

Modèle de perception multi-sens pour des agents dans un environnement virtuel

Tristan de Blauwe, Domitile Lourdeaux, Nicolas Sabouret

► **To cite this version:**

Tristan de Blauwe, Domitile Lourdeaux, Nicolas Sabouret. Modèle de perception multi-sens pour des agents dans un environnement virtuel. Workshop sur les Affects, Compagnons artificiels et Interactions (WACAI 2020), CNRS, Université Toulouse Jean Jaurès, Université de Bordeaux, Jun 2020, Saint Pierre d'Oléron, France. hal-02933481

HAL Id: hal-02933481

<https://hal.inria.fr/hal-02933481>

Submitted on 8 Sep 2020

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.

Modèle de perception multi-sens pour des agents dans un environnement virtuel (Version résumée)

Tristan de Blauwe
tristan.de-blauwe@hds.utc.fr
Alliance Sorbonne Université,
Université de technologie de
Compiègne, CNRS, Heudiasyc UMR
7253
Compiègne, France

Domitile Lourdeux
domitile.lourdeux@hds.utc.fr
Alliance Sorbonne Université,
Université de technologie de
Compiègne, CNRS, Heudiasyc UMR
7253
Compiègne, France

Nicolas Sabouret
nicolas.sabouret@limsi.fr
LIMSI CNRS
Université Paris-Sud
Université Paris-Saclay
Orsay, France

RÉSUMÉ

Cet article présente les préliminaires d'un modèle de perception multi-sens, *PRÉSENCE*, qui définit une architecture modulaire et générique pour permettre à des agents évoluant dans un environnement virtuel de percevoir à l'aide de plusieurs sens différentes informations provenant de leur environnement. Ce modèle a été mis en place dans une plate-forme de formation de leaders médicaux, dans le cadre du projet VICTEAMS [7], bien que son utilisation ne se limite pas à ce type d'utilisation.

KEYWORDS

Système multi-agents, Personnage Virtuel Autonome, Perception multi-sens, Erreur, Environnement Virtuel

ACM Reference Format:

Tristan de Blauwe, Domitile Lourdeux, and Nicolas Sabouret. 2020. Modèle de perception multi-sens pour des agents dans un environnement virtuel (Version résumée). In *Proceedings of WACAI (WACAI 2020)*. ACM, New York, NY, USA, 3 pages.

1 INTRODUCTION

Les personnages virtuels autonomes (*abrégié PVA*) sont des agents virtuels capables d'interagir avec un utilisateur mais aussi avec différents éléments de l'environnement virtuel (*abrégié EV*) dans lequel ils sont situés. Ils sont utilisés, entre autres, pour la formation professionnelle [3, 7, 11]. Afin d'interagir pertinemment avec cet environnement, ils doivent pouvoir raisonner sur les connaissances acquises via leurs perceptions. Par exemple, le modèle *smart objects* de Thalmann et al. [2] intègre des informations dans les objets afin de permettre aux agents situés à proximité de les manipuler. Ces informations permettent en particulier de contrôler l'animation de l'agent dans l'EV.

Notre objectif est de reproduire les mécanismes de construction de croyances erronées à partir de perceptions. Nous proposons ici un modèle informatique qui permet :

- **Une perception erronée** : l'état de l'environnement fait que l'information ne peut pas être communiquée correctement au PVA, ce qui conduit à une croyance erronée.
- **L'influence de facteurs internes sur la perception** : l'état cognitif et/ou physiologique simulé du PVA le conduit à mal intégrer l'information qui lui a été communiquée par l'environnement.
- **Et qui soit générique** : le modèle n'est pas dépendant d'un EV en particulier, ce qui nous permet, selon les besoins de l'application, de se connecter avec n'importe quel EV.

Notre but est ainsi que les PVAs exhibent des comportements individualisés et variés selon leur état, la situation et leur environnement. Contrairement aux *smart objects* [2], nous voulons proposer des mécanismes grâce auxquels un PVA puissent percevoir des informations erronées depuis l'environnement. Ces comportements erronés pourraient permettre, par exemple, comme dans le projet VICTEAMS [7], d'entraîner des leaders d'équipes médicales à gérer des équipes variées et à s'adapter à chaque membre de l'équipe.

2 PERCEPTION

Après une étude de la littérature sur les approches biologiques, psychologiques et informatiques sur la perception, nous avons décidé de fonder nos travaux sur la notion de filtrage du modèle d'Herrero [6] pour l'intégration d'erreurs et sur le modèle de perception du framework multi-agents *DIVA* de Steel et al. [12], ainsi que de Kuiper et Wenkstern [8–10]. Ce modèle permet de satisfaire nos différents objectifs, sauf celui de généricité. En effet, *DIVA* n'utilise pas de moteur de jeu mais intègre son propre moteur de visualisation et n'a donc pas vocation à être utilisé avec un moteur de jeu.

Aussi, nous avons mis en évidence, à partir de l'état de l'art, quatre étapes consécutives dans la perception de l'environnement (voir figure 1).

- **ÉTAPE I** : Comment récupérer les informations pertinentes depuis l'environnement ?
- **ÉTAPE II** : Comment les données des signaux sont-elles encodées ?
- **ÉTAPE III** : Comment les signaux sont-ils traités par les capteurs ?
- **ÉTAPE IV** : Comment l'information traitée par les capteurs est-elle perçue par l'entité ?

Nous vous présentons succinctement le modèle dans la section suivante.

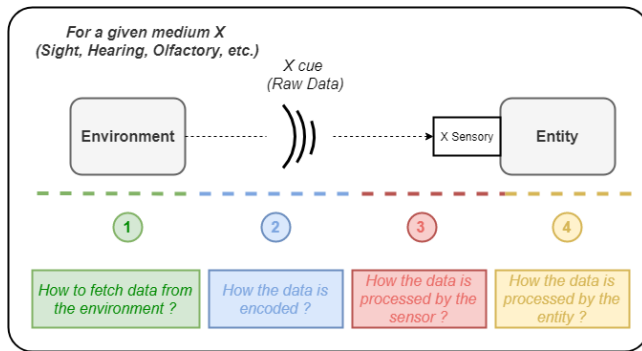


FIGURE 1: Schématisation de la perception et des différentes étapes

3 CONTRIBUTIONS

Bien qu’encore préliminaires, notre modèle de perception multi-sens *PRÉSENCE* définit une architecture informatique modulaire et générique, pour permettre à des PVAs de percevoir leur environnement de manière potentiellement erronée, soit par modification, soit par omission.

Concernant la première étape de perception que nous avons identifiée (voir Figure 1), qui consiste à la récupération des informations pertinentes pour la perception, nous souhaitons tirer avantage des fonctionnalités offertes par les moteurs de jeu actuels (Unity [13], etc.). Ces derniers sont très performants dans cette tâche (par exemple, voir conférence sur les optimisation des calculs physiques accomplis dans l’architecture DOTS de Unity [1]). Nous ne cherchons donc pas à redéfinir ce fonctionnement.

3.1 Approche générale

Notre modèle propose une architecture qui se veut simple d’utilisation et qui permet de :

- (1) Transmettre facilement différents types d’informations depuis l’environnement vers les PVAs concernés, afin notamment de simuler différents sens, tels que la vision, le toucher, etc.
- (2) Mettre en place une structure générique, nous permettant aisément de modifier l’information transmise à deux niveaux :
 - Au niveau du capteur pour simuler par exemple, la difficulté à entendre en fonction du bruit ambiant.
 - Au niveau de l’intégration, pour permettre l’influence de facteurs internes, comme l’état cognitif et/ou physiologique simulé du PVA, ce qui le conduit à mal intégrer l’information qui lui a été communiquée par l’environnement.

Et ce dans le but de leur inculquer des connaissances potentiellement erronées.

3.2 PRÉSENCE

Ce modèle propose de gérer plusieurs médiums¹ qui décrivent des sens, ou alors d’autres types d’informations moins courants

1. Dans le modèle spatial [4], un médium décrit un canal d’informations

comme le sens de l’orientation. En entrée, nous avons les informations propres à chaque médium, sous formes d’indice, qui proviennent de l’environnement virtuel. En sortie, nous avons l’information filtrée à travers différentes étapes, au niveau du capteur et de l’entité.

3.2.1 Vue d’ensemble. Chaque médium possède un système qui gère les interactions entre les entités pour ce médium. Le but du système est d’agir comme interface entre l’environnement virtuel et les différentes entités.

Ensuite, un médium doit définir les propriétés d’un indice (par exemple, pour un indice auditif, il peut s’agir de la fréquence et de la hauteur du son, ou alors la signification d’un évènement, comme une alarme). Un indice consiste à transporter des informations relatives à un médium.

Un indice peut-être émis par une entité sous forme de signal lorsqu’elle subit des modifications qui amènent à un changement relatif au médium (par exemple, une alarme s’active et émet un son). Lorsque c’est le cas, un signal est envoyé au système concerné qui en informe ensuite l’environnement virtuel.

L’environnement peut lui aussi écouter ses signaux afin de se modifier pour refléter les changements (par exemple, si une entité se met à changer de couleur, elle informe le système du changement. Comme l’environnement peut écouter ce signal, il peut le répercuter dans l’environnement virtuel). Un capteur ou sens peut aussi récupérer les informations pertinentes sans avoir à attendre un signal. Dans ce cas-là, le sens demande au système correspondant les informations auxquelles il a accès (ce comportement nous est utile pour la vision par exemple, qui nous permet de récupérer les informations présentes dans son champ de vision, sans avoir à attendre un changement de l’environnement).

3.2.2 Réception d’un indice. Lors de la réception d’un indice (voir section précédente) par le sens, les informations vont être traitées en deux phases distinctes.

La première est à la charge du sens. En fonction des informations spatiales envoyées par l’environnement, il se peut que certaines informations soient atténuées, à cause de la distance par exemple. Ensuite, selon les paramètres du sens, certaines informations peuvent être filtrées. L’implémentation de ces différents traitements dépend du besoin de l’application. Un sens peut, par-exemple, être la vision humaine, ou encore la vision d’un crapaud, etc.

Ensuite, une fois que l’information a été traitée par le capteur, elle est envoyée à l’entité. Dans ce cas-là, le traitement dépendra de l’agent et donc des besoins de l’application. Dans le cas d’un agent humain, il peut-être intéressant de mettre en place un mécanisme de filtrage supplémentaire, notamment à l’aide des mécanismes d’attention [5].

4 CONCLUSION, LIMITES ET PERSPECTIVES

Nous avons mis en place les premières bases d’un modèle de perception erronée multi-sens, en proposant un processus qui prend en entrée les informations provenant d’un environnement virtuel et en sortie les informations retenues par une entité. Ce processus peut-être facilement modifiable au niveau de la perception du capteur et de l’entité afin d’altérer la perception selon les objectifs de l’utilisateur, notamment pour permettre une perception erronée.

5 REMERCIEMENTS

Ce travail a été réalisé dans le cadre du projet LEON financé par la Région Hauts-de-France et le FEDER 2014/2020. Il s’appuie sur des scénarios et un environnement virtuel développé dans le cadre du projet VICTEAMS (ANR-14-CE24-0027) financé par l’ANR et la DGA.

RÉFÉRENCES

- [1] [n.d.]. Leveraging DOTS-powered physics - Unity at GDC 2019. <https://www.youtube.com/watch?v=yuqM-Z-NauU>
- [2] Tolga Abaci, Ján Ciger, and Daniel Thalmann. 2005. Planning with Smart Objects. In *WSCG*.
- [3] Camille Barot, Domitile Lourdeaux, Jean-Marie Burkhardt, Kahina Amokrane, and D. Lenne. 2013. V3S : A virtual environment for risk-management training based on human-activity models. *Presence : Teleoperators and Virtual Environments* 22, 1 (Jan. 2013), 1–19. https://doi.org/10.1162/PRES_a_00134 tex.hal_id : hal-00857947 tex.hal_local_reference : 5+ tex.hal_version : v1 tex.publisher : Massachusetts Institute of Technology Press (MIT Press).
- [4] Steve Benford and Lennart Fahlén. 1993. A Spatial Model of Interaction in Large Virtual Environments. In *Proceedings of the Third European Conference on Computer-Supported Cooperative Work 13–17 September 1993, Milan, Italy ECSCW ’93*, Giorgio de Michelis, Carla Simone, and Kjeld Schmidt (Eds.). Springer Netherlands, Dordrecht, 109–124. https://doi.org/10.1007/978-94-011-2094-4_8
- [5] Jonathan M. Fawcett, Evan F. Risko, and Alan Kingstone (Eds.). 2015. *The handbook of attention*. MIT Press, Cambridge, MA, US.
- [6] Pilar Herrero. 2003. Introducing Human-like Hearing Perception in Intelligent Virtual Agents. (2003), 8.
- [7] Lauriane Huguet, Domitile Lourdeaux, and Nicolas Sabouret. 2016. Présentation du projet VICTEAMS. In *Workshop Affect & Compagnon Artificiel & Interaction (WACAI 2016)*. Brest, France. <https://hal.archives-ouvertes.fr/hal-01342677>
- [8] Dane Kuiper and Rym Z. Wenkstern. 2011. Virtual Agent Perception in Large Scale Multi-agent Based Simulation Systems. In *The 10th International Conference on Autonomous Agents and Multiagent Systems - Volume 3 (AAMAS ’11)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1235–1236. <http://dl.acm.org/citation.cfm?id=2034396.2034503>
- [9] Dane Kuiper and Rym Z. Wenkstern. 2011. Virtual Agent Perception in Large Scale Multi-Agent Based Simulation Systems (Extended Abstract). (2011), 2.
- [10] Dane M. Kuiper and Rym Z. Wenkstern. 2013. Virtual Agent Perception Combination in Multi Agent Based Systems. In *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems (AAMAS ’13)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 611–618. <http://dl.acm.org/citation.cfm?id=2484920.2485017>
- [11] Barry G. Silverman, Michael Johns, Jason Cornwell, and Kevin O’Brien. 2006. Human Behavior Models for Agents in Simulators and Games : Part I : Enabling Science with PMFserv. *Presence : Teleoperators and Virtual Environments* 15, 2 (2006), 139–162. <https://doi.org/10.1162/pres.2006.15.2.139>
- [12] T. Steel, D. Kuiper, and R. Z. Wenkstern. 2010. Virtual Agent Perception in Multi-agent Based Simulation Systems. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, Vol. 2. 453–456. <https://doi.org/10.1109/WI-IAT.2010.220>
- [13] Unity Technologies. [n.d.]. Plateforme de développement en temps réel Unity | Visualisations en 3D, 2D, RV et RA. <https://unity.com/fr>