

Adoiraccourcix : Sélection de Commandes sur Écrans Tactiles Multi-Points par Identification des Doigts

Alix Goguey¹, Géry Casiez², Thomas Pietrzak², Daniel Vogel³ & Nicolas Roussel¹

¹Inria Lille, ²Université Lille 1, France, ³ Université de Waterloo, Canada
{alix.goguey, nicolas.roussel}@inria.fr, {gery.casiez, thomas.pietrzak}@lifl.fr,
dvogel@uwaterloo.ca

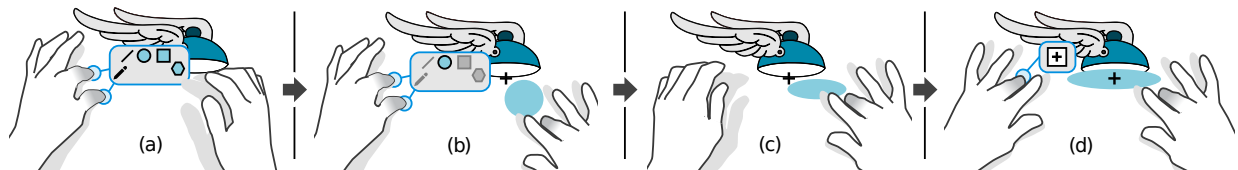


Figure 1: Illustration partielle de la logique d'Adoiraccourcix : (a) un accord de doigts de la main non dominante affiche du *feedforward* précisant les différentes commandes associées aux doigts de la main dominante, pour cet accord. Ici le pouce et l'index affichent les cinq outils de dessin associés aux doigts de la main dominante ; (b) le majeur de la main dominante sélectionne l'outil de création d'ellipses ; (c) le mouvement de la main dominante contrôle les paramètres de la commande, ici la taille et la forme de l'ellipse ; (d) un autre accord de la main non dominante définit une contrainte pour la commande en train d'être exécutée. Ici le majeur de la main non-dominante contraint la création d'ellipse suivant son centre plutôt que son coin supérieur gauche.

RÉSUMÉ

Alors que les raccourcis clavier permettent aux utilisateurs experts des interfaces WIMP d'exécuter rapidement des commandes, il existe aujourd'hui peu d'équivalents pour grands écrans tactiles multi-points. Afin de remédier à ce problème, nous avons conçu Adoiraccourcix, une technique d'interaction tirant parti de l'identification des doigts sur tables interactives, pour la sélection rapide de commandes associée au contrôle continu de paramètres de celles-ci. Après avoir présenté la logique de conception, nous illustrons Adoiraccourcix au travers d'une application de dessin vectoriel.

Mots Clés

Multi-points ; identification des doigts ; raccourcis ; sélection de commandes ; manipulation directe

ACM Classification Keywords

H.5.2 Information Interfaces and Presentation (e.g. HCI): User interfaces

INTRODUCTION

La sélection de commandes, qui permet à l'utilisateur de réaliser ses tâches en agissant sur des objets virtuels, constitue l'un des aspects essentiels des systèmes interactifs. Elle s'effectue généralement à l'aide de menus et de barres d'outils, mais aussi via des raccourcis clavier qui permettent un accès rapide et facile aux commandes les plus usuelles. Si de tels raccourcis sont communs dans les interfaces WIMP traditionnelles, il existe aujourd'hui peu d'équivalents pour l'interaction sur grands écrans multi-points.

Les boutons physiques étant rares sur les dispositifs tactiles, les concepteurs d'interfaces ont généralement recours à des boutons graphiques, à des techniques prenant en compte le nombre de doigts, les gestes effectués ou encore la temporisation, pour déclencher différentes commandes. Ces approches ne permettent cependant d'accéder qu'à un nombre relativement limité de commandes, elles occupent de la place sur l'écran et peuvent aussi poser des problèmes de performance et de fiabilité.

Pour pallier le manque de techniques d'accès rapide à un grand nombre de commandes sur les écrans tactiles ainsi que le permettent les raccourcis clavier, nous proposons Adoiraccourcix, une technique d'interaction tirant parti de l'identification des doigts en contact avec l'écran qui permet, dans une même et unique interaction, le déclenchement rapide de commandes *et* le contrôle continu des paramètres de celles-ci (Figure 1).

PROTOTYPE D'IDENTIFICATION DES DOIGTS

L'identification fiable des doigts sur une surface multi-points reste un problème technique difficile. L'utilisation d'empreintes digitales est prometteuse [1], mais n'est pas encore opérationnelle pour de grandes surfaces. La vision

par ordinateur n'est pas robuste [2, 3], à moins d'utiliser un système de marqueurs attachés sur une paire de gants [4]. Nous présentons ci-dessous le prototype que nous avons mis en place pour obtenir une identification robuste des doigts et ainsi mettre en œuvre Adoiraccourcix.

Notre prototype utilise des *GameTraks* qui sont des contrôleurs de jeu servant à l'origine à simuler des *swings* de club de golf. Cinq *GameTraks* nous permettent de repérer les positions des extrémités de dix doigts dans un espace en 3 dimensions en y attachant les bouts de chacun des fils.

Nous avons écrit une variante d'un serveur TUIO¹ en C++ et utilisant la librairie *libgametrak*². Cette implémentation établit la correspondance entre la position 3D des doigts et la position 2D des contacts sur un écran 32" 3MTM C3266PW multi-touch surface. À chaque doigt est associé une homographie pour établir cette correspondance. Ces homographies sont calculées dans une phase préalable de calibrage où des points 3D sont associés à une position 2D connue. Une fois le système mis en place, l'identification d'un doigt en contact se fait par la recherche du point 3D le plus proche. Le système génère ensuite un événement TUIO comprenant l'identité du doigt et de la main. Tout le système, ainsi que l'application de dessin vectoriel décrite par la suite, fonctionnent sur un MacBook Pro doté d'un processeur 2.5GHz Intel Core i5 et de 8Go de RAM.

L'architecture de notre système est modulaire, ce qui nous permet d'utiliser d'autres prototypes. Par exemple, nous avons aussi implémenté une identification des doigts avec un *Leap Motion*³ fixé au dessus de notre surface tactile. Le *Leap Motion* est capable de repérer l'extrémité des doigts en 3D à 100Hz. La phase de calibrage est similaire à celle précédemment décrite, mais puisque toutes les positions 3D sont données dans le même référentiel, une seule homographie est nécessaire. Cependant, comme la plupart des systèmes basés sur de la vision par ordinateur à un seul point de vue, le *Leap Motion* est sensible aux occultations, ce qui pénalise la fiabilité, même si l'utilisateur n'est plus attaché. Pour ces raisons, la démonstration suivante et les évaluations préliminaires utilisent la version à base de *GameTraks* qui est plus fiable.

APPLICATION DE DÉMONSTRATION

Afin d'expérimenter Adoiraccourcix, nous avons développé une application de dessin vectoriel dans l'esprit d'Adobe Illustrator⁴ et InkScape⁵. Ce type d'applications a d'abord l'intérêt d'être connu par beaucoup d'utilisateurs et ensuite l'avantage de posséder de nombreuses commandes afin de créer, éditer et manipuler du contenu, ce qui est intéressant pour tester notre technique.

L'application possède six modes et 26 commandes (Figure 2). Les six modes sont : un mode de *sélection* pour la manipulation d'objets (un doigt pour la translation, deux

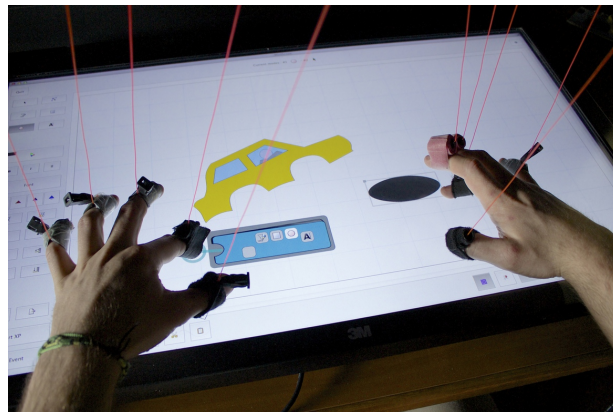


Figure 2: Notre application de dessin vectoriel intègre une GUI, mais elle peut être entièrement contrôlée grâce aux interactions Adoiraccourcix.

doigts pour le RST), un mode d'*édition* pour modifier certains attributs des objets tels que les points de contrôles, et quatre modes de *dessin* pour créer des rectangles, ellipses, courbes et du texte. Afin de comparer les interactions Adoiraccourcix avec une GUI classique, notre application possède des barres d'outils qui permettent l'accès à tous les modes et commandes. En mode expert ces barres d'outils peuvent être cachées afin d'augmenter l'espace de travail disponible à l'écran.

L'application a pour but d'illustrer Adoiraccourcix mais elle met également en œuvre des techniques d'interaction analogues à d'autres techniques de l'état de l'art [4] pour lesquelles aucun mécanisme de *feedforward* n'est disponible. Il s'agit des commandes associées à un seul doigt et des accords uni-manuels.

NOTE

Pour plus d'information sur Adoiraccourcix, veuillez vous référer au papier long accompagnant cette démonstration.

REFERENCES

1. Holz, C., and Baudisch, P. Fiberio: a touchscreen that senses fingerprints. In *Proc. UIST* (2013). ttp.
2. Kung, P., Küser, D., Schroeder, C., DeRose, T., Greenberg, D., and Kin, K. An augmented multi-touch system using hand and finger identification. In *CHI EA* (2012), 1431–1432.
3. Malik, S., Ranjan, A., and Balakrishnan, R. Interacting with large displays from a distance with vision-tracked multi-finger gestural input. In *Proc. UIST* (2005), 43–52.
4. Marquardt, N., Kiemer, J., Ledo, D., Boring, S., and Greenberg, S. Designing user-, hand-, and handpart-aware tabletop interactions with the TouchID toolkit. In *Proc. ITS* (2011), 21–30.

¹ <http://www.tuio.org/>

² <https://code.google.com/p/libgametrak/>

³ <https://www.leapmotion.com>

⁴ <http://www.adobe.com/products/illustrator.html>

⁵ <http://inkscape.org/>