

Survol d'un demi-siècle de recherche sur l'interaction gestuelle

David Bonnet

1. INTRODUCTION

Encouragé par l'essor des *smartphones*, et des *tablettes tactiles*, de plus en plus de systèmes interactifs reposent sur l'interaction gestuelle, c'est-à-dire sur l'association entre une commande et une suite de points. La littérature concernant ce domaine étant particulièrement vaste, nous nous restreindrons aux gestes effectués dans un plan et à la présentation des travaux de recherche majeurs qui leur sont consacrés.

Sketchpad (Sutherland, 1963) fait figure de précurseur dans le paysage de l'interaction homme-machine. Il s'agit du premier système permettant à l'utilisateur d'entrer un geste, à l'aide d'un stylo optique, pour créer et modifier des formes géométriques par manipulation directe.

Les travaux que nous allons présenter s'inscrivent dans la lignée des problématiques soulevées par *Sketchpad* : le développement de nouveaux **périphériques** d'entrée gestuelle plus performants et fiables, la création d'algorithmes permettant de **reconnaître** les gestes effectués, la mise en œuvre de **guides** et d'indices permettant d'apprendre des vocabulaires de gestes et enfin la proposition de **méthodes** permettant de les rendre efficaces. Ces projets de recherche sont également motivés par des **applications** qui vont au-delà des outils de dessin, que nous décrirons également.

2. SYSTÈMES D'ENTRÉE GESTUELLE

Ce qui suit est un court historique permettant de situer dans l'ordre chronologique l'apparition de nouveaux appareils d'entrée gestuelle plane :

- Vers 1950, le *Light Gun* (Everett, ~1950) fut le premier périphérique optique capable de lire la position d'un point sur un écran afin d'effectuer des diagnostics.
- En 1952, le premier *Trackball* (Cranston et Longstaff, 1952) fut inventé et se présentait comme une grande souris inversée. L'utilisateur manipulait la boule, maintenue sur des coussins d'air pour minimiser les frictions, à l'intérieur du socle inamovible.
- En 1957, le *Lightpen* (Gurley, 1957) adopte la forme d'un stylo et est le premier du genre à pouvoir être utilisé sur un écran.
- En 1963, arrive la « *Rand Tablet* » (Davis et Ellis, 1963), l'ancêtre des tablettes graphiques telles que

nous les connaissons aujourd'hui.

- Dès 1964, la *souris* (English et Engelbart, 1967) fut inventée.
- En 1965, un mécanisme fonctionnel d'écran tactile capacitif est décrit (Johnson, 1965).
- En 1972, le premier écran tactile à être massivement distribué dans l'état d'Illinois voit le jour (Ebeling et al., 1972). Des lumières infrarouges réparties sur le cadre intérieur de l'écran permettant de détecter la présence d'un doigt sur l'écran.
- En 1982, le premier appareil multi-touch (Mehta, 1982) est créé. Il repose sur l'emploi d'une caméra placée derrière une plaque de verre rétroéclairée filmant les ombres créées par la main de l'utilisateur. Le flux vidéo est alors numérisé et transmis à un processeur pour analyse.
- En 1983, ce qui semble être le premier appareil tactile sur le marché, le *HP 150*, est apparu (HP, 1983).
- En 1984, le premier écran multi-touch avec une nappe de détection transparente est inventée (Boie, 1984).
- Enfin, en 2009, *ThinSight*, le premier écran LCD doté de capteurs optiques sur toute sa surface (Izadi et al., 2009), permettant d'émettre et de capter de la lumière est réalisé. Il permet alors de lire et même scanner tout objet sur ou proche de la surface de l'écran.

3. INTÉRÊT COGNITIF

Appert et Zhai (2009) montrent au travers d'une étude empirique le fort potentiel des raccourcis gestuels. Les participants ont su apprendre plus rapidement davantage (jusqu'à 16) de gestes arbitraires que de raccourcis claviers en ayant su les restituer avec moins d'erreurs.

4. ALGORITHMES DE RECONNAISSANCE

Plusieurs algorithmes de reconnaissance de gestes ont été mis au point.

Le plus connu est celui de Rubine (1991), qui utilise un mécanisme d'apprentissage supervisé. Il classifie les gestes selon 13 caractéristiques, dont les angles initiaux, la longueur du tracé, les dimensions du rectangle englobant, mais aussi la vitesse maximale et la durée du geste.

Le reconnaissseur *\$I* (Wobbrock et al., 2007) s'abroge de la nécessité de l'apprentissage et permet d'être facilement utilisable pour le prototypage rapide d'applications basées sur les gestes. Néanmoins, l'algorithme testant incrémentalement différentes rotations du geste, la durée de calcul augmente rapidement avec la taille du vocabulaire utilisé.

Le *Protractor* (Li, 2010) étend l'algorithme 1\$ en apportant une solution analytique qui permet de déterminer immédiatement la rotation optimale entre le geste entré et celui du vocabulaire.

L'algorithme d'Appert et Bau (2010) permet de reconnaître les gestes et de déterminer leur échelle en cours de tracé (reconnaisseur *à priori*). Il se base pour cela sur un vecteur d'angles que fait chaque segment avec un axe de référence.

5. GUIDES ET TECHNIQUES D'APPRENTISSAGE

Plusieurs projets de recherche ont étudié et mis au point des techniques dans le but de soutenir l'utilisateur dans la découverte et l'apprentissage de gestes.

Deux mécanismes sont utilisés : le **feedforward**, qui renseigne sur la forme des gestes et leur association avec une commande particulière, et le **feedback**, qui informe sur la reconnaissance du geste pendant ou après sa réalisation.

Ces mécanismes mettent en jeu deux modes : le mode **expert** et le mode **novice**. Dans le mode expert, seul le tracé effectué est affiché ainsi que, une fois le geste accompli, le nom de la commande sélectionnée ou alors directement le résultat de l'opération. L'utilisateur bascule dans le mode novice lors d'une hésitation caractérisée par une absence de mouvement du périphérique d'entrée. Ce mode consiste en général à afficher les gestes possibles avec le nom ou le symbole des commandes associées. L'efficacité des techniques proposées résulte dans la façon de passer du mode novice au mode expert.

5.1. Menus (« *menu gestures* »)

Les *Marking Menus* (Kurtenbach et Buxton, 1991) étendent le principe des menus circulaires (Callahan et al., 1988) en introduisant ces deux modes. Dans le mode expert, l'utilisateur fait une marque dans la direction du secteur désiré sans que le menu soit affiché, alors qu'il l'est dans le mode novice.

Des études empiriques (Kurtenbach et Buxton, 1993) montrent que les utilisateurs peuvent faire des marques jusque dans 12 directions différentes avec un taux d'erreur acceptable. Les *Marking Menus hiérarchiques* (Kurtenbach et Buxton, 1993) permettent alors d'augmenter le nombre de commandes en faisant des gestes en forme de zigzags. Une autre variante, les *Multi-Strokes Menus* (Zhao et Balakrishnan, 2007) permettent d'augmenter davantage le nombre de commandes et

de réduire l'espace requis puisque la navigation dans la hiérarchie se fait à l'aide de traits discontinus. Cette richesse accrue de commandes se fait néanmoins au prix d'une difficulté de mémorisation accrue, la seule dimension de direction n'étant pas très expressive.

Afin d'y remédier, les *Marking Menus hybrides* (Isokoski et Käki, 2002), tout comme les *Flower Menus* (Bailly et al., 2008), introduisent une dimension supplémentaire : les courbures, optionnelles, à la fin des gestes, les rendent plus faciles à retenir et conviennent également à une utilisation sur de petites surfaces tactiles.

Enfin, les *FlowMenus* (Guimbretière et Winogard, 2000) et les *Control Menus* (Pook et al., 2000) proposent de sélectionner une commande et de définir, dans la continuité du geste, la valeur d'un paramètre.

5.2. Gestes emblématiques (« *iconic gestures* »)

Les gestes emblématiques sont, par leur expressivité, plus faciles à apprendre, mais nécessitent des guides plus élaborés. Les *anti-sèches* avec des animations en contexte (Kurtenbach et al., 1994) permettent à la fois de découvrir l'ensemble des commandes disponibles ainsi que la manière de les utiliser en conjonction avec un objet sélectionné. Par exemple, si l'opération de suppression d'un objet consiste à le barrer horizontalement, une animation barrera l'objet sélectionné invitant alors l'utilisateur à faire de même.

Ces guides sont néanmoins très gourmands en termes de surface d'affichage et l'absence de feedback continu ne permet pas de savoir si le geste en train d'être réalisé est correctement reconnu ou pas. *Fluid Sketches* (Arvo et Novins, 2000) transforme au fur et à mesure le tracé en des formes simples (ronds, carrés...) les plus proches. Il exige cependant de tracer une portion considérable de la forme avant de voir l'effet d'embellissement (notamment, plusieurs boucles sont requises avant que le système ne détecte un rond). La *détection des coins* en analysant les variations de la vitesse du tracé (Agar et Novins, 2003) permet alors de le segmenter plus facilement et d'appliquer plus rapidement la transformation sur des objets polygonaux. Ainsi, un carré sera reconnu dès que le quatrième bord sera dessiné, au lieu de tracer deux fois son contour.

OctoPocus (Bau et Mackay, 2008) propose une aide plus sophistiquée en ne se limitant pas à des formes géométriques « classiques ». Elle consiste à afficher les chemins des gestes possibles à partir de la position actuelle du curseur et à faire varier en continu leur épaisseur pour informer de leur taux de reconnaissance. Si l'un d'entre eux passe sous un certain seuil, le tracé correspondant disparaît complètement.

5.3. Gestes multi-touch

Principalement pour des raisons d'occlusion et de complexité visuelle, les guides précédents sont difficilement exploitables pour des gestes multi-touch. *ShadowGuides* (Freeman et al., 2009) affiche sur une zone déportée la main telle que vue par le système (une *ombre*) en la complétant avec des ombres, flèches et autres annotations graphiques symbolisant les positions possibles pour activer différentes commandes. L'impact positif sur la mémorisation par rapport aux vidéos montrant les gestes filmés est significatif.

Gesture Play (Bragdon et al., 2010) utilise une approche basée sur des métaphores physiques afin d'encourager l'apprentissage de gestes. Bien que les utilisateurs n'aient pas retenus davantage les gestes par rapport à un apprentissage par vidéos, ils ont néanmoins montré plus d'intérêt et d'envie à apprendre les gestes en question.

6. CONCEPTION DE VOCABULAIRES DE GESTES

Les méthodes de mise au point de vocabulaires de gestes ont pour but de les rendre facile à réaliser, à apprendre et à mémoriser.

Pour les *Marking Menus*, Kurtenbach et Buxton (1993), se posent la question sur la manière de répartir les commandes dans les différents niveaux. Les résultats de l'étude montrent que les largeur et profondeur maximales sont respectivement de 8 et 2, le facteur limitant étant le nombre d'erreurs acceptable. Néanmoins, pour une quantité fixe de commandes, alterner entre 4x4 éléments et 8x8 éléments donne les mêmes résultats. Enfin, les performances identiques avec la souris et le stylet (avec un très léger avantage pour celui-ci) montrent que ces menus sont compatibles avec les souris.

Long et al. (2000) donnent un ensemble de recommandations de conception. Par exemple, la similitude perceptive des gestes est en corrélation avec des caractéristiques comme la courbure.

MicroRolls (Roudaut et al., 2009) propose un vocabulaire particulièrement large s'il est mis en rapport avec la taille des gestes. En effet, un MicroRoll s'opère en faisant simplement rouler le pouce, ce qui le rend particulièrement adapté aux appareils tactiles mobiles.

7. APPLICATIONS

Cette section présente des applications de l'interaction gestuelle dans des domaines variés.

7.1. Dessin

Si Buxton et Rubine ont créé des logiciels de dessin (respectivement *GEdit* et *GDP*), c'est avant tout pour illustrer leurs théories ou algorithmes d'interaction gestuelle. Nous présentons ici quelques travaux dont le but est d'aller au-delà du simple prototype, et qui reprennent

souvent des métaphores et techniques anciennes d'illustrations.

Ligne Claire (Baudel, 1994) s'inspire des méthodes des artistes de bande dessinée et permet d'éditer les tracés selon quatre modes. Le premier consiste à créer un trait, le second permet de modifier une partie d'un trait en retraçant le segment à ajuster, le troisième supprime un segment en le hachurant et le dernier permet de joindre deux extrémités. L'édition des lignes peut se faire alors de manière successive et naturelle, sans détériorer le dessin en cours de réalisation.

Digital Tape Drawing (Balakrishnan et al., 1999) reprend une technique utilisée par les graphistes d'automobiles : l'utilisation de scotch photographique permettant d'obtenir une courbure particulière contrôlée avec les deux mains qu'il est difficile, voire impossible, d'obtenir avec un stylet numérique, à main levée.

SKETCH (Zelevnik et al., 1996) permet d'exploiter des tracés effectués sur un plan afin d'en faire des objets tridimensionnels. En fonction de l'orientation et de la position des lignes par rapport au point d'origine, certaines seront extrudées le long d'un trait ou alors par circonvolution. Cette technique autorise ainsi à obtenir des primitives aux dimensions voulues afin de réaliser rapidement une maquette.

Imaginary Interfaces (Gustafson et al. 2010) pousse le concept encore plus loin en effectuant des tracés dans l'espace devant soi qui peuvent être capturés par des téléphones dépourvus de tout écran. Une caméra infrarouge fixée sur la veste capte les mouvements de la main dominante dans le plan défini par la main non dominante, formant un « L » avec le pouce et l'index. Le seul feedback visuel des gestes effectués s'effectue alors dans l'imagination de l'utilisateur. Les tracés capturés peuvent par exemple être retransmis *via* le téléphone à un correspondant pour illustrer plus précisément le chemin qui le mène à l'endroit demandé auparavant.

7.2. Texte

L'utilisation de gestes pour l'entrée de texte présente plusieurs avantages. La simple position des tracés procure suffisamment d'information au système pour alterner rapidement entre texte, schéma, formule mathématique, tableau... Alors que sur les systèmes traditionnels, l'emploi explicite de commandes ou de modes est souvent exigé pour y parvenir.

Les premiers reconnaisseurs de caractères datent de l'époque de *Sketchpad*: (Teitelman, 1964), (Brown, 1964) et (Bernstein, 1964). Celui du *MessagePad* (Apple, 1992-1998) était basé sur les mots. Cela permet de bénéficier du contexte pour corriger les fautes de frappe éventuelles, mais posait des problèmes lorsque le mot écrit ne figure pas dans le dictionnaire.

Certains reconnaisseurs utilisent des formes plus ou moins proches des caractères classiques, altérés au profit d'une entrée gestuelle plus rapide. *Graffiti* (Palm, 1996), utilisé par les assistants personnels tournant sous Palm OS, dispose d'un vocabulaire constitué de caractères majuscules simplifiés pour qu'ils soient réalisables en un seul trait, sans lever le stylet. *Unistroke* (Goldberg et Richardson, 1993), est dans la même lignée, mais propose un alphabet encore plus éloigné de l'alphabet traditionnel afin d'optimiser davantage la vitesse d'écriture. Une étude empirique (Castellucci et MacKenzie, 2008) montre qu'*Unistroke* est plus rapide que *Graffiti*, au prix d'un temps d'apprentissage légèrement plus long.

D'autres systèmes ont une reconnaissance encore plus souple. *Aha! InkWriter* (Aha! Software Corp., 1993) reconnaît la morphologie d'un document rédigé à l'aide d'un stylet. Ainsi, sans en connaître la signification, à partir des caractéristiques des tracés, il est en mesure de détecter des mots, des paragraphes et des schémas sans en connaître la moindre signification. Les connaissances sont suffisantes pour supprimer un ou plusieurs mots d'un simple trait et réarranger les autres mots pour qu'ils occupent l'espace vide, comme dans un logiciel de traitement de texte. Les dessins peuvent être redimensionnés ou même embellis en les remplaçant par la forme primitive la plus proche (par exemple, un rond dessiné à main levée par un rond parfait).

S'il est naturel de vouloir réaliser avec un stylo numérique ce que nous faisons habituellement avec un stylo traditionnel (à savoir, dessiner et écrire des mots qui devront être reconnus par le système), la combinaison geste et clavier logiciel (« *softkeyboard* ») peut se montrer plus efficace et moins sujette aux erreurs. De plus, en combinant le clavier avec les *Marking Menus* (Hashimoto et Togasi, 1995), on peut, d'un geste court, à la fois entrer un caractère et un modificateur (« *majuscule* », « *ctrl* », « *alt* »...) ou alors accéder rapidement à des caractères spéciaux couramment utilisés comme « *enter* », « *suppr* », « *espace* ».

Enfin, d'autres solutions proposent l'entrée de texte en utilisant des gestes continus.

QuickWriting (Perlin, 1998) consiste à sélectionner des caractères, disposés de façon circulaire en huit portions, en traversant les délimitations de chaque portion dans un sens particulier. Si la lettre se trouve au milieu de la portion, en y entrant depuis le centre puis en y retournant la sélectionne. Si elle se trouve sur un bord, il suffit d'entrer dans la portion depuis le centre, puis de retourner vers le centre en passant d'abord dans la portion voisine dont la lettre à sélectionner est la plus proche. Il y a deux avantages inhérents à cette technique: le fait que chaque caractère soit choisi en partant puis en revenant au centre supprime totalement les problèmes de segmentation rencontrés dans les précédents systèmes.

De plus, du fait de la continuité des gestes, le système peut-être utilisé sans regarder, permettant alors de porter l'attention sur ce qui motive la prise de notes et non plus sur ce qui prend les notes.

7.3. Musique

Les gestes sont aussi particulièrement pertinents dans des systèmes de notation qui comportent plus de dimensions que l'écriture.

Dans le cadre du *Structured Sound Synthesis Project* (SSSP), plusieurs logiciels dédiés aux compositeurs furent réalisés. L'un d'entre eux est un outil d'écriture de notes musicales, *Char-rec* (Buxton, 1979). Les notes sont insérées en démarrant le tracé à la position où elles doivent apparaître sur la portée et la forme du trait indique la durée de la note souhaitée (blanche, noire, croche, ...). Il introduit alors la notion de raccourcis, de marques qui caricaturent l'objet à créer. En un geste, on spécifie la hauteur, la position et la durée de la note (3 paramètres en un seul trait).

Musink (Tsandilas et al., 2009) permet aux compositeurs d'associer les caractéristiques de marques faites au stylo (par exemple, les dimensions du tracé, son envergure, la forme générale qui lui est associée...) à leurs logiciels de composition musicale.

7.4. Autres

Il existe évidemment un grand nombre d'autres applications, comme le vocabulaire de gestes dédié au contrôle du trafic aérien (Chatty et Lecoanet, 1996) ou encore le système basé sur les gestes pour contrôler le comportement de robots à distance (Sakamoto et al., 2009).

8. CONCLUSION

L'interaction gestuelle peut, au travers d'outils et d'un vocabulaire de gestes bien défini, rendre accessible des fonctions qui l'étaient difficilement, voire pas du tout, avec les paradigmes usuels.

Sur le plan de la recherche, un certain nombre de voies restent à explorer, notamment l'étude des caractéristiques dynamiques, qui remet alors en question tous les points abordés précédemment (algorithmes, guides, conception de vocabulaires...). L'arrivée et la démocratisation de nouveaux appareils de capture de gestes dans l'espace tridimensionnel soulèvent par ailleurs les mêmes questions.

9. RÉFÉRENCES

- P. Agar et K. Novins, « Polygon recognition in sketch-based interfaces with immediate and continuous feedback », in *Proceedings of the 1st international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, New York, NY, USA, 2003, p. 147–150.
- C. Appert et O. Bau, « Scale detection for a priori gesture recognition », *Proceedings of the 28th international conference on Human factors in computing systems*, p. 879–882, 2010.
- C. Appert et S. Zhai, « Using strokes as command shortcuts: cognitive benefits and toolkit support », in *Proceedings of the 27th international conference on Human factors in computing systems*, New York, NY, USA, 2009, p. 2289–2298.
- J. Arvo et K. Novins, « Fluid sketches: continuous recognition and morphing of simple hand-drawn shapes », *Proceedings of the 13th annual ACM symposium on User interface software and technology*, p. 73–80, 2000.
- G. Bailly, E. Lecolinet, et L. Nigay, « Quinze ans de recherche sur les menus: critères et propriétés des techniques de menus », in *Proceedings of the 19th International Conference of the Association Francophone d'Interaction Homme-Machine*, New York, NY, USA, 2007, p. 119–126.
- G. Bailly, E. Lecolinet, et L. Nigay, « Flower menus: a new type of marking menu with large menu breadth, within groups and efficient expert mode memorization », *Proceedings of the working conference on Advanced visual interfaces*, p. 15–22, 2008.
- R. Balakrishnan, G. Fitzmaurice, G. Kurtenbach, et W. Buxton, « Digital tape drawing », in *Proceedings of the 12th annual ACM symposium on User interface software and technology*, New York, NY, USA, 1999, p. 161–169.
- O. Bau et W. E. Mackay, « OctoPocus: a dynamic guide for learning gesture-based command sets », *Proceedings of the 21st annual ACM symposium on User interface software and technology*, p. 37–46, 2008.
- T. Baudel, « A mark-based interaction paradigm for free-hand drawing », in *Proceedings of the 7th annual ACM symposium on User interface software and technology*, New York, NY, USA, 1994, p. 185–192.
- E. A. Bier, M. C. Stone, K. Pier, W. Buxton, et T. D. DeRose, « Toolglass and magic lenses: the see-through interface », in *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, New York, NY, USA, 1993, p. 73–80.
- B. Boie, Multi-Touch Touch Screen, Bell Labs, Murray Hill, N.J., 1984.
- A. Bragdon, A. Uguray, D. Wigdor, S. Anagnostopoulos, R. Zeleznik, et R. Feman, « Gesture play: motivating online gesture learning with fun, positive reinforcement and physical metaphors », *ACM International Conference on Interactive Tabletops and Surfaces*, p. 39–48, 2010.
- S. J. Castellucci et I. S. MacKenzie, « Graffiti vs. unistrokes: an empirical comparison », *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, p. 305–308, 2008.
- S. Chatty et P. Lecoanet, « Pen computing for air traffic control », in *Proceedings of the SIGCHI conference on Human factors in computing systems: common ground*, New York, NY, USA, 1996, p. 87–94.
- T. Cranston et F. Longstaff, *Trackball*, Ferranti Canada.
- M.R. Davis et T.O. Ellis, The Rand Tablet: A Man-Machine Graphical Communication Device, *Proceedings of the Fall Joint Computer Conference*, 1964a, 325.
- Ebeling, F.A., Johnson, R.L. & Goldhor, R.S., Uof Illinois, Champaign. Infrared Light beam x-y position encoder for display devices. *US Patent 3,775,560*, Filed Feb 1972 / Issued Nov. 1973.
- W.K. English et D.C. Engelbart, Display Selection Techniques for Text Manipulation, *IEEE Transactions on Human-Factors in Electronics*, 1967, 8(1), 5–15.
- D. Freeman, H. Benko, M. R. Morris, et D. Wigdor, « ShadowGuides: visualizations for in-situ learning of multi-touch and whole-hand gestures », *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, p. 165–172, 2009.
- D. Goldberg et C. Richardson, « Touch-typing with a stylus », in *Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems*, New York, NY, USA, 1993, p. 80–87.
- F. Guimbretière et T. Winograd, « FlowMenu: combining command, text, and data entry », *Proceedings of the 13th annual ACM symposium on User interface software and technology*, p. 213–216, 2000.
- B.M. Gurley et C.E. Woodward, Light-pen links computer to operator, *Electronics*, 1966, 32, 85–87.
- S. Gustafson, D. Bierwirth, et P. Baudisch, « Imaginary interfaces: spatial interaction with empty hands and without visual feedback », *Proceedings of the 23rd annual ACM symposium on User interface software and technology*, p. 3–12, 2010.
- M. Hashimoto et M. Togasi, « A virtual oval keyboard and a vector input method for pen-based character input », in *Conference companion on Human factors in computing systems*, New York, NY, USA, 1995, p. 254–255.
- HP-150 touchscreen personal computer with HP 9121 dual drives, 1983 : <http://www.hp.com/hpinfo/about/hp/histnfacts/museum/personalsystems/0031/>
- P. Isokoski et M. Käki, « Comparison of two touchpad-based methods for numeric entry », *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, p. 25–32, 2002.
- S. Izadi et al., « ThinSight: a thin form-factor interactive surface technology », *Communications of the ACM*, vol. 52, p. 90–98, déc. 2009.
- Johnson, E.A. (1965). Touch Display – A novel input/output device for computers. *Electronics Letters*, 1(8), 219–220.
- P.-O. Kristensson et S. Zhai, « SHARK2: a large vocabulary shorthand writing system for pen-based computers », in *Proceedings of the 17th annual ACM symposium on User interface software and technology*, New York, NY, USA, 2004, p. 43–52.

- G. Kurtenbach, T. P. Moran, et W. Buxton, « Contextual animation of gestural commands », *COMPUTER GRAPHICS FORUM*, vol. 13, p. 83–90, 1994.
- G. Kurtenbach et W. Buxton, « Issues in combining marking and direct manipulation techniques », *Proceedings of the 4th annual ACM symposium on User interface software and technology*, p. 137–144, 1991.
- G. Kurtenbach et W. Buxton, « The limits of expert performance using hierarchic marking menus », *Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems*, p. 482–487, 1993.
- Y. Li, « Protractor: a fast and accurate gesture recognizer », *Proceedings of the 28th international conference on Human factors in computing systems*, p. 2169–2172, 2010.
- J. Long, J. A. Landay, L. A. Rowe, et J. Michiels, « Visual similarity of pen gestures », *Proceedings of the SIGCHI conference on Human factors in computing systems*, p. 360–367, 2000.
- Metha, Nimish (1982), *A Flexible Machine Interface*, M.A.Sc. Thesis, Department of Electrical Engineering, University of Toronto.
- M. Nancel, S. Huot, et M. Beaudouin-Lafon, « Un espace de conception fondé sur une analyse morphologique des techniques de menus », *Proceedings of the 21st International Conference on Association Francophone d'Interaction Homme-Machine*, p. 13–22, 2009.
- K. Perlin, « Quikwriting: continuous stylus-based text entry », in *Proceedings of the 11th annual ACM symposium on User interface software and technology*, New York, NY, USA, 1998, p. 215–216.
- S. Pook, E. Lecolinet, G. Vaysseix, et E. Barillot, « Control menus: execution and control in a single interactor », *CHI '00 extended abstracts on Human factors in computing systems*, p. 263–264, 2000.
- A. Roudaut, E. Lecolinet, et Y. Guiard, « MicroRolls: expanding touch-screen input vocabulary by distinguishing rolls vs. slides of the thumb », *Proceedings of the 27th international conference on Human factors in computing systems*, p. 927–936, 2009.
- D. Rubine, « Specifying gestures by example », *ACM SIG-GRAPH Computer Graphics*, vol. 25, p. 329–337, juill. 1991.
- D. Sakamoto, K. Honda, M. Inami, et T. Igarashi, « Sketch and run: a stroke-based interface for home robots », *Proceedings of the 27th international conference on Human factors in computing systems*, p. 197–200, 2009.
- I. E. Sutherland, « Sketchpad: a man-machine graphical communication system », *Proceedings of the May 21-23, 1963, spring joint computer conference*, p. 329–346, 1963.
- T. Tsandilas, C. Letondal, et W. E. Mackay, « Musink: composing music through augmented drawing », in *Proceedings of the 27th international conference on Human factors in computing systems*, New York, NY, USA, 2009, p. 819–828.
- J. O. Wobbrock, A. D. Wilson, et Y. Li, « Gestures without libraries, toolkits or training: a \$1 recognizer for user interface prototypes », *Proceedings of the 20th annual ACM symposium on User interface software and technology*, p. 159–168, 2007.
- R. C. Zeleznik, K. P. Herndon, et J. F. Hughes, « SKETCH: an interface for sketching 3D scenes », *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, p. 163–170, 1996.
- S. Zhai et P.-O. Kristensson, « Shorthand writing on stylus keyboard », in *Proceedings of the SIGCHI conference on Human factors in computing systems*, New York, NY, USA, 2003, p. 97–104.
- S. Zhao et R. Balakrishnan, « Simple vs. compound mark hierarchical marking menus », *Proceedings of the 17th annual ACM symposium on User interface software and technology*, p. 33–42, 2004.