

葉や花を対象とした動的プロジェクションマッピングの自動生成

末吉, 知樹,

<https://hdl.handle.net/2324/6787621>

出版情報 : Kyushu University, 2022, 博士 (芸術工学) , 課程博士
バージョン :
権利関係 :



**葉や花を対象とした
動的プロジェクションマッピングの自動生成**

**Automatic Generation of Dynamic Projection Mapping
for Leaves and Flowers**

末吉 知樹

Tomoki Sueyoshi

2023 年 3 月

概要

本研究では、人の手や風等によって動き変形する植物へのプロジェクションマッピング自動生成システムの開発を目指す。様々なシーンにおける植物への投影を実現するため、本論文では4つの手法を提案し、各手法の結果を検証する。また、本手法を用いて、複数の植物の葉に投影を行い、作品を制作する。

プロジェクションマッピングは、様々な立体物を投影面として、映像投影によって対象物体の見た目を変えることができる技術である。一方、植物の葉や花は、外力によって動き変形する物体であり、プロジェクションマッピングの投影対象としては、投影映像の位置合わせが非常に困難である。そこで本研究では、動的で可変な葉や花を追跡し、自動で投影映像の位置あわせを行う。また、植物に投影するには、植物自身が持つ多様な形状への対応やインタラクションによる遮蔽、多数の葉の取得と高速な追跡など、課題が多くある。本研究では、様々なシーンにおける植物への投影を実現するため、段階的に目標を設定し、解決する手法を提案する。また、作品制作においては、生物発光のように植物自身が発光しているような表現を目指し、投影対象の形状を用いたエフェクトアニメーションの生成を行う。

本論文は、9章で構成される。

第1章では、プロジェクションマッピング技術の活用事例を紹介しながら、社会におけるプロジェクションマッピングの位置づけを述べる。また、本研究が大きく影響を受けた作品を紹介し、プロジェクションマッピング制作における主な課題を挙げ、本研究の目的と取り組む内容の概要を述べる。最後に、第2章以降の本論文の構成を述べる。

第2章では、本研究に関連する研究や事例、作品を挙げ、本研究の目的をより詳細に明らかにする。まず、プロジェクションマッピングの研究と事例を紹介し、比較することで、プロジェクションマッピング分野における本研究の位置づけを述べる。つづいて、本研究と画像処理手法の関連について事例を挙げながら述べる。次に、植物を用いたメディアアートの事例を挙げ、本研究で目指す表現について述べる。最後に本研究の目的をまとめる。

第3章では、本研究で提案する4つの植物への動的プロジェクションマッピング生成手法において、共通するシステム構成と手法の流れについて述べる。

第4章から第7章では、4つの手法について各章で述べる。それぞれ手法の詳細を述べ、実験の結果や作品の評価に基づいて考察する。

第4章では、楕円形の葉を投影対象とした動的プロジェクションマッピング生成手法について述べる。この手法は、本研究の初期段階として、投影対象を楕円形の葉に限定し、投影を実現することを目指す。主に、事前準備による映像の位置合わせ手法、葉の輪郭上の最も遠い2つの頂点を求める追跡手法、対象形状からのエフェクトアニメーション自動生成方法を提案する。

第5章では、投影対象を様々な形状の葉や花に拡大し、人の手等の接触にも頑健な動的プロジェクションマッピング生成手法を提案する。各投影領域について、2つのスケールにおけるテンプレートマッチングを行った上で、全体の形状を維持する追跡を行う。

第6章では、植物へのインタラクティブな動的プロジェクションマッピング生成を目指す。手で触れる等のインタラクションによる遮蔽を考慮した追跡手法を提案する。投影領域の輪郭の各頂点において、追跡とともに遮蔽判定を行い、遮蔽されている点は他の頂点との位置関係から再計算する。さらに、タッチセンサを組み合わせたシステムを実装し、人の手によるインタラクションが発生した時、投影されたエフェクトアニメーションがインタラクティブに変化するコンテンツを作成する。

第7章では、大量の植物への動的プロジェクションマッピング生成を目指す。投影対象を楕円形の葉に限定し、投影領域を直感的に選出するユーザーインターフェース、楕円フィッティングをベースとした高速な追跡手法を提案する。また、投影対象同士の間を考慮することで、各投影領域を頑健に追跡する。さらに、植物の枝に沿った動きや向きを、多数の領域のエフェクトアニメーションに適用するため、枝にあたる線をマウス入力で描画できるユーザーインターフェースを作成する。

第8章では、第4章から第7章で提案した4つの手法について比較を行い、各手法の汎用性と追跡における処理時間の速さ、インタラクションに対するロバスト性などを検証する。

最後に第9章において、第4章から第8章で得られた結果と考察から本研究の結論をまとめる。

本研究では4つの手法を提案し、葉や花への動的プロジェクションマッピングを実現した。本研究をさらに発展させ、様々なシーンに対して1つの手法で対応することが将来課題である。また、本手法を用いることで、イルミネーションやインテリアなど、様々な応用例が期待できる。

目次

概要.....	2
第1章 はじめに	5
第2章 背景	9
2.1 プロジェクションマッピングにおける研究と事例.....	10
2.2 本研究と関連する画像処理手法	18
2.3 植物を用いたメディアアート	19
2.4 本研究の目的	20
第3章 手法概要	21
3.1 システム構成	22
3.2 手法の流れ	23
第4章 手法1：楕円形の葉への DPM	24
4.1 手法詳細	25
4.2 結果考察	33
4.3 作品展示	37
第5章 手法2：接触を考慮した様々な形状の植物への DPM	38
5.1 手法詳細	39
5.2 結果考察	44
第6章 手法3：様々な形状の植物へのインタラクティブな DPM	47
6.1 手法詳細	48
6.2 結果考察	56
第7章 手法4：大量の楕円形の葉への DPM	59
7.1 手法詳細	60
7.2 結果考察	66
第8章 本手法で提案する4手法の比較と考察	72
8.1 追跡手法の比較と考察	75
8.2 エフェクトアニメーションの比較と考察	77
8.3 将来課題	78
第9章 結論	79
謝辞.....	82
付録.....	89

第1章

はじめに

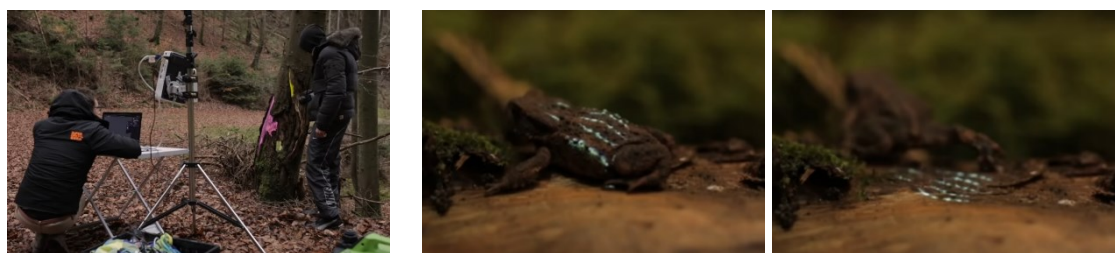
本論文は、植物への動的プロジェクションマッピングにおける筆者の一連の研究をまとめたものである。プロジェクションマッピング (PM) とは、立体物を投影面として、その形状に合わせて映像を投影することで、対象物体の見た目を変えることができる技術である。そのため、PM では、建物や美術、人体など、様々な物体を投影対象として扱うことが可能である。また、投影する物体そのものを違う風に見せることで現実と仮想表現の融合を実現することができるため、複合現実 (Mixed Reality : MR) の技術の一つとして扱われることも多い。近年では、プロジェクターの高解像度化や色域の拡大、輝度の向上もあり、大規模イベント[1][2]やアート作品[3][4]などの成果物としてだけでなく、直感的な情報の伝達を行う手段としても用いられている。例えば、車やスニーカーなどのプロダクトに投影してプロモーションを行う広告[5][6]、歴史的文化財に付加情報を投影して展示するような教育[7]、臓器に映像を投影し手術をガイドする医療[8]など、様々な目的で応用されている。また、低価格のプロジェクターのモデルが増えたことで、個人による PM 作品の制作例も増えている。PM 技術を応用した研究としては、日々の活動や体験を向上させるようなシステム、アプリケーション等も複数提案されている。例を挙げると、ユーザーインターフェース (UI) を手のひらに投影することで、手をデバイス化する手法[9]や、部屋に映像投影することでゲーム体験をディスプレイ外まで拡張するシステム[10]、制作中の絵画や彫刻に直接指示を投影して創作活動を補助するアプリケーション[11][12]などが提案されている。

このように PM は様々な場面で用いられており、作品や情報伝達手段など多様な形で見られる。その中でも本研究が大きく影響を受けている PM 作品である「bioluminescent forest」[3]は、植物や生き物を投影対象とし、生物発光をモチーフにした幻想的な映像作品となっている (図 1)。しかしこの作品では、屋外の森で映像と投影対象の位置合わせを手作業で行う必要があり (図 2 左)、膨大な手間と労力がかかる。また、動く投影対象の追跡を行っていないため、カエルや芋虫などの対象が動くと、投影映像と投影対象の間にずれが生じる (図 2 右)。このように、PM 制作において、投影映像の位置合わせは大きな課題である。特に動く対象への投影を実現するには、最初に位置合わせを行うだけでなく、対象物体の動きや変形をリアルタイムで追跡し、追跡結果にあわせて投影映像も変化させる必要がある。

プロジェクションマッピングは投影対象の位置や形状が変化しない静的 PM と投影対象の位置が動き、その動きに合わせて映像投影を行う動的 PM (Dynamic Projection Mapping: DPM) に大きく分けることができる。また、DPM の中でも映像が逐次自動的に画像処理などで対象を追跡するものとそうでないものがある。後者の例としては、あらかじめ対象物体の動きや位置に合わせて映像を作成し、その映像に合わせて対象が動くというものがある[13]。



図 1 生き物へ投影を行った作品[1].



手作業による映像の位置合わせ

動く投影対象への映像のズレ

図 2 PM 制作時の課題.

本研究では、投影対象が動く場合に逐次追跡をし、さらにその変形に合わせて映像投影領域を変化させる DPM の制作支援システムの開発を目指す。具体的には、葉や花を対象とし、風や人の接触による動きや変形を画像処理で自動追跡する。

また、多様な形状の葉や花、風や接触によるインタラクション、より多数の葉への投影など、様々なシーンが想定される。そのような多様なシーンに対応するため、本研究では段階的に設定した 4 つの目標に対する手法を提案する。1 つ目の手法では、本研究の初期段階として、楕円形の葉を投影対象とした DPM 生成手法を提案する。次に、2 つ目の手法では、投影対象を様々な形状の葉や花などに拡大し、追跡する。さらに、人の素手による接触にも頑健な追跡手法を提案する。3 つ目の手法では、人の接触の際に物体が手などで物体が隠れ、遮蔽が起こる場合の追跡の頑健性を高めるため、遮蔽領域を考慮して追跡する手法を提案する。同時に、タッチセンサを用いたインタラクティブコンテンツを実装した。4 つ目の手法では、より大規模な DPM を実現するため、多数の葉を高速に追跡する手法を提案する。それぞれ投影実験や作品制作を行い、比較、検証する。

本論文の構成について述べる．まず，第 2 章では，関連事例や先行手法を挙げ，比較しながら，本研究の位置づけと目的を述べる．第 3 章では，本研究で提案する 4 つの DPM 生成手法で共通するシステムと手法の流れについて述べる．つづいて，第 4 章から第 7 章では，4 つの手法について，それぞれ手法の詳細を述べ，実験の結果や作品の評価に基づいて考察する．提案した 4 つ手法について，第 8 章では比較を行い，各手法の汎用性と追跡における処理時間の速さ，インタラクションに対するロバスト性等を検証する．最後に第 9 章において，得られた考察から本研究の結論をまとめる．

第2章

背景

本章では、まず PM における研究と事例を紹介し、PM 分野における本研究の位置づけを述べる。次に、本研究に関連する画像処理手法を挙げ、本研究で提案する手法との関連を述べる。つづいて、植物を用いたメディアアートや研究を挙げ、本研究の成果として作成する表現について述べる。最後に、本研究の目的をまとめる。

2.1 プロジェクションマッピングにおける研究と事例

PM を実現するためには、様々なステップがあるが、最終目標は指定した投影面に空間的かつ時間的にずれることなく、正確な色で映像を投影することである。最終目標を達成するためのアプローチとしては、ソフトウェアとハードウェアの 2 つの研究分野に大きく分けられる[14]。

ソフトウェアの研究では、カメラ等のセンサを用いて取得した投影面に整合するように、映像の位置や色を変更することを目的としている。そのため、主な課題として、投影対象への映像の位置合わせ、DPM での動的対象への追跡と投影、投影映像の色補償がある。まず、投影対象への映像の位置合わせを達成するためには、プロジェクターとカメラを用いた手法が主に提案されている。次に、DPM においては、動き、変形する物体に対して追跡と投影を行うため、マーカーや形状入力、複数のカメラ等を用いた手法が複数提案されている。色補償では、様々な模様や質感を持つ投影面に対して、入力映像を正確に投影するため、投影前の映像の色に補正を行う試みがなされている。例えば、白と黒で構成される模様の面に投影すると映像と模様が混ざってしまうが、白い領域への投影映像は暗く、黒い領域への投影映像は明るくすることで、模様の見えを打ち消すことができる。

一方、ハードウェアの研究では、プロジェクターの投影映像における色域や輝度、解像度、焦点深度、速度等を向上させ、表現幅を拡大することを目的としている。投影映像の表現幅向上を目指す手法としては、プロジェクター単体の開発だけでなく、複数台のプロジェクターやカメラを用いた解決手法も提案されている。例えば、1 台のプロジェクターでは輝度や色域、解像度に限界があるが、2 台のプロジェクターの投影映像を重ね合わせることで、コントラストが高い映像やより精細な映像を提示できる[15][16]。また、視点方向に応じて映像が変わる Light Field 投影の試みもあり、PM への応用としては構造色のような発色表現を投影面に提示することもできる[17]。

本研究で主に取り組む課題は、ソフトウェアの研究に分類される、投影対象への映像の位置合わせと DPM である。特に DPM は、投影対象が剛体か非剛体かによって手法が大きく分けられる。そこで本節では以下のように、

- A) 2.1.1 項) 投影対象への映像の位置合わせ
- B) 2.1.2 項) 剛体への DPM
- C) 2.1.3 項) 非剛体への DPM
- D) 2.1.4 項) インタラクティブな PM の研究や
- E) 2.1.5 項) PM 作品

を紹介し、本研究の位置づけを明らかにする.

2.1.1 投影対象への映像の位置合わせ

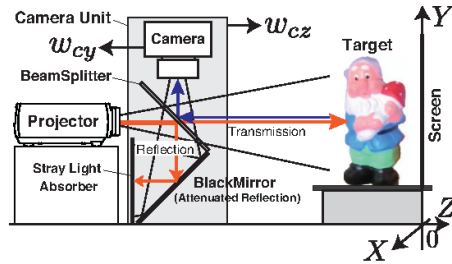
コンピュータで生成した映像を現実空間で指定した投影対象の領域にずれることなく投影することは、PM の基本要素である. PM ではカメラで投影シーンを取得し、プロジェクターで対象に投影するプロジェクター - カメラシステム (**Projector-Camera: ProCam**) [14]が使用される. ここでは 2 つの位置合わせが必要となる. 一つはプロジェクターとカメラなど、異なるデバイス間で同一の座標に合わせる処理であり、本稿では **image registration** (または **registration**) と呼ぶ. もう一つは、デバイス間のイメージレジストレーションを行った上で投影対象領域に映像の位置や形状を合わせる処理である. 本稿ではこれを **mapping** と呼ぶ.

投影対象が平面の時、ProCam 間の **registration** を行う簡単な方法として、射影変換による変形がある. プロジェクターとカメラの領域内で対応する 4 点を用いて射影変換行列を作成し、カメラの取得画像に適応する. しかし、投影面が立体の時、投影対象に対する相対的位置や向きによって、プロジェクター方向とカメラ方向から見た投影シーンでは大きく異なる. そのため、ProCam の相対位置と方向を計算し、位置と方向のずれを補正する. また、プロジェクターとカメラには、レンズの歪みや焦点距離等の内部パラメータがある. そのため、高精度な **registration** を行うには、内部パラメータを計算し、歪み等を補正する.

ProCam 間の **registration** を行う手法を紹介する. Yamazaki らの手法[18]では、プロジェクターで構造化光とよばれる縞模様などのパターン光を立体物に投影し、カメラで取得することで、ProCam 間のピクセル単位の対応関係を求める. この対応関係からプロジェクターとカメラのレンズの歪み等の内部パラメータ、相対的位置関係を計算し、立体物の形状も再構成できる.

このほか、ProCam 間の **registration** 手法は複数提案されている[19][20][21]. Amano らの手法[19]では、入射光を 2 方向に分割するビームスプリッターによって、プロジェクターとカメラの軸を揃えた上で、対象物体に投影したグリッドの取得と計算を繰り返すことで、高精度な **registration** を実現している (図 3 左). iLamps[20]では、傾斜センサ

によって取得した ProCam システムの向きと、カメラで捉えた画像の歪みを組み合わせることで、周囲の 3 次元空間を再構成し、歪みが少ない映像を提示する (図 3 右)。また、Willi らの手法[21]では、複数台のプロジェクターとカメラにおける構造化光投影によって、registration を高速かつ高精度で実現している。



Amano らの手法[19]



iLamps[20]

図 3 ProCam システムにおける位置合わせ手法。

また、投影映像の幾何補正によって投影対象への mapping を行う事例として、MadMapper 等のクリエイター向け PM 制作支援ソフトウェア[22][23][24]もある。これらは、映像をマウสดラッグなどの入力により、UI 上で変形し、立体物の各面などへ mapping を行えるが、手作業である。一方で、PM 制作を半自動で支援する商用デバイスとして、LIGHTFORM[25]がある。LIGHTFORM は、投影した構造化光スキャンをカメラで取得することで静止物体の形状を取得する手法[18]を用いており、取得形状に対して任意の動画像を自動的に mapping し、投影することができる (図 4)。

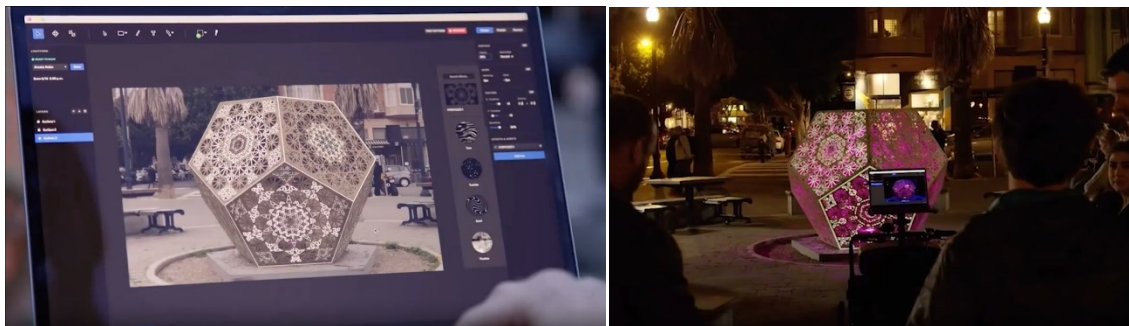


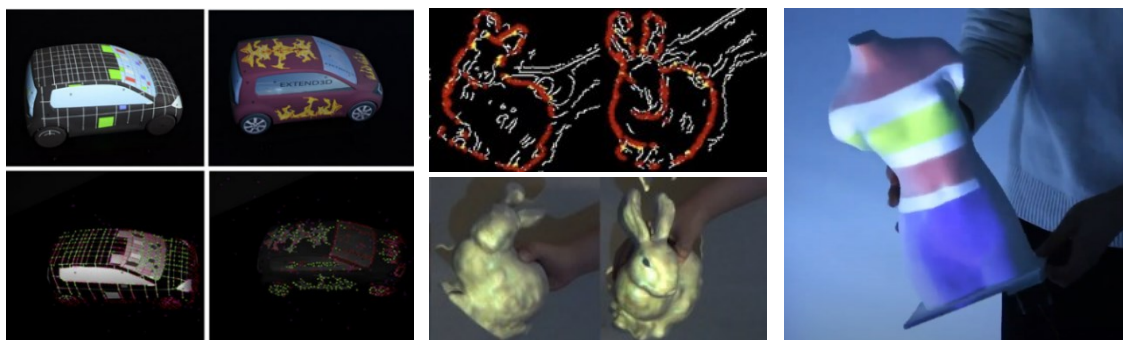
図 4 LIGHTFORM[25].

本研究においても、ProCam システムを用いるが、registration に必要なプロジェクターの投影領域とカメラ取得領域の対応関係は事前に簡易的に求め、投影時に適切な幾何補正を適用する。また投影時は、自動で抽出された領域から、追跡、投影したい領域をマウスの簡単な操作によって選出できる UI を作成する。

2.1.2 剛体への動的プロジェクションマッピング

動的で可変な対象への投影を実現するには、画像処理などで投影対象を捉え追跡する必要がある。DPM のための追跡手法は、投影対象の種類によって様々な手法が提案されており、本節ではまず剛体への DPM 生成手法について述べる。

任意の 3D オブジェクトの追跡を行う手法は複数提案されている[26] [27] [28] [29] [30] [31]。Sticky Projections[26]や Morikubo らの手法[27]，Pmomo[28]では、投影対象の 3D 形状を事前に入力し、特徴点やエッジ画像、深度カメラで得られた点群と比較することで投影対象の 3 次元的な移動と回転 (6DoF) を得る。これを 3D モデルに適用し、位置や姿勢を合わせることができる (図 5)。



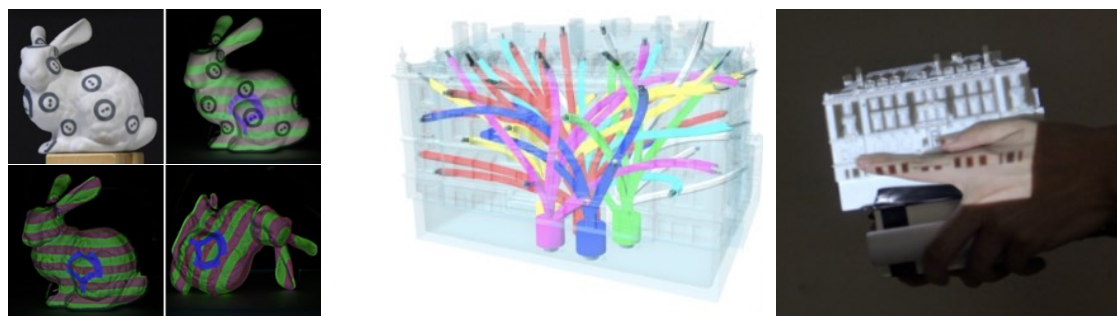
Sticky Projections[26]

Morikubo らの手法[27]

Pmomo[28]

図 5 3D 形状の事前入力を用いた剛体への DPM.

Asayama らの手法[29]や FibAR[30]では、3D プリンターで作成した物体を投影対象としており、対象物体の印刷時に赤外線マーカーや光ファイバーマーカーを適切な位置に埋め込むことで、対象の安定した追跡を実現している (図 6)。



Asayama らの手法[29]

FibAR[30]

図 6 マーカーを用いた剛体への DPM.

2.1.3 非剛体への動的プロジェクションマッピング

人の顔や布、粘土などの変形のある非剛体への DPM 生成手法では、剛体への DPM と同様に位置や角度に加え、形状も追跡する必要がある。

人の顔を投影対象とした事例として Makeup Lamps [31]では、投影対象である顔の三次元モデルを事前に入力することで、表情が変化する顔への位置合わせと投影を高速に行う。紙や布への投影を行った事例として、Narita らの研究[32][33]がある。赤外線マーカーを用いることで、紙や布の形状変化も考慮した投影を実現する。これらの手法[31][32][33]では、事前の投影対象の 3D 形状の入力や追跡マーカーが必要であるが、生き物や液体など、3D 形状の取得やマーカーの貼付が困難な投影対象もある。そのため、事前の形状入力やマーカー無しで投影対象の追跡を行うことは、DPM 研究において大きな課題の一つである。

一方、MIDAS[34]では、事前の形状入力や追跡マーカーなしで、動的で可変な対象への PM を実現している。3 台の赤外線カメラを用いて、対象の法線情報を取得することで、手や粘土、液体などの様々な物体の形状変化を追跡し、マテリアルやテクスチャを変更する表現を行うことが出来る。さらに、画像処理によって、人の手等の接触を検出することで、触れた箇所から映像が変化するシステムを提案している。

また、DPM では、映像のズレの発生を防ぐため、計算速度とハードウェアの通信速度が求められる。Ng らの調査結果では、人間が遅延を認識する遅延時間は 6.04ms であるとしており[35]、機材のフレームレートは 165.5fps 以上が求められる。そのため MIDAS[34]を含む複数の手法では、東京大学が開発した 1000fps の高速プロジェクターを使用されているが、フルカラーモデルが 500 万円以上するなど高価であることから使用のハードルは高い。映像のズレを解消するため、追跡結果を用いて予測値を計算し、投影映像に適用する手法も提案されている[31]。

本研究では、投影対象の形状入力や追跡マーカーの貼付など、事前準備無しで、非剛体である葉や花の追跡と投影を実現する（表 1）。また、各 1 台の赤外線カメラと 60fps の家庭用プロジェクターで構成される MIDAS[30]と比べ比較的安価な機材を用いるが、予測値等を使用することで、機材の範囲内で可能な限り高速な DPM 生成を実現する。

表 1 非剛体を投影対象とする DPM の先行事例と本研究の比較

	Makeup Lamps[31]	Narita らの研究 [32][33]	MIDAS[34]	本研究
投影対象	 人の顔	 紙, 布	 手や液体 粘土等	 葉や花
入力	3D 形状	なし	なし	なし
マーカー	なし	マーカー	なし	なし
カメラ	1 台	1 台	3 台	1 台
プロジェクター	1 台	1 台	3 台	1 台

2.1.4 インタラクティブなプロジェクションマッピング

PM 生成手法を応用したアプリケーションの中でも、体験者が投影映像や投影対象に
関与するインタラクティブなシステムの事例を複数紹介する。インタラクティブな
PM においても、投影対象自体は変化せず、体験する人の動き等を取得して映像の内容
が変化する静的 PM と、投影対象自体が動き、その動きも映像内容の変化に用いる DPM
がある。また、インタラクティブな PM における主な課題として、体験者が投影対象
に干渉する際に起こる映像の遮蔽が挙げられる。本項では、遮蔽に対処する手法とイ
ンタラクティブな PM 事例について述べる。

まず、静的な PM における、遮蔽に対応した手法として、Hiratani らの研究[36]では、
再帰透過光学系という光学素子を用いることで、遮蔽物に映る映像と、投影対象に差
す影が生じにくい投影システムを提案している。再帰透過光学系は、光源と面対象な
位置に像を作ることができる。そのため、投影対象と、対象と同じ形状をもつダミー
を、光学素子を中心に面対象な位置に配置し、ダミーに映像を投影する。ダミーに投
影された像は光学素子を通した複数経路の光線によって、面対象の位置にある対象上
に像を映すため、光線全てを遮らない限り、本影がない PM を鑑賞できる (図 7 左)。
また、Kim らの研究[37]では、深度カメラによって、遮蔽する人の骨格を検出し、2 台
のプロジェクターで遮蔽領域を補完している (図 7 右)。



Hiratani らの研究[36]

Kim らの研究[37]

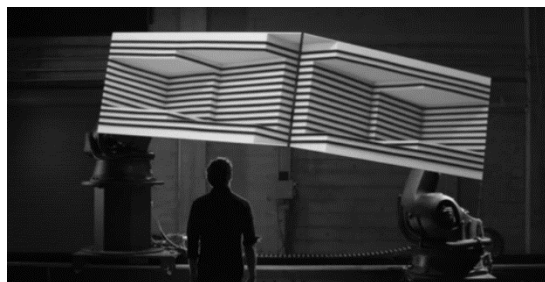
図 7 遮蔽に対応したプロジェクションマッピング.

Kagami らの手法[38]では、ユーザーが動かす平面上にインタラクティブな投影を行う DPM 手法を提案している. 投影されるコンテンツの動きが、重力やイラストとの衝突などを考慮して計算されることで、ユーザーは紙を動かして、投影映像に変化を与えることができる.

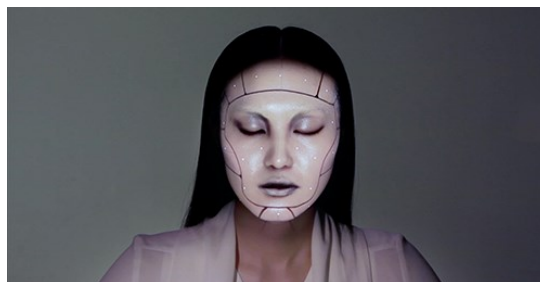
本研究においても、人の手の接触や遮蔽等のインタラクションを想定した追跡手法を提案する. さらに一部の手法では、画像処理に加え、静電容量式センサを用いて投影対象への接触を検出し、エフェクトを変化させる.

2.1.5 プロジェクションマッピングを用いた作品

この節では、PM を用いた作品を紹介する. まず、建物を投影対象とした PM 作品として、東京駅舎への投影作品を紹介する[1]. 投影映像中の線や面の色/位置が駅舎の形状に沿って変化していくことで、駅舎自体が変化、変形しているような錯覚を起こす作品になっている. つづいて、動く対象への投影事例を挙げる. Box[13]はロボットアームが動かす 2 つのキャンバスに投影を行った映像作品である (図 8 左). キャンバスと映像、撮影するカメラの位置をコンピュータで正確に動かすことで、キャンバスの中に奥行きを感じさせる映像作品となっている. 次に、顔に投影を行った事例として、OMOTE[4]がある (図 8 右). モデルの顔をあらかじめ取り込んでおくことで、人の顔の表情や質感が変化しているかのような投影を実現している.



Box[13]



OMOTE[4]

図 8 DPM 作品.

生き物に投影を行った作品としては、花と魚[39]がある。水族館の水槽に映像を投影し、通過する魚に映像が映る。また、映像は魚の動きに影響を受けながら変化していく。同じく魚に投影を行った事例として、ChatFish[40]がある。魚のシルエットをマスクとして、投影映像を切り抜くことで、動く魚の領域部分のみに投影している。

本研究の制作では、bioluminescent forest [3]から着想を得、生物発光を題材とする。ホタルやクラゲなどの生物発光のように、映像投影によって植物自身が発光しているような作品を目指す。ホタルの鑑賞や光る生物の展示などが各地で見られるように、生物発光は人を惹きつける魅力があると考え、生物発光等をモチーフとしたエフェクトアニメーション（エフェクト）を作成し、投影する。本制作では自然界における光、植物の葉脈や細胞などもモチーフとする。また、取得した投影領域の形状を用いてエフェクトを自動生成することで、より植物自身が発光しているような表現を目指す。複数種類のエフェクトを作成し、本研究による成果として植物に適用する。

2.2 本研究と関連する画像処理手法

本研究では、動的な植物に DPM を実現するため、カメラで投影対象を取得し、画像処理を用いて投影領域の選出と追跡を行う。そこで本節では、本研究に関連する画像を用いたトラッキング手法を紹介する。トラッキングは、動画において前フレームと現フレームの画像上の位置を毎フレーム対応付けることで達成される。

まず、2 枚の画像間で画像上の位置を対応付ける手法を紹介する。特徴点のマッチングは、特徴を持つ点における特徴量を記述し、特徴量を 2 つの画像間で比較することで、対応付ける。特徴量の抽出については、様々な方法が提案されている[41][42][43]。次に、テンプレートマッチングは、画像中から注目部分を切り出したものをテンプレートとし、もう一枚の画像中で走査して、最も類似度が高い箇所を探索する手法である。

つづいて、Lucas-Kanade 法[44]によるオプティカルフローを紹介する。Lucas-Kanade 法では、動画におけるフレーム間において、動きが微小であること、明るさが変化しないこと、近傍は同様の動きをすることを仮定し、移動ベクトルとしてオプティカルフローを計算する。

本研究の手法では、紹介した手法を適宜用いながら、投影対象の追跡を達成する。

2.3 植物を用いたメディアアート

植物を用いたメディアアートの事例を複数紹介する。Interactive Plant Growing[45]は、鑑賞者が植物に触れることで、仮想空間の植物が育っていくインスタレーションである（図 9 左）。また、Botanicus Interacticus[46]は、植物に触れた位置等を推測できるセンシング技術によって、ディスプレイにエフェクトを表示するインスタレーションである（図 9 右）。



The interactive plant growing[45]



Botanicus Interacticus[46]

図 9 植物を用いたメディアアート。

一方で、植物自体を動かす事例として、Botanical Puppet[47]では、電気刺激により、オジギソウの枝 1 本ごとの動作を制御している。植物そのものを変化させるため、植物や鉢に直接電極を接続する必要がある。また、plant[48]では、アクチュエーターで動く造葉の群で構成されたディスプレイを提案しており、人の動きに対してインタラクティブに変化する。

また、植物をモチーフとしたインテリアプロダクトとして、floura [49]のような植物形状の照明も開発されている。

本研究では、対象の形状情報等から生成したエフェクトを現実の植物に投影し、現実の植物自体の見え方を変えることで、より対話的なコンテンツの作成を目指す。また、本研究で提案する 4 つのシステムは、基本的には植物と非接触でインタラクションの入出力を行うため、様々な植物に適応可能である。

2.4 本研究の目的

本研究の目的は、人の手や風などの外力によって動き変形する植物に対して、追跡を行い、DPM を実現することである。2.1.3 項で紹介しているように、DPM 生成手法は複数提案されているが、本研究では、事前の形状入力やマーカー無し、かつ簡易な機材によって植物への DPM の実現を目指す。また、複雑な形状をもつ対象への投影やインタラクションを伴う投影、多数の対象への投影など、様々なシーンを想定し、DPM 生成手法を複数提案し、比較、考察する。

さらに、作品としては、植物自身が発光しているような表現を目指し、投影対象である植物に由来したエフェクトを複数作成し、本手法によって適用する。植物の動きや変形、植物への接触や遮蔽を考慮した追跡手法を応用することで、より対話的な作品を制作する。

第3章

手法概要

本研究では、植物の中でも、特に薄型の葉や花に限定して投影を行う。また、様々なシーンを想定し、葉や花への DPM 生成手法を 4 つ提案する。本章では、第 4 章、第 5 章、第 6 章、第 7 章で後述する 4 つの手法において、共通するシステム構成と手法の流れについて述べる。

3.1 システム構成

投影面を赤外線カメラで取得し、投影面に適応させた映像をプロジェクターで投影する ProCam システム (2.1.1 節) を用いる。カメラとプロジェクターは、双方の光軸のズレを最小限に抑えるために可能な限り近くに配置し、葉の全面が見える位置に設置する。また、投影対象を照らすように赤外線照明を設置する (図 10)。

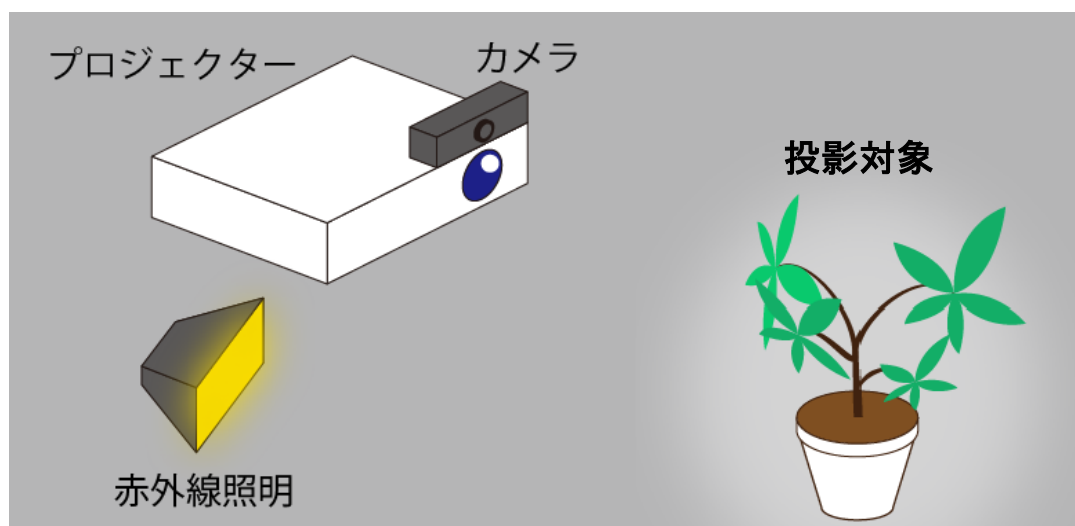


図 10 設置図。

体験者がプロジェクターで投影された映像を鑑賞しやすいようにするため、本システムの実行環境は暗い環境が適している。しかし、暗い環境で投影対象撮影する場合、可視光では 2 つ問題がある。一つは暗くて対象をうまく撮影できないことである。もう一つは、本システムによって対象上に投影された映像がカメラに写りこんでしまい、追跡の画像処理へ影響を与えてしまうことである。そこで、本システムでは、不可視光線である赤外線光を投影対象に照射し、赤外線カメラで撮影する。これによって、プロジェクターによる投影内容は含まず、暗い環境でも投影対象の植物をとらえることができる。

3.2 手法の流れ

提案する4つの手法では、共通して以下の手順で処理を行い、植物へのDPMを実現する。

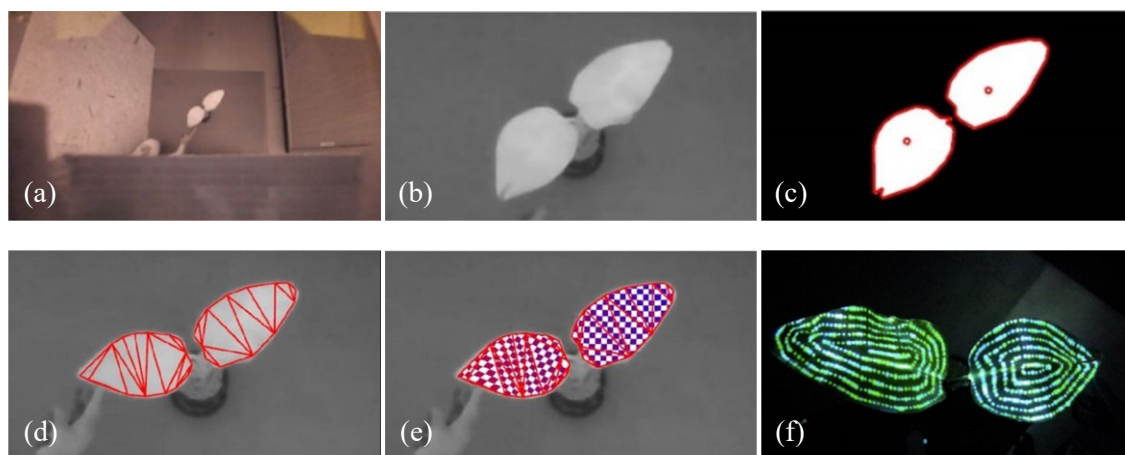


図 11 本手法の概要：(a)赤外線カメラ画像，(b)射影変換後，(c)投影領域，(d)追跡結果，(e)エフェクトのテクスチャマッピング，(f)投影結果。

-
- ① 赤外線カメラで画像を取得する（図 11a）。
 - ② 入力画像に対して射影変換を適用し（図 11b），プロジェクターの投影領域とカメラの取得領域における位置座標を揃える。射影変換には，前準備で記録した情報を用いることで効率よく行うことができる（4.1.1節）。
- <1フレーム目>
- ③ 画像から投影領域を選出する（図 11c）。画像処理によって輪郭を抽出し，ユーザーのマウス入力を用いることで，投影領域を決定する。
- <2フレーム目以降>
- ④ 投影領域を追跡する（図 11d）。
 - ⑤ エフェクトを生成し，追跡された投影領域にテクスチャマッピングする（図 11e）。
 - ⑥ 生成された結果をプロジェクターで投影する（図 11f）。
-

第4章

手法 1 : 楕円形の葉への DPM

本章では、本研究の初期段階として、主に楕円形の葉を投影対象とした DPM を実現する手法を提案する。DPM 実現のため、image registration（プロジェクターとカメラの位置合わせ）手法、投影領域の追跡手法、3 種類のエフェクト生成方法を提案する。image registration においては、事前準備を行うことで、投影時に要する労力・時間を軽減する。追跡においては、投影対象の輪郭上の最も遠い 2 つの頂点を追跡することで、全ての頂点の位置を計算する。エフェクトは生物発光のように、投影対象自身が発光しているような表現を目指し、作成する。本手法は、芸術科学会論文誌第 20 巻[50]に掲載され、SIGGRAPH ASIA 2018 Poster[51]にて発表した。また、NICOGRAPH 2018 展示発表[52]にて優秀展示作品賞、2018 アジアデジタルアート大賞展 FUKUOKA[53]にて学生カテゴリー/インタラクティブアート部門 優秀賞、福岡市長賞を受賞した。

4.1 手法詳細

4.1.1 プロジェクターとカメラの image registration

現実空間でプロジェクターが投影可能な最大領域（図 12B）をカメラで取得し（図 12A），取得画像（図 12A'）中の投影領域（図 12B'）をプロジェクターの出力解像度に合わせて射影変換することで（図 12B''），image registration を行う。

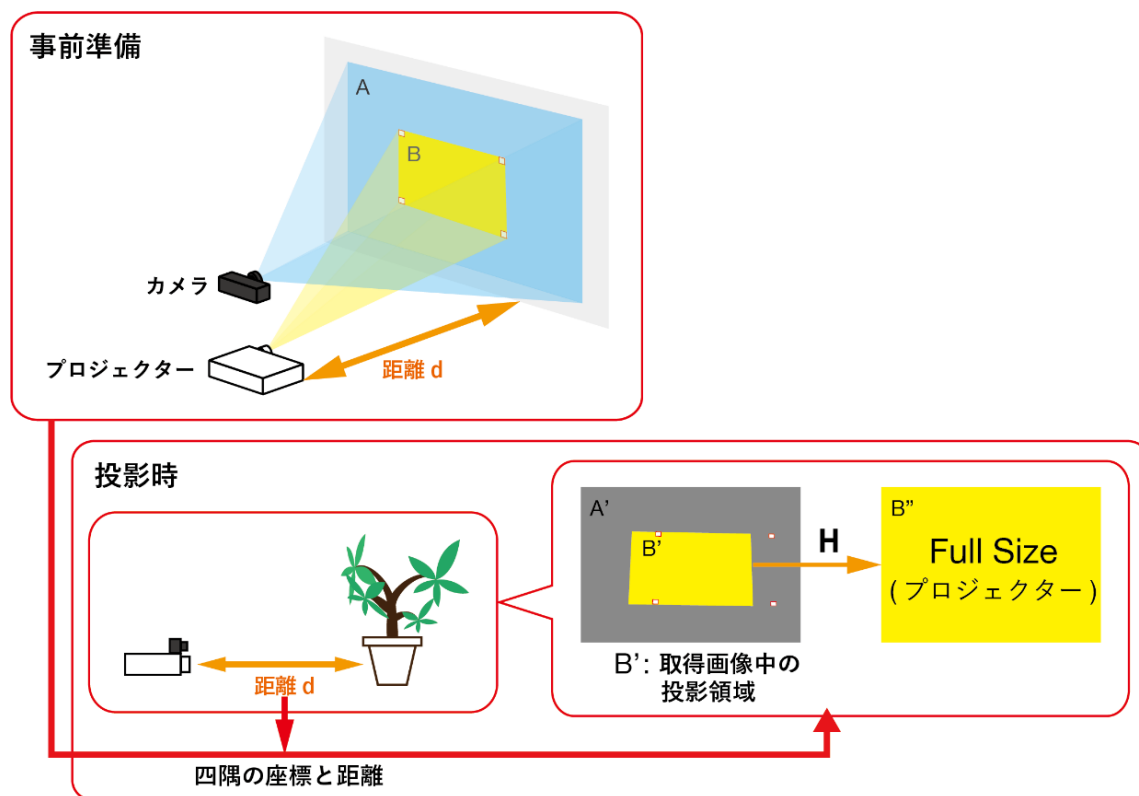


図 12 射影変換による registration.

以下のような手順で **registration** を比較的簡単に行うことができる。

＜事前準備内容＞

1. 黒背景の画像に4隅が白い正方形を描画した画像を投影
2. プロジェクター投影領域の4隅の座標を記録

＜エフェクト投影の際＞

3. 記録した座標をもとに射影変換（後述）

上述の 1-2 は事前に一度だけ行う。事前準備時は、平らなスクリーンに対して垂直に配置したプロジェクターによって投影を行う。プロジェクターとスクリーンの距離 d を 10cm ずつ手作業で変えながら 0.5m から 2.5m まで、プロジェクター投影領域を自動で記録する（付録）。このとき投影内容として 4 隅に白の四角形を配置した画像を投影し、四角形の位置を画像処理によって取得した。このように対象物との距離に対応する投影領域を一度記録することで、その後は投影時に投影対象とプロジェクターの距離に応じて記録した 4 つの座標を射影変換に用いる。プロジェクターの解像度のサイズを記録した 4 隅に一致するよう射影変換を行うことで、**registration** を行う。

実際に投影対象となる植物は記録時のような平面ではないが、このような処理によって画像の投影サイズを合わせる必要がある。本手法で扱う葉は縦横最大 20cm 以内の薄型のものであり、葉の付け根はある程度固定されているため、インタラクション等による奥行方向の移動を考慮しても最大で約 30cm 以内と想定した。投影対象の大きさと移動距離の想定した条件から、距離 d の範囲と量子化幅を決定した。

4.1.2 投影領域の追跡

最初に、簡単な手入力を含む処理によって、投影領域を決定する。まず、赤外線カメラで取得した初期フレーム画像を二値化する。二値化の閾値は、ユーザーがスライドバーを操作して調整することによって決定する。二値化画像の白領域の輪郭を抽出し、ユーザーが輪郭内部をマウスクリックによって指定することで、追跡する葉の領域を選択する。ここで、1 つの葉の領域における輪郭の各頂点の位置を $p_i \in \mathbb{R}^2, i=1,2,\dots,n$, n : 頂点数, とする。輪郭の頂点 p_i に基づいて内部をドロネー分割することで、三角形メッシュに分割する（図 13 赤線）。次のフレーム以降、時刻 $t-1$ と時刻 t において対応する輪郭の各頂点 p_i を求め、追跡する。頂点 p_i^t で構成される三角形メッシュに対し、テクスチャマッピングによってエフェクトを適用する。

輪郭の各頂点の対応付けでは、投影対象とする葉の形状を楕円形で長軸と短軸があるとし、時刻 $t = 0$ における輪郭上の最も遠い2つの端点 p_{far1}^0, p_{far2}^0 を追跡する。このとき、 $far1, far2$ は輪郭の頂点 p_i^0 中の端点の番号である（図 13）。

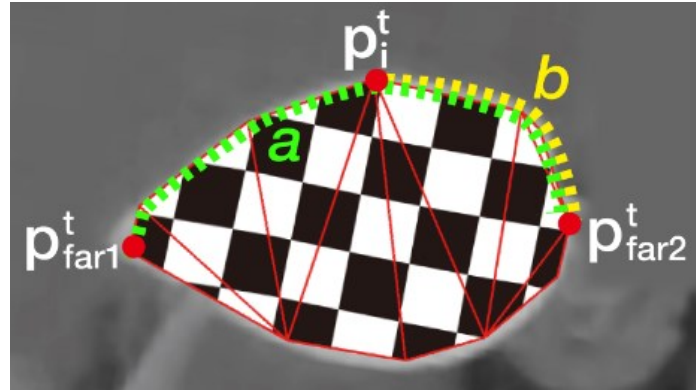


図 13 輪郭上の端点 p_{far1}^t, p_{far2}^t と残りの点 p_i^t 。

次のフレーム以降、時刻 t における端点 p_{far1}^t, p_{far2}^t を追跡する。まず、時刻 $t-1$ から t 間における $p_{far1}^{t-1}, p_{far2}^{t-1}$ の移動ベクトルとして、オプティカルフロー $\mathbf{o}_{far1}^t, \mathbf{o}_{far2}^t$ をLucas-Kanade法によって求める。すると、2つの端点の移動後の位置は

$$p_{opt1}^t = p_{far1}^{t-1} + \mathbf{o}_{far1}^t,$$

$$p_{opt2}^t = p_{far2}^{t-1} + \mathbf{o}_{far2}^t$$

と計算される。しかし、これらの点は現時刻における葉の輪郭線上にあるとは限らない。よって、時刻 t に抽出した葉の輪郭の頂点 $q_j^t \in \mathbb{R}^2, j=1,2,\dots,m, m$: 頂点数, より、以下の処理によって最適な点を選択する。まず、葉の輪郭の頂点 q_j^t の中で最も遠い2点 q_{farA}^t, q_{farB}^t ($farA, farB$: q_j^t 中の端点の番号)を求める。次に、オプティカルフローによる移動後の2点 p_{opt1}^t, p_{opt2}^t と対応する頂点 q_{far1}^t, q_{far2}^t を q_{farA}^t, q_{farB}^t から以下のよう求める。

$$d_{sum1} = |p_{opt1}^t - q_{farA}^t| + |p_{opt2}^t - q_{farB}^t|$$

$$d_{sum2} = |p_{opt1}^t - q_{farB}^t| + |p_{opt2}^t - q_{farA}^t|$$

$$q_{far1}^t, q_{far2}^t = \begin{cases} q_{farA}^t, q_{farB}^t & (d_{sum1} < d_{sum2}) \\ q_{farB}^t, q_{farA}^t & (d_{sum2} < d_{sum1}) \end{cases}$$

次に, q_{far1}^t, q_{far2}^t と p_{opt1}^t, p_{opt2}^t の距離 d_1, d_2 によって, 端点 p_{far1}^t, p_{far2}^t を以下のよう求める.

$$d_1 = |p_{opt1}^t - q_{far1}^t|$$

$$d_2 = |p_{opt2}^t - q_{far2}^t|$$

$$p_{far1}^t = \begin{cases} q_{far1}^t & (d_1 \leq \alpha) \\ \arg \min_{r \in \{q_j^t\}} |p_{opt1}^t - r| & (d_1 > \alpha) \end{cases}$$

$$p_{far2}^t = \begin{cases} q_{far2}^t & (d_2 \leq \alpha) \\ \arg \min_{r \in \{q_j^t\}} |p_{opt2}^t - r| & (d_2 > \alpha) \end{cases}$$

閾値 α は実験によって経験的によい追跡結果を得られた $\alpha=5$ とする. 端点以外の点 p_i^t は初期状態での端点からの長さの比を用いて計算する. 輪郭線上の p_{far1}^t から p_{far2}^t 間の長さを a , p_{far1}^t から p_i^t の長さを b とすると, 比 b/a が初期状態の比と等しくなる点を追跡結果とする (図 13).

しかし, 上記の結果には, 画像のノイズによるズレが含まれる. そのため, 投影するエフェクトの動きは時間経過の前後で滑らかではなくなる. よって, ズレによる出力映像への影響を軽減する処理を行う. 時刻 t の各点の移動ベクトル $\mathbf{v}_i^t \in \mathbb{R}^2$ の時刻 $t-2$ 以降の加重平均結果 $\mathbf{v}_i^{avg}$ は以下である.

$$\mathbf{v}_i^{avg} = \frac{w^{t-2}\mathbf{v}_i^{t-2} + w^{t-1}\mathbf{v}_i^{t-1} + w^t\mathbf{v}_i^t}{w^{t-2} + w^{t-1} + w^t = 1}$$

このとき, 重み $w^{t-2}=0.2, w^{t-1}=0.3, w^t=0.5$ とした. また, 時刻 $t-2$ 以降の移動ベクトル $\mathbf{v}_i^{t-2}, \mathbf{v}_i^{t-1}, \mathbf{v}_i^t$ のうち, 移動量が中央値を持つベクトル $\mathbf{v}_i^{med}$ とする. 各移動ベクトルの移動量が全て閾値 β 以下の時は $\mathbf{v}_i^{avg}$ を, その他の場合は $\mathbf{v}_i^{med}$ をズレが軽減された結果 $\mathbf{p}_i^t$ として以下のように求める.

$$\mathbf{p}_i^t = \begin{cases} \mathbf{p}_i^{t-1} + \mathbf{v}_i^{avg} & (|\mathbf{v}_i^{t-2}| < \beta \text{ and } |\mathbf{v}_i^{t-1}| < \beta \text{ and } |\mathbf{v}_i^t| < \beta) \\ \mathbf{p}_i^{t-1} + \mathbf{v}_i^{med} & (|\mathbf{v}_i^{t-2}| > \beta \text{ or } |\mathbf{v}_i^{t-1}| > \beta \text{ or } |\mathbf{v}_i^t| > \beta) \end{cases}$$

閾値 β は経験的に $\beta = 10$ とした. また, ズレを軽減した結果 $\mathbf{p}_i^t$ は出力結果のみに用い, 次のフレームの処理では, $\mathbf{p}_i^t$ を用いる.

4.1.3 3 種類のエフェクトアニメーションの作成

本手法では、3 種類のエフェクトを生成する。生物発光のように、投影対象自身が発光しているような表現を目指し、葉自身から得られる輪郭や形状、葉脈生成アルゴリズムを用いてエフェクトを生成する。エフェクトは初期フレームに得た取得画像の輪郭情報等を元に作成し、作成した映像を追跡時に読み込む。

1) ホタルをモチーフとしたエフェクト ホタルのように葉内部が発光しているような表現を目指し、葉が一定のリズムで投影領域全体が点滅するエフェクトを作成する（図 14、図 15）。投影領域に対して、モルフォロジー演算（収縮処理）を用いる。以下のような手順を毎フレーム繰り返し、アニメーションを作成する。アニメーションの現フレームを n フレーム目とする。

1. 収縮処理とガウシアンマスクの大きさを計算する。
2. 1 で計算した大きさのマスクで、収縮処理を行う。
3. 1 で計算した大きさのマスクでぼかし処理を行う。

手順 1 では、 n を用いて、マスクのサイズを計算する。点滅に強弱をつけるため、 n に正弦関数をかけて、計算する。手順 2 では、得られたサイズのマスクで、収縮処理を行う。手順 3 では、手順 1 で得られたサイズで、手順 2 で得られた結果に対し、ぼかし処理を行う。ぼかし処理には、ガウシアンフィルタを用いる。

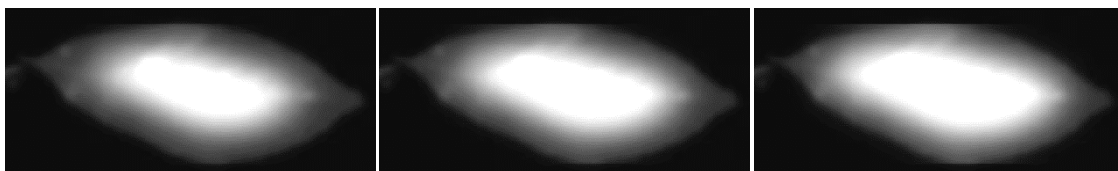


図 14 ホタルをモチーフとしたエフェクトの出力結果。



図 15 ホタルをモチーフとしたエフェクトの投影結果。

2) 葉脈をモチーフとしたエフェクト 葉脈によって水や養分が流れていくイメージのエフェクトを生成する(図 17, 図 18). 選出された投影領域内で葉脈を生成し, 葉脈を溝に見立て, 粒子が葉脈の始点から移動する. 生物発光では, 青や緑の光が多く見られることから, 粒子の色を青や緑を含む色に設定する. 投影領域の選出後の処理の流れは以下である.

1. ユーザーがマウスクリックにより, 葉脈の始点を指定する.
2. 葉脈を生成する.
3. 葉脈に沿って粒子を動かす.

Runions らの手法[54]では, ランダムに配置された栄養分に向かって, 葉脈が形成されるというアルゴリズムに則って葉脈の生成をシミュレーションしている. 本手法ではこの栄養分を抽出された輪郭内部に配置する. また, 葉脈の始点をマウスで指定することで, より適切な葉脈の形を生成する.

葉脈を元に投影領域に速度場(図 16)を生成し, これに基づき粒子が加速度運動によって移動する. 葉脈を生成する際, 葉脈上の点それぞれに前の点から自身の点に向かう速度ベクトルを計算し, 正規化したベクトル $\vec{v}$ を保存する. 全ての投影領域内の各画素において, 注目画素に最も近い葉脈のベクトル $\vec{v}$ を探索する. $\vec{v}$ に垂直なベクトルを $\vec{u}$ とすると, 各画素に対して働く力 $\vec{w}$ は $\vec{w} = \vec{v} + \vec{u}$ となる.

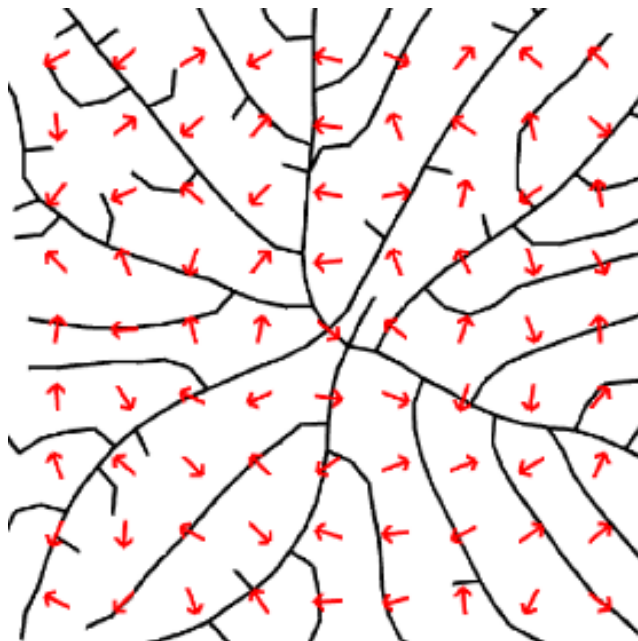


図 16 生成した葉脈とベクターフィールド

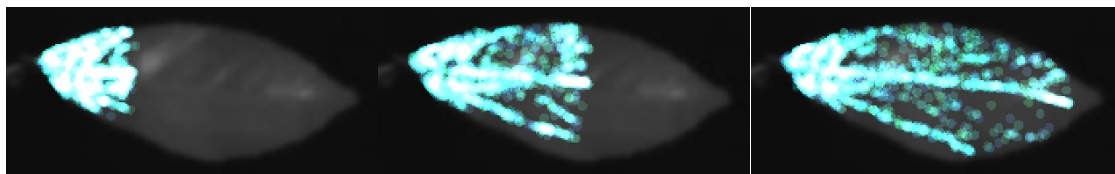


図 17 葉脈をモチーフとしたエフェクトの出力結果.

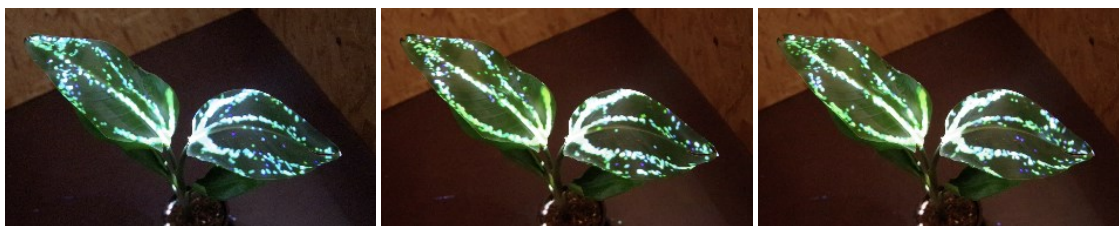


図 18 葉脈をモチーフとしたエフェクトの投影結果.

3) オワンクラゲをモチーフとしたエフェクト 形状に沿って身体が発光するオワンクラゲをモチーフに、投影対象の縁に配置された粒子が発光するエフェクトを生成する（図 19、図 20）。処理の手順は以下のようになる。

＜初期フレーム＞

1. 投影領域に対して、モルフォロジー演算（収縮処理）を行う。
2. 輪郭を抽出する。このとき、輪郭の頂点間は距離が等しくなるように計算する。
3. 得られた頂点上に、粒子を配置する。

＜毎フレーム＞

4. 粒子の色や大きさを変化させ、点滅させる。

粒子の色は、葉脈エフェクトと同じく、青や緑を選択する。

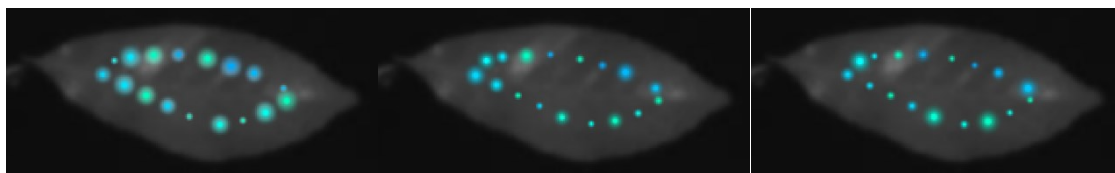


図 19 オワンクラゲをモチーフとしたエフェクトの出力結果.

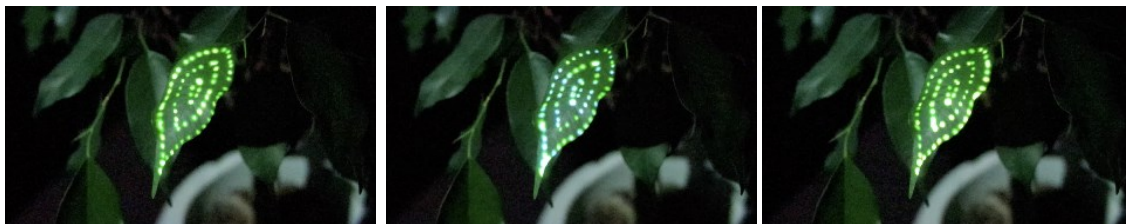


図 20 オワンクラゲをモチーフとしたエフェクトの投影結果.

4.2 結果考察

本研究によって作成した追跡結果と投影結果を図 21, 図 22, 図 23, 図 24 に示す. 本手法で示す追跡結果と投影結果の作成には, Windows PC(Intel Core i9-9900(3.1GHz) PC)を使用した. また, カメラは解像度 $2048 \times 1088 \text{px}$ の STC-MBCM200U3V-NIR, プロジェクターは解像度 $1280 \times 720 \text{px}$ の Taxan KG-PL081W, 赤外線照明は S8100-IR を用いた. エフェクトの投影結果や作品制作のみ, コンピュータは 16GB のメモリを搭載した Macbook Pro (Intel Core i7 (3.2 GHz) PC with 3 GB RAM and a C++ compiler), カメラは解像度 1920×1080 の BUFFALO BSW200MBK を使用した. 場所は暗い室内で実験を行った.

まず, 本システムによるプロジェクターの registration において, 手作業では約 40 分程度かかるところを, 約 10 分で行うことができた. また, 作品展示なども含めて, 投影結果に大きな問題はなかった.

つづいて, 本システムによる追跡結果を確認するため, 手動による 4 種類の葉の動かし方に対し, 追跡と投影を行った結果を図 22 に示す. 動かし方は, 平行移動 (図 22ab), カメラ方向 (z 軸方向) を中心とした軸回転 (図 22cd), 葉の長軸方向 (x 軸方向) を中心とした軸回転 (図 22ef), 葉の折り曲げ (図 22gh) とした. これらの結果より, 本手法では葉の追跡, および現実空間での投影の双方において問題なく自動で行えることがわかる. 本手法では最も遠い 2 つの端点を追跡しているため, 端点を同定できないような極度の変形や端点の遮蔽が無い限り, 移動や z 軸回転, x 軸回転, 変形に対して, 柔軟に追跡が可能である (図 21, 図 22).

課題としては, 端点の同定が難しい, 正円に近い形状やカエデ等の複雑な形状, 大きな形状変形への対応が困難という点である. 本手法では, 葉が長軸と短軸をもつ楕円形という前提で, 2 つの端点を追跡することで, 輪郭の各頂点の位置を計算しているため, 端点の追跡が難しいと考えられる正円や複雑な形状の葉には投影できなかった.

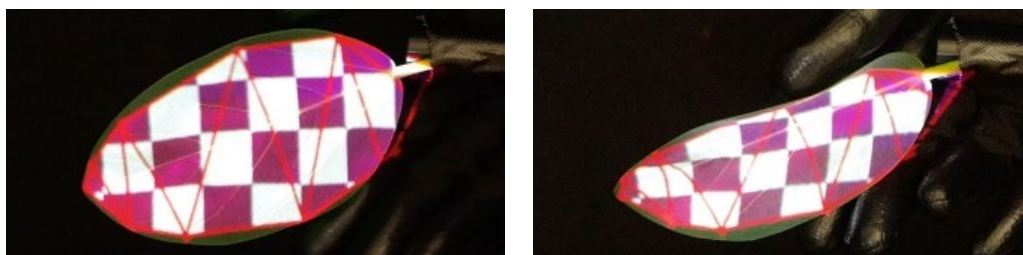


図 21 中心部を凹ませるように変形させた葉への投影結果.

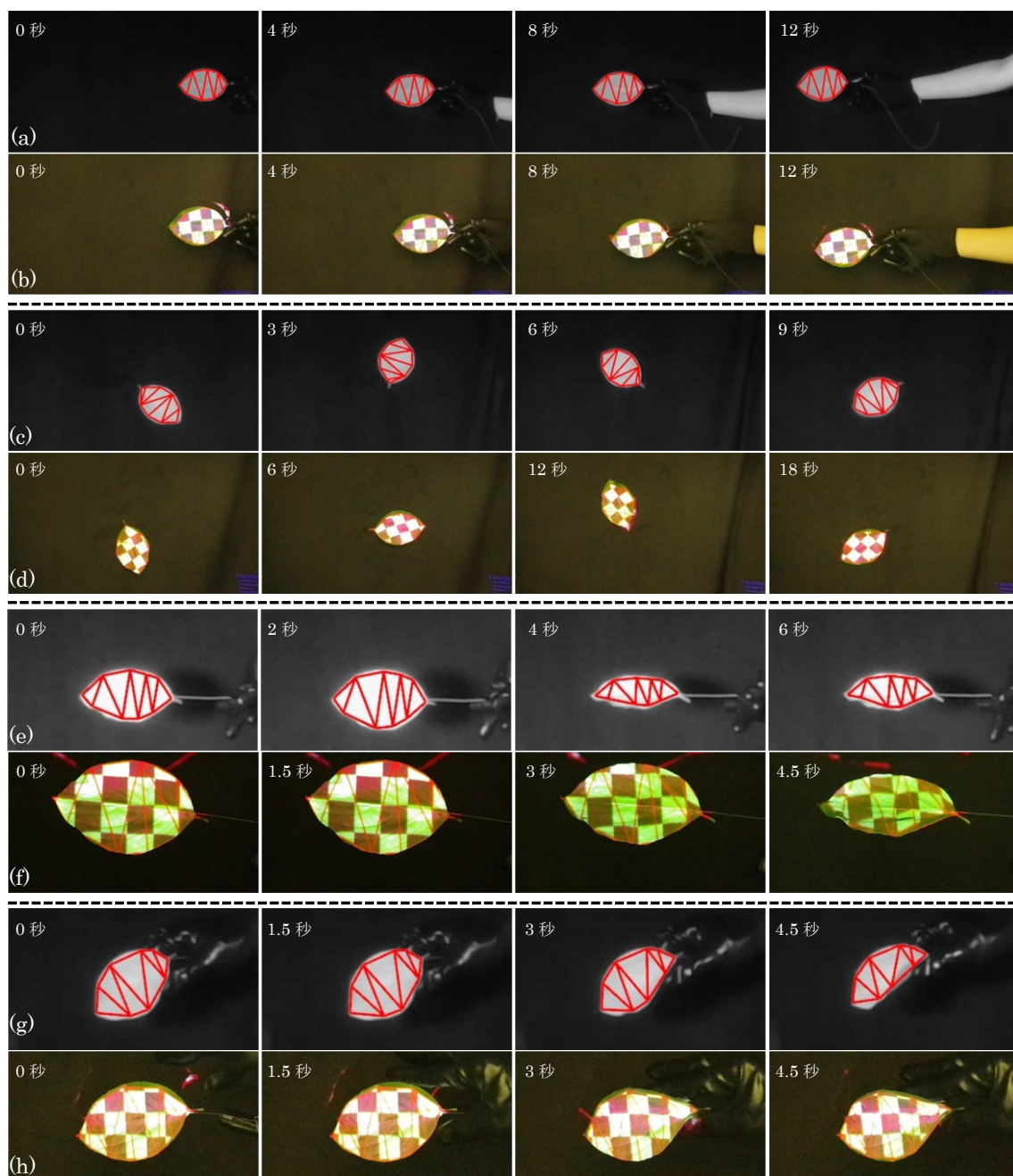


図 22 動く葉に対する追跡結果と投影結果（長軸 11cm，短軸 7cm の葉を手動で動かした。） (a), (b) 約 40cm 平行移動させた葉に対する追跡・投影結果，(c), (d) カメラ方向の軸を中心に約 360 度回転させた葉に対する追跡・投影結果，(e), (f) 葉の長軸を中心に約 45 度軸回転させた葉に対する追跡・投影結果，(g), (h) 一部を折り曲げるように変形させた葉に対する追跡・投影結果。

また、エフェクトの投影結果を図 23に示す。本手法は楕円形の葉の追跡を前提としたものであるが、実際に様々な形状の葉に適用したところ、楕円形の葉だけでなく長軸と短軸の差が大きい形状であれば追跡することができた(図 23d)。本手法による追跡とエフェクトの生成は、投影対象が1つの時、追跡に約2.4msの時間を要し、約80fpsの速度で処理を行うが、システム全体としては使用したカメラの最高速度である30fpsで動作する。通常、遅延を感じない機材環境は165.6fps以上であるため[35]、30fpsのカメラの場合、遅延を避けることはできない。本手法はそのような制約の下、安定した追跡を行うことができた。実際に、平行移動する葉に対しては60cm/s、回転する葉に対しては2.4rad/sまで追跡できたが、投影時には4cm/s以上で平行移動する葉、0.3rad/s以上で回転する葉において、システム遅延によって生じるズレを確認した(図 24)。また、投影対象の葉の数を変えた時、処理時間は増加するものの、14枚までは30fps(1フレーム33.3ms)で追跡できることを確認した(図 23 abcef, 図 25)。

生成した3種類のエフェクトは本システムによって、投影対象の形状や変形の条件を満たす限り、動く葉に対して大きくずれることなく、いくつかのエフェクトを適用できた。これによって、「葉自体が発光しているような表現」を行った(図 23)。

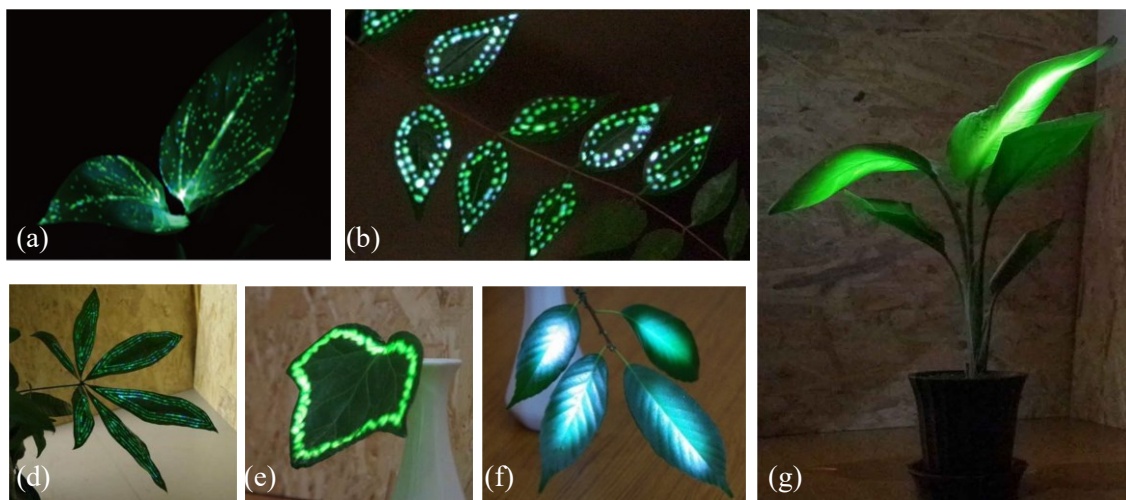


図 23 エフェクトの投影結果 (a)葉脈をモチーフとしたエフェクト, (b)(c)(d)オワンクラゲをモチーフとしたエフェクト, (e)(f)ホタルをモチーフとしたエフェクト。

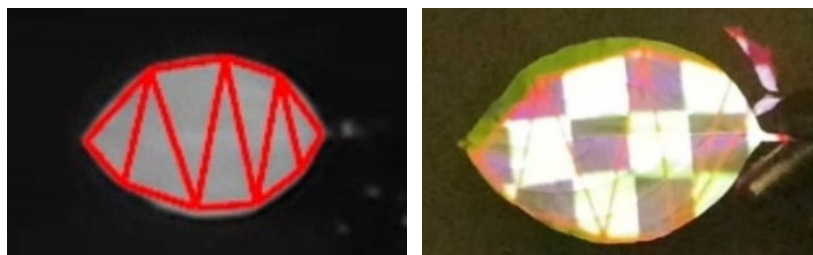


図 24 同速度で動く葉に対する追跡結果（左）とズレが生じる投影結果（右）

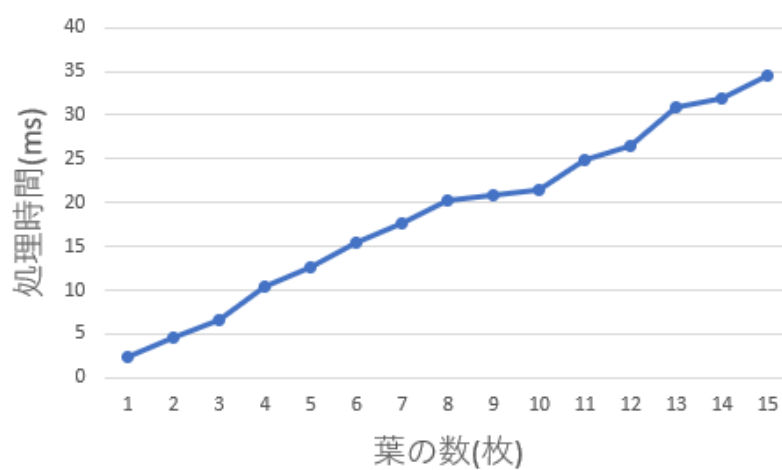


図 25 葉の枚数に伴う処理速度の変化

4.3 作品展示

本システムを用いた作品制作の実践として、NICOGRAPH2018 において「Projection for Leaves」[52]，アジアデジタルアート大賞展（ADAA）2018 において「Projection Mapping for Leaves」を発表し，展示を行った[53]．展示では体験者は赤外線カメラに写らない手袋を着用した上で葉を動かせるようにした．

「Projection for Leaves」の展示は，2 日間にわたり各 5 時間行い，多くの研究者や学生に鑑賞してもらった．1 鉢の植物に投影を行い，インタラクションによる葉の移動量によって，3 種類のエフェクトが切り替わるようにした．「Projection Mapping for Leaves」の展示は，6 日間にわたり各 10 時間，美術館で行い，小学生程度の子供からご高齢の方まで幅広い世代の方々やアーティストの方々に鑑賞してもらった．鑑賞時間は，作品説明の時間を含めて 3～5 分程度であった．投影対象毎に異なるエフェクトの適用と時間変化に伴う切り替えによって，3 種類のエフェクトを提示した（図 26）．

両展示合わせて，100 人以上の来場者に鑑賞してもらうことができた．多くの来場者に興味を持ってもらい，体験者からは様々な意見が得られた．まず「綺麗」「面白い」等の意見が得られたため，来場者に作品を楽しんでもらうという，本作品の目標を達成することができた．エフェクトは生物発光をモチーフとし，「生命感を感じる」等の意見が得られたため，目指していた表現をある程度達成することができた．また，両作品では本手法の自動追跡システムを活かし，葉を触って動かす様子を鑑賞してもらった．その結果，作品を動かした際に鑑賞者が驚く様子が多く見られたため，可変な物体への DPM によって，従来の静止した投影対象への PM とは異なり，より対話的な体験を提供することができた．その他，「落ち着く」「部屋に欲しい」等の意見も得られたため，インテリアとしての展開も期待できる．一方，機材環境の制約の下，作品を展示した結果，作品の鑑賞において遅延は特に問題にならなかった．また，両作品において受賞するなど高い評価を得た．



図 26 アジアデジタルアート大賞展 2018 の展示様子．

第5章

手法 2 : 接触を考慮した様々な形状の植物への DPM

本章では、投影する対象をカエデやイチョウ、花等の様々な植物を対象に拡大し、人の手などの接触にも頑健な DPM を実現する手法を提案する。手法 1 では、輪郭上の最も遠い 2 点を追跡するため、2 点の同定ができない形状や変形の追跡は困難である。また、画像上における輪郭面積を大きく変えるような、接触を伴うインタラクションが生じた場合も追跡できない。そこで本手法では、各投影領域について、2 つのスケールでテンプレートマッチングを行い、輪郭の頂点同士の位置関係を保つ追跡結果を求めることで、局所的な変形を考慮しながら、領域毎の全体形状を維持する追跡を行う。本手法は、**UIST 2019 Demo[55]** にて発表した。

5.1 手法詳細

プロジェクターとカメラの位置合わせと投影領域の選出は、手法 1 と同様に行う。また、投影映像についても、手法 1 で作成したエフェクトを適用する。そのため本節では、投影領域の追跡手法について述べる。

5.1.1 投影領域の追跡

手法 1 と同様、初期フレーム ($t=0$) で選出された投影領域における輪郭の各頂点 p_i^0 について、時刻 t において時刻 $t-1$ と対応する位置 p_i^t を計算し、追跡する。まず、エッジ画像上で、投影領域全体と各頂点周辺の 2 つのスケールでテンプレートマッチングを行うことで局所的な変形に対応する。つづいて、輪郭上における両隣の頂点 p_{i-1} , p_{i+1} の重心から頂点 p_i へ向かうベクトルとしてラプラシアン[56]を計算し、初期フレームと前フレームのラプラシアンを維持する点を追跡結果とすることで、全体形状を維持する (図 27)。毎フレーム、以下の流れで処理を行う。

- A. Canny 法により取得したエッジ画像 E^t 上で、葉領域全体のテンプレート T^L を用いて、葉全体のテンプレートマッチング TM^L を行う。
- B. テンプレート T^L から小領域を切り出し、テンプレート T^S を作成し、 T^S のテンプレートマッチング TM^S を行う。
- C. TM^S によるマッチング率結果から候補点を求め、候補点から最も形状を維持する点を求め、追跡結果 p_i^t とする。
- D. コーナーを補正する。
- E. p_i^t に Kalman フィルタを適用したものを出力結果とする。
- F. ユーザーが触れる等のインタラクションによる遮蔽に対応するため、エフェクトにマスクをかける。

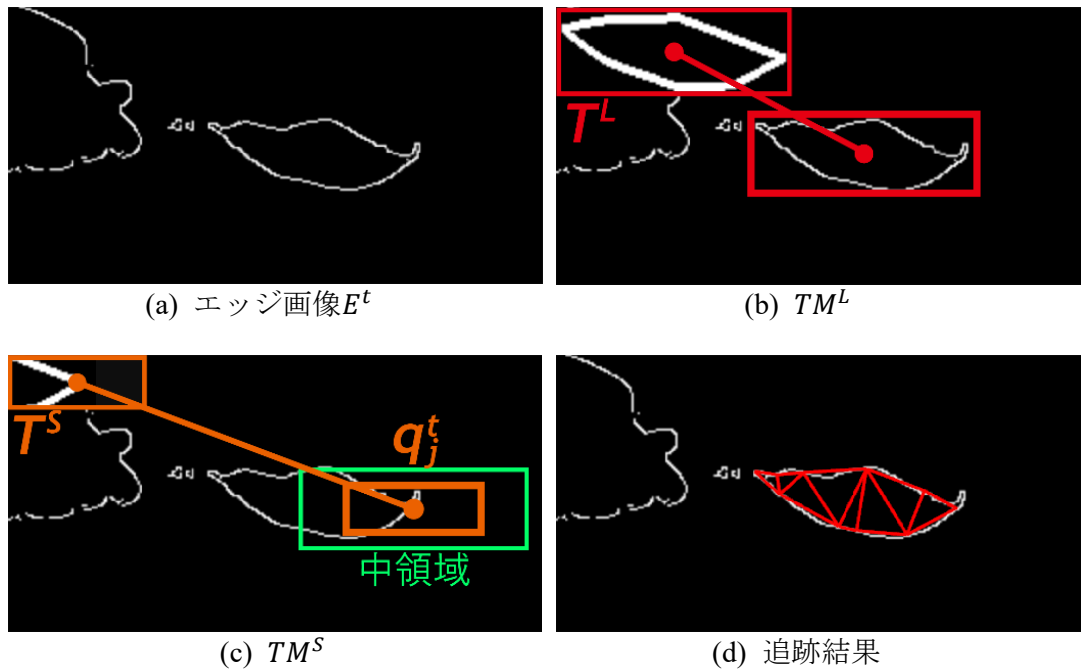


図 27 テンプレートマッチングによる追跡手法の概要.

各手順の詳細を以下で述べる.

A) 葉全体のテンプレートマッチング 葉全体の領域のテンプレートマッチング TM^L を実行する. まず, Canny 法により, 入力画像 I^t からエッジ画像 E^t を取得する (図 27). 次に, 前フレームの追跡結果 p_i^{t-1} を用いて, 線を描画した画像をテンプレート T^L とする. T^L を用いて, エッジ画像 E^t 上で葉全体のテンプレートマッチング TM^L を行い, 葉全体領域の追跡結果 $P^t \in \mathbb{R}^2$ を求める. マッチング結果 P^t に基づいて, 以下のように葉全体の移動量 $V^t \in \mathbb{R}^2$ を計算し, p_i^{t-1} に加算することで, $q_i^t \in \mathbb{R}^2$ を計算する (図 27).

$$V^t = P^t - P^{t-1}$$

$$q_i^t = p_i^{t-1} + V^t$$

B) 輪郭の各頂点のテンプレートマッチング 各頂点のテンプレートマッチング TM^S を実行する (図 27c). テンプレート T^L から, p_i^{t-1} 周辺の小領域を切り出し, テンプレート T^S を作成する (図 28). q_i^t 周辺の中領域で, T^S のテンプレートマッチング TM^S を行い, マッチング率結果画像を取得する.

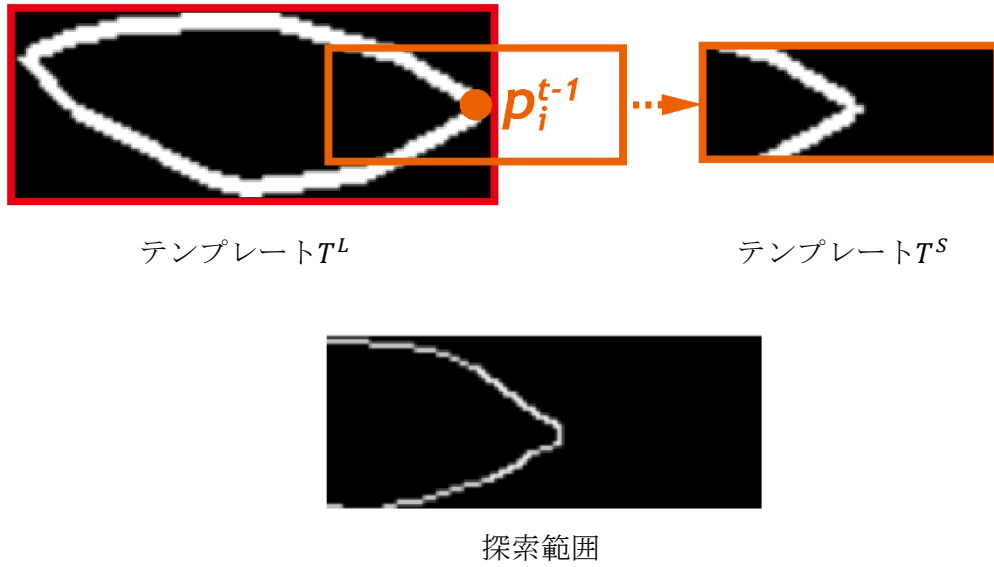


図 28 テンプレート T^S の切り出しとテンプレートマッチング TM^S

C) 形状を維持する追跡結果の選出 TM^S によるマッチング率結果から, 最も形状を維持する点 p_i^t を求め, 追跡結果とする. まず, マッチング率画像と同領域のエッジ画像 E^t を乗算することで, 結果をエッジ上のみに限定する (図 29).



図 29 マッチング率画像とエッジ画像 E^t の一部の乗算

乗算画像上で, マッチング率が閾値より高い点を, 候補点 $r_j \in \mathbb{R}^2, j=0,1,2,\dots, n$: 候補の頂点数とする.

この追跡候補点 r_j の中から, 時刻 $0, t-1$ の局所形状を最も維持する追跡結果を選出する (図 30). ここでは, 局所変形に対するロバストな追跡のために, ラプラシアン[56]を求め, ラプラシアンを最も維持する点を求める. ラプラシアンは, 輪郭上における両隣の頂点の重心との位置関係を表すベクトルであり, 以下のように計算する.

$$\begin{aligned}\Delta p_i^t &= (p_{i-1}^t - p_i^t) - (p_i^t - p_{i+1}^t) \\ \Delta r_j &= (p_{i-1}^t - r_j) - (r_j - p_{i+1}^t)\end{aligned}$$

次に，時刻 0, t-1 の Δp_i^0 と Δp_i^{t-1} を以下のように合成する．

$$\Delta p_i^{com} = w\Delta p_i^0 + (1-w)\Delta p_i^{t-1}, (0 \leq w \leq 1)$$

さらに， $\arg \min_{x \in \{r_j\}} |\Delta p_i^{com} - \Delta r_j|$ ， $\min |\Delta p_i^{com} - \Delta r_j|$ となる候補点 r_j を追跡結果 p_i^t とする．

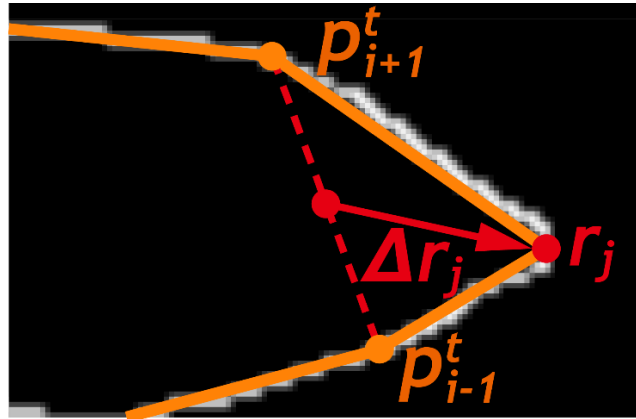


図 30 ラプラシアン Δr_j

D) コーナー補正 コーナーを補正する．手順 C の結果では，コーナーの点が実際とは，ずれが生じることがあるため，コーナーの位置の補正を行う．ここでは，120 度以下の角度を持つ点 p_i^t を，コーナー p_i^* とする．エッジ情報のコーナーは，エッジ画像 E^t に輪郭抽出を行うことで得られる輪郭の頂点より選出し， $s_k \in \mathbb{R}^2, k=0,1,2,\dots, o$: 頂点数とする． s_k と p_i^* の間が距離 d 以下の時， s_k と p_i^* のラプラシアンを比較する (図 31)．ここでは， $d=10$ である． $|\Delta p_i^t - \Delta s_k|$ が，閾値以下なら， $\Delta p_i^t = \Delta s_k$ とする．

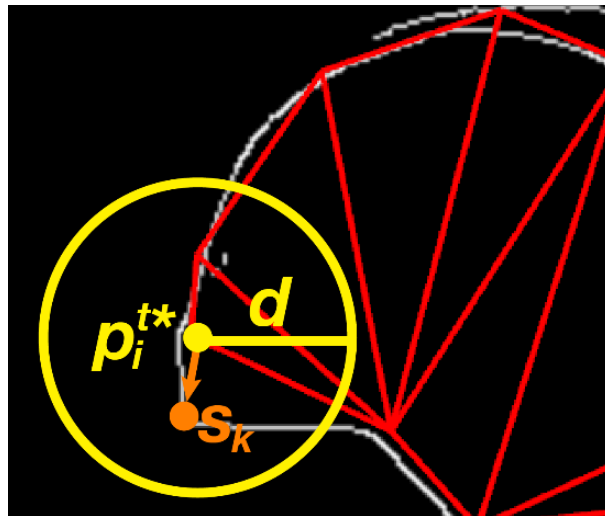


図 31 コーナー補正

E) Kalman フィルタによるノイズ除去と予測 ノイズ除去と予測を行うため、追跡結果 $\Delta p_{i,j}^t$ に対し、Kalman フィルタを適応する (図 27d) . Kalman フィルタは、前の状態から現在の状態を予測し、観測値を元に修正を行うことを繰り返す. ここでは、時刻 $t-1$ の追跡結果 p_i^{t-1} から、現在の状態 (予測値) を計算し、観測値 p_i^t を元に修正する. 投影の際は、Kalman フィルタによって求めた予測値を出力する.

F) 遮蔽領域を考慮した映像のマスキ処理 投影領域が初期状態にあるときのみ限定して、プロジェクターと投影対象間の遮蔽が発生する時に、遮蔽物への映像投影を行わないようにマスキ処理を適用する. まず、投影対象が初期状態にあるか判断するため、毎フレーム、時刻 0 の葉全体のテンプレートを初期位置周辺でマッチングする. マッチング率が高い時、葉は初期状態 (図 32a) にあると判断し、 $p_i^{t-1} = p_i^0$ とする. また、追跡結果にマスキ処理によって、手などに遮蔽された領域にはエフェクトアニメーションの投影を行わない (図 32c). マスクは入力画像 I^t (図 32b) に二値化を行うことで生成する. ここで、投影対象は赤外線照明によって背後から照らされているため、遮蔽した領域は入力画像 I^t 中で黒く映る.



図 32 (a) 葉の初期状態, (b) 遮蔽された際の入力画像,
(c) (b)を2値化した画像で投影内容をマスクした結果.

5.2 結果考察

本研究によって作成した投影結果を図 33, 図 34, 図 35, 図 36 に示す. 本手法で示す追跡結果と投影結果の作成には, Windows PC(Intel Core i9-9900(3.1GHz) PC)を使用した. また, エフェクトの投影結果や作品制作では, Windows PC(Intel Core i7-7700HQ(2.8GHz) PC)を使用した. カメラは解像度 $2048 \times 1088\text{px}$ でフレームレート 169fps の STC-MBCM200U3V-NIR を, プロジェクターは解像度 $1280 \times 720\text{px}$ の Taxan KG-PL081W, 赤外線照明は S8100-IR を使用した.

まず, 本追跡手法では, 端点だけでなく輪郭の各頂点の位置を求めているため, 手法 1 では投影できなかったイチョウ形やカエデ形の葉など, 様々な形状の葉に投影することができた (図 33abc). さらに, 葉のみに限らず, 薄型の花にも投影できた (図 33d). また, 植物の動きや局所的な変形に対しても, 比較的安定して追跡することができた (図 34, 図 35).



図 33 様々な形状の葉, 花への投影結果

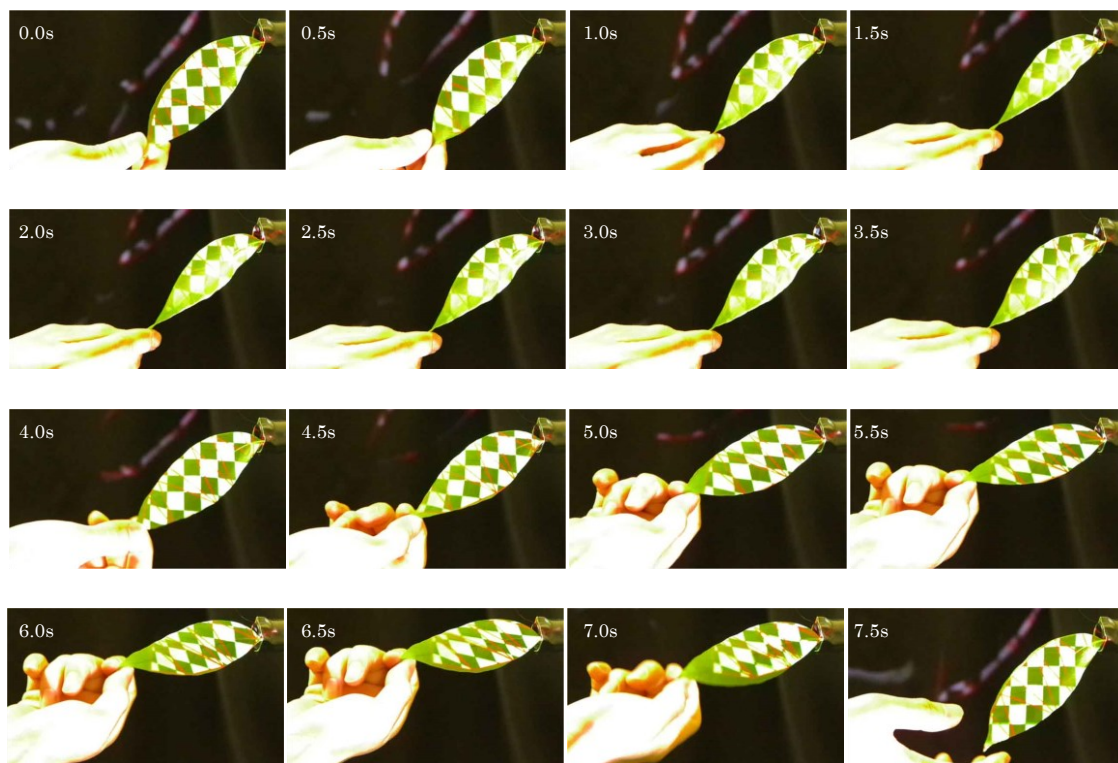


図 34 動く対象への投影結果.



図 35 葉の局所変形に対する追跡結果.

ユーザーインタラクションにおいては、素手で対象に触れて動かすことが可能になった（図 33，図 34，図 35）。また，遮蔽に対しては，投影対象が初期位置にあるときのみ，映像のマスキングを行い，投影対象のみに対して映像投影を行うことができた（図 36）。しかし，遮蔽領域の判定を行っていないため，手などで触れて動かすとき，追跡結果は触れた箇所を避けるような結果となり，ズレが生じた（図 34 における 5.0s~7.0s）。



図 36 初期状態の投影対象において遮蔽領域を考慮した投影映像のマスキング.

第6章

手法 3 : 様々な形状の植物へのインタラクティブな DPM

本章では、手で触れる等の接触や遮蔽を伴うインタラクションに頑健な追跡手法を提案し、インタラクティブな DPM システムを作成する。手法 2 では、投影対象が初期位置にある時に限定して、遮蔽時の映像投影に対応しているが、動く対象への遮蔽が発生する時、安定した追跡を行うことができない。そのため本手法では、毎フレーム、投影領域の輪郭の各頂点において、遮蔽の有無を判定し、遮蔽されている点を再計算することで、遮蔽時も頑健に追跡する。また、静電容量式センサを用いて接触時間を取得し、遮蔽判定と組み合わせることで、インタラクティブにエフェクトが変化する DPM システムを実装する。そのため、インタラクティブなエフェクトを 4 種類生成する。本手法は、**NICOGRAPH International 2021 Full Paper[57]**、**SIGGRAPH 2021 Labs[58]**、**SIGGRAPH ASIA 2021 Posters[59]**、**Laval Virtual 2021[60]**にて発表した。

6.1 手法詳細

プロジェクターとカメラの位置合わせは、手法 1 と同様に行う。本節では、投影領域の選出方法、遮蔽判定を含む追跡手法、インタラクティブな DPM システム、インタラクティブな 4 種類のエフェクトについて述べる。

6.1.1 投影領域の選出

初期フレーム($t=0$)において、投影領域を決定する。まず、赤外線カメラで取得した初期フレーム画像から、canny 法によってエッジ画像 E^0 を取得する。このとき、投影対象は遮蔽されていないものとする。次に、入力画像 I^0 からエッジ画像 E^0 を減算した画像 $I^{0'}$ に対し、ガウス式適応型閾値処理を適用する。二値化画像にモルフォロジー処理のオープニングを適用し、白領域の輪郭を抽出する。抽出された輪郭の内部を、マウスクリックにより、追跡する植物の領域を選択する。選択した葉の領域の輪郭の頂点 $p_i \in \mathbb{R}^2, i=1,2,\dots, n, n$: 頂点数を毎フレーム追跡する。また、輪郭の各頂点 p_i^0 について内角 θ_i^0 を計算し、 $\theta_i^0 = 0 \sim 2\pi/3, 4\pi/3 \sim 2\pi$ のとき、特徴点 $q_j \in \mathbb{R}^2, j=1,2,\dots, m, m$: 特徴点数, とする (図 37 赤点)。

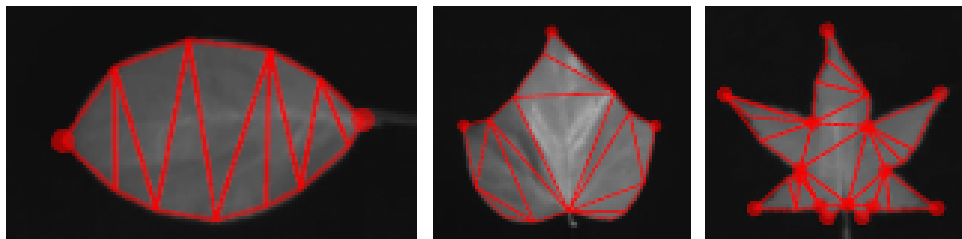


図 37 追跡する輪郭と特徴点.

6.1.2 投影領域の追跡

毎フレーム、輪郭の各頂点 p_i^{t-1} のオプティカルフローを計算し、同時に遮蔽判定を行う。遮蔽されていない点は、エッジ画像上で探索し、遮蔽された点は、遮蔽されていないと判定された点を用いて再計算する。

毎フレーム、以下の流れで処理を行う。

- A. 輪郭の各頂点のオプティカルフローを計算する。
遮蔽判定を行う。
- B. 遮蔽されていない点を追跡する。
- C. 遮蔽されている点の位置を再計算する。
- D. 追跡結果に Kalman フィルタを適用したものを追跡結果とする。

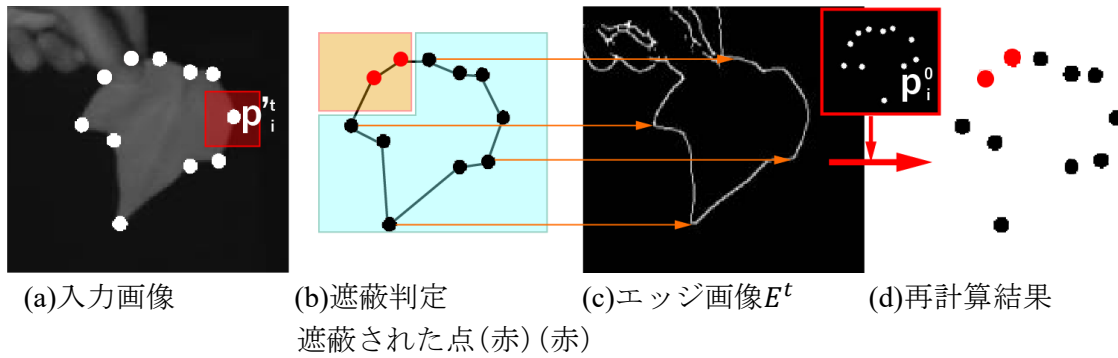


図 38 追跡手法概要図。

A) オプティカルフローの計算と遮蔽判定 まず、入力画像 I^{t-1}, I^t から、各頂点 p_i^{t-1} の移動ベクトルとしてオプティカルフロー $o_i^t \in \mathbb{R}^2$ を計算し、移動後の点 $p_i^t (= p_i^{t-1} + o_i^t)$ を求める (図 38a)。時刻 t における各頂点の平均誤差値 e_i^t を、過去 10 フレームでそれぞれ p_i^{t-1} および p_i^t の周辺 18 ピクセル (図 38a 赤領域) の輝度の差分を平均化して求める (図 38b)。各頂点の e_i^{avg} を k-means 法によって 2 つのクラス C_1 : 遮蔽されている可能性が低いクラス, C_2 : 遮蔽されている可能性が高いクラスに分割する。ただし、クラス数が 2 個以下の時のクラス内誤差平方和 (SSE) の差が閾値 0.01 以下なら、クラス数は 1 つとし、全ての点をクラス C_1 に入れる。 C_1 に 25 フレーム以上連続で属したら点 p_i^t は「遮蔽されていない」、 C_2 に 25 フレーム以上連続で判定されたら点 p_i^t は「遮蔽されている」と判定する。この判定を用いて、遮蔽されていない点の追跡から計算する。このとき、 e_i^t 算出に用いるフレーム数、クラス数の判定に用いる閾値、遮蔽判定に用いる連続フレーム数は、実験によって経験的に得られた値を用いた。

B) 遮蔽されていない点の追跡 遮蔽されていない点は，入力画像 I^t から取得したエッジ画像 E^t 上で探索する（図 38d 黒点）． E^t は， I^t を二値化収縮した画像を，cannyエッジ画像から減算することで取得する（図 39）．

(a) bcの入力 I^t

(b) cannyエッジ

(c) 二値化し収縮

(d) $E^t=b-c$

図 39 エッジ画像の抽出.

まず，特徴点 q_j^t を求める．輪郭抽出によってエッジ画像 E^t から得た輪郭の頂点から，内角が $0 \sim 2\pi/3$, $4\pi/3 \sim 2\pi$ である点 $r_k \in \mathbb{R}^2$, $k=1,2,\dots,1,1$: 内角 $0 \sim 2\pi/3$, $4\pi/3 \sim 2\pi$ の頂点数，を算出し， q_j^t の候補とする． p_i^t と同様に求めた q_j^t 周辺（図 40 緑点線内）で， q_j^t と r_k^{edge} の内角の大きさ（図 40 オレンジ）が最も近い点を追跡結果とする．ただし，1 フレーム間における投影対象の回転量は微小であると仮定し，互いの内角の二等分線方向の角度（図 40 青矢印）の差は $\pi/18$ 以下とし，内角の向きが近い点を選ぶ．この値は，実験によって経験的に得られた値を用いた．

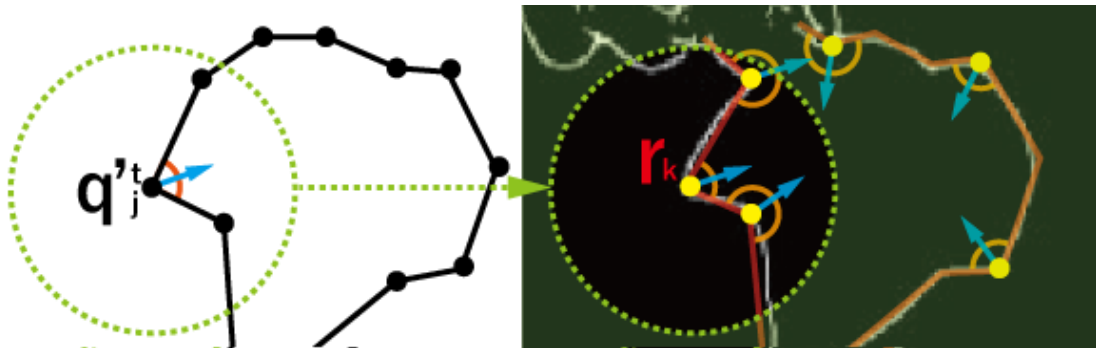
(a) optical flow による移動後の点 q_j^t (b) E^t 上の候補点 r_k

図 40 追跡で考慮する内角（オレンジ）と二等分線方向（青矢印）と探索範囲(緑点線内).

次に，特徴点以外の点の追跡結果を求める．まず， E^t におけるエッジ上の画素 s_l ($s \in \mathbb{R}^2$, $l=1,2,\dots,o$, o : エッジ上のピクセル数)で，頂点 p_i^t とのユークリッド距離が最も小さい点を p_i^t とする．次に，初期フレームの輪郭形状をなるべく保つため，頂点毎に時刻 0 におけるラプラシアン Δp_i^0 に最も近い点をエッジ上の点 s_l から探索する．以下の式の反復計算において，各頂点における一つ前の結果との距離の平均が 0.01 以下になった時，または反復計算の回数が 10 回になった時，計算を収束する．

$$\begin{aligned}\Delta \mathbf{p}_i^0 &= (\mathbf{p}_{i-1}^0 - \mathbf{p}_i^0) - (\mathbf{p}_i^0 - \mathbf{p}_{i+1}^0), \\ \Delta \mathbf{s}_l &= (\mathbf{p}_{i-1}^t - \mathbf{s}_l) - (\mathbf{s}_l - \mathbf{p}_{i+1}^t), \\ \mathbf{p}_i^t &= \arg \min_{x \in \{\mathbf{s}_l\}} |T_i \Delta \mathbf{p}_i^0 - \Delta \mathbf{s}_l|.\end{aligned}$$

拡大縮小と回転を表す行列 T_i は、 $\mathbf{p}_{i+1}^0 - \mathbf{p}_{i-1}^0$ をこれによって一次変換したベクトルが $\mathbf{p}_{i+1}^t - \mathbf{p}_{i-1}^t$ となるよう計算される。

C) 遮蔽されている点の追跡 遮蔽されていないと判定された点をもとに、リジッド変形[61]を行うことで、遮蔽されている点の追跡結果を求める。リジッド変形は、複数の制御点の変形前後の位置から、歪みが少ない形状変形を行う手法である。ここでは、遮蔽されていない点を制御点とし、初期形状をリジッド変形することで、遮蔽されている点の位置を求め、追跡結果 $\mathbf{p}_i^t$ とする（図 38d 赤点）。

D) Kalman フィルタの適用 手法 2 と同様、投影エフェクトの描画時には、追跡結果 $\mathbf{p}_i^t$ に Kalman フィルタを適用した予測値を用いる（図 38e）。予測値を用いることで、ノイズ除去と予測を行う。

6.1.3 インタラクティブシステム

遮蔽判定と、静電容量式センサによる接触判定を組み合わせることで、植物との接触箇所 $\mathbf{p}_{touch}^t$ 、接触時からの経過時間 $t_{touch}(\geq 0)$ を計算する。静電容量式タッチセンサはコンピュータと抵抗で構成される。コンピュータで電荷の発生・検出を制御することで、接触を検出する。抵抗を介して両電極を接続し、分岐させた電極を投影対象の植物の鉢植えに電極を挿し、葉への接触を検出する（図 41）。接触は、電極から電圧を送信して受信までにかかる時間の計測によって判定する。ユーザーが投影対象に触れる際、静電容量が大きくなり、電圧の送受信の時間差も大きくなる。送受信の時間差が 10ms 以上の時、接触と判定する。静電容量式センサで接触を検出した際、遮蔽判定された点 $\mathbf{p}_i^t$ の中で $\mathbf{e}_i^{avg}$ が最も高い一点を、接触した点 $\mathbf{p}_{touch}^t$ として判定する。

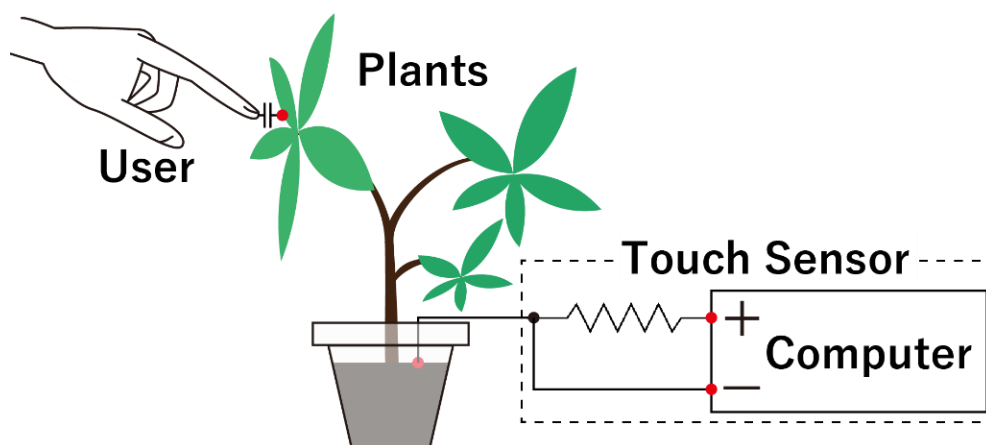


図 41 静電容量式タッチセンサの設置図.

6.1.4 4 種類のインタラクティブなエフェクトアニメーション

エフェクトは、生物発光や自然界における光をモチーフに、投影対象自身が発光しているように感じられる表現を目指し、対象物体の輪郭形状から自動で生成する。また、接触時間と遮蔽箇所を用いて、インタラクティブに変化するエフェクトを作成する。

1) ホタルをモチーフとしたエフェクト ホタルをモチーフに、 $1/f$ ゆらぎを用いた発光のリズムで投影領域全体が点滅するエフェクトを作成する（図 42，図 43）。 $1/f$ ゆらぎは、様々な自然現象に見られ、人に癒し効果を与えるとされている現象である。まず、投影領域内の各ピクセルにおいて、輪郭線からの最近距離を計算し、距離場を生成する。この距離場を用いて、各ピクセルの色を決定する。接触により対象の光が増すように色を変化させる。

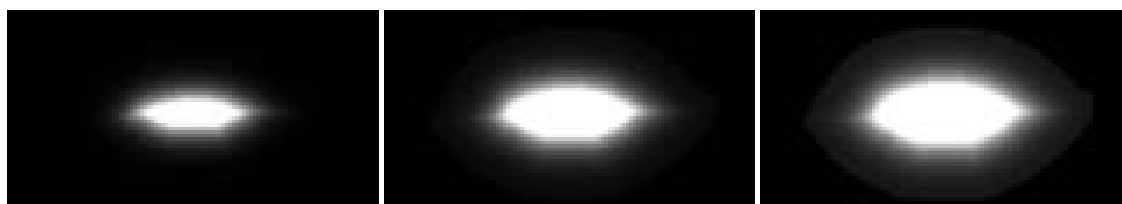


図 42 ホタルをモチーフとしたエフェクトの出力結果.

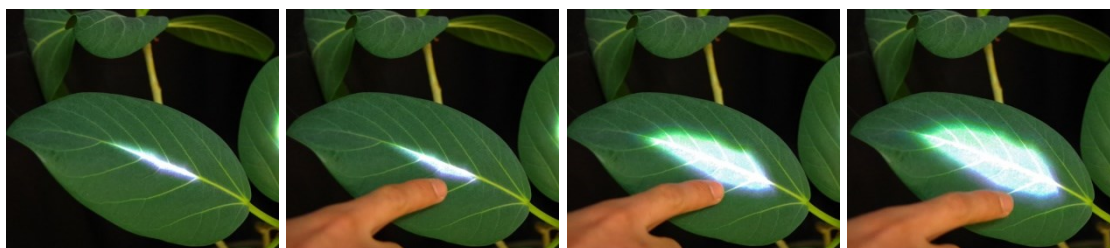


図 43 ホタルをモチーフとしたエフェクトの投影結果.

2) クシクラゲをモチーフとしたエフェクト 光の反射によってネオンサインのように光るクシクラゲをモチーフに、輪郭形状に沿って配置した光球が点滅するエフェクトを作成する(図 45, 図 46). まず、輪郭の頂点の中からユーザーが始点を選び、輪郭形状に沿って始点から各特徴点に向かう線(図 44 赤線)を計算する. 求めた線上に一定の間隔で光球を配置し、点滅させる. クラゲの動きにも $1/f$ ゆらぎが含まれていることから、この点滅にも $1/f$ ゆらぎを用いる. 接触により、接触箇所から光球の色を変化させる.

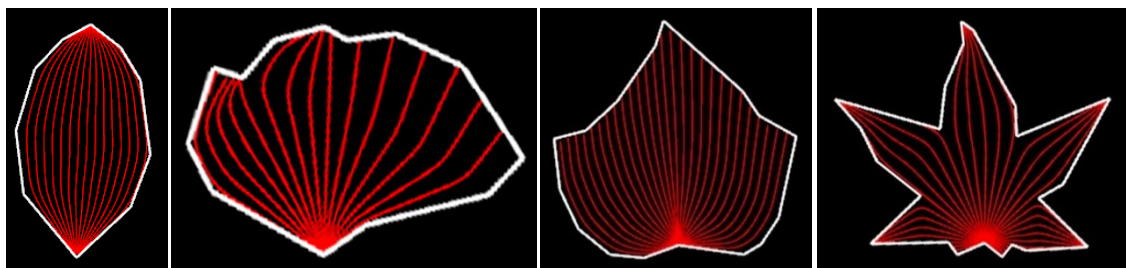


図 44 輪郭形状に沿って、始点から各特徴点に向かう線.

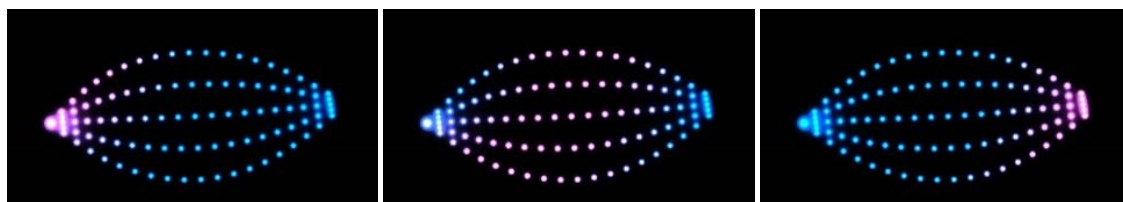


図 45 クシクラゲをモチーフとしたエフェクトの出力結果.

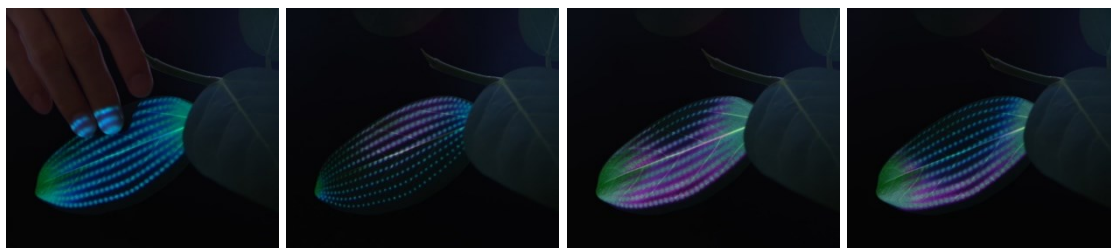


図 46 クシクラゲをモチーフとしたエフェクトの投影結果.

3) 細胞をモチーフとしたエフェクト 植物の細胞をモチーフに、輪郭内をボロノイ分割し、各領域の面積が変化するエフェクトを作成する(図 47, 図 48). 輪郭内にランダムに母点を配置し、ボロノイ分割する. 各領域の角を丸くすることで、細胞のような形状を生成する. また、細胞が核をもつように、領域内に歪んだ円を配置する. 接触により、細胞の反応が伝搬していくように、触れた箇所から細胞領域の大きさを収縮させる.

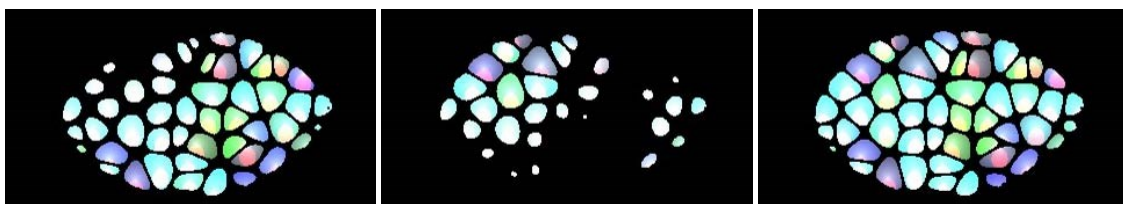


図 47 細胞をモチーフとしたエフェクトの出力結果.

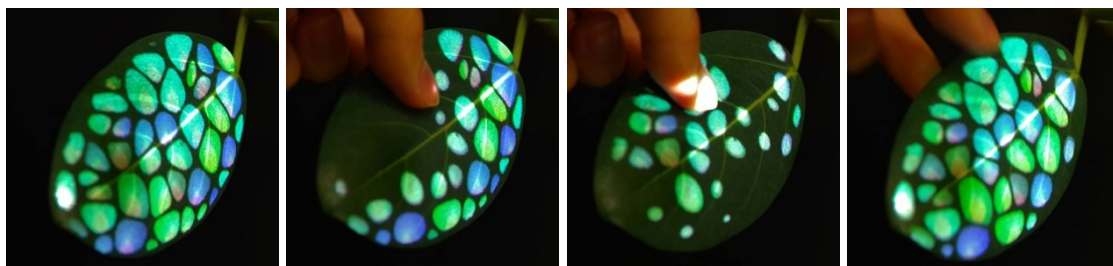


図 48 細胞をモチーフとしたエフェクトの投影結果.

4) 水晶をモチーフとしたエフェクト 光の屈折や反射によって輝く水晶をモチーフに、投影領域を輝く三角形で構成するエフェクトを作成する (図 50, 図 51). まず、輪郭内に点をランダムに配置し、輪郭内をドロネー分割する. 次に、複数の光球が動く映像を生成し、映像上で各三角形の重心位置のピクセルの色を参照し (図 49 左), 三角形の色とする (図 49 右). 光球が動く映像の色を参照することで、三角形の位置に基づいて滑らかに色を変化させることができる. 接触により、触れた箇所から光球を生成し、三角形の色を変化させる.

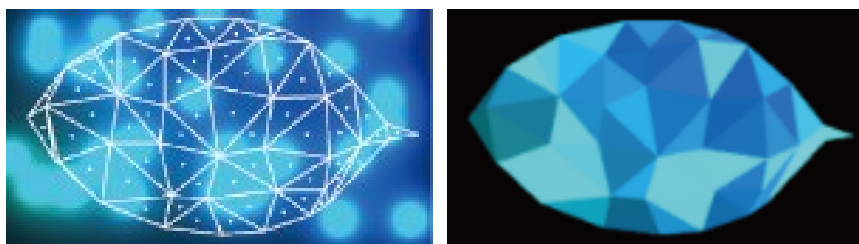


図 49 光球を動かした映像と三角形メッシュ (左), メッシュエフェクト (右).



図 50 水晶をモチーフとしたエフェクトの出力結果.



図 51 水晶をモチーフとしたエフェクトの投影結果.

6.2 結果考察

本研究によって作成した追跡結果と投影結果を図 52, 図 53, 図 54 に示す. 本手法で示す結果の作成には, Windows PC(Intel Core i9-9900(3.1GHz) PC)を使用した. カメラは解像度 $2048 \times 1088\text{px}$ でフレームレート 169fps の STC-MBCM200U3V-NIR を, プロジェクターは解像度 $1280 \times 720\text{px}$ の Taxan KG-PL081W, 赤外線照明は S8100-IR を用いた. 静電容量式タッチセンサには, Arduino Uno, $1\text{M}\Omega$ の抵抗を用いた.

本システムを様々な形状の葉に適用し (図 53, 図 54), エフェクトアニメーションを投影した (図 52). これらの結果より, 本手法によって植物の追跡, および現実空間での投影の双方が, 自動で行えることがわかる. 本手法では, 特徴点の追跡結果を最初に求めるため, 特徴点が一つ以上同定できる限り, 動きや変形を柔軟に追跡できる. また, 一枚の葉 (追跡点 10 個) に対し, 3.6ms の速度で追跡でき, 大きくずれることなく追跡できた. 対象の数 (追跡点の数) が増えると, 処理時間は増加するものの, 9 枚まで $16.6\text{ms}(60\text{fps})$ で追跡できることを確認した.

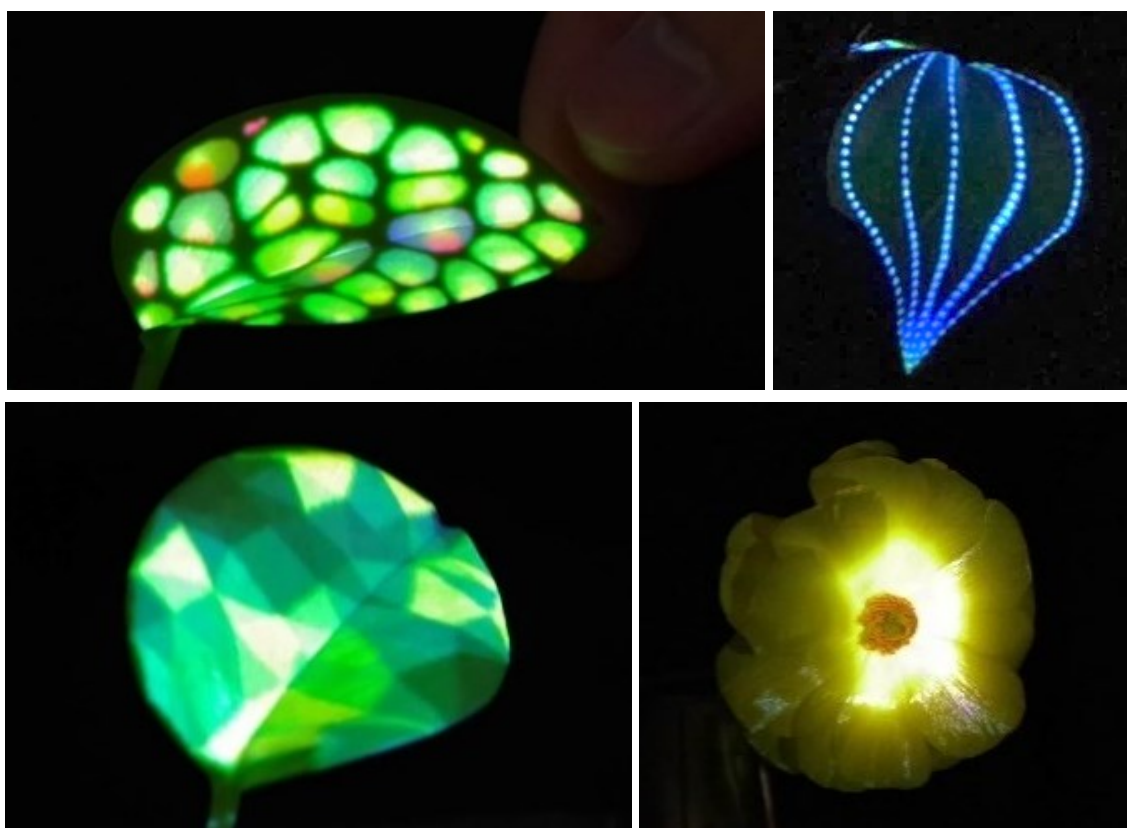


図 52 エフェクトの投影結果 (a)細胞をモチーフとしたエフェクト, (b)クシクラゲをモチーフとしたエフェクト, (c)水晶をモチーフとしたエフェクト, (d)ホテルをモチーフとしたエフェクト.

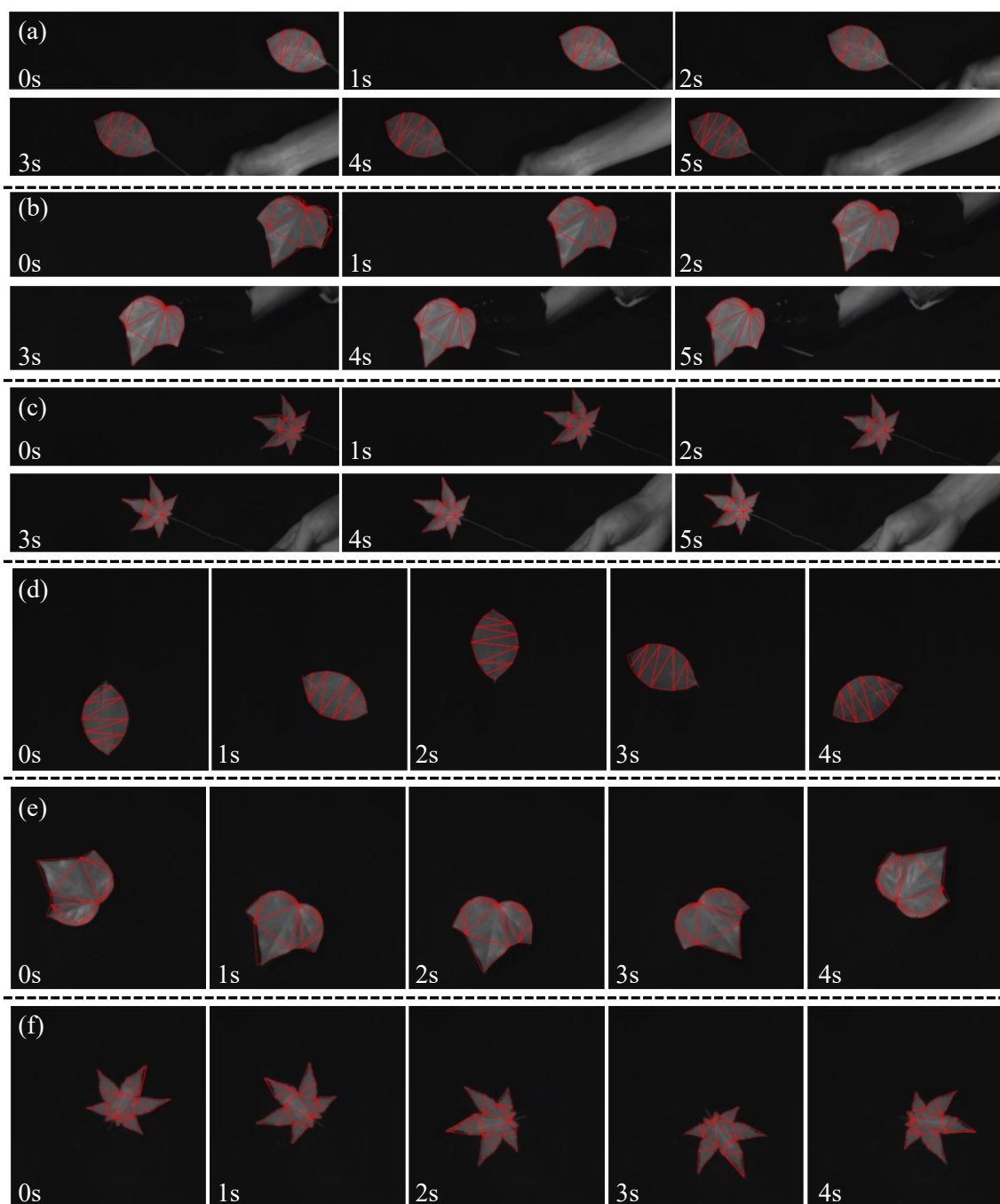


図 53 動く葉に対する追跡結果 (a), (b), (c)平行移動させた（楕円形，ハート形，カエデ形の）葉に対する追跡結果，(d), (e), (f)カメラ方向の軸を中心に約 360 度回転させた（楕円形，ハート形，カエデ形の）葉に対する追跡結果．

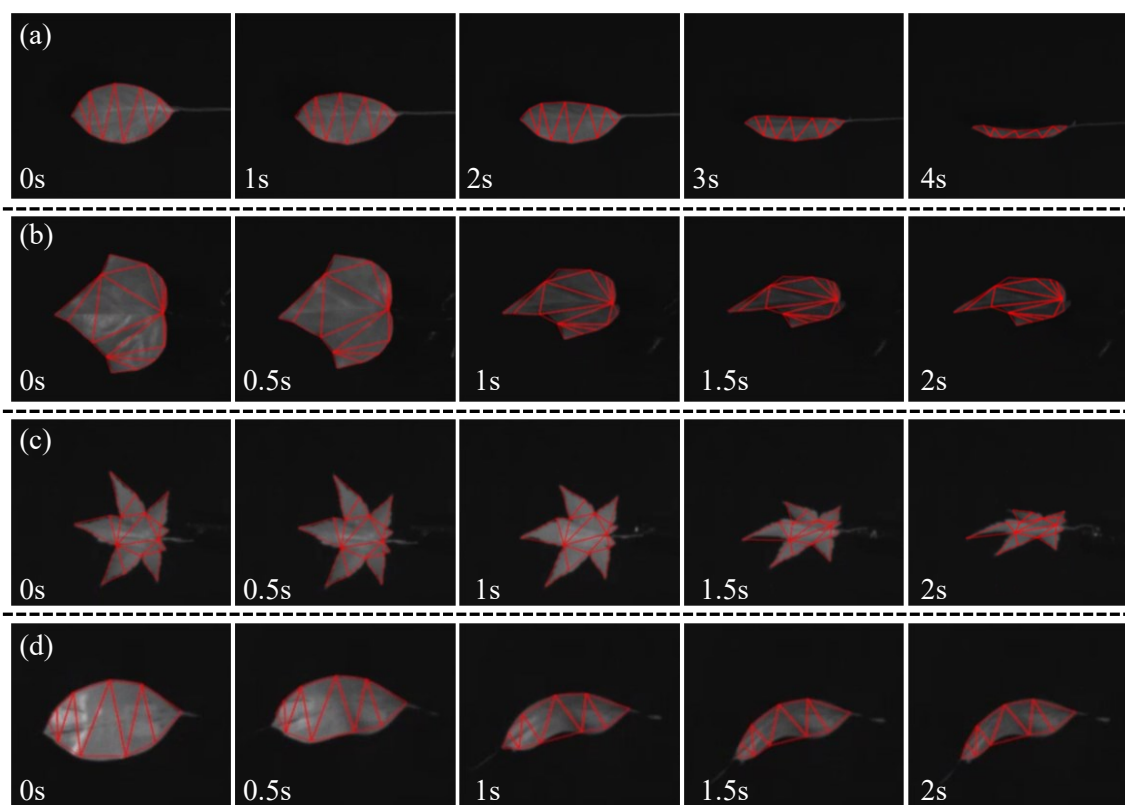


図 54 動く葉に対する追跡結果 (a), (b), (c)葉の軸方向を中心に回転させた（楕円形，ハート形，カエデ形の）葉に対する追跡結果，(d)変形させた楕円形の葉に対する追跡結果．



図 55 遮蔽判定の結果とリジッド変形による補正．

また，遮蔽時は図 55 のように，輪郭の各頂点について正しい遮蔽判定が行えており，リジッド変形による補正ができている．インタラクションは，画像処理による遮蔽判定とタッチセンサを組み合わせることで，より直感的な反応を得られた．

第7章

手法 4 : 大量の楕円形の葉への DPM

本章では、大量の投影対象への DPM を実現するため、対象を楕円形の葉に限定し、対象領域を選出・追跡する手法を提案する。これまでの手法ではおよそ 10 枚程度の対象まで処理が可能であったが、本手法では 100 枚を目指す。事前の投影領域選出においては、大量の葉の選択が必要であるため、少ない入力で直感的に選出する UI を提案する。追跡においては、大量の対象の位置を高速に求めるため、対象を近似した楕円の位置と角度、画像上における大きさのみを最小二乗法で求め、追跡する。また、対象同士における接触・重なりの可能性が高い位置を判定し、可能性が低い位置から、初期形状を維持する位置を求めることで、対象同士の接触・遮蔽にも頑健な追跡手法を提案する。

7.1 手法詳細

プロジェクターとカメラの位置合わせは、手法 1 と同様に行う。本節では、柔軟な投影領域の選出方法、楕円近似を用いた追跡手法、ユーザー入力を適用したエフェクト作成について述べる。

7.1.1 投影領域の選出

最初に得たシーンの赤外線画像に対し、画像処理を用いて自動で投影領域を抽出した後、そのうち投影対象としない領域の削除や、2 枚の葉が接触し一つの領域となっている部分の分割などの修正を、マウスを用いたユーザー入力によって柔軟に行う。

初期フレーム($t=0$)において、以降追跡する投影領域を以下のように決定する。

- A. 赤外線カメラで取得した初期フレーム画像 I^0 に対し、ガウシアンフィルタを適用し、平滑化画像を取得する。
- B. 平滑化画像から canny 法によってエッジ画像 E^0 を取得する。
- C. 平滑化画像に閾値処理を適用し二値化画像を取得する。
- D. 二値化画像からエッジ画像 E^0 を減算した画像に対し、モルフォロジー処理のオープニングを適用し、白領域の輪郭を抽出する。
- E. 抽出された輪郭に対し、凹部分の輪郭の頂点を用いて、輪郭の分割と削除を行う。
- F. E で求めた輪郭に対し、マウス入力によって分割と削除を行う。

D の結果では、葉同士の隣接や重なりによって、複数の対象が一つの領域として繋がっている輪郭や、枝等の投影対象ではない物体との隣接によって、投影対象では無い領域を含む輪郭が抽出されることがある。そのため、E では、輪郭における内角が π 以上の頂点を凹部分の頂点として求め、輪郭の分割と削除を行う。まず、D で抽出された輪郭に対し、制約付きドロネー分割を行う。次に、分割された各三角形において、凹部分の頂点を 2 つ以上持つとき、その凹部分の頂点を結ぶ線分によって輪郭を分割する。

F における、マウス入力を用いた分割では、マウスドラッグの押し放しで輪郭上の 2 つの頂点を選択し、2 点を結ぶ線分によって輪郭を分ける。削除は、輪郭を遮るようにマウスドラッグで線を描画し、入力線と各輪郭が 2 回以上交差する時に実行する。

また、F の選出結果、追跡には適さないような小さい輪郭も選出される。そのような輪郭を削除するため、輪郭面積が閾値以下の時、追跡対象から外す。閾値は、UI 上のスライダーを移動させることによって、変更することができる。

分割と削除の結果、残った各輪郭の頂点を毎フレーム追跡する輪郭の頂点 $p_i^0 \in \mathbb{R}^2, i=1,2,\dots, n, n$: 頂点数, とする。つづいて、選出された各領域の輪郭頂点の点列を楕円 O^0 で近似する。また、近似楕円 O^0 の長軸と短軸を定数倍（本研究では 1.1 倍）に拡大した、大きい楕円 Q^0 を求める。さらに、それぞれの楕円 O^0, Q^0 において、楕円中心から p_i^0 方向に伸ばした半直線との交点を $o_i^0, q_i^0 \in \mathbb{R}^2$ とする。本手法の追跡では、毎フレーム、前フレームの楕円 O^{t-1} 上の点 o_i^{t-1} を用いながら、楕円 O^t を追跡する。

7.1.2 投影領域の追跡

追跡する各対象の領域はその頂点を楕円で近似し、その周囲のエッジ上のみに次のフレームの対象の頂点があると想定することで、計算速度および追跡の精度を上げる。前フレームの近似楕円から現フレームの頂点 p_i^t を求め、点列を用いて映像をテクスチャマッピングする。このとき、他の対象が重なると、他の対象の頂点を現フレームの頂点 p_i^t として求める可能性があり、追跡の精度が下がる。そのため、他の対象との重なった頂点の位置を修正した上で楕円近似する。この処理は毎フレーム行う。以下はその詳細である。

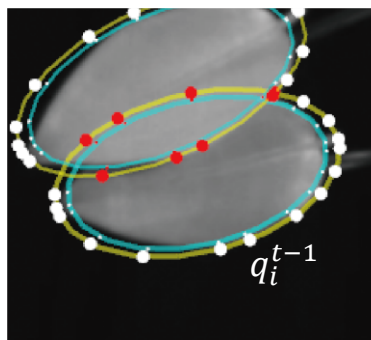
楕円近似は、点集合に対して二乗誤差が最小になる楕円を求める[62]。これまでの追跡手法ではオプティカルフローやテンプレートマッチングなど画像ベースの方法でフレーム間の類似する箇所を求め追跡することから、計算時間を要する。本手法では、エッジ画像から抽出したベクタ情報ベースで一つ一つの対象を楕円近似することで高速な追跡を目指す。

また，追跡に用いるエッジ画像 E^t は以下の処理で求める．

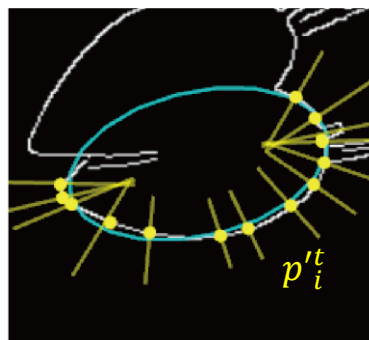
1. 赤外線カメラで取得した初期フレーム画像 I^t に対し，ガウシアンフィルタを適用し，平滑化画像を取得する．
2. 平滑化画像から canny 法によって canny エッジ画像を取得する．
3. I^t を二値化収縮した画像を，canny エッジ画像から減算し，エッジ画像 E^t とする．

下記の A～E は処理を示す．A では投影領域同士の重なりによる他の対象の頂点を修正するため，各頂点における他の投影領域と重なりを楕円の衝突判定によって求める．B, C では追跡結果 p_i^t を求める．A で衝突しない点の追跡結果を近傍のエッジ画像 E^t 上で求め，衝突した点の追跡結果は初期フレームの位置関係から相対的に求める．D ではこれらの点に対して楕円近似を行い，現フレームの楕円 O^t を求める．具体的には以下の流れで追跡する（図 56）．

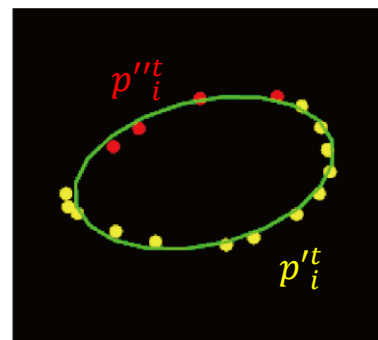
- A. 大きい楕円 Q^{t-1} 同士の衝突判定を行い，楕円上の点 q_i^{t-1} の中で他の投影領域と衝突した点を求める（図 56a）．
- B. 点 o_i^{t-1} で， O^{t-1} における法線とエッジ画像 E^t のエッジの交点を計算し， $p_i'^t$ とする（図 56b）．
- C. 他の投影領域と衝突した頂点の追跡結果 $p_i''^t$ を計算する（図 56c）．
- D. B, C で求めた頂点 $p_i'^t$ ， $p_i''^t$ を追跡結果 p_i^t とし，Kalman フィルタを適用したものを出力結果とする．
- E. 頂点 p_i^t を楕円 O^t で近似する（図 56c）．



(a)重なり判定
衝突した点（赤）



(b)楕円の法線と
エッジの交点（黄）



(c)衝突した点の対応点 $o_i''^t$
と近似楕円 O^t

図 56 追跡手法の概要．

A) 投影領域同士の重なり 複数の投影対象が干渉する場合、 Q^{t-1} 同士の衝突判定を行うことで、投影対象同士で重なる可能性が高い箇所を疑似的に求める。まず、 Q^{t-1} 同士の衝突判定を行う（図 56a）。 Q^{t-1} 同士が衝突しているとき、頂点 q_i^{t-1} ごとに内外判定を行い、他方の楕円の内側にあるとき衝突した点（図 56a 赤点）とする。

B) 近似楕円の法線とエッジの交点の計算 衝突しない箇所について、現フレームにおける対応箇所をエッジ上で求める。まず、 o_i^{t-1} の位置で O^{t-1} における法線（図 56B 黄線）を計算する。次に、エッジ画像 E^t から輪郭抽出によってエッジベクタを抽出する。このとき、枝と葉が接する箇所では、エッジベクタを抽出できない可能性がある。そのため、抽出したエッジベクタの連結情報を用いて、隣接点で囲まれる内角が $0 \sim 2\pi/3$, $4\pi/3 \sim 2\pi$ の点を特徴点として計算し、特徴点同士で結んだ線をエッジベクタとして追加する。このエッジベクタと法線の交点 p_i^t （図 56B 黄点）を求める。一つの法線に対し交点が複数ある時は、 p_i^{t-1} とのユークリッド距離が最も短い交点を p_i^t とする。また、交点が見つからない時は、前フレームの追跡結果 p_i^{t-1} を p_i^t とする。

C) 衝突した点の追跡 B で求めた対応点 p_i^t （図 56c 黄点）を制御点としたリジッド変形[61]（6.1.2 C）を初期形状に行うことで、衝突した点の対応点を計算し、点 p_i^t とする。

D) Kalman フィルタの適用 B, C で求めた頂点 p_i^t , p_i^{t-1} を追跡結果 p_i^t とする。手法 2, 3 と同様、投影エフェクトの描画時には、追跡結果 p_i^t に Kalman フィルタを適用した予測値を用いることで、ノイズ除去と予測を行う。

E) 近似楕円の追跡 まず、B, C で求めた点 p_i^t , p_i^{t-1} を用いて、最小二乗法による楕円近似を行う（図 56c）。求めた楕円（図 56c 緑線）を近似楕円の追跡結果 O^t とする。また、初期フレームと同様、大きい楕円 Q^t 、各楕円中心から p_i^t 方向に伸ばした半直線との交点を o_i^t , q_i^t を計算する。

7.1.3 ユーザー入力を適用したエフェクト作成

枝の先端に向かってエフェクトを生成するため、枝に対する葉の位置・順番が必要である．UI 上でユーザーが葉と枝の接する位置を頂点 $r_j \in \mathbb{R}^2, j=1,2,\dots,m, m$: 頂点数, としてマウスクリックで順番に指定 (図 57 赤点) することによって, 枝にあたる線を描画し (図 57 白線), 枝と葉の位置関係を取得する．本システムでは, この情報を用いた 2 つのエフェクトを提案する．

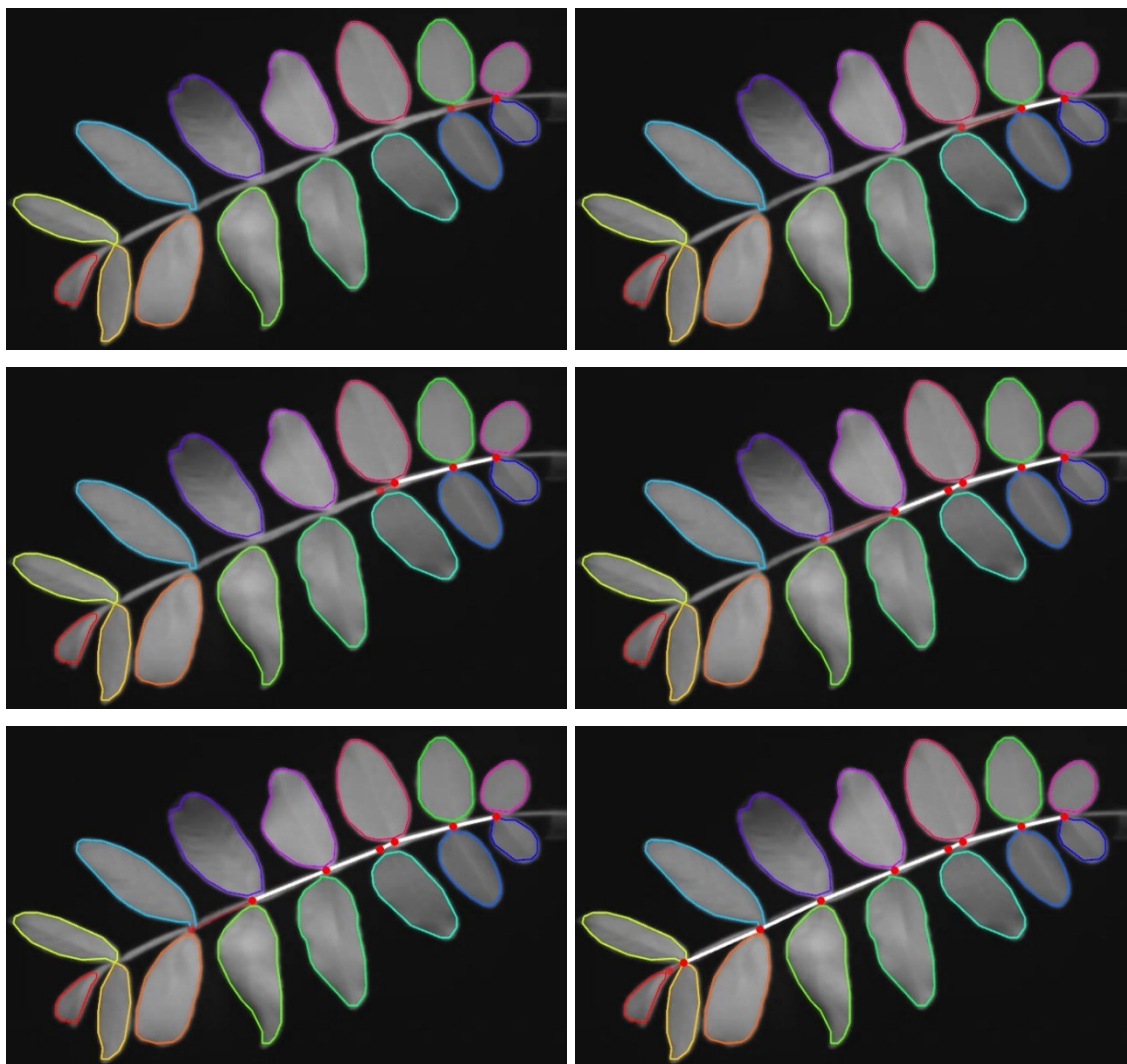


図 57 マウスクリック指定 (赤点) による線の描画 (白線)．

1) 葉脈をモチーフとしたエフェクトにおける始点指定 葉脈をモチーフとしたエフェクトを生成する．手法 1 と同様，Runions らの手法[54]を用いて葉脈を生成し，葉脈から計算した速度場上で粒子を動かす．葉脈の始点の指定に，枝状にマウスクリックで順番に指定した頂点 r_j の連結情報を用いる．輪郭の各頂点 p_i^0 と頂点 r_j の距離 d を計算し，距離 d が最小の時の頂点 p_i^0 を始点 p_{origin}^0 として採用し，葉脈を生成する（図 58）．



図 58 線入力で始点を指定することによって生成した葉脈．

2) ホタルをモチーフとしたエフェクトにおける光る順番の指定 手法 3 で提案したホタルをモチーフとしたエフェクトを作成する．手法 3 と同様，ホタルの発光リズムに見られる $1/f$ ゆらぎを用いて投影領域全体が点滅するエフェクトを作成する．また，光が枝に沿って伝搬していくようなエフェクトを作成するため，ユーザーが描画した線を用いる．まず，葉脈をモチーフとしたエフェクトと同様に，輪郭の各頂点 p_i^t と指定した頂点 r_j の距離 d を計算し，距離 d が最小の時の頂点 r_l （ l ：距離 d が最小の時の頂点番号）を求める．描画した線の始点 r_0 から求めた点 r_l までの長さを用いて，エフェクトの光の大きさを変えることで，投影領域が枝に沿って順番に光るようなエフェクトを生成する（図 59）．

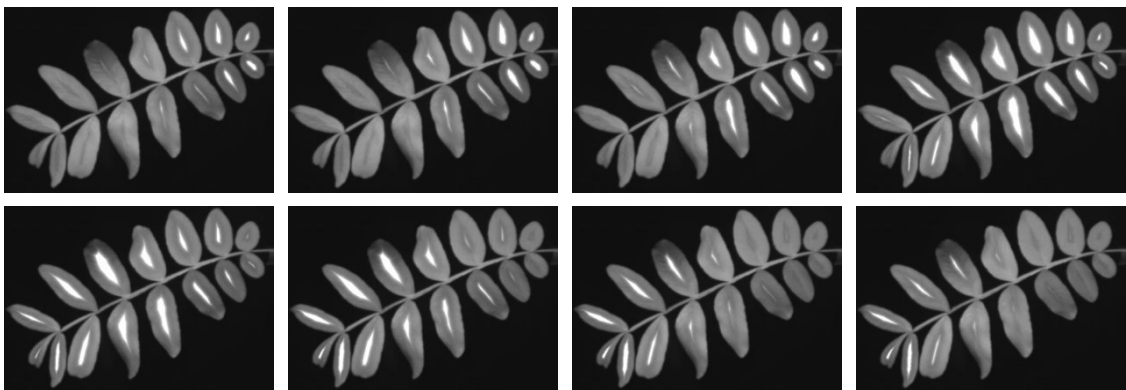


図 59 線入力を適用したホタルモチーフのエフェクト．

7.2 結果考察

本研究によって作成した追跡結果と投影結果を図 60, 図 66 に示す. 本手法で示す結果の作成には, Windows PC(Intel Core i9-9900(3.1GHz) PC)を使用した. カメラは解像度 $2048 \times 1088\text{px}$ でフレームレート 169fps の STC-MBCM200U3V-NIR を, プロジェクターは解像度 $1280 \times 720\text{px}$ の Taxan KG-PL081W, 赤外線照明は S8100-30-B/C-IR を用いた.

まず, 投影領域の選出では, UI 上のマウสดラッグで投影領域の削除と分割を行うシステムによって, 多数の葉が含まれるシーンにおいて柔軟に領域を選出することができた.

つづいて, 本追跡手法では, 平行移動, z 軸回転, x 軸回転する楕円形の葉を追跡することができた (図 60).

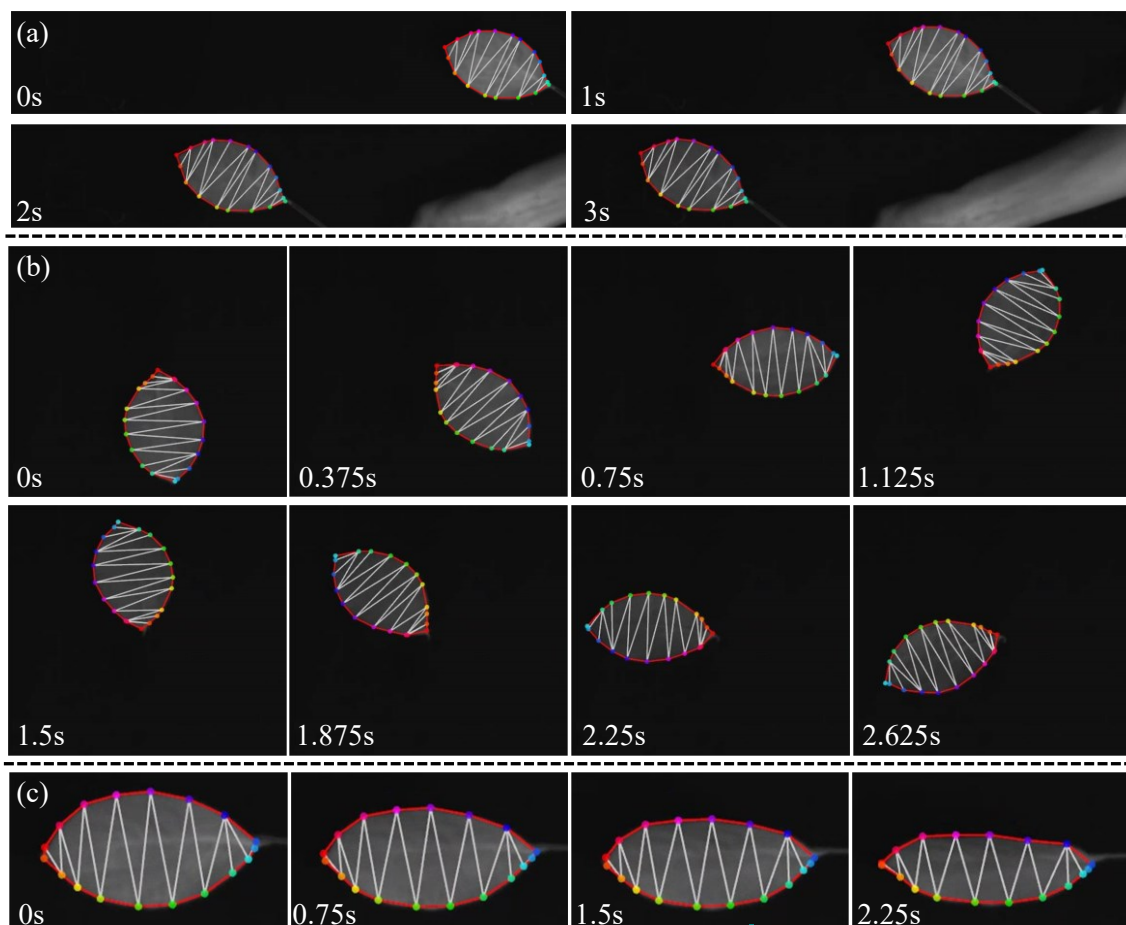


図 60 動く楕円形の葉に対する追跡結果 (a)平行移動させた葉, (b)カメラ方向の軸を中心に約 360 度回転させた葉, (c)葉の長軸方向を軸に回転させた葉.

一方、楕円近似を用いていることから、イチョウ等の楕円に近い形状は追跡できたが（図 61）、楕円形とは大きく異なる形状であるハート形の葉を追跡することはできなかった（図 62）。

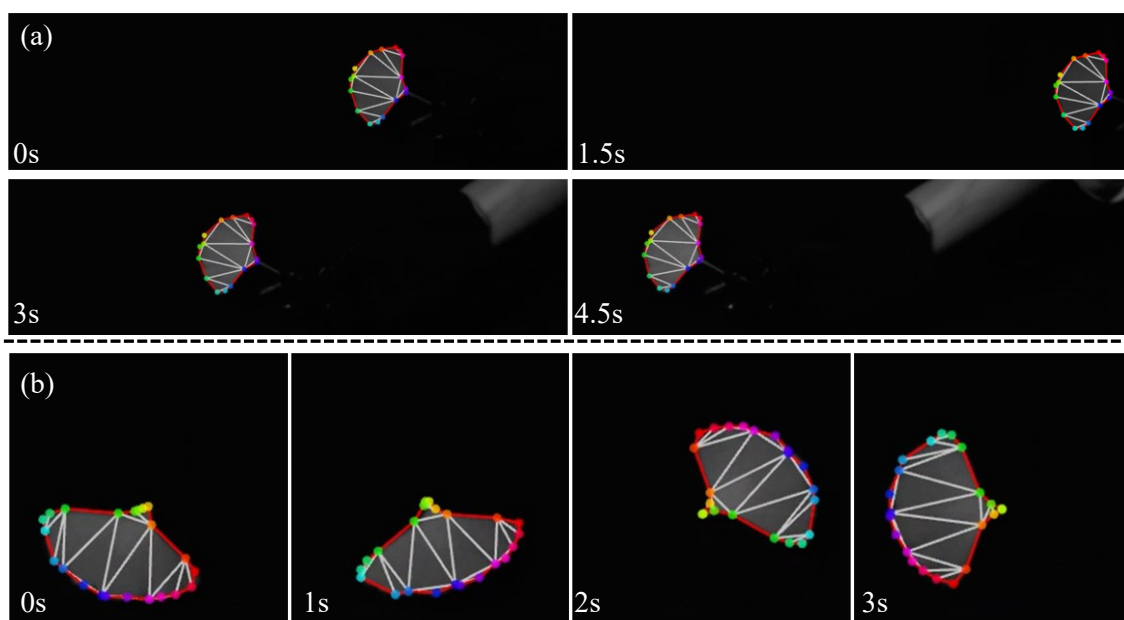


図 61 動く葉に対する追跡結果 (a)平行移動させたイチョウの葉に対する追跡結果, (b)カメラ方向の軸を中心に約 360 度回転させたイチョウの葉に対する追跡結果.

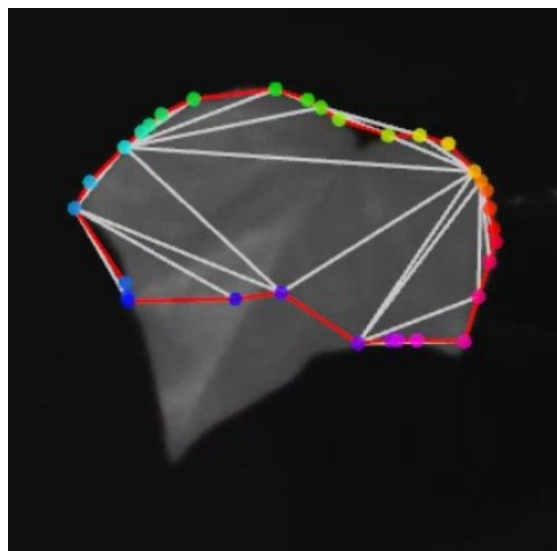


図 62 動くハート形の葉に対する追跡結果

また、投影領域同士の重なりを考慮し、楕円の衝突判定によって衝突した箇所を計算した結果、図 63 赤点のように、正しく判定することができた。そのため、複数の葉が重なった時も 5 割程度の頂点であれば重なっていても安定して追跡することができた。一方で、追跡対象でない葉や物体と重なる時の衝突を判定していないため、追跡対象以外の物体と重なった箇所を正しく追跡できない。

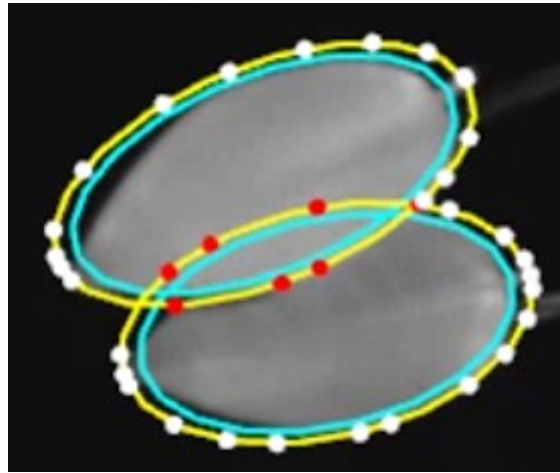


図 63 衝突判定の結果。

本手法では葉と葉がぶつかった際に、衝突した点の位置を **Rigid** 変形で補間して用いた(7.1.2 冒頭)。しかし、単純に楕円近似の対象から衝突した点を除く方法もある。そこで比較のため、衝突した点を補間せずに楕円近似の対象から除いた実験を行ったところ、衝突していない点(図 64 白点)に沿った弧を持つ楕円を検出し、実際の投影領域とは大きくずれる結果となり、追跡できなかった(図 65)。葉と葉が衝突した際、衝突した点を除いて楕円検出を行うと、衝突している隣の葉の輪郭上に頂点を求めてしまい、実際の葉と大きくずれる。このとき、**Rigid** 変形で補間した点(図 65 赤点)を用いれば、重なった葉との間に楕円近似する点ができやすいため、重なった葉の外側の輪郭線に頂点を求めることを回避しやすい。しかし、**Rigid** 変形で補間した点も必ずしも正しくないため、より複雑なシーンでは追跡できない可能性がある。

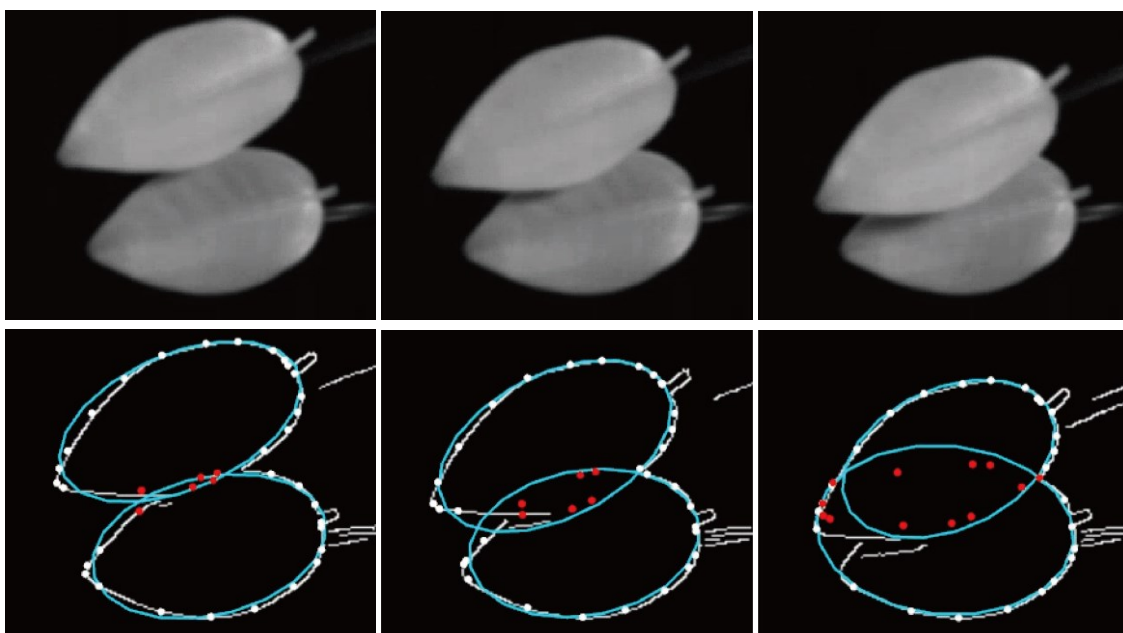


図 65 入力画像（上段）と衝突した点を除いて楕円近似を行った結果（下段）．白点は楕円の法線と画像のエッジ（白線）の交点，赤点は衝突した点を Rigid 変形によって補間した結果．左から時系列順．

速度においては，1枚の葉に対して，1フレームあたり 2.02ms（494fps）で追跡することができた．さらに，90枚の葉まで，1フレーム 15.7ms（63fps）で追跡できた．

エフェクトにおいては、マウスドラッグ等のユーザー入力を適用し、入力に応じたアニメーションを生成することができた (図 66). また、屋外の葉や大量の葉、水上の葉に対して投影することができた (図 67).

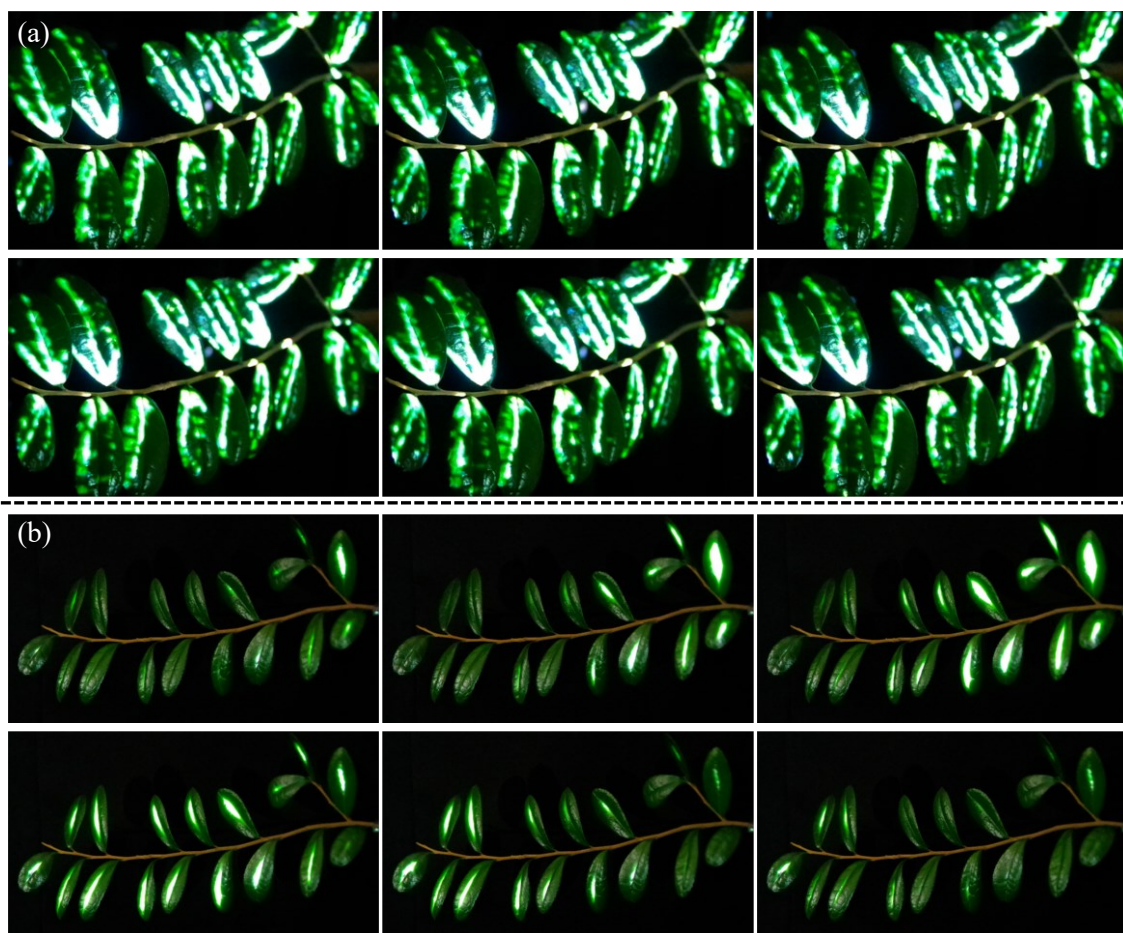


図 66 エフェクトの投影結果 (a)葉脈モチーフのエフェクトの投影結果, (b)ホタルモチーフのエフェクトの投影結果.

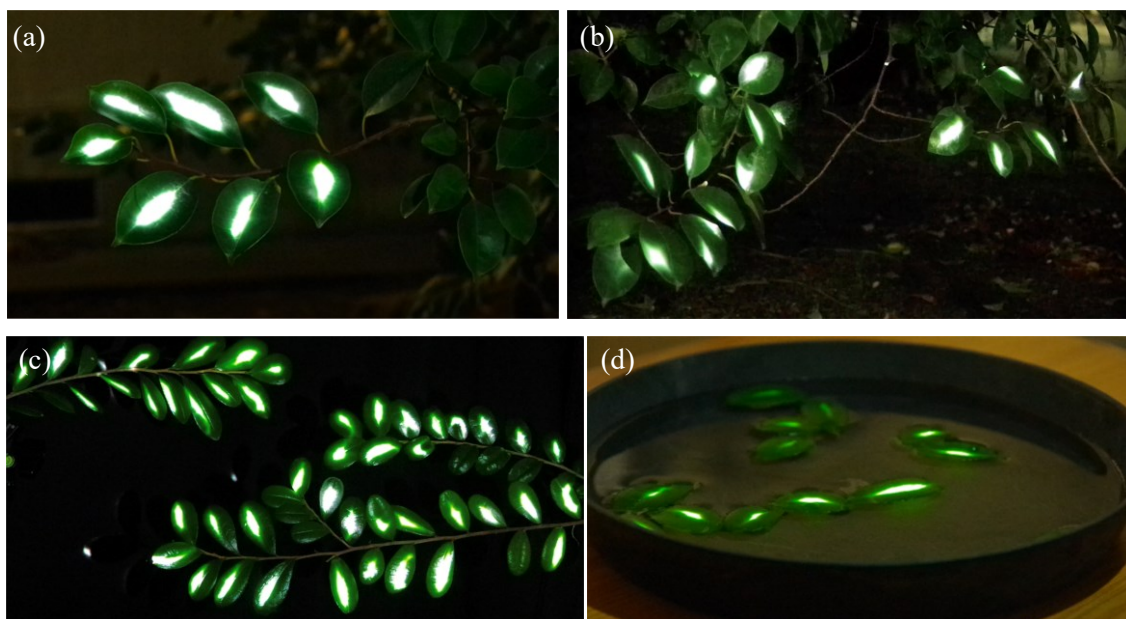


図 67 投影結果 (a)(b)屋外の葉への投影結果, (c)約 45 枚の葉への投影, (d)水上の葉への投影結果.

第8章

本手法で提案する 4 手法の比較と考察

本章では、第4章から第7章まで述べた4つの各DPM生成手法について、特に追跡手法と作成したエフェクトについて、比較し考察する(表2)。また、各手法で使用した機材を表3に示す。

表2 各DPM生成手法の比較

項目/手法	手法1 (第4章) 楕円形の葉への DPM	手法2 (第5章) 接触を考慮した 様々な形状の植 物へのDPM	手法3 (第6章) 様々な形状の植 物へのインタラ クティブなDPM	手法4 (第7章) 大量の楕円形の 葉へのDPM
参考論文	芸術科学会論文 誌第20巻[50] SIGGRAPH ASIA 2018 Poster[51]		NICOGRAPH International 2021 Full Paper[57] SIGGRAPH ASIA 2021 Posters[59]	SIGGRAPH Art Papersに投稿予 定.
参考作品	NICOGRAPH201 8[52] アジアデジタル アート大賞展 2018[53]	UIST 2019 Demo[55]	SIGGRAPH 2021 Labs[58] Laval Virtual 2021[60]	
追跡手法	投影領域の最も 遠い2点を葉の 端点として追 跡.	葉全形と葉の輪 郭の各頂点の2 つのスケールの テンプレートマ ッチングで追跡 後、頂点位置を 初期形状に基づ き補正.	オプティカルフ ローベースの方 法で輪郭の各頂 点を追跡. 遮蔽 点は他の頂点と の位置関係から 再計算.	エッジ画像上で 楕円近似. 投影 対象同士が重な る箇所は、他の 頂点との位置関 係より計算.
追跡可能な形状	△	○	○	△
・楕円形の葉	○	○	○	○
・ハート形の葉	○	○	○	×
・カエデ形の葉	×	○	○	×
・イチョウ形の 葉	×	○	○	△
・花	×	○	○	×
追跡可能な動き	△	○	○	△
・平行移動	○	○	○	○
・z軸回転	○	○	○	○
・x軸回転	○	○	○	○
・局所変形	△	○	○	×
追跡速度(ms)	4.97	6.22	4.57	2.02

インタラクショ ンに対する頑健 性	× 接触/遮蔽が発生 すると追跡でき ない.	△ 接触に対応. 遮蔽には限定的 に対応.	◎ 接 触 / 遮 蔽 に 対 応.	○ 接触に対応. 対象同士の遮蔽 のみ対応.
エフェクト生成 に用いる入力情 報	輪郭 葉脈エフェクト の始点の位置	輪郭 葉脈エフェクト の始点の位置	輪郭 遮蔽箇所 接触判定 (一部, 始点の 位置)	輪郭 対象が光る順番 葉脈エフェクト の始点の位置
エフェクトの モチーフ	ホタル 葉脈 オワンクラゲ	ホタル 葉脈 オワンクラゲ	ホタル 櫛クラゲ 細胞 水晶	ホタル 葉脈

表 3 各 DPM 生成手法における使用機材の比較

項目/手法	手法1	手法2	手法3	手法4
追跡結果/投影 結果作成時の 使用PC	Windows PC Intel Core i9-9900(3.1GHz)			
エフェクト投影 /作品制作時の 使用PC	Macbook Pro Intel Core i7 (3.2 GHz)	Windows PC Intel Core i7- 7700HQ (2.8GHz)	Windows PC Intel Core i9-9900(3.1GHz)	
プロジェクター	Taxan KG-PL081W 解像度 1280×720 px 60fps			
追跡結果/投影 結果作成時の カメラ	STC-MBCM200U3V-NIR 2048×1088 px 169fps			
エフェクト投影 /作品制作時の カメラ	BUFFALO BSW200MBK 1920×1080 px 30fps	STC-MBCM200U3V-NIR 2048×1088 px 169fps		
赤外線照明	S8100-IR			S8100-30-B/C-IR
その他			Arduino Uno 1MΩの抵抗	

8.1 追跡手法の比較と考察

4 手法の特に追跡手法において、投影対象の様々な形状や動きへの汎用性、処理速度、ユーザーによる遮蔽やインタラクションへの対応の点で、比較・考察する。

まず、各手法で追跡可能な形状を比較する。主に楕円形、ハート形、イチヨウ形、カエデ形の葉（図 68）を例に比較する。手法 2 と手法 3 では、イチヨウ形やカエデ形など様々な形状の葉に加え、花も追跡し投影できた。一方、手法 1 と手法 4 は、特定の形状の投影対象のみ追跡することができた。手法 1 では、葉の端点を追跡しているため、楕円形やハート形等の、動きや変形によって端点が変わらない形状は、安定して追跡することができた。一方、カエデのような複雑な形状は端点の同定ができず、追跡が困難であった。手法 4 は、投影対象の形状を楕円形に限定し、エッジ画像上で楕円近似を行っているため、楕円形の葉のみ追跡できた。

また、すべての手法において、正円に近い形状の植物は、対象の向きを求めるための特徴が少なく、追跡が困難であった。

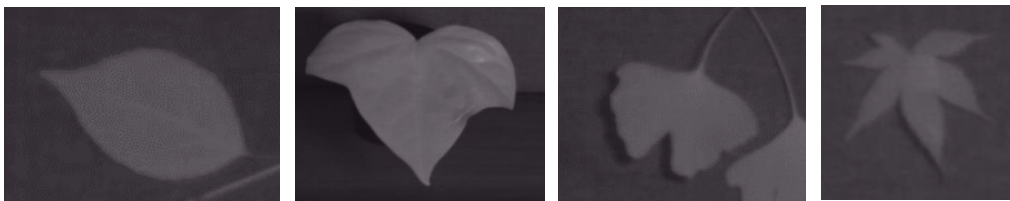


図 68 追跡で用いた形状の葉。

次に、投影対象の様々な動きに対する追跡について比較する。投影対象の平行移動、z 軸回転、x 軸回転、局所変形に対して追跡実験を行った。まず、すべての手法において、各手法対応形状の平行移動、z 軸回転、x 軸回転を追跡できた。一方、局所変形は、手法 2 と手法 3 のみ追跡できた。手法 1 は、投影対象の二つの端点の位置が変わるような局所変形を追跡できなかった。手法 4 は、近似楕円の位置と角度、大きさのみを追跡しているため、局所変形に対応できなかった。

つづいて、各手法の計算速度を比較する。輪郭の頂点を 11 個も一つの投影対象で実験を行った。手法 4 において、最も計算時間が短く、90 枚の対象まで 60fps で追跡できた。

最後に、投影対象への接触と遮蔽などのインタラクションについて、比較する。まず手法 1 では、画像処理で得た輪郭を用いて追跡しているため、画像上の輪郭領域が大きく変化する、接触や遮蔽には対応できなかった。手法 2 では、全体の形状を維持

しながら追跡するため、接触を伴うインタラクションに対応できた。遮蔽には、対象が初期位置にある時に限定して対応できた。手法 3 では、輪郭の各頂点について遮蔽判定を行い、位置の再計算を行っているため、接触・遮蔽に対応できた。手法 4 では、エッジ画像から得られた複数の点から楕円近似をおこなっているため、接触に対しては安定して追跡できた。遮蔽においては、投影対象同士の遮蔽可能性のみ考慮しているため、人の手など、投影対象以外の遮蔽物には対応できなかった。

8.2 エフェクトアニメーションの比較と考察

各手法で作成したエフェクトについて比較し，考察する．すべての表現において，生物発光や自然界における光をモチーフに，投影対象自身が発光しているように感じられる表現を目指し，対象物体の輪郭形状を用いて自動で生成した．

手法1，手法2で適用した，ホタル，葉脈，オワンクラゲをモチーフとした3種類のエフェクトは，作品展示[52][53]において投影し，体験者からは「生命感を感じる」等の意見を得ることができた．また，葉脈モチーフのエフェクトのみ始点の指定をユーザーが行う必要がある．

手法3では，遮蔽判定と接触判定を用いることで，インタラクティブに映像が変化するエフェクトを作成した．エフェクトのモチーフは，ホタル，櫛クラゲ，細胞，水晶であり，各エフェクトは接触箇所から色や形の変化，接触時からの時間を用いた光の強さの変化を作成することができた．また，クシクラゲモチーフのエフェクトのみ始点をユーザーが指定する．

手法4においてユーザーは，UI上における線の描画によって，多数の投影対象のエフェクトの動きや向きを，より直観的かつ一括で指定することができた．ホタルと葉脈をモチーフとしたエフェクトを適用した．ホタルモチーフのエフェクトでは，枝に接している葉の順番を光の強さに適用するようなエフェクトを作成した．葉脈をモチーフとしたエフェクトでは，描画した線との距離を用いて，枝に接する箇所から葉脈が始まるようなエフェクトを生成した．

8.3 将来課題

本手法における課題についてまとめる。

まず、追跡における課題として、様々な形状をもつ葉の多数追跡がある。解決方法としては、手法 4 のように楕円近似により投影領域の大まかな位置を求めた上で、手法 1~3 のように形状の特徴情報を用いて、詳細な形状を追跡することなどが考えられる。

また、現在は初期フレームにおいて、ユーザー入力によって投影領域の選出を行う必要があるが、領域選出を行っている間に投影対象が動くとき設定が難しい。そのため、全て自動で投影領域を選出する手法や、追跡と同時に領域を選出できるシステムの開発が求められる。

本研究の手法 3 や手法 4 では、投影対象への遮蔽に対応したが、およそ 5 割程度以上の頂点が遮蔽された場合、安定して対応できないため、今後は完全に葉が遮蔽された時でも追跡を継続する手法が求められる。また、遮蔽時の投影映像においては、葉同士が重なったとき、片方の葉への投影映像は片方の葉によってマスクされるべきだが、これに関して本稿では扱っていない。これに対応するためには、マスク領域の抽出に加え、葉の上下関係を求める必要がある。

エフェクト作成における課題として、画像上の枝や葉脈の領域抽出が挙げられる。このような情報を得ることができれば、より対象の形状に沿ったエフェクトの作成が可能である。本研究では、枝や葉脈の大きさや反射率、照明の方向や強さの違いによって、枝や葉脈等の詳細情報を赤外線画像から安定して取得することは困難であったため、ユーザー入力や自動生成モデルによって疑似的に作成した。しかし、RGB カメラ等と組み合わせることによって、これらのような詳細情報を取得できる可能性がある。

また、手法 4 では、オワンクラゲモチーフ等の他のエフェクトを適用することができなかった。今後、様々なエフェクトを汎用的に適用できるシステムの開発が求められる。

第9章

結論

本研究では、葉や花への DPM の自動生成システムを提案した。具体的には、様々なシーンにおける DPM 自動生成を実現するため、動く植物への投影における課題を段階的に設け、4 つの手法を提案した。また、4 つの手法について、実験を行い、比較、検証した。各手法の概要と成果を以下にまとめる。

手法 1：楕円形の葉への DPM 投影対象の葉が楕円形であるという仮定で、投影領域の輪郭の最も遠い 2 点を追跡する手法を提案した。また、プロジェクターの投影領域とカメラの取得領域の位置合わせ方法として、前もって投影領域をカメラで取得し記録しておくことで、投影時の労力を大幅に軽減した。さらに、投影領域の形状から自動でエフェクトを生成するシステムも実装し、エフェクトを投影することで、楕円形の葉への DPM 作品を制作した。

手法 2：接触を考慮した様々な形状の植物への DPM 楕円形の葉に限らず、様々な形状の葉や花を投影対象とした。さらに、手法 1 では対応できなかった、体験者の手による接触を考慮することを目的とし、投影領域毎の全形を維持しながら、局所領域である輪郭の各頂点を追跡する手法を提案した。本手法によって、イチョウやカエデ、花など様々な植物への DPM を達成した。また、対象への素手によるインタラクションも可能である。

手法 3：様々な形状の植物へのインタラクティブな DPM 体験者のインタラクション時に生じる、遮蔽に頑健な DPM 生成手法を提案し、インタラクティブな DPM 生成システムを実装した。投影領域の輪郭の各頂点を追跡すると同時に遮蔽箇所を判定し、遮蔽領域の追跡結果は疑似的に求めた。さらに、体験者の植物への接触を判定するタッチセンサと組み合わせることで、インタラクティブに映像が変化するシステムを実装した。本システムによって、より様々な形状をもつ植物へのインタラクティブな DPM を安定して生成することができた。

手法 4：大量の楕円形の葉への DPM 大量の植物への DPM を実現するため、投影対象を楕円形の葉に限定し、多数の葉の投影領域を選出し、高速に追跡する手法を提案した。毎フレーム楕円近似を行うことで、投影領域の近似楕円の位置、角度、大きさのみを高速に求める。手法 1 から手法 3 では、60fps で追跡できる対象は 10 枚程度のみであったが、本手法では、80 枚程度の対象領域を 60fps 以上で追跡することができた。

本研究では、事前の形状入力なし、かつ簡易な機材によって、葉や花への DPM 生成手法を 4 つ提案した。各手法において、複雑な形状持つ植物、インタラクションに伴

って遮蔽される植物，大量の葉等の，様々なシーンにおいて生じる課題を解決し，葉や花への動的プロジェクションマッピングを実現することができた．今後の課題として，インタラクションによって動き変形する様々な形状の植物を大量に追跡することが挙げられる．

また，本研究を応用することで，イルミネーション等のエンターテイメントへの制作のほか，植物の仕組みを教えるような教育目的のコンテンツ，ユーザーを癒すようなインテリアなど，様々な実装例が期待できる．

謝辞

本論文の執筆にあたり，ご指導，ご協力頂きました芸術工学研究院メディアデザイン部門の鶴野玲治教授に深く感謝いたします。

同部門の原健二教授，並びに石井達郎准教授には，本論文をまとめるにあたり，貴重なご助言をいただきました。深く感謝申し上げます。

同部門の森本有紀助教授には，本研究及び本論文の執筆を進めるにあたり，6年間，終始ご指導いただきました。深く感謝いたします。

研究室の皆さんには，本研究に対する意見や助言，相談など，協力して頂いたことに謹んで感謝の意を表します。

本研究が JSPS 科研費 18K11956 の助成を受けたこと，感謝の意を示します。また，本研究が公益財団法人中山隼雄科学技術文化財団の援助を受けたこと，感謝の意を示します。

最後に，常に筆者を支えてくださった家族に感謝いたします。

参考文献

- [1] 株式会社ネイキッド. 2012. TOKYO HIKARI VISION. <https://naked.co.jp/works/852/>. (Retrieved 16 Dec 2022).
- [2] チームラボ株式会社. 2017. かみさまの御前なる岩に憑依する滝 / Universe of Water Particles on a Sacred Rock. <https://www.teamlab.art/jp/w/sacred-rock/>. (Retrieved 16 Dec 2022).
- [3] Friedrich van Schoor, and Tarek Mawad. 2015. bioluminescent forest. Cosmic Cine Festival Nominee. <https://glowingatoms.com/bioluminescent-forest>. (Retrieved 22 Dec 2022).
- [4] Nobumichi Asai, Jin Hasegawa, and Takashi Ishibashi. 2014. OMOTE / REAL-TIME FACE TRACKING & PROJECTION MAPPING. <https://www.nobumichiasai.com/works/131/>. (Retrieved 16 Dec 2022).
- [5] 文田昌宏, 山田浩徳, 福本和睦, 川田直樹. 2011. TOYOTA プリウス α CM. <https://www.symunity.co.jp/beampaint/report/report-priuscm.html>. (Retrieved 16 Dec 2022).
- [6] SmartPixels. 2016. NIKEiD AUGMENTED. <https://www.youtube.com/watch?v=Xf3VdMENFKs>. (Retrieved 16 Dec 2022).
- [7] 迎山 和司, 小林 真幸. 2015.12. 国宝「中空土偶」へのインタラクティブ・プロジェクションマッピングを使った新しい展示方法の検討, 芸術科学会論文誌 第14巻 第6号 pp.248-256.
- [8] 三鷹光器株式会社, 京都大学医学部附属病院, パナソニック株式会社. 2019. Medical Imaging Projection System : MIPS. <https://www.innervision.co.jp/sp/products/release/20200313>. (Retrieved 16 Dec 2022).
- [9] Goshiro Yamamoto, and Kosuke Sato. 2007. PALMbit: A Body Interface Utilizing Light Projection onto Palms. The Journal of The Institute of Image Information and Television Engineers. 61. 797-804. DOI: <https://doi.org/10.3169/itej.61.797>.
- [10] Brett R. Jones, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. 2013. IllumiRoom: peripheral projected illusions for interactive experiences. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). Association for Computing Machinery, New York, NY, USA, 869–878. DOI: <https://doi.org/10.1145/2470654.2466112>.
- [11] Matthew Flagg and James M. Rehg. 2006. Projector-guided painting. In Proceedings of the 19th annual ACM symposium on User interface software and technology (UIST '06). Association for Computing Machinery, New York, NY, USA, 235–244. DOI: <https://doi.org/10.1145/1166253.1166290>.
- [12] Alec Rivers, Andrew Adams, and Frédo Durand. 2012. Sculpting by numbers. ACM Trans. Graph. 31, 6, Article 157, 7 pages. DOI: <https://doi.org/10.1145/2366145.2366176>.
- [13] Bot&Dolly. 2013. Box. <https://youtu.be/IX6JcybgDFo>. (Retrieved 16 Dec 2022).

- [14] Anselm Grundhöfer, and Daisuke Iwai. 2018. Recent Advances in Projection Mapping Algorithms, Hardware and Applications. *Computer Graphics Forum*. 37. 653–675. DOI: <https://doi.org/10.1111/cgf.13387>.
- [15] Yuqi Li, Aditi Majumder, Dongming Lu, and M. Gopi. 2015. Content-Independent Multi-Spectral Display Using Superimposed Projections. *Comput. Graph. Forum* 34, 2, 337–348. DOI: <https://doi.org/10.1111/cgf.12564>.
- [16] Daniel G. Aliaga, Yu Hong Yeung, Alvin Law, Behzad Sajadi, and Aditi Majumder. 2012. Fast high-resolution appearance editing using superimposed projections. *ACM Trans. Graph.* 31, 2, Article 13, 13 pages. DOI: <https://doi.org/10.1145/2159516.2159518>.
- [17] Toshiyuki Amano and Kensuke Minami. 2015. Structural color display on retro-reflective objects. In *Proceedings of the 25th International Conference on Artificial Reality and Telexistence and 20th Eurographics Symposium on Virtual Environments (ICAT - EGVE '15)*. Eurographics Association, Goslar, DEU, 37–44.
- [18] Shuntaro Yamazaki, Masaaki Mochimaru, and Takeo Kanade, Simultaneous self-calibration of a projector and a camera using structured light. 2011. *CVPR 2011 WORKSHOPS*, pp. 60–67, DOI: <https://doi.org/10.1109/CVPRW.2011.5981781>.
- [19] Toshiyuki Amano. 2014. Projection Center Calibration for a Co-located Projector Camera System. 2014 *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 449–454, DOI: <https://doi.org/10.1109/CVPRW.2014.72>.
- [20] Ramesh Raskar, Jeroen van Baar, Paul Beardsley, Thomas Willwacher, Srinivas Rao, and Clifton Forlines. 2003. ILamps: geometrically aware and self-configuring projectors. *ACM Trans. Graph.* 22, 3, 809–818. DOI: <https://doi.org/10.1145/882262.882349>
- [21] Simon Willi, and Anselm Grundhöfer. 2017. Robust Geometric Self-Calibration of Generic Multi-Projector Camera Systems. 2017 *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 42–51, DOI: <https://doi.org/10.1109/ISMAR.2017.21>.
- [22] garageCube. 2021. MadMapper 5. <https://www.garagecube.com>. (Retrieved 16 Dec 2022).
- [23] Resolume B.V. 2022. Resolume Arena. <https://resolume.com>. (Retrieved 16 Dec 2022).
- [24] 株式会社アイデアクラウド, 2017, LAMP. <https://ideacloud.co.jp/works/5783/>. (Retrieved 16 Dec 2022).
- [25] Lightform, inc. 2018. LIGHTFORM. <https://lightform.com/>. (Retrieved 16 Dec 2022).
- [26] Christoph Resch, Peter Keitler, and Gudrun Klinker. 2016. Sticky Projections-A Model-Based Approach to Interactive Shader Lamps Tracking. in *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 3, pp. 1291–1301, 1 March 2016, DOI: <https://doi.org/10.1109/TVCG.2015.2450934>.
- [27] Yuki Morikubo, and Naoki Hashimoto. 2018. Edge-based object tracking for dynamic

- projection mapping. 2018 International Workshop on Advanced Image Technology (IWAIT), pp. 1-4, DOI: <https://doi.org/10.1109/IWAIT.2018.8369709>.
- [28] Yi Zhou, Shuangjiu Xiao, Ning Tang, Zhiyong Wei, and Xu Chen. 2016. Pmomo: Projection Mapping on Movable 3D Object. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16). Association for Computing Machinery, New York, NY, USA, 781–790. DOI: <https://doi.org/10.1145/2858036.2858329>
- [29] Hirotaka Asayama, Daisuke Iwai, and Kosuke Sato. 2018. Fabricating Diminishable Visual Markers for Geometric Registration in Projection Mapping. in IEEE Transactions on Visualization and Computer Graphics, vol. 24, no. 2, pp. 1091-1102, 1 Feb. 2018, DOI: <https://doi.org/10.1109/TVCG.2017.2657634>.
- [30] Daiki Tone, Daisuke Iwai, Shinsaku Hiura, and Kosuke Sato. 2020. FibAR: Embedding Optical Fibers in 3D Printed Objects for Active Markers in Dynamic Projection Mapping. IEEE Transactions on Visualization and Computer Graphics. PP. 1-1. DOI: <https://doi.org/10.1109/TVCG.2020.2973444>.
- [31] Amit H. Bermano, Markus Billeter, Daisuke Iwai, and Anselm Grundhöfer. 2017. Makeup Lamps: Live Augmentation of Human Faces via Projection, 2017. Eurographics. DOI: <https://doi.org/10.1111/cgf.13128>.
- [32] Gaku Narita, Yoshihiro Watanabe, and Masatoshi Ishikawa. 2015. Dynamic projection mapping onto a deformable object with occlusion based on high-speed tracking of dot marker array. In Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology (VRST '15), Stephen N. Spencer (Ed.). ACM, 149-152. DOI: <https://doi.org/10.1145/2821592.2821618>.
- [33] Gaku Narita, Yoshihiro Watanabe, and Masatoshi Ishikawa, Dynamic Projection Mapping onto Deforming Non-Rigid Surface Using Deformable Dot Cluster Marker. 2017. in IEEE Transactions on Visualization and Computer Graphics, vol. 23, no. 3, pp. 1235-1248. DOI: <https://doi.org/10.1109/TVCG.2016.2592910>.
- [34] Leo Miyashita, Yoshihiro Watanabe, and Masatoshi Ishikawa. 2018. MIDAS projection: markerless and modelless dynamic projection mapping for material representation. ACM Trans. Graph. 37, 6, Article 196, 12 pages. DOI: <https://doi.org/10.1145/3272127.3275045>.
- [35] Albert Ng, Julian Lepinski, Daniel Wigdor, Steven Sanders, and Paul Dietz. 2012. Designing for low-latency direct-touch input. In Proceedings of the 25th annual ACM symposium on User interface software and technology (UIST '12). Association for Computing Machinery, New York, NY, USA, 453–464. DOI: <https://doi.org/10.1145/2380116.2380174>.
- [36] Kosuke Hiratani, Daisuke Iwai, Parinya Punpongsanon and Kosuke Sato. 2019. Shadowless Projector: Suppressing Shadows in Projection Mapping with Micro Mirror Array Plate. 2019

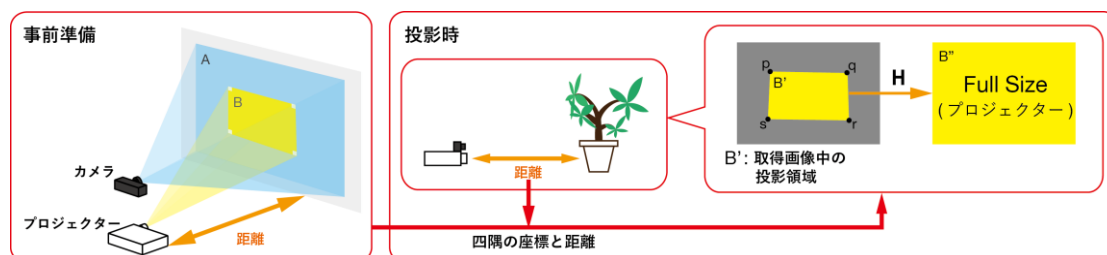
- IEEE Conference on Virtual Reality and 3D User Interfaces (VR), Osaka, Japan, pp. 1309–1310. DOI: <https://doi.org/10.1109/VR.2019.8798245>.
- [37] Jaedong Kim, Hyunggoog Seo, Seunghoon Cha, and Junyong Noh. 2018. Real-Time Human Shadow Removal in a Front Projection System. *Computer Graphics Forum*. 38. DOI: <https://doi.org/10.1111/cgf.13541>.
- [38] Shingo Kagami and Koichi Hashimoto. 2020. Interactive Stickies: Low-latency projection mapping for dynamic interaction with projected images on a movable surface. In *ACM SIGGRAPH 2020 Emerging Technologies (SIGGRAPH '20)*. Association for Computing Machinery, New York, NY, USA, Article 16, 1–2. DOI: <https://doi.org/10.1145/3388534.3407291>.
- [39] teamLab. 2015. 花と魚 / Flowers and Fish - Enoshima Aquarium Big Sagami Bay Tank. <https://www.teamlab.art/jp/w/flowersandfish/>. (Retrieved 16 Dec 2022).
- [40] Yasushi Matoba, Yoshiharu Ooide, Shingo Yamano, Shumpei Yamakawa, Taro Tokui, and Hideki Koike. 2013. ChatFish: Projection on the Surface of Fish. *Laval Virtual 2013*.
- [41] David G. Lowe. 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60, 91–110. DOI: <https://doi.org/10.1023/B:VISI.0000029664.99615.94>
- [42] Michael Calonder, Vincent Lepetit, Christoph Strecha, Pascal Fua. 2010. BRIEF: Binary Robust Independent Elementary Features. *European Conference on Computer Vision*. 6314–778–792. DOI: https://doi.org/10.1007/978-3-642-15561-1_56.
- [43] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. 2011. ORB: An efficient alternative to SIFT or SURF. *2011 International Conference on Computer Vision*, pp. 2564–2571, DOI: <https://doi.org/10.1109/ICCV.2011.6126544>.
- [44] Bruce Lucas, and Takeo Kanade. 1981. An Iterative Image Registration Technique with an Application to Stereo Vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 674–679.
- [45] Christa Sommerer, and Laurent Mignonneau. 1993. Interactive Plant Growing. *SIGGRAPH Machine Culture*. <http://www.medienkunstnetz.de/works/the-interactive-plant-growing/video/1/>. (Retrieved 16 Dec 2022).
- [46] Ivan Poupyrev, Philipp Schoessler, Jonas Loh, and Munehiko Sato. 2012. Botanicus Interacticus: interactive plants technology. In *ACM SIGGRAPH 2012 Emerging Technologies (SIGGRAPH '12)*. Association for Computing Machinery, New York, NY, USA, Article 4, 1. DOI: <https://doi.org/10.1145/2343456.2343460>.
- [47] Wataru Kurihara, Akito Nakano, and Hisakazu Hada. 2017. Botanical puppet: Computer controlled shameplant. *Proceedings – 2017 NICOGRAPH International, NICOInt 2017*, 68–

71. DOI: <https://doi.org/10.1109/NICOInt.2017.16>.
- [48] 中安翌, 富松潔. 2010. plant: Shape Memory Alloy Motion Display による葉群のざわめきの表現. 日本バーチャルリアリティ学会論文誌 (アート&エンターテイメント 2 特集号), Vol.15, No.3, pp289-296. DOI: https://doi.org/10.18974/tvrsj.15.3_289.
- [49] color+light. 2022. Floura. <https://fluoraplant.com>. (Retrieved 16 Dec 2022).
- [50] 末吉知樹, 森本有紀. 2021. 葉を対象とした動的プロジェクションマッピングの自動生成. 芸術科学会論文誌 第 20 巻 第 1 号 pp.21-29
- [51] Tomoki Sueyoshi and Yuki Morimoto. 2018. Automatic generation of interactive projection mapping for leaves. In SIGGRAPH Asia 2018 Posters (SA '18). Association for Computing Machinery, New York, NY, USA, Article 2, 1–2. DOI: <https://doi.org/10.1145/3283289.3283310>.
- [52] 末吉 知樹, 森本有紀. 2018. Projection for Leaves. NICOGRAPH 2018 展示発表 (Exhibition track). 福岡. 優秀展示作品賞.
- [53] 末吉知樹 (アドバイザー: 森本有紀). 2018. Projection Mapping for Leaves. 2018 アジアデジタルアート大賞展 FUKUOKA. 福岡. 学生カテゴリー/インタラクティブアート部門 優秀賞, 福岡市長賞. https://youtu.be/S_zL3dW8aSc. (Retrieved 16 Dec 2022).
- [54] Adam Runions, Martin Fuhrer, Brendan Lane, Pavol Federl, Anne-Gaëlle Rolland-Lagan, and Przemyslaw Prusinkiewicz. 2005. Modeling and visualization of leaf venation patterns. In ACM SIGGRAPH 2005 Papers, Markus Gross (Ed.). ACM, 702-711. DOI: <https://doi.org/10.1145/1073204.1073251>.
- [55] Tomoki Sueyoshi and Yuki Morimoto. 2019. Tangible Projection Mapping onto Deformable Moving Thin Plants via Markerless Tracking. In The Adjunct Publication of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19). Association for Computing Machinery, New York, NY, USA, 105–106. DOI: <https://doi.org/10.1145/3332167.3356887>.
- [56] Olga Sorkine-Hornung, Daniel Cohen-Or, Yaron Lipman, Marc Alexa, Christian Rössl, and Hans Peter Seidel. 2004. Laplacian surface editing. In Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing (SGP '04). Association for Computing Machinery, New York, NY, USA, 175–184. DOI: <https://doi.org/10.1145/1057432.1057456>.
- [57] Tomoki Sueyoshi, and Yuki Morimoto. 2021. Interactive Dynamic Projection Mapping onto Thin Plants with Bioluminescent Effect Animations. NICOGRAPH International 2021. DOI: <https://doi.org/10.1109/NICOINT52941.2021.00008>.
- [58] Tomoki Sueyoshi and Yuki Morimoto. 2021. Dynamic Projection Mapping for Thin Plants using a Robust Tracking Method against Occlusion. In ACM SIGGRAPH 2021 Labs

- (SIGGRAPH '21). Association for Computing Machinery, New York, NY, USA, Article 6, 1–2. DOI: <https://doi.org/10.1145/3450616.3464521>.
- [59] Tomoki Sueyoshi and Yuki Morimoto. 2021. Interactive DPM for Thin Plants with the Latency Measurement. In ACM SIGGRAPH 2021 Posters (SIGGRAPH '21). Association for Computing Machinery, New York, NY, USA, Article 6, 1–2. DOI: <https://doi.org/10.1145/3450618.3469151>
- [60] Morimoto lab team (Tomoki Sueyoshi, and Yuki Morimoto). 2021. Tangible Bioluminescent Plants, Laval Virtual Revolution Competitions #Students Demo. DOI: <https://youtu.be/uKSEwtbAFto>. (Retrieved 16 Dec 2022).
- [61] Scott Schaefer, Travis McPhail, and Joe Warren. 2006. Image deformation using moving least squares. In ACM SIGGRAPH 2006 Papers (SIGGRAPH '06). Association for Computing Machinery, New York, NY, USA, 533–540. DOI: <https://doi.org/10.1145/1179352.1141920>.
- [62] Andrew Fitzgibbon and Robert Fisher. 1970. A Buyer's Guide to Conic Fitting. In Proceedings of the 6th British conference on Machine vision (Vol. 2) (BMVC '95). BMVA Press, GBR, 513–522.

付録

以下では，事前準備において，カメラの取得画像における，投影領域の四隅の座標を距離毎に示す．



射影変換における四隅の座標 (p, q, r, s)

距離	p.x	p.y	q.x	q.y	r.x	r.y
250	520	45	1534	44	1554	618
240	523	48	1532	49	1552	618
230	523	50	1532	51	1551	618
220	523	50	1534	49	1551	621
210	522	52	1534	51	1552	623
200	523	54	1534	53	1551	625
190	523	58	1534	56	1551	628
180	523	60	1534	59	1551	631
170	523	63	1534	62	1552	634
160	522	65	1535	64	1552	637
150	522	68	1536	67	1552	641
140	521	72	1536	71	1553	645
130	521	76	1536	75	1553	651
120	521	81	1537	80	1554	656
110	520	87	1537	85	1554	664
100	520	95	1539	93	1554	671
90	520	102	1538	103	1555	684
80	521	116	1537	115	1551	691
70	520	129	1539	128	1553	707
60	519	146	1539	145	1554	727
50	519	170	1541	171	1555	755

A 前準備により取得したプロジェクターの投影領域の四隅の座標