

2016 年度 卒 業 論 文

サンドアニメーションの  
疑似体験システムについての研究

指導教員：渡辺 大地 講師

メディア学部 ゲームサイエンス プロジェクト

学籍番号 M0113174

小嶋 しおり

2017 年 12 月

2016年度 卒業論文概要

論文題目

サンドアニメーションの  
疑似体験システムについての研究

メディア学部

学籍番号：M0113174

氏名

小嶋 しおり

指導  
教員

渡辺 大地 講師

キーワード

サンドアニメーション、Leap Motion、インタラクティブアート、  
疑似体験、シミュレーション

近年は、インターネットは普及し、インターネット環境も発達している。それに伴い様々なソーシャルネットワーキングサービスも広まり、アート作品などを目にする機会が増えている。動画専用のソーシャルネットワーキングサービスである youtube によって気軽に製作工程も含めたアート作品を見ることができる。他にも多くの人々に情報を共有することができる facebook、twitter によって、さらに一人が発信した情報が広まるようになった。それによりアート作品を多くの人目にするようになった。

しかし、アート作品の多くは実際に触れる機会が少なく、道具や環境を含め準備に手間がかかることが多いため自分から行うには手を出しにくい場合も多い。それにより、結果としてソーシャルネットワーキングサービスなどで動画などを見るだけで終わってしまう。そのため、環境や場所を問わず疑似体験できるものが必要である。パソコンはインターネット環境さえあれば場所や環境を問わず使用することができる。そのため、パソコンを利用することで場所や環境を問わず疑似体験を行うことができると考えた。

本研究ではパソコン上で砂を多数の点で表現し、Leap Motion を使用することで実際に手を動かすことで操作を行うシステムを作成する。そのシステムにより、実際に手を動かすことで砂を操るというサンドアニメーションの疑似体験を行うシステムを実装した。そのシステムにおいて、砂のリアルな表現よりも実際のサンドアニメーションと同様の操作で体験するという点を優先した。提案手法の評価、検証ではユーザーからの意見や感想から実際のサンドアニメーションの面白さを表現することができたかを評価、検証した。

検証結果としては、砂に見立てた多数の点を移動することでサンドアニメーションの様子を再現した。また、サンドアニメーションで実際に用いられる技法の一部を再現した。

アンケートも行い、砂を動かす体験が楽しかったという意見がある一方感覚や扱いが難しいという意見もありシステム、インターフェース共に改善点があることが判明した。

# 目次

第1章	はじめに	1
1.1	研究背景と目的	1
1.2	論文構成	3
第2章	サンドアニメーションとは	4
2.1	概要	4
2.2	技法	6
2.3	取り扱った技法	10
第3章	実装したシステム	12
3.1	機能、システムの説明	12
3.2	Leap Motion の機能	12
3.3	砂の作成	14
3.4	指の位置の取得	15
3.5	砂を動かす表現	16
3.5.1	反応する指が1本の場合	16
3.5.2	反応する指が2本の場合	19
3.6	砂の追加	20
第4章	評価、検証	22
4.1	プログラムの検証	22
4.1.1	プログラムの実行結果と実物との比較	22
4.1.2	実際のサンドアニメーションとの比較	28
4.2	試験者からの意見、感想	29
4.3	考察	30
第5章	まとめ	32

謝辭 33

参考文献 34

# 目 次

2.1	サンドアニメーションの制作環境	5
2.2	サンドアニメーションのスクリーンの様子	5
2.3	指で砂を退かして絵を描く様子	7
2.4	砂を手で広げた様子	8
2.5	砂を置いて絵を描く様子	9
2.6	砂を広範囲に撒く様子	10
3.1	Leap Motion	13
3.2	実際に Leap Motion を使用する様子	14
3.3	Leap Motion で検出する情報	15
3.4	砂が指を避ける様子	16
3.5	指の高さによる影響範囲の違い	17
3.6	$r$ を固定した場合の $h$ と $r'$ の関係性	18
3.7	砂が動く様子と計算式	18
3.8	砂が 2 つの影響範囲に入った場合の模式図	20
3.9	砂が 2 つの影響範囲に入った場合の模式図	20
3.10	砂を追加する際の範囲の広がり の模式図	21
4.1	ランダムな位置に砂を描画した画面	23
4.2	指先の位置を球で示した様子	24
4.3	指先で砂を動かした後の様子	25
4.4	砂を退かす技法の比較	25
4.5	低い位置から砂を追加した様子	26
4.6	高い位置から砂を追加した様子	27
4.7	砂を置く技法の比較	28
4.8	砂を撒く技法の比較	28
4.9	サンドアニメーションとシステムで描いた絵の比較の比較	29

# 第 1 章

## はじめに

### 1.1 研究背景と目的

近年は、インターネットの発達により、ニコニコ動画 [1]、youtube[2] や pixiv[3] などの動画、画像専用のソーシャルネットワーキングサービスの他、twitter[4] や facebook[5] などによって情報を多くの人々が共有することで、様々なアート作品を目にする機会が増えた。インターネット環境の他、コンピューターグラフィックスの技術も発達しており、アート作品の中でも絵画をデジタルで行うデジタルペインティングについても研究や開発が数多く行われている。写真を編集、加工するソフトウェアの開発、ペイントツールに関しては近年特に開発や研究が進んでおり、水彩画や油絵、クレヨンなどの特殊な塗り方の他、スポンジやスプレーなどの特殊な道具を使用した表現も開発が進んでおり、ペンタブレットなどでそれらを使用した絵を擬似的に描くことができるようになった。他にも寺井ら [6] の水墨画やまた、Photoshop[7] や SAI[8] などの広く広まっているペイントソフトについては解説本や pixiv[3] で「描き方」「講座」などのタグで描き方も紹介されている。砂にかかわる研究としては既存の絵をサンドアートのように表現するという平岡ら [9][10] や森田 [11] の研究がある。アート作品の創作活動の支援という観点の研究には浦 [12][13] の研究がある。ペイントツール以外のアートと呼ばれるものをデジタル上で再現するという内容

の研究は小嶋 [14] のドリッピングアートの研究や武藤 [15] のラテアートの研究、寺川ら [16] のバルーンアートの研究などでおこなわれている。しかし、同じアート作品でも擬似体験するシステムがあまり作られていないものも存在する。本研究ではその中からサンドアニメーション [17] を対象として擬似体験をするシステムを提案する。

サンドアニメーションを題材とした先行研究としては浦ら [18][19] の研究や原ら [20] の研究がある。「テーブルトップインタフェースを利用したサンドアニメーションの生成シミュレーション」という研究 [18] ではサンドアニメーションの制作技術を 6 つに分類し、様々なデバイスを組み合わせたインタフェースを使用し、制作技術を再現した。砂を置く、砂を撒く、砂を飛ばす、砂を延ばす、砂を削る、砂を拭うという 6 種類である。砂の表現においてはハイトフィールドというフィールドを格子状に分割し、各セルごとに高さを管理するデータ構造を用いている。高さによって画面の色を変化させることで砂の濃淡を表している。この研究では実際のサンドアニメーションと同様の手順を行うことで実際のサンドアニメーションと酷似した絵の作成にも成功している。しかしこの研究に使われているインタフェースは赤外線 LED や赤外線のみ透過する Wed カメラ、アクリル板や左右反転投影を行うプロジェクターなど数多くのデバイスが組み合わさってできている。そのため、この研究で使っているインタフェース自体準備の手間がかかる。本研究では少ないデバイスを用いた疑似体験を行うシステムを実現した。

先述のように近年ではペイントソフトの開発が多く進んでいるが、それに比べサンドアニメーションをはじめとしたアート作品は研究や疑似体験が進んでいない。一方インターネットの発達により多くのアート作品の画像や映像を目にする機会は増えている。本研究では疑似体験を通して、見るだけでなくアート作品を作ることに関心を持たせる環境を支援するため、アート作品の中からサンドアニメーションを場所を選ばず疑似体験するシステムの作成を目的とする。

シミュレーションを行うという浦らの先行研究 [18] においてもデバイスに多くの準備が必要であり場所を選ばず疑似体験するという本研究には適さない。少ないデバイスを用いた疑似体験シ

システムの作成を目的とする。

本研究では、立体空間上に多数の点を描画することで砂を表現する。本研究でデバイスとして Leap Motion を使用した。Leap Motion とは手のジェスチャーによつて操作を目的としたデバイスであり、手の立体的な情報を取得することができる。取得した手の位置情報を使用して砂を動かすシステムを作成することで、手で砂を動かすというサンドアニメーションの特徴を疑似体験するシステムを作成した。また、実際のサンドアニメーションで使用する技法を再現することでより本物のサンドアニメーションに近い表現を再現した。本研究においては砂の表現よりも砂を手で動かすという手法を重視した。

ユーザー作成したシステムを実際にプレイしてもらい、サンドアニメーションの疑似体験を行うことができたかアンケートを行った。アンケートの結果、砂を動かして絵を描くという体験が面白かったという感想が多かった。一方高さの感覚の掴みにくさや細かな絵を描く難しさについての感想もあり、インターフェース、システム共に改善の余地があることが分かった。

## 1.2 論文構成

本論文は全 5 章で構成する。1 章では研究目的と使用するデバイスである Leap Motion の説明、2 章ではサンドアニメーションについての説明を行う。3 章では VisualStudio 上で実装したシステムについて説明とサンドアニメーションで行われている技法の中で実装できた内容について説明する。4 章では実際に使用してもらったユーザーからの意見を評価、検証を行う。最後に 5 章で本研究を統括する。

## 第 2 章

# サンドアニメーションとは

### 2.1 概要

サンドアニメーションとはガラスのキャンパスを下から光で照らし、その上に撒いた砂を動かすことで光と砂に夜陰影を用いて様々な絵を描く。砂絵を少しずつ動かすことでこま撮りされたアニメーションのように物語を展開させるアート技法のことであり、サンドパフォーマンスとも呼ぶ。上記のように砂絵を少しずつ動かす様子をオーバーヘッドで撮影し、砂絵が移り変わる様子を撮影する。また、その映像をスクリーンに投影して披露するというものである。

近年では、テレビ CM や音楽の PV 映像になる、クラシック、オペラ、三味線などの音楽とコラボレーションするなど活動の幅を広げている。また、サンドアニメーションには、投影するスクリーンの横にパフォーマーが立ち、砂絵を作るパフォーマーの様子も同時に見て楽しむという要素もある。

実際にサンドアニメーションで絵を描くためのキャンパスとなるガラス板、下からキャンパスを照らすライト、そしてオーバーヘッドで砂絵を作るキャンパスを撮影するカメラは必須となっている。また、そのほかに人に披露する場合はその様子を映すプロジェクターとスクリーンが必要になる。このように、サンドアニメーションを実際に行うには多くの準備が必要になる。図

2.1[18] は実際のサンドアニメーションの制作環境の模式図である。

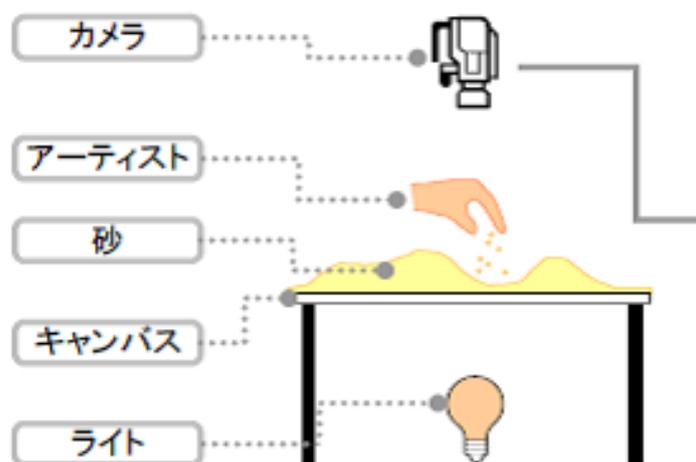


図 2.1 サンドアニメーションの制作環境

文献 [18] 図 2-1 より転載

キャンパスを見た時、砂がある部分は光が遮られて影になるため黒くなり、砂がない部分は砂がなくキャンパスをライトが照らすため白く見える。図 2.2 がサンドアニメーションのキャンパスの映像である。

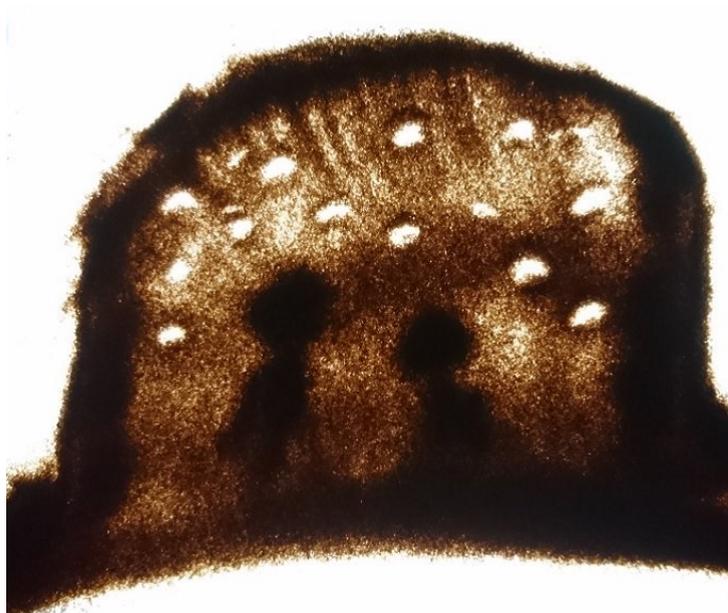


図 2.2 サンドアニメーションのスクリーンの様子

サンドアニメーションにはスクリーンの隣に立ち砂絵を作っている様子も楽しむという内容がある。しかし今回の研究からこの内容は除外する。

## 2.2 技法

サンドアニメーションでは砂を動かして絵を描く際に手の様々な部分を用いて制作を行う。その中には様々な技法があり、サンドアニメーションの疑似体験を行うにあたり、技法の幾つかを再現した。

使われている技法を本研究では、4種類に分類する。

第1の技法は広がっている砂を指で退かすことで光を通すようにするものである。この技法は指先や手の側面などを用いて砂を退かしてスクリーンに光が通るようにする。また、退かした後、淵に砂が溜まることで強い影ができて輪郭がはっきりするため、文字を書いたりする場合にも多用する。他にも砂を動かす範囲が明確になっているため、細かな部分の作業や、手全体を動かすことで複数の指を平行して動かすことで波や髪の実現に利用することもある。範囲が広い場合は新しく描く絵の妨げにならないよう既にある砂を全て退かすことで無地のキャンパスを準備するためにも使う。

図 2.3 は砂絵的一幕であり、髪の流れの様子などに技法を使っている。



図 2.3 指で砂を退かして絵を描く様子

第 2 の技法も第 1 と同様に砂を指で動かす技法であるが、こちらは 1 か所に溜まっている砂を手の側面や手のひらなどを使って広げるというものである。砂を拡げるという技法は光を通すようにするものとは逆に砂を広範囲に広げることで光を遮り、影を作る技法である。この広げるという技法はのちに広げた砂を指で退かす技法を使うための下地を作るために行うことが多い。図 2.4 は手で砂を広げた砂の様子である。



図 2.4 砂を手で広げた様子

第3の技法は砂を新たに追加するという点において新たに砂を加えるというものである。キャンパスからあまり離れていない低い位置から砂を追加し、置く技法である。この技法は既に撒かれている砂を退かして表現するものとは逆に、無地のキャンパスに砂を置くことで影を作り絵を描く際に多く用いる。また、砂が退かされて一部無地となった部分に砂を置き、組み合わせで絵を描くことも多い。砂を退かした絵と比べて輪郭が不明瞭になること、薄く砂がある場所にも描くことができる点も特徴である。図 2.5 は砂を置いて描いたものである。



図 2.5 砂を置いて絵を描く様子

第 4 の技法は第 3 と同様にキャンパスに砂を追加する技法であるが、こちらはキャンパスから高い位置から砂を追加することで広範囲に砂を撒く技法である。広い範囲に砂を撒き、砂を広げる技法と同様に砂を退かして絵を作る下地を作る際に用いる技法である。また、全体的に薄く砂を置き、画面を薄暗くすることができる点もこの技法の特徴である。グラデーションの表現などにも用いる。また、勢いをつけて砂を撒くことで速さなどを表現し、絵の一部として使うことも多い。

図 2.6 は砂を撒いている様子である。

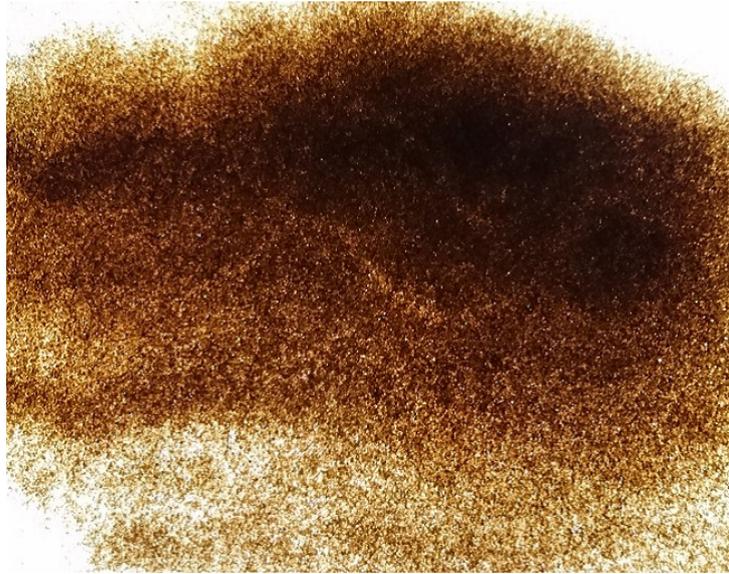


図 2.6 砂を広範囲に撒く様子

## 2.3 取り扱った技法

本研究では以上の中から砂を退かす技法、砂を置く技法、砂を撒く技法の3種類を取り扱った。

砂を退かす技法を取り扱った理由はサンドアニメーションにおいて特に欠かせない技法だからである。砂を置くことで絵を描くことも可能ではあるが、その場合砂ではなくインクなどでも代用することができる。退かすことで陰影を作り絵を描くという点は移動させるということができる砂ならではの技法であり、他の絵を扱うアート作品との大きな違いであると考えられる。

砂を置く技法、砂を撒く技法は特に砂を取り扱った理由は後から追加するという点において重要な技法だからである。実際のサンドアニメーションでは砂を手放す高さによって砂の広がり方に違いが出るため、それも Leap Motion を使用することで高さの情報もシステム上で扱うことができたため再現可能であった。

砂を広げる技法を取り扱わなかった理由としては再現が困難であったためである。実際のサンドアニメーションでこの技法を扱う場合、基本的に一か所に集まっている砂を崩して広げること

である。山のようにになっている状態を想定する必要があるが、今回作成したシステムにおいて手の高さは利用するが、砂の高さについては考慮していない。また、似た状況を再現する場合、集まっている砂の一部を移動させ、一部をその場に残すことでも再現は可能であるが今回のシステムでは全ての砂に同じ処理を行うため、一部のみを動かすことは難しい。

以上より今回は再現可能であった砂を退かす、砂を置く、砂を撒くという3つの技法を取り扱った。

## 第 3 章

# 実装したシステム

### 3.1 機能、システムの説明

本章ではサンドアニメーションの操作を再現した提案技法を、砂の作成、指の位置の取得、砂を操る表現、砂を追加する機能の順に説明する。本研究ではキャンパスの前に立った状態を想定し、キャンパスを斜め下に見ている状態とする。立体空間上に多数の点を描画することで巻かれた砂を表現し、Leap Motion を用いた手の位置情報を利用することで、実際に手を動かすことで砂を操るというサンドアニメーションの特徴を再現するシステムを実装した。

### 3.2 Leap Motion の機能

本研究で実装したシステムを説明する際に、使用するデバイスである Leap Motion について説明する。Leap Motion とは、2012 年 Leap Motion 社から販売されたデバイスであり、手のジェスチャーによってコンピュータの操作を行うという用途で使用するものである。Leap Motion を使用することで奥行や高さなどの立体的な操作を行うことができ、さらに指のや関節の位置情報、手の動きや捻りなど、手に関する様々な情報を検出することができる。

また、デバイスがとても小さく持ち運びなども容易なため大がかりな準備が必要なく使用する

ことができる。図 3.1 にあるデバイスが Leap Motion である。図 3.1 のように持っている手と比べてもとても小さい。



図 3.1 Leap Motion

次に Leap Motion が扱う情報について述べる。Leap Motion はデバイスの上にあるものを検知するため、手は Leap Motion の上で動かさなければならない。また、画面に手全体が映されていることから分かるように、指先だけではなく手のひら、指の関節などの情報も使用することができる。位置情報のみではなく動きも取得可能である。他にも、両手などが重なった場合でも情報を補填するため取得情報が破綻しない。図 3.2 は実際に Leap Motion を使用して、手の検知を行っている様子である。



図 3.2 実際に Leap Motion を使用する様子

### 3.3 砂の作成

砂の作成について述べる。実装するシステムは立体空間を想定し、キャンパスを斜め下に見下ろす様子を想定する。その立体空間内に多数の点を描画し 1 つ 1 つを砂に見立てることで、撒かれている砂を表現する。最初のキャンパスが真っ白だと、1 度砂を撒く手間がかかり、また、撒く砂の量が多すぎた場合に処理が遅くなってしまうため、砂は最初から一定数撒いてあるものとする。描画する砂の初期位置については、描画する立体空間を  $(x, y, z)$  で表し 3 つの値はそれぞれ、 $x$  を横幅、 $y$  を高さ、 $z$  を奥行を表し、1 つ 1 つの砂の位置ベクトルを  $\mathbf{P}^i$  と表す。

ガラス板の上に置かれた砂を表現するため、砂を描画する高さ  $P_y^i$  は全て  $W$  という同じ値を入れる。前以て  $-10 \sim 10$  の値の乱数を規定しておき、その数値を  $P_x^i$  と  $P_z^i$  の値に入れる。これを用いて、たくさんの点を高さ  $W$  の横幅が  $-10 \sim 10$ 、奥行が  $-10 \sim 10$  の範囲でランダムに描画する。それにより、キャンパスに撒いた砂を表現する。

### 3.4 指の位置の取得

次に手の位置情報の取得について述べるが、それには前述の通り Leap Motion SDK を用いて指先の位置情報を取得する。Leap Motion は左右の手の判別、指の判別その他、1つ1つの指の付け根や関節などの位置の情報も取得することができる。今回はその中から各指の指先の情報のみを検出して使用する。図 3.3 は Leap Motion で取得する情報を基にシステム上で手の形を再現したものである。

取得した指先の位置をシステム上で可視化するため、指先の位置に黄色の球体を描画する。手を動かすことで黄色の球もその動きに合わせて移動する。



図 3.3 Leap Motion で検出する情報

## 3.5 砂を動かす表現

### 3.5.1 反応する指が 1 本の場合

次に砂を動かすシステムについて述べる。砂を動かすシステムには各砂の位置と指先の位置の情報を利用する。

砂の位置ベクトル  $\mathbf{P}^i$  と指先の中心の位置ベクトル  $\mathbf{F}^j$  の距離  $\overline{\mathbf{P}^i\mathbf{F}^j}$  を計算する。そしてその距離が一定範囲  $r$  以内になった場合砂がその影響範囲の外まで指から離れるように移動する。この  $\mathbf{P}^i\mathbf{F}^j$  間の距離が一定範囲  $r$  以内の空間を影響範囲と呼ぶ。影響範囲にある砂が  $P_x^i, P_z^i$  方向に、 $\mathbf{F}^j$  から離れるように移動する様子を示した模式図が図 3.4 である。

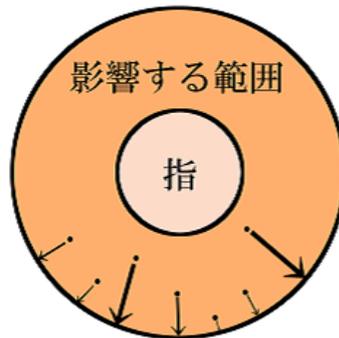


図 3.4 砂が指を避ける様子

次に指の高さによる影響範囲について述べるが、影響範囲は指先の位置ベクトル  $\mathbf{F}^j$  からの立体的な距離によって決まるため指  $F_y^j$  と砂  $P_y^i$  が違う高さにある場合、実際に砂が動く範囲の距離は影響範囲の距離よりも狭くなる。指と砂の高さの差を  $h$ 、実際に影響する砂の半径を  $r'$  とする。それをを図示したものが図 3.5 である。

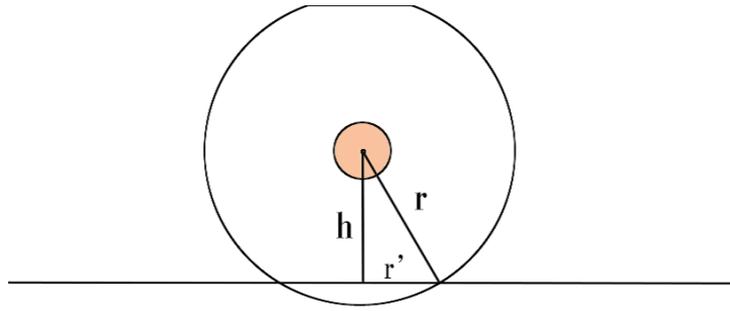


図 3.5 指の高さによる影響範囲の違い

$$r^2 = r'^2 + h^2 \quad (3.1)$$

$$r' = \sqrt{r^2 - h^2} \quad (3.2)$$

ピタゴラスの定理により、3つの長さには式 (3.1) のような関係が成り立つ。よって  $r'$  の値は式 (3.2) のようになる。この計算を説くと、指  $F_y^j$  と砂  $P_y^i$  の高さ  $h$  と  $r'$  の長さは比例しておらず実際に砂が動く範囲  $r'$  は指  $F_y^j$  と砂  $P_y^i$  の高さの差が大きくなる程急激に  $r'$  が小さくなる。そのため、砂を動かすことで細かな線を描くことは現状難しい。 $r$  を一定とした時の  $r'$  と指と砂の高さの差  $h$  の関係を図示したものが図 3.6 である。

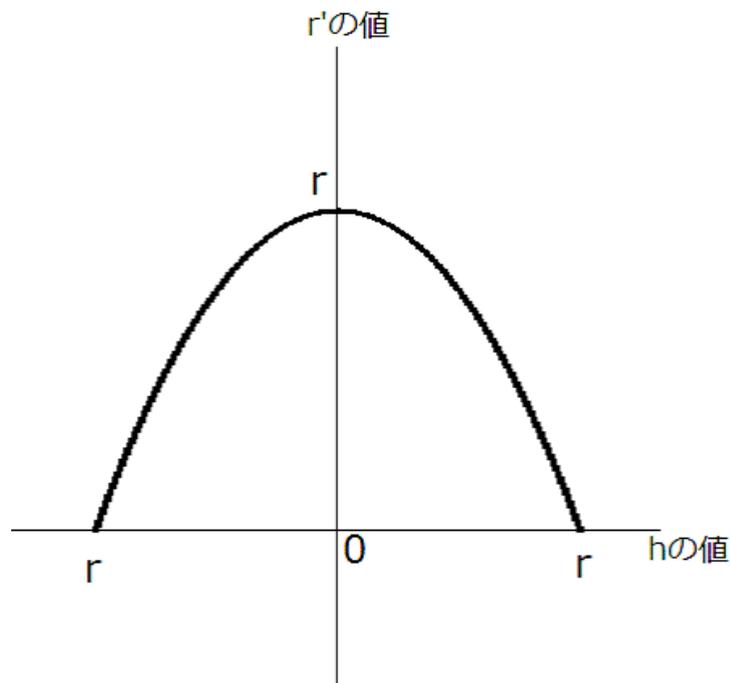


図 3.6  $r$  を固定した場合の  $h$  と  $r'$  の関係性

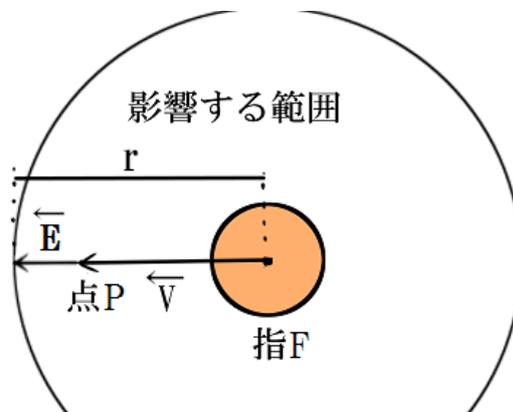


図 3.7 砂が動く様子と計算式

指先の位置ベクトルを  $\mathbf{F}^j$ 、それぞれの砂の位置ベクトルを  $\mathbf{P}^i$  とおく。式 (3.3) のように、指先から砂へ向かうベクトルを  $\mathbf{v}^i$ 、指先から砂へ向かうベクトルと同じ方向で、砂を範囲の外まで移動するベクトルを  $\mathbf{E}^i$  とする。図 3.7 はベクトルの模式図である。この時、指から砂へ向かうベ

クトル  $\mathbf{v}^i$  と砂を影響範囲の外まで移動するベクトル  $\mathbf{E}^i$  を式にすると

$$\mathbf{v}^i = \sum_{j=1}^5 \overrightarrow{\mathbf{F}^j \mathbf{P}^i} \quad (i = 1, 2, \dots, n) \quad (3.3)$$

$$\mathbf{E}^i = r' \frac{\mathbf{v}^i}{|\mathbf{v}^i|} - \mathbf{v}^i \quad (3.4)$$

となる。式 (3.4) のように  $\mathbf{v}$  を絶対値で割ることで正規化し、それを  $r'$  倍にすることで指から影響する範囲の外までのベクトルを出すことができる。しかしそのままでは砂の位置  $\mathbf{P}^i$  から範囲の外側まで  $r'$  分の距離動くため、影響範囲の境界のさらに外側まで移動する。そのベクトルから移動し過ぎる分である指先から砂へ向かうベクトル  $\mathbf{v}^i$  を引くことで、影響範囲の境界線の位置まで砂を移動させることができる。

動かした砂が境界で止まるため、動かした砂で指の跡の輪郭を表現することができた。

### 3.5.2 反応する指が 2 本の場合

次に 2 つの指の距離が近く、影響範囲が重なった場合について述べる。砂がどちらかの影響範囲にしか入っていない場合は 3.5.1 項の計算を適用する。2 つの影響範囲が重なった場合、先に親指に近い人差し指、中指、薬指、小指の順に優先して 1 本の時と同じ処理を行う。次にもう 1 本の指との処理を行う。処理後に影響範囲内にあった場合は再び同じ処理を繰り返す。2 つの指が横並びだった場合、横幅  $x$  は 2 つの指からそれぞれ逆方向のベクトルを受けるため 2 つの影響範囲の中を行き来するが、奥行  $z$  は同じ方向のベクトルを受ける。そのため、2 つの指が横並びだった場合は  $z$  方向に砂が移動していき、影響範囲の外まで移動する。2 つの指の影響範囲にある場合と片方の指の影響範囲にしか入っていない場合で分けて式にすると

$$\mathbf{v}^i = \begin{cases} \overrightarrow{\mathbf{F}^j \mathbf{P}^i} + \overrightarrow{\mathbf{F}^{j+1} \mathbf{P}^i} & (\text{if } |\overrightarrow{\mathbf{F}^j \mathbf{P}^i}| < r \text{ and } |\overrightarrow{\mathbf{F}^{j+1} \mathbf{P}^i}| < r) \\ \overrightarrow{\mathbf{F}^j \mathbf{P}^i} & (\text{else}) \end{cases} \quad (3.5)$$

となる。移動の結果として、2つの指の影響範囲の外であり、砂に最も近い  $z$  の値を持つ点へ移動する。それは高さ  $W$  で2つの影響範囲の境界線が重なった2点であるため、処理が高速で行われ、2つの影響範囲が重なる範囲にある砂は重なった点の境界線が重なった2点の内近い方へ移動しているように見える。図 3.8 と図 3.9 は2つの影響範囲に入った場合の砂の動きの模式図である。

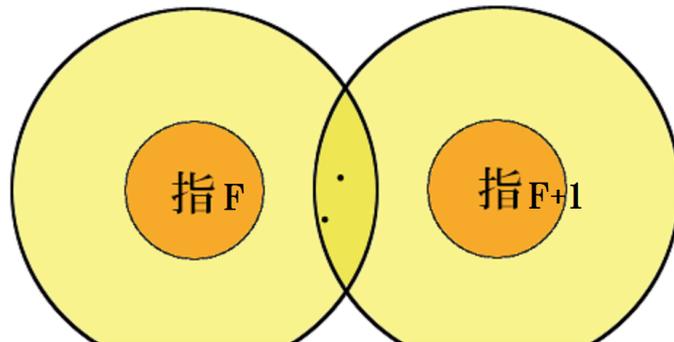


図 3.8 砂が2つの影響範囲に入った場合の模式図

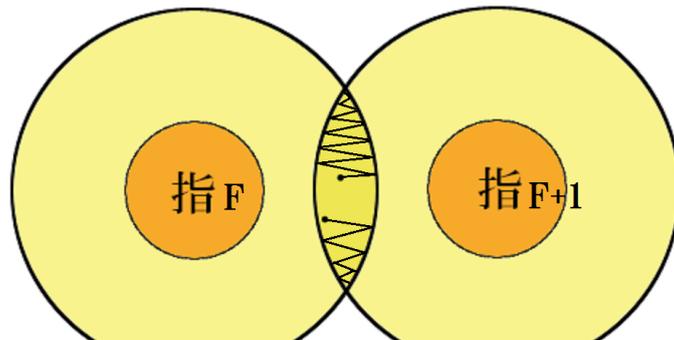


図 3.9 砂が2つの影響範囲に入った場合の模式図

### 3.6 砂の追加

特定のキーを押すことで新たに砂を追加する。砂として追加する点の位置ベクトルを  $\mathbf{Q}$  とし、砂が重ならないために加える乱数を  $d$  とする。また、計算に指の位置ベクトル  $\mathbf{F}^j$  を使用するが、

対象は人差し指のみであるため  $j$  の値はシステム上人差し指を示す 2 に固定すると砂の位置ベクトルを表す  $\mathbf{Q}$  それぞれの成分は

$$\begin{cases} Q_x = F_x^2 + dF_y^2 \\ Q_y = W \\ Q_z = F_z^2 + dF_y^2 \end{cases} \quad (3.6)$$

という計算式となり、人差し指の下の位置に新たな砂を追加し、実際のサンドアニメーションにもある砂を追加する技法を表現した。また、追加する砂の値  $Q_x$  と  $Q_z$  には砂が完全にならぬように誤差として乱数を  $F_x^j$ 、 $F_z^j$  に加える。加える乱数は指の高さ  $F_y^j$  に比例しており、手が高い位置にある状態で砂を追加する程誤差が大きくなり、広い範囲に砂を分布する。これにより、砂を置くという操作と砂を撒くという操作を高さによって使い分けることができる。図 3.10 は手の位置と撒く砂の範囲を図示した模式図である。

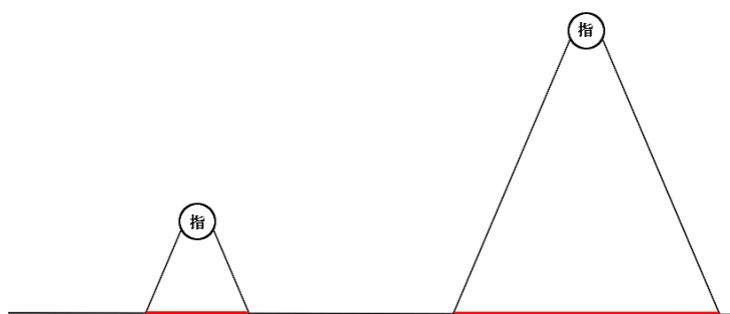


図 3.10 砂を追加する際の範囲の広がり の模式図

3 章の中の画像はすべて左手で検出を行っている。両手を検出することも可能であるが、Leap Motion の反応する範囲が狭く両手を検出すると画面が埋まること、両手を検出すると処理する量が 2 倍になり処理が遅くなってしまふことなどから左手で検出を行い右手でキー操作を行うことが基本的な形となっている。

# 第 4 章

## 評価、検証

### 4.1 プログラムの検証

#### 4.1.1 プログラムの実行結果と実物との比較

3 章内で実装したシステムの実行結果について記述する。

砂を作成するシステムについて記述する。高さを表す  $P_y$  に同じ数値  $W$  を入れたことで、すべての砂が同じ高さに描画され、平面に撒いた様子を表現することができた。また、 $-10 \sim 10$  の乱数を用いたことでランダムな数値を横幅を表す  $P_x$  と奥行を表す  $P_z$  に入れたことで範囲内のランダムな位置に砂を描画し、満遍なく撒いた砂を表現することができた。点を描画した初期画面が図 4.1 である。

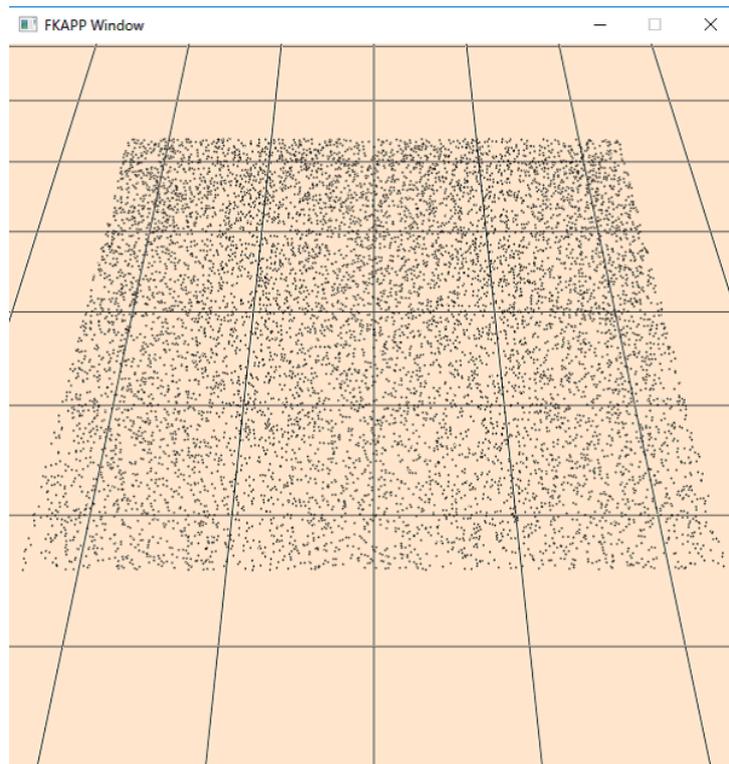


図 4.1 ランダムな位置に砂を描画した画面

次に指の位置情報について記述する。Leap Motion を使用したことで実際の手の立体的な位置情報を取得することができた。また、指先の位置だけを球体で示したことで、手によって砂が隠れて見えなくなるという状況を防ぐことができた。カメラを斜めからの視点にしたことで指の高さについても視認することができた。

図 4.2 は検出した指先の位置情報をシステム上で反映した画像である。

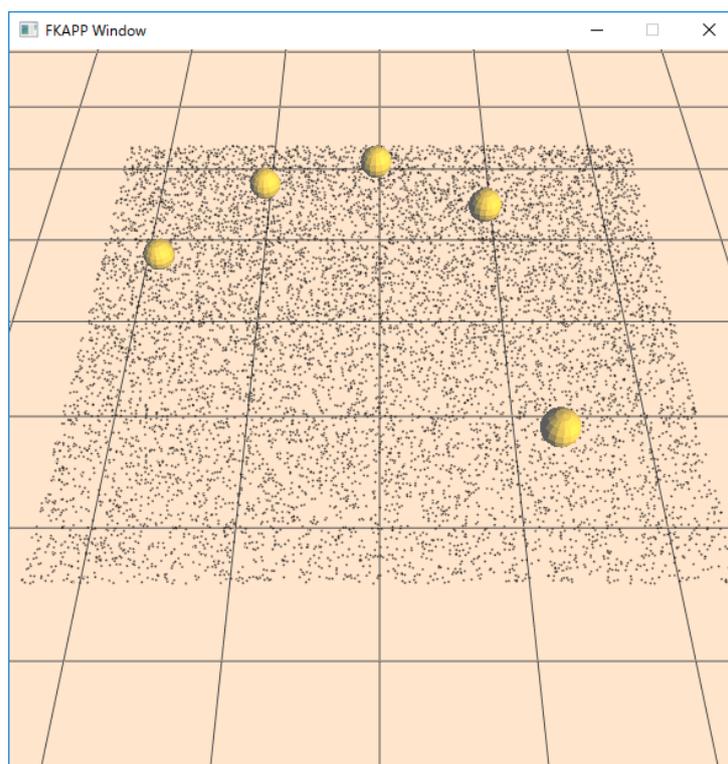


図 4.2 指先の位置を球で示した様子

次に砂を動かす表現について記述する。3.5 節で記述した数式を使用することで砂が指から離れていくシステムを実現した。図 4.3 画像がシステム上で指で砂を動かした画像である。指を動かした脇に砂が溜まり、輪郭が残っている。2.2 節で記述したように、本物のサンドアニメーションにもある、淵に砂が溜まり輪郭が残るといった表現を再現した。図 4.4 の画像は実際のサンドアニメーションで砂を退かす技法を使用している画像とシステムの比較画像である。

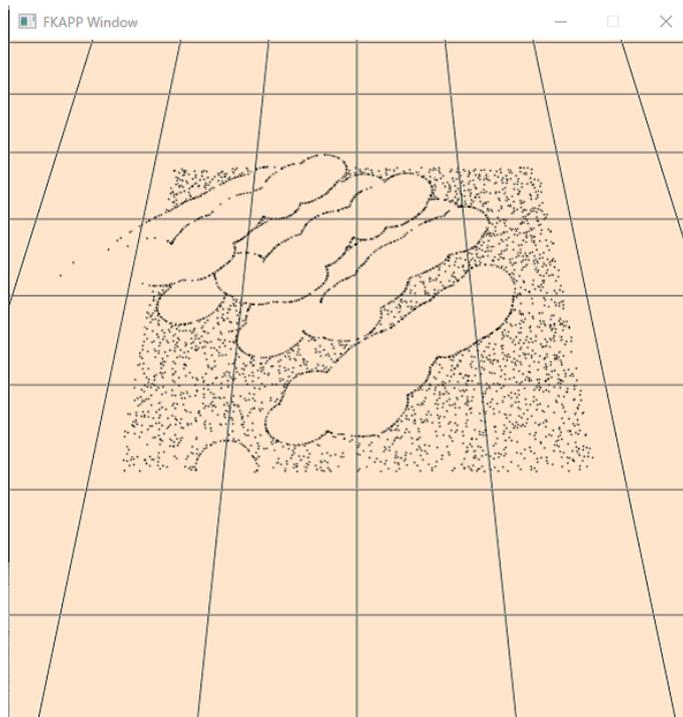


図 4.3 指先で砂を動かした後の様子

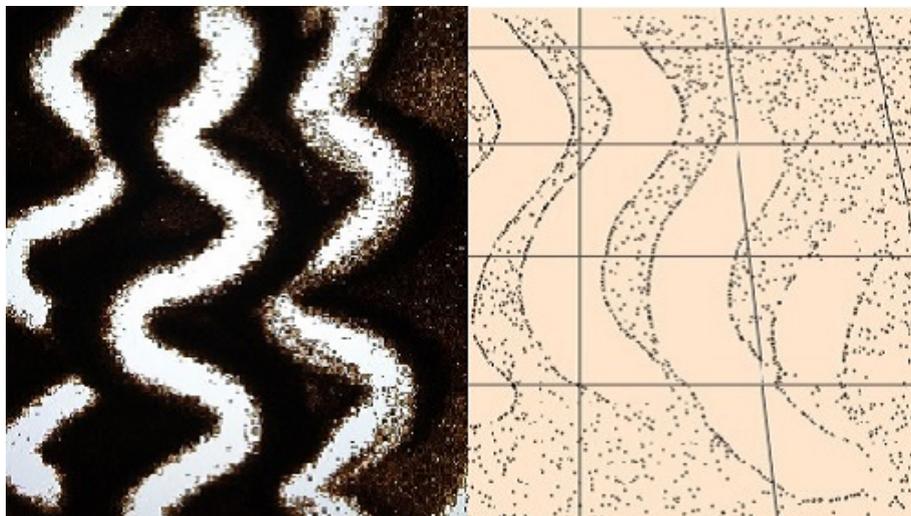


図 4.4 砂を退かす技法の比較

次に砂を追加するシステムについて記述する。手の高さの情報を使用することで、高さによって砂を撒く範囲を変更することができた。それにより、砂を置く技法と砂を撒く技法を使い分け

ることができた。図 4.5 は実際に砂を退かして見えやすくした部分に低い高さから砂を追加した様子である。図 4.6 は図 4.5 の後にさらに高い位置から砂を追加した様子である。低い位置から追加した砂と比べ広い範囲に散布されていることが分かる。

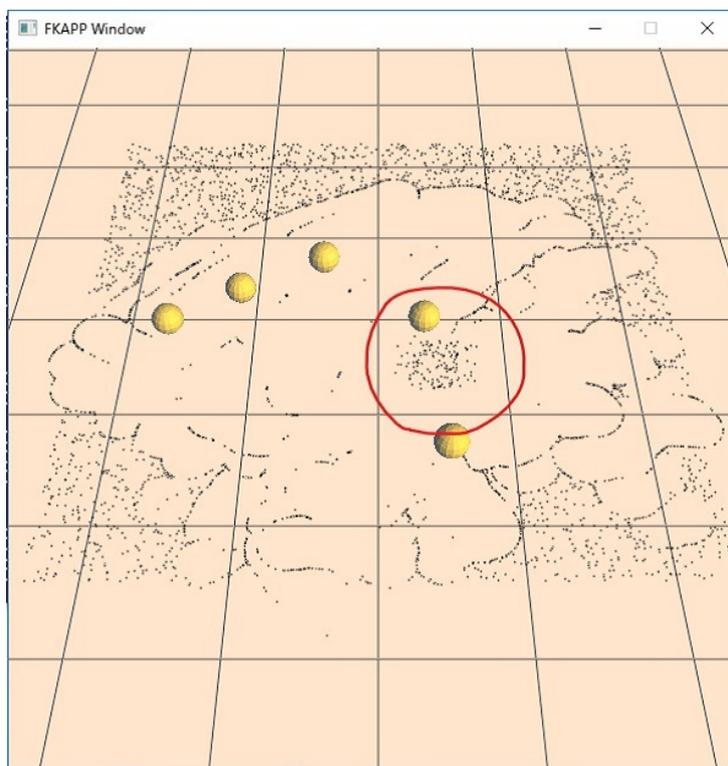


図 4.5 低い位置から砂を追加した様子

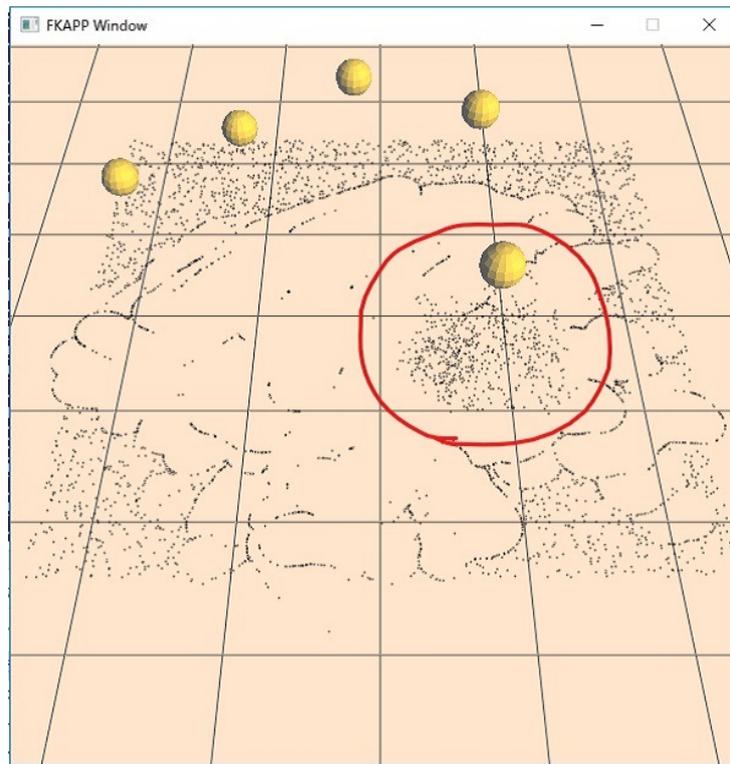


図 4.6 高い位置から砂を追加した様子

実際のサンドアニメーションで使用する技術としては2つに分類するため、比較は砂を置く技法と砂を撒く技法で別に行う。第1に砂を置く技法について述べる。乱数を加えることで砂が重なることを防いでいるため、狭い範囲に砂を分布した。分布する砂の濃淡によって絵を作成することができた。図 4.7 は実際のサンドアニメーションで砂を置く技法で描いた木と、システムで描いた木の比較画像である。

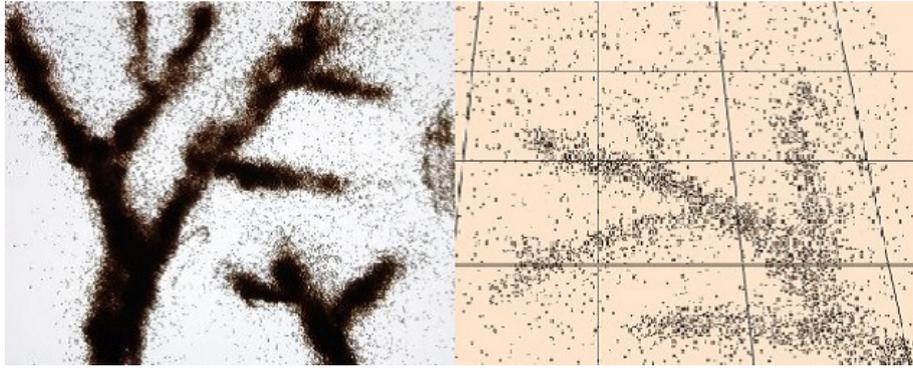


図 4.7 砂を置く技法の比較

第 2 に砂を撒く技法について述べる。指の高さによって乱数の値を大きくすることで、指の高さを高くすることで広い範囲に分布することができた。それにより広い範囲に濃淡をつけることができた。図 4.8 は実際のサンドアニメーションで砂を撒く技法で作った下地とシステムで砂を撒いて作った下地の比較画像である。

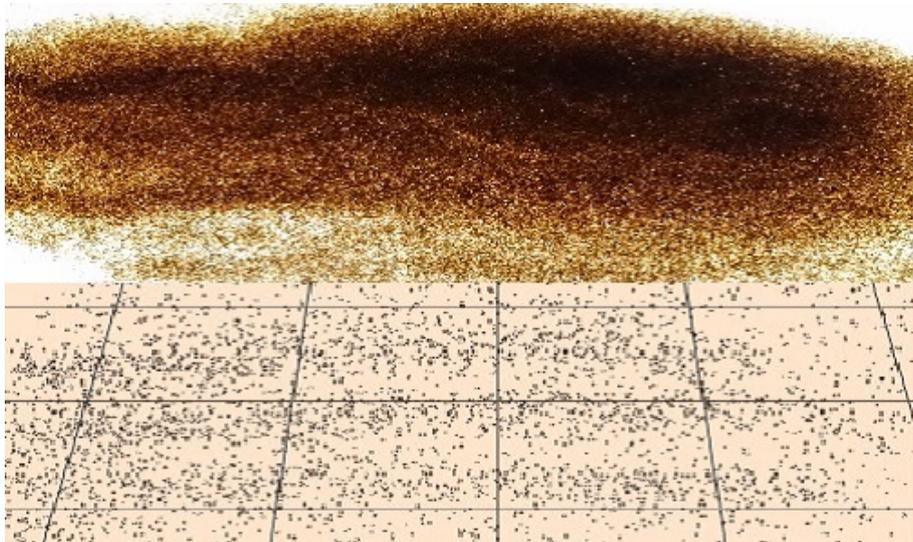


図 4.8 砂を撒く技法の比較

#### 4.1.2 実際のサンドアニメーションとの比較

実際のサンドアニメーションのとの比較においては 2.2 節で記述した技法の中から説明する。

砂を退かす技法については 3.5 節で説明したように輪郭をはっきりさせて線を描くことが可能であるため、再現できている。次に砂を置く、砂を撒くという 2 つの技法については 3.6 節で説明したように新たに砂を追加する機能を実装しており、低い位置で砂を追加することで砂を置く技法、高い位置で砂を追加することで砂を撒く記号を再現している。

図 4.9 は実際のサンドアニメーションの作品である人の横顔と、同じようにシステムで描いた人の横顔の比較画像である。



図 4.9 サンドアニメーションとシステムで描いた絵の比較の比較

## 4.2 試験者からの意見、感想

本研究では 4 人のユーザーから提案技法を実装したシステムを使用してもらい、その際に得られた意見などから本システムでサンドアニメーションの疑似体験を行うことができるか、今後の改善点になる内容を以下にまとめる。

- 砂を動かすという体験が面白かった。
- 波などの表現ができたのが面白かった。

- 細かな部分を描くのが難しい。
- 慣れるまで砂の高さの感覚が掴みにくい。
- 指1本で書こうとすると他の指も反応してしまう。

### 4.3 考察

「砂を動かすという体験が楽しかった」などの意見から、サンドアニメーションに触れる楽しさについて感じてもらうことができたと考えられる。また、実装した実際のサンドアニメーションの技法についても一部再現できたと分かる。

しかし改善点も多く、「慣れるまで砂の高さの感覚が掴みにくい」という感想については Leap Motion 自体が空中で手を使うことを想定したデバイスであるため、透明板を上を設置するなど、インターフェースの改善をする必要があると考えられる。次に「細かい部分を描くのが難しい」という感想については、前の質問にもある高さの感覚が掴みにくいという理由もあると考えられる。影響する範囲や形を含めて改善する必要がある。次に「指1本で書こうとすると他の指も反応してしまう」という感想についてであるが、この感想においてもやはり高さが分かりにくいという点があると考えられる。指を低くしすぎると砂が他の指の影響範囲に入ってしまうため、予想していない部分の砂が動いてしまうと考えられる。

上記の3つを考え、全てにおいてインターフェースの改善が必要であると考えられる。透明板を置くなど簡素なものであっても反応する位置が分かるインターフェースを用意することが目下の改善点である。

他にも

- 指の位置情報が飛んでしまう
- 端の方に行くとブレが強くなる

などの感想もあった。位置情報自体は Leap Motion 自体の機能であるためシステムで防ぐことは難しいが、数秒前の状態に戻すなど、デジタルの特徴を利用して対応策を考える必要がある。

本システムにおいても実際のサンドアニメーションの技法である「砂を払げる」「勢いをつけて飛ばす」などの技法の実装はまだできていないため、それらの技法の実装も今後の課題である。

## 第 5 章

### まとめ

本研究では Leap Motion SDK を使用した指先位置の取得と点の描画を利用した砂の表現を用いてサンドアニメーションの疑似体験を行うシステムを作成した。指先に近づくと砂を動かすことができる、新たに砂を追加することができるなど、実際のサンドアニメーションの技法も一部再現することができた。以上のことにより、砂を動かすことで絵を描くというサンドアニメーションの疑似体験を行うという内容を提案することはできたと言える。また、本研究使用したデバイスは Leap Motion のみであり、研究目的とした少ないデバイスを用いた疑似体験を行うシステムを実現できたと言える。

今後の展望としては 4.3 節で記述したようにインターフェースの改善とまだ実装できていない技法の再現がある。他にも絵を描きやすいように人差し指のみを出していたら人差し指の指先にしか反応しない、など手の形と反応する指を関連付ける形でのシステムの改善も考慮に入れる必要がある。

また、失敗した時に数秒前の状態に戻す、などのデジタル特徴を利用した改善も加えることで、より利用しやすいシステムにすることができると考えられる。

# 謝辞

本研究を締めくくるにあたり、多くのご指導、助言を頂きました渡辺先生、阿部先生、学習支援センターの先生方に深く感謝申し上げます。また、分からない部分の相談や検証のためのアンケートにも協力してくれた研究室のメンバーや友人にも本当に感謝しております。プログラミングから論文まで色々な難題もありましたが、皆様のお陰で創成課題の頃から続けてきた研究を無事こうして形にすることができ、とても嬉しく思います。お世話になりっぱなしの一年間でしたが、本当にありがとうございました。

## 参考文献

- [1] ニコニコ動画. 株式会社ニワンゴ. <http://www.nicovideo.jp/>. 参照:2016.12.01.
- [2] 動画共有サービス youtube. グーグル株式会社. <https://www.youtube.com/>. 参照:2016.12.01.
- [3] イラストコミュニケーションサイト pixiv. ピクシブ株式会社. <http://www.pixiv.net/>. 参照:2016.12.01.
- [4] twitter. twitter,inc. <https://twitter.com/>. 参照:2016.12.01.
- [5] Facebook. Facebook,icn. <https://ja-jp.facebook.com/>. 参照:2016.12.01.
- [6] 寺井良太, 中村徳裕, 西尾孝治, 小堀研一. Haptic device を用いた仮想水墨画システムの提案. 電子情報通信学会総合大会講演論文集, 2005.
- [7] Photoshop. アドビシステムズ. <http://seiga.nicovideo.jp/>. 参照:2016.12.20.
- [8] Easy PaintTool SAI. Systemax. <http://systemax.jp/ja/sai/>. 参照:2016.12.20.
- [9] 平岡透, 熊野稔, 浦浜喜一. 反復強調バイラテラルフィルタによる砂絵風画像の生成. 2015 芸術科学会論文誌 Vol.14,No.1,pp.2025, 2014.
- [10] 平岡透, 浦浜喜一. サンドフィルタによる砂絵風画像の生成. 2013 映像情報メディア学会誌, Vol. 67, No.1, pp.J32-J35, 2013.

- [11] 森田 援. 砂絵風画像を生成するノンフォト・リアリスティックレンダリング, 2014.
- [12] 浦正広, 山田雅之, 遠藤守, 宮崎慎也, 安田孝美. サンドアニメーション風の画像生成のためのペイントツールの開発. 映像情報メディア学会技術報告 33,PP.7 22, 2011.
- [13] 浦正広, 山田雅之, 遠藤守, 宮崎慎也, 安田孝美. 複数本バルーンにより構成されるバルーンアートの構造解析. 電子情報通信学会技術研究報告. MVE, マルチメディア・仮想環境基礎,112(474),pp.295 300, 2013.
- [14] 小嶋 要. ドリッピングアートを模倣したデジタル絵画手法の提案. 卒業論文, 東京工科大学メディア学部 ゲームサイエンス・ゲームイノベーションプロジェクト, 2014.
- [15] 武藤 泉. ラテアートのミルクの動きを再現した描画手法の提案. 卒業論文, 東京工科大学メディア学部 ゲームサイエンス・ゲームイノベーションプロジェクト, 2014.
- [16] 寺川 晃司, 浦正広, 中貴俊, 山田雅之, 遠藤守, 宮崎慎也, 安田孝美. テーブルトップインタフェースを用いたバルーンアートのシミュレーション. 電子情報通信学会技術研究報告. MVE, マルチメディア・仮想環境基礎,110(238), pp.37-42, 2010.
- [17] サンドアート パフォーマンスグループ SILT. Silt. <http://www.otomeru.com/silt/history.html>. 参照:2017.1.12.
- [18] 浦正広, 山田雅之, 遠藤守, 宮崎慎也, 安田孝美. テーブルトップインタフェースを利用したサンドアニメーションの生成シミュレーション. 2011 芸術科学会論文誌 Vol.10,No.2,pp.58-67, 2011.
- [19] 浦正広. 創作活動支援のための制作過程の構造解析およびシミュレーションに関する研究. PhD thesis, 名古屋大学, 2012.
- [20] 原健輔, 浦正広, 山田雅之, 遠藤守, 宮崎慎也, 安田孝美. サンドアニメーションの制作技法とそのシミュレーション. 第 25 回 NICOGRAPH 論文コンテスト論文集, I-3 (2009.10), 2009.