

浮遊物体を用いたインタラクティブなシャドウアート生成システム

村上 葉^{1,a)} Edmond S. L. Ho^{2,b)} Hubert P. H. Shum^{2,c)} 土橋 宜典^{1,d)}

概要：シャドウアートは、物体によって投影された影が、人間の顔や文字などの形状を形成するアートである。近年、コンピュータグラフィックスの研究者により、ユーザが指定した形状の影を生成する物体の形状や配置をモデリングする方法が提案されている。本研究では、ドローンなどの浮遊物体トを使用してシャドウアートを作成する方法を開発している。本稿では、ユーザがこのようなシャドウアートを設計するのに役立つインタラクティブなシステムを提案する。このシステムを使用することで、ユーザは2D平面で影を操作して容易にシャドウアートを設計できる。システムは、ユーザの指定した影を生成するための浮遊物体の3D空間上での最適な位置を計算する。このシステムを使用して作成されたシャドウアートの例を用いて提案手法の有効性を示す。

キーワード：シャドウアート, インタラクティブシステム, 浮遊物体

Interactive System for Shadow Art Using Floating Objects

YO MURAKAMI^{1,a)} EDMOND S. L. HO^{2,b)} HUBERT P. H. SHUM^{2,c)} YOSHINORI DOBASHI^{1,d)}

Abstract: Shadow art is an artwork where shadows cast by objects form the desired pattern, such as human faces or letters. In recent years, computer graphics researchers have developed methods for modeling objects with different configurations that generate the user-specified shadow pattern. Our research goal is to develop a method for creating shadow arts using floating objects such as drones. We propose an interactive system that helps the user design such shadow art. Using our system, the user operates in a 2D plane to design the shadow art while the system finds out the optimal position in 3D for producing the required shadow. We demonstrate some shadow arts created by using our system.

Keywords: shadow art, interactive system, floating objects

1. まえがき

影は芸術家の創造性を表現する手段として長年使用されている。影を使ったアートには様々なものが存在する。古くは古代中国の影の人形劇、最近では Sue Webster 氏と

Tim Noble 氏による作品や、Kumi Yamashita 氏による作品が代表例である。これらのアート作品では、多数の物体を適切に配置し、ある方向から光を照射することで、テーマを持った影が形成される。これらの作品の魅力は、一見無造作に配置されているように見える物体から、意味を持った影が投影されることである。

コンピュータグラフィックス分野においては、シャドウアートを計算機によって作成するための手法が提案されている [1]。これらの方法により、影を投影する物体の最適な形状や配置方法が求められ、目的の形状の影が生成できる。これらの手法を用いることで、興味深い形状の影を投

¹ 北海道大学

Hokkaido University

² ノーザンブリア大学

Northumbria University

a) murakami@ime.ist.hokudai.ac.jp

b) e.ho@northumbria.ac.uk

c) hubert.shum@northumbria.ac.uk

d) doba@ime.ist.hokudai.ac.jp

影する物体を生成できる。

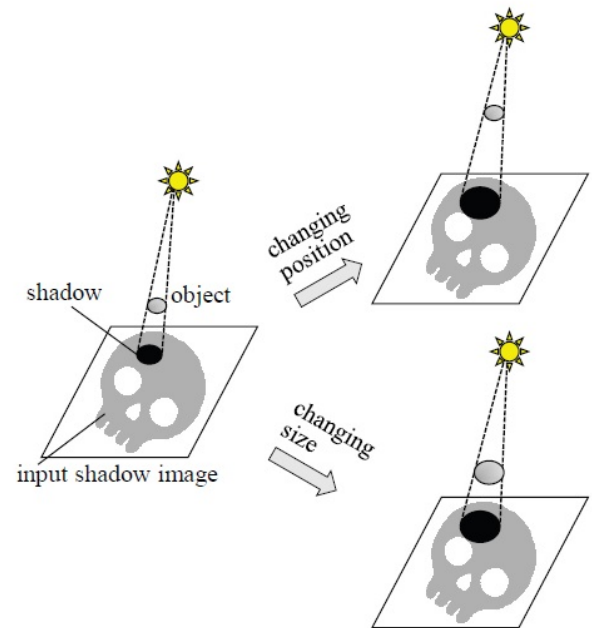
本稿では、浮遊する多数の物体を利用したシャドウアートをインタラクティブに生成する方法について提案する。インタラクティブな要素を取り入れることによって、ユーザは多数の物体の配置方法を自在に決定することができる。ユーザは物体の集合体が生成するシルエット画像などを指定する。これらの入力から、物体をどこに配置すればよい、または物体の大きさをどうすればよいかを提案システムが自動的に求める。本稿の最後に浮遊する球を使用した例をいくつか示すことにより、システムの有用性を示す。

本稿は以下のような構成になっている。2節ではシャドウアートや影の編集に関する従来法について議論する。3節では、本システムの概要について説明する。4節では、本システムの技術的な部分について述べる。5節ではシステムによって生成されたいくつかの例を示す。最後に、6節でまとめと今後の課題について述べる。

2. 関連研究

影を用いたアート作品を生成するための研究は、すでにいくつか存在する。文献[1]では、一見何の意味もなさないように見えるが、ある方向から光を照射すると何らかの意味を持った形状の影が生成される3次元物体の生成方法についてまとめている。この論文では、光源の方向に応じて複数の異なる形状の影を落とす物体の形状を計算する方法を示している。文献[2]では、文献[1]のような複数の影を生成する3D物体をワイヤーで作成する手法について述べている。また、文献[3]では立体ではなく、2次元の遮断層を複数用いたシャドウアート作品を生成する手法を提案している。遮断層を用いることで半影が現れる領域が出現するので、シルエットしか表現できないほかの研究と違い、濃淡画像を表現できるという利点がある。文献[4]では人体を使って影を生成している。目的の影が生成されるよう人体のポーズを最適化する手法が提案されている。文献[6]では自己遮蔽を利用し、複数のグレースケール画像を表示できる平面状の物体を生成する手法を説明している。これらの手法は全て物体を自動的に生成しており、インタラクティブに影を生成するシステムはこれまでほとんど提案されていない。

一方で、コンピュータグラフィックスを用いた映像生成における演出表現を目的としてインタラクティブに影を編集する方法が提案されている。文献[7]では、影をインタラクティブに設計する手法が提案されている。ユーザは、影の平行移動や回転などを直感的に操作でき、システムはユーザの操作に応じた影を投影するよう物体の位置と方向を決定する。文献[8]では光線を曲げることで、ユーザが影の形状を変形できる手法が提案されている。また、文献[9]では、影の境界を編集する手法が提案されている。これらの方法は影の形状を編集できるが、シャドウアートに用い



(a) システムを用いたシャドウアートの生成



(b) 操作中のシステムの様子

図 1: システムの概要

るには適していない。

3. システムの概要

図1にシステムの概要を示す。このシステムでは入力として所望の影の形状を示すシルエット画像(二値画像)を用意する。このシステムでは、単一の点光源を想定しており、その位置はユーザにより指定する。3次元空間内に複数の物体を適切に配置すれば、それらの影が入力画像の形状を形成できる。しかし、3次元空間中の物体の位置を全て手作業で指定し、適切な配置を実現することは容易ではない。また、実際、浮遊物体を用いてシャドウアートを再現するためには、物体の個数は少ない方が作業量が少なく済む。提案システムでは、これらの条件を考慮してユーザをサポートする。

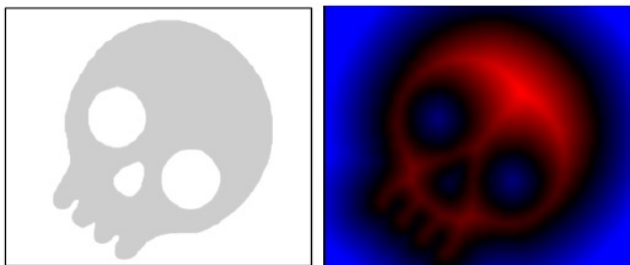
このシステムでは、ユーザは物体の位置を操作するのではなく、は物体の影の位置を操作する。具体的には、ユーザは、影を生成したい平面上の点を指定する。システムは、その位置に影を落とす物体を生成する。このとき、影

の大きさが目的の影形状を表すシルエット内部で最大化されるよう位置や大きさが自動的に調整される。ユーザは、満足いく結果が得られるまで、任意の数の物体を生成できる。図 1(a) に示すように、ユーザは平面上の物体の影を選択して移動することができ、それに応じて物体の位置や大きさが調整され、その結果はリアルタイムに表示される。操作中に物体間の衝突が検出された場合には、通知メッセージが表示され、操作中の影は衝突する方向への移動が制限される。ユーザは、衝突を避けるために物体の高さをスライダーで調整できる。

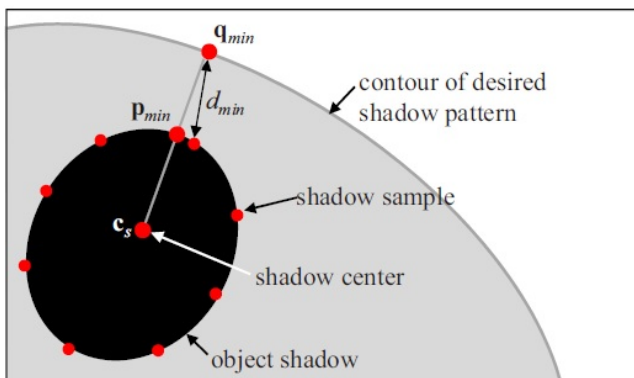
図 1(b) は、操作中のシステムの様子を示している。この例では、球体が使用されている。ユーザは、画面右半分においては真上からみた画像が表示されており、球体の生成や影の移動はこの画面で行うことができる。目的の形状を表す入力画像は明るい灰色で表示されている。左の画面には、異なる視点からレンダリングされた画像が表示され、物体の 3 次元空間での配置と、それによって生じる影が表示されている。

4. 提案手法

本章では、前章で説明したシステムを構成する技術について説明する。このシステムには、2つの課題がある。一つ目は、物体の最適な位置や大きさを算出し、目的画像のシルエット内部で影の大きさを最大にすることである。二つ目は、インタラクティブな操作を実現するのに十分な速さで計算することである。これらの課題に対応できるよう



(a) 入力画像と生成される距離マップ



(b) 入力画像の境界上の最近傍点の計算

図 2: 手法の概要

に手法を設計している。以下、本手法を図 2 を用いて説明する。基本的な考え方は、各物体の位置または大きさを逐次的に更新し、その影の境界が入力画像の境界に接する最適な位置または大きさを求めることである。

まず、目的の影の形状を表す入力シルエット画像から符号付距離マップを計算する。この距離マップの各ピクセルには、そのピクセルから最も近い入力画像の境界線上のピクセルまでの符号付き距離が格納される。ピクセルが目的画像のシルエットの外側にある場合は負の距離とする。図 2(a) は、左側に示される入力画像から生成された距離マップを示している。青は負の距離、赤色は正の距離を表す。符号付き距離とともに、距離マップ内の最も近いピクセルの位置も保存する。

次に、この距離マップを用いて、物体の最適な位置または大きさを決定する。本研究では、物体は三角形メッシュで形成されているものとする。新たな物体が生成されると、システムは物体を表すメッシュの頂点の中で、光源から見たときの稜線（光源方向に向く三角形と反対方向に向く三角形に共有される線）を構成する頂点を抽出する。次に、光源からそれらの頂点方向へレイを生成し、このレイと影平面との交点を計算する。物体が凸な場合、これらの交点は影の境界上に存在する (図 2 参照)。これらの交点を影サンプル点と呼ぶ。影の中心座標 c_s は、光源から物体の中心にレイを生成することによって計算することができる。次に、距離マップを参照して各影サンプル点から境界までの距離を調べ、最小の距離の影サンプル点を求める。以下、最小距離と最近傍点をそれぞれ d_{min} および q_{min} と表す (図 2(b) 参照)。また、物体の影の輪郭と c_s と q_{min} を結ぶ線分が交差する点の座標を計算する (図 2(b) の p_{min})。次に、 p_{min} が q_{min} と一致するように物体の位置または大きさを更新する。なお、 d_{min} が負の場合、影が目標画像の輪郭からはみ出ていることになるので、 p_{min} ははみ出ている点の中で目標画像の輪郭から最も遠い点を選択されることになる。これらのプロセスを $|d_{min}|$ が十分小さくなるまで繰り返す。以下、物体の位置、大きさを更新する方法について述べる。

4.1 物体位置の更新

図 3(a) を用いて物体の位置の計算方法を説明する。光源と影の中心を結ぶ直線に沿って物体の位置を更新する。すなわち、物体と光源との間の距離 t を計算することで位置を定める。光源 p_l 、物体の中心 c 、点 q_{min} は同じ平面上にある。図 3(a) に示すように、 p_{min} に対応する物体上の位置を v であらわす。 v が p_l と q_{min} の間の線分上に位置するように物体の位置を更新する。 v と c の更新された位置を、それぞれ v^* と c^* であらわす。この時、 v^* は以下の方程式を満たす必要がある。

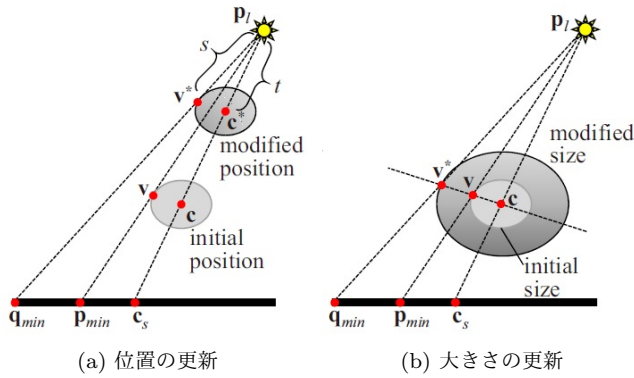


図 3: 物体の位置及び大きさの計算

$$\mathbf{v}^* = s\mathbf{e}_q \quad (1)$$

$$\mathbf{v}^* = \mathbf{p}_l + t\mathbf{e}_c + (\mathbf{v} - \mathbf{c}) \quad (2)$$

ここで s は光源から $\mathbf{v}^*$ までの距離, $\mathbf{e}_q$ 及び $\mathbf{e}_c$ はそれぞれ光源から $\mathbf{q}_{min}$, $\mathbf{c}_s$ への方向の単位ベクトルである. 上記の式を解くことで, 新しい物体の中心位置 $\mathbf{c}^*$ を決定できる.

4.2 物体の大きさの更新

物体の大きさの計算はより簡単である (図 3(b) を参照). 最初に, $\mathbf{v}$ と $\mathbf{c}$ をつなぐ直線と $\mathbf{p}_l$ と $\mathbf{q}_{min}$ をつなぐ直線が交差する点 $\mathbf{v}^*$ を計算する. 次に, 物体のスケールは元の $|\mathbf{v}^* - \mathbf{c}|/|\mathbf{v} - \mathbf{c}|$ の大きさにすることで, 最適な物体の大きさを求めることができる.

5. 実験結果

本節では, 本システムを用いて作成したシャドウアートの例を提示する. Unity を用いてシステムを実装した. 物体には球体を用いて, システムの有用性を検証した.

図 4 は提案システムを用いて設計したシャドウアートのシミュレーション例である. 本システムを用いて球体の配置をデザインし, 髑髏 (図 4(a)), ミッキーマウス (図 4(b)), ドラえもん (図 4(c)) の影を生成した. 入力画像はそれぞれ左上に表示されている. それぞれに要した球体の数とかった時間を表 1 に記す.

図 5 は, システムで設計した球体の位置をもとに, 実際に制作したシャドウアートの例である. 発泡スチロールの球を用いて影を表現している. 外部の光を遮断する箱を用意し, 箱の天井から球体をつるしている (図 5(a)). 天井の中心には, 光を通すための小さな穴が開いている. 光源には, スマートフォンのライトを用いている. 提案システムによって設計した配置に沿って球体を配置する. 実際に配置するとき, それぞれの球体の影の境界線をプリントしたシートを底面に置き, それに従って球体を配置した. 球体を吊るす糸の長さは, それぞれの球体の高さから計算する.

表 1: 実際にシミュレーションした作品とそれぞれに要した物体数及び制作時間

	物体数	時間 (分)
髑髏	68	10
ミッキーマウス	10	2
ドラえもん	17	4

これらのシャドウアートの制作には数時間を要した. 図 5 の (b) や (c) は図 4 の (b) や (c) に従って影を作った例である. システムに沿って制作した結果, 入力画像のシルエットを忠実に再現した影ができた.

6. 結論

本稿では, 浮遊物体を使用してインタラクティブにシャドウアートを作るためのシステムを提案した. このシステムにより, ユーザは物体の影をインタラクティブに直接移動できる. ユーザが影を操作すると, システムは物体の位置または大きさを変更して, その影が目的の影画像内のできるだけ大きな領域を占めるようにする. 球体を使用して, システムが目的のシャドウアートをインタラクティブに作成できることを示した.

今後の課題の一つ目として, 物体の最適な形状変形が挙げられる. 現在, システムは物体の位置またはサイズを変更するが, 物体の形状を最適化することも可能である. これによりシャドウアート制作の複雑さが軽減されることが期待できる. 複雑なシャドウアートをシステムで手動で作成することは非常に困難であるため, 完全自動アプローチへの拡張も検討したい. また, ドローンを使用してシャドウアートを生成することも考えている. これにより, 動的にシルエットが変わるシャドウアートを生成できる.

謝辞

本研究は JSPS 科研費 JP15H05924 および 15K21742 の助成を受けたものです.

参考文献

- [1] Niloy J. Mitra, Mark Pauly, "Shadow Art," *ACM Transactions on Graphics* 28, 5, 156:1-156:7. (2009)
- [2] Kai-wen Hsiao, Jia-bin Huang, Hung-kuo Chu, "Multi-view Wire Art," *Proceeding SIGGRAPH Asia '18 SIGGRAPH Asia 2018 Technical Papers Article No. 242* (2018)
- [3] Sehee Min, Jaedong Lee, Jungdam Won, Jehee Lee, "Soft Shadow Art," *Proceedings of the symposium on Computational Aesthetics Article No. 3* (2017)
- [4] Jungdam Won, Jehee Lee, "Shadow Theatre: Discovering Human Motion from a Sequence of Silhouettes," *ACM Transactions on Graphics* 35, 4. (2016)
- [5] Yunjoo Park, Jyh-Ming Lien, "Fabricate 2.5D Shadow Art Sculpture," *Proceedings of SM' 2016 Fabrication and Sculpting Event (FASE), June 2016* (2016)
- [6] Amit Bermano, Ilya Baran, Marc Alexa, and Wojciech

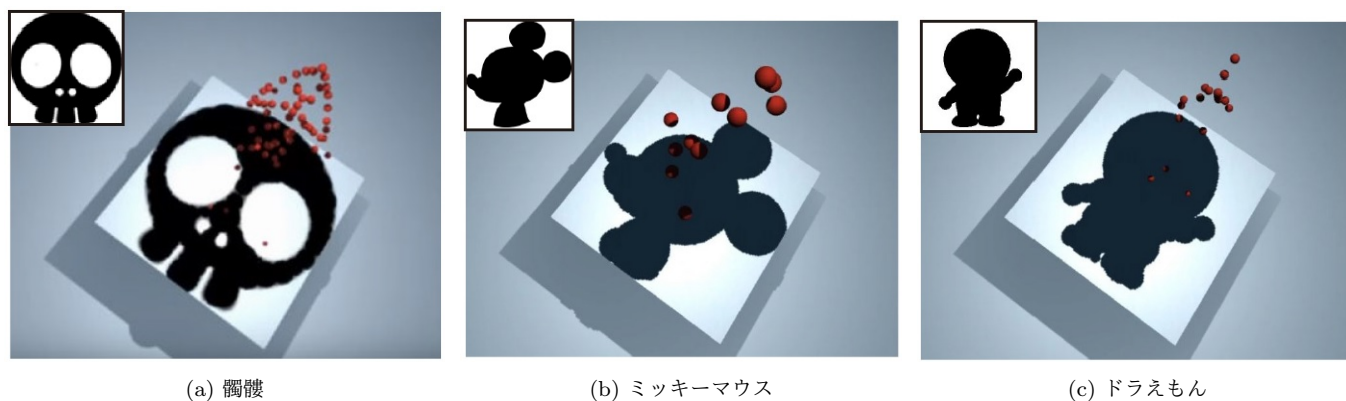


図 4: 入力画像とシミュレーション結果の例

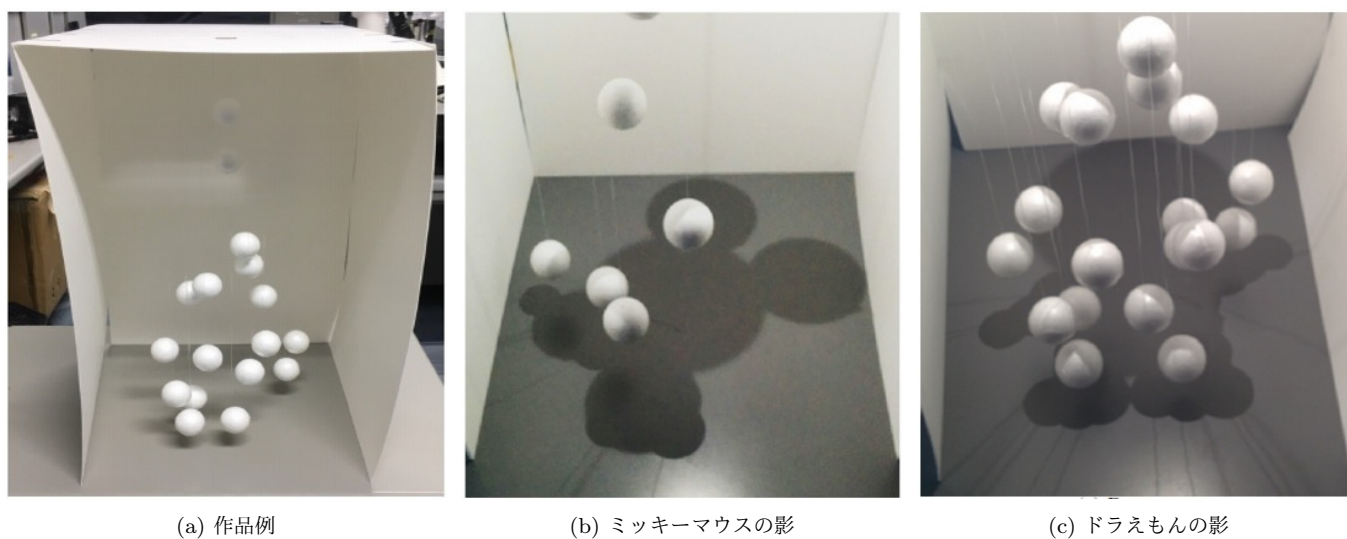


図 5: 実際に制作したシャドウアートの例

- Matusik, "SHADOWPIX: Multiple Images From Self Shadowing," *Computer Graphics Forum (Proceedings of Eurographics 2012)* 31, 2?3 (2012), 593?602. (2012)
- [7] Fabio Pellacini, Parag Tole, and Donald P. Grenberg, "A user interface for interactive cinematic shadow design," *ACM Transactions on Graphics* 31, 3 (2002), 563?566. (2002)
- [8] William B. Kerr, Fabio Pellacini, and Jonathan D. Denning, "BendyLights: Artistic Control of Direct Illumination by Curving Light Rays," *Computer Graphics Forum* (2010). <https://doi.org/10.1111/j.1467-8659.2010.01742.x> (2010)
- [9] O. Mattausch, T. Igarashi, and M. Wimmer, "Freeform Shadow Boundary Editing," *Computer Graphics Forum* 32, 2pt2 (2013), 175?184 (2013)