筑波大学大学院博士課程

システム情報工学研究科修士論文

レーザポインタを用いたストロークによる 大画面向けインタラクション手法

張進

(コンピュータサイエンス専攻)

指導教員 田中二郎

2008年3月

近年、大画面環境向けの新しい入力方法として、レーザポインタを用いたインタフェース が提案されている。本研究は先行研究を踏まえ、既存問題を解決するためにレーザポインタ を用いたストロークによる大画面向けインタラクション手法を提案する。

提案手法はレーザポインタを用いて、値を調節する操作に重きを置く。値の具体例はズー ミングレート、時間、位置、音量などである。値の調節は精確な調節、おおざっぱな調節、低 速な調節と高速な調節、いくつかの方式がある。既存インタラクション手法は値の調節には 不向きであり、実行時間が遅いなどの問題点がある。提案手法で用いるレーザポインタを用 いたストロークとは、レーザスポットがスクリーン上で移動した軌跡である。ストロークか らレーザスポットの移動ベクトルとレーザスポットの移動角度を得られる。移動ベクトルと 移動角度の変わりを利用することにより、値の調節が可能とする。移動ベクトルの長さ及び 移動角度の変化の速度により、低速な調節と高速な調節が可能となる。さらに、ストローク の移動角度の変わりによって、円形のポップアップメニュを操作する手法を提案する。円形 メニューによって、おおざっぱな調節だけではなく、精確な調節も可能となる。

本研究は、提案手法を実装し、また、提案手法の応用例として地図ビューアーとミュージッ クプレイヤーを作成した。提案手法の有効性を検証するために、実行時間などの基本的な性 能を測定する評価実験を行った。実験の結果により、提案手法は値の調節には実行時間が速 く、使いやすい手法であることが明らかになった。

概要

目 次

第1章	はじめに	1
第2章 2.1 2.2	既存インタラクション手法の問題点 レーザスポットを止めることによるインタラクション手法 クロッシング手法とエンサークリング手法 2.2.1 クロッシング手法 2.2.2 エンサークリング手法	3 3 4 4 5
第3章 3.1	レーザポインタを用いたストロークによるインタラクション手法 レーザポインタを用いたストローク	6 6 7 7
3.2 3.3	値の調節の説明 提案手法	8 8 9 11 11
第4章 4.1 4.2 4.3 4.4	実装 システムの構成	15 15 16 17 18
第5章 5.1	アプリケーション 地図ビューアー 5.1.1 パン操作 5.1.2 ズーミング操作 5.1.3 回転操作 5.1.4 操作の起動法	 19 19 19 20 20 20 20 20
5.2	ミュージックブレイヤー	21 22 23

	5.2.3 ミュージックファイルの再生	23
	5.2.4 音量の調節	23
	5.2.5 再生中の時間移動	25
第6章	評価実験	26
6.1	地図ビューアーのパン操作についての評価	26
	6.1.1 実験方法	27
	6.1.2 実験結果	29
	6.1.3 考察	29
6.2	円形メニューによる数字の調整についての評価..............	29
	6.2.1 実験方法	31
	6.2.2 実験結果	33
	6.2.3 考察	33
第7章	関連研究	34
7.1	大画面環境とレーザポインタを用いたインタフェース	34
7.2	ストロークによるインタラクション手法	34
第8章	まとめ	36
	謝辞	37
	参考文献	38

図目次

1.1	システムの構成図.................................	1
2.1	止めることによるマウスクリックをエミュレートする	3
2.2	クロツンンク手法・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	4
2.3		4
2.4	4つの基本クロッシンク	5
3.1	レーザポインタを用いたストローク........................	6
3.2	レーザスポットの移動ベクトル	7
3.3	指定された点を回転中心とするレーザスポットの移動角度	7
3.4	スクロールバーによる閲覧領域の調節..................	8
3.5	スライドバーによる音量の調節	8
3.6	ドロップダウンメニューによる日付の調節	9
3.7	スライドバーによる再生中の時間移動.........................	9
3.8	点灯位置による画面表示領域を四等分する	10
3.9	点灯位置による画面表示領域を六等分する	10
3.10	フォントを調節する例	10
3.11	移動ベクトルを分解する例.................................	11
3.12	8 等分の円形メニュー (a)、6 等分の円形メニュー (b)、4 等分の円形メニュー (c)	12
3.13	円形メニューを利用してファイルリストを閲覧する例	12
3.14	円形メニューを利用して数値を調節する例	13
3.15	オクタントの選択	13
3.16	レーザスポットがオクタントの外に出ても選択は有効である	13
3.17	レーザスポットが円形メニューの中心から離れると、低速な操作が可能である	14
4.1	システムの構成	15
4.2	露光過度の画像(a)、減光用フィルターとして使われる緑の下敷き(b、c)、減	
	光用フィルターを装着した後キャプチャされる画像 (d)	16
4.3	レーザスポット	17
4.4	画面表示領域の四隅を順番にポイントしてキョリブレーションを行う	17
4.5	円形メニューを操作する時右回りと左回りの判断	18
5.1	パン操作....................................	19

5.2	ズーミング速度の選択	20
5.3	ズーミング操作	21
5.4	操作の起動法	21
5.5	ミュージックプレイヤーの状態遷移図...........................	22
5.6	再生ファイルの選択 (a)、クロッシング操作とミュージックファイルの選択操	
	作を統合する操作法 (b)	23
5.7	再生ファイルを選択する状態では、レーザスポットで左側の縁を横切ると、選	
	択されたミュージックファイルを再生する (a)、再生状態のイメージ (b)	24
5.8	音量調節状態のイメージ (a)、クロッシング操作と音量調節の操作を統合する	
	操作法 (b)	24
5.9	再生中の時間を移動する状態のイメージ(a)、1%の選択肢が選択される(b)	25
6.1	実験環境	26
6.1 6.2	実験環境....................................	26 27
6.16.26.3	実験環境....................................	26 27 28
6.16.26.36.4	実験環境	26 27 28 29
 6.1 6.2 6.3 6.4 6.5 	実験環境 実験で使用したシステムの画面 ターゲットの位置 平均移動時間 距離による平均移動時間	26 27 28 29 30
 6.1 6.2 6.3 6.4 6.5 6.6 	実験環境実験で使用したシステムの画面ターゲットの位置平均移動時間正離による平均移動時間ターゲットの角度による平均移動時間	26 27 28 29 30 30
 6.1 6.2 6.3 6.4 6.5 6.6 6.7 	実験環境実験で使用したシステムの画面ターゲットの位置平均移動時間距離による平均移動時間ターゲットの角度による平均移動時間使えない状態の Number Interactor	26 27 28 29 30 30 31
 6.1 6.2 6.3 6.4 6.5 6.6 6.7 6.8 	実験環境	26 27 28 29 30 30 31 32
 6.1 6.2 6.3 6.4 6.5 6.6 6.7 6.8 6.9 	実験環境実験で使用したシステムの画面ターゲットの位置平均移動時間距離による平均移動時間クーゲットの角度による平均移動時間使えない状態の Number Interactor使える状態の Number InteractorNumber Interactor によるデザインされた円形メニュー	26 27 28 29 30 30 31 32 32

表目次

6.1	ターゲットの位置									•																2	28
-----	----------	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	----

第1章 はじめに

近年、プロジェクターによってコンピュータの画面を大型スクリーンに投影して利用する 場合が多くなってきている。例えば、講義をスクリーンに投影して授業を行う例が挙げられ る[12]。コンピュータを操作するためのデバイスはマウス・キーボードが一般的であるが、こ のような大画面環境では、ユーザがコンピュータの前に釘付けにされなく、直接大画面に向 かってコンピュータを操作する手法が望ましい。大画面環境に向け、ジェスチャー、音声な どを利用して、コンピュータを操作するという新たな操作法について種種の研究が試みられ ている[1][2]。しかし、ジェスチャーによるインタラクション手法は手に何かしらのデバイス を装着したり、ユーザの移動がスクリーンから近い範囲に制限られる問題がある。音声によ るインタラクション手法は多様な操作ができるように、ジェスチャーと併用する場合が多い。

Kirstein[3] らは 1998 に始めてレーザーポインタを用いた大画面向けインタフェースを提案 した。図 1.1 はその典型的なシステム構成である。スクリーン上のレーザポインタの光点をカ メラでキャプチャし、その位置をリアルタイムに検出してカーソルの移動やコンピュータの 操作に利用する。



図 1.1: システムの構成図

大画面環境で、レーザポインタを用いたインタフェースには下記のような利点がある。

- ユーザはコンピュータの前に釘付けにされなく、直接大画面に向かってコンピュータを 操作するので、操作感が良い。一方、マウスやキーボードといった従来の入力インタ フェースでは、ユーザはコンピュータの前にいなければコンピュータの操作ができない といった制約がある。
- 画面を遮ることなく離れた位置から大画面を指すことによってコンピュータを操作する ことができる。そして、レーザポインタは手元のわずかな動きにより、容易に広い範囲 を高速に動かすことが可能であるので、画面の大きさに制限されなく、大画面上に手が 届かない場所でも操作ができる。例えば、大画面環境を利用して授業を行う例では[12]、 先生は教室のどこでもレーザポインタを用いてコンピュータを遠隔操作することがで きる。
- 一本のレーザポインタで複数のコンピュータを操作することが可能である。

本研究の目的

本研究は大画面向けレーザーポインタを用いたインタフェースに注目し、斬新且つ有効な インタラクション手法を提案する。我々は様々なアプリケーションを利用する時、値を調節す る操作が多い。例えば、メディアプレイヤーを利用する場合に、音量調節の操作、地図ビュー アーを利用する場合、ズーミングの操作など。本研究では、地図ビューアーでは表示領域の 調節即ちスクロールバーによるパン操作も値の調節の一種と見なす。しかし、既存のレーザ ポインタを用いたインタラクション手法は値の調節操作には不向きであり、実行時間が遅い などの問題がある。本研究では、先行研究を踏まえ、レーザポインタを用いたストロークに よるインタラクション手法を提案し、実装する。評価実験により、値の調節操作に対し、本 手法の有効性を示すことを目的とする。

本論文の構成

第2章では既存のレーザポインタを用いたインタラクション手法を考察し、値を調節する 操作には既存手法は不向きであり、実行時間が遅いことが明らかになった。これを解決する ために、第3章ではレーザポインタを用いたストロークによるインタラクション手法を提案 する。さらに、第4章では、システムの実装について述べ、第5章では、提案手法の適用例 として地図ビューアーとミュージックプレイヤーを示し、使い方について説明する。第6章 において、実行時間などの基本的な性能を測定するための評価実験を行い、結果について考 察する。第7章では関連研究を挙げ、最後に第8章でまとめる。

第2章 既存インタラクション手法の問題点

本章では関連研究を考察し、既存のレーザポインタを用いたインタラクション手法の問題 点について述べ、本研究の目的をより明らかにする。

2.1 レーザスポットを止めることによるインタラクション手法

レーザポインタの点灯、消灯及びレーザスポットの移動がマウスプレス、マウスリリース 及びマウスカーソルの移動と対応付けられれば便利であるが、[7] により、ユーザが思い通り の場所でレーザポインタを点灯、消灯できないことが分かる。そのため、レーザポインタの 点灯、消灯の動作がマウスプレス、マウスリリースと対応できない。

Kirstein[3] らのシステムでは、レーザスポット¹を同じ場所に一定時間止めることによるマウスクリックをエミュレートするインタラクション手法を提案した。図 2.1 はマウスクリックをエミュレートして、ボタンをクリックする例である。またレーザポインタを用いたインタフェースに関する研究 [4][6] では、同様の手法を利用している。



図 2.1: 止めることによるマウスクリックをエミュレートする

しかし、この手法は各操作の開始に1秒ぐらい止める時間を要するために高速に操作でき ない問題がある。

また、レーザポインタが手持ちデバイスなので、手振れの影響を受ける。そのため、レー ザスポットを同じ場所に止め続けるのは難しい。Myers[7]らは、レーザスポットを同じ場所 に止めることにおいてレーザポインタのパフォーマンスを評価した。スクリーンの広さが8

¹レーザスポット:レーザビームがスクリーン上に照射された際に形成される光点

フィートで、解像度が1024 × 768 ピクセルにて、被験者がスクリーンから10フィートのと ころに実験を行う場合は、予測フィルタリング後であっても、最高でも±2ピクセルの精度 しかないという結果を得た。一方、マウスを使う場合は、手振れの影響を全然受けない。そ のため、マウス・ベースシステムで使われている小さなメニュー、アイコンなど操作対象は レーザポインタ・ベースシステムに対して不向きであることが分かる。

2.2 クロッシング手法とエンサークリング手法

レーザスポットを止めることによるインタラクション手法の問題点を解決するために、レー ザスポットの動きを利用するインタラクション手法が提案された。志築[8]らはクロッシング 手法、久松[9][10]らはエンサークリング手法を提案した。クロッシング手法は(図 2.2)レー ザスポットでターゲットを横切ることで、コマンドを実行する。エンサークリング手法は(図 2.3)レーザスポットでターゲットを囲むことで、対象を選択する。





図 2.2: クロッシング手法

図 2.3: エンサークリング手法

2.2.1 クロッシング手法

志築らのシステムでは、クロスするべきターゲットを画面表示領域の縁とした。縁はクロ スするターゲットとして十分大きく、システムのユーザーにとって高速な操作が容易である。 この手法では、図 2.4 に示す 4 種類の基本的なクロッシングを用いることとした。

- In-out 画面表示領域の内から画面表示領域の外へ移動するクロッシング
- Out-in 画面表示領域の外から画面表示領域の内へ移動するクロッシング
- In-out-in In-outとOut-inを連続して行うクロッシング
- Out-in-out Out-in と In-out を連続して行うクロッシング

これらの基本クロッシングと、画面の四辺の組み合わせにそれぞれコマンドを割り当てる ことができる。



図 2.4:4 つの基本クロッシング

2.2.2 エンサークリング手法

この手法では、ユーザはアイコンやボタン、サムネイルなどの GUI オブジェクトをレーザー スポットで囲むことで、オブジェクトの選択を行う (図 2.3)。レーザーポインタを用いたシス テムでは、円を描くような動作は比較的容易であり、この手法は、レーザスポットを一定時 間止めることによるクリック操作に比べて容易にアイコンなどのオブジェクトを選択できる。 そして、この手法の操作速度も速い。レーザスポットを一定時間止めることによるクリック 操作では、操作速度は既定の時間に大きく左右される。クリックと見なすのに要する時間を 短くすることで操作速度を早くすることはできるが、引き換えにシステムの誤認識率が高く なると考えられる。円を描くという、特徴的な動作であれば、誤認識を少なくかつ比較的高 速にターゲットの選択ができる。

これらの手法は動きを利用するので手振れの影響が軽減でき、さらに高速かつ連続的な操 作も実現できる。クロッシングとエンサークリング手法はマウスクリックとダブルクリック にあたる。しかし、これらの手法だけでは、様々な操作にはまだ不十分である。例えば、パ ン、ズーミング、音量の調節など値を調節する操作には難しい。ここで、値の具体例は位置、 ズーミングレート、音量、時間、日付などである。値の調節は精確な調節、大雑把な調節、低 速な調節と高速な調節、いくつかの方式がある。

値を調節するために、Olsen[4] らは専用アプリケーションに対する Number Interactor、Date Interactor、Time Interactor、Enumeration Interactor などをデザインした。しかし、これらのインタラクターの選択、調節には、レーザスポットを一定時間止める必要があるので、前述のように、高速な操作ができなく、手振れの影響も受けやすいので、操作性を損う。

第3章 レーザポインタを用いたストロークによ るインタラクション手法

本章では、値を調節する操作には既存インタラクション手法の問題点を解決するために、高 速且つ使いやすいインタラクション手法を提案する。本研究では、値の具体例は、時間、位 置、ズーミングレート、角度、音量などである。レーザスポットの動きを利用するクロッシ ング手法とエンサークリング手法とは手振れの影響が軽減でき、高速かつ連続的な操作もで きるという利点がある。この利点を受け継ぐために、本研究はレーザスポットの動きを利用 するインタラクション手法を提案する。

3.1 レーザポインタを用いたストローク

クロッシング手法とエンサークリング手法とはレーザスポットの動きを利用して、ターゲットを横切ったり、囲んだりする手法である。本研究はレーザスポットの動き自体の情報に注目する。本研究では、レーザスポットがスクリーン上で移動した軌跡をレーザポインタを用いたストロークと呼ぶ(図 3.1)。以下はレーザポインタストロークと略称する。本研究ではレーザポインタストロークと略称する。本研究ではレーザポインタストロークから次に述べる二つの情報を得る。



図 3.1: レーザポインタを用いたストローク

3.1.1 レーザスポットの移動ベクトル

一つはレーザスポットのポイント開始位置から現在位置までの移動ベクトルである(図 3.2)。



図 3.2: レーザスポットの移動ベクトル

3.1.2 レーザスポットの移動角度

もう一つは、指定された回転中心に対して、レーザスポットのポイント開始位置と現在位置とが成す移動角度である(図 3.3)。



図 3.3: 指定された点を回転中心とするレーザスポットの移動角度

3.2 値の調節の説明

本研究では様々な値の調節を扱う。文章を閲覧する場合には、スクロールバーをスクロールして閲覧領域の調節(図 3.4)、スライドバーをスライドして音量の調節(図 3.5)、カレンダーではドロップダウンメニューによる日付の調節(図 3.6)、及びメディアプレイヤーでは再生中の時間移動(図 3.7)などすべて値を調節する操作と見なす。



図 3.4: スクロールバーによる閲覧領域の調節



図 3.5: スライドバーによる音量の調節

そのうち、閲覧領域と音量の調節はおおざっぱな調節であり、操作の誤差にはあまり気に しない。日付の調節は精確な調節であり、誤差を許容しない。そして、閲覧領域と音量の調 節には、低速な調節と高速な調節がある。

3.3 提案手法

提案手法はレーザポインタストロークから連続にレーザスポットの移動ベクトルと移動角 度を取得し、ベクトルと角度の変わりを利用することにより、値の調節が可能とする。







図 3.7: スライドバーによる再生中の時間移動

3.3.1 手法1

移動ベクトルは長さと方向という二つの要素を含む。この手法では、ベクトルの方向によっ て調節の対象を選択し、ベクトルの長さによって選択した対象を調節する。同じような操作手 法はマウスを用いて円形メニューを操作する場合にも使われている[16]。レーザポインタが 画面表示領域の内に点灯する際、レーザスポットのポイント開始位置を中心に、画面表示領域 を分割する。分割された画面表示領域は大きなメニューと見なすことができる(図 3.8、3.9)。 例えば、フォントを調節する場合は(図 3.10)、レーザスポットが点灯位置から離れると、移 動ベクトルの方向によるフォントのスタイルを選択し、ベクトルの長さによるフォントのサ イズを調節する。

Marking Menu はスタイラス或いはマウスを用いて操作するメニューである。Marking Menu では、ストロークを描くとその方向に対応したコマンドが実行される。そして、ストローク の長さ、方向の変わりを利用することによる複数階層メニューを表現可能である。[11]では、 4項目、8項目、12項目の Marking Menu のそれぞれで、1~4階層までのメニューに到達する までの時間とエラー率を調べている。到達時間は項目数が増えると急激に増加する傾向があ り、エラー率は、8項目3階層以上、12項目2階層以上において、急激に高くなる傾向があ る。この結果から、メニューの項目の数が多くなると、選択のエラー率が高くなることが分 かる。レーザポインタの操作精度がマウス、スタイラスより悪いことを考慮し、提案手法で は画面表示領域が四等分或いは六等分とした。レーザポインタストロークはマウスストロー ク、スタイラスストロークよりサイズが大きいので、複数階層メニューに不向きである。



図 3.8: 点灯位置による画面表示領域を四等分する



図 3.9: 点灯位置による画面表示領域を六等分する



図 3.10: フォントを調節する例

3.3.2 手法2

移動ベクトルを縦方向と横方向に分解することができる(図 3.11)。そのため、同時に関連 がある二つの値を調節することが可能となる。例えば、地図ビューアーにおいて、表示位置 を調節する場合は、移動ベクトルを分割することにより、同時に縦方向の位置と横方向の位 置を調節することができる。各方向の位置の調節速度は、各方向の移動ベクトルの長さによ る変わる。そのため、高速な調節も、低速な調節も容易に行える。そして、一つのストロー クにより、縦と横の二つのスクロールバーの動作を統合できるので、便利な手法であると考 えられる。



図 3.11: 移動ベクトルを分解する例

3.3.3 手法3

前述の二つの手法はレーザスポットの移動ベクトルの長さにより、値の調節を行う。しか し、レーザポインタが手持ちデバイスなので、ユーザが移動ベクトルの長さを精確にコント ロールすることが難しい。そのため、これらの手法はおおざっぱな調節にのみ有効である。日 付の調節、またメディアプレイヤーでは、再生中の時間を前に10秒を進みたい場合はどうす るか。これに対して、本研究はレーザスポットの移動角度による円形ポップアップメニュー を操作するインタラクション手法を提案する。レーザポインタの操作精度が悪いことを考慮 し、本研究では最大8等分の円形メニューを使う(図 3.12)。

レーザスポットで円形メニューを選択しやすいように、円形メニューのサイズはできるだけ大きめにデザインしたほうが良いと考えれるが、画面表示領域の内容が遮られる。そのため、本研究では、画面の解像度が1024 × 768 ピクセルの場合は、円形メニューの直径は300 ピクセルに設定されている。

以下には、円形メニューを利用する二つの例を挙げ、それぞれ円形メニューの操作方法に ついて説明する。

図 3.13 は 8 等分の円形メニューを利用してファイルリストを閲覧する例である。この例で は、画面表示領域が円形メニューにより八つの部分に分けられる。レーザポインタが画面表



図 3.12:8 等分の円形メニュー(a)、6 等分の円形メニュー(b)、4 等分の円形メニュー(c)

示領域内に点灯すると、レーザスポットのポイント開始位置と対応する円形メニューのオク タントが選択される。オクタントを選択し、右に回すと、ファイルリストが上へスクロール し、左に回すと、ファイルリストが下へスクロールする。8等分の円形メニューなので、一周 回すと、8つのファイルと対応する。円形メニューをぐるぐる回すと、長いファイルリストで も閲覧することが可能である。



図 3.13: 円形メニューを利用してファイルリストを閲覧する例

図 3.14 は 8 等分の円形メニューを利用して数値を調節する例である。円形メニューでは四 つのオクタントしか選択できない。1、10、100、1000 は調節の速度を表示する。ファイルリ ストを閲覧する例と同じような方法で選択したいオクタントを選択できれば、便利であるが、 [7] により、ユーザが思い通りの場所でレーザポインタを点灯できない。そのため、この例で は、まず円形メニューの外にレーザポインタを点灯し、選択したいオクタントにレーザスポッ トを移動させ、レーザスポットがオクタント内に入ると、このオクタントが選択される(図 3.15)。オクタントが選択されると、レーザスポットがオクタントの外に出ても、この選択は 依然有効である(図 3.16)。レーザポインタを消灯すると、選択は無効になる。例えば、レー ザスポットで1という選択肢を選択したら、他の選択肢が消え、一周右に回すと、8 をプラス し、一周左に回すと、8をマイナスする。



図 3.14: 円形メニューを利用して数値を調節する例



図 3.15: オクタントの選択



図 3.16: レーザスポットがオクタントの外に出ても選択は有効である

回転操作では、速度vと角速度との間に次式のような関係がある。

$$v = r \, \mathbf{x} \tag{3.1}$$

ここでrは回転操作の半径である。この式により、レーザスポットが円形メニューの中心に近づくと、高速なメニュー操作が可能であり、円形メニューの中心から離れると、低速であるが、精確なメニュー操作が可能である(図 3.17)。例えば、ファイルリストを閲覧する例では、

高速に進みたい場合は、レーザスポットを円形メニューの中心に近づけ、ファイルリストを はっきり見たい場合は、レーザスポットを円形メニューの中心から離す。



図 3.17: レーザスポットが円形メニューの中心から離れると、低速な操作が可能である

本研究で、提案する三つのインタラクション手法はマウスを用いて操作するドロップダウンメニュー、スクロールバー、スライドバー、ファイルリストなどに対応できる。

第4章 実装

4.1 システムの構成

図 4.1 に示すように、本研究で作ったシステムはコンピュータ、USB カメラ、プロジェクタ、 スクリーン及びレーザポインタで構成される。ユーザはレーザポインタでスクリーン上を指 し示す。スクリーンが USB カメラの撮影領域に収まるように、スクリーンから 3m のところ にカメラを設置し、スクリーンを撮影する。撮影した映像を、ビットマップ画像としてキャ プチャする。コンピュータによって、キャプチャした画像を解析し、レーザスポットの座標 を算出する。

コンピュータのメモリは 2GB で、CPU は Intel 社の Core2 T7600 @ 2.33GHz である。レー ザーポインタはカメラ視認性の高いグリーンレーザーポインタを用いた。波長は 532nm で、 出力は 5mW である。カメラは logitech 社の QuickCam Pro 4000 を用いて、フレームレートは 最大 30fps で、画像数は 30 万である。システムの実装には、Java と Java Media Framework を 利用している。



図 4.1: システムの構成

4.2 レーザスポットの検出

USB カメラの解像度が160×120 ピクセル、320×240 ピクセル、640×480 ピクセルなど いくつかある。解像度を大き目に設定すれば、画像の解析に時間がかかるので、画像を解析 する処理が撮影の進行に追いつかなくなる場合には、キャプチャされたフレームが解析しな いまま捨てられる。そのため、高速にレーザーポインタを動かしている場合、失われるレー ザスポットの軌跡は長くなる。一方、解像度を小さめに設定すれば、算出のレーザスポットの 座標の精度が低く、様々な操作で誤操作を招き易い。以上のことを考慮し、本システムでは、 160 × 120 ピクセルの低解像度と 640 × 480 ピクセルの高解像度を使わず、USB カメラの解 像度を 320 × 240 ピクセルに設定している。モノクロの画像の処理が速いので、カメラはモ ノクロ画像を撮影する。そして、カメラのフレームレートを15fps、露光時間を1/30秒に設定 している。プロジェクタの投影が明るいので、1/30秒の露光時間で、露光過度の問題がある (図 4.2(a))。露光過度の場合には、キャプチャされた画像が真っ白になり、レーザスポットを 検出できない。露光時間を1/50秒、1/100秒に設定すれば、露光過度の問題を解決できるが、 レーザスポットが暗くなるのでうまく検出されない。本システムで使う安価な USB カメラで この問題を絞りでは完全に解決できないので、減光用フィルターを装着する(図4.2(c))。本研 究で用いた減光フィルターは四枚の緑の下敷きである(図4.2(b))。この減光フィルターを装着 することで、緑のレーザスポットのほか、プロジェクタの投影の光が大分削減され、レーザ スポットのみ撮影されるようになる (図 4.2(d))。

カメラにより撮影した映像をビットマップ画像としてキャプチャする。そのビットマップ 画像を全画像走査し、もっとも明るい点をレーザスポットとして検出する。図 4.3 に示すよう に、レーザスポットがいくつかピクセルを含む。本研究はレーザスポットの重心座標を算出 して、レーザスポットの座標として使っている。



図 4.2: 露光過度の画像 (a)、減光用フィルターとして使われる緑の下敷き (b、c)、減光用フィルターを装着した後キャプチャされる画像 (d)



図 4.3: レーザスポット

4.3 キョリブレーション

レーザスポットにポイントされるスクリーン上の点に対応するコンピュータディスプレイ 座標を算出するために、カメラの座標をコンピュータディスプレイの座標に対応付ける必要 がある。そのため、システムを立ち上げると、まずキョリブレーションを行う。図 4.4 に示す ように、画面表示領域の四隅を左上、左下、右下、右上の順にポイントする。この四つの点 はコンピュータディスプレイ上の四隅に対応することになるので、射影変換によるカメラの 座標をコンピュータディスプレイの座標に対応付けることができる。射影変換を用いた手法 を実装するに当たっては Sukthankar らの研究 [6] を参考にした。

射影変換による補正を行うことで、画面のどの位置に於いても同様の精度でレーザースポットを検出できるようになった。また、カメラを画面に対して真正面に配置する必要もなく、表示画面が収まりさえすれば斜め方向など、任意の位置に配置することもでき、設置が容易になった。



図 4.4: 画面表示領域の四隅を順番にポイントしてキョリブレーションを行う

4.4 円形メニューを操作する時右回りと左回りの判断

図 4.5(a) はレーザスポットで 8 等分の円形メニューの一つのオクタントを選択してから、一 周回したレーザスポットの軌跡を示す。その軌跡が USB カメラにキャプチャされ、レーザス ポットの重心座標を算出すれば、いくつかの点の座標を得ることができる (図 4.5(b))。点の位 置により、円形メニューに対する操作が右回りか左回りかを判断する必要がある。図 4.5(c) に 示すように、現在レーザスポットが位置するオクタントを 1 とする。次回取得するレーザス ポットの座標が 2、3、4 の領域にあれば、右回りと見なす。次回取得するレーザスポットの座 標が 6、7、8 の領域にあれば、左回りとみなす。次回取得するレーザスポットの座 標が 5 の 領域にあれば、右回りか左回りか判断できないが、これまでの操作の傾向により、判断でき る。例えば、これまでの操作は右回りの場合は、右回りと見なす。これまでの操作は左回り の場合は、左回りと見なす。実際右回りの場合でも、次回取得するレーザスポットの座標が 6、7、8 の領域にある可能性もある。しかし、カメラのフレームレートが 15fps に設定してい るので、1 秒 7 周半以上の速度で回らないと、そういう状況にはならない。即ち、乱暴な操作 ではない限り、この方法で右回りか左回りかを判断できる。



図 4.5: 円形メニューを操作する時右回りと左回りの判断

第5章 アプリケーション

5.1 地図ビューアー

地図ビューアーで実行可能な操作はパン、ズーミング、回転の三種類がある。そのうち、パン操作は提案手法2を利用し、ズーミング操作は提案手法1を利用し、回転操作は移動角度を利用して円形メニューの操作手法と似ている。

5.1.1 パン操作

移動ベクトルを縦方向と横方向に分解する。地図を移動ベクトルの反対方向へスクロール させる。縦(横)方向のスクロール速度と縦(横)方向移動ベクトルの長さの関係は次式とする。

$$speed = constant \times length$$
 (5.1)

移動ベクトルが分かりやすいように、画面上で直線を描画することで移動ベクトルを表示する(図 5.1)。この操作は一つのストロークにより、縦と横の二つのスクロールバーの動作を統合できる。



図 5.1: パン操作

5.1.2 ズーミング操作

レーザポインタが点灯する際、レーザスポットのポイント開始位置を中心に、地図の表示 領域を四つの部分に分ける(図 5.2)。各部分がズーミングの速度に対応する。レーザポインタ の点灯位置を中心として、小さな白色の円を表示する(図 5.3)。レーザスポットがこの円の内 に移動すれば、拡大も縮小もしないが、移動ベクトルの方向によるズーミングの速度を選択 することができる。パン操作と同じように、画面上で直線を描画することで移動ベクトルを 表示する。そして、ズーミングの速度が変わると、線の色が変わる。例えば、高速な拡大の場 合には、黄色の線で表示する。レーザスポットが白色の円の外に出る際に、移動ベクトルの 方向によるズーミングの速度を決める。即ち、レーザスポットが白色の円の外に出ると、移 動ベクトルの方向が変わってもズーミングの速度を変更することができない。ズーミングの 速度を決めると、移動ベクトルの長さによりズーミングレートの調整が可能である。



図 5.2: ズーミング速度の選択

5.1.3 回転操作

地図の中央に対して算出したレーザスポットの移動角度により、地図を回転する。

5.1.4 操作の起動法

地図ビューアーでは Out-In のクロッシングをトリガーとして使うことで、前の操作を終わ らせ、次の操作を行う。レーザスポットでスクリーン上地図表示領域の縁を横切ると、クロッ シングとする (図 5.4)。例えば、レーザスポットで地図表示領域の左側の縁を横切ると、前の 操作を終わらせ、パン操作を行う。システムがクロッシングを検出すると、beep でユーザに フィードバックする。クロッシングの直後に、レーザスポットが地図表示領域の縁周辺にあ るので、レーザポインタをつけたまま地図ビューアーの操作がすぐ始まったら、操作しにく



図 5.3: ズーミング操作

い。というわけで、想定する操作はクロッシングの直後にレーザポインタを消して、地図表 示領域の真ん中あたりへ移動して、もう一回つけて、地図ビューアーの操作が始まるの流れ である。



図 5.4: 操作の起動法

5.2 ミュージックプレイヤー

ミュージックプレイヤーで実行可能な操作は再生ファイルの選択、音量の調節、及び再生 中の時間移動の三種類がある。三つの操作は全て円形メニューを利用する。

5.2.1 操作の起動法

地図ビューアーと同じように、ミュージックプレイヤーでも Out-In のクロッシングをトリ ガーとして使うことにより、前の操作を終わらせ、次の操作を行う。レーザスポットがスク リーン上のミュージックプレイヤー表示領域の縁を横切ると、クロッシングとする。図 5.5 は ミュージックプレイヤーの状態遷移図である。この図により、初期状態では、レーザスポット がミュージックプレイヤー表示領域の左側の縁を横切ると、再生ファイルを選択する状態に なる。再生したいファイルを決め、再び左側の縁を横切ると、再生が始まる。状態遷移図は 複雑なので、全て説明するのは不可能である。以下は、初期状態 再生ファイルの選択 再 生 音量の調節 再生中の時間移動 初期状態の流れで説明する。



図 5.5: ミュージックプレイヤーの状態遷移図

5.2.2 再生ファイルの選択

円形メニューを利用してファイルリストを操作する方法はすでに 3.3.3 節で説明した。図 5.6(a) は再生ファイルを選択する状態のイメージである。八等分の円形メニューを使って、現 在選択されるファイルを赤色で表示する。円形メニューを右に回すと、ファイルリストが上 ヘスクロールし、円形メニューを左に回すと、ファイルリストが下へスクロールする。初期 状態では、レーザスポットがミュージックプレイヤー表示領域の左側の縁を横切ると、再生 ファイルを選択する状態になる。ユーザがクロッシング操作とファイルの選択操作に熟達す ると、図 5.6(b) に示すように、クロッシング操作とファイルの選択操作を統合する操作法が 可能である。例えば、初期状態から、①のようにクロッシングをすると、再生ファイルを選 択する状態になり、そのまま下のほうへファイルリストをスクロールする。



図 5.6: 再生ファイルの選択 (a)、クロッシング操作とミュージックファイルの選択操作を統合 する操作法 (b)

5.2.3 ミュージックファイルの再生

再生ファイルを選択する状態では、ファイルリストを上下にスクロールし、再生ファイルを 決める場合には、レーザポインタを消灯する。もう一回ミュージックプレイヤー表示領域の 外で、レーザポインタを点灯し、レーザスポットで左側の縁を横切ると、選択されたミュー ジックファイルを再生する (図 5.7(a))。図 5.7(b) は再生状態のイメージである。再生時間をス ライドバーで表示する。

5.2.4 音量の調節

再生の状態では、レーザスポットでミュージックプレイヤー表示領域の上側の縁を横切ると、 音量調節の状態になる。図 5.8(a) は音量調節状態のイメージである。八等分の円形メニュー



図 5.7: 再生ファイルを選択する状態では、レーザスポットで左側の縁を横切ると、選択されたミュージックファイルを再生する (a)、再生状態のイメージ (b)

が使われ、円形メニューの操作法は再生ファイルの選択状態での操作法と同じである。音量の大きさをピンクのバーで表示する。円形メニューを右に回すと、音量を上げ、左に回すと、音量を下げる。また、ユーザがクロッシング操作と音量調節の操作に熟達すると、図 5.8(b) に示すように、クロッシング操作と音量調節の操作を統合する操作法が可能になる。例えば、 再生状態から、①のようにクロッシングすると、音量調節の状態になり、そのまま音量を下 げる操作が始まる。



図 5.8: 音量調節状態のイメージ(a)、クロッシング操作と音量調節の操作を統合する操作法(b)

5.2.5 再生中の時間移動

音量調節状態では、レーザスポットでミュージックプレイヤー表示領域の右側の縁を横切 ると、再生中の時間移動が可能となる。図 5.9(a) はその状態のイメージである。円形メニュー には 1% と 10% の二つの選択肢しかなく、それぞれ時間移動の速度を表示する。図 5.9(b) に示 すように、レーザスポットが 1%の選択肢と対応する領域内に入ったら、1%の選択肢が選択 され、10%の選択肢が消える。選択しやすいように、この操作では六等分の円形メニューを 使っている。例えば、円形メニューを一周右に回すと、再生時間を表示するスライドバーを前 へ 6% 動かす。再生中の時間を移動する状態では、レーザスポットでミュージックプレイヤー 表示領域の下側の縁を横切ると、ミュージックプレイヤーが再生を停止し、最初状態に戻る。



(a)

(b)

図 5.9: 再生中の時間を移動する状態のイメージ(a)、1%の選択肢が選択される(b)

第6章 評価実験

本章では,提案手法の有効性を確認するために、二つの評価実験を行った。被験者は25 オから38 才までの男性6人で、全員レーザポインタを用いたインタフェースの使用経験がない。 被験者はスクリーンから3mのところに立つ。画面の解像度は1024 × 768 ピクセルであり、 スクリーン上に投影されたサイズは1.5m × 1.1m である(図 6.1)。カメラのフレームレートは 15fps に設定した。



図 6.1: 実験環境

6.1 地図ビューアーのパン操作についての評価

この実験では、地図ビューアーのパン操作の有効性を評価するために、ターゲットへの移 動時間を測定する。

6.1.1 実験方法

実験の画面を図 6.2 に示す。被験者は、画面右上のナビウィンドウで表示されたターゲット へ移動するタスクが与えられ、ターゲットが画面中心の円の中に入るようにスクロールする とタスクが終わる。即座に次のタスクが始まり、別の位置にターゲットが表示される。画面 中心の円の直径は 300 ピクセルである。ターゲットまでの距離は三種類とし、ターゲットの 角度は八種類とした (表 6.1、図 6.3)。即ち、すべてのターゲットが (1024 × 5) × (768 × 5) ピクセルの領域に納まる。

本実験では、レーザスポットを止めることによりマウスクリックをエミュレートしてスク ロールバーを操作する手法と提案手法を比較した。止める動作を検出するまでの時間は[4]を 参考にし0.8秒に設定した。最初はスクロールバーが使えない状態であり、レーザスポットを 0.8秒以上スクロールバー上に止めると、マウスクリックイベントが発生し、スクロールバー が選択され、使える状態になる。その後は、スクロールバーがレーザスポットの移動により、 スクロールする。レーザポインタを消灯すると、スクロールバーがまた使えない状態になる。 レーザポインタでスクロールバーをポイントしやすいように、スクロールバーの幅が 50 ピク セルに設定している。

被験者は1セットあたり24回のタスクを行い、各手法3セットずつ行った。6人の被験者 は6×2×3×24=864回タスクを行った。実験を始める前に、各手法を5分間ずつ練習し てもらった。そして、先に提案手法で実験をする被験者は3人であり、先にスクロールバー を操作する手法で実験をする被験者は3人である。



図 6.2: 実験で使用したシステムの画面

ターゲットまでの距離 (ピクセル)			3	800, 1	.000	1700		
ターゲットの角度 (度)	0,	45、	90,	135、	180,	225、	270、	315

表 6.1: ターゲットの位置



図 6.3: ターゲットの位置

6.1.2 実験結果

図 6.4 は一回タスクあたりの平均移動時間を示す。提案手法の方が速いことが分かる。図 6.5 はターゲットまでの距離による平均移動時間を示す。距離が短いほど、提案手法の効率が 良いことが分かる。図 6.6 にターゲットの角度による平均移動時間を示す。水平及び垂直方向 へのスクロールには差が見られないが、斜め方向へのスクロールについては、提案手法の方 が速いことが分かる。この理由は、一つのストロークにより、縦と横のスクロールの動作を 統合しているからである。



図 6.4: 平均移動時間

6.1.3 考察

主観的評価を尋ねたところ、被験者全員が提案手法の方が良いと答えた。レーザスポット の移動ベクトルによって、スクロール操作のスピードも、方向もコントロールしやすいとい う回答が多かった。一方、レーザポインタでスクロールバーを操作する手法では、手振れの 影響を受けて操作しにくいという問題が指摘された。また、一つのストロークにより、縦と 横のスクロールの動作を統合できることは、本手法の利点として好評を得た。

6.2 円形メニューによる数字の調整についての評価

この実験ではレーザスポットの移動角度による円形メニューを操作する手法の効率を評価 するために、数字を入力するタスクが与えられ、一連の数字入力終了までにかかった時間を 総計する。比較の対象として、[4] で提案した Number Interactor を用いた。



図 6.5: 距離による平均移動時間



図 6.6: ターゲットの角度による平均移動時間

6.2.1 実験方法

本実験では、[4]の説明により、二桁の Number Interactor を作った。図6.7 は Number Interactor が使えない状態であり、図6.8 は Number Interactor が使える状態である。Number Interactor が レーザスポットを止めることにより操作される。止める動作を検出するまでの時間は0.8 秒 である。Number Interactor が使えない状態の場合には、レーザスポットを Number Interactor 上に0.8 秒以上止めると、Number Interactor が使える状態になる。入力された数字が真ん中に 表示され、最初は50を表示する。真ん中より上の項目は、それぞれ+1、+5、+10 であり、下 の項目は、それぞれ-1、-5、-10 である。例えば、レーザスポットを+1 の項目上に0.8 秒以上 止めると、+1 の項目が選択され、レーザスポットを+1 の項目上に止め続ける限り、0.8 秒ご と真ん中の数字に1をプラスする。レーザスポットを2 秒以上 Number Interactor を外すと、 Number Interactor が使えない状態に戻す。[4] では、Number Interactor のサイズが詳しく述べ られていないので、大き目に設定している。Number Interactor の横幅を180 ピクセル、縦幅 を 420 ピクセルに設定している。

同じ数字を入力する場合は、足し算及び引き算の回数ができるだけ同じになるように、Number Interactor の構成と同様になるよう、図 6.9 のような円形メニューをデザインした。円形メ ニューの直径は 300 ピクセルである。入力された数字が円形メニューの中心に表示され、最 初は 50 を表示する。右に回すのは足し算であり、左に回すのは引き算である。

入力される数字が画面の右上に表示され、全部で10回であり、表示の順番はランダムである。6人の被験者はNumber Interactor と円形メニューを使って、数字を入力して、入力終了までにかかった時間を総計する。先にNumber Interactor を使う被験者は3人であり、先に円形メニューを使う被験者は3人である。



図 6.7: 使えない状態の Number Interactor



図 6.8: 使える状態の Number Interactor



図 6.9: Number Interactor によるデザインされた円形メニュー

6.2.2 実験結果

図 6.10 は入力終了までにかかった時間の被験者ごとの合計を示す。被験者によって、両手 法においてかかった時間があまり差のない人もおり、差が大きい人もいる。平均すると、円 形メニューの方が Number Interactor より 20 パーセントぐらい速いことが分かる。



図 6.10: 所要時間

6.2.3 考察

主観的評価を尋ねたところ、被験者6人のうち5人がレーザスポットの移動角度による円形 メニューを操作するインタラクションのほうが良いと答えた。円形メニューを使う場合は、被 験者の回す動作により足し算、引き算を行うので、コントロールしやすい。Number Interactor を使う場合は、止める時間により足し算、引き算を行うので、被験者の意図しない足し算、引 き算を起こしやすい。一方、一人の被験者から以下のような指摘が出てきた。「レーザポイン タが消灯しないと、他のメニュー項目を選択できないが、Number Interactor を使う場合は、点 灯したまま他のメニュー項目を選択できる。」

第7章 関連研究

7.1 大画面環境とレーザポインタを用いたインタフェース

大画面環境に適した新しいインタフェースとして、レーザポインタを用いた研究が様々な されている。Oh[5]らは、カルマンフィルタリングを用いるコンピュータ制御式レーザポイン タを設計した。レーザポインタに付けたボタンを押すことでクリック操作などを行う。しか しながら、評価実験により、そのようなレーザポインタでは、比較的大きな40ピクセル径の ターゲットを選択する際に、誤り率が依然として約40%であった。さらに、この研究では、同 時に複数のレーザポインタを使用できるシステムを構成した。Davis[15]らも複数のレーザポ インタを使用できるシステム LumiPoint を提案した。ユーザはレーザーポインタをペンのよ うに用いてドローイングしたり、絵を描いたりするので、このシステムは画面から比較的近 距離で利用するものである。

羽山[13]らは、レーザポインタを用いて対話的に聴衆の理解を促すことを目指したプレゼ ンテーション支援システムを開発した。この研究では、レーザーポインタで、指し示した場 所を強調して表示し、画面の隅をポイントし続けることでスライドをめくるなどの操作を行 うことができる。田川[14]はAfterglowというプレゼンテーション支援システムを開発した。 このシステムでは、レーザスポットをある場所に一定時間止めることで、様々な機能を選択 することができる。また、レーザポインタをペンのように使ってスライドに下線を引いたり、 注釈を入れたりすることもできる。

以上の研究では、本研究のようなレーザポインタストロークによるインタラクション手法 は見当たらない。

7.2 ストロークによるインタラクション手法

ストロークによるインタラクション手法はマウス、スタイラスを用いた操作にもよく使われている。本研究で用いた手法は、マウス、スタイラス・ベースシステムでのインタラクション手法を参考にした。Marking Menu [17] では、ストロークを描くとその方向に対応したコマンドが実行される。そして、Marking Menu は階層化することが可能である。しかしこのメニューでは、階層が深くなるほど、最初の位置から離れたところまでストロークを描かなければならないので、操作時間とエラーレートが増える。FlowMenu [18] では、ペンが元の位置に戻るようにストロークを描くことで、1 つのコマンドを実行する。このため、複数のコマンドを連続したストロークで実行可能である。また、円形のストロークを描く事で連続的なプ

ロパティを設定するような操作も可能である。Control Menu [19] はストロークによるズーム や回転などの連続したプロパティを入力することに特化したメニューである。ペンを動かし て方向によって変化させたいプロパティを決定したあと、ストロークの起点からペンがどれ だけ離れているかでその値を決定している。

五十嵐 [20] らはマウスのストロークの方向と長さにより、スクロール操作とズーミング操作を合わせて行う手法を提案した。

第8章 まとめ

本研究では、大画面環境においてレーザポインタを用いて値を調節するために、レーザポ インタストロークを利用した。レーザポインタストロークからレーザスポットの移動ベクト ルとレーザスポットの移動角度を得られる。これを利用することで、本研究は三つのインタ ラクション手法を提案し、実装した。提案手法により、レーザポインタを用いて高速な調節、 低速な調節、おおざっぱな調節及び精確な調節が可能となった。また、本研究は提案手法の 適用例として地図ビューアーとミュージックプレイヤーを示し、これを実装した。最後に、提 案手法の有効性を検証するために、実行時間などの基本的な性能を測定する評価実験を行っ た。評価実験により、提案手法は値の調節には実行時間が速く、使いやすい手法であること が明らかになった。

今まで、本研究は遠隔操作を想定し、行った。今後はスクリーンから近いところでレーザポ インタを用いて、値の調節についても研究したいと思う。スクリーンから近いところで、レー ザスポットの照射の精度が良いので、スタイラスのように使えると思う。

謝辞

今回の修士論文を完成するにあたり、指導教官である筑波大学システム情報工学研究科田 中二郎教授に様々な助言やご指導をいただきました。心から御礼申し上げます。また、筑波大 学システム情報工学研究科志築文太郎講師にはグループミーティングなどにおきまして適切 なご指導をいただきました。深く感謝いたします。さらに、有益な助言、ご指導を下さった筑 波大学システム情報工学研究科三末和男準教授ならびに高橋伸講師に心から感謝いたします。

また、田中研究室の皆様にも研究活動や私生活の両方にわたって大変お世話になりました. 特に、WAVE グループの皆様に評価実験の協力をいただきました。本当にありがとうござい ました。

最後に、私を支えてくださった家族や友人にもこころより感謝いたします。

参考文献

- [1] Richard A. Bolt. "Put-that-there": Voice and gesture at the graphics interface. *ACM SIG-GRAPH Computer Graphics*, Vol. 14, No. 3, pp. 262-270, Jul. 1980.
- [2] Daniel Vogel and Ravin Balakrishnan. Distant Freehand Pointing and Clicking on Very Large, high resolution displays. In UIST'05: Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology, pp. 33-42, Oct. 2005.
- [3] C. Kirstein and H. Mueller. Interaction with a projection screen using a camera-tracked laser pointer. In *Proceedings of The International Conference on Multimedia Modeling. IEEE Computer Society*, pp. 191-192, Oct. 1998.
- [4] Jr. Dan R. Olsen and Travis Nielsen. Laser pointer interaction. In *CHI'01: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 17-22, Apr. 2001.
- [5] Ji-Young Oh and Wolfgang Stuerzlinger. Laser pointers as collaborative pointing devices. In *Proceedings of Graphics Interface*, pp. 141-149, May 2002.
- [6] Rahul Sukthankar, Robert G. Stockton and Matthew D. Mullin. Smarter presentations: Exploiting homography in camera-projector systems. In *Proceedings of International Conference on Computer Vision*, pp. 247-253, Jul. 2001.
- [7] Brad A. Myers, Rishi Bhatnagar, Jeffrey Nichols, Choon Hong Peck, Dave Kong, Robert Miller and A. Chris Long. Interacting at a distance: measuring the performance of laser pointers and other devices. In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*, pp. 33-40. Apr. 2002.
- [8] 志築文太郎, 久松孝臣, 高橋伸, 田中二郎. 画面の周囲を利用したレーザポインタ向けイン タラクション手法. インタラクション 2006 論文集, 情報処理学会, pp. 167-168, 2006 年 3 月.
- [9] Takaomi Hisamatsu, Buntarou Shizuki, Shin Takahashi and Jiro Tanaka. A novel click-free interaction technique for large-screen interfaces. The 7th Asia-Pacific Conference on Computer-Human Interaction, 10pages, Oct. 2006.
- [10] 久松孝臣. 交差と囲む操作を用いたレーザーポインタによる大画面インタラクション修 士論文, 筑波大学システム情報工学研究科コンピュータサイエンス専攻, 2007 年 3 月.

- [11] Gordon Kurtenbach and William Buxton. The limits of expert performance using hierarchic marking menus. In *Proceedings of ACM Conference on Human Factors in Computing Systems*, pp. 482–487, Apr. 1993.
- [12] Richard R. Eckert and Jason A. Moore. The Classroom of the 21st Century: The Interactive Learning Wall. The *Interactive Learning Wall*, *SIGHI Bulletin*, 32(2): pp. 33-40, 2000.
- [13] 羽山徹彩. テキスト情報の理解し易い提示手法に関する研究. 博士論文, 北陸先端科学技 術大学院大学知識科学研究科知識社会システム学専攻, 2006 年 3 月.
- [14] Afterglow. URL: http://www.lleedd.com/afterglow/top.html¿.
- [15] J. Davis and X. Chen. Lumipoint: Multi-user laser-based interaction on large tiled displays. In *Displays*, pp. 89-102, Mar. 2002.
- [16] Don Hopkins. The design and implementation of pie menus. Dr. Dobb's Journal, Vol.16, No.12, pp.16-26, Dec. 1991.
- [17] Gordon Kurtenbach and William Buxton. The limits of expert performance using hierarchic marking menus. In CHI'93: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 482-487, Apr. 1993.
- [18] Franc, ois Guimbreti'ere and Terry Winograd. FlowMenu: Combining command, text, and data entry. In *Proceedings of ACM User Interface Software and Technology 2000*, pp. 213-216, May 2000.
- [19] Stuart Pook, Eric Lecolinet, Guy Vaysseix, and Emmanuel Barillot. Control menus: excecution and control in a single interactor. ACM CHI Conference on Human Factors in Computing Systems, pp. 263-264, Apr. 2000.
- [20] 五十嵐 健夫 Ken Hinckley. 移動速度に応じた自動ズーミングによる効率的ナビゲーション. インタラクティブシステムとソフトウェアに関するワークショップ VIII (日本ソフトウェア科学会 WISS2000), pp. 57-66. 2000 年 12 月.