Vol.2014-CG-154 No.9
2014/2/20

# BendyLightsを利用した影の直感的な編集システム

葛西　哲[1]　土橋 宜典[1]　山本 強[1]

**概要**：CG において影はシーンやキャラクターの印象に大きく影響を与える重要な役割を持っている．しかし，影をデザインするためには光源の位置，向き，大きさといった様々なパラメータを調整する必要がある．そのため，専門的な技術を持たない初心者が所望の影をデザインするには非常に煩雑な作業を必要とする．本稿はソフトシャドウを含む影をインタラクティブに編集出来るシステムを提案する．提案システムでは，陰影や影の位置の移動といった物理法則に従った編集から，影を曲げる・引き延ばすといった物理法則に従わない編集操作まで行うことが出来る．ユーザはマウス操作を用いた直感的な指定から，これらの効果を簡単に実現することが出来る．

**キーワード**：シェーディング, インタラクティブ，直感的，ライティング

# An Intuitive Editing System for a Shadow Using BendyLights

Tetsu Kasai[1]　Yoshinori Dobashi[1]　Tsuyoshi Yamamoto[1]

***Abstract:*** Shadows are important visual clues for computer graphics. In order to obtain user-desired results, the user is required to adjust many parameters of light such as intensity, direction, and size. Then, the shadow-design process is often difficult and tedious for novice users. In this research, we propose an interactive editing system for shadows. Our system can create both physically-correct and incorrect results. In our system, the user can design shadows by intuitive operations.

***Keywords:*** shading, interactive, intuitive, lighting

## 1. はじめに

近年のコンピュータグラフィックス（CG）技術の発達により，現実の世界をリアルに再現することや現実の世界では起こりえないことを表現することが可能になった．このような CG 技術は映画やゲーム，災害シミュレーションなどの様々な場面で応用されるとともに，多くの研究が行われている．その中でも物体に陰影を施すシェーディング技術は CG 技術の中で基礎的でかつ重要な技術の一つである．物体の見え方を決めるシェーディング結果は，キャラクターやシーンの印象に大きく影響を与える．そのため，アーティストはシーン中に配置する光源の位置や方向などの様々なパラメータの編集作業を繰り返すことで，特定のシーンに対して所望の雰囲気やムードを表現する．中でも面光源を用いたソフトシャドウの表現は，シーンのリアリティを向上させる重要な表現である．また，単一の物体に対して複数の光源を組み合わせることで空間ごとに別々のシェーディングを施すライトリンクと呼ばれる手法もよく行われる．しかし，光源の位置，向き，面光源の大きさといった様々なパラメータ調整をしたり，ライトリンクのように複数の光源を組み合わせて所望の陰影効果を表現することは，専門的な知識を持たない初心者には難しい作業である．

本稿ではソフトシャドウを含む影をインタラクティブに編集出来るシステムを提案する．提案システムでは，BendyLights[1] と呼ばれる手法を応用し，簡単なマウス操作によって影に対して様々な編集を行うことが出来る．提案システムでは，陰影や影の位置の移動といった物理法則

[1] 北海道大学大学院情報科学研究科
Graduate School of Information Science and Technology, Hokkaido University

ⓒ 2014 Information Processing Society of Japan

に従った影の表現から，影を曲げる・引き延ばすといった物理法則に従わない影の表現まで行うことが出来る．

BendyLights について簡単に説明する．BendyLights では，スポットライト型の点光源の光の放射方向の中心軸（光軸）を，スプライン曲線を用いて表現する．本稿ではこの曲げられるスポットライトを BendyLight と呼ぶ．ユーザはこのスプライン曲線の制御点を操作することで，光の通り道を操作し，物理的に正しい計算からは得ることが出来ない影を表現することが出来る．しかし，各制御点位置を 3 次元空間上で指定することは難しく，また，所望の効果を実現するためにどのような指定をすればいいかは分かりにくいという問題があった．また，点光源による影しか実現出来ていない．そこで本研究では，BendyLight を面光源へと拡張し，さらに，目的の影を表現出来るように制御点の位置を自動的に計算する方法を提案する．提案システムは，ユーザのマウス操作を基に，BendyLight の制御点や光源位置を最適に決定する．これによりユーザは BendyLight の制御点位置などを直接編集することなく，所望の陰影効果を簡単に実現することが出来る．

以降，本稿の構成は，2 節で関連研究について紹介を行い，3 節で提案手法で利用する BendyLights の概要，そして 4 節で提案手法について述べる．その後，5 節で提案手法を用いた実験結果とその考察について述べる．最後に 6 節でまとめとする．

## 2. 関連研究

ユーザによる直感的な陰影の編集を可能とするシステムは，これまで数多く提案されている．文献 [2] では，物体の影や拡散反射をスクリーン上のマウス操作でインタラクティブに編集することが出来るシステムが提案されている．この手法ではスクリーン上のマウス操作を基に最適な光源の位置や向きを決定している．しかし，この手法では単一の点光源から得られたシェーディング効果のみを対象としているため，ソフトシャドウのような効果は実現出来ない．文献 [3] ではバネモデルを用いて物体表面上の陰影を変形する手法が提案されている．この手法では物体表面上に仮想的な膜を張り，この膜を引き延ばすことで物体表面上の陰影を直接編集することが出来る．しかし，この手法ではバネモデルを用いているため，不自然な変形を行ってしまう可能性がある．また，得られる結果はバネモデルのサンプル解像度に大きく依存し，物体の移動やトポロジの変化が生じるシーンには適用出来ない．文献 [4] ではユーザがハードシャドウの輪郭を直接指定することで，影の変形を行う手法が提案されている．この手法ではあらかじめ影の計算のみに用いる物体を用意し，ユーザの指定を満たすように物体の変形を行うことで所望の効果を実現する．しかし，この手法ではソフトシャドウの編集は出来ない．提案法では，物理法則に従った計算によって得られる影から，物理法則に従わない影まで，ソフトシャドウを考慮しながら直感的に編集が可能なシステムを目指す．

## 3. BendyLights

まず，BendyLights について詳しく説明する．前述の通り BendyLights ではスポットライト型の点光源の光の放射方向の中心軸（光軸）を，スプライン曲線を用いて表現する．図 1(a) に示すように，ユーザはこのスプライン曲線の制御点を操作することで，光の通り道を操作し，物理的な計算からは得ることが出来ない陰影表現を得ることが出来る．以降，詳しく説明する．

まず，BendyLight による光の照射空間の決定方法について説明する．BendyLight による光の照射空間はスプライン曲線を用いて表現する．通常，スポットライト型の点光源の光の照射空間は，光源の向きに対して底面が垂直な円錐として与えられる．BendyLights ではこの円錐の中心軸をスプライン曲線で表現する．提案システムでは，二次の B-スプライン曲線を用いた．そして，図 1(a) に示すように，この中心軸を表すスプライン曲線を編集することにより，光の通り道を編集することが出来る．図 1(b) に BendyLight のモデルを示す．$cp_0$，$cp_1$ および $cp_2$ は二次の B-スプライン曲線の制御点位置であり，$cp_0$ は光源位置に等しい．このとき，中心軸の B-スプライン曲線 $C_c$ はパラメータ $u \in [0,1]$ を用いて以下の式で表わされる．

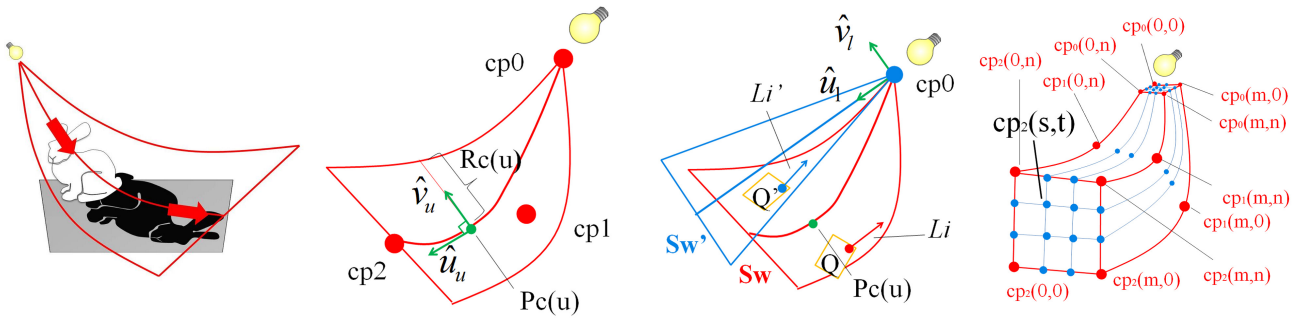
$$P_c(u) = cp_0(1-u)2 + cp_1 2(1-u)u + cp_2 u^2 \qquad (1)$$

ここで，$P_c(u)$ はスプライン曲線 $C_c$ 上の点の位置である．また，BnedyLights では円錐の半径もスプライン曲線を用いて表現し，これを編集することで光の照射空間を局所的に増減させることが出来る．提案システムでは中心軸を表すスプライン曲線と同様に，半径を表わすスプライン曲線も二次の B-スプライン曲線を用いた．半径の B-スプライン曲線 $R_c$ はパラメータ $u \in [0,1]$ を用いて以下の式で表わされる．

$$R_c(u) = cr_0(1-u)2 + cr_1 2(1-u)u + cr_2 u^2 \qquad (2)$$

ここで $cr_0$，$cr_1$ および $cr_2$ は二次の B-スプライン曲線で制御する半径であり，それぞれ，$cp_0$，$cp_1$ および $cp_2$ における円錐の半径を表す．

次に，陰影計算を行うためにシーン全体に対して行う非線形な変形について説明する．BendyLights では光の進む経路が必ずしも直線であるとは限らない．そこで，図 1(c) に示す通り，光の照射空間 $Sw$ が円錐形 $Sw'$ になるように，非線形な変換を施す．この変換後の空間において通常通りの陰影計算を行い，得られた輝度を変換前の物体の輝度として割り当てる．こうすることで BendyLights による陰影計算を効率よく行うことが出来る．以下で詳しく説明する．




(a)BendyLight の使用イメージ　(b)BendyLight の構成　(c) 頂点変形　(d) 面光源化

**図 1**　BendyLights 概要

まず，中心軸を表すスプライン曲線 $C_c$ に沿った局所直交座標系 $(\hat{u}_u, \hat{v}_u, \hat{w}_u)$ を考える．ここで，$\hat{u}_u$ は位置 $P_c(u)$ における中心軸の単位接ベクトルであり，$\hat{v}_u$ および $\hat{w}_u$ は中心軸に直交する平面上の互いに直交する単位ベクトルである．この時，BendyLight による光の照射空間は以下の式を用いて表現することが出来る．

$$P_w(u, v, w) = P_c(u) + R_c(v\hat{v}_u + w\hat{w}_u) \tag{3}$$

ここで，$P_w(u, v, w)$ は光の照射空間内の一点を表す．また $u, v$ および $w$ は $P_w$ を決定するパラメータである．ここで，物体のある頂点 $Q$ に対するパラメータ $u, v$ および $w$ を求める．$u$ は以下の式を解くことで求める．

$$(Q - P_c(u)) \cdot \hat{u}_u = 0 \tag{4}$$

得られた $u$ を用いて $v$ および $w$ についても以下の式を解くことで求める．

$$v = ((Q - P_c(u)) \cdot \hat{v}_u)/R_c(u) \tag{5}$$

$$w = ((Q - P_c(u)) \cdot \hat{w}_u)/R_c(u) \tag{6}$$

求められた $u, v, w$ を使用することで，光の進む経路が曲がった光の照らす空間 $Sw$ を図 1(c) のように円錐形 $Sw'$ に変形することが出来る．円錐形の光の照らす空間 $Sw'$ における新しい頂点位置 $Q'$ は以下のように求められる．

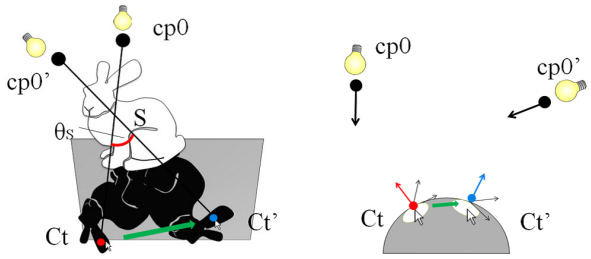
$$Q' = cp_0 + u\hat{u}_l + v\hat{v}_l + w\hat{w}_l \tag{7}$$

ここで $cp_0$ は光源位置，$\hat{u}_l$ は光源位置における光源方向の単位ベクトル，$\hat{v}_l, \hat{w}_l$ は $\hat{u}_l$ に対して直交する平面上の互いに直交する単位ベクトルである．ある頂点 $Q$ でのシェーディングは，変形した頂点位置 $Q'$ における通常通りの陰影計算を行い，シェーディング結果を $Q$ に割り当てることで行う．

## 4.　提案手法

### 4.1　BendyLight の面光源への拡張

提案システムでは 4 つの制御用 BendyLight を用いて矩形の面光源を表現する（図 1(d) 参照）．それぞれの制御用 BendyLight は面光源の四隅に相当し，これらの BendyLight の制御点を編集することにより，ユーザは所望の光の通り道を表現出来る．ここで，面光源の四隅以外の領域に関しても BendyLight を用いて表現する．これらの BendyLight の制御点位置は制御用 BendyLight の制御点位置を補間することにより求める．ここで，面光源を $m \times n$ 個の BendyLigth を用いて表現する場合，$s$ 行，$t$ 列目の BendyLight の $i$ 番目の制御点位置 $cp_i(s, t)$ はとその半径 $cr_i(s, t)$ は次式のように決定される．




(a)　(b)

**図 2**　影を移動するモード・拡散反射を移動するモードの概要

$$\begin{aligned} cp_i(s,t) =& (1-a)(1-b)cp_i(0,0) + (1-a)bcp_i(0,n) \\ &+ a(1-b)cp_i(m,0) + abcp_i(m,n) \end{aligned} \tag{8}$$

$$\begin{aligned} cr_i(s,t) =& (1-a)(1-b)cr_i(0,0) + (1-a)bcr_i(0,n) \\ &+ a(1-b)cr_i(m,0) + abcr_i(m,n) \end{aligned} \tag{9}$$

$$a = \frac{s}{m-1}, b = \frac{t}{n-1} \tag{10}$$

このようにして求めた BendyLight に関して別々にシェーディングを行い，その結果を加算することで，ソフトシャドウの効果を表現する．

### 4.2　影の編集方法

BendyLight の各制御点位置をユーザが指定することで，光の進行方向を編集出来る．しかし，各制御点位置を直接 3 次元空間上で指定することは難しく，また，所望の効果を実現するためにどのような指定をすればいいかは分かり

にくい．そこで提案システムでは，BendyLight を用いて表現出来る局所的な影の移動や拡大縮小モードといった表現を直感的に実現出来る編集モードを用意している．これらのモードを用いることで，ユーザは制御点位置を直接編集することなく，所望の陰影効果を簡単に実現することが出来る．また，物理法則に従った陰影の編集方法には，文献 [2] の手法を応用する．以下，順に説明する．

#### 4.2.1 BendyLights を用いた直感的な影の編集

BendyLights を用いた陰影表現の編集モードとして，局所的な影の移動モードと拡大縮小モードの 2 つを用意する．まず，局所的な影の移動モードについて説明する．この編集モードでは，物体のシェーディング結果を出来るだけ維持しながら，影を掴むように曲げる・伸縮するといった操作を行える．この編集モードの概要を図 3 に示す．このモードでは，まず影上でクリックした点を $C_t$ とし，光源位置 $cp_0$ から $C_t$ に対してレイを放射し，初めに交差した点の $S$ を求める．この点 $S$ はクリックした点 $C_t$ に対して影を落とす点と考えられる．提案システムは，クリックした点 $C_t$ を $C'_t$ に移動したとき，そこでの光の放射方向を保つとともに，影を落とす $S$ における光の放射方向も出来る限り保つように，BendyLight のパラメータを算出する．BendyLight における光の放射方向は 3 節で解説したパラメータ $(u, v, w)$ に依存する．そしてこれらのパラメータは，制御点位置 $cp_0, cp_1, cp_2$ が移動することによって変化する．すなわち，上記を実現するためには，移動後の $C'_t$ のパラメータとクリック時の $C_t$ のパラメータの差分および移動後とクリック時の $S$ のパラメータの差分を小さくする必要がある．ここで，ドラッグ後の $S$ のパラメータを $u'_s, v'_s, w'_s$，$C'_t$ のパラメータを $u'_c, v'_c, w'_c$ とし，クリック時の $S$ のパラメータを $u_s, v_s, w_s$，$C_t$ のパラメータを $u_c, v_c, w_c$ とする．このとき，以下の評価関数を考える．

$$\begin{aligned} E_1(cp_0, cp_1, cp_2) =& (u_c - u'_c)^2 + (v_c - v'_c)^2 + (w_c - w'_c)^2 \\ &+ (u_s - u'_s)^2 + (v_s - v'_s)^2 + (w_s - w'_s)^2 \end{aligned} \tag{11}$$

最適な制御点位置は，$cp_0, cp_1, cp_2$ を変化させ，式 (11) を最小化することで求められる．それぞれの $x, y, z$ 座標を $(x_0, y_0, z_0)$,$(x_1, y_1, z_1)$,$(x_2, y_2, z_2)$ とし，最急降下法を用いて最適な制御点位置を決定する．例えば，次式は $cp_0$ の $x$ 座標に関する更新式を表したものである．

$$\begin{aligned} x_0(t+1) &= x_0(t) - \alpha \frac{\partial E}{\partial x_0} \\ &= x_0(t) - \alpha \frac{E(cp_{0+\Delta D}, cp_1, cp_2) - E(cp_0, cp_1, cp_2)}{\Delta D} \end{aligned} \tag{12}$$

$D$ はステップ幅で，$cp_{0+\Delta D}$ は $cp_0$ をステップ幅 $D$ 分 $x$ 軸方向へ移動した位置である．$\alpha$ は係数，$t$ は探索回数，および初期値である $x(0)$ はクリック時の $cp_0$ の $x$ 座標である．

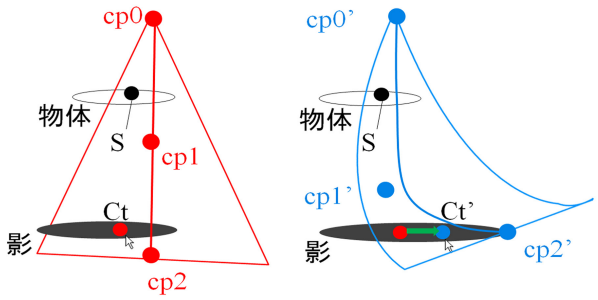



図 3　BendyLight を用いた直感的な影の編集

そのほかの制御点の座標についても同様に求められる．提案システムはユーザがドラッグ操作をしている間，制御点位置を最適に変更し，リアルタイムで陰影結果を更新することで，シーン中の影を掴むように曲げる・伸縮させることが出来る．

また，局所的な影の拡大縮小モードでは同様の処理を，BendyLight の半径に対して行う．このモードでは，物体の陰影結果を出来るだけ維持しながら，影を掴むように拡大縮小するといった操作を行える．BendyLights において，影の輪郭の大きさは光の通り道の直径に依存する．提案システムは，$S$ における光の放射方向を出来る限り保つとともに，ドラッグ操作中の点 $C'_t$ での光の放射方向を，クリックした点 $C_t$ での光の放射方向と出来る限り同じになるように，光の通り道の直径を変更する．これは前項 4.2.1 と同様に，移動後の $C'_t$ のパラメータとクリック時の $C_t$ のパラメータの差分，移動後とクリック時の $S$ のパラメータの差分を小さくすることで実現する．このモードではパラメータ $(u, v, w)$ を変化させる $cr_0, cr_1, cr_2$ の最適化を，次式を最小化するように最急降下法を用いる．

$$\begin{aligned} E_2(cr_0, cr_1, cr_2) =& (u_c - u'_c)^2 + (v_c - v'_c)^2 + (w_c - w'_c)^2 \\ &+ (u_s - u'_s)^2 + (v_s - v'_s)^2 + (w_s - w'_s)^2 \end{aligned} \tag{13}$$

例えば $cr_0$ 次式は cr0 に関する更新式を表したものである．

$$\begin{aligned} cr_0(t+1) &= cr_0(t) - \alpha \frac{\partial E}{\partial cr_0} \\ &= cr_0(t) - \alpha \frac{E(cr_0 + \Delta D, cr_1, cr_2) - E(cr_0, cr_1, cr_2)}{\Delta D} \end{aligned} \tag{14}$$

$D$ はステップ幅，$\alpha$ は係数，$t$ は探索回数，初期値である $cr(0)$ はクリック時の半径 $cr_0$ である．そのほかの制御点の半径についても同様に求められる．このように，提案システムはユーザがドラッグ操作をしている間，制御点の半径を最適に変更する．そして，リアルタイムで陰影結果を更新することで，シーン中の影を掴むように拡大縮小させることが出来る．

以上の方法により，物理的には正しくないが，ユーザの所望する影の表現を直感的な編集で得ることができる．

ⓒ 2014 Information Processing Society of Japan

#### 4.2.2 物理法則に従った陰影の編集方法

物理法則に従った陰影の編集方法として，提案システムは影の移動モードと拡散反射の移動モードを用意する．

影の移動モードでは，影を掴むようにドラッグ操作で動かすことが可能である．図 2(a) にこの編集モードの概要を示す．このモードではまず，光源位置 $cp_0$ からユーザがクリックした点に対してレイを放射し，初めに交差した点 $S$ を求める．次に，この点を中心として光源位置を含むすべての BendyLight 制御点を回転することにより，ユーザがドラッグ操作を終えた点に影を移動する．例えば，編集後の光源位置 $cp'_0$ は以下の式で与えられる．

$$cp'_0 = (cp_0 - S)R(\theta_S) + S \tag{15}$$

ここで $cp'_0$ は新しい光源位置，$cp_0$ は初期の光源位置である．また，$S$ は影を落とす点，$\theta_S$ は $S$ と $cp_0$ を結ぶ直線と $S$ と $cp'_0$ を結ぶ直線がなす角度，$R$ は回転行列である．残りの制御点位置である $cp_1$，$cp_2$ に対しても $cp_0$ に行った処理と同様の処理を行うことで，新しい制御点位置 $cp'_1$，$cp'_2$ を決定する．提案システムはユーザがドラッグ操作をしている間，回転角を計算しリアルタイムでシェーディング結果を更新することで，シーン中の影を掴むように移動させることが出来る．

拡散反射の移動モードでは，物理法則に従った範囲で，物理法則に従った範囲で拡散反射の明るさを掴むように移動を行える．の編集モードの概要を図 2(b) に示す．このモードではまず，ユーザがクリックした点における法線と光源方向のなす角度を求める．次に，この角度を満たすようにユーザのドラッグ操作に応じて光源位置と光の放射方向を変化させることにより，ユーザがドラッグ操作を終えた点にクリックした点と同様の拡散反射光成分の輝度を与える．例えば，編集後の光源位置 $cp'_0$ は以下の式で与えられる．

$$cp'_0 = (cp_0 - C_t)R(\theta_N) + C'_t \tag{16}$$

ここで $cp'_0$ は新しい光源位置，$cp_0$ は初期の光源位置である．また，$C_t$，$C'_t$ はマウスでクリックしたスクリーン座標に対応するシーン中の三次元座標とそのドラッグ後の三次元座標である．$\theta_N$ は $C_t$ における法線ベクトルと $C'_t$ における法線ベクトルがなす角度である．残りの制御点位置である $cp_1$，$cp_2$ に対しても同様の処理を行うことで，新しい制御点位置 $cp'_1$，$cp'_2$ を決定する．以上により，ユーザのドラッグ操作をする間，シーン中の拡散反射を掴むように移動することが出来る．

**表 1** 頂点の変形にかかる計算時間 (ms/frame)

| 頂点数 | CPU | GPU |
|---|---|---|
| 2000 | 121.2 | 46.0 |
| 20000 | 666.0 | 110.2 |

## 5. 実験結果

本節では提案システムを用いて作製した結果例を示す．全ての結果例はユーザによるインタラクティブな編集操作によって得られた．実験環境は，CPU が Intel Core i7 2600K，GPU が NVIDIA GeForce GTX 580，グラフィックス API が OpenGL である．頂点の変形を，CPU のみで行った計算時間と，GPU を用いて行った計算時間を表 1 に示す．提案システムはすべての頂点に変形を行うが，GPU を用いて高速化することで，リアルタイムな編集を可能にしている．

影の移動モード，局所的な影の移動モードおよび拡大縮小モードを比較した結果を図 4 に示す．図 4(a) のように魔女のオブジェクトがあった場合に，図 4(b) は影の移動モードを用いて編集を行った結果である．図 4(c) は局所的な影の移動モードを用いて編集を行った結果である．図 4(d) は局所的な影の拡大縮小モードを用いて編集を行った結果である．単純な影の移動を行う図 4 では，赤い楕円部のシェーディングも変わり，影がなくなってしまっている．一方，局所的な影の移動モードの図 4(c) では赤い楕円部のシェーディングを維持したまま，影をドラッグするかのように伸縮させながら移動出来ていることが分かる．図 4(d) も同様に，赤い楕円部のシェーディング結果を維持したまま，影をドラッグするかのように拡大出来ていることが分かる．このように，提案システムでは物理的に正しいとは言えない結果ではあるが，ユーザの意図した局所的な影の編集を行うことが出来る．

また，面光源を使用して，局所的な影の移動モードを用いて編集した結果例を図 5 に示す．面光源のサンプリング数は $10 \times 10$ 個である．図 5（a）は影を下へ移動するストローク方向を示す．図 5（b）はその結果設定された新しいスプライン曲線の制御点位置を使用してレンダリングした画像である．ソフトシャドウでも点光源の影の移動モードと同様に，直接ドラッグするかのように影の移動が出来ていることを確認出来る．

## 6. まとめ

本稿では，マウス操作によるインタラクティブな影の編集システムを提案した．提案システムの各モードを使い分けることで，ユーザは直感的に所望の影を表現することが可能である．提案システムにより，BendyLight の制御点位置とその半径を最適に決定することで，影の移動や拡大縮小といった局所的な編集操作も直感的に行えるようになった．今後の課題は，より柔軟な編集を可能にするために，BendyLight のスプライン曲線やシーン中の制御点を増やすこと，輝度変化を考慮することが考えられる．

ⓒ 2014 Information Processing Society of Japan

(a) (b) (c) (d)

図 4 結果画像

(a) (b)

図 5 ソフトシャドウを利用した例

## 参考文献

[1] William B. Kerr, Fabio Pellacini, Jonathan D.Denning, "BendyLights: Artistic Control of Direct Illumination by Curving Light Rays," Computer Graphics Forum (In Proceedings of Eurographics Symposium on Rendering 2010), No.2, pp.175-184, 2010.

[2] Fabio Pellacini, Parag Tole, Donald P. Greenberg,"A User Interface for Interactive Cinematic Shadow Design," ACM Transactions on Graphics (In Proceedings of ACM SIGGRAPH 2002), Volume.21, No.3, pp.563-566, 2002.

[3] T. Ritschel, T. Thormahlen, C. Dachsbacher, J. Kautz and H.P. Seidel,"Interactive On-Surface Signal Deformation," ACM Transactions on Graphics (In Proceedings of ACM SIGGRAPH 2010), Volume.29, No.4, Article No.36, 2010.

[4] Oliver Mattausch, Takeo Igarashi, Michael Wimmer, "Freeform Shadow Boundary Editing,"Computer Graphics Forum, Volume.32, No.2, pp.175-184, 2013.

ⓒ 2014 Information Processing Society of Japan