

タッチパネル上において 指の接触点の関係を活用するメニュー

吉川 拓人^{†1} 志築 文太郎^{‡2} 田 中 二 郎^{†2}

大型のタッチパネル上において指の接触点の関係を活用することにより表示されるメニューを開発した。ユーザは両手の指の接触点を用いて任意の位置と方向にメニューを表示させる。そして、それらの接触点を用いてメニューとのインタラクションを行う。また、本メニューは従来のタッチジェスチャと共存することができる。

A Menu Utilizing the Relationship between Finger Contacts on a Multi-touch Panel

TAKUTO YOSHIKAWA,^{†1} BUNTAROU SHIZUKI^{†1}
and JIRO TANAKA^{†1}

In this paper, we present a menu that utilizes the relationship between finger contacts on a large multi-touch panel. Using finger contacts, a user can invoke the menu in any position and orientation and interact with it. The menu can coexist with conventional multi-touch gestures.

1. はじめに

大型タッチパネルが複数人または個人による作業などに用いられている。本研究の対象とする大型タッチパネルは、一方の画面端からもう一方の画面端に人間の手が届かないほど

^{†1} 筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻
Department of Computer Science, Graduate School of Systems and Information Engineering,
University of Tsukuba

^{‡2} 筑波大学システム情報系
Faculty of Engineering, Information and Systems, University of Tsukuba

大きく、またタッチパネル面と表示系が一致しており、複数の指の接触点を検出できるタッチパネルを指す。このようなタッチパネルには、テーブル状に水平に置かれたテーブルトップ型と呼ばれるものと、垂直に置かれたウォール型と呼ばれるものがある。

大型タッチパネルにおけるメニューにはユーザの手元に見やすい方向に表示されて欲しいという要求がある。この要求は大型タッチパネルの物理的なサイズが大きいこと、複数ユーザが大型タッチパネルに対して様々な位置や方向から同時に指の接触を用いた操作を行うことから発生するものである。

しかしながら、従来のプルダウンやポップアップのようなメニューを大型タッチパネルに適用しただけでは、この要求を満たすことができない。従来のプルダウンメニューは固定の位置に置かれることが多く、物理的なサイズの大きなタッチパネル上においては、ユーザは自身の位置によっては画面上のメニューに手を伸ばすだけでは届かないことがある。この場合、ユーザはメニューに手が届く位置まで移動してから操作を行うことが求められる。また、大型タッチパネル上においてユーザは様々な方向から操作を行うため、従来の表示される方向が一定であるメニューは、ユーザ自身の位置や方向によっては読みにくいことがある。さらに、ユーザが様々な位置や方向から同時にタッチパネルを使用するために、メニューをすべてのユーザにとって手が届きやすく、読み取りやすい固定の位置に置くことは難しい。

ポップアップメニューを任意の位置に表示させることは可能であるが、表示される方向が固定であるため、ユーザの位置や方向によっては読みにくいことがある。

本稿において示すメニューは、先に述べた要求をユーザのタッチパネル上における両手の指の接触点の関係を活用することにより実現する。また、本メニューは指の接触点のみを用いるため、¹⁴⁾ や⁴⁾ のような手の腹の接触情報や手の近接情報を用いたメニューと異なり、3点以上の指の接触点を認識可能な従来のタッチパネルにおいて実現できる。さらに3点よりも多くの指の接触点を用いることによって、より多様なインタラクションを実現できる。

2. 関連研究

大型タッチパネル向けのメニューは現在までも研究がなされている。それらには、ユーザが任意の位置や方向に配置できるものと固定の位置に配置されるものがある。

2.1 ユーザが任意の位置や方向に配置できる大型タッチパネル向けメニュー

¹⁴⁾ は片手の5本の指と手の腹の接触を検知して5本の指の周辺にメニューを表示する。表示されたメニューをもう一方の手の指によって選択する。¹³⁾ においては、ユーザはオブジェクトをダブルタップすることにより、そのオブジェクトに対するポップアップメニュー

やポップアップツールバーを表示することができる。表示したそれらをタッチ操作により任意の位置、方向に移動、回転させることができる。また、Kammer らは、マルチタッチデバイス向けのリングメニューを示した⁸⁾。このリングメニューは、グローバルメニューとコンテキストメニューからなる。前者は、非利き手の5本の指のタッチパネルとの接触により表示でき、後者は、1つの指の接触点によるタップ&ホールドにより表示できる。メニューは、指の方向に合わせて出現する。Banovic らは手の形状に合わせて一部が切り取られたパイメニューを示した²⁾。このパイメニューは、親指の接触によって表示できるのとその他の指の接触によって表示できるものが用意されている。³⁾ において、ユーザは一方の手のシングルタッチによって指の周辺に表示されたパイメニューからモードを選び、もう一方の手のシングルタッチによって選んだモードによる選択操作を行う。Brandl らはユーザの利き手を考慮したパイメニューをペンの周辺に配置した⁵⁾。¹⁰⁾ はユーザがペンによって描いた軌跡に沿ってコンテキストメニューの項目を並べる。⁴⁾ は、非利き手をタッチパネル面上にホバーさせ静止させることによって表示できるツールパレットを示した。¹⁴⁾ および⁷⁾ は¹²⁾ において示された手法を用いて、タッチパネル面とその縁との境界線の任意の位置を指によってクロッシングすることにより表示できるメニューを開発した。視覚フィードバックを用いず任意の位置において用いることが可能なマーキングメニューをタッチパネル向けに応用した研究も行われている。Lepinski らは、タッチパネル面に接触した指の種類および数を活用したマーキングメニューを開発した¹¹⁾。また、Kin らは⁹⁾ において、両手の指の接触点によるストロークを用いたマーキングメニューを開発した。

本研究はメニューを任意の位置や方向に配置できる点でこれらの研究と同様であるが、両手の指の接触点の関係をメニューとのインタラクションに用いる点において異なる。

2.2 固定の位置に配置されるタッチパネル向けメニュー

⁶⁾ は半円状に切り取られた多層のパイメニューによって、階層メニューを表現する手法を示した。このパイメニューは画面下部の固定された半円状のボタンをタップすることによって表示できる。

Bailly らはメニューバーの項目を両手の指の接触点の数をを用いて選択する手法および、一方の手の指の2つの接触点と他方の手の指の接触点のストロークを用いて選択する手法を示した¹⁾。

本研究はメニューをユーザが任意の位置に表示できる点でこれらの研究と異なる。

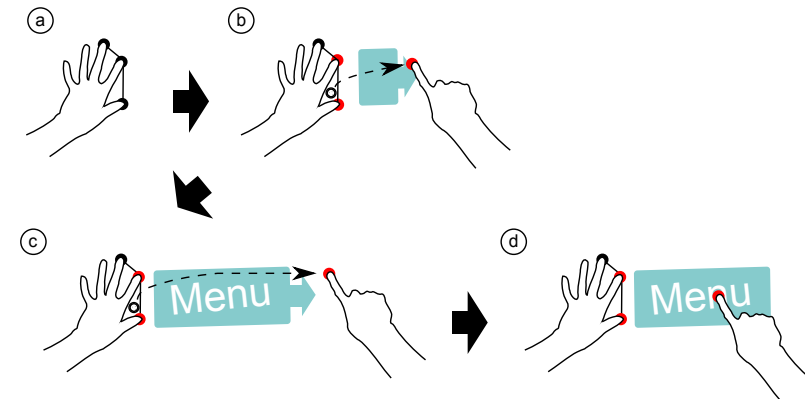


図1 メニューの表示と選択

3. 提案手法

本節では提案するメニューの表示および選択方法、削除および維持方法とその特徴を述べる。

3.1 メニューの表示および選択

ユーザがメニューを表示、選択する際のインタラクション手法を図1に時系列順に示す。図中の黒および赤の点はタッチパネル上における指の接触点を表す。まず、ユーザは一方の手の複数の指をタッチパネル面に図1aのように接触させる。システムはタッチパネル面上の接触点を認識しそれらの接触点間に仮想的な線分を生成する。以降、これをクロッシング線分と呼ぶことにする。図1における点を結ぶ線分はクロッシング線分を表す。ユーザはクロッシング線分を、図1bのようにもう一方の手の指の接触点によってクロスすることによってメニューを表示させる。ユーザは図1cのようにクロッシングを行なった指をタッチパネル面に接触させたまま、クロッシング線分から遠ざけるように移動させることによって、表示したメニューのサイズを変更することができる。また、クロッシングを行なった指をタッチパネル面から離すことによって、メニューのサイズは固定される。ユーザはこのようにして表示したメニューに対して、図1dのように離れた指を用いてタップを行い選択操作を行う。

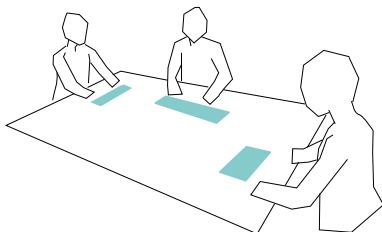


図 2 複数のユーザが任意の位置、方向、サイズのメニューを表示する

3.2 メニューの削除と維持

表示されたメニューは原則的に対応するクロッシング線分と同じ存在期間を持つ。そのため、クロッシング線分を定義する指の接触点をタッチパネル面から離れた時、メニューも同じく削除される。たとえばある指の接触点がメニューを選択していたとしても削除は行われる。この設計を活用して、メニュー項目を誤ったタップにより選択してしまった際に、タップを維持したままクロッシング線分を削除することによって選択をキャンセルするという応用が可能である。

一方、メニューをその位置に表示したままにしたい場合がある。その場合には、メニューに表示された維持ボタンを押すことによってメニューをその位置に留めることができる。また、留めたメニューを削除する場合は、もう一度維持ボタンを押す。

3.3 特徴

本提案手法の特徴を次に述べる。

ユーザによるメニュー位置、方向およびメニューのサイズ指定を組み込んだインタラクション手法

本メニューのインタラクション手法は、メニュー位置、方向およびメニューのサイズ指定を含む。ユーザはクロッシング線分の位置、方向によってメニューの位置、方向を決定することができる。また、クロッシング線分とその線分をクロスするもう片方の手の指の接触点との関係によってメニューのサイズを指定できる。すなわち複数のユーザが図 2 のようにそれぞれ任意の位置、方向、サイズのメニューを表示することができる。

従来のタッチジェスチャとの共存

クロッシング線分に対するクロッシングは従来のマルチタッチジェスチャにおいて用いられない。ゆえに、ピンチや 3、4 本指ジェスチャとも共存することが可能である。

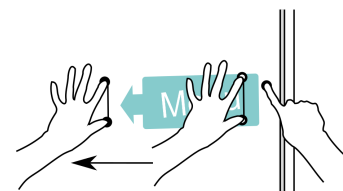


図 3 画面の端におけるメニューの表示

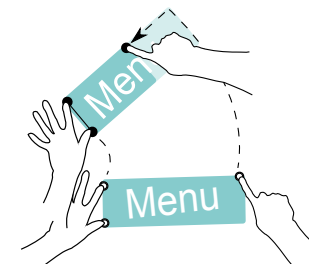


図 4 メニューの再配置

誤作動のしにくさ

クロッシング線分に対するクロッシングは偶発的に起こりづらい。ゆえに、メニューが誤って表示されることは発生しにくいと考えられる。

画面の端におけるメニューの表示

表示される位置や方向が固定であるメニューは画面の端に表示される際、メニューの一部が画面外に表示されてしまったり選択できないことがある。もしくは、画面内にメニューが表示されるように補正されることが多い。本メニューは画面端においても、図 3 のようにクロッシング線分を定義する指の接触点を移動させることによって任意のサイズのメニューを表示することができる。

3.4 メニューの再配置

一度、配置したメニューを再配置したいことがある。その場合に、表示されているメニューに対して図 4 のようにメニューを表示する際と同様のジェスチャを用いてメニューの位置、方向やメニューのサイズを変更することができる。

片手による操作

片手を用いて図 5 のようにクロッシング線分の定義およびクロッシングを行うことができる。片手のみを用いてメニューを表示すれば、より多様なメニューの使い分けを実現することができる。

4. 設計空間

本節では提案手法の設計空間について述べる。

メニューのモデル

ユーザは図 1 c、d において赤色の点によって示される指の接触点によってメニューを調

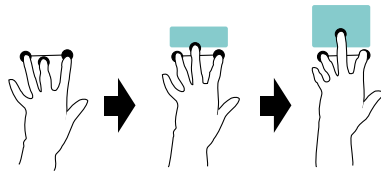


図 5 片手による操作



図 6 4つのメニューのモデル

整することができる。我々は、このように調整されるメニューに対して4つのモデルを示す。図6はそれらのモデルの概念図である。以下にそれらの各モデルについて説明する。

切り取りモデル (図 6a)

切り取りモデルは、指の接触点同士の距離に合わせてメニューの左端からメニュー項目を徐々に表示するモデルである。

引出しモデル (図 6b)

引出しモデルは、指の接触点同士の距離に合わせてメニューの右端からメニュー項目を徐々に表示するモデルである。

拡大・縮小モデル (図 6c)

拡大・縮小モデルは、指の接触点同士の距離に合わせてメニューの表示サイズを変更するモデルである。

変形モデル (図 6d)

変形モデルは、指の接触点同士の距離に合わせてメニューの項目数や項目の位置を動的に変更するモデルである。

本メニューを用いる場合は、アプリケーションに適したモデルを用いることが望ましい。例えば、本メニューをキー入力に用いる際には変形モデルを用いて、指の接触点同士の距離に合わせて図7のようにテンキーキーボードから QWERTY キーボードへ変形するものが

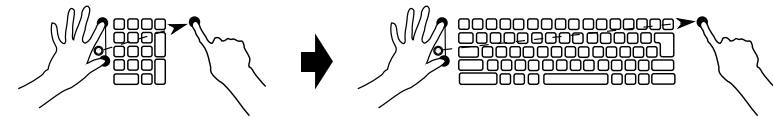


図 7 テンキーキーボードから QWERTY キーボードへの変形

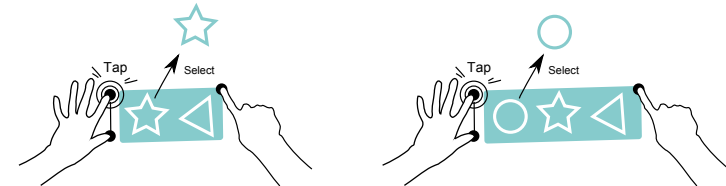


図 8 クロッシング線分を定義する指の接触点のさらなる活用

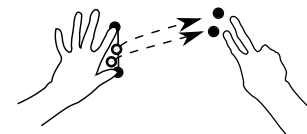


図 9 2本の指によるクロッシング

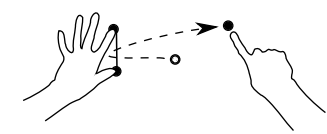


図 10 ダブルクロッシングの利用

考えられる。

クロッシング線分を定義する指の接触点のさらなる利用

クロッシング線分を定義する指の接触点をさらに利用し、メニューとのインタラクションをより多様にすることが考えられる。図8のように、クロッシング線分を定義する指の一方をタップすることによってメニューの最もクロッシング線分に近い項目を選択することが考えられる。メニューの項目を変化させながらこの操作を行うことによって、異なる項目を連続的に選択することができる。

2本の指によるクロッシングおよびダブルクロッシングの利用

図9のように2本の指によるクロッシング線分へのクロッシングを用いることが考えられる。また、図10のようにクロッシング線分を続けて2度クロスする手法であるダブルクロッシングを利用することが考えられる。これらのクロッシングによりグローバルなメニューを表示する一方、通常のクロッシングによりアプリケーションに対するメニューを表示するという用途が考えられる。

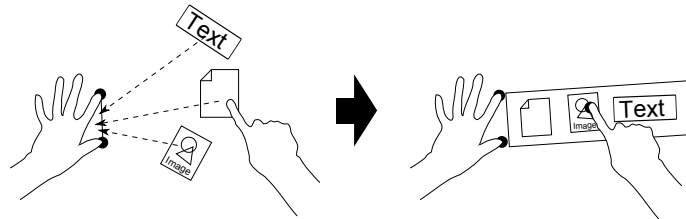


図 11 オブジェクトをドラッグした状態でのクロッシング線分へのクロッシング



図 12 複数のクロッシング線分の利用

オブジェクトをドラッグした状態でのクロッシング線分へのクロッシング

図 11 のようにオブジェクトをドラッグした状態のまま、クロッシング線分を横切る動作も操作として用いることが可能である。その用途としては、ドラッグしているオブジェクトをクリップボードに格納する操作が考えられる。なお、このようにクリップボードに格納したオブジェクトを通常のクロッシングにより表示したメニューから選択し、貼付ける。

複数のクロッシング線分の利用

近接するクロッシング線分それぞれから異なるメニューを出すことが考えられる。この手法を用いることによって、ユーザは複数のメニューを使い分けることができる。例えば図 12 のように、あるクロッシング線分から生成されるメニューをアプリケーション向けにする一方、他のクロッシング線分から表示されるメニューにはクリップボードの中身を表示するという用途が考えられる。

5. プロトタイプの実装

本節では、3 節において述べたメニューのインタラクション手法を確認するために実装し

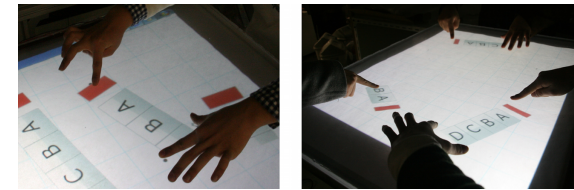


図 13 プロトタイプを使用する様子

たプロトタイプについて述べる。プロトタイプを用いて 4 節において示したメニューモデルである切り取りモデル、引出しモデル、拡大・縮小モデル、変形モデルのそれぞれを実装し、その振舞いを確認した。プロトタイプを用いている様子を図 13 に示す。

用いたプロトタイプのタッチパネル面のサイズは縦約 60cm、横約 80cm である。また、その解像度は 1024×768 px である。タッチ点認識には Community Core Vision (CCV)*¹を用い、アプリケーションの開発には Windows Presentation Foundation (WPF) を用いた。

クロッシング線分の生成手法は以下の通りである。システムはある指の接触点が連続して τ 回認識された時、距離が最も小さく閾値 λ px を超えていない他の指の接触点とクロッシング線分を生成する。 τ 回認識された後に認識処理を行う理由は複数の指がほぼ同時にタッチパネル面に接触する際、それぞれの指がシステムに認識されるまでには若干の差があるためである。また、閾値 λ px を設定するのは、あまりに離れた指の接触点同士にクロッシング線分を作ってしまうと、その線分を他の指の接触点が誤ってクロッシングしてしまう可能性があるからである。さらに λ の効果により、クロッシング線分は近接する指の接触点集合の間にのみ生成されるため、複数のユーザが同時にクロッシング線分を生成させることが可能である。プロトタイプにおいて τ および λ は発見的に求め、それぞれ 20、400 とした。

著者がプロトタイプを用いた結果、それぞれのメニューモデルを使用することができた。また、親指と人差し指の接触点間に生成されるクロッシング線分が最もクロッシングを行いやすかった。一方、その他の指の間に生成されるクロッシング線分を上手くクロッシングできないことがあった。その理由は、用いたタッチパネルの分解能が低かったために、クロッシングを行う指の接触点と、クロッシング線分を定義する指の接触点を識別できなかったからである。より分解能の細かいタッチパネルを用いることによって、クロッシングを上手く行えるようになると思われる。

*1 <http://ccv.nuigroup.com/>

また稀に正しく隣り合った指の接触点の間にクロッシング線分を生成できないことがあった。これは現在のクロッシング線分の生成手法は、ユーザが短い時間の間にそれぞれの指をタッチパネル上に接触させた場合に、上手くクロッシング線分を生成することができる一方、ユーザがそれぞれの指を大きな時間差を伴ってタッチパネル上に接触させた場合に、上手くクロッシング線分を生成することができないからである。クロッシング線分の生成手法はより洗練されたものにすべきであると考ええる。

6. まとめと今後の予定

本稿において示したメニューは、ユーザが任意の位置、方向に表示することができ、かつメニューのサイズを指定することができる。これらは、タッチパネル上において指の接触点の関係を活用することにより実現される。また、本稿では本メニューの設計空間についても述べ、プロトタイプを実装した。

今後の予定は、4節において述べた設計空間を実装すること、本メニューを用いたアプリケーションを開発することである。また、複数ユーザが本メニューを同時に用いることができることを確かめる。さらに本メニューの特徴として述べた誤動作のしにくさについて評価を行い検証したい。メニューを表示する際のストロークの形状に沿ってメニュー項目を配置することにより人間工学に基づくメニュー配置も可能であると考えているので、その検討を進めたい。今回はテーブルトップ型のタッチパネルを用いてプロトタイプを実装したが、本メニューはウォール型の大型タッチパネルに対しても適用可能であると考えているのでウォール型の大型タッチパネルを用いた実装も進めたい。

参 考 文 献

- 1) G.Bailly, E.Lecolinet, and Y.Guiard. Finger-count & radial-stroke shortcuts: 2 techniques for augmenting linear menus on multi-touch surfaces. In *Proceedings of the 28th international conference on Human factors in computing systems*, CHI '10, pp. 591–594, 2010.
- 2) N.Banovic, F.C.Y. Li, D.Dearman, K.Yatani, and K.N. Truong. Design of uni-manual multi-finger pie menu interaction. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '11, pp. 120–129, 2011.
- 3) H.Benko, A.D. Wilson, and P.Baudisch. Precise selection techniques for multi-touch screens. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, pp. 1263–1272, 2006.
- 4) A.Bragdon, R.DeLine, K.Hinckley, and M.R. Morris. Code space: touch + air gesture hybrid interactions for supporting developer meetings. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '11, pp. 212–221, 2011.
- 5) P.Brandl, J.Leitner, T.Seifried, M.Haller, B.Doray, and P.To. Occlusion-aware menu design for digital tabletops. In *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, CHI EA '09, pp. 3223–3228, 2009.
- 6) T.Hesselmann, S.Flöring, and M.Schmitt. Stacked Half-Pie menus: navigating nested menus on interactive tabletops. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '09, pp. 173–180, 2009.
- 7) K.Hinckley, K.Yatani, M.Pahud, N.Coddington, J.Rodenhouse, A.Wilson, H.Benko, and B.Buxton. Pen + touch = new tools. In *Proceedings of the 23rd annual ACM symposium on User interface software and technology*, UIST '10, pp. 27–36, 2010.
- 8) D.Kammer, F.Lamack, and R.Groh. Enhancing the expressiveness of fingers: multi-touch ring menus for everyday applications. In *Proceedings of the First international joint conference on Ambient intelligence*, AmI'10, pp. 259–264, 2010.
- 9) K.Kin, B.Hartmann, and M.Agrawala. Two-handed marking menus for multitouch devices. *ACM Trans. Comput.-Hum. Interact.*, 18:16:1–16:23, 2011.
- 10) D.Leithinger and M.Haller. Improving menu interaction for cluttered tabletop setups with user-drawn path menus. In *Second Annual IEEE International Workshop on Horizontal Interactive Human-Computer Systems*, TABLETOP '07, pp. 121–128, 2007.
- 11) G.J. Lepinski, T.Grossman, and G.Fitzmaurice. The design and evaluation of multitouch marking menus. In *Proceedings of the 28th international conference on Human factors in computing systems*, CHI '10, pp. 2233–2242, 2010.
- 12) V.Roth and T.Turner. Bezel swipe: conflict-free scrolling and multiple selection on mobile touch screen devices. In *Proceedings of the 27th international conference on Human factors in computing systems*, CHI '09, pp. 1523–1526, 2009.
- 13) C.Shen, M.S. Hancock, C.Forlines, and F.D. Vernier. CoR²Ds. In *CHI '05 extended abstracts on Human factors in computing systems*, CHI EA '05, pp. 1781–1784, 2005.
- 14) R.Zeleznik, A.Bragdon, F.Adeputra, and H.-S. Ko. Hands-on math: a page-based multi-touch and pen desktop for technical work and problem solving. In *Proceedings of the 23rd annual ACM symposium on User interface software and technology*, UIST '10, pp. 17–26, 2010.