

Mouvements pseudo-rigides pour des interactions multi-doigts plus flexibles

Yosra Rekik
Inria Lille
yosra.rekik@inria.fr

Nicolas Roussel
Inria Lille
nicolas.roussel@inria.fr

Laurent Grisoni
Université Lille 1, LIFL, Inria Lille
laurent.grisoni@lifl.fr

ABSTRACT

Multi-touch interaction requires a trade-off between users' desires and capabilities, and gesture recognition constraints. Current approaches to that problem lack flexibility. The number of fingers used for a gesture usually plays a key part in the recognition process, for example. In this paper, we propose the use of *pseudo-rigid movements* to increase the flexibility of this process. We show how these movements can be determined in real time from the contact information usually available. We explain how they allow to free the recognition process from the number of fingers used and to move towards multi-movement gestures, independent or coordinated. Finally, we present an interaction technique that takes advantage of this increased flexibility.

Author Keywords

Tactile interaction ; multi-touch interaction ; pseudo-rigid movement ; flexibility ; power ; simplicity

ACM Classification Keywords

H.5.2 [Information interfaces and presentation] : User interfaces

General Terms

Design ; Human Factors

INTRODUCTION

L'avènement des surfaces tactiles multi-doigts appelle à la création de nouvelles techniques d'interaction adaptées à ces interfaces, à leurs utilisateurs et à leurs contextes d'usage. L'interaction gestuelle joue un rôle crucial dans ce cadre, mais sa mise en œuvre nécessite un compromis difficile entre les désirs et capacités des utilisateurs, et les contraintes de la machine. Pour celle-ci, les gestes doivent être précé-

sément définis et reproduits afin de limiter les ambiguïtés et erreurs d'interprétation. Ceci conduit généralement à des commandes gestuelles simplistes ou au contraire difficiles à apprendre et à exécuter. Pour les utilisateurs, l'interaction gestuelle multi-doigts manque de fait souvent de flexibilité, leur imposant des contraintes qu'ils ne comprennent pas ou ne savent pas toujours respecter. Augmenter la flexibilité de l'interaction tout en préservant son efficacité technique est aujourd'hui clairement un des défis posés par les interfaces tactiles multi-doigts.

Les interactions possibles sur les pavés (*touchpads*), écrans ou tables tactiles multi-doigts reposent sur quelques gestes élémentaires prédéfinis (e.g. balayer, pincer, pivoter) ou sur des mouvements libres. Dans un cas comme dans l'autre, le nombre de doigts utilisés intervient souvent assez tôt dans la classification des gestes, celle-ci restant par ailleurs souvent implémentée de manière ad-hoc. En réduisant au minimum l'importance des trajectoires, le nombre de doigts posés peut même être l'unique critère utilisé, pour la sélection de commandes dans des menus par exemple [2]. Interpréter différemment un même geste selon le nombre de doigts utilisés augmente bien sur le vocabulaire d'entrée. Mais cela introduit des modes qui peuvent être difficiles à percevoir et dont il peut être impossible de sortir s'ils conditionnent trop fortement le processus d'interprétation.

Pour des interactions mono-trace¹, des guides dynamiques ont été proposés pour expliciter l'état interne du processus de reconnaissance et faciliter l'apprentissage et la production des gestes utiles [3]. Dans ce même contexte, des méthodes de reconnaissance relativement simples permettent d'augmenter le vocabulaire d'entrée en se basant sur la dynamique du mouvement, e.g. [10, 9, 6, 8, 1]. Jiang et al. ont récemment proposé d'extraire une trace unique de traces multi-doigts pour utiliser dans ce contexte des méthodes mono-trace [4]. Telle qu'ils la proposent, cette réduction est

1. Nous utilisons cette expression pour désigner des interactions gestuelles où un seul point est suivi dans le temps (e.g. la position d'un doigt ou d'un stylo sur une tablette, ou d'un curseur à l'écran), indépendamment du nombre de segments constituant la trace

toutefois incompatible avec des interactions multi-doigts indépendantes, e.g. à deux mains, ou à plusieurs.

Afin de s'affranchir des contraintes posées par les implémentations ad-hoc, des outils ont également été proposés pour spécifier les gestes multi-doigts à l'aide d'expressions régulières [5] ou d'exemples [7]. Mais le nombre de doigts utilisés reste dans ces outils un facteur déterminant. Une étude réalisée par Wobbrock et al. [12] suggère pourtant que le nombre de doigts mobilisés pour produire un geste n'est pour l'utilisateur qu'un facteur secondaire qui ne devrait pas servir à déterminer la commande à exécuter, mais plutôt à la paramétrer.

Dans cet article, nous proposons le concept de *mouvements pseudo-rigides* pour augmenter la flexibilité de l'interaction tactile multi-doigts. Nous donnons une définition de ce concept et montrons comment ces mouvements peuvent être déterminés en temps réel à partir des informations fournies par les surfaces tactiles multi-doigts. Nous expliquons comment ils permettent de s'affranchir du nombre de doigts et comment leurs caractéristiques et leur dynamique peuvent être utilisées pour mettre en œuvre des techniques d'interaction multi-mouvements, indépendants ou concertés. Nous illustrons enfin quelques-unes de ces nouvelles possibilités à l'aide d'un exemple de technique d'interaction pour la manipulation d'objets graphiques.

MOUVEMENTS PSEUDO-RIGIDES

Les surfaces tactiles multi-doigts fournissent typiquement de manière régulière des informations sur les points de contact détectés (e.g. identifiant unique, position 2D) et leur dynamique (e.g. apparition, déplacement, disparition). Nous proposons de considérer qu'un ensemble de points de contact participent au même *mouvement rigide* lorsque la dynamique propre à chacun peut être approximée de manière satisfaisante à l'aide de deux transformations : la translation du barycentre des points, et une rotation de l'ensemble autour de lui.

La détermination des mouvements rigides en présence nécessite l'analyse des déplacements des points de contact entre instants successifs, la fréquence de cette analyse pouvant être inférieure à celle à laquelle la surface tactile fournit ses informations. Considérant les positions d'un ensemble de points observées aux instants t_{i-1} et t_i , nous proposons de calculer la translation de leur barycentre, d'estimer la rotation subie par les points autour de lui puis d'évaluer la qualité de l'approximation résultant de ces deux transformations pour décider si les points participent à un unique mouvement rigide ou à plusieurs. Dans le second cas, nous proposons d'utiliser une méthode classique de partitionnement (*clustering*) afin de déterminer les mouvements en question. Nous proposons également de préserver dans le temps sous

une certaine condition les regroupements de points effectués lorsqu'il ne correspondent plus à des mouvements strictement rigides (d'où le terme *pseudo-rigide*). Pour chaque groupe préservé, on pourra ainsi déterminer un changement d'échelle éventuel, dans le repère local du barycentre.

La suite de cette section détaille la manière dont les transformations sont calculées, la méthode de partitionnement utilisée, et les nouvelles possibilités qui en découlent.

Calcul de la translation, de la rotation et du changement d'échelle d'un ensemble de points

Si l'on considère un ensemble $\{P_i^k\}_k$ de points de contact à un instant t_i , le barycentre de ces points est donné par $C_i = \frac{1}{n} \sum_k P_i^k$ et sa translation par $T_i = C_i - C_{i-1}$.

En utilisant la notation standard $\det(u,v)$ pour désigner le déterminant de deux vecteur, soit en 2D $\det(u,v) = u_x v_y - u_y v_x$, l'angle de rotation de l'ensemble des points autour du barycentre est tel que :

$$\tan(\theta_i) = \frac{\sum_k \det(\overrightarrow{C_{i-1}P_{i-1}^k}, \overrightarrow{C_iP_i^k})}{\sum_k \overrightarrow{C_{i-1}P_{i-1}^k} \cdot \overrightarrow{C_iP_i^k}} \quad (1)$$

L'équation (1) fournit deux valeurs possibles : l'angle θ que l'on cherche et $\theta + \pi$, en radians. La valeur à retenir est celle qui minimise le facteur d'erreur E_i donné par (2), R_θ désignant la rotation vectorielle d'angle θ :

$$E_i = \left\| R_\theta(\overrightarrow{C_{i-1}P_{i-1}}) - \overrightarrow{C_iP_i} \right\|^2 \quad (2)$$

Pour finir, le facteur d'échelle ρ_i est donné directement par :

$$\rho_i = \frac{\sum_k \overrightarrow{C_{i-1}P_{i-1}^k} \cdot \overrightarrow{C_iP_i^k}}{\sum_k \left\| \overrightarrow{C_{i-1}P_{i-1}^k} \right\|^2} \quad (3)$$

Détermination et suivi des mouvements

L'approche proposée repose sur une méthode classique de partitionnement. A chaque instant, nous commençons par mettre tous les points de contact considérés dans un même groupe, puis nous exécutons la procédure suivante :

1. nous calculons la translation et la rotation du groupe selon le mode de calcul ci-dessus ;
2. nous appliquons ces deux transformations à chaque point P_{i-1} du groupe observé à l'instant t_{i-1} pour obtenir sa position théorique à l'instant t_i , notée \hat{P}_i ;

3. nous évaluons la qualité de l'approximation des mouvements observés en calculant la moyenne des distances euclidiennes entre \hat{P}_i et P_i ;
4. si cette moyenne est supérieure à un seuil, nous considérons que nous ne sommes pas en présence d'un mouvement rigide : nous décomposons le groupe à l'aide de l'*algorithme des k-moyennes* [11] avec $k = 2$ et un critère de similarité prenant en compte la position et le déplacement des points ;
5. nous fusionnons les groupes décrits par des transformations similaires, l'algorithme des k-moyennes ne faisant que diviser des groupes incohérents sans chercher à minimiser leur nombre global ;
6. nous exécutons sur chaque nouveau groupe produit la présente procédure.

Le critère de classification utilisé étant sensible aux déplacements, celle-ci tend à regrouper les points de contact se déplaçant à petite vitesse ou à l'arrêt. Des tests informels ayant révélé que ce comportement correspondait rarement aux attentes des utilisateurs, notre implémentation ne considère pas à chaque instant l'ensemble des points de contact détectés mais ignore ceux qui participaient à l'instant précédent à un mouvement dont le déplacement était inférieur à un seuil. Cette condition est celle qui permet de préserver des mouvements non nécessairement rigides, i.e. pseudo-rigides, pour lesquels le changement d'échelle éventuel peut être calculé. Les nouveaux points de contact sont eux-aussi traités séparément par notre implémentation. Une fois la classification effectuée, on cherche pour chaque nouveau point le mouvement dont le barycentre est le plus proche. Si la distance est inférieure à un seuil, le point est ajouté au mouvement. Dans le cas contraire, un nouveau mouvement est créé pour lui.

L'approche proposée associe un mouvement pseudo-rigide à chaque point de contact. En conservant la mémoire de ces associations d'un instant à l'autre, il devient possible de s'intéresser à la dynamique des mouvements. On peut ainsi détecter l'apparition d'un nouveau mouvement, sa progression (déplacement des points qui le composent avec possible apparition ou disparition de quelques uns), sa division en de nouveaux mouvements ou sa fusion avec d'autres (Figure 1), et sa disparition. Dans chacun des cas, un ou plusieurs événements peuvent être générés pour informer les couches logicielles supérieures.

Avantages de l'approche proposée

La translation, la rotation et le changement d'échelle caractérisant un mouvement pseudo-rigide peuvent être utilisés pour mettre en œuvre des interactions classiques de type balayage, pivotement ou pincement. La trace correspondant au déplacement de son barycentre peut également alimenter un algorithme de reconnaissance mono-trace afin de reconnaître des gestes plus variés.

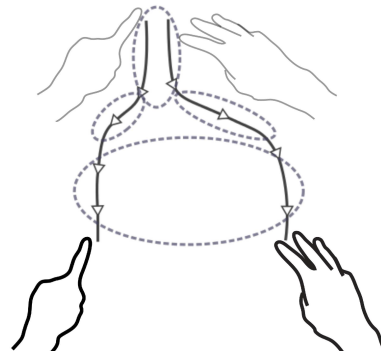


Figure 1 : Division d'un mouvement pseudo-rigide en deux mouvements indépendants, puis fusion de ceux-ci en un nouveau mouvement pseudo-rigide

Un premier avantage de notre approche par rapport à l'état de l'art est que ces différents gestes peuvent être détectés quel que soit le nombre de doigts utilisés, celui-ci pouvant même varier au cours de l'interaction. Comme suggéré par Wobbrock et al [12], le nombre de doigts peut ainsi paramétrer la commande au lieu de la déterminer, pour ajuster la précision d'un zoom ou spécifier le nombre de copies d'un objet à créer, par exemple. Un deuxième avantage est que la reconnaissance de gestes peut s'effectuer simultanément sur différents mouvements identifiés, permettant ainsi des interactions multi-mouvements (e.g. bimanuels, multi-utilisateurs), indépendants ou concertés.

En suivant l'apparition, la progression et la disparition des mouvements, on peut envisager l'adaptation à des mouvements concertés des méthodes existantes pour la reconnaissance de gestes multi-doigts. La rotation d'une carte pourrait par exemple être déclenchée par deux mouvements ou plus dont les barycentres suivraient ensemble une trajectoire circulaire. La division et la fusion des mouvements n'ayant pas d'équivalent dans le domaine du multi-doigts, leur suivi offre des degrés de liberté supplémentaires à explorer.

EXEMPLE D'UTILISATION

Un premier travail de conception et de mise en œuvre de techniques d'interaction basées sur les mouvements pseudo-rigides a été réalisé. L'une des techniques conçues à cette occasion illustre l'intérêt de l'indépendance de notre approche au nombre de points de contacts et du suivi des divisions et fusions de mouvements.

La technique en question permet la manipulation de formes géométriques simples sur un écran tactile. Tout mouvement pseudo-rigide initié sur le fond de l'écran agit comme un outil de sélection : les objets intersectés par la boîte englobante des points de contacts participant à ce mouvement lui sont associés et subissent par la suite ses transformations. Les objets étant associés au mouvement et non aux points de

contact qui le déterminent, ceux-ci peuvent varier au cours de l'interaction. En exécutant un mouvement pseudo-rigide à deux mains, l'utilisateur peut collecter très rapidement un grand nombre d'objets. Il peut ensuite lever une main et quelques doigts de l'autre pour réduire fortement la taille de la boîte englobante et augmenter ainsi la précision de l'outil de sélection tout en gardant le contrôle des objets sélectionnés. Il peut ensuite reposer des doigts ou sa deuxième main pour ajuster la portée de l'outil.

Plusieurs mouvements pseudo-rigides peuvent être utilisés simultanément, à deux mains ou à plusieurs, pour créer des sélections d'objets indépendantes (Figure 2). Lorsqu'une fusion de mouvements indépendants est détectée, les sélections d'objets sont elles-aussi fusionnées. Pour ajouter un objet à une sélection commencée avec la main droite, on peut ainsi le sélectionner avec la main gauche et effectuer un mouvement identique avec les deux mains, par exemple. Lorsqu'un mouvement disparaît, la sélection qui lui était associée persiste et peut être reprise en initiant un nouveau mouvement sur l'un de ses objets. Lorsqu'un mouvement se divise, la sélection est par contre détruite et une commande de remplacement automatique des objets est optionnellement exécutée.



Figure 2 : Sélections de formes géométriques à l'aide de deux mouvements pseudo-rigides indépendants

CONCLUSION

Nous avons introduit le concept de *mouvements pseudo-rigides* et montré comment ces mouvements peuvent être extraits des informations fournies par les surfaces tactiles multi-doigts actuelles. Nous avons expliqué comment ils permettent de s'affranchir du nombre de doigts. Nous avons expliqué comment les méthodes de reconnaissance mono-trace peuvent être appliquées sur la trace associée à la dynamique du barycentre de chaque mouvement. Nous avons également expliqué comment la dynamique de l'ensemble de ces mouvements peut être utilisée pour reconnaître des gestes multi-mouvements, et non plus multi-doigts ou mono-trace. Nous avons enfin présenté une technique d'interaction illustrant quelques unes de ces possibilités. Nous poursuivons notre travail de conception et de mise

en œuvre de techniques d'interaction basées sur ces mouvements pseudo-rigides. La facilité d'apprentissage et d'exécution des gestes dépendant de nombreux facteurs notamment liés au contexte dans lequel ils sont exécutés, nous travaillons également à la définition de protocoles et de cadres expérimentaux permettant d'évaluer nos techniques.

REMERCIEMENTS

Ce travail a été financé par le projet Interreg 2 Mers "SHIVA".

BIBLIOGRAPHIE

1. Anthony, L., and Wobbrock, J. O. \$N-Protractor : A fast and accurate multistroke recognizer. In *Proceedings of GI'12*, 117–120. CIPS.
2. Bailly, G., Lecolinet, E., and Guiard, Y. Finger-Count & Radial-Stroke shortcuts : two techniques for augmenting linear menus on multi-touch surfaces. In *Proceedings of CHI'10*, 591–594. ACM.
3. Bau, O., and Mackay, W. E. OctoPocus : a dynamic guide for learning gesture-based command sets. In *Proceedings of UIST'08*, 37–46. ACM.
4. Jiang, Y., Tian, F., Zhang, X., Liu, W., Dai, G., and Wang, H. Unistroke gestures on multi-touch interaction : supporting flexible touches with key stroke extraction. In *Proceedings of IUI'12*, 85–88. ACM.
5. Kin, K., Hartmann, B., DeRose, T., and Agrawala, M. Proton : multitouch gestures as regular expressions. In *Proceedings of CHI'12*, 2885–2894. ACM.
6. Li, Y. Protractor : a fast and accurate gesture recognizer. In *Proceedings of CHI'10*, 2169–2172. ACM.
7. Lü, H., and Li, Y. Gesture coder : a tool for programming multi-touch gestures by demonstration. In *Proceedings of CHI'12*, 2875–2884. ACM.
8. Malacria, S., Lecolinet, E., and Guiard, Y. Clutch-free panning and integrated pan-zoom control on touch-sensitive surfaces : the cyclostar approach. In *Proceedings of CHI'10*, 2615–2624. ACM.
9. Roudaut, A., Lecolinet, E., and Guiard, Y. MicroRolls : expanding touch-screen input vocabulary by distinguishing rolls vs. slides of the thumb. In *Proceedings of CHI'09*, 927–936. ACM.
10. Rubine, D. H. *The automatic recognition of gestures*. PhD thesis, Carnegie Mellon University, Dec. 1991.
11. Tan, P.-N., Steinbach, M., and Kumar, V. *Introduction to Data Mining*. Addison-Wesley, 2006.
12. Wobbrock, J. O., Morris, M. R., and Wilson, A. D. User-defined gestures for surface computing. In *Proceedings of CHI'09*, 1083–1092. ACM.