



HAL
open science

Une lampe interactive ? Automatisation d'un comportement autonome interactif

Yann Boniface, Nicolas P. Rougier

► **To cite this version:**

Yann Boniface, Nicolas P. Rougier. Une lampe interactive ? Automatisation d'un comportement autonome interactif. Drôles d'objets: un nouvel art de faire, Oct 2021, La Rochelle, France. 10.5281/zenodo.6059093 . hal-03597281

HAL Id: hal-03597281

<https://hal.univ-lorraine.fr/hal-03597281v1>

Submitted on 4 Mar 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.

Une lampe interactive ?

Automatisation d'un comportement autonome interactif

Yann Boniface¹, Nicolas Rougier²

1. Loria, Université de Lorraine, 2. IMN, Inria, Université de Bordeaux

Résumé Nous proposons de présenter dans cet article l'utilisation d'une carte auto-organisatrice dans le processus de décision de l'action d'un objet robotisé muni d'un capteur, ici une caméra, interagissant avec un ou plusieurs humains. Cette carte utilise le capteur pour s'adapter à ses interlocuteurs, ici les traits de leur visage, et ainsi ajuster ses mouvements au comportement des humains, sans toutefois les interpréter.

Introduction

Le groupe [Psyphine](#) (Psyphine, 2011), groupe pluridisciplinaire né en 2011, rassemble des chercheurs issus de l'université de Lorraine en intelligence artificielle, philosophie, psychologie, neurosciences, sociolinguistique et anthropologie. Il s'interroge sur les interactions homme/robot et plus particulièrement sur l'attribution ou non d'intentions, d'intelligence voire de conscience à un objet robotisé non humanoïde. Pour mener ses expériences, il dispose d'une lampe (Figure 1) robotisée et munie d'une caméra, au-dessus de son ampoule, qui est utilisée en salle d'expérience comme dans des lieux publics (médiathèque, marché alimentaire, etc.). Ces expériences mettent en relation cet objet avec un ou plusieurs humains pour étudier leurs réactions et tenter de les interpréter (André, Boniface, 2018). L'une des difficultés de nos expériences passées réside dans les biais dus aux manipulateurs, qui contrôlent tout ou partie du comportement de la lampe. Nous avons donc développé un algorithme permettant à cette lampe d'acquérir une forme d'autonomie, c'est-à-dire de bouger sans contrôle ni intervention de notre part, tout en adaptant son comportement à celui de ses interlocuteurs. C'est ce comportement, qui conjugue suivi de visages et mouvements déclenchés par les attitudes des sujets, que nous proposons de présenter. Ces attitudes sont apprises au cours de la passation et donc spécifiques à chaque expérience.

1. Vers des expériences autonomes et qualitatives

Pour étudier ces éventuelles attributions d'intentions à un objet robotisé, le groupe Psyphine a effectué différentes expériences au cours des années que nous décrivons succinctement dans cette section, pour une description plus précise voir (André, Boniface, 2018).

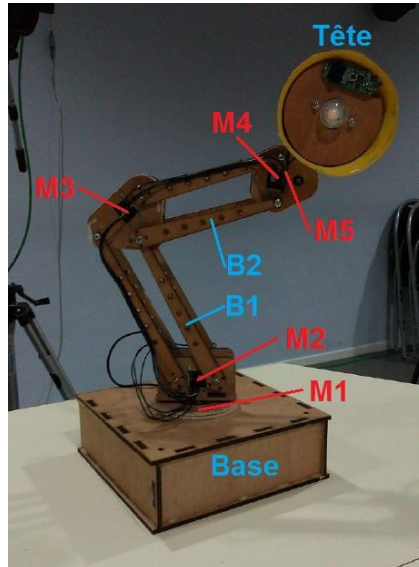


Figure 1: La lampe robotisée du groupe Psyphine. Elle dispose de 5 moteurs (M1 à M5) et d'une caméra dans son abat-jour.

1.1. Expériences dirigées et analyses quantitatives

Passation. Pour nos premières expériences, nous avons créé des comportements pour notre lampe qui sont des séries de commandes motrices prédéterminées et décrivant un mouvement complet de l'objet afin de tenter d'exprimer une émotion (surprise, joie, ennui, peur, etc.). Nous placions, en laboratoire, un sujet devant la lampe (Figure 2) et un manipulateur, caché au sujet et qui voyait la passation à travers une caméra, déclenchait des comportements en fonction des attitudes du sujet. Une passation alternative permettant une comparaison des interactions proposait une série déterministe de comportements s'exécutant donc sans tenir compte des réactions du sujet.

Évaluation quantitative. Nous faisons ensuite remplir un questionnaire au sujet pour comparer les résultats entre les expériences avec séquence déterministe et celles avec un opérateur.



Figure 2: Un exemple de dispositif d'expérience en laboratoire. Le sujet est placé devant la lampe et est filmé par une caméra.

Bilan.

Ces expériences se sont avérées non satisfaisantes pour plusieurs raisons :

- Les comportements construits n'étaient pas interprétés de la même façon par les différents sujets, et jamais, ou très rarement, comme ils avaient été imaginés lors de leurs conceptions. Une expérience dédiée à cette question, *Qualcom* (Becker *et al.*, 2019) le confirme.
- Les expériences réalisées en laboratoire influent fortement sur l'attitude des sujets en raison du caractère trop solennel des lieux, de l'aspect intimidant du dispositif (caméras, projecteurs, écrans, etc.) et de la connaissance même qu'il s'agit d'une expérience contrôlée, mesurée et enregistrée.
- Les expériences sont fortement biaisées par le manipulateur, son interprétation de l'attitude du sujet, son interprétation des séquences de mouvements et certains dérapages de jeux avec le sujet (qu'il faudrait un jour étudier, mais ce n'est pas notre propos ici). Le tout variant en fonction des manipulateurs.
- Les questionnaires quantitatifs se sont avérés peu informatifs, en raison principalement de l'interprétation des questionnaires et de son vocabulaire par les sujets, des interprétations des barèmes qualitatifs et de l'orientation des réponses induites par la formulation des questionnaires.

1.2. Expériences semi-dirigées et analyses qualitatives

Pour corriger ces limitations, nous nous sommes orientés vers une nouvelle série d'expériences, appelée *Decide*.



Figure 3: Les expériences en milieu ouvert (marché alimentaire) consistaient à demander à deux personnes de discuter en présence de la lampe afin de déterminer si celle-ci est autonome ou non, sachant que cette dernière est contrôlée à distance par un opérateur caché.

Passation

Les passations se font, comme on peut le voir dans la figure 3, dans des lieux publics (marché alimentaire, médiathèque, etc.). Les sujets sont placés par deux (s'ajoutent des participants spontanés présents sur le lieu et curieux de l'expérience en cours) devant

l'objet et doivent se mettre d'accord sur une question qui leur est posée au sujet de la lampe (sur son autonomie, son rôle fonctionnel par exemple). L'expérience s'arrête quand ils s'accordent.

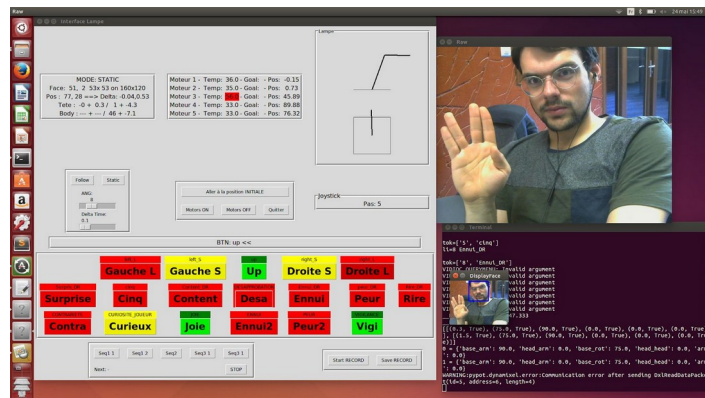


Figure 4: l'interface de contrôle de l'expérience telle que vue par l'opérateur, à gauche l'interface de contrôle de la lampe. Il est possible de la manipuler en mode manuel (au joystick), d'utiliser la fonctionnalité de suivi de visages ou de déclencher, par les boutons de couleur, ce que nous appelons un comportement (joie, peur, ennui, etc.), c'est-à-dire une série de mouvements pré-construits. A noter que la seule entrée vidéo est celle de la caméra de la lampe et il n'y a donc pas de vue globale pour l'opérateur.

Le comportement de la lampe est contrôlé par un opérateur qui dispose de différentes fonctionnalités (Figure 4). Pour permettre une forme d'autonomie de la lampe, et s'inspirant du projet Pinokio (Adam, Zhou, 2012) nous l'avons dotée d'un comportement de suivi de visages. Il s'agit de détecter et d'isoler un visage dans le champ visuel de sa caméra (située dans l'abat-jour) et d'utiliser ses moteurs afin de centrer ce visage dans l'image. Ce comportement est toutefois insuffisant, notre objet ne bouge pas en l'absence de visage et le perd fréquemment en cours d'expérience. Pour retrouver les sujets et relancer le suivi, l'opérateur, qui ne voit l'expérience que par la caméra de l'objet, dispose des comportements définis précédemment et d'un contrôle au joystick.

Évaluation qualitative

L'évaluation de nos passations se fait en deux étapes.

- Les passations sont filmées et sont ensuite analysées par nos soins. Nous cherchons à comprendre les perceptions et interrogations des participants par l'analyse de leur dialogue.
- Les sujets sont interrogés par l'un d'entre nous pour décrire leur sensations et interprétations. Ces interviews sont enregistrées et ensuite analysées par nos soins.

Bilan

Ces séries d'expériences se sont avérées plus satisfaisantes, notamment grâce aux lieux ouverts, au dialogue entre les participants et aux interviews mais comportent encore des biais, les principaux étant :

- Pour l'analyse des expériences, les sujets sont encore trop influencés par la personne effectuant l'entretien. Nous nous sommes tournés ensuite vers des méthodes d'autoconfrontation (André, 2020).
- Le comportement de suivi de visages, trop simple, est rapidement identifié par les sujets.
- Les comportements de 'recentrages', au joystick ou à l'aide des comportements construits, qui permettent de relancer le suivi de visages, restent trop dépendant de l'opérateur, avec tous les biais énoncés plus haut.

1.3. Limitations

Au-delà du suivi de visages, qui autonomise en partie notre lampe au cours des passations, nous restons très dépendants de l'opérateur et des biais induits : interprétations des comportements construits de la lampe, interprétation des attitudes des sujets, souhait de créer sa propre interaction avec les sujets, de répondre à certaines de leur -supposées- demandes d'interaction, etc. Nous souhaitons donc enrichir notre suivi de visages en évitant à la fois des mouvements stéréotypés ou systématiques et des interventions humaines (commandes manuelles) trop sujettes à biais. Nous souhaitons toutefois que les mouvements de la lampe soient spécifiques aux sujets et cohérents par rapport à leurs attitudes, par exemple une attitude identique du sujet devra déclencher une même *réaction*, un même mouvement, de la lampe.

2. Les SOM, classifieurs neuronaux

Pour obtenir cette adaptation dépendante du sujet, nous utilisons l'algorithme DSOM (Rougier, Boniface, 2011), variation des traditionnels classifieurs *cartes auto-organisatrices* (SOM) (Kohonen, 2001), permettant d'apprendre des données dynamiques, c'est-à-dire changeantes au cours du temps. Ici ce sont les sujets, et leurs attitudes, qui changent au cours du temps (au cours d'une expérience ou d'une expérience à une autre) et le comportement de la lampe doit s'y adapter. Plus exactement, ce sont les traits des visages, les coordonnées des points des yeux, bouches, etc. dans le visage des sujets qui sont les données prises en compte pour notre classification.

2.1. Cartes auto-organisatrices

L'algorithme des *cartes auto-organisatrices* est un algorithme de classification des données permettant de les représenter avec une cohérence topologique : deux classes

proches dans l'espace de représentation de la carte représenteront des données proches. Cette classification permet de représenter les données dans un espace fini (la taille de la carte), chacune des données (un vecteur) étant *représentée* par une classe, celle qui a le vecteur le plus *proche* (au sens d'une mesure numérique, la distance euclidienne par exemple) de la donnée. Ces classes sont apprises à l'aide des données, permettant ainsi une classification propre à chaque jeu de données, ou à chaque expérience produisant des données.

2.2. Carte auto-organisatrice dynamique

Nous avons modifié l'algorithme original de SOM afin de rendre sa règle d'apprentissage et son voisinage indépendants du temps. Il en résulte un couplage étroit entre l'environnement et le modèle qui assure à la fois stabilité et plasticité. Plus précisément, La carte DSOM est une carte neurale dotée d'une structure (un hypercube ou un réseau hexagonal) où chaque neurone i se voit attribuer une position fixe \mathbf{p}_i avec q la dimension de la carte (généralement 1 ou 2). Le processus d'apprentissage est un processus itératif dans lequel les vecteurs $\mathbf{s} \in \Omega$ sont présentés séquentiellement sur la carte par rapport à la fonction de densité de probabilité f . Pour chaque vecteur \mathbf{w}_i présenté, un gagnant $s \in N$ est déterminé selon l'équation. Tous les codes \mathbf{i} du codebook W sont alors décalés vers \mathbf{v} selon l'équation :

$$\Delta \mathbf{w}_i = \varepsilon \|\mathbf{v} - \mathbf{w}_i\|_{\Omega} h_{\eta}(i, s, \mathbf{v})(\mathbf{v} - \mathbf{w}_i) \quad (1)$$

avec ε le pas d'apprentissage (constant), et $h_{\eta}(i, s, \mathbf{v})(\mathbf{v} - \mathbf{w}_i)$ la fonction de voisinage de la forme:

$$h_{\eta}(i, s, \mathbf{v})(\mathbf{v} - \mathbf{w}_i) = \exp \frac{-1 \|\mathbf{p}_i - \mathbf{p}_s\|^2}{\eta 2 \|\mathbf{v} - \mathbf{w}_s\|_{\Omega}^2} \quad (2)$$

avec η le paramètre d'élasticité. A noter que lorsque $\mathbf{v} = \mathbf{w}_s$, on a $h_{\eta}(i, s, \mathbf{v}) = 0$.

L'algorithme DSOM est donc essentiellement une variante de l'algorithme SOM dont la dépendance temporelle a été supprimée. Les fonctions d'apprentissage et de voisinage ont été remplacées par les équations précédentes pour répondre à deux idées principales :

- Si un neurone est suffisamment proche des données, il n'est pas nécessaire que les autres apprennent quoi que ce soit : le gagnant peut représenter les données.
- S'il n'y a pas de neurone suffisamment proche des données, tout neurone apprend les données en fonction de sa propre distance par rapport aux données.

Cela entraîne plusieurs conséquences sur la notion de voisinage qui est désormais dynamique et conduit à une auto-organisation qualitativement différente qui peut être contrôlée à l'aide d'un paramètre d'élasticité libre η .

3. Classifier pour interagir

Nous décrivons dans cette section l'utilisation de notre classifieur dans le cadre de nos expériences avec la lampe du groupe Psyphine.

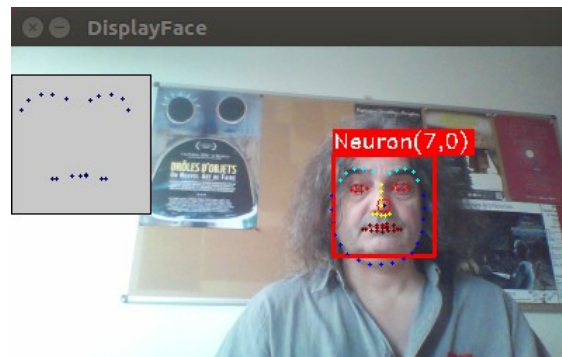


Figure 5: La localisation des visages et la capture des traits principaux à partir de l'image de la caméra de la lampe sont effectuées en temps réel via la librairie *Open-CV*. Seules les informations relatives aux sourcils et à la bouche sont ici utilisées par la carte auto-organisatrice.

3.1. Extraire d'une image les traits du visage

Comme nous pouvons le voir sur la figure 5, à partir de l'image reçue de la caméra, nous isolons un visage (le carré rouge) à l'aide de la librairie *opencv* (Itseez, 2015). Nous extrayons ensuite de chaque image un vecteur de points (les points bleus dans le carré gris en haut à gauche de la figure) représentant certains traits¹ de ce visage à l'aide de l'algorithme de (Kazemi, Sullivan, 2014). C'est ce vecteur de points qui constitue la donnée numérique apprise par notre classifieur.

3.2. Classifier les attitudes des sujets

La figure 6 montre les vecteurs qui sont appris en temps réel par notre le classifieur DSOM. Pour décrire rapidement l'apprentissage d'une carte, il faut noter qu'une classe est considérée comme *gagnante*, pour un visage, quand les points extraits sont les *plus proches* des points appris par cette classe. Cette classe se modifie alors en fonction de cette nouvelle donnée pour s'en rapprocher, et entraîne ses voisins (dans la topologie de la carte) pour qu'ils s'en approchent aussi, mais de moins en moins en fonction de la distance dans la carte.

Chaque image capturée par la caméra est traitée par l'algorithme d'extraction de traits et apporte ainsi un nouvel exemple pour l'apprentissage de la carte, c'est-à-dire pour la spécialisation de la classe gagnante et de ses voisines. Chaque classe représente ainsi une posture caractéristique, une attitude, (représentée par les points) de visage, et nous

¹ Ici sourcils et lèvres, mais tous les points visibles dans le carré rouge sont disponibles.

disposons ainsi en temps réel d'une information sur les postures de visages les plus fréquentes du ou des sujets placés dans le champ de la caméra.



Figure 6: Une carte DSOM avec 64 (8x8) neurones. Chaque neurone est représenté par son prototype (points bleus) dans l'espace des images. L'auto-organisation permet de s'assurer que deux neurones proches (au sens de la topologie de la carte) possèdent des prototypes proches (au sens de la distance entre prototypes). Dans l'exemple montré ci-dessus, les neurones dans les coins supérieur gauche et inférieur droit sont opposés, mais il existe un chemin continu dans la carte permettant de passer de l'un à l'autre). Le paramètre d'élasticité permet de contrôler la similitude entre les prototypes de deux neurones voisins.

Cette classification est unique et totalement dépendante des traits et des postures du sujet au cours de l'expérience, une personne très expressive aura une distance entre les classes la représentant plus importante qu'une autre moins expressive, néanmoins, le nombre de classes cartographiant ses expressions sera le même. Cette classification se fait ainsi de manière totalement autonome et sans aucune interprétation d'éventuelles émotions portées par ces postures. Notre question est d'isoler des postures de visages, pas d'en comprendre la signification.

Pour finir, DSOM permettant l'apprentissage de données dynamiques, cette classification s'adaptera à plusieurs sujets, c'est-à-dire que certaines classes se spécialiseront sur l'un

des sujets, les autres sur d'autres, et même à des changements de personnes, l'élasticité de DSOM permet l'apprentissage en temps réel de nouveaux prototypes.

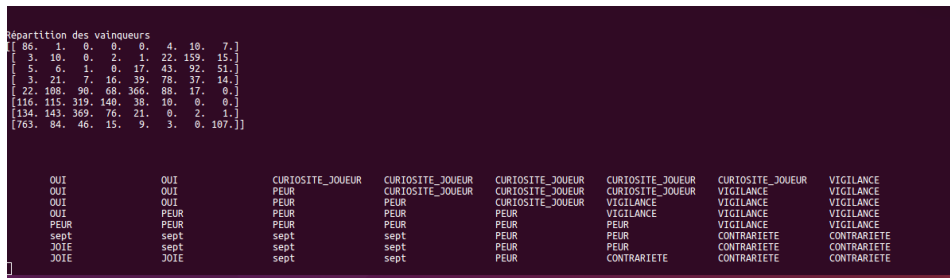


Figure 7: Distribution des comportements sur les classes. Nous pouvons voir au dessus la répartition des classes gagnantes, c'est-à-dire le nombre d'images pour lesquelles chaque classe fut la plus proche.

3.3. Associer des comportements aux classes

Au cours de la mise en place de nos différentes expériences, nous nous sommes construit un catalogue assez varié de *comportements* de la lampe, c'est-à-dire de mouvements préprogrammés (Becker *et al.*, 2019), certains ayant été conçus pour tenter d'exprimer une émotion (peur, etc.) d'autres non (vers la droite, vers le haut, etc.). Comme le montre la figure 7, en utilisant toujours notre classificateur DSOM, nous distribuons nos comportements sur une carte de même taille que celle des traits. Ce sont les vecteurs de commandes motrices qui sont cette fois appris, de la même manière que précédemment les points extraits des images de la caméra. Les deux cartes étant de la même taille, nous pouvons lier un comportement de la lampe à chaque expression des sujets apprise. Ainsi la classe gagnante (en rouge sur la figure 6) sera associée au comportement *PEUR*. Cette distribution des comportements de la lampe se fait au début de l'expérience et reste donc, dans l'état actuel de l'algorithme, la même tout au long de l'expérience. Elle peut aussi être gardée d'une passation à une autre.

4. Une expérience autonome

Avec ces cartographies, nous disposons maintenant d'un algorithme permettant de déclencher un mouvement de notre objet robotisé en fonction des attitudes du ou des sujets ayant été présents dans le champ de sa caméra. Il reste à construire une expérience à l'aide des fonctions dont dispose notre lampe, une expérience qui, en plus d'être autonome, est spécifique aux sujets présents et à leurs attitudes au cours de l'expérience. L'expérience s'articule donc principalement sur le suivi de visages, puis, quand ce suivi échoue, sur l'apprentissage décrit ci-dessus. Le suivi de visages peut échouer pour de nombreuses raisons, la lampe ne peut physiquement plus suivre le mouvement (les articulations sont aux limites de leurs possibilités), le sujet suivi a été perdu (déplacement trop rapide par exemple), le visage est masqué, etc. L'expérience proposée est décrite par l'algorithme suivant :

Apprendre la carte des comportements (DSOM)

RÉPÉTER

Capturer une image depuis la caméra de la lampe

SI un visage peut être extrait de l'image

. Apprendre ce nouveau visage dans la carte des expressions à l'aide de DSOM

. Incrémenter le nombre de victoires de la classe gagnante

. $TI = 0$;

. Déplacer la lampe pour centrer le visage dans l'image de la caméra

SINON

SI Le temps d'impatience (TI) est dépassé

. Déclencher le comportement associé à la classe des expressions qui a le plus gagné.

. $TI=0$

. Mise à zéro des victoires de la carte des expressions

A noter que les paramètres de cet algorithme sont :

- La taille des cartes auto-organisatrices.
- Le nombre de comportements de la lampe.
Il n'est pas souhaitable d'avoir plus de comportements que de classes, il est même conseillé d'en avoir beaucoup moins, pour distribuer un même comportement sur plusieurs classes.
- Les temps d'impatience.
Temps au terme duquel une action autre que le suivi de visages sera déclenchée. Il est généralement de 1500 milli-secondes.
- Les traits pris en compte.
Nous disposons de différentes catégories de traits (nez, sourcils, lèvres, yeux, chacun pouvant être décomposé : droite/gauche, haut/bas). Il n'est pas toujours utile de considérer tous les points dans l'apprentissage, d'autant que certains traits (lèvres par exemple, par leur nombre de points, écrasent les autres). Dans l'expérience qui illustre cet article nous n'utilisons que les sourcils et les lèvres.

5. Discussion

Le modèle présenté dans cet article permet de rendre le comportement de notre lampe autonome et interactif. Autonome en ce qu'il ne requiert plus l'intervention d'un opérateur et interactif en ce qu'il réagit aux informations captées par sa caméra (les visages en particulier) sans pour autant chercher à interpréter ces informations. C'est là une propriété importante puisqu'elle garantit l'absence de biais de la part de l'opérateur et autorisera à terme des interactions plus neutres, du moins du côté de la lampe. Nous obtenons une chorégraphie propre aux sujets placés devant la lampe et à leurs attitudes, sans jamais que ces dernières ne soient qualifiées en terme d'émotions.

Nous travaillons actuellement sur une version plus robuste permettant de laisser notre objet dans un lieu ouvert et de le laisser interagir au grès du passage aléatoire des personnes et des visages entrant dans le champ de sa caméra. Cela requiert de régler quelques paramètres comme le temps d'impatience (pas de visage détecté) au-delà duquel la lampe déclenche des comportements moteurs qui sont alors susceptibles d'attirer des personnes à elle. Cependant, toute la difficulté de ces expériences réside dans la nature du milieu, à la fois ouvert et non contrôlé. Comme nous l'avons expliqué auparavant, la nature solennelle des expériences en laboratoire semble perturber l'interaction entre les participants et la lampe, ce qui ne semble plus être le cas lorsque les sujets sont libres d'interagir (ou non) avec la lampe. Nous devons donc à terme définir une méthode de mesure objective afin de rendre compte de l'expérience.

À noter que l'utilisation de DSOM sur un capteur, ici une caméra, pour apprendre le contexte de l'expérience peut en théorie être appliquée à tout autre capteur disponible.

Bibliographie

- Adam, B-D. et Zhou, S. (2012). Projet Pinokio. <http://www.ben-dror.com/pinokio>.
- André, V. (2020). Apports de l'autoconfrontation dans l'analyse d'une expérience robotique. *Wacai 2020*. [hal-02933468v1](https://hal.archives-ouvertes.fr/hal-02933468v1)
- André, V. et Boniface, Y (2018). Quelques considérations interactionnelles autour d'une expérience robotique. *WACAI 2018*. <https://hal.archives-ouvertes.fr/hal-01862725>.
- Becker, J. André, V., Dutech, A. (2019). QUALCOM : Une Expérience Sur La Qualification Des Comportements d'une Lampe Robotique. *Techniques & culture : Revue semestrielle d'anthropologie des techniques*, Varia. Éditions de l'EHESS. <https://hal.archives-ouvertes.fr/hal-02075467>.
- Itseez (2015). "Open Source Computer Vision Library". <https://github.com/itseez/opencv>.
- Kazemi, V. et Sullivan, J. (2014). One Millisecond Face Alignment with an Ensemble of Regression Trees. *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, 1867–74. CVPR '14. Washington, DC, USA: IEEE Computer Society. doi:10.1109/CVPR.2014.241.
- Kohonen, T. (2001). *Self-Organizing Maps*. 3rd ed. Berlin, Heidelberg: Springer-Verlag.
- Psyphine. 2011. <https://psyphine.hypotheses.org>.
- Rougier, N. P. et Boniface, Y. (2011). Dynamic Self-Organising Map. *Neurocomputing* 74 (11). Elsevier: 1840–7. doi:10.1016/j.neucom.2010.06.034.