

Apprentissage de couplages sensorimoteurs

Dans cette partie, nous allons nous intéresser à différents modèles d'encodage sensorimoteur. Le but est d'explorer successivement différentes stratégies et d'observer le type de résultats et de propriétés que l'on obtient, notamment du point de vue de l'utilisation d'outils. Dans un premier temps (Sec. 2.1), nous allons nous baser sur de précédents travaux du laboratoire (voir [Rolland de Rengerve, 2013]), et explorer la stratégie appelée absolue, dans la mesure où celle-ci propose l'apprentissage du couplage entre les coordonnées dans l'espace sensoriel (la vision) et celles dans l'espace moteur (espace proprioceptif). Posant le constat que l'ajout d'un outil entraîne le ré-apprentissage complet d'un nouveau couplage sensorimoteur, nous proposerons dans un second temps (Sec. 2.2) un couplage permettant d'ajouter facilement une "transformation" (du type rotation ou translation) de ce couplage en nous basant sur le comportement des neurones dit à champ de gains ("gain-field"). Au couplage "absolu" est donc ajouté une transformation "relative", permettant d'obtenir la plasticité d'encodage requise pour l'extension du schéma corporel liée à l'utilisation d'outils (voir Sec. 1.1.1). Mais ce sera alors l'architecture choisie par le modélisateur, et non l'exploration sensorimotrice elle-même, qui déterminera quels apprentissages seront absolus ou relatifs. Dans une dernière partie (Sec. 2.3), nous proposerons donc une première exploration d'un encodage totalement relatif, et nous expérimenterons quelques expériences exploitant les bénéfices d'un tel encodage.

2.1 Modèle "absolu" couplant des positions dans l'espace visuel et moteur

Le premier modèle auquel nous nous sommes intéressés propose l'encodage de neurones de lieu ainsi que le recrutement de transitions entre ces neurones (voir section 1.2.4). Ces neurones de lieu et le graphe de transitions entre ceux-ci peuvent être utilisés au sein d'une carte cognitive, laquelle permet notamment d'agir en vue d'un but, en pondérant les choix d'actions disponibles fournis par le graphe de transitions. Dans sa thèse, De Rengervé [Rolland de Rengerve, 2013] a proposé de porter le modèle des cartes cognitives au contrôle du bras de robot (voir Sec. 1.2.4). En utilisant un apprentissage sensorimoteur basé sur le maintien d'un homéostat, il a ainsi obtenu des comportements de type imitation immédiate, ou différée. En ajoutant l'utilisation d'une carte

cognitive, il a pu effectuer une tâche de tri de canettes dans laquelle, selon le contexte (traduit par la couleur de la canette) le bras katana dépose celles-ci dans l'emplacement correspondant. Dans des travaux communs, nous avons notamment travaillé sur les conséquences d'un signal de renforcement négatif lors d'une telle tâche [De Rengervé et al., 2012].

Dans cette section, nous regarderons dans quelle mesure cet apprentissage visio-moteur est utilisable pour l'utilisation d'outil. Nous commencerons par rappeler le modèle utilisé, et nous expliquerons les expériences qui ont été faites avec ce modèle et les conclusions que nous en avons tirées.

2.1.1 Apprentissage visuo-moteur

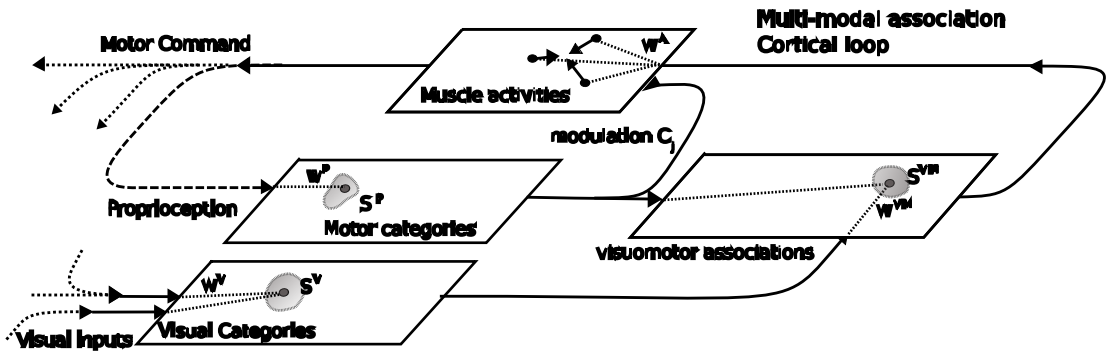


FIGURE 2.1 – Encodage sensorimoteur entre la vision et la proprioception. Un mécanisme de catégorisation (le SAW) encode la vision, la proprioception ainsi que l'association de ces deux catégories. Après apprentissage, lors d'une inadéquation entre ces deux modalités, des commandes motrices sont générées jusqu'à atteindre un état proprioceptif approprié. Issu de [Rolland de Rengerve, 2013].

Avec cette méthode, l'encodage sensorimoteur se fait en trois étapes, représentées sur la figure 2.1. Les entrées du modèle sont la proprioception (les positions articulaires du robot) et la vision (les positions x et y de la main). Chacune de ces modalités est catégorisée à l'aide d'un réseau de neurones, appelé "Selective Adaptive Winner" (SAW). Ce réseau utilise une règle proche de celle des K-moyens, et est couplé à un mécanisme de recrutement. Pour une entrée de dimension N , le calcul de l'activité (eq. 3.3) et de l'apprentissage (eq. 3.4) est donné par les équations suivantes :

$$A_k = 1 - \frac{1}{N} \sum_j |w_{kj} - e_j| \quad (2.1)$$

$$\Delta w_{kj} = \epsilon^{SAW} (e_j - w_{kj}) \quad (2.2)$$

$$\epsilon^{SAW} = \begin{cases} 1 & \text{si } A_k > \textit{vigilance} \\ 0 & \text{sinon} \end{cases} \quad (2.3)$$

où A_k est l'activité du $k^{ième}$ neurone de la couche de sortie du SAW, ϵ le taux d'apprentissage, w_{kj} le poids de la connexion entre le neurone de sortie k et le $j^{ième}$ neurone de la couche d'entrée dont l'activité est e_j . La variable *vigilance* est un seuil au-delà duquel le réseau effectue un recrutement. Au recrutement du neurone i , l'initialisation des poids se fait de la manière

suivante : $\forall j, w_{ij} = e_j$. Notons que le choix du seuil de vigilance détermine la granularité des états dans l'espace d'entrée.

Une troisième catégorie est apprise et associe les deux précédentes, celle motrice et celle visuelle. C'est ici que l'association sensorimotrice est apprise, et c'est l'homéostat qui devra être conservé, en envoyant les commandes motrices compensatoires. Ainsi, lorsque la catégorie motrice est en inadéquation avec la catégorie visuelle, du point de vue de cette association visio-motrice, alors une commande motrice est envoyée pour compenser l'écart constaté entre la catégorie motrice courante, et celle apprise lors de l'association visio-motrice, en se basant sur un modèle du muscle appelé Dynamic-Muscle PerAc [De Rengervé et al., 2015].

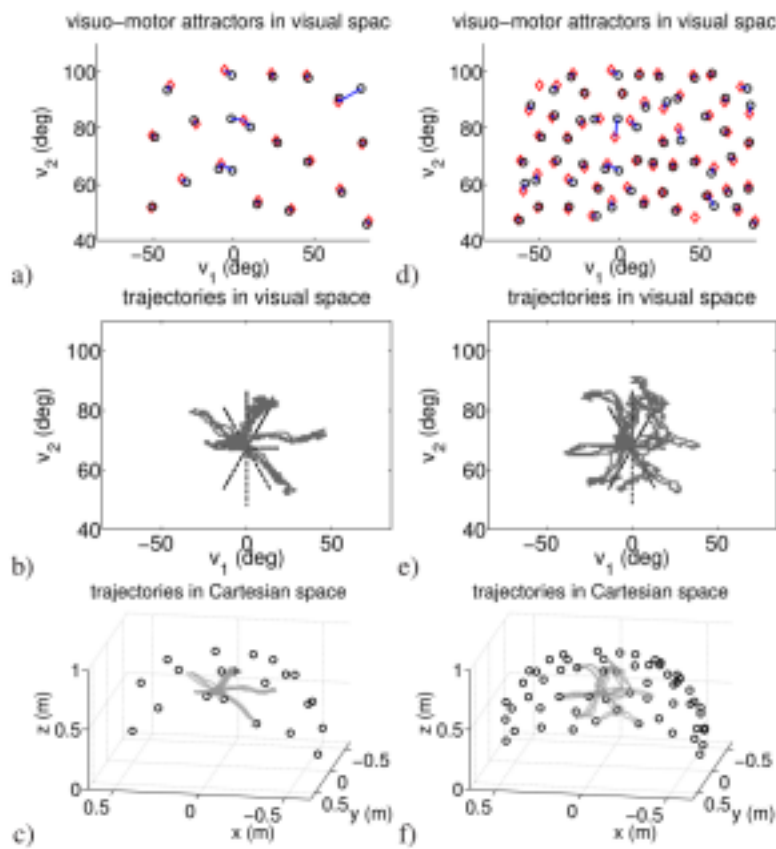


FIGURE 2.2 – Simulation de l'apprentissage en ligne d'attracteurs sensorimoteurs, avec 4 degrés de liberté et une caméra 2D. La colonne gauche représente les résultats obtenus avec une vigilance basse, et la colonne de droite avec une vigilance élevée. a) et d) représentent les états proprioceptifs (cercles noirs) et visuels (diamants rouges) recrutés. Chaque état proprioceptif est associé avec un état visuel (lien bleu). Après l'apprentissage, l'entrée visuelle forme artificiellement une étoile dans l'espace visuel (croix noire), et la reconnaissance visuelle entraîne un mouvement du bras vers l'endroit appris (ligne grise pointillée), ce que l'on observe dans b) et e). Dans c) et f) les mouvements résultants sont montrés dans l'espace cartésien 3D. Issu de [De Rengervé et al., 2015].

Considérant le développement frustré de la vision du bébé, dans ses premiers mois, il est possible d'envisager une ambiguïté de la perception qui conduit l'enfant à percevoir la main d'autrui comme étant la sienne. Une telle inadéquation entre la vision et la proprioception entraîne, de par

le mécanisme que nous avons décrit, lié au maintien d'un homéostat dans l'apprentissage visio-moteur, un comportement dit d'"imitation immédiate" dans lequel la main de l'enfant cherchera alors à rejoindre la main d'autrui observée [Rolland de Rengerve, 2013]. Les résultats de cette architecture, selon le degré de vigilance choisi, sont montrés dans la figure 2.2.

2.1.2 Application à l'utilisation d'outils

Afin de tester les possibilités offertes par ce modèle, nous avons expérimenté un tel encodage visio-moteur en ajoutant un outil au cours de l'expérience. Le but sera ici de conserver les propriétés de ce modèle provenant du maintien d'un homéostat, et les comportements offerts par l'ambiguïté de la perception. Les expériences qui suivent sont donc séparées en deux phases, celle de l'apprentissage visio-moteur, et celle de l'utilisation de l'homéostat issu de cet apprentissage dans des expériences dites d'imitation immédiate.

Nous basant sur les résultats de la psychologie et de la neurobiologie (voir sec. 1.1.1 et [Maravita and Iriki, 2004]), nous considérerons qu'un outil peut-être perçu comme une extension de soi-même. Du point de vue de notre modèle, cela signifie que l'entrée visuelle de l'outil pourra être assimilée à la main (c'est-à-dire à l'effecteur) du robot, que ce soit à la suite d'un apprentissage, ou encore de part une simple ambiguïté de la perception.

Expérience : ambiguïté de la perception avec un outil

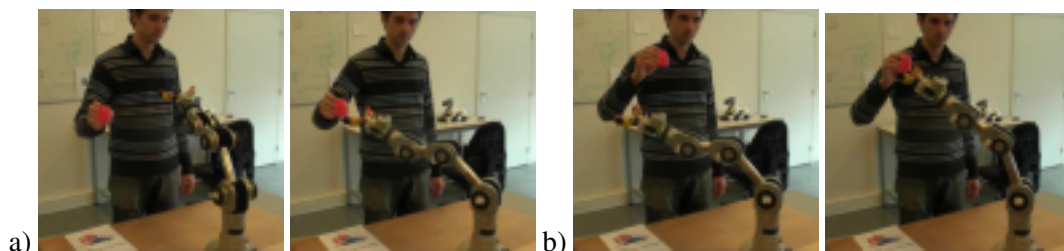


FIGURE 2.3 – Expérience 1, dite "d'imitation immédiate" dans laquelle l'homéostat, maintenu entre la vision de la main (ici la balle rose) et la proprioception associée du robot, est utilisé pour permettre à la main du robot d'atteindre la position supposée de la main du robot, qui se trouve être celle de l'expérimentateur tenant la balle rose. La main du robot suit donc celle de l'expérimentateur. Dans a), on voit dans la première image que la balle est éloignée (à dessein par l'expérimentateur) de la main robotique. Dans la seconde image, la main du robot a rejoint la position de la balle et se stabilise à cette position. b) montre un autre exemple d'une situation similaire.

En ce qui concerne l'ambiguïté de la perception, nous avons, dans une première série d'expériences, testé trois apprentissages différents. Chaque fois, ces apprentissages sont suivis du protocole d'imitation immédiate décrit précédemment (dans lequel l'entrée visuelle est substituée à une autre).

Dans le premier apprentissage visio-moteur (expérience 1), la vision de la main est associée à la proprioception de la main. En utilisant la position visuelle de la main (tenant un objet rose) de l'expérimentateur à la place de celle de la main du robot, l'expérience d'imitation immédiate fonctionne correctement (fig.2.3).

Dans un second apprentissage (expérience 2), nous associons à la proprioception de la main la position visuelle de l'outil (en forme de râteau, voir fig.2.4), lequel est saisi par la main.

Notons que dans cette configuration, la vision et la vigilance sont suffisamment précises pour que l'écart soit significatif.



FIGURE 2.4 – Outil, en forme de râteau, utilisé lors de nos expériences.

Puisque l'outil est dans la main, il y a une cohérence entre la position visuelle de cet outil, et la configuration proprioceptive au cours des déplacements. Ainsi, l'expérience d'imitation immédiate dans laquelle la position visuelle de la main (tenant toujours l'objet rose) remplace la position de l'outil ne pose pas plus de problème et donne des résultats satisfaisants (fig.2.5).



FIGURE 2.5 – Expérience 2 : même expérience que précédemment, mais l'homéostat appris repose cette fois sur la vision de l'outil, perçu comme correspondant à celle de la main (suivant l'hypothèse de l'extension du schéma corporel), associée à la proprioception. La main tenant l'outil suit donc la balle rose de l'expérimentateur selon le même mécanisme que précédemment. a) et b) illustre deux exemples dans lesquels l'outil, tenu par le main robotique, rejoint la position de la balle rose.

Le dernier apprentissage (expérience 3) est identique au premier, la configuration proprioceptive de la main est associée à l'entrée visuelle de la main réelle du robot. Toutefois, pour l'expérience d'imitation immédiate, nous avons donné comme pour la seconde expérience l'entrée visuelle correspondant à la position de l'outil, lequel est encore une fois situé dans la main. Cette expérience vise à reproduire les conditions de l'utilisation d'un outil non encore utilisé, après un premier apprentissage visio-moteur sans outil. Cette fois, le mécanisme précédemment expliqué engendre des commandes motrices qui vont faire se rapprocher la main du robot de ce qu'il perçoit comme étant sa main, à savoir l'outil. Or, puisque l'outil reste dans la main du robot

qui se déplace, celui-ci conserve sa distance initiale à la main : les commandes motrices visant à le rapprocher continuent alors indéfiniment jusqu'à atteindre une butée articulaire (fig.2.6).

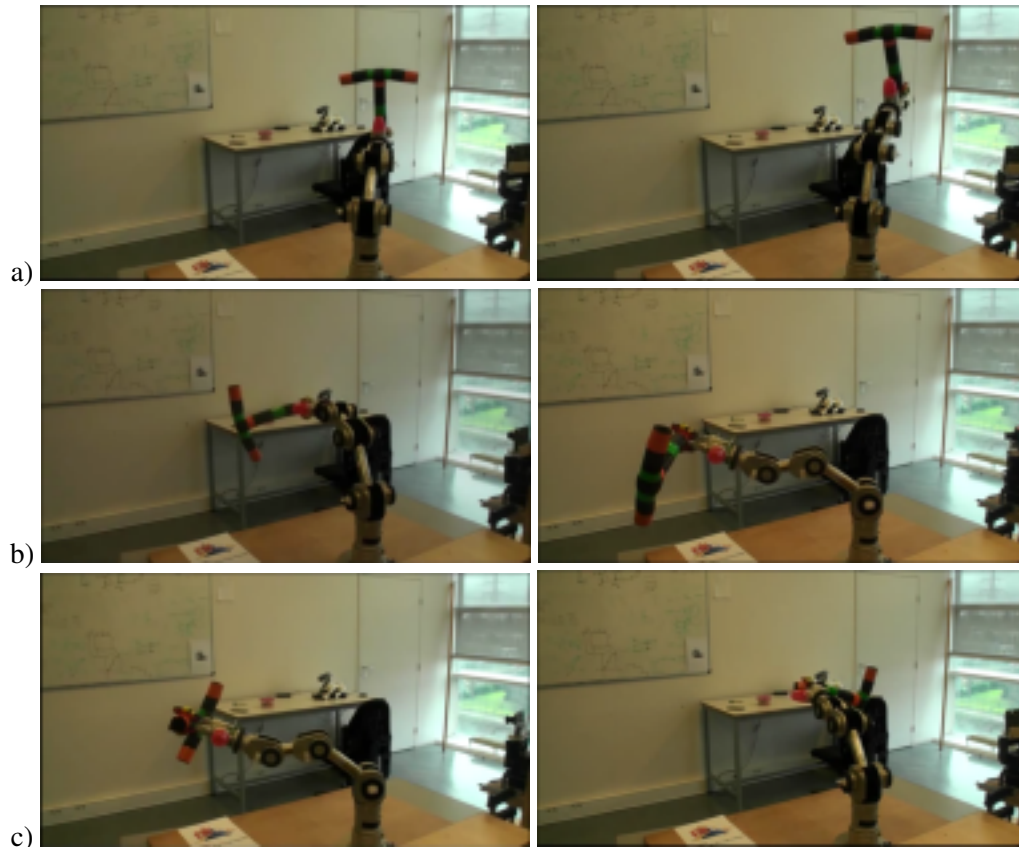


FIGURE 2.6 – Expérience 3 d’imitation immédiate, dans laquelle l’outil est perçu par le robot comme une extension de sa main mais après un apprentissage visio-moteur avec sa main réelle. Pour conserver l’homéostat, la main tente d’atteindre l’outil qu’elle tient, ce qui engendre un comportement de poursuite infinie. En a) l’outil est saisi de telle sorte qu’il est situé au dessus de la main. Le bras du robot se dirige donc vers le haut pour l’atteindre. Dans b), celui-ci est situé sur la gauche et cela engendre un mouvement infini vers la gauche, et en c), l’outil est à droite de la main, ce qui entraîne un mouvement de poursuite sur la droite.

Expérience : changement de la géométrie du bras

Les expériences effectuées ci-dessus ont donc été séparées en deux phases, celle de l’apprentissage visio-moteur, et celle de l’utilisation de l’homéostat issu de cet apprentissage, dans les expériences dites d’imitation immédiate. Dans cette partie, nous avons effectué une autre expérience (expérience 4), en reprenant l’expérience 3 mais ici nous avons laissé l’apprentissage se poursuivre après l’ajout de l’outil, afin d’observer le comportement du modèle.

Nous avons donc effectué un premier apprentissage permettant d’associer la proprioception du bras avec la vision de la main robotique. Nous avons obtenu les résultats de la figure 2.7 a). Les états visuels (représentés sous forme de diamants rouges) sont positionnés sur la figure en

se basant sur l'entrée visuelle de la main (reconnaissance de couleurs), et les états proprioceptifs (représentés sous forme de cercles noirs) sont obtenus en calculant les coordonnées de la main robotique dans l'espace cartésien à partir des valeurs angulaires de chaque articulation, en utilisant le modèle direct du bras.

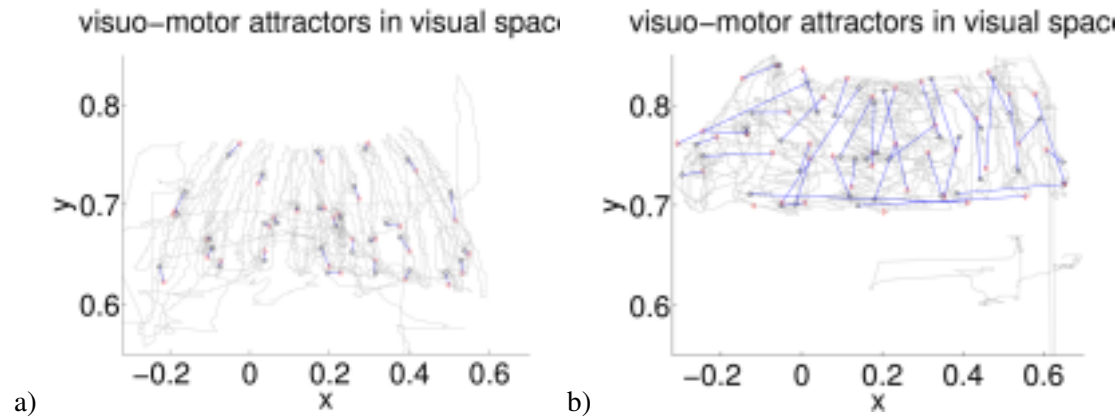


FIGURE 2.7 – Expérience 4 : apprentissage en ligne d’attracteurs sensorimoteurs, avec 4 degrés de liberté et une caméra 2D. Représente les états proprioceptifs (cercles noirs) et visuels (diamants rouges) recrutés. Chaque état proprioceptif est associé à un état visuel (lien bleu).

Les états visuels et proprioceptifs ne se superposent pas totalement à cause du bruit de la reconnaissance visuelle, mais les liens bleus, représentant l’association visio-motrice, sont malgré tout assez petits, ce qui marque la relative qualité de l’encodage du modèle dans ce cas.

Dans un second temps, nous avons ajouté l’outil afin d’observer les résultats obtenus après apprentissage, en permettant au robot de considérer cet outil comme étant sa main. Le robot prend alors les positions de l’outil comme entrée visuelle de son apprentissage visio-moteur. Dans cette expérience, l’outil (toujours le râteau de la fig.2.4) a été positionné vers le haut. La figure 2.7 b) montre ainsi que les états proprioceptifs et visuels ont tous été décalés vers le haut. On note que ce décalage vers le haut est important, d’une grandeur plus importante même que les distances entre états : ainsi, on ne pourrait baisser la valeur de vigilance jusqu’à ce que l’écart engendré par l’ajout de l’outil devienne imperceptible par le SAW, sans en même temps atteindre une granularité bien trop basse, qui effacerait du même coup presque toutes les distinctions entre les états.

Un réapprentissage est donc bien nécessaire, et on note cette fois de longs liens bleus, qui indiquent un décalage important entre la position visuelle de ce qui est perçu comme étant la main (biaisée par la présence de l’outil) et la position (théorique) de la main fournie par les données de la proprioception, en appliquant le modèle direct.

2.1.3 Conclusion

Nos expériences montrent que si ce modèle d’apprentissage visio-moteur, qualifié d’absolu, permet d’encoder efficacement les données sensorimotrices tout en permettant le comportement dit “d’imitation immédiate”, tant lorsque l’apprentissage concerne la main réelle (expérience 1) que l’outil (expérience 2), c’est au moment de l’ajout de l’outil qu’un problème se pose. En effet, il

est alors nécessaire (expérience 3) de réapprendre totalement les associations visio-motrices de la main (expérience 4) pour avoir un apprentissage utilisable.

Notons deux choses lorsque l'on retire l'outil de la main :

- Si le modèle n'apprend pas, alors il y aura un écart entre la position perçue de la main du robot et la proprioception. Et en voulant maintenir l'homéostat de l'apprentissage, la main se déplacera vers l'endroit appris, par exemple vers le bas si l'outil allait vers le haut : on observera un comportement similaire à celui de l'expérience 3 avec un mouvement permanent pour compenser un écart irréductible.
- Si le modèle apprend, alors un second réapprentissage, se ramenant à l'apprentissage initial qui aura été perdu, est à effectuer (expérience 4).

Il semble donc qu'un tel modèle ne soit pas le plus adapté pour une adaptation à la présence d'un outil dans la main, du moins lorsque celui-ci apporte une transformation à la géométrie du robot. Car alors le comportement lié à l'homéostat de l'apprentissage visio-moteur, utile pour atteindre des cibles grâce à l'ambiguïté de la perception, se révèle problématique dans ce cadre. L'apprentissage statique de positions absolues dans l'espace visuel ou proprioceptif, entraîne un réapprentissage total du robot lorsque sa géométrie varie brutalement (comme c'est le cas lors de l'ajout d'un outil).

Notons que, de par la capacité d'adaptation du SAW, un changement lent et progressif de la géométrie du robot (qui serait également à même de simuler, par exemple, la lente croissance de l'enfant), n'aurait pas le même effet. Les états concernés, en étant adaptés à chaque itération, déplaceraient ainsi leurs centroïdes (voir l'équation 3.4) ce qui leur permettraient de s'ajuster continuellement à ces évolutions (si du moins la vitesse d'apprentissage est "plus rapide" que ne l'est la croissance). Toutefois, cela ne correspond pas à de brusques changements de géométrie, comme ceux impliqués par l'utilisation ponctuelle d'outil.

Dans la partie suivante, nous nous sommes donc intéressés à la capacité de certains neurones, dit à champ de gains, à être le support de transformations sensorimotrices. L'idée est alors de voir dans quelle mesure il serait possible d'appliquer une transformation à un apprentissage pré-existant, plutôt que d'avoir à effectuer un ré-apprentissage complet.

2.2 Transformations sensorimotrices et neurones à champ de gains

Ce second modèle a pour but de s'inspirer de travaux de la neurobiologie afin de proposer une solution aux problèmes des transformations sensorimotrices, qui soit implémentable dans un robot.

Dans une première partie, nous verrons les travaux portant sur les neurones dit à champ de gains ("gain-fields" en anglais, noté GF), puis, dans une seconde partie, nous verrons le modèle que nous proposons et ses résultats (que l'on peut retrouver dans [Mahé et al., 2015]). Enfin, nous discuterons de ses avantages et de ses limitations.

Au cours des deux dernières décennies, les études du cortex post-pariétal (PPC) ont permis de mieux comprendre les mécanismes neuronaux impliqués dans la représentation spatiale du corps humain. Notre représentation corporelle est très plastique, et notre cerveau exploite pleinement l'ambiguïté perceptuelle liée à nos sens pour représenter spatialement non seulement nos membres, mais aussi les objets proches, et par extension les personnes autour de nous.

Dans cette perspective, les études d'Irîki, effectuées sur des macaques, sont particulièrement intéressantes car elles indiquent une réadaptation dynamique du schéma corporel lors de différentes situations [Irîki et al., 1996]. En jouant sur le retour visuel d'un téléviseur que le singe observe pour guider ses mouvements de bras, Irîki a montré comment les neurones pariétaux réajustent continuellement l'image corporelle (en l'occurrence celui de la main) de manière à être en adéquation avec de nouveaux cadres de référence [Okanoya et al., 2008]. Ceux-ci peuvent être issus de transformations spatiales complexes et non linéaires telles que la combinaison de translations, de changements d'échelle ou de rotations. Ce résultat a également été testé dans le cadre de l'utilisation d'outils, et les champs réceptifs spatiaux, associés à la main, des neurones pariétaux se sont étendus pour englober l'outil [Maravita and Irîki, 2004; Goldenberg and Irîki, 2007].

Les mécanismes neuronaux impliqués dans les représentations spatiales nécessitent simultanément des transformations non linéaires et un traitement rapide.

Le PPC est idéalement situé pour l'intégration multimodale, puisqu'il est l'une des premières structures corticales à recevoir les signaux sensoriels issus des différentes modalités [Andersen, 1997; Pouget and Snyder, 1997]. Cependant, il n'est pas simple de lier ces signaux, puisque chacun est encodé différemment et correspond à différentes parties du corps et à différents cadres de références. Ainsi, les personnes atteintes de lésions du cortex pariétal peuvent présenter des difficultés d'ajustement spatial, des troubles de la coordination ou encore des négligences spatiales [Keyzers, 2004].

La découverte de cellules associées aux mouvements d'atteintes, ou à des postures correspondant à une orientation particulière de la main, associée au contexte courant ou au plan moteur, a permis de mieux comprendre l'organisation fonctionnelle de ce type de réseau [Blohm and Crawford, 2009; McGuire and Sabes, 2009; Bremner and Andersen, 2012]. Andersen et collègues [Andersen and Mountcastle, 1983; Andersen et al., 1985] ont tout d'abord découvert des neurones, répondant à des commandes motrices spécifiques de saccades oculaires modulées par la position de l'oeil par rapport à la tête. Les résultats montrent que (1) ces neurones sont des neurones bimodaux, car ils codent deux informations à la fois, et (2) leur niveau d'amplitude est informatif, ce qui peut être modélisé. En outre, cet effet de modulation de gain, observé lors de ce type de comportement, ne correspond pas à celui d'une intégration nerveuse comme c'est le cas des neurones Intègre-et-Tire (en anglais "Integrate-and-Fire neurons"). Un modèle

mathématique plus adapté à cette réponse serait une intégration multiplicative entre les signaux sensoriels entrants, ce qui peut être approximée en une fonction de base non linéaire [Pouget and Snyder, 1997].

Dans de précédents travaux, il a été montré comment le mécanisme de modulation de neurones GF peut être, à la manière de certains neurones pariétaux, appliqué à l'intégration de signaux audio-visuels et proprioceptifs dans un robot comprenant une tête, un cou et des yeux [Pitti et al., 2012]. Les neurones GF ont été utilisés pour relocaliser l'emplacement d'un signal sonore (dans un cadre de référence centré sur la tête) en coordonnées de la rétine (dans un cadre de référence centré sur les yeux). Le modèle de GF a permis au système d'améliorer la précision avec laquelle est estimée la position d'un stimulus visuel (en l'occurrence la position de la bouche quand une personne parle), en utilisant le signal sonore qui y est associé.

Dans nos expériences, un robot constitué d'un bras et d'une caméra va tout d'abord effectuer un apprentissage des associations visuomotrices à l'aide de GF. Puis, une fois son schéma corporel appris, un second module de GF est ajouté pour apprendre à accéder à un nouvel "ensemble de tâches" sensorimotrices (en anglais, "task-set") [Pitti et al., 2013a, 2009], lequel doit être appris notamment suite à des transformations liées à des mouvements de la tête (rotations, translations), ou à l'utilisation d'outils.

2.2.1 Modèle

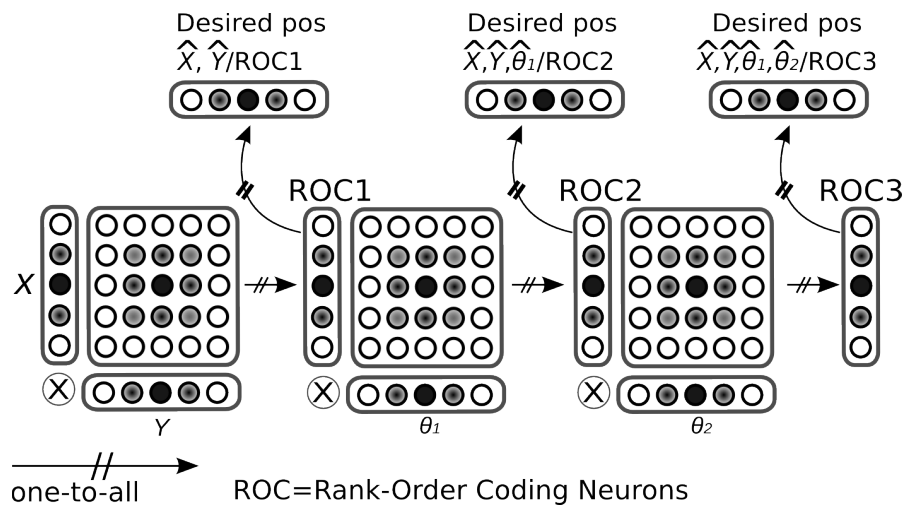


FIGURE 2.8 – Architecture neuronale pour neurones GF. Ceux-ci lient deux activités, et en réutilisant les neurones GF résultant de la multiplication de deux modalités comme l'une des deux modalités d'une autre carte de GF, ils peuvent ainsi lier l'activité de plusieurs entrées. Ces neurones peuvent servir de fonctions de base à partir desquelles les représentations souhaitées peuvent être cartographiées. (ROC) représente l'algorithme de "Rank-Order Coding". Les signaux sensoriels I peuvent alors être estimés à partir de la combinaison linéaire des neurones ROC. La sortie souhaitée \hat{I}/ROC , est alors l'estimation du signal I par ROC.

Notre architecture implémente des neurones multiplicatifs, appelés neurones GF, qui multiplie, unité par unité, la valeur de deux populations neurales entrantes, ou plus voir fig. 2.8. Son fonctionnement est similaire à celui des fonctions de base radiale (RBF), car les GF trans-

forment les signaux d'entrées en un ensemble de fonctions de base, un espace fonctionnel, qui peut également être exploité pour représenter les stimuli dans différents référentiels [Salinas and Thier, 2000; Pouget and Snyder, 1997]. La multiplication entre les signaux sensoriels afférents de deux modalités différentes (M_1, M_2) est la multiplication par éléments de deux distributions de probabilité X_{m_1} et X_{m_2} , deux vecteurs de dimensions $\{\mathbb{M}_1$ et $\mathbb{M}_2\}$ respectivement, avec $\{m_1, m_2 \in \mathbb{M}_1, \mathbb{M}_2\}$. Nous considérerons la matrice dont l'activité $X_m, m \in \mathbb{M}_1 \times \mathbb{M}_2$ est apprise par les neurones GF. L'équation du n -ième neurone GF X_n^{GF} , $n \in N$, dont le poids synaptique $w_i, i \in \mathbb{M}_1, \mathbb{M}_2$ est donnée par :

$$X_m = X_{m_1} \times X_{m_2} \quad (2.4)$$

$$X_n^{GF} = \sum_{i=0}^{\mathbb{M}_1 \times \mathbb{M}_2} w_i X_i \quad (2.5)$$

L'idée ici est que la carte des X^{GF} encode une combinaison de deux valeurs, et l'amplitude du GF encode une modalité de manière conditionnelle, et dans une dimension de plus petite taille [Braun et al., 2009, 2010]. Cette caractéristique est utilisée pour modéliser différents signaux sensoriels en tant que combinaison linéaire f de neurones GF, ainsi $f(X_n^{GF}) = \sum_{n=0}^N \omega_n X_n^{GF}$, avec $n \in N$ neurones GF, et ω la pondération.

Ces neurones apprennent les associations entre diverses modalités de la manière suivante (voir fig. 2.8) : une fois une première association effectuée, disons entre les variables X et Y de la fig. 2.8, il est possible de chaîner les cartes de neurones de telle sorte que l'activité des dernières (par exemple, codant la nouvelle modalité θ_1) dépend de l'activité des premières (en l'occurrence des modalités X et Y). A titre d'exemple, voici les équations pour la deuxième carte X_n^{GF2} , construite à partir de l'activité de la première carte $f(X_n^{GF1})$ et de la nouvelle modalité X_{m_3} , un vecteur de dimension \mathbb{M}_3 :

$$X'_m = X_{m_3} \times f(X_n^{GF1}) \quad (2.6)$$

$$X_n^{GF2} = \sum_{i=0}^{\mathbb{M}_3 \times N} w_i X'_i \quad (2.7)$$

Les nouveaux neurones GF X_n^{GF2} dépendent de X_{m_3} et $f(X_n^{GF1})$ et encodent alors une carte relative à l'activité de toutes les modalités précédentes, X_{M_1} et X_{m_2} . Notons que X^{GF1} représente la multiplication des valeurs des deux modalités d'entrée, et donc ne représente pas de manière unique la combinaison de ces deux valeurs. Cependant, d'un point de vue biologique, l'observation d'un seul neurone GF montre un comportement similaire, au sein des neurones pariétaux [Salinas and Thier, 2000]. Alors qu'une seule carte de GF n'est pas suffisante pour résoudre l'ambiguïté de certains cas, la combinaison linéaire de multiples cartes (X^{GF1} et X_n^{GF2} par exemple), peut aider à la généralisation. Ainsi, les neurones GF représentent chacun un modèle d'activité global spécifique de la carte X^{GF} , qui code une combinaison de paires de valeurs. Ce faisant, le système retranscrit l'information d'une modalité relativement aux autres. Ce processus multimodal est semblable au chaînage de règles conditionnelles bayésiennes entre plusieurs variables [Deneve and Pouget, 2004]. Comme le nombre de dimensions augmente, une combinaison linéaire d'un nombre approprié de neurones GF peut permettre de représenter les données d'entrée "suffisamment bien", si l'espace d'entrée est suffisamment faible. Dans un tel cas, nous ne devrions pas observer de perte de résolution après la phase d'apprentissage.

L'algorithme de "Rank-Order Coding"

Dans la continuité de précédent travaux [Pitti et al., 2012, 2013b], nous avons utilisé une règle proposée par Thorpe et collègues [Van Rullen et al., 1998; Thorpe et al., 2001; Van Rullen and Thorpe, 2002] appelé algorithme "Rank-Order Coding" (ROC).

La sortie des neurones X^{GF} est calculée en multipliant non pas directement les valeurs d'amplitude du vecteur de signaux sensoriels I , mais l'inverse de son rang $rank(I)$ par les poids synaptiques w ; $W \in [0, 1]$.

Pour un vecteur d'entrée de dimension M et pour une population de N neurones GF X , nous remplaçons l'eq. 2.5 by :

$$X_n^{GF} = \sum_{m \in M} \frac{1}{rank(I_m)} w_{n,m}^{GF} \quad (2.8)$$

La règle de mise à jour des poids des neurones est similaire à l'algorithme d'apprentissage "Winner Takes All" des cartes auto-organisées de Kohonen [Kohonen, 1982]. Pour le neurone gagnant $s \in N$ et pour tous les signaux afférents $m \in M$, nous avons :

$$\Delta w_{s,m}^{GF} = \alpha \left(\frac{1}{rank(I_m)} - w_{s,m}^{GF} \right), \quad (2.9)$$

Où α est le taux d'apprentissage. Nous avons choisi $\alpha = 0.01$. Notons que de manière similaire aux RBF, plus le signal d'entrée est distant au neurone, plus son niveau d'activité est bas, ce qui permet de les utiliser comme champs réceptifs.

2.2.2 Apprentissage de transformations sensorimotrices

Une fois que les neurones GF ont appris les règles visuomotrices à partir de leur espace d'entrée, ils peuvent être exploités pour comparer toutes les inadéquations visuomotrices constatées, lors de changements de perception physique ou encore d'ambiguïtés sociales. Le modèle de comparaison, représenté sur la figure 2.9, calcule la différence entre la valeur d'un signal réel A avec une valeur estimée \hat{A} en sortie de notre réseau de GF. Ceci peut alors, par la suite, servir à produire des transformations rendant compte des erreurs constatées.

En ce sens, les neurones GF utilisés dans notre modèle sont proches des fonctions de base radiales (RBF) utilisées dans le traitement d'image dans lesquelles la combinaison linéaire de RBF peut servir à déformer l'espace d'entrée vers un modèle particulier.

Considérons le point A et le nouveau point \hat{A} estimé par le réseau de GF, avec :

$$\begin{aligned} f(A) &= \sum_{i=1}^n w_i X_i^{GF}(A) \\ f(\hat{A}) &= \sum_{i=1}^n w_i X_i^{GF}(\hat{A}) \end{aligned} \quad (2.10)$$

Exprimons $f(\hat{A})$ en fonction de $f(A)$:

$$\begin{aligned} f(\hat{A}) &= \sum_{i=1}^n w_i (X_i^{GF}(A) + (X_i^{GF}(\hat{A}) - X_i^{GF}(A))) \\ f(\hat{A}) &= \sum_{i=1}^n w_i X_i^{GF}(A) + \sum_{i=1}^n w_i (X_i^{GF}(\hat{A}) - X_i^{GF}(A)) \\ f(\hat{A}) &= f(A) + \sum_{i=1}^n w_i (X_i^{GF}(\hat{A}) - X_i^{GF}(A)) \end{aligned} \quad (2.11)$$

Cela signifie que plus \hat{A} sera proche de A , plus $f(\hat{A})$ sera proche de $f(A)$, autrement dit plus un point \hat{A} sera proche d'un point A utilisé pour l'entraînement des neurones GF, plus la transformation associée sera proche de la transformation donnée en exemple.

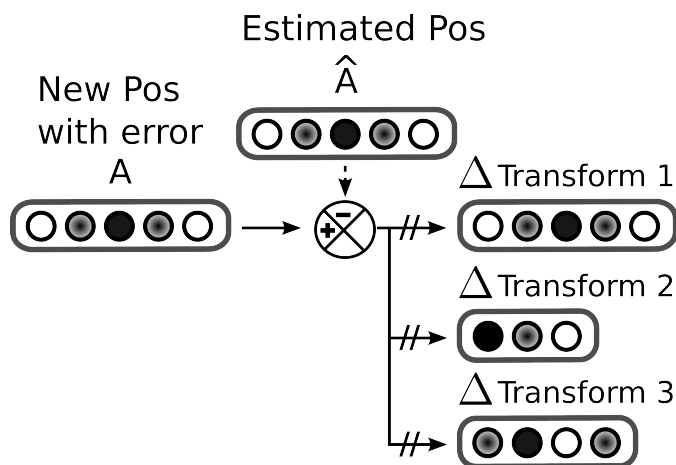


FIGURE 2.9 – Mécanisme de transformation avec signal d’erreur. On peut calculer un signal d’erreur ΔA entre une variable A et son estimation \hat{A} effectuée par le réseau neuronal de la figure 2.8. Cette erreur peut être apprise par différentes cartes en fonction du contexte auquel il correspond. La carte d’erreur peut alors servir à apprendre la transformation permettant de passer du point A au point \hat{A} .

Le second terme correspond à la fonction de transformation qui minimise la distance entre $X_i^{GF}(A)$ et $X_i^{GF}(\hat{A})$. Un perceptron peut estimer cette fonction autour du point A en identifiant les poids qui minimisent le problème de correspondance.

2.2.3 Expériences

Dans nos expériences, nous utilisons le bras robotique Kinova avec 7 degrés de liberté et une caméra fixe pour l’observer, comme le montre la figure 2.10 a).

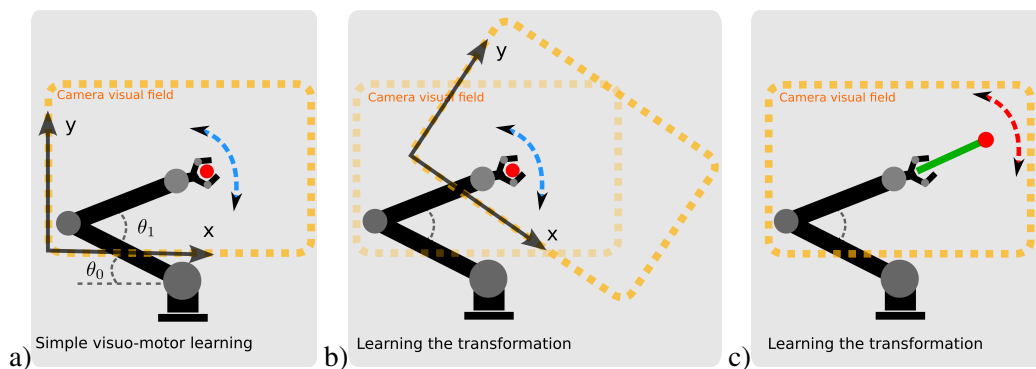


FIGURE 2.10 – Cadre expérimental pour l’apprentissage de transformations sensorimotrices. (a) Cette première situation correspond à l’apprentissage de la correspondance visuomotrice entre les variables motrices (deux degrés de liberté) et la position (X, Y) dans le champ visuel de la caméra. Bien que cette relation soit non linéaire, une légère modification du référentiel ne nécessite pas de réapprendre tous les liens visuomoteurs mais plutôt la fonction de transformation responsable de l’erreur globale. Ces changements peuvent être dus à une transformation visuelle, comme c’est le cas pour une rotation de caméra (b), à un changement corporel comme c’est le cas lors de l’utilisation d’outil (c).

Par souci de simplicité, nous avons limité le bras à deux degrés de liberté (DoF) dans le plan visuel de la caméra, mais l'architecture peut être facilement étendue à plus DoFs en chaînant les cartes GF. Les deux articulations motrices correspondent respectivement aux articulations de l'épaule et du coude, θ_0 et θ_1 , avec $\theta_0 \in [0^\circ, 200^\circ]$ et $\theta_1 \in [0^\circ, 100^\circ]$.

Chaque angle moteur est codé en un vecteur discrétisé de 22 valeurs suivant une gaussienne centrée sur l'angle moteur actuel. Un système de vision basé sur la couleur fournit les coordonnées (x, y) de la couleur rouge, correspondant à la main, dans le cadre de référence de la rétine. De la même manière, avec les angles moteurs, les coordonnées visuelles sont codées en deux vecteurs discrétisés de 22 valeurs suivant une gaussienne centrée sur la position courante. En utilisant l'architecture de GF de la figure 2.8, le système est composé de trois cartes où chaque groupe ROC est composé de 22 neurones. Ainsi, le dernier groupe ROC apprend à associer les angles moteurs (θ_0, θ_1) aux coordonnées de la main selon les axes (x, y) . La sortie du réseau GF donne alors une représentation, basée sur une RBF, des coordonnées apprises de la main.

Pour tirer profit des GF, un perceptron sur la couche de sortie adaptera sa sortie aux coordonnées estimées de la main dans le champ visuel $((\hat{x}, \hat{y}))$ et dans les angles moteurs $((\hat{\theta}_0, \hat{\theta}_1))$. Cela correspond à l'intégration des quatre entrées à travers les trois cartes chaînées (voir fig. 2.8).

Nous utilisons donc cette couche de sortie pour apprendre les transformations produites dans des situations d'ambiguïté perceptuelles, comme le déplacement de la caméra, ou l'utilisation d'outils (voir fig. 2.10 (b) et (c)). Enfin, une dernière couche est ajoutée pour apprendre la différence entre les valeurs réelles et estimées des coordonnées de la main après qu'une transformation ait été appliquée au système (voir eq. 2.11). La sortie de cette couche de neurones fournit alors une transformation qui donne, pour tout point donné, la transformation censée annuler celle réellement appliquée au système.

2.2.4 Résultats

Le robot apprend sa coordination visuomotrice à l'aide des neurones GF. Nous vérifions ensuite comment s'est effectué cet apprentissage, en observant les coordonnées spatiales de la main, tant dans l'espace visuel que dans l'espace peripersonnel.

Tout d'abord, le robot effectue un babillage moteur de façon aléatoire afin que les neurones ROC remplissent uniformément l'espace visuomoteur. Après cette période d'apprentissage, qui a durée 12 minutes et correspond à 12000 échantillons, correspondant à une nouvelle configuration de bras toutes les 0,05 secondes, les neurones ROC s'auto-organisent eux-mêmes et reproduisent le comportement des neurones GF du PPC.

Pour illustrer cette phase, nous avons tracé dans la figure 2.11 le niveau d'activité de trois différents neurones ROC de la troisième carte, par rapport aux paires de variables (θ_0, θ_1) et (x, y) . Le code de couleur indique la valeur d'activité pour chaque neurone et leurs champs réceptifs respectifs dans le domaine moteur et dans le domaine visuel.

Notons que les champs récepteurs visuels et moteurs ne sont pas séparés les uns des autres, et le niveau d'activité des neurones GF est lié aux quatre variables. Dans la figure 2.12 par exemple, le niveau d'amplitude d'un neurone GF varie par rapport à l'angle moteur θ_0 pour une plage donnée de valeurs sur l'axe X. Le code couleur indique trois intervalles différents de l'angle moteur θ_0 . On voit que son champ réceptif n'est pas strictement centré sur une valeur particulière de l'axe X, mais se déplace par rapport au retour proprioceptif, phénomène également observé dans les neurones biologiques du PPC [Salinas and Thier, 2000].

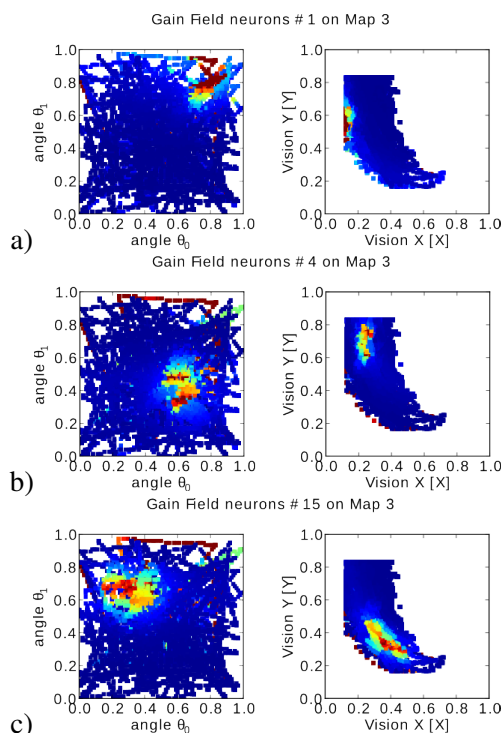


FIGURE 2.11 – Activité des neurones GF. Les neurones 1, 4 et 15 resp. a), b) et c) encodent différents emplacements sensorimoteurs dans (θ_0, θ_1) et (x, y) . Chaque variable est normalisée dans l’intervalle $[0, 1]$.

Pour estimer les propriétés de notre système à généraliser les transformations visuomotrices, nous avons effectués des expériences sur trois transformations visuelles différentes, en s’inspirant des travaux de Iriki, qui a montré sur un macaque une adaptation rapide d’une nouvelle position visuelle de ses membres lors de changements opérés sur un écran [Iriki et al., 1996] ou sur le corps [Maravita and Iriki, 2004; Okanoya et al., 2008]. La première transformation consiste en une légère rotation de la caméra dans le plan du bras robotique. La seconde correspond à une extension de la main du robot avec un outil, et la troisième consiste en une translation de la caméra, en suivant le plan des mouvements du bras (voir fig. 2.10 (b) et (c)).

Dans la figure 2.13, nous présentons les transformations estimées correspondant à trois cas a), b) et c), respectivement celui de la rotation, de l’utilisation d’outils et de la translation.

Chaque neurone du perceptron de la couche de sortie estime les valeurs $(\Delta x, \Delta y)$ apprises à partir des valeurs réelles (x, y) , ainsi que de celles attendues (x', y') provenant des cartes de GF.

Afin de visualiser les transformations qui relient les deux espaces sensorimoteurs, nous avons tracé des flèches liant chaque point (x, y) avec le point après transformation qui y est associé $(x + \Delta x, y + \Delta y)$. Dans les figures 2.13, on voit en bleu la trajectoire de la main avant transformation, et en rouge la trajectoire de la main après transformation. Notons que la longueur des flèches ne correspond pas à l’amplitude réelle de la transformation, mais est réduite pour qu’elles soient mieux visualisées sur la figure. La figure 2.13 a) affiche la transformation correspondant à la rotation de la caméra. Comme on peut le voir, les flèches au centre, là où

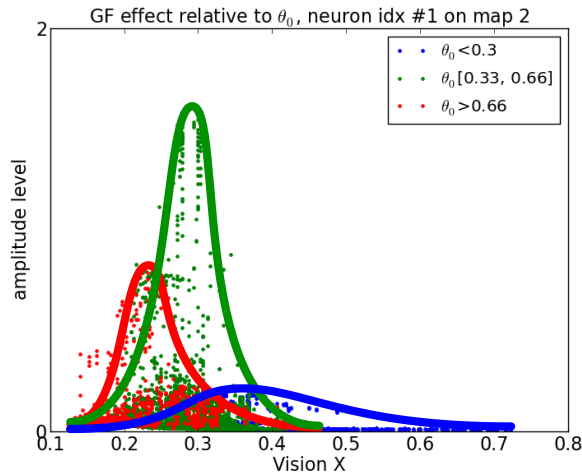


FIGURE 2.12 – Dynamique de l’amplitude d’un neurone GF par rapport à la variable motrice θ_0 sur l’axe X . Notons que les deux variables sont liées ensemble de par leur amplitude. Les centres sur X se déplacent légèrement aussi par rapport à la variable θ_0 comme on le voit également dans les neurones biologiques.

se déplace le robot, approximativement correctement la rotation. Ainsi la généralisation effectuée sur un voisinage proche des exemples appris est relativement bonne. Mais au-delà d’une certaine distance par rapport aux exemples appris (soit, dans notre cas, à l’extérieur de la zone centrale), nous pouvons voir des estimations peu précises, et parfois mauvaises, dans lesquelles l’orientation des flèches ne correspond plus vraiment à la direction de la rotation.

La deuxième transformation de la figure 2.13 b) correspond à l’ajout d’un petit bâton à l’extrémité du bras, qui étend celui-ci vers le haut. Au cours de cette expérience, le système de vision interprète la position de l’extrémité du bâton comme étant celle de la main. Dans ce cas également, la transformation est correctement apprise, et cette fois la généralisation est efficace dans une partie de l’espace des paramètres plus important, ce qui est très certainement dû à la simplicité de cette transformation.

La dernière transformation effectuée est un simple décalage vertical de la caméra (voir fig. 2.13 c). Une fois de plus, la transformation est correctement apprise, et cette fois la généralisation s’applique efficacement à presque tout l’espace des paramètres (ce qui s’explique par le fait que la transformation apprise est véritablement globale).

D’autres résultats, notamment sur le problème dit de la correspondance lors d’interactions interpersonnelles, sont à trouver dans l’article que nous avons écrit sur ce sujet [Mahé et al., 2015].

2.2.5 Conclusion

Dans cette section, nous avons présenté une architecture basée sur les neurones GF susceptible d’encoder l’information sensorimotrice en utilisant des propriétés multiplicatives. Le chaînage de ce type d’opérations multiplicatives, présenté à la figure 2.8, permet d’avoir en sortie des neurones GF encodant bien plus de deux informations sensorielles, en utilisant des multiplications du type $X_m = X_{m_1} \times X_{m_2}$ (voir eq.2.4), permettant de calculer une valeur du neurone

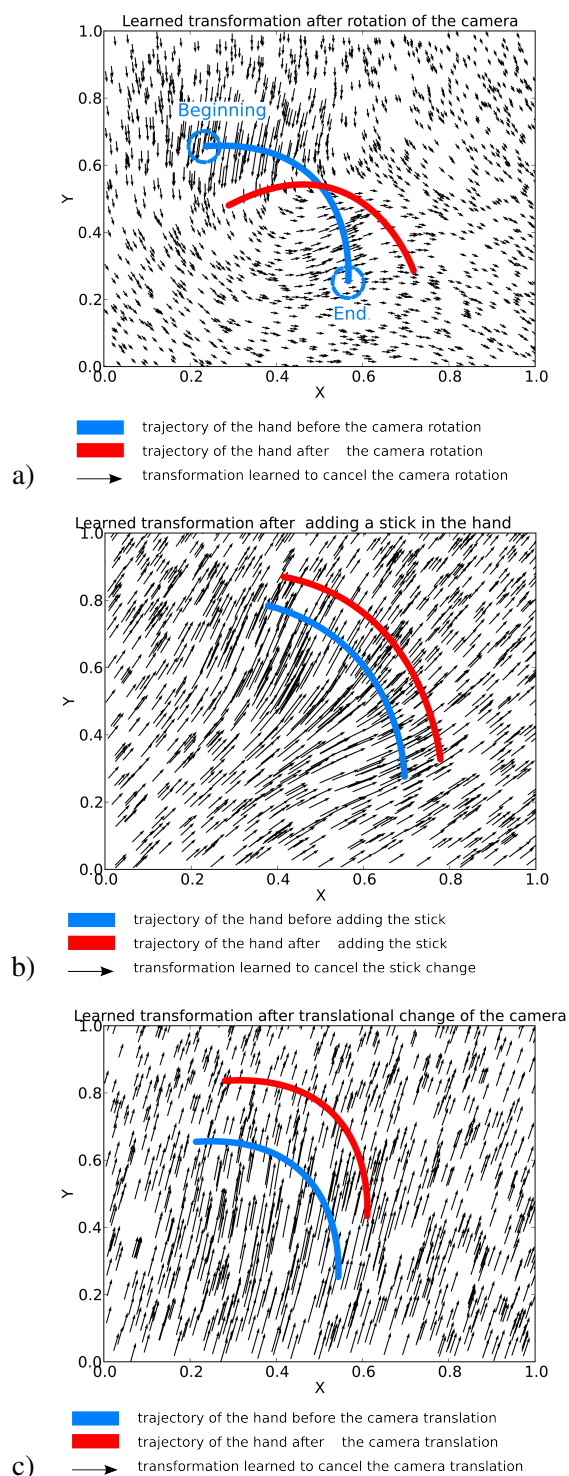


FIGURE 2.13 – Transformations par rotation, utilisation d’outils et translation. Les flèches correspondent aux transformations locales estimées, les lignes bleues correspondent aux trajectoires originales de la main du robot avant transformation, et les lignes rouges aux trajectoires après transformation. (a) Transformation apprise après rotation de la caméra. (b) Transformation apprise après avoir ajouté un bâton à la main robot. (c) Transformation apprise après translation de la caméra.

GF X_n^{GF} , suivie du type $X_m' = X_{m_3} \times f(X_n^{GF})$ (voir eq.2.6), et ainsi de suite. Outre la ressemblance de ce type de neurones avec ceux que l'on retrouve dans le PPC, ceci permet un encodage de la sensorimotricité qui permet de facilement ajouter de nouvelles opérations les unes à la suite des autres, et ainsi de conserver l'apprentissage précédent tout en ajoutant un correctif en cas de constat d'erreur. Notons que l'idée d'ajouter un correctif permettant de ne pas perturber l'apprentissage précédent sera reprise et développée dans le chapitre 4.

Toutefois, le système actuel présente deux limites. La première est que le chaînage est choisi par le modélisateur, et non déterminé par la sensorimotricité du robot. Or l'ordre des opérations a une influence sur le résultat obtenu. De plus, le fait qu'il soit statique empêche d'ajouter facilement de nouvelles modalités, ou de nouvelles transformations à la volée. Si toutes les modalités ne sont pas pertinentes à ajouter a priori pour le modélisateur (comme le tactile dans notre expérience), notons que cette information peut s'avérer utile pour reconnaître un contexte lié à l'usage d'un outil.

Ce qui nous amène à la seconde limitation. Celle-ci concerne l'absence de moyen de lier l'erreur constatée, et potentiellement corrigée, à un contexte défini. En effet, notre architecture permet d'apprendre à corriger une erreur en apprenant une transformation, laquelle peut correspondre à une translation, une rotation, ou encore à l'utilisation d'outil [Mahé et al., 2015]. Si un contexte particulier rend cette transformation nécessaire, il manque un moyen d'attribuer une transformation donnée à un contexte qu'il resterait à déterminer. Ceci permettrait en effet de profiter du fait que le correctif appris ne perturbe pas le précédent apprentissage, et peut donc être dynamiquement activé ou désactivé en fonction de la reconnaissance du contexte courant.

Notons que l'idée que nous venons de présenter est une alternative à une autre méthode qui consiste à déterminer la transformation (de manière plus générale, l'apprentissage pertinent) à partir du signal d'erreur lui-même (voir par exemple la notion de responsabilité dans [Wolpert et al., 2003]). Au lieu de déduire le contexte, nous proposons de l'apprendre et de prédire les transformations associées aux différents contextes appris. Comme nous le verrons dans le chapitre suivant (chapitre 3), l'inversion de cet apprentissage (dans un sens similaire à celui du modèle inverse) permettrait également de rendre de tels contextes "désirables".

De fait, la question centrale à laquelle nous proposerons d'apporter des solutions sera celle de l'apprentissage de covariations : nous venons d'évoquer que la loi sensorimotrice varie quand le contexte varie et cette idée sera développée dans les chapitres suivants. Mais dans la section qui suit nous allons nous intéresser à la simple covariations de senseurs et de moteurs, et aux propriétés émergentes d'un tel encodage sensorimoteur.

Nous venons d'évoquer que la loi sensorimotrice varie quand le contexte varie, et cette idée sera développée dans les chapitres suivants. Par ailleurs notons que, de fait, la question centrale à laquelle nous proposerons d'apporter des solutions à partir d'ici sera celle de l'apprentissage de covariations. Dans la section qui suit nous allons nous intéresser à la simple covariations de senseurs et de moteurs, et aux propriétés émergentes d'un tel encodage sensorimoteur.

2.3 Modèle couplant flot optique et mouvements oscillatoires

Contrairement aux parties précédentes, dans cette partie nous nous sommes intéressés à un modèle apprenant non pas des associations entre des points dans un espace statique, mais apprenant des associations entre mouvements provoqués par les mouvements du robot, et mouvements perçus. Afin d'explorer les possibilités offertes par de telles associations, nous nous sommes intéressés à un modèle couplant entre elles des primitives bas niveaux, visuelles et motrices, basées sur le mouvement. Pour les primitives motrices, nous avons utilisé de simples oscillateurs actionnant certaines synergies motrices. Pour les primitives visuelles, nous avons utilisé une estimation du flot optique perçu. Grâce à l'apprentissage d'association entre les deux, et en ajoutant un modèle permettant la reconnaissance de formes basé sur la reconnaissance des vues locales en mouvement, nous avons, dans une approche développementale, effectué des expériences permettant d'obtenir un comportement d'atteinte (de "reaching") bas niveau.

Dans une première partie, nous allons décrire les primitives visuelles et motrices implémentés pour nos expériences. Nous verrons ensuite comment nous avons proposé d'apprendre ensemble ces primitives, afin d'obtenir un comportement proche de l'imitation immédiate décrite en Sec. 2.1.1. Puis, nous décrirons le système visuel permettant de reconnaître la forme du bras, qui sera apprise durant la phase de babillage moteur. Enfin, nous verrons dans une dernière partie comment tirer profit d'un tel apprentissage relatif couplé à une reconnaissance de formes, pour faire émerger, en jouant de nouveau sur l'ambiguïté de la perception, un comportement de "reaching".

2.3.1 Primitives visuelles et motrices

Primitives motrices basées sur des oscillateurs

De récentes études ont montré que la réponse du cortex moteur durant un comportement de "reaching" contenait une composante oscillatoire, courte mais marquée, même lorsque le mouvement lui-même ne contient pas d'oscillations [Churchland et al., 2012]. En s'inspirant de ces études neurobiologiques, nous avons exploité la notion de configuration rythmique, et proposons d'utiliser un contrôle moteur basé sur des oscillations. Ce choix a également été fait pour éviter d'utiliser un contrôleur moteur doté d'une précision que l'on ne peut attendre dans les premiers stades du développement.

Pour ces expériences, nous avons de nouveau utilisé le bras Katana. Le contrôleur moteur est illustré figure 2.16-A. Chaque articulation du bras Katana est contrôlée par un ensemble d'oscillateurs. Un oscillateur est basé sur un modèle simple, composé de deux neurones, N_1 et N_2 [Revel and Andry, 2009]. La fréquence des oscillateurs dépend de trois paramètres α_1 , α_2 et β :

$$N_1(n+1) = N_1(n) - \beta N_2(n) + \alpha_1 \text{ and } N_2(n+1) = N_1(n) + \beta N_2(n) + \alpha_2 \quad (2.12)$$

Le signal de contrôle de chaque articulation est ainsi obtenu en faisant la somme pondérée des différents oscillateurs :

$$\theta_j(t) = \sum_{i=1}^n w_i^j \cdot O_i(t). \quad (2.13)$$

Avec $O_i(t)$ le signal de sortie de l'oscillateur i , w_i^j le poids représentant la contribution de l'oscillateur i au signal de contrôle $\theta_j(t)$ de l'articulation j .

Primitives visuelles basées sur des neurones sensibles à la direction du mouvement

En neurobiologie, des enregistrements de cellules provenant des aires V1 et MT du cerveau ont montré que des neurones de V1, ainsi que beaucoup de cellules de MT, étaient sensibles à certaines directions privilégiées de mouvement. Ces neurones ont donc une composante sensible à la direction et sélectionnent ainsi une direction particulière (“direction-selective neurons”) [Movshon et al., 1992]. De plus, une petite part des neurones de MT sélectionne des configurations de directions, en répondant à des configurations plus globales de directions (“pattern selective neurons”).

Afin de simuler ces neurones, nous avons utilisé un estimateur de flot optique (une implémentation hiérarchique de l’algorithme de Horn et Schunk [Horn B. K., 1981] basé sur les travaux d’Amiaz *et al.* [Amiaz et al., 2007]). Puis, nous avons défini des neurones sélectionnant les directions de cette manière : l’activation de chacun des neurones A_i est proportionnelle à la distance angulaire entre le stimulus visuel (le flot optique) et la direction associée, pondérée par l’intensité du mouvement :

$$A_i = \exp\left(-\frac{(\beta - \beta_i)^2}{2\tau_1^2}\right) \cdot \left(1 - \exp\left(-\frac{V^2}{2\tau_2^2}\right)\right) \quad (2.14)$$

Avec β la direction du flot optique estimé, β_i la direction privilégiée du neurone i , V l’intensité du mouvement. Enfin, τ_1 et τ_2 sont les coefficients régulant la dynamique d’activation du neurone, respectivement pour la direction du mouvement et pour son intensité. Dans la suite des expériences, τ_1 et τ_2 ont été empiriquement réglés à 20, afin d’optimiser les dynamiques des neurones réagissant à l’intensité de mouvement observé.

D’autres études ont montré que ces neurones étaient réactifs pour une amplitude pouvant varier, autour de la direction privilégiée, d’environ 40 à 60 degrés [Hol and Treue, 2001]. C’est pourquoi nous avons défini 6 différentes classes de ces neurones, réagissant pour les directions privilégiées suivantes : 0° , 60° , 120° , 180° , 240° et 300° . (Notons que comme nous utilisons les coordonnées de l’image, l’axe des Y est orienté vers le sud (90°)). Pour des mouvements de grande intensité, ce type de neurone répondra avec une valeur de 1 si le flot optique est orienté dans la direction privilégiée, et son intensité diminuera graduellement pour des intensités de mouvement moins importantes, ainsi que pour des directions éloignées de celle qui est privilégiée. Enfin, chaque vecteur vitesse de chaque pixel de l’image, lesquels sont estimés par l’algorithme de flot optique, sera par la suite codé en 6 neurones, sensibles aux orientations citées. Pour une image de 640×480 pixels, on obtiendra donc $6 \times 640 \times 480$ neurones sélectionnant les directions.

Nous avons également ajouté des neurones encodant des configurations de directions (“pattern selective neurons”, PSN), en intégrant la réponse des neurones précédents (“direction-selective neurons”). Par simplicité, et afin d’assurer une réponse temps réel de l’interaction, nous en avons défini seulement 6 qui correspondent aux précédentes directions privilégiées (voir la figure 2.16-B). Afin que ceux-ci soient sensibles à la direction i du mouvement, chacun intègre l’activité de chacun des “direction-selective neurons” de la même direction i .

Une illustration de la réponse de ces PSN pour différentes directions de mouvements du bras Katana, et captée par une caméra 640×480 , est donnée figure 2.14.

Par la suite le modèle associera, justement, le codage de ces directions de mouvements aux activités motrices les produisant. Notons que ce type de résultats est en adéquation avec les résultats observés dans [Georgopoulos et al., 1986; Kakei et al., 1999] qui montrent que des

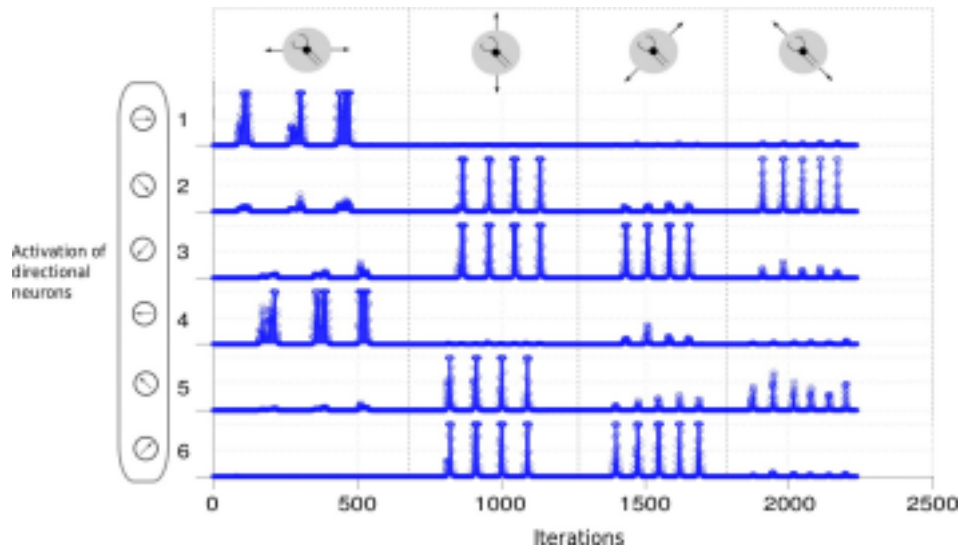


FIGURE 2.14 – Activités des neurones répondant aux directions de mouvements. En haut sont représentés les 4 types de mouvements effectués par le bras Katana face à la camera (oscillation horizontale, verticale, et deux en diagonale). À gauche sont représentées les directions privilégiées des neurones (qui correspondent aux 6 directions énumérées dans le texte).

neurones moteurs peuvent avoir une activité qui va co-varier avec des directions visuelles de mouvements.

2.3.2 Apprentissage visio-moteur et imitation bas niveau

Comme nous l’avons rappelé dans la section 2.1, il a été proposé qu’une imitation bas niveau pouvait émerger des propriétés d’un simple homéostat entre perception et action, en se basant sur une ambiguïté de la perception (voir [Gaussier et al., 1998; Rolland de Rengerve, 2013]. Poursuivant cette idée, nous avons proposé un modèle de réseau de neurones, que l’on peut voir sur la figure 2.16-A, permettant au robot, en apprenant les perceptions de ses propres mouvements, d’obtenir un comportement d’imitation d’un partenaire humain, grâce à l’ambiguïté visuelle.



FIGURE 2.15 – Système expérimental constitué d’un bras Katana dont on utilise 4 DOF, d’une caméra 640x480, et d’une personne interagissante.

Le système expérimental proposé est montré sur la figure 2.15.

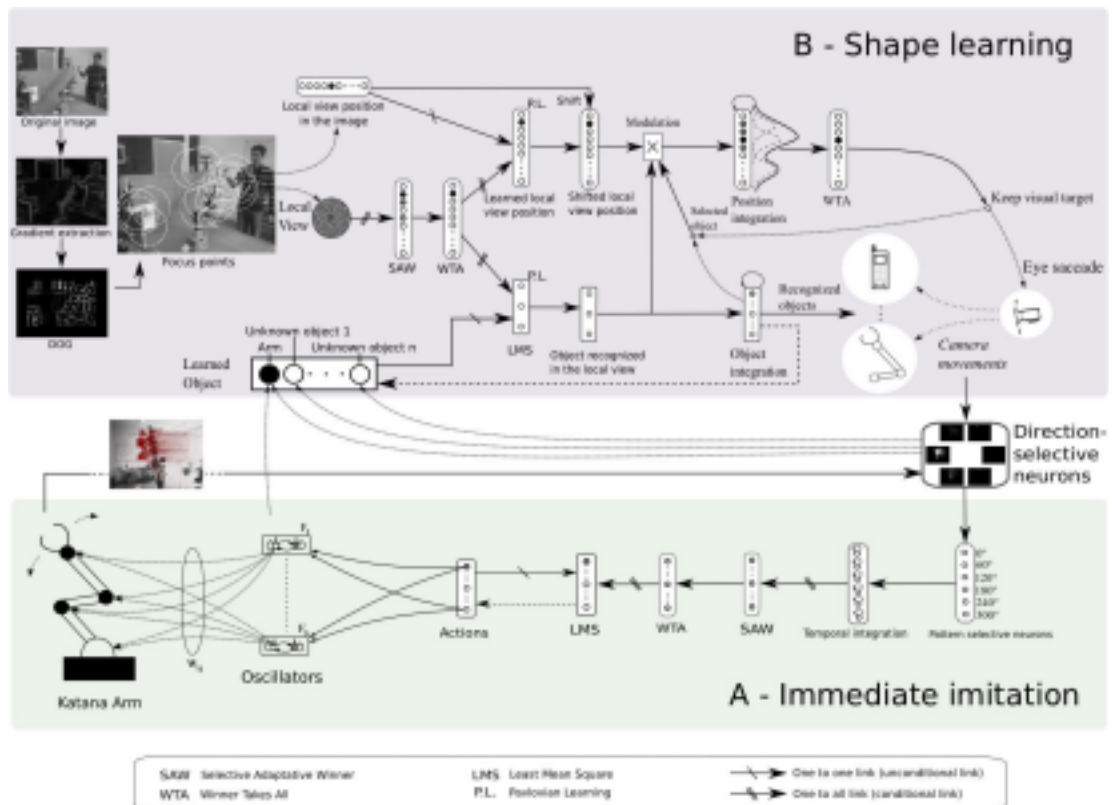


FIGURE 2.16 – Architecture globale. Partie-A : Modèle neuronal pour l'apprentissage du mouvement et l'imitation immédiate. Partie-B : Modèle neuronal pour l'apprentissage et la reconnaissance d'objets, et du bras.

Le contrôleur moteur est composé de 9 oscillateurs de 3 fréquences différentes, avec 3 décalages de phase distincts. Le signal de contrôle de chaque articulation du robot j est obtenu en additionnant les différentes sorties des oscillateurs, et en les modulant par le poids w_i^j (voir l'équation 2.13). Dans un premier essai, les poids w_i^j ont été choisis aléatoirement, le robot bougeant selon cet ensemble de paramètres. Malheureusement, même avec ce petit nombre d'oscillateurs la plupart des mouvements obtenus sont difficiles à analyser et pas vraiment biologiquement plausibles (ce qui est en partie dû aux caractéristiques mécaniques du bras Katana). Ainsi, pour plus de simplicité, nous avons dans l'expérience qui a suivi décidé de choisir les poids w_i^j permettant d'obtenir 3 actions rythmiques différentes : un déplacement horizontal (A1), et deux diagonaux (A2 et A3).

Le modèle, illustré sur la figure 2.16-A fonctionne en deux phases :

- Tout d'abord, durant une phase de babillage moteur simple, le robot apprend sa propre dynamique. Pour cela, il commence par bouger en alternant aléatoirement les trois actions A1, A2 et A3. Le "Selective Adaptive Winner" (SAW, voir section 2.1.1) a en entrée l'intégration temporelle des 6 neurones sensibles aux différentes directions de mouvements (PSN), lesquels réagissent spécifiquement à chacune des actions oscillantes observées. Selon le paramètre de vigilance du SAW, si les entrées sont suffisamment différentes de ce que les neurones encodent précédemment, alors de nouveaux neurones sont recrutés.

À la suite du SAW, un réseau de neurones de type “Winner Takes All” (WTA) est utilisé, et permet de sélectionner les neurones du SAW encodant au mieux les PSN. Pour finir, ces neurones gagnants constituent l’entrée inconditionnelle d’un réseau de neurones utilisant la règle du Least Mean Square (LMS), un perceptron simple utilisant la règle de Widrow-Hoff. Ce réseau apprend l’association entre l’encodage visuel (le flot optique) et les actions motrices représentées par l’ensemble des paramètres w_i^j .

- Après la phase d’apprentissage, lorsqu’un humain commence à bouger dans le champs visuel du robot, ses mouvements seront caractérisés par les PSN. Or, comme nous l’avons dit, le contrôleur du bras peut être vu comme étant un système homéostatique cherchant à maintenir la cohérence entre les actions produites et celles perçues, que celles-ci soient ou non exécutées par le robot lui-même ou par l’humain. Ainsi, si une configuration de mouvements induite par un humain est suffisamment proche de mouvements précédemment appris par le robot, alors le LMS lancera les paramètres d’oscillateurs w_i^j correspondants. Le robot, en lançant une action “proche” en terme de flot optique, semblera ainsi imiter le type de mouvement observé par l’humain.

2.3.3 Apprentissage de la forme du bras pendant le babillage moteur

Afin d’illustrer les possibilités d’un tel système, nous l’avons appliqué à une tâche consistant à atteindre des objets. Pour ce faire, nous allons détailler ici comment, en utilisant le mécanisme des saccades oculaires, nous obtenons une reconnaissance de la main en utilisant l’attention qui est portée sur elle. Pour cela, nous utilisons un modèle neuronal permettant la reconnaissance d’objets inspiré des travaux de [Gaussier et al., 1997] et [Lepretre et al., 2000]. Le principe général de ce modèle est d’apprendre des vues locales d’objets en se basant sur la détection de points d’intérêts (l’apprentissage successif de chacun des points attirant l’attention simule ainsi les saccades oculaires). Comme illustré dans la figure 2.16-B, le gradient spatial d’information est d’abord extrait de l’image de gris. Ce gradient est ensuite convolué à un filtre DOG (différence de gaussiennes). La sortie de ce processus fournit une carte de salience permettant de mettre en évidence les régions de l’image ayant une structure de forme proche de celle d’un coin. Enfin, les maxima locaux sont sélectionnés sur cette carte de salience.

Autour de ces points d’intérêts, des vues locales sont extraites en collectant les pixels entourant ces points (ici sur un rayon de 20 pixels). Celles-ci sont alors encodées en les filtrant par une transformation en log polaire, afin d’être robuste aux changements d’échelle et aux variations par rotations. Ces vues locales filtrées forment alors les entrées d’un autre SAW, lequel recrutera un nouveau neurone chaque fois que les entrées seront suffisamment différentes de celles rencontrées précédemment. Enfin, un réseau de type WTA est utilisé pour sélectionner les vues locales gagnantes.

Le modèle présenté dans la figure 2.16-B peut être divisé en deux parties. La reconnaissance du “what”, identifiant ce qu’est l’objet, et celle du “where”, sa localisation. Un premier algorithme de LMS est utilisé dans la voie “what” afin d’associer à chaque vue locale un objet donné. Le nombre de neurones du LMS correspond ainsi au nombre d’objet appris. Dans la voie “where”, deux LMS, pour les axes x et y , sont utilisés pour associer le centre de l’objet relativement à la vue locale. Comme représenté dans la figure 2.17, après une telle phase d’apprentissage de la voie “where”, chaque vue locale (centrée sur un point d’intérêt) propose sa

propre prédiction de la localisation du centre de l'objet auquel la voie "what" l'associe. Ainsi, le centre d'un objet sera bien positionné si la majeure partie des vues locales associées vote pour la même position. Dans le cas contraire, plusieurs positions du centre de l'objet seront prédites sans qu'une majorité de votes permette d'identifier un gagnant (voir fig.2.17). Puisque ce qui est appris est une position relative, si un objet est appris à une position donnée et détecté à une autre, la sortie du LMS décalera, pour chaque vue locale, la position relative apprise à la position courante, afin de déterminer la position courante de l'objet.

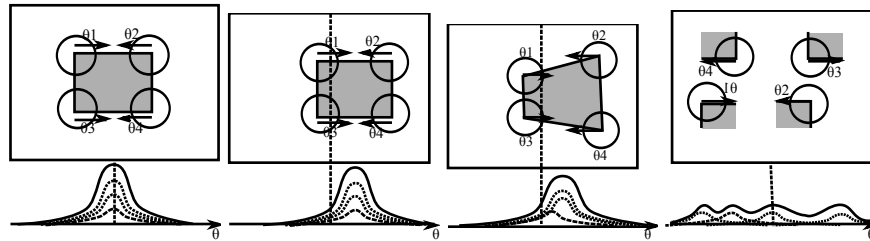


FIGURE 2.17 – Exemple schématique montrant l'estimation de la position d'un objet

Le précédent modèle est alors utilisé pour que le robot apprenne son propre bras, sans ajouter de connaissance *a priori*. Ainsi, durant la phase de babillage, le robot détecte les points d'intérêts comme décrit précédemment, et la carte de salience obtenue après le filtrage par la DOG est modulée par l'intensité du mouvement estimé (le flot optique). Ainsi, si l'on suppose que le robot percevra son bras en mouvement durant la phase de babillage, les caractéristiques détectées sur les points d'intérêts (vues locales) appartiendront, statistiquement, au bras du robot davantage que tout autre objet alentour. Puisque ses propres activités motrices, celles de son bras, génèrent ce flot optique, le robot apprendra "statistiquement" la forme de son bras.

2.3.4 Ambiguïté visuelle et émergence d'un comportement de "reaching"

Après les phases d'apprentissage (de babillage), le robot est capable de reconnaître son bras et de localiser son attention visuelle dessus. Nous avons ensuite, en plus du bras du robot, ajouté la présence d'un autre stimulus visuel susceptible d'attirer l'attention visuelle du robot. Cet objet sera également appris en utilisant le même réseau neuronal permettant l'apprentissage de formes. Il sera agité en face du robot pendant que son bras ne bouge pas, les vues locales correspondront donc statistiquement au nouvel objet.

Comme expliqué précédemment, le modèle utilisé pour la reconnaissance d'objet est aussi capable d'estimer leurs positions. Ainsi, lorsqu'un objet (ou le bras du robot) est détecté et localisé, nous pouvons utiliser cette estimation de position pour bouger la caméra pan-tilt et la centrer sur l'objet. Mais surtout, si deux objets sont reconnus simultanément dans l'image et partagent l'attention visuelle du robot, sa caméra pan-tilt bougera alternativement d'un objet à l'autre, simulant ainsi des saccades oculaires.

En conséquence, lorsque le robot perçoit son bras et centre son attention dessus, et qu'en même temps un humain lui montre un objet appris dans son champ visuel, alors la caméra commence à bouger alternativement du bras à l'objet. Or de tels déplacements, provenant du partage de l'attention visuelle, génèrent un flot optique particulier. En utilisant le modèle neuronal permettant l'imitation de l'action (voir la section 2.3.2 et la figure 2.16), le flot optique obtenu sera

codé par les neurones de directions et les PSN, lesquels, par le LMS, engageront les primitives motrices associées permettant de générer un mouvement oscillant, qui sera de fait dirigé dans la direction où l'objet a été localisé.

Les résultats expérimentaux sont illustrés à la figure 2.18. Après les phases d'apprentissage, le robot est capable de produire, et ainsi d'imiter 3 mouvements différents. Il est également capable de percevoir et de centrer son attention sur son bras ainsi que sur un objet externe (ici, le robot Aibo). La figure 2.18-A montre 3 captures d'images provenant de la caméra pendant l'expérience. Sur ces images apparaissent les vues locales (correspondant aux points d'intérêts), colorées en blanc si elles ne sont associées à aucun objet, en vert si elles sont associées au bras Katana, et en rouge si elles sont associées au robot Aibo. La vue locale "vote" pour un objet donné lorsque l'activité du LMS correspondant dépasse un seuil fixé, et le diamètre du cercle illustre la marge avec laquelle ce seuil est dépassé (les grands diamètres correspondent donc à une reconnaissance plus certaine). La figure 2.18-B montre les activités des neurones du LMS dédié à la reconnaissance des objets (verte pour le bras, et rouge pour l'Aibo). Un seuil décide de si l'objet est détecté ou non, et est représenté par la ligne noire en pointillée. La figure 2.18-C et D montre les axes X et Y de l'activité des champs de neurones (lignes bleues) et les mouvements Pan et Tilt de la caméra (ligne verte en pointillé). La figure 2.18-E montre les activités des 6 PSN. La figure 2.18-F illustre les activités des neurones du LMS dédiés à la reconnaissance des 3 actions (en lignes pointillées verte, rouge et bleu), et les lignes verte, rouge et bleu pleines montrent, quant à elles, les actions effectuées par le robot, lesquelles sont déclenchées lorsque la reconnaissance dépasse un seuil (représenté par une ligne droite en pointillé noir). Enfin, sur la figure 2.18-G et H, nous pouvons voir l'évolution de la position de l'extrémité du bras dans l'espace cartésien, calculée depuis le modèle direct du bras Katana.

Considérons maintenant le scénario proposé qui suit la période d'apprentissage, dans son ensemble. Tout d'abord, le robot ne perçoit que son bras. Celui-ci est reconnu et localisé au centre de l'image. L'objet (Aibo) est ensuite présenté par l'humain, sur le côté supérieur gauche de l'image. En conséquence, le bras et le robot sont simultanément reconnus et localisés (aire marron de la figure 2.18-B). Du fait de cette attention partagée entre son bras et un objet, la caméra se déplace vers l'objet reconnu (figure 2.18-C et D), ce qui induit un flot optique qui est encodé par les PNS (figure 2.18-E). Comme la caméra se déplace vers l'Aibo dans le coin supérieur gauche, celle-ci génère un flot optique inversé, orienté vers le bas et la droite (0° et 60° de part l'inversion de l'axe des Y). Les PNS déclenchent alors le mouvement oscillant du bras correspondant à ce qui a été appris précédemment (soit l'action 2, sur la figure 2.18-F. Le bras du robot, comme le montre la figure 2.18-G et H, se déplace alors vers l'objet détecté, vers la gauche sur l'axe des X, et vers le haut sur l'axe des Y.

2.3.5 Conclusion

Ces expériences sont basées sur l'association entre un flot optique observé et des actions permettant de produire de tels flots. Si les résultats sont convaincants pour les mouvements générant un flot relativement simple, il s'avère bien plus compliqué de catégoriser un flot optique composé de différentes directions qui s'enchaînent. De plus, il y a, du moins en théorie, une infinité de mouvements possibles combinant différentes directions atomiques, et s'il est possible de les approximer à l'aide d'une combinaison pondérée de mouvements complexes appris, ces approxi-

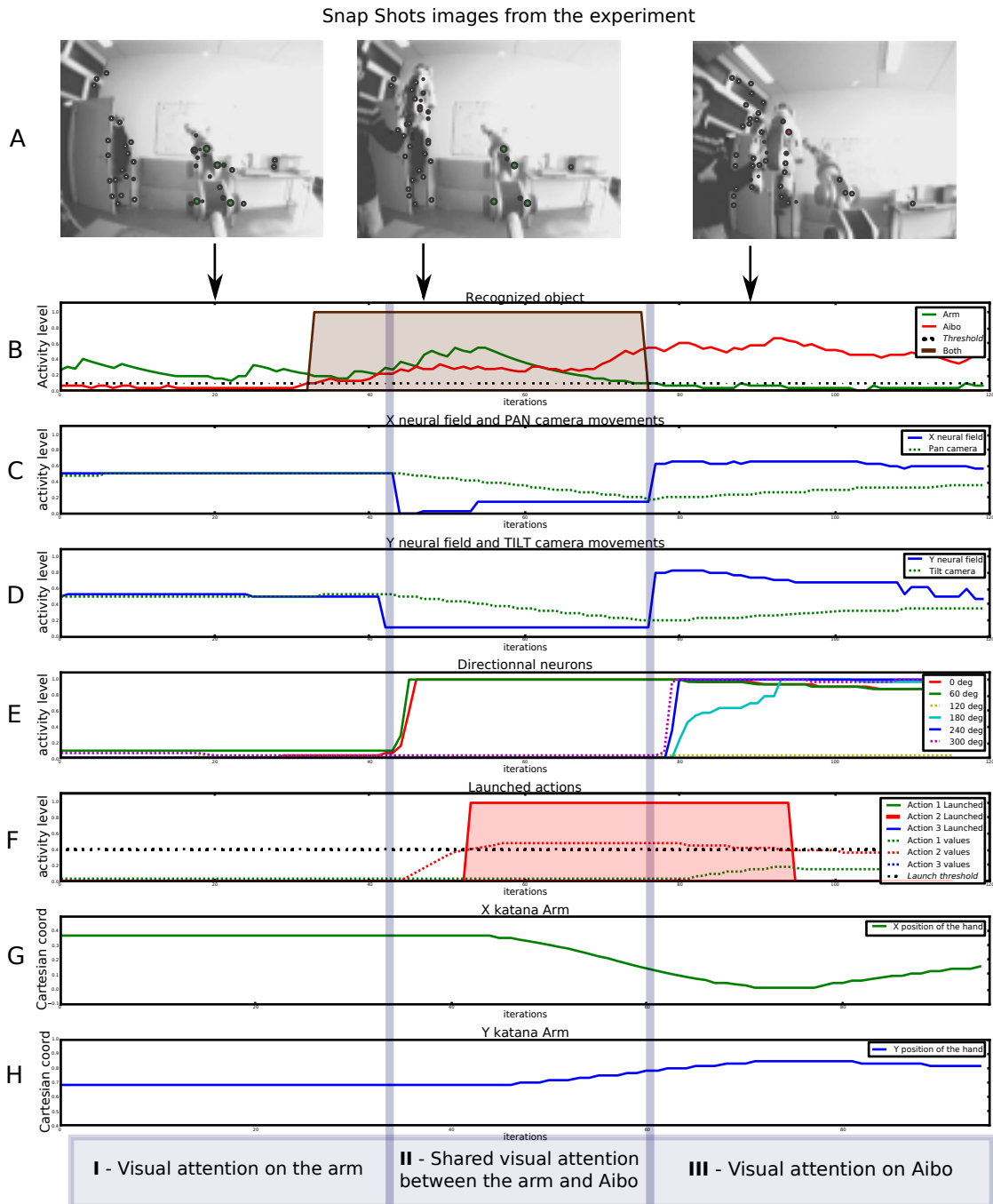


FIGURE 2.18 – Résultats expérimentaux

mations seront d'autant moins bonnes que l'apprentissage d'un seul mouvement complexe est déjà peu précis.

On pourra objecter qu'en pratique, il n'y a probablement pas une infinité de mouvements complexes qui auront à être réellement appris, si l'on considère les contraintes biomécaniques, la minimisation de l'énergie consommée, la richesse comportementale de nos primitives motrices,

ou encore le nombre limité de situations type auxquelles un enfant est confronté de fait (atteindre un objet, le saisir, le pousser, etc.). Cependant, si le nombre de mouvements complexes est de fait probablement restreint, les cas d'applications, et donc de reconnaissances possibles, sont eux effectivement infinis de par leur contexte d'application qui change chaque fois les modalités de leur application.

Ainsi, afin de pouvoir décomposer au mieux les différents mouvements observés, il semble qu'il serait plus efficace d'apprendre chaque direction atomique de manière indépendante (quand bien même une primitive motrice induit plusieurs directions atomiques). Ceci permettrait de percevoir une combinaison de flots optiques variés comme une combinaison de mouvements unitaires. Et non, comme c'est le cas dans nos expériences, de voir cette même combinaison comme un motif complexe, dont la variété de mouvements génère des caractéristiques trop peu discriminables d'un autre mouvement complexe, qui aurait un nombre non négligeable de mouvements unitaires en commun. Une telle décomposition permettrait notamment de reconnaître des combinaisons de mouvements non apprises en tant que telles, du seul fait que chaque mouvement dont cette combinaison particulière est composée serait individuellement connu et donc reconnaissable.

Le principe derrière cette idée est qu'une infinité de mouvements est possible, à partir d'un ensemble fini de mouvements. Et qu'il serait donc plus efficace d'apprendre cet ensemble fini plutôt qu'un sous-ensemble fini de combinaisons (afin éventuellement de les combiner, pour approximer de manière encore moins précise l'infinité de combinaisons possible). Notons que si d'un point de vue théorique cela peut sembler mieux, la biomécanique du corps ou l'existence de primitives motrices limite notre capacité pratique à atomiser ainsi chaque mouvement. Mais quelque soit le stade de développement d'un bébé, ou son degré de maturation, l'idée de chaque fois apprendre, séparément, les effets des commandes motrices sur lesquelles il lui est possible d'agir (quelque soit la complexité réelle de ce qu'une commande engendre), peut s'appliquer, et évoluer au cours du développement.

Notons enfin, comme nous l'avons déjà souligné, que le contexte (comme la position, l'environnement dans lequel l'enfant évolue, les objets avec lesquels il interagit, etc.) perturbe les modalités d'application des mouvements effectués, ainsi que la loi sensorimotrice permettant d'associer un mouvement à un flot optique donné. Ceci apparaît clairement lorsque l'on considère la posture dans laquelle se trouve le robot : un mouvement de l'articulation du coude ne générera pas le même mouvement selon la position dans laquelle se trouve l'épaule.

Suivant la logique que nous venons de décrire, il faudrait donc, en plus d'apprendre les mouvements relatifs à chaque flot optique unitaire, également apprendre de manière contextuelle les lois pseudo-linéaire entre moteurs, et variation de senseurs (flot optique dans le cas courant).

2.4 Conclusion

Dans ce chapitre, nous nous sommes donc intéressés à différentes manières d’encoder l’information sensorimotrice. Nous avons tout d’abord exploré une stratégie absolue en constatant son inefficience lors de l’utilisation d’outils, lié à la nécessité d’un ré-apprentissage complet lorsque le couplage de la main étendue à l’outil et de la vision échoue.

Puis nous avons proposé une architecture permettant à un apprentissage absolu d’être adapté, transformé en fonction de différents contextes parmi lesquels celui de l’utilisation d’outils. Si les résultats sont prometteurs, cette architecture, basée sur un chaînage de réseaux de neurones GF, trouve sa limite dans le fait que la plasticité qu’elle permet est dépendante du chaînage défini, lequel doit de plus être défini a priori, en dehors de toute exploration sensorimotrice spécifique au robot. Les transformations apprises, quant à elle, sont relatives à la chaîne de neurones GF qui la précède, mais aussi à un contexte donné. Or, celui-ci n’est pas appris en tant que tel, il est donc déduit quand son apprentissage pourrait permettre de prédire et donc activer la transformation associée. Dans la partie suivante, nous verrons comment la présence d’un tel apprentissage pourra également permettre au robot de “vouloir” un contexte associé à une transformation désirée (comme celle liée à l’utilisation d’outil).

Partant du principe que le robot découvre son corps par l’exploration sensorimotrice, et que ce corps peut évoluer, que ce soit à l’image du développement du bébé humain ou de part l’utilisation d’outil, nous avons pour objectif un modèle permettant d’apprendre des “transformations”, du type que nous venons de voir, de manière indépendante de tout a priori sur la mécanique du robot. Afin d’obtenir une telle plasticité nous avons, dans la dernière partie, commencé à explorer un modèle totalement relatif dans lequel des mouvements (des activités motrices dotées d’une certaine magnitude, et non des positions absolues dans un référentiel donné, comme par exemple celui de la proprioception) sont associés à une estimation du flot perçu, et donc à une variation dans l’espace sensoriel. Nous avons exploré différentes possibilités offertes par ce modèle, toutefois, les covariations apprises étant basées sur des mouvements oscillants complexes, la covariation à apprendre est le fruit de nombreuses non linéarités et réclame soit des outils d’apprentissages plus efficaces, soit un plus grand découplage des différents mouvements effectués et perçus. Enfin, soulignons que ces apprentissages sont eux aussi dépendants de contextes dans lesquels ils s’appliquent.

Dans le chapitre qui suit, nous allons nous intéresser à un modèle relatif permettant de pallier les différentes limitations que nous avons relevées dans les modèles expérimentés jusque là, tout en exploitant les avantages de ces mêmes modèles. Ce modèle sera relatif, et apprendra de manière indépendante chaque variation de capteur en contextualisant l’apprentissage à la volée.