Clay VR

Laurent Gaidon Jean-Baptiste Guignard Ophir Paz Kim Savaroche

HINS
ophir.paz@hins.fr

Résumé
Clay VR est un SDK pour la reconnaissance de gestes sur smartphone depuis n'importe quelle caméra intégrée. Il est conçu pour être utilisé en VR et en commande distale pour enrichir drastiquement les possibilités d'interactions de l'utilisateur — en affichant sa main détourée dans l'environnement virtuel. Cette expérience *touchless* permet de s'affranchir des problématiques liées aux périphériques d'entrées matériels, mais implique de nouveaux enjeux techniques. Ainsi, la solution proposée est basée sur un réseau de neurones récurrent contrôlé heuristiquement par un système expert.

Mots clef
Reconnaissance gestuelle, VR, Computer vision, RNN, Système expert.

Abstract
Clay VR is a SDK for gesture recognition on smartphones from any embedded lens (no need for additional hardware). It is designed for uses in VR and distal control to drastically enrich the user's interaction possibilities – it displays one's own hands contoured in a virtual environment. Such a touchless experience is control without the hardware, touch, pinpoint or remote-control pains, ensures immersion by preserving self-perception, but implies massive technical difficulties. Thus the presented solution is RNN-based, and heuristically jugulated by an expert system.

Keywords
Gesture recognition, VR, Computer vision, RNN, Expert system.

1 Contexte
La reconnaissance des mains et, conséquemment, celle des gestes effectués par celles-ci sont des biométries essentielles à l'identification de l'utilisateur dans un environnement virtuel. Avec l'essor de technologies disruptives comme la VR et l'AR, les organismes privés et publics affairés dans la quête immersive cherchent à offrir une expérience utilisateur toujours plus intuitive et naturelle, tant du côté de l'UX que de celui des contenus et de leur réalisme. Or, les réponses à cet effort d'intuitivité sont et restent matérielles. Elles consistent en l'ajout d'un *device* – manette, joystick, exosquelette, etc. – qui rappelle continuellement la rupture entre monde réel et monde virtuel. Une visualisation des mains en temps réel, sans recours à quelque *hardware* que ce soit, dans un environnement projeté ou mixte amenuiserait pourtant le sentiment de dissociation.

Pour une reconnaissance gestuelle (GR) ouverte au grand public, il est nécessaire d'analyser des images 2D capturées dans des environnements (notamment spatiaux et lumineux) changeants à l'extrême. La main des utilisateurs, dont les morphologies sont hétérogènes (1), varie continuellement dans l'espace (X, Y, Z).

2 Solution
Clay est une solution software de reconnaissance gestuelle sur smartphone. Elle permet, à partir d'une caméra 2D, de repérer et suivre les mains d'un utilisateur en temps réel, y compris en profondeur (Z), dans une zone de captation vidéo en vue d'interpréter ses gestes. Une fois le mouvement (pré-encodé ou appris) identifié, il est possible de le coupler à une action/instruction : augmenter ou baisser le son d'un fichier sonore, saisir un objet en 3D, activer des commandes embarquées dans un habitacle, etc.

La présente solution se concentre sur le procédé même de captation des mains. Elle se propose en particulier d'agir en amont des filtres d'interprétation. Sa finalité est, en effet, de permettre à cette solution logicielle – sans ajout de matériel de type PYR, de sonde, ou de caméra à profondeur de champ – de s'adapter aux variations, souvent radicales, de l'environnement (lumière, mouvement, arrière-plan, etc.) afin de fournir une représentation

stable et des mains sur un fond noir. Lesdites mains doivent être détourées, affichées en premier plan, et soustraites d'un fond remplaçable en fonction des contextes (notamment VR), ce qui place ce travail sous l'égide thématique de la computer vision, des NN ou RNN, des systèmes experts et de la logique floue.

Clay doit être multiplateforme, *i.e.* exploitable par n'importe quel appareil muni d'une caméra (y compris 2D frontale). La nature du langage de programmation utilisé étant directement corrélatif du résultat obtenu – en cela, il lui est constitutif, fidèlement à la conception « prothétique » de l'outil comme anthropologiquement constitutif (2) – l'anticipation de la portabilité (ou de la compatibilité d'emblée) du code au travers des systèmes d'exploitation est primordiale. Pour une programmation orientée « objet » et pour la création d'une architecture générique adaptable à tous les hardwares, C++ s'impose. Corrélativement, la mémoire vive strictement nécessaire au bon fonctionnement du programme est réservée et ne peut pas être empruntée par un autre processus lancé sur un smartphone. Cette allocation statique améliore le niveau de performances du système et, par conséquent, les filtres de traitement sont plus puissants pour un résultat plus pertinent. Une économie des ressources internes du *device* permet un traitement d'image à 60 FPS pour un rendu en temps réel.

Le système doit être capable de suivre les formes et de les identifier entre les *frames* successives. Le défi technique est de créer un système assez intelligent pour extraire des propriétés suffisantes permettant de différencier et identifier une main au milieu d'autres formes parasites. Ainsi si une forme a été repérée comme une main ouverte et que l'utilisateur la ferme pour montrer son poing alors un *matching* de formes permettra de savoir que malgré une différence notable, l'élément est le même.

Il ne faut pas simplement reconnaître des gestes mais être également capable de détecter l'intention de l'utilisateur sous-entendue par ce geste. Un déplacement de la main peut avoir une intention neutre dans le cas où l'utilisateur se détend les muscles. A contrario, il peut avoir un sens quand il souhaite effectuer un balayage vers la droite ou la gauche. En VR, les mouvements de la tête accentuent cette incertitude : la main ne change pas de position, mais bouge par rapport à la caméra.

Pour réussir ceci, le système peut être présenté sous trois étapes : la reconnaissance du geste, sa contextualisation et sa validation.

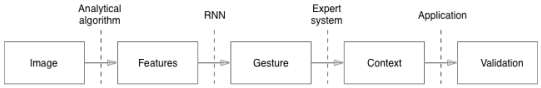

Figure 1 Les différentes phases de la solution ClayVR

2.1 Reconnaissance du geste

La première partie consiste à extraire les caractéristiques de l'image pour alimenter un réseau de neurones afin d'identifier le geste effectué. Dans un premier temps, l'extraction de ces *features* est réalisée en deux phases : isoler la forme de la main, puis analyser cette forme. Isoler le contour de la main sous-entend isoler les éléments au premier plan puis détourer la main. Cette dernière effectue une succession d'analyses permettant de calculer des données comme sa position en X, en Y, en Z ou encore les coordonnées de ses doigts.

Les données étant extraites, elles alimentent un réseau de neurones récurrents pour être catégorisées par geste. Un outil probabiliste à apprentissage supervisé s'est en effet imposé pour déceler les patterns dont nous avions l'intuition (3), de part la quantité et la complexité des informations déduites de l'image. De plus, un geste s'étale par définition sur plusieurs frames et son identification nécessite de prendre en compte les variations des données dans le temps. Le RNN est donc apparu comme le plus approprié puisqu'il permet la classification de séquences de données.

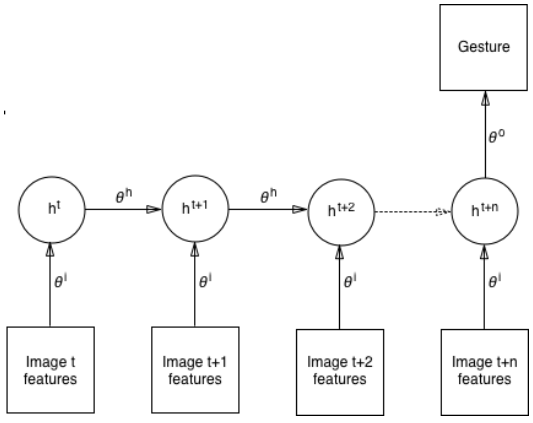

Figure 2 RNN utilisé en « many to one » dans Clay VR

Néanmoins l'apprentissage supervisé nécessite une phase d'apprentissage qui demande un jeu de données significatif et qualitatif. Contrairement à la reconnaissance d'un objet dans une image qui peut exploiter une base de données de photographies facilement accessibles, notre jeu d'entraînement est très spécifique. En effet, l'angle de la caméra est déterminant, les gestes n'ont pas le même aspect selon si le point de vue est devant ou au-dessus de l'appareil. Ajoutons à cela que la manière de faire des gestes est propre à chaque individu et au contexte dans lequel ils sont exécutés. Ceci nous a amené à créer une application spécifique qui permet de reproduire les conditions équivalentes à l'environnement de production afin d'obtenir des données pertinentes pour entraîner le système.

2.2 Contextualisation

Cependant la reconnaissance simple des gestes ne suffit pas. Un geste n'est pas systématiquement déclencheur d'une action précise. Par exemple, la forme d'une main en ombre chinoise, ne permet pas de faire la différence entre la droite côté paume et la gauche sur le dos. Pourtant pour reconnaître le *flip* de la main gauche, nous avons besoin de savoir que celle-ci est présente et qu'elle a été retournée. Cette contextualisation du geste est donc cruciale et peut être modélisée sous forme de prédicats logiques. L'utilisation d'un système expert, appliquant ces règles grâce à un chaînage avant, transforme un geste en signe contextualisé, porteur d'une intention.

2.3 Validation

Enfin, une fois le geste reconnu et remis dans son contexte, il nous reste à vérifier la cohérence globale de l'action demandée. Si le geste est synonyme de baisser le son mais qu'aucune musique n'est lancée alors il ne semble pas cohérent de d'influencer le volume sonore. Cette phase de validation permet d'assurer la liaison entre les gestes et l'état de l'application.

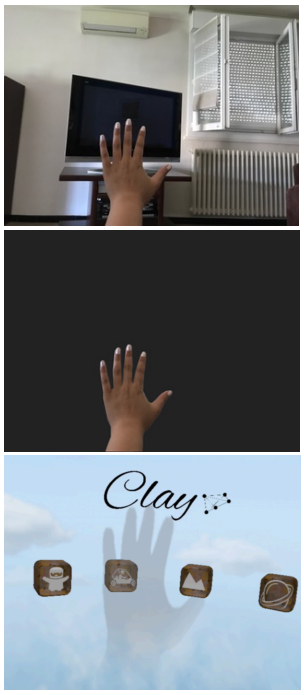

Figure 3 Hand Processing

3 Conclusion

La particularité et la difficulté de notre projet réside ainsi dans la liberté de mouvements de l'utilisateur. Contrairement à une manette dont les actions sont sans ambiguïté (2)grâce à des *input*s prédéfinis, ClayVR a pour ambition d'être un outil prothétique. L'utilisateur doit oublier sa présence afin de profiter pleinement de l'expérience proposée. Si le système est perfectible alors l'usager se distanciera de la solution software. Cette succession d'outils contrôle les résultats dans la seule intention d'interpréter correctement les gestes et déclencher l'action souhaitée.

Bibliographie

1. **Elgammal, Ahmed, Muang, Crystal et Hu, Dunxu.** *Skin Detection - a Short Tutorial.* Piscataway, : s.n., 2009.
2. **Samuel, A. L.** *Some Studies in Machine Learning Using the Game of Checkers. II-Recent Progress A.* s.l. : IBM Journal of Research and Development (Volume:44, Issue:1.2), 1959.
3. **Simondon, Gilbert.** *L'individuation psychique et collective.* Paris : Aubier, 1989.
4. **Simondon, Gilbert .** *Du mode d'existence des objets techniques.* Paris : Aubier, 1958.
5. **Clark, Andy.** *Natural-Born Cyborgs. Minds, Technologies and the Future of Human.* Oxford/New York : Oxford UP, 2003.