10/18/2281> (visité le 22/09/2021).
- MOHAMMADI, Arash et al. (oct. 2017). “OpenGA, a C++ Genetic Algorithm Library”. In : *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, p. 2051-2056. DOI : 10.1109/SMC.2017.8122921.
- MOHEMMED, Ammar et al. (août 2012). “Span : spike pattern association neuron for learning spatio-temporal spike patterns”. eng. In : *International Journal of Neural Systems* 22.4, p. 1250012. ISSN : 1793-6462. DOI : 10.1142/S0129065712500128.
- MORAITIS, Timoleon, Abu SEBASTIAN et Evangelos ELEFThERIOU (sept. 2020). “Short-term synaptic plasticity optimally models continuous environments”. In : *arXiv :2009.06808 [cs, q-bio]*. arXiv : 2009.06808. URL : <http://arxiv.org/abs/2009.06808> (visité le 23/04/2021).
- MORENO-TORRES, Jose G. et al. (jan. 2012). “A unifying view on dataset shift in classification”. en. In : *Pattern Recognition* 45.1, p. 521-530. ISSN : 0031-3203. DOI : 10.1016/j.patcog.2011.06.019. URL : <https://www.sciencedirect.com/science/article/pii/S0031320311002901> (visité le 20/08/2021).
- MOZAFARI, M. et al. (déc. 2018). “First-Spike-Based Visual Categorization Using Reward-Modulated STDP”. In : *IEEE Transactions on Neural Networks and Learning Systems* 29.12. Conference Name : IEEE Transactions on Neural Networks and Learning Systems, p. 6178-6190. ISSN : 2162-2388. DOI : 10.1109/TNNLS.2018.2826721.
- MOZAFARI, Milad, Mohammad GANJTABESH, Abbas NOWZARI-DALINI et Timothée MASQUELIER (2019). “SpykeTorch : Efficient Simulation of Convolutional Spiking Neural Networks With at Most One Spike per Neuron”. In : *Frontiers in Neuroscience* 13, p. 625. ISSN : 1662-453X. DOI : 10.3389/fnins.2019.00625. URL : <https://www.frontiersin.org/article/10.3389/fnins.2019.00625> (visité le 20/09/2021).
- MOZAFARI, Milad, Mohammad GANJTABESH, Abbas NOWZARI-DALINI, Simon J. THORPE et al. (oct. 2019). “Bio-inspired digit recognition using reward-modulated spike-timing-dependent plasticity in deep convolutional networks”. In : *Pattern Recognition* 94. arXiv : 1804.00227, p. 87-95. ISSN : 00313203. DOI : 10.1016/j.patcog.2019.05.015. URL : <http://arxiv.org/abs/1804.00227> (visité le 15/09/2021).
- MÜLLING, Katharina, Jens KOBER et Jan PETERS (oct. 2010). “A biomimetic approach to robot table tennis”. In : *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. ISSN : 2153-0866, p. 1921-1926. DOI : 10.1109/IROS.2010.5650305.
- NEIL, Daniel, Michael PFEIFFER et Shih-Chii LIU (avr. 2016). “Learning to be efficient : algorithms for training low-latency, low-compute deep spiking neural networks”. In : *Proceedings of the 31st Annual ACM Symposium on Applied Computing*. SAC '16. New York, NY, USA : Association for Computing Machinery,

- 
- p. 293-298. ISBN : 978-1-4503-3739-7. DOI : 10.1145/2851613.2851724. URL : <https://doi.org/10.1145/2851613.2851724> (visité le 22/09/2021).
- NGUYEN, Duy-Anh, Xuan-Tu TRAN et Francesca IACOPI (juin 2021). “A Review of Algorithms and Hardware Implementations for Spiking Neural Networks”. en. In : *Journal of Low Power Electronics and Applications* 11.2. Number : 2 Publisher : Multidisciplinary Digital Publishing Institute, p. 23. DOI : 10.3390/jlpea11020023. URL : <https://www.mdpi.com/2079-9268/11/2/23> (visité le 09/07/2021).
- NGUYEN, Vincent A., Ian P. HOWARD et Robert S. ALLISON (avr. 2005). “Detection of the depth order of defocused images”. en. In : *Vision Research* 45.8, p. 1003-1011. ISSN : 0042-6989. DOI : 10.1016/j.visres.2004.10.015. URL : <https://www.sciencedirect.com/science/article/pii/S0042698904005413> (visité le 09/06/2021).
- NIJHAWAN, Romi (juil. 1994). “Motion extrapolation in catching”. en. In : *Nature* 370.6487. Bandiera\_abtest : a Cg\_type : Nature Research Journals Number : 6487 Primary\_atype : Research Publisher : Nature Publishing Group, p. 256-257. ISSN : 1476-4687. DOI : 10.1038/370256b0. URL : <https://www.nature.com/articles/370256b0> (visité le 21/09/2021).
- (avr. 2008). “Visual prediction : Psychophysics and neurophysiology of compensation for time delays”. en. In : *Behavioral and Brain Sciences* 31.2. Publisher : Cambridge University Press, p. 179-198. ISSN : 1469-1825, 0140-525X. DOI : 10.1017/S0140525X08003804. URL : <https://www.cambridge.org/core/journals/behavioral-and-brain-sciences/article/abs/visual-prediction-psychophysics-and-neurophysiology-of-compensation-for-time-delays/F0AE60F24239AD6435157A2C401C8BC6> (visité le 21/09/2021).
- NIKHIL, Nishant et Brendan Tran MORRIS (2019). “Convolutional Neural Network for Trajectory Prediction”. en. In : *Computer Vision – ECCV 2018 Workshops*. Sous la dir. de Laura LEAL-TAIXÉ et Stefan ROTH. Lecture Notes in Computer Science. Cham : Springer International Publishing, p. 186-196. ISBN : 978-3-030-11015-4. DOI : 10.1007/978-3-030-11015-4\_16.
- NINIO, Jacques (1989). *L’empreinte des sens*. fr. Odile Jacob. URL : [https://www.odilejacob.fr/catalogue/sciences/neurosciences/empreinte-des-sens\\_9782738127228.php](https://www.odilejacob.fr/catalogue/sciences/neurosciences/empreinte-des-sens_9782738127228.php) (visité le 20/09/2021).
- NØKLAND, Arild et Lars Hiller EIDNES (mai 2019). “Training Neural Networks with Local Error Signals”. In : *arXiv :1901.06656 [cs, stat]*. arXiv : 1901.06656. URL : <http://arxiv.org/abs/1901.06656> (visité le 20/09/2021).
- NOLTE, Aleke et al. (mar. 2017). “A Note on the Depth-from-Defocus Mechanism of Jumping Spiders”. In : *Biomimetics* 2.1. ISSN : 2313-7673. DOI : 10.3390/biomimetics2010003. URL : <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6477618/> (visité le 09/06/2021).
- O’REGAN, J. Kevin et Alva NOË (oct. 2001). “A sensorimotor account of vision and visual consciousness”. en. In : *Behavioral and Brain Sciences* 24.5. Publisher : Cambridge University Press, p. 939-973. ISSN : 1469-1825, 0140-525X. DOI : 10.1017/S0140525X01000115. URL : <https://www.cambridge.org/core/journals/behavioral-and-brain-sciences/article/abs/sensorimotor-account-of-vision-and-visual-consciousness/BA1638CB7389102A12B336CE687EC270> (visité le 20/09/2021).

- 
- OHZAWA, I., G. C. DEANGELIS et R. D. FREEMAN (août 1990). “Stereoscopic depth discrimination in the visual cortex : neurons ideally suited as disparity detectors”. eng. In : *Science (New York, N. Y.)* 249.4972, p. 1037-1041. ISSN : 0036-8075. DOI : 10.1126/science.2396096.
- OLAWOYIN, Anifat et Yangjuin CHEN (jan. 2018). “Predicting the Future with Artificial Neural Network”. en. In : *Procedia Computer Science*. Cyber Physical Systems and Deep Learning Chicago, Illinois November 5-7, 2018 140, p. 383-392. ISSN : 1877-0509. DOI : 10.1016/j.procs.2018.10.300. URL : <https://www.sciencedirect.com/science/article/pii/S1877050918319732> (visité le 01/06/2021).
- OLSHAUSEN, Bruno A et David J FIELD (août 2004). “Sparse coding of sensory inputs”. en. In : *Current Opinion in Neurobiology* 14.4, p. 481-487. ISSN : 0959-4388. DOI : 10.1016/j.conb.2004.07.007. URL : <https://www.sciencedirect.com/science/article/pii/S0959438804001035> (visité le 19/05/2021).
- (déc. 1997). “Sparse coding with an overcomplete basis set : A strategy employed by V1?” en. In : *Vision Research* 37.23, p. 3311-3325. ISSN : 0042-6989. DOI : 10.1016/S0042-6989(97)00169-7. URL : <https://www.sciencedirect.com/science/article/pii/S0042698997001697> (visité le 19/05/2021).
- ORCHARD, G. et al. (oct. 2013). “A spiking neural network architecture for visual motion estimation”. In : *2013 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, p. 298-301. DOI : 10.1109/BioCAS.2013.6679698.
- ORCHARD, Garrick et al. (2015). “Converting Static Image Datasets to Spiking Neuromorphic Datasets Using Saccades”. English. In : *Frontiers in Neuroscience* 9. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2015.00437. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2015.00437/full> (visité le 23/04/2021).
- ORORBIA, Alexander (jan. 2020). “Spiking Neural Predictive Coding for Continual Learning from Data Streams”. In : *arXiv :1908.08655 [cs, q-bio]*. arXiv : 1908.08655. URL : <http://arxiv.org/abs/1908.08655> (visité le 01/05/2021).
- OSSWALD, Marc et al. (jan. 2017). “A spiking neural network model of 3D perception for event-based neuromorphic stereo vision systems”. en. In : *Scientific Reports* 7.1. Bandiera\_abtest : a Cc\_license\_type : cc\_by Cg\_type : Nature Research Journals Number : 1 Primary\_atype : Research Publisher : Nature Publishing Group Subject\_term : Computational models;Electrical and electronic engineering;Network models;Visual system Subject\_term\_id : computational-models;electrical-and-electronic-engineering;network-models;visual-system, p. 40703. ISSN : 2045-2322. DOI : 10.1038/srep40703. URL : <https://www.nature.com/articles/srep40703> (visité le 02/07/2021).
- OUDJAIL, Veís et Jean MARTINET (juin 2021a). “Bio-inspired Event-based Motion Analysis with Spiking Neural Networks”. In : p. 389-394. ISBN : 978-989-758-354-4. URL : <https://www.scitepress.org/Link.aspx?doi=10.5220/0007397303890394> (visité le 14/06/2021).
- (juin 2021b). “Meta-parameters Exploration for Unsupervised Event-based Motion Analysis”. In : p. 853-860. ISBN : 978-989-758-402-2. URL : <https://www.scitepress.org/Link.aspx?doi=10.5220/0009324908530860> (visité le 14/06/2021).

- 
- PACK, Christopher C., Vladimir K. BEREZOVSKII et Richard T. BORN (déc. 2001). “Dynamic properties of neurons in cortical area MT in alert and anaesthetized macaque monkeys”. en. In : *Nature* 414.6866. Bandiera\_abtest : a Cg\_type : Nature Research Journals Number : 6866 Primary\_atype : Research Publisher : Nature Publishing Group, p. 905-908. ISSN : 1476-4687. DOI : 10.1038/414905a. URL : <https://www.nature.com/articles/414905a> (visité le 26/07/2021).
- PADALA, Vandana, Arindam BASU et Garrick ORCHARD (2018). “A Noise Filtering Algorithm for Event-Based Asynchronous Change Detection Image Sensors on TrueNorth and Its Implementation on TrueNorth”. English. In : *Frontiers in Neuroscience* 12. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2018.00118. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2018.00118/full> (visité le 09/07/2021).
- PAREDES-VALLES, F., K. Y. W. SCHEPER et G. C. H. E. De CROON (2019). “Un-supervised Learning of a Hierarchical Spiking Neural Network for Optical Flow Estimation : From Events to Global Motion Perception”. In : *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1-1. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2019.2903179.
- PARK, Seongsik, Seijoon KIM, Hyeokjun CHOE et al. (juin 2019). “Fast and Efficient Information Transmission with Burst Spikes in Deep Spiking Neural Networks”. In : *Proceedings of the 56th Annual Design Automation Conference 2019*. DAC '19. New York, NY, USA : Association for Computing Machinery, p. 1-6. ISBN : 978-1-4503-6725-7. DOI : 10.1145/3316781.3317822. URL : <https://doi.org/10.1145/3316781.3317822> (visité le 28/06/2021).
- PARK, Seongsik, Seijoon KIM, Byunggook NA et al. (mar. 2020). “T2FSNN : Deep Spiking Neural Networks with Time-to-first-spike Coding”. In : *arXiv :2003.11741 [cs, stat]*. arXiv : 2003.11741. URL : <http://arxiv.org/abs/2003.11741> (visité le 28/06/2021).
- PARKER, Andrew J. (mai 2007). “Binocular depth perception and the cerebral cortex”. en. In : *Nature Reviews Neuroscience* 8.5. Number : 5 Publisher : Nature Publishing Group, p. 379-391. ISSN : 1471-0048. DOI : 10.1038/nrn2131. URL : <https://www.nature.com/articles/nrn2131> (visité le 19/05/2021).
- PASTERNAK, Tatiana, James W. BISLEY et David CALKINS (2003). “Visual Processing in the Primate Brain”. en. In : *Handbook of Psychology*. Section : 6 \_eprint : <https://onlinelibrary.wiley.com/doi/pdf/10.1002/0471264385.wei0306>. American Cancer Society, p. 139-185. ISBN : 978-0-471-26438-5. DOI : 10.1002/0471264385.wei0306. URL : <https://onlinelibrary.wiley.com/doi/abs/10.1002/0471264385.wei0306> (visité le 29/07/2021).
- PAUGAM-MOISY, Hélène et Sander BOHTE (2012). “Computing with Spiking Neuron Networks”. en. In : *Handbook of Natural Computing*. Sous la dir. de Grzegorz ROZENBERG, Thomas BÄCK et Joost N. KOK. Berlin, Heidelberg : Springer, p. 335-376. ISBN : 978-3-540-92910-9. DOI : 10.1007/978-3-540-92910-9\_10. URL : [https://doi.org/10.1007/978-3-540-92910-9\\_10](https://doi.org/10.1007/978-3-540-92910-9_10) (visité le 08/06/2021).
- PFEIFFER, Michael et Thomas PFEIL (2018). “Deep Learning With Spiking Neurons : Opportunities and Challenges”. English. In : *Frontiers in Neuroscience* 12. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2018.00774.

- 
- URL : <https://www.frontiersin.org/articles/10.3389/fnins.2018.00774/full#T1> (visité le 08/07/2021).
- PFISTER, Jean-Pascal et Wulfram GERSTNER (sept. 2006). “Triplets of Spikes in a Model of Spike Timing-Dependent Plasticity”. In : *Journal of Neuroscience* 26.38, p. 9673-9682.
- PLATKIEWICZ, Jonathan et Romain BRETTE (juil. 2010). “A Threshold Equation for Action Potential Initiation”. en. In : *PLoS Computational Biology* 6.7. Publisher : Public Library of Science, e1000850. ISSN : 1553-7358. DOI : 10.1371/journal.pcbi.1000850. URL : <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1000850> (visité le 29/07/2021).
- POLI, Roberto (jan. 2010). “The many aspects of anticipation”. In : *Foresight* 12.3. Sous la dir. de Riel MILLER et Roberto POLI. Publisher : Emerald Group Publishing Limited, p. 7-17. ISSN : 1463-6689. DOI : 10.1108/14636681011049839. URL : <https://doi.org/10.1108/14636681011049839> (visité le 20/09/2021).
- PONULAK, Filip et Andrzej KASIŃSKI (fév. 2010). “Supervised learning in spiking neural networks with ReSuMe : sequence learning, classification, and spike shifting”. eng. In : *Neural Computation* 22.2, p. 467-510. ISSN : 1530-888X. DOI : 10.1162/neco.2009.11-08-901.
- PORTZ, Travis, Li ZHANG et Hongrui JIANG (juin 2012). “Optical flow in the presence of spatially-varying motion blur”. In : *2012 IEEE Conference on Computer Vision and Pattern Recognition*. ISSN : 1063-6919, p. 1752-1759. DOI : 10.1109/CVPR.2012.6247871.
- POSCH, Christoph, Daniel MATOLIN et Rainer WOHLGENANT (jan. 2011). “A QVGA 143 dB Dynamic Range Frame-Free PWM Image Sensor With Lossless Pixel-Level Video Compression and Time-Domain CDS”. en. In : *IEEE Journal of Solid-State Circuits* 46.1, p. 259-275. ISSN : 0018-9200, 1558-173X. DOI : 10.1109/JSSC.2010.2085952. URL : <http://ieeexplore.ieee.org/document/5648367/> (visité le 24/09/2020).
- POSCH, Christoph, Teresa SERRANO-GOTARREDONA et al. (oct. 2014). “Retinomorphic Event-Based Vision Sensors : Bioinspired Cameras With Spiking Output”. In : *Proceedings of the IEEE* 102.10. Conference Name : Proceedings of the IEEE, p. 1470-1484. ISSN : 1558-2256. DOI : 10.1109/JPROC.2014.2346153.
- POSNER, Michael I., Mary J. NISSEN et Raymond M. KLEIN (1976). “Visual dominance : An information-processing account of its origins and significance”. In : *Psychological Review* 83.2. Place : US Publisher : American Psychological Association, p. 157-171. ISSN : 1939-1471(Electronic),0033-295X(Print). DOI : 10.1037/0033-295X.83.2.157.
- QIU, Weichao et al. (oct. 2017). “UnrealCV : Virtual Worlds for Computer Vision”. In : *Proceedings of the 25th ACM international conference on Multimedia*. MM '17. New York, NY, USA : Association for Computing Machinery, p. 1221-1224. ISBN : 978-1-4503-4906-2. DOI : 10.1145/3123266.3129396. URL : <https://doi.org/10.1145/3123266.3129396> (visité le 16/07/2021).
- QUERLIOZ, Damien et al. (mai 2013). “Immunity to Device Variations in a Spiking Neural Network With Memristive Nanodevices”. In : *IEEE Transactions on Nanotechnology* 12.3. Conference Name : IEEE Transactions on Nanotechnology, p. 288-295. ISSN : 1941-0085. DOI : 10.1109/TNANO.2013.2250995.

- 
- RAWAT, Waseem et Zenghui WANG (juin 2017). “Deep Convolutional Neural Networks for Image Classification : A Comprehensive Review”. In : *Neural Computation* 29.9. Publisher : MIT Press, p. 2352-2449. ISSN : 0899-7667. DOI : 10.1162/neco\_a\_00990. URL : [https://doi.org/10.1162/neco\\_a\\_00990](https://doi.org/10.1162/neco_a_00990) (visité le 15/10/2020).
- REBECQ, Henri, Daniel GEHRIG et Davide SCARAMUZZA (oct. 2018). “ESIM : an Open Event Camera Simulator”. en. In : *Conference on Robot Learning*. ISSN : 2640-3498. PMLR, p. 969-982. URL : <http://proceedings.mlr.press/v87/rebecq18a.html> (visité le 16/07/2021).
- REICH, Daniel S., Ferenc MECHLER et Jonathan D. VICTOR (mar. 2001). “Temporal Coding of Contrast in Primary Visual Cortex : When, What, and Why”. In : *Journal of Neurophysiology* 85.3. Publisher : American Physiological Society, p. 1039-1050. ISSN : 0022-3077. DOI : 10.1152/jn.2001.85.3.1039. URL : <https://journals.physiology.org/doi/full/10.1152/jn.2001.85.3.1039> (visité le 08/06/2021).
- REZAEI-MAZINANI, Shahab (oct. 2017). “Development of novel organic optoelectronic technologies for biomedical applications”. Thèse de doct.
- ROSENBLATT, F. (1958). “The perceptron : A probabilistic model for information storage and organization in the brain”. In : *Psychological Review* 65.6. Place : US Publisher : American Psychological Association, p. 386-408. ISSN : 1939-1471. DOI : 10.1037/h0042519.
- RUDENKO, Andrey et al. (juil. 2020). “Human Motion Trajectory Prediction : A Survey”. In : *The International Journal of Robotics Research* 39.8. arXiv : 1905.06113, p. 895-935. ISSN : 0278-3649, 1741-3176. DOI : 10.1177/0278364920917446. URL : <http://arxiv.org/abs/1905.06113> (visité le 01/06/2021).
- RUECKAUER, Bodo et al. (2017). “Conversion of Continuous-Valued Deep Networks to Efficient Event-Driven Networks for Image Classification”. English. In : *Frontiers in Neuroscience* 11. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2017.00682. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2017.00682/full> (visité le 05/07/2021).
- SAXE, Andrew M. et al. (jan. 2011). “On Random Weights and Unsupervised Feature Learning”. en. In : URL : <https://openreview.net/forum?id=BJbXZj-OZS> (visité le 14/06/2021).
- SCHEMMEL, Johannes et al. (mai 2010). “A wafer-scale neuromorphic hardware system for large-scale neural modeling”. In : *2010 IEEE International Symposium on Circuits and Systems (ISCAS)*. ISSN : 2158-1525, p. 1947-1950. DOI : 10.1109/ISCAS.2010.5536970.
- SCHRIMPF, Martin, Jonas KUBILIUS, Ha HONG et al. (sept. 2018). “Brain-Score : Which Artificial Neural Network for Object Recognition is most Brain-Like?” en. In : *bioRxiv*. Publisher : Cold Spring Harbor Laboratory Section : New Results, p. 407007. DOI : 10.1101/407007. URL : <https://www.biorxiv.org/content/10.1101/407007v1> (visité le 23/04/2021).
- SCHRIMPF, Martin, Jonas KUBILIUS, Michael J. LEE et al. (nov. 2020). “Integrative Benchmarking to Advance Neurally Mechanistic Models of Human Intelligence”. English. In : *Neuron* 108.3. Publisher : Elsevier, p. 413-423. ISSN : 0896-6273. DOI : 10.1016/j.neuron.2020.07.040. URL : [https://www.cell.com/neuron/abstract/S0896-6273\(20\)30605-X](https://www.cell.com/neuron/abstract/S0896-6273(20)30605-X) (visité le 23/04/2021).

- 
- SCHUMAN, Catherine D. et al. (mai 2017). “A Survey of Neuromorphic Computing and Neural Networks in Hardware”. In : *arXiv :1705.06963 [cs]*. arXiv : 1705.06963. URL : <http://arxiv.org/abs/1705.06963> (visité le 07/07/2021).
- SENOT, Patrice et al. (déc. 2005). “Anticipating the Effects of Gravity When Intercepting Moving Objects : Differentiating Up and Down Based on Nonvisual Cues”. In : *Journal of Neurophysiology* 94.6. Publisher : American Physiological Society, p. 4471-4480. ISSN : 0022-3077. DOI : 10.1152/jn.00527.2005. URL : <https://journals.physiology.org/doi/full/10.1152/jn.00527.2005> (visité le 20/09/2021).
- SERRANO-GOTARREDONA, T. et al. (mai 2015). “ConvNets experiments on SpiNNaker”. In : *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*. ISSN : 2158-1525, p. 2405-2408. DOI : 10.1109/ISCAS.2015.7169169.
- SERRANO-GOTARREDONA, Teresa et Bernabé LINARES-BARRANCO (2015). “Poker-DVS and MNIST-DVS. Their History, How They Were Made, and Other Details”. English. In : *Frontiers in Neuroscience* 9. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2015.00481. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2015.00481/full> (visité le 05/07/2021).
- SHE, Xueyuan et al. (2021). “A Heterogeneous Spiking Neural Network for Unsupervised Learning of Spatiotemporal Patterns”. English. In : *Frontiers in Neuroscience* 14. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2020.615756. URL : [https://www.frontiersin.org/articles/10.3389/fnins.2020.615756/full?utm\\_source=Email\\_to\\_authors\\_&utm\\_medium=Email&utm\\_content=T1\\_11.5e1\\_author&utm\\_campaign=Email\\_publication&field=&journalName=Frontiers\\_in\\_Neuroscience&id=615756](https://www.frontiersin.org/articles/10.3389/fnins.2020.615756/full?utm_source=Email_to_authors_&utm_medium=Email&utm_content=T1_11.5e1_author&utm_campaign=Email_publication&field=&journalName=Frontiers_in_Neuroscience&id=615756) (visité le 05/07/2021).
- SHMUEL, Amir et Amir GRINVALD (nov. 1996). “Functional Organization for Direction of Motion and Its Relationship to Orientation Maps in Cat Area 18”. en. In : *Journal of Neuroscience* 16.21. Publisher : Society for Neuroscience Section : Articles, p. 6945-6964. ISSN : 0270-6474, 1529-2401. DOI : 10.1523/JNEUROSCI.16-21-06945.1996. URL : <https://www.jneurosci.org/content/16/21/6945> (visité le 15/09/2021).
- SHON, A. P., R. P. N. RAO et T. J. SEJNOWSKI (jan. 2004). “Motion detection and prediction through spike-timing dependent plasticity”. In : *Network : Computation in Neural Systems* 15.3. Publisher : Taylor & Francis \_eprint : [https://doi.org/10.1088/0954-898X\\_15\\_3\\_002](https://doi.org/10.1088/0954-898X_15_3_002), p. 179-198. ISSN : 0954-898X. DOI : 10.1088/0954-898X\_15\_3\_002. URL : [https://doi.org/10.1088/0954-898X\\_15\\_3\\_002](https://doi.org/10.1088/0954-898X_15_3_002) (visité le 06/07/2021).
- SHRESTHA, Amar et al. (nov. 2017). “A spike-based long short-term memory on a neurosynaptic processor”. In : *2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. ISSN : 1558-2434, p. 631-637. DOI : 10.1109/ICCAD.2017.8203836.
- SHRIKI, Oren, Adam KOHN et Maoz SHAMIR (juin 2012). “Fast Coding of Orientation in Primary Visual Cortex”. en. In : *PLOS Computational Biology* 8.6. Publisher : Public Library of Science, e1002536. ISSN : 1553-7358. DOI : 10.1371/journal.pcbi.1002536. URL : <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1002536> (visité le 08/06/2021).
- SON, Bongki et al. (fév. 2017). “4.1 A 640×480 dynamic vision sensor with a 9µm pixel and 300Meps address-event representation”. In : *2017 IEEE International*



- 
- Solid-State Circuits Conference (ISSCC)*. ISSN : 2376-8606, p. 66-67. DOI : 10.1109/ISSCC.2017.7870263.
- SP730 Near-IR/Colorless Dichroic Block Shortpass Filter* (juin 2015). en-US. URL : <https://midopt.com/filters/sp730/> (visité le 23/11/2020).
- SPERING, Miriam et al. (avr. 2011). “Keep your eyes on the ball : smooth pursuit eye movements enhance prediction of visual motion”. In : *Journal of Neurophysiology* 105.4. Publisher : American Physiological Society, p. 1756-1767. ISSN : 0022-3077. DOI : 10.1152/jn.00344.2010. URL : <https://journals.physiology.org/doi/full/10.1152/jn.00344.2010> (visité le 18/08/2021).
- SPRAGUE, William W. et al. (2015). “Stereopsis is adaptive for the natural environment”. In : *Science Advances* 1.4. Publisher : American Association for the Advancement of Science, e1400254. DOI : 10.1126/sciadv.1400254. URL : <https://www.science.org/doi/10.1126/sciadv.1400254> (visité le 19/10/2021).
- STEFFEN, Lea, Max ELFGEN et al. (mai 2021). “A Benchmark Environment for Neuromorphic Stereo Vision”. In : *Frontiers in Robotics and AI* 8. DOI : 10.3389/frobt.2021.647634.
- STEFFEN, Lea, Daniel REICHARD et al. (2019). “Neuromorphic Stereo Vision : A Survey of Bio-Inspired Sensors and Algorithms”. English. In : *Frontiers in Neurobotics* 13. Publisher : Frontiers. ISSN : 1662-5218. DOI : 10.3389/fnbot.2019.00028. URL : <https://www.frontiersin.org/articles/10.3389/fnbot.2019.00028/full> (visité le 07/07/2021).
- SUDA, Shuya, Yasutoshi MAKINO et Hiroyuki SHINODA (mar. 2019). “Prediction of Volleyball Trajectory Using Skeletal Motions of Setter Player”. In : *Proceedings of the 10th Augmented Human International Conference 2019*. AH2019. New York, NY, USA : Association for Computing Machinery, p. 1-8. ISBN : 978-1-4503-6547-5. DOI : 10.1145/3311823.3311844. URL : <https://doi.org/10.1145/3311823.3311844> (visité le 29/05/2021).
- TAUNYAZOV, Tasbolat et al. (juil. 2020). “Event-Driven Visual-Tactile Sensing and Learning for Robots”. en. In : *Robotics : Science and Systems XVI*. Robotics : Science et Systems Foundation. ISBN : 978-0-9923747-6-1. DOI : 10.15607/RSS.2020.XVI.020. URL : <http://www.roboticsproceedings.org/rss16/p020.pdf> (visité le 24/09/2020).
- TAVANAELI, Amirhossein, Masoud GHODRATI et al. (mar. 2019). “Deep learning in spiking neural networks”. en. In : *Neural Networks* 111, p. 47-63. ISSN : 0893-6080. DOI : 10.1016/j.neunet.2018.12.002. URL : <https://www.sciencedirect.com/science/article/pii/S0893608018303332> (visité le 28/05/2021).
- TAVANAELI, Amirhossein et Anthony MAIDA (fév. 2019). “BP-STDP : Approximating backpropagation using spike timing dependent plasticity”. en. In : *Neurocomputing* 330, p. 39-47. ISSN : 0925-2312. DOI : 10.1016/j.neucom.2018.11.014. URL : <https://www.sciencedirect.com/science/article/pii/S0925231218313420> (visité le 28/05/2021).
- TAVANAELI, Amirhossein et Anthony S. MAIDA (juin 2017). “Bio-Inspired Spiking Convolutional Neural Network using Layer-wise Sparse Coding and STDP Learning”. In : *arXiv :1611.03000 [cs]*. arXiv : 1611.03000. URL : <http://arxiv.org/abs/1611.03000> (visité le 12/04/2021).
- TAYARANI-NAJARAN, Mohammad-Hassan et Michael SCHMUKER (2021). “Event-Based Sensing and Signal Processing in the Visual, Auditory, and Olfactory Do-

- 
- main : A Review”. English. In : *Frontiers in Neural Circuits* 15. Publisher : Frontiers. ISSN : 1662-5110. DOI : 10.3389/fncir.2021.610446. URL : <https://www.frontiersin.org/articles/10.3389/fncir.2021.610446/full> (visité le 01/07/2021).
- TEDALDI, David et al. (2016). “Feature detection and tracking with the dynamic and active-pixel vision sensor (DAVIS)”. In : *2016 Second International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP)*. DOI : 10.1109/EBCCSP.2016.7605086.
- THIELE, Johannes C., Olivier BICHLER et Antoine DUPRET (2018). “Event-Based, Timescale Invariant Unsupervised Online Deep Learning With STDP”. English. In : *Frontiers in Computational Neuroscience* 12. Publisher : Frontiers. ISSN : 1662-5188. DOI : 10.3389/fncom.2018.00046. URL : <https://www.frontiersin.org/articles/10.3389/fncom.2018.00046/full> (visité le 15/03/2021).
- THIER, Peter et Uwe J. ILG (déc. 2005). “The neural basis of smooth-pursuit eye movements”. eng. In : *Current Opinion in Neurobiology* 15.6, p. 645-652. ISSN : 0959-4388. DOI : 10.1016/j.conb.2005.10.013.
- THORPE, Simon, Arnaud DELORME et Rufin VAN RULLEN (juil. 2001). “Spike-based strategies for rapid processing”. en. In : *Neural Networks* 14.6, p. 715-725. ISSN : 0893-6080. DOI : 10.1016/S0893-6080(01)00083-1. URL : <http://www.sciencedirect.com/science/article/pii/S0893608001000831> (visité le 18/11/2020).
- THORPE, Simon, Denis FIZE et Catherine MARLOT (juin 1996). “Speed of processing in the human visual system”. en. In : *Nature* 381.6582. Number : 6582 Publisher : Nature Publishing Group, p. 520-522. ISSN : 1476-4687. DOI : 10.1038/381520a0. URL : <https://www.nature.com/articles/381520a0> (visité le 07/06/2021).
- THORPE, Simon et Jacques GAUTRAIS (déc. 1998). *Rank Order Coding*. Journal Abbreviation : Computational Neuroscience : Trends in Research Pages : 118 Publication Title : Computational Neuroscience : Trends in Research. ISBN : 978-1-4613-7190-8. DOI : 10.1007/978-1-4615-4831-7\_19.
- THORPE, Simon J. et al. (juin 2004). “SpikeNet : real-time visual processing with one spike per neuron”. en. In : *Neurocomputing*. Computational Neuroscience : Trends in Research 2004 58-60, p. 857-864. ISSN : 0925-2312. DOI : 10.1016/j.neucom.2004.01.138. URL : <https://www.sciencedirect.com/science/article/pii/S0925231204001432> (visité le 10/06/2021).
- TRIOLET, Celine et al. (avr. 2013). “Quantifying the nature of anticipation in professional tennis”. In : *Journal of Sports Sciences* 31.8. Publisher : Routledge \_eprint : <https://doi.org/10.1080/02640414.2012.759658>, p. 820-830. ISSN : 0264-0414. DOI : 10.1080/02640414.2012.759658. URL : <https://doi.org/10.1080/02640414.2012.759658> (visité le 09/07/2021).
- TUSTISON, Nicholas J., Brian B. AVANTS et James C. GEE (déc. 2019). “Learning image-based spatial transformations via convolutional neural networks : A review”. en. In : *Magnetic Resonance Imaging*. Artificial Intelligence in MRI 64, p. 142-153. ISSN : 0730-725X. DOI : 10.1016/j.mri.2019.05.037. URL : <https://www.sciencedirect.com/science/article/pii/S0730725X19300037> (visité le 19/07/2021).

- 
- TVERSKY, Tal et Risto MIIKKULAINEN (juin 2002). “Modeling directional selectivity using self-organizing delay-adaptation maps”. In : *Neurocomputing* 44-46, p. 679-684. DOI : 10.1016/S0925-2312(02)00457-5.
- uzh-rpg/event-based\_vision\_resources (juil. 2021). original-date : 2017-06-27T08 :00 :43Z. URL : [https://github.com/uzh-rpg/event-based\\_vision\\_resources](https://github.com/uzh-rpg/event-based_vision_resources) (visité le 05/07/2021).
- VANRULLEN, Rufin (juil. 2008). “The power of the feed-forward sweep”. In : *Advances in Cognitive Psychology* 3.1-2, p. 167-176. ISSN : 1895-1171. DOI : 10.2478/v10053-008-0022-3. URL : <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2864977/> (visité le 07/06/2021).
- VANRULLEN, Rufin, Rudy GUYONNEAU et Simon J. THORPE (jan. 2005). “Spike times make sense”. en. In : *Trends in Neurosciences* 28.1, p. 1-4. ISSN : 0166-2236. DOI : 10.1016/j.tins.2004.10.010. URL : <http://www.sciencedirect.com/science/article/pii/S0166223604003546> (visité le 21/09/2020).
- VIGNERON, Alex et Jean MARTINET (mar. 2020). *A critical survey of STDP in Spiking Neural Networks for Pattern Recognition (Preprint)*. DOI : 10.13140/RG.2.2.24086.09287.
- WACONGNE, Catherine, Jean-Pierre CHANGEUX et Stanislas DEHAENE (mar. 2012). “A neuronal model of predictive coding accounting for the mismatch negativity”. eng. In : *The Journal of Neuroscience : The Official Journal of the Society for Neuroscience* 32.11, p. 3665-3678. ISSN : 1529-2401. DOI : 10.1523/JNEUROSCI.5003-11.2012.
- WELIKY, Michael, William H. BOSKING et David FITZPATRICK (fév. 1996). “A systematic map of direction preference in primary visual cortex”. en. In : *Nature* 379.6567. Bandiera\_abtest : a Cg\_type : Nature Research Journals Number : 6567 Primary\_atype : Research Publisher : Nature Publishing Group, p. 725-728. ISSN : 1476-4687. DOI : 10.1038/379725a0. URL : <https://www.nature.com/articles/379725a0> (visité le 15/09/2021).
- WENISCH, Oliver G., Joachim NOLL et J. Leo van HEMMEN (oct. 2005). “Spontaneously emerging direction selectivity maps in visual cortex through STDP”. en. In : *Biological Cybernetics* 93.4, p. 239-247. ISSN : 1432-0770. DOI : 10.1007/s00422-005-0006-z. URL : <https://doi.org/10.1007/s00422-005-0006-z> (visité le 29/04/2019).
- WILLIAMS, R. W. et K. HERRUP (1988). “The control of neuron number”. eng. In : *Annual Review of Neuroscience* 11, p. 423-453. ISSN : 0147-006X. DOI : 10.1146/annurev.ne.11.030188.002231.
- WLODARCZYK, Agnieszka I. et al. (2013). “Tonic GABAA conductance decreases membrane time constant and increases EPSP-spike precision in hippocampal pyramidal neurons”. eng. In : *Frontiers in Neural Circuits* 7, p. 205. ISSN : 1662-5110. DOI : 10.3389/fncir.2013.00205.
- WOHRER, Adrien et Pierre KORNPORST (avr. 2009). “Virtual Retina : a biological retina model and simulator, with contrast gain control”. eng. In : *Journal of Computational Neuroscience* 26.2, p. 219-249. ISSN : 1573-6873. DOI : 10.1007/s10827-008-0108-4.
- WU, Jibin, Yansong CHUA et Haizhou LI (juil. 2018). “A Biologically Plausible Speech Recognition Framework Based on Spiking Neural Networks”. In : *2018*

- 
- International Joint Conference on Neural Networks (IJCNN)*. ISSN : 2161-4407, p. 1-8. DOI : 10.1109/IJCNN.2018.8489535.
- WU, Y. et al. (mai 2013). “The role of visual perception in action anticipation in basketball athletes”. eng. In : *Neuroscience* 237, p. 29-41. ISSN : 1873-7544. DOI : 10.1016/j.neuroscience.2013.01.048.
- WU, Yujie et al. (2018). “Spatio-Temporal Backpropagation for Training High-Performance Spiking Neural Networks”. English. In : *Frontiers in Neuroscience* 12. Publisher : Frontiers. ISSN : 1662-453X. DOI : 10.3389/fnins.2018.00331. URL : <https://www.frontiersin.org/articles/10.3389/fnins.2018.00331/full> (visité le 28/06/2021).
- WYATTE, Dean, Tim CURRAN et Randall O'REILLY (nov. 2012). “The limits of feedforward vision : recurrent processing promotes robust object recognition when objects are degraded”. eng. In : *Journal of Cognitive Neuroscience* 24.11, p. 2248-2261. ISSN : 1530-8898. DOI : 10.1162/jocn\_a\_00282.
- XING, Yannan, Gaetano DI CATERINA et John SORAGHAN (2020). “A New Spiking Convolutional Recurrent Neural Network (SCRNN) With Applications to Event-Based Hand Gesture Recognition”. In : *Frontiers in Neuroscience* 14, p. 1143. ISSN : 1662-453X. DOI : 10.3389/fnins.2020.590164. URL : <https://www.frontiersin.org/article/10.3389/fnins.2020.590164> (visité le 07/09/2021).
- YANG, Zhijun et al. (mar. 2006). “A neuromorphic depth-from-motion vision model with STDP adaptation”. In : *IEEE Transactions on Neural Networks* 17.2. Conference Name : IEEE Transactions on Neural Networks, p. 482-495. ISSN : 1941-0093. DOI : 10.1109/TNN.2006.871711.
- YOONESSI, Ali et Ahmad YOONESSI (avr. 2011). “Functional Assessment of Magno, Parvo and Konio-Cellular Pathways; Current State and Future Clinical Applications”. In : *Journal of Ophthalmic & Vision Research* 6.2, p. 119-126. ISSN : 2008-2010. URL : <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3306093/> (visité le 24/06/2021).
- ZAMARRENO-RAMOS, C. et al. (fév. 2013). “Multicasting mesh AER : a scalable assembly approach for reconfigurable neuromorphic structured AER systems. Application to ConvNets”. eng. In : *IEEE transactions on biomedical circuits and systems* 7.1, p. 82-102. ISSN : 1940-9990. DOI : 10.1109/TBCAS.2012.2195725.
- ZHANG, Yifeng et al. (mai 2014). “Spin observation and trajectory prediction of a ping-pong ball”. In : *2014 IEEE International Conference on Robotics and Automation (ICRA)*. ISSN : 1050-4729, p. 4108-4114. DOI : 10.1109/ICRA.2014.6907456.
- ZHANG, Zhengtao, De XU et Min TAN (déc. 2010). “Visual Measurement and Prediction of Ball Trajectory for Table Tennis Robot”. In : *IEEE Transactions on Instrumentation and Measurement* 59.12. Conference Name : IEEE Transactions on Instrumentation and Measurement, p. 3195-3205. ISSN : 1557-9662. DOI : 10.1109/TIM.2010.2047128.
- ZHAO, Yu et al. (avr. 2018). “Applying deep bidirectional LSTM and mixture density network for basketball trajectory prediction”. en. In : *Optik* 158, p. 266-272. ISSN : 0030-4026. DOI : 10.1016/j.ijleo.2017.12.038. URL : <https://www.sciencedirect.com/science/article/pii/S0030402617316662> (visité le 01/06/2021).

---

ZHU, Alex Zihao et al. (juil. 2018). “The Multi Vehicle Stereo Event Camera Dataset : An Event Camera Dataset for 3D Perception”. In : *IEEE Robotics and Automation Letters* 3.3. arXiv : 1801.10202, p. 2032-2039. ISSN : 2377-3766, 2377-3774. DOI : 10.1109/LRA.2018.2800793. URL : <http://arxiv.org/abs/1801.10202> (visité le 05/07/2021).

# Chapitre 9

## Annexe

### 9.1 DMA

La caméra NeuroSoc utilise une puce Zynq7020 pour recevoir le flux vidéo de la caméra ainsi que la contrôler, effectuer le filtrage spatio-temporel (section 3.1.1), générer les spikes, puis les transmettre (à un ordinateur par exemple). Le Zynq est lui même divisé en deux parties, une partie circuit logique programmable *Programmable Logic* (PL) et processeur (*Processing System* (PS) (figure 9.1). La partie PL est en réalité un FPGA, qui s'occupe du traitement de la réception du flux vidéo, jusqu'à la génération des spikes. La partie PS, quant à elle, s'occupe de récupérer les spikes générés par la PL, puis à transmettre les spikes générés via une connexion Ethernet.

Afin de faire communiquer les deux différentes parties (PS et PL), nous avons utilisé un module de accès mémoire direct (*Direct Memory Access* (DMA). Le DMA contient une *First In First Out* (FIFO) qui stocke les données en paquets de 16 spikes avant d'accéder à la DDR. Une fois ces paquets reçus, le DMA passe en mode *idle* pour attendre de lecture des données par la PS. Une seconde FIFO fait office de mémoire tampon en entrée de DMA et permet de contenir les spikes et éviter les pertes de données lors du passage en mode *idle* du DMA. Pour cela, une ip est instanciée pour une DDR DRAM (mémoire), la PL écrit les spikes générés dans la DDR via cette ip. La PS se connecte elle aussi à cette même ip, elle récupère ce qui y est écrit puis, transmet les données via une connexion Ethernet (seulement accessible via la PS) en paquet de 16 spikes. Ces données sont ensuite récupérés par un PC pour les enregistrer et/ou les afficher. Ceux-ci sont encodés sous un format de 32 bits :

- Bits 0 à 8 → Position X du Spike
- Bits 9 à 17 → Position Y du Spike
- Bits 18 à 21 → Filtre du Spike
- Bits 22 à 29 → temporalité du Spike (inverse de l'intensité)
- Bit 30 → Signe du Spike (ON/OFF)
- Bit 31 → Flag de nouvelle image.

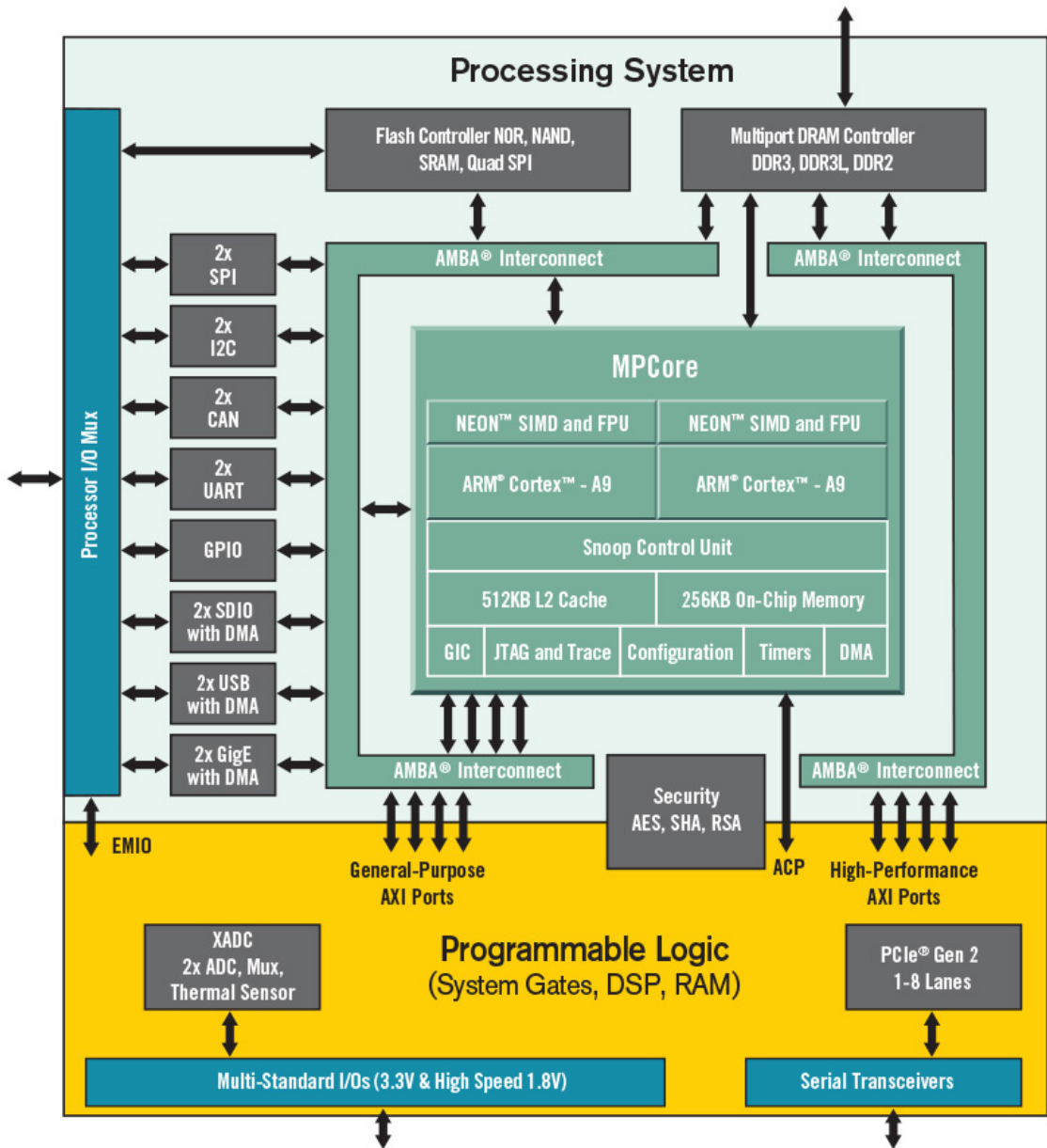


FIGURE 9.1 – Architecture du Zynq7020