

パーティクルエフェクトのための 3次元流れ場のVRスケッチデザインシステム

井上弦^{†1} 三武裕玄^{†2} 五十嵐悠紀^{†3}

概要: パーティクルの動きをデザインする際、流れ場のような位置に応じて動きが決まる手法は、調整後の動きが想像しやすいという利点がある。流れ場を調整するには大域的な流れと細部の流れどちらも調整できる必要があるが、従来システムでは大域的な流れの調整に手間がかかったり、どのパラメータをどの程度変更すれば望む流れを得られるのかわからなかったりと課題がある。本研究では、パーティクルの動きを決定する3次元の流れ場をVR空間中でスケッチするデザインシステムを提案する。ユーザは、大域的な流れを調整できる特異点の配置と、細部の流れの調整ができるブラシを併用することで、VR空間中に存在するベクトル場を直観的に調整できる。

1. はじめに

近年の3Dゲームや映像の制作において、パーティクルエフェクトによる演出は欠かせない存在となっている。作品内で作りたい視覚的演出を実現するためには、パーティクルの動きを操作者が思い通りに調整できるようなシステムが必要である。

パーティクルの動きを調整する手法として、UnityのParticle System[1]のようにパーティクルが発生してから経過時間に応じて速度が変動する手法と、VectorayGen[2]のように流れ場を定義し、パーティクルの位置に応じて速度が変動する手法に大別することができる(図1)。以下では、前者を時間依存手法、後者を位置依存手法と呼称する。

位置依存手法では、特定の位置でのベクトルがそのままパーティクルの動く方向となるため、調整後の動きが想像しやすいという利点がある。

位置依存手法で使用される流れ場を編集するシステムには、大域的な流れと細部の流れの両方を調整できる機能が必要である。VectorayGenなどの従来システムで主に使用されるパラメータ入力による流れ場の編集方法では、細部の流れの調整は難しい。さらに多種多様なパラメータが存在するため、所望の動きを得るための適切なパラメータ設定が困難である。Draco[3]のようなスケッチインタフェースで流れを直接的に書き込む手法であれば、細部の調整を行いやすく、想像した通りの流れを作ることが容易となる。その一方で、大域的な流れの調整は手間がかかるという問題がある。

本研究では、パーティクルの動きを決定する3次元流れ場をVR空間中で編集するシステムを提案する。特異点を空間中に配置することで大域的な流れを作り、流れを書き込むブラシを用いることで、細部の流れの調整も行うこと

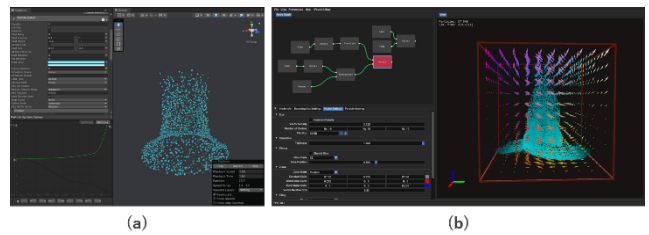


図1 既存システムの実行画面。(a) UnityのParticle System。左下のグラフは、横軸が寿命で縦軸は上方向にかかる力を表している。(b) VectorayGen。

ができる。VR空間中でコントローラを用いて直接的に流れ場を編集するため、時間依存手法やパラメータ入力による流れ場の編集と比較して直観的である。

2. 関連研究

2.1 パーティクルの動きの制御手法とスケッチインタフェース

時間依存手法が主に使われるパーティクルデザインシステム[1][4][5]では、細かな速さの調整がしやすいという利点がある。例えば徐々に減速するような動きをつけたい場合、減速させたいタイミングに応じて、速さを調整することができるため、望んだ減速運動を得やすい。

位置依存手法が主に使われるパーティクルデザインシステム[2][6]では、特定の位置でのベクトルがそのままパーティクルの速度となるため、調整後の動きが想像しやすいという利点がある。そのため、望んだ通りの方向にパーティクルを動かしやすい。例えば何度も曲がるような動きをつけたい場合、曲がってほしい位置のベクトルを任意の方向に変更すると、その通りにパーティクルを動かすことができる。

^{†1} 明治大学大学院

^{†2} 明治大学

^{†3} お茶の水女子大学

手法に限らず、これらの既存ソフトウェアでは、パーティクルを制御するパラメータが非常に多く(図 1)どのパラメータをどの程度変更すれば望む動きを得られるのかわかりづらいという課題がある。さらに位置依存手法のシステムでは、方向の調整がしやすいのが利点であるが、キーボードでの数値入力によるパラメータ調整では、作りたい流れが細くなるほど手間が増え、難しくなるという課題がある。

提案システムでは、3次元の流れ場に対して「流れブラシ」という流れを書き込むブラシを使用することで、パーティクルの動きをデザインする。位置依存手法とスケッチインタフェース[7]を採用することで、方向の調整がしやすく、細かな流れ場であっても複雑なパラメータ設定をすることなく直観的にユーザが望む結果を得ることができる。

2.2 スケッチによる流れの制御

スケッチすることで流れをデザインする研究システムは他にも存在する。Kaziらはイラストにテクスチャ、エミッター、モーションパスを描き、パーティクルによるアニメーション効果を与えるシステムを提案した[3]。Huらはユーザのスケッチ入力からcGANによって生成された速度場に基づいて流体シミュレーションを行った[8]。松原らはユーザがスケッチしたパスから速度場を生成し、それに沿った炎のシミュレーションを行った[9]。有原らは2次元の煙のシミュレーションに着目し、手描きスケッチからcGANを用いて流体の構造を考慮した速度場を生成する支援システムを提案した[10]。Zhuらは流体回路の形状や接続を設計することができるリアルタイム図解スケッチシステムを制作した[11]。これらのシステムでは、おおよその流れをユーザがスケッチし、ミクロな動きはシステムが補うことで直観的に望んだ流れを得ることができる。しかし、大域的な流れを作るには手間がかかるという課題がある。例えば一点を中心に収束するような流れを作りたい場合、外側からその点に向かって何度もストロークを書く必要があり、この手間は制御範囲が大きくなるほど増える。また、これらの研究では3次元の流れには対応していない。

3次元の流れに対応したスケッチシステムも存在するが、流れではなく、キーフレームを書き込むというアプローチを取っている。YanらはVR環境で液体の初期状態を3Dスケッチし、cGANによって時間経過したスプラッシュ表現を生成した[12]。Panらは、シミュレーションされた流体に対してスケッチで流体の形状のキーフレームを指定し、指定された形状に沿った流れのシミュレーションを行った[13]。これらのシステムでは、キーフレームでしかデザインに関与できないため、ユーザのデザインの柔軟性は高くない。

提案システムでは、ヘッドマウントディスプレイを装着し、VR空間中でコントローラを用いることで、3次元の流

れ場をスケッチする。さらに、特異点を配置することで大域的な流れを手軽に作るができる。制御範囲が大きくなっても、特異点のサイズを変更するだけで対応が可能である。

2.3 ジェスチャーによる流れの制御

3次元の流れを制御する手法として、ジェスチャーを採用した研究も存在する。早川らはLeap Motionを用いて手のモーションデータを取得し、特定のモーションと紐づいている流体シミュレーションのパラメータを調整することで、直接的に3D流体シミュレーションの制御を行った[14]。AroraらはVR空間中で鑑賞する3Dアニメーションの制作ツールを開発した。アニメーターにVR空間中でのアニメーション制作に必要なジェスチャーを考案してもらい、それを元に開発したプロトタイプにはジェスチャーでパーティクルを制御する機能が存在する[15]。Xieらは深度センサから取得したユーザのジェスチャーを入力とし、ジェスチャーと紐づいている花火のパラメータを調整することができるシステムを開発した[16]。これらのシステムでは、大域的な流れを簡単に作るができるが、細部の流れの調整は困難である。提案システムでは、流れブラシで細部の流れの調整も可能である。ジェスチャーベースの制御では、繊細な動きの表現や認識に限度があるため、パラメータの微調整が難しいという課題もある。コントローラを用いれば、パラメータの微調整がスティック操作で可能となり、ユーザのデザイン自由度が高くなる。そのため提案システムでは、インタフェースにジェスチャーでなく、コントローラを採用した。

2.4 特異点による流れの制御

Jakubらは基礎となるエネルギーをモデル化し、流れ場を制御するモデルを提案した[17]。本研究では、この基礎となるエネルギーを特異点と呼称する。特異点は流れ場に配置し強度を調整することで流れ場を制御することができる。影響範囲を変更することで、大域的な流れも細部の流れも調整を施すことが可能である。

特異点をスケッチインタフェースに落とし込んだ研究が存在する。Xingらは特異点のモーションパスを描くことで時間変化する流れ場を生成し、イラストをアニメーションさせるインタラクティブスケッチシステムEnergy-Brushesを提案した[18]。Energy-brushesでは固定されたイラストをアニメーションさせるシステムのため、2次元の流れ場が対象であり、収束、発散させる特異点がない。本研究では、3次元上でのパーティクルの動きを制御対象とするため、VR空間中で3Dスケッチを行うことで3次元の流れ場を制御し、空間中を移動するパーティクルの制御が目的のため、収束や発散させる特異点を用意するなど3Dパーティクルに最適化したシステムとなっている。

3. 提案システム

ユーザは、ヘッドマウントディスプレイを装着し VR 空間中で、コントローラを用いて流れ場のデザインを行う。VR 空間中には立方体状の流れ場(図 2a)が存在し、これに対して特異点の配置(3.2.1 節)と流れブラシによる塗り(3.2.2 節)を行い、編集を施すことで 3 次元流れ場のデザインをする。システムはリアルタイムで動作し、ユーザが流れ場に編集を与えると即座にパーティクルの動きに反映される。パーティクルはランダムにベクトル場内で発生し、3 秒経過すると消えるよう設定されている。システム実行中のユーザの視点を図 2 に示す。

システムは Unity で実装し、バージョンは 2021.3.22f1 を使用した。

3.1 ユーザインタフェース

左手のコントローラに、UI が配置されているキャンバス(図 2b)が追従しており、右手のコントローラから出ているレーザーポインター(図 2c)で照準を合わせて人差し指のトリガーボタンを押すことで、これら进行操作することができる。キャンバス上には流れブラシ、特異点の選択ボタンと流線の表示非表示切り替えチェックボックスが用意されている(図 3)。

右手のコントローラには、現在選択している球状の特異点か流れブラシが追従しており(図 2d)、これをベクトル場内に持っていき中指のトリガーボタンを押すことで、効果を適用できる。右手のコントローラのスティックの上下で球の大きさを変更できる。

3.2 3次元流れ場のデザイン

ユーザは、作りたい流れに応じて特異点の配置と流れブラシを使い分け、立方体内に存在する $16 \times 16 \times 16$ のベクトル場を編集する。内部的には、特異点配置用の場と流れブラシ用の場を別に保存し、毎フレーム加算した結果の場が流れ場となり、パーティクルの速度に適用される。ベクトルのスカラー値には上限があり、上限に達している状態で加算を行った場合は加算した結果の方向が適用され、スカラー値は上限の数値がそのまま適用される。

3.2.1 特異点の配置

特異点は、配置することで特異点の範囲内にあるベクトル場に対して、特異点の種類ごとに固有のベクトル場を加算することができる機能である。ユーザは、任意の特異点を選択し、配置したい場所まで持っていき、右手中指のトリガーボタンを押すことで特異点を配置できる。特異点は球の形をしており、中心が最もベクトル場に与える影響の強度が強く、外に行くほど減衰する。

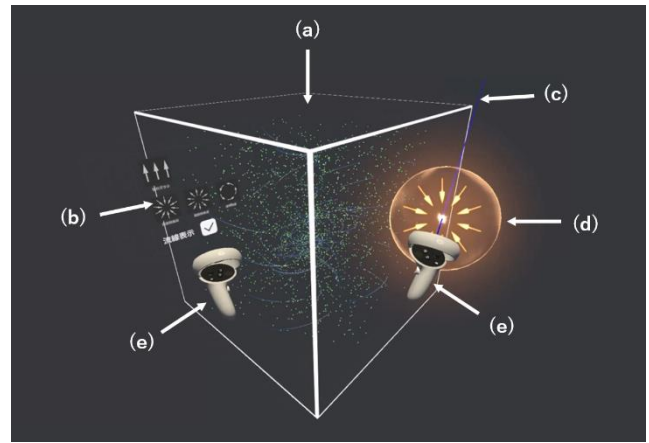


図 2 システム実行中のユーザの視点

(a) 流れ場 (b) キャンバス (c) レーザーポインター
(d) 選択中のツール (e) ユーザが操作するコントローラ



図 3 UI が配置されているキャンバス



図 4 3種類の特異点

(a)収束特異点, (b)発散特異点, (c)渦特異点.

特異点は、収束、発散、渦という 3 種類が用意されており、それぞれベクトル場に与える影響が異なるため、ユーザは求める流れ場に応じて適宜配置する特異点を選択する(図 4)。収束特異点は、流れを中心に向かわせる特異点で、発散特異点は、流れを中心から外に向かわせる特異点である。渦特異点は、ユーザが自由に設定できる回転軸を中心の流れを回転させる特異点である。特異点は一度配置したあとでも、配置位置、大きさ、強度を自由に変更できるため、パーティクルの動きを見ながら調整することが可能である。

ユーザは、特異点を配置することで、大域的な流れを手軽に設定することができる。

3.2.2 流れブラシによる塗り

流れブラシは、塗った軌道上のベクトル場に対して、任意方向のベクトルを加算することができるブラシである。流れブラシを選択中に右手中指トリガーボタンを押すとブラシが発光し、この状態でコントローラを移動させることで塗ることができる(図5)。塗った箇所が視覚的にわかるように一定時間塗った箇所に対してエフェクトを発生させている(図5b)。この塗った範囲内にあるベクトルに対して、ブラシ内に表示されている矢印の方向に、設定したスカラー値の分だけベクトルが加算される。

ユーザは、流れブラシを使用することで、特異点だけでは表現できない細部の流れのデザインを実現できる。

3.3 流れ場の可視化

流れ場は流線により可視化を行う。図6,7などで見られる流れ場の中にある青白い線が流線である。ユーザが流れ場に変更を与えると、その変更がリアルタイムで流線に反映される。流線の表示非表示はいつでも切り替え可能である。

4. 流れ場のデザイン例

提案システムを用いてデザインした流れ場の例を図6,7に示す。

図6は渦特異点を2つ設置、収束特異点を1つ設置し、その間を流れブラシで塗ることで、回転しながら1つの点に集まっていくという流れ場を作った様子である。

図7では、渦特異点を流れ場全体を覆うように1つ設置、発散特異点を真ん中上部に1つ設置、全体的に上方向に流れるよう流れブラシを使うことで、回転しながら舞い上がるようなパーティクルの動きを表現する流れ場の制作例を示す。この流れ場でデザインされたパーティクルの動きに対し、テクスチャを煙らしいものに変更し、エミッターを中央下部にのみ設置、パーティクルの色や大きさを調整することで竜巻のようなエフェクトを作ることができると想定される。

5. ユーザスタディ

4名の大学院生に提案システムを用いて自由に流れ場をデザインしてもらい、コメントをもらった。

機能面について「1つ前の操作に戻る機能がほしい」「場のリセットがしたい」という要望があった。特に場のリセットについては4人中3人から言及があったため今後実装を行う。

操作性については「人差し指と中指のトリガーのどちらがボタンの選択でどちらがブラシの適用なのか迷う」という意見があった。これについては、状況によって操作でき

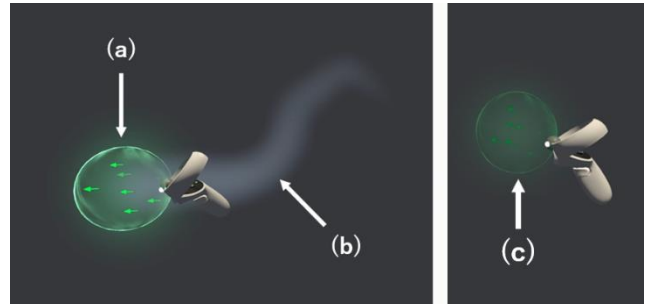


図5 流れブラシでの塗り。(a) ブラシ適用中の発光している流れブラシ (b) 塗った箇所を可視化しているエフェクト (c) 未使用時の流れブラシ

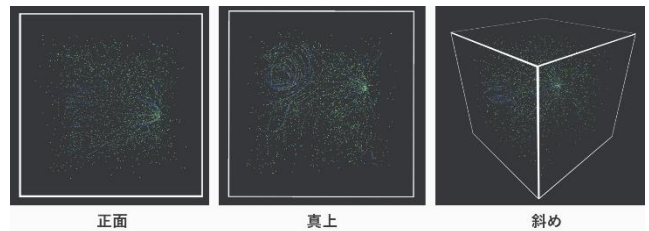


図6 2つのポイントで回転しながら1点に集まっていく流れ場の制作例

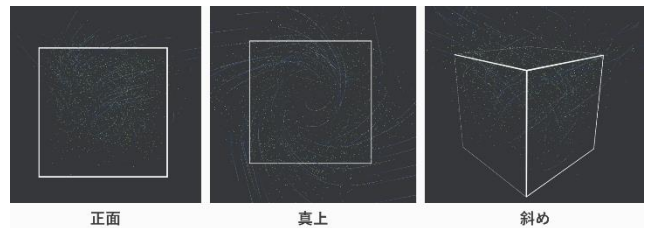


図7 竜巻のように回転しながら舞い上がり外側に広がっていく流れ場の制作例

るコントローラのボタンをハイライトすることで対応する。具体的には、キャンバスにポインターが向いているときは、ボタン選択をするための人差し指トリガーをハイライトし、流れ場の中にブラシがあるときは、ブラシを適用するための中指トリガーボタンをハイライトして補助する。

流れ場の可視化については「流線がリアルタイムで形状変化するため、どのように場が変化したかわかりやすい」という肯定的な声があがった一方で、「流れ場の認識に少し時間がかかる」と感じる被験者もいた。認識しづらいと感じた被験者は「パーティクルを大きくしたり、密度を上げたり、色幅を持たせたりすることで改善を期待できる」というコメントをしており、主に流線ではなくパーティクルの動きを見て流れ場の認識を試みていた。これらのことから、流れ場を認識しやすい流線やパーティクルの大きさなどのパラメータは人によって違うと考えられる。そのため、流線やパーティクルのパラメータをユーザ自身で流れ場を認識しやすい設定に変更できるような機能が必要だと考える。

その他、「発散の扱い方が難しい、動きがどうなるのか想

像しにくい」というコメントや「電磁場など目に見えない場の動きを理解するのに使えそう」「子どもが遊ぶためのツールとして良さそう」など応用例についてのコメントがあった。

6. まとめと今後の課題

本研究では、パーティクルの動きを決定する3次元の流れ場をVR空間中でデザインするシステムを提案した。ユーザは、特異点の配置と流れブラシを併用することで、大域的な流れも細部の流れも直観的に調整できる。

パーティクルエフェクトをデザインするには、テクスチャ、シェーダ、エミッターの形状、色や大きさなどを設定することでデザインを構築していくが、その中に動きを設定するという工程が存在する。提案システムを用いることでこの工程を従来システムよりもわかりやすく、自由度高く行うことができると期待する。

ユーザスタディの結果、場のリセット機能や流線とパーティクルのパラメータ調整機能などの追加が必要であるとわかったため、今後はそれらの実装を進めていく。

参考文献

- [1] Unity Particle System, https://docs.unity3d.com/ja/current/Manual/class-ParticleSystem.html(最終閲覧日: 2023年12月21日)
- [2] VectorayGen, https://jangafx.com/software/vectoraygen/(最終閲覧日: 2023年12月21日)
- [3] Rubaiat Habib Kazi, Fanny Chevalier, Tovi Grossman, Shengdong Zhao, and George Fitzmaurice. Draco: bringing life to illustrations with kinetic textures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2014.
- [4] Unreal Engine Niagara, https://docs.unrealengine.com/4.27/ja/RenderingAndGraphics/Niagara/(最終閲覧日: 2023年12月21日)
- [5] Adobe After Effects CC Particle Systems II
- [6] Unity Particle System Force Field, https://docs.unity3d.com/ja/2018.4/Manual/class-ParticleSystemForceField.html (最終閲覧日: 2023年12月21日)
- [7] Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka. Teddy: a sketching interface for 3D freeform design. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques (SIGGRAPH '99)*.
- [8] Hu Z, Xie H, Fukusato T, Sato T, Igarashi T. Sketch2VF: Sketch-based flow design with conditional generative adversarial network. *Comput Anim Virtual Worlds*, 2019.
- [9] 松原 凜太郎, 佐藤 周平, 高 尚策, 唐 政. "ペイント入力を用いた炎の流れのガイドに関する一実験". *NICOGRAPH 2022 Conference track*, No. S-13.
- [10] 有原 啓介, 謝 浩然, 佐藤 周平, 宮田 一乗. "ラグランジアンコヒーレント構造によるスケッチベース煙デザイン手法の提案". *コンピュータグラフィックスとビジュアル情報学*, 2022-CG-185(4), 2022, pp.1-11.
- [11] Bo Zhu, Michiaki Iwata, Ryo Haraguchi, Takashi Ashihara, Nobuyuki Umetani, Takeo Igarashi, and Kazuo Nakazawa. Sketch-based Dynamic Illustration of Fluid Systems. *ACM Trans. Graph.* 30, 6 (December 2011),
- [12] Guowei Yan, Zhili Chen, Jimei Yang, and Huamin Wang. Interactive liquid splash modeling by user sketches. *ACM Trans. Graph.* 39, 6, Article 165 (December 2020),
- [13] Zherong Pan, Jin Huang, Yiyong Tong, Changxi Zheng, and Hujun Bao. Interactive localized liquid motion editing. *ACM Trans. Graph.* 32, 6, Article 184 (November 2013).
- [14] 早川雄登, 藤代一成. "直接操作による3D流体シミュレーションの制御". *情報処理学会研究報告*, Vol.2017-DCC-17, No.24, 1-6.
- [15] Rahul Arora, Rubaiat Habib Kazi, Danny M. Kaufman, Wilmot Li, and Karan Singh. MagicalHands: Mid-Air Hand Gestures for Animating in VR. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*.
- [16] Xie, H., Xie, D., Miyata, K. Body2Particles: Designing Particle Systems Using Body Gestures. In *Entertainment Computing – ICEC 2020*.
- [17] Jakob Wejchert and David Haumann. Animation aerodynamics. *SIGGRAPH Comput. Graph.* 25, 4 (July 1991), 19–22.
- [18] Jun Xing, Rubaiat Habib Kazi, Tovi Grossman, Li-Yi Wei, Jos Stam, and George Fitzmaurice. Energy-Brushes: Interactive Tools for Illustrating Stylized Elemental Dynamics. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*.