

## 2次元テクスチャを用いた炎の動的表現†

大島 登志一†† 板橋 秀一†††

CGによる自然対象物の表現について、すでに多くの研究が報告されているが、その中で雲などに代表される種類の不定形な自然対象物(以下、雲状対象物)のCG表現に関しては、アニメーションを中心にしたものは多くない。本研究では、アニメーションの制作を目的として、モデルの理論的妥当性や写実性よりもデザインの柔軟性とコストの低さを重視した雲状対象物の表現のための一手法の開発を行った。本手法は、2次元テクスチャを適宜加工することにより、高度に複雑な視覚的実体である炎などの雲状対象物の流れを動的に表現しようとするものである。本稿では、まず従来のテクスチャを時間軸方向に拡張した動きを伴ったテクスチャである「モーション・テクスチャ」の概念を提案する。これを用いることによって、3次元情景モデルそのものには手を加えることなく、映像に効果的に動きを付加することができる。そして、大まかなデザインが感覚的に行えるように意図して雲状対象物を視覚的な特徴から2次元的にモデル化し、2次元モーション・テクスチャをデザインすることによって雲状対象物を表現する「2次元テクスチャ法」について述べる。また、本手法による炎の動的表現の例を示す。燃えさかる炎をリアルに表現できることが本手法の目標の一つであるが、実験の結果、この目標をある程度達成することができた。

### 1. はじめに

コンピュータ・グラフィクス(以下CG)による自然対象物の表現についての研究は、よりリアルなCG映像を制作するためのみならず、それ自体非常に興味深い領域であり、すでに多くの研究が報告されている<sup>1)</sup>。本稿では、自然対象物の中でも特に、炎・雲・煙などに代表される種類の自然対象物(以下、雲状対象物)の「流れ」を2次元的なモデル化によって表現する一手法(2次元テクスチャ法)について述べる。

ここでいう雲状対象物とは、対象物全体の大きさに比べて個々の大きさを無視できるほど小さい無数の構成要素(微粒子、分子など)が、密に集中して固定的な塊を形成することなく比較的疎で流動的な分布状態を示すような対象物である。構成要素の空間密度とその特性および状態(温度、発光スペクトルなど)の分布パターンによって観察される形態が決まるが、これは乱流の場において時々刻々と変化し、一般に非常に複雑である。このような雲状対象物に独特な形態の非定常性は、雲状対象物をその他の対象物から区別する顕著な特徴であると同時に、そのCG表現技術の開発を困難にしている要因そのものでもある。雲状対象物の代表的な例である雲の表現については、霧の中での照明効果や様々な雲の形状定義法および画像生成手

法が提案されている<sup>2)-8)</sup>。しかし、雲の動きにはあまり言及していない研究が多い。一方、炎の場合、雲とは異なりアニメーションに静止像を使えないのであるが、形態の時間的変化が非常に激しく複雑なため、その動的表現が特に難しく、研究の報告が少ない。

炎の表現に関する代表的な研究例としては、Reevesのパーティクル・システム(particle system)<sup>9)</sup>などがある。Reevesの手法は、炎を多数の発光性の微粒子からなる系と見なし、系内での各微粒子の生成・移動・消滅を制御することによって動的に炎を構成するものである。これは十分遠距離から観察した大規模な火災や花火の表現などには向いているが、間近に見る焚き火のような炎を表現するのは難しい。最近では、河野らがセル・オートマトンを用いて燃焼をシミュレーションすることにより炎の表示を行うことを研究中である<sup>11)</sup>。また、千葉らは単純化した流体のモデルを考案して炎のシミュレーションを試みており<sup>12)</sup>、これらの今後の研究成果が期待される。焚き火のような燃えさかる炎をリアルに表現できる技法は、概してまだ確立されてはいない現状にある。本手法は、以下に述べるように、シミュレーション的手法とは異なる観点から「焚き火」の映像表現を実現しようとするものである。

CG映像制作のための手法を大まかに「シミュレーション的手法」と「マニュアル・デザイン的手法」に分けることができる。「シミュレーション的手法」では、ユーザが介入する部分をより少なく抑えつつ、表現すべき対象物の範囲を逸脱することなく自然な映像を得られるようなモデルを考える。しかしすべての自

† Representation of Flaring Flames by Using 2-D Texture by TOSHIKAZU OHSHIMA (Doctoral Program in Engineering, Graduate School, University of Tsukuba) and SHUICHI ITAHASHI (Institute of Information Sciences and Electronics, University of Tsukuba).

†† 筑波大学大学院博士課程工学研究科

††† 筑波大学電子・情報工学系

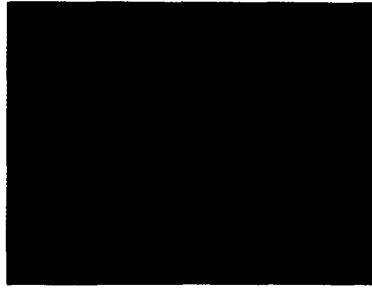


図 1 「台風」  
Fig. 1 Typhoon.

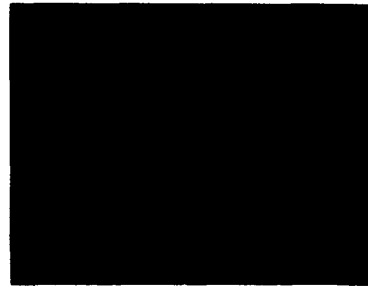


図 2 「星雲」  
Fig. 2 Nebula.

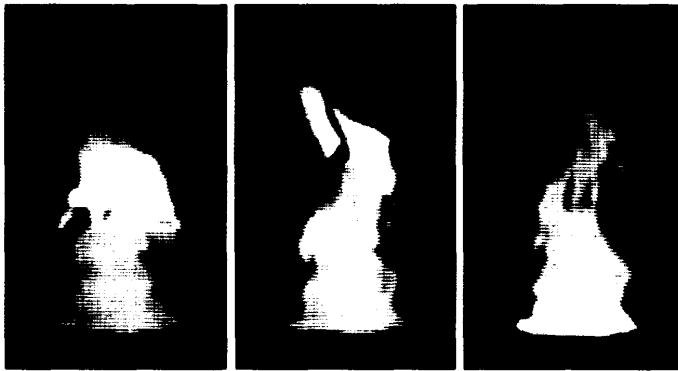


図 19 炎 (実写)  
Fig. 19 Real flames.

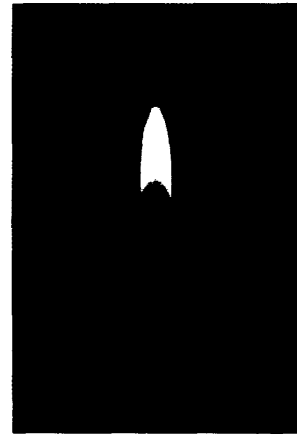


図 20 ろうそくの火 (実写)  
Fig. 20 A real candle light.

然対象物についてそのようなモデル化が実現されているわけではなく、また用途により必要十分なレベルの映像を得るには効率が良くないという場合も考えられる。一方「マニュアル・デザイン的手法」ではデザイナーの積極的介入を前提として、表現する対象物の範囲を狭く特定したモデルは使用せず、本質的に試行錯誤を伴うが、採用したモデルの範囲内で自由な表現が可能である。写実性を否定するものではないが、必ずしも写実性にはとらわれない感覚的な表現に適している。本手法では、マニュアル・デザイン的手法の柔軟性を活かして、シミュレーション的手法では現在難しい炎など雲状対象物の高度に複雑な形態変化に対応する。

本稿では「モーション・テクスチャ」の概念を提案し、モーション・テクスチャをデザインすることにより雲状対象物の流れを表現する「2次元テクスチャ法」について述べる。また本手法による炎の表現例を示す。

## 2. モーション・テクスチャ

写実的な CG 映像を制作する際、物体表面上に見られる種々のディテイルを描写するための効果的な手法として、テクスチャ・マッピング (texture mapping)<sup>13)</sup> が頻繁に使用される。これは、3次元空間でモデル化された物体表面上の各点に対し、2次元空間で一種の画像、すなわちテクスチャとして定義されたある属性パラメータを対応付けることによって、観察される様々な面の状態をシミュレートするものである。色彩をはじめ凹凸・反射・屈折・透過・影などあらゆる応用が考えられ、基本的な技法として普及している<sup>5), 14)-16)</sup>。また、使用するテクスチャについても数多くの研究がなされている。さらに、テクスチャは2次元から3次元のソリッド・テクスチャに拡張され、大理石などの石材や木材など3次元の素材から削り出された物体の質感をリアルに表現できるようになった<sup>10), 17)</sup>。本稿で提案する「モーション・テクスチャ (motion texture)」は、従来のテクスチャを時

間軸方向に拡張するものであり、動きを伴ったテクスチャとして理解される。アニメーションをテクスチャとして使用するというのがモーション・テクスチャの考え方である。「 $n$ 次元モーション・テクスチャ」というとき、 $n$ 次元テクスチャの時系列を意味する。モーション・テクスチャを用いることによって、3次元情景モデルそのものには手を加えることなく、映像に効果的に複雑な動きを付加することができる。

### 3. 2次元テクスチャ法

#### 3.1 概要

著者らは、2次元テクスチャを加工することによって炎の動的表現を試み、炎の持つ独特の雰囲気や良く表す映像が制作できることを見出した<sup>18)</sup>。そして、テクスチャを中心的な表現素材とした対象物の表現を提案し、炎の表現に用いた手法をより一般化して炎のほかにも台風・星雲の動的表現を試みることにより、このような手法で雲状対象物の流れを表現できることを示唆した(図1・2参照)<sup>19)~22)</sup>。

本稿でまとめる2次元テクスチャ法は、雲状対象物を視覚的な特徴から2次元的にモデル化し、素材として選択した2次元テクスチャに対して画像変換処理を施し適宜加工することにより、高度に複雑な視覚的実体である炎などの雲状対象物を表現する2次元モーション・テクスチャをデザインするものである。

#### 3.2 モーション・テクスチャ関数

本手法におけるすべての処理は関数として扱う。関数の説明で共通に使用する記号を表1に示す。式中ではこれらの記号に適宜添え字を付ける。「その他のパラメータ」とは、関数の入力のうち、1)座標ベクトル・2)濃度値・3)時刻以外のものである。例えば、与えられた配列を参照して出力値を決定する関数の場合、配列を指す変数名が「その他のパラメータ」のひとつとなる。大文字の記号は、各々の小文字の記号が表すパラメータの0個以上の組を意味する。これ

が空であるときには $\{\emptyset\}$ のように記述する。濃度値は常に $[0, 1]$ の範囲をとるものとする。座標ベクトルの定義域については後述する。また、座標ベクトルはその定義域外では、あらかじめ「定義域外」であることを意味すると決めた状態を持つものとする。座標ベクトルのこのような状態を「無効」とであると表現する。関数内部での計算過程で出力座標ベクトルが有効な値を持ちえない場合も同様である。

本手法で構成する2次元モーション・テクスチャの基本的な形式は、式(1)に示すように、2次元単位領域における座標ベクトル  $\mathbf{p}_p$  および任意に設定した時間スケールでの時刻  $t$  を入力すると濃度値  $d_{out}$  を返す関数(以下、モーション・テクスチャ関数)である。出力の濃度値は、このテクスチャの用途によって、任意にベクトルなどに変換する。代表的な例として、色彩画像を得るためには、これを色ベクトルに変換する。実際に画像データを生成する際には、これを定義域内の点に対応するすべての画素について評価する。

$$d_{out} = \text{motion-texture}(\mathbf{p}_p, t, C) \quad (1)$$

このようなモーション・テクスチャ関数を、後述する「基本操作関数」を組み合わせる構成する。基本操作関数はモデル化した雲状対象物の各要素に対応する関数である。これらの基本操作関数は一つ以上の「要素関数」から構成する。要素関数はモーション・テクスチャ関数の最小構成単位である。基本操作関数は、要素関数と同様の形式であり、基本操作関数ごとに定めた特定の条件を満たすように特殊化した要素関数であると見なすこともできるので、他の基本操作関数を要素関数として新しい基本操作関数を構成することができる。要素関数の形式を表2に示す。要素関数は座標ベクトルまたは濃度値を出力し、各々を「座標ベクトル関数」・「濃度値関数」として区別する。入力座標ベクトルの中に前述した「無効」のものがある場合には、座標ベクトル関数は「無効」の座標ベクトルを、濃度値関数は0を返すものとする。要素関数のこの条件は上位の関数である基本操作関数に継承される。要素関数は、入力の座標ベクトルと濃度値の各々

表1 式中の記号  
Table 1 Symbols used in the equations.

記号	意味
$\mathbf{p}$	2次元座標ベクトル
$\mathbf{P}$	0個以上の2次元座標ベクトルの組
$d$	濃度値 $[0, 1]$
$D$	0個以上の濃度値 $[0, 1]$ の組
$c$	その他のパラメータ
$C$	0個以上のその他のパラメータの組
$t$	時刻

表2 要素関数  
Table 2 The primitive functions.  
基本形式:  $output = primitive(\mathbf{P}, D, t, C)$

関数 $primitive$	出力 $output$
座標ベクトル関数	$\mathbf{p}$
濃度値関数	$d$

表 3 雲状対象物の視覚的要素と操作および基本操作関数  
Table 3 The visual components, operations, and basic functions.

基本操作関数の基本形式:  $output = basic(P, D, t, C)$

視覚的要素	操 作	関数 basic	P 入力	D	output	記号
1) 歪み・揺らぎ	歪み変換 (座標変換)	歪み変換関数	$\{P_p\}$ or $\{P_w\}$	$\{\phi\}$	$P_w$	歪
2) 流れ	流れ変換 (座標変換)	流れ変換関数	$\{P_w\}$	$\{\phi\}$	$P_f$	流
3) 微細構造	素材テクスチャの供給	テクスチャ関数	$\{P_t\}$	$\{\phi\}$	$d$	微
4) 大局的形状	成型処理 (局所的濃度変換)	成型処理関数	$\{P_w\}$	$\{d\}$	$d$	成
5) (映像全体)	補助的操作: 濃度階調変換	濃度変換関数	$\{\phi\}$	$\{d\}$	$d$	変
6) (映像全体)	補助的操作: 濃度値の混合	濃度値混合関数	$\{\phi\}$	$\{d_1, \dots, d_n\}$	$d$	混

の数によって多くのバリエーションが考えられる。

3.3 雲状対象物のモデル化

図 3 に示すように、雲状対象物の視覚的特徴を次に挙げる四つの要素に分解することにより、雲状対象物の 2 次元的なモデル化を考える。

- 1) 歪み・揺らぎ
- 2) 大局的形状
- 3) 流れ
- 4) 微細構造

このモデルにおいて、雲状対象物のおおまかな形状を表す「大局的形状」は「歪み・揺らぎ」にしたがって変形する。雲状対象物の局所的なディテイルを表す「微細構造」は、「流れ」にしたがって「大局的形状」

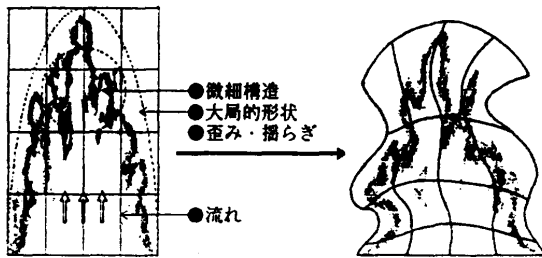


図 3 雲状対象物の視覚的特徴 (例: 炎)  
Fig. 3 Visual components of a fuzzy object (ex. flames).

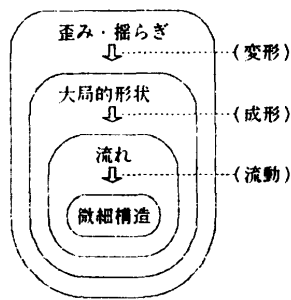


図 4 要素間の作用関係  
Fig. 4 Actions of the components on the others.

表 4 特別な座標ベクトルの記号

Table 4 Symbols used for the special coordinate vectors.

記号	意 味	定 義 域
$P_p$	主座標ベクトル	$\{(x, y)   0 \leq x \leq 1, 0 \leq y \leq 1\}$
$P_w$	歪み座標ベクトル	$\{(x, y)   0 \leq x \leq 1, 0 \leq y \leq 1\}$
$P_f$	流れ座標ベクトル	$\{(x, y)   -\infty < x < \infty, -\infty < y < \infty\}$

の枠組みの中で流動する。この「流れ」もまた「歪み・揺らぎ」の作用を受け「大局的形状」とともに変換する。各要素が他の要素に及ぼす作用の、このような階層的上下関係を図 4 に示す。

「歪み・揺らぎ」と「流れ」は座標系に効果を及ぼす要素であり、「微細構造」と「大局的形状」は各点の濃度に関する要素である。これらの各要素に対して、各々を表現するための操作を表 3 に示すように対応させる。すなわち、雲状対象物の視覚的・微細構造を表現する素材テクスチャを変形 (歪み変換) ・シフト (流れ変換) することによって歪み・揺らぎ・流れを表し、局所的な濃度変換 (成型処理) を施すことによって全体的な形状を決定する。これらの処理は各々に対応した基本操作関数によって行う。基本操作関数としては、4 種類に分類した視覚的要素の各々を受け持つ関数のほかに、補助的操作のために 2 種類の関数があり、表 3 に示した計 6 種類の関数を使用する。補助的操作には、1) 画像全体の濃淡変化の調子を変えるための濃度階調変換と、2) 複数の中間生成画像を合成するための濃度値の混合とがある。基本操作関数の入出力となる座標ベクトルは、各々特別な意味を持つ座標系に属するものであり、定義域が異なる。表 3 の中では、これらの座標ベクトルを識別するために、座標ベクトルの記号  $P$  に添え字を付けてあり、表 4 にその一覧を示す。座標系については 3.4 節で述べる。

特定の接続関係の下で基本操作関数を組み合わせる

ことによりモーション・テクスチャ関数を構成する。モーション・テクスチャ関数の構成は、基本操作関数を表すノードをデータの流れる表す矢印で接続する一種の有向グラフによって記述する。このようなダイアグラムは一般に処理の流れを把握するのに有用である。ノードは円の中に基本操作関数の種類を識別するための記号として表3に示すような漢字一字を書き入れて表記する。モーション・テクスチャ関数の表記を座標ベクトル・濃度値の流れを表す矢印と共に図5(a)に示す。モーション・テクスチャ関数の基本的な構成例を図5(b)に示す。これは、二つの歪み変換関数と各ひとつの流れ変換関数・テクスチャ関数・成型処理関数から成る。図5(b)では、実質的に1段にまとめることのできる歪み変換関数を2段に分解したまま記述してある。これは、2種類の補助的な基本操作関数(濃度変換関数・濃度値混合関数)以外の基本操作関数の、可能な接続関係のすべてを、この図に示すためである。

濃度変換関数は、図5(c)に示すように、濃度値を出力する関数(\*印)の出力側に任意に付加することができる。また、図5(d)に示すように、途中で分岐した処理(\*印)によって得られる複数の濃度値は、濃度値混合関数によってひとつの濃度値にまとめる。濃度値混合関数は、各々機能の異なる複数の濃度値入力を持つので、それらを識別するために、その端子と

して、ノードを囲む同心円を一つずつ対応させることにする。各端子は、その内側からの順番で、引数並びの中の濃度値との対応が一意に決定される。

続く三つの節において、各処理について述べる。

### 3.4 歪みと流れの座標系

モーション・テクスチャ関数の内部では、処理に応じて、次のような3段階の階層的な座標系を扱う。

- 1) 主座標系
- 2) 歪み座標系
- 3) 流れ座標系

「主座標系」とは、対象物の形状を記述するための基準となる座標系であり、モーション・テクスチャ関数の入力座標ベクトルが属する座標系である。主座標系上のベクトル「主座標ベクトル」は、モーション・テクスチャ関数の定義域である単位領域  $\{(x, y) | 0 \leq x \leq 1, 0 \leq y \leq 1\}$  で定義する。

「歪み座標系」とは、対象物の形状とその動き・歪み・揺らぎを表現するための、主座標系上で歪められた座標系であり、成型処理関数の入力座標ベクトルが属する座標系である。歪み座標系上のベクトル「歪み座標ベクトル」はその単位領域で定義し、この領域で対象物を表現するものとする。通常の用法では、歪み座標系の単位領域の全域が主座標系の単位領域内に収まるように、歪み座標系を定義する。任意に歪ませた直交座標系のほか、極座標系など種々の直交曲線座標系の使用が考えられる。

「流れ座標系」とは、対象物の微細構造の流れを表現するための座標系であり、テクスチャ関数の入力座標ベクトルが属する座標系である。流れ座標系の上にテクスチャが固定されており、歪み座標系の単位領域を通して、このテクスチャを参照する。この歪み座標系上の参照領域が流れ座標系に対して相対的に移動することによってテクスチャがシフトされ、流れが表現される。このような用法のため、流れ座標系上のベクトル「流れ座標ベクトル」は、十分広い(理論的には無限大の)領域で定義される必要がある。図6に以上三つの座標系間の関係を示す。図6では、便宜上、歪み座標

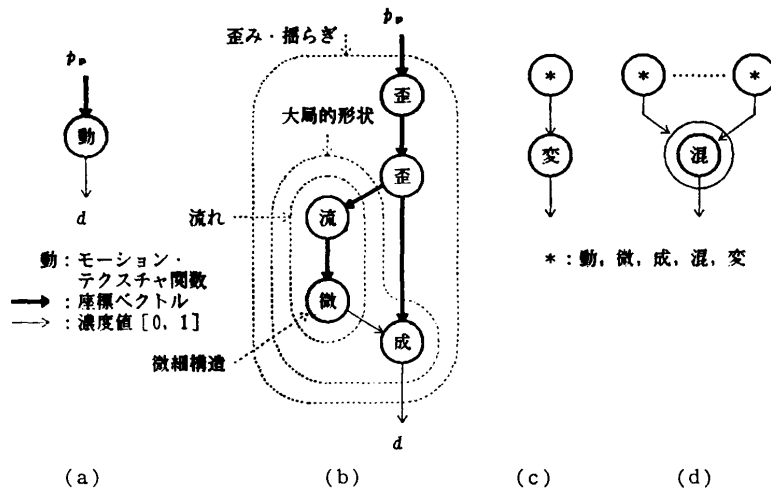


図5 モーション・テクスチャ関数の構成  
(a)モーション・テクスチャ関数, (b)基本構成例, (c)濃度変換関数の導入, (d)濃度値混合関数の導入.

Fig. 5 Construction of a motion-texture function.  
(a)A motion-texture function,  
(b)A typical construction,  
(c)Usage of a gray-scale transformation function,  
(d)Usage of a gray-level blending function.

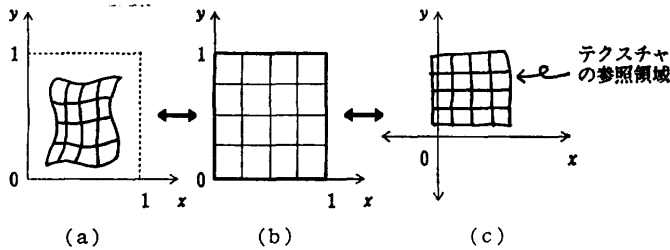


図 6 三つの座標系の階層関係

(a)主座標系, (b)歪み座標系, (c)流れ座標系.

Fig. 6 Hierarchy of the three coordinate systems.

(a)A primary coordinate system, (b)A warp coordinate system, (c)A flow coordinate system.

系の単位領域に対応する点が主座標系上および流れ座標系上で連続した領域を占めているが、実際には不連続に散在する場合も考えられる。

これらの座標系の中に、「歪み変換」および「流れ変換」の2種類の座標変換を考える。「歪み変換」は、

- 1) 主座標系 → 歪み座標系, および,
- 2) 歪み座標系 → 歪み座標系

の座標変換として定義する。後者は、歪み変換を複数段の変換で構成できることを示す。「流れ変換」は、

歪み座標系 → 流れ座標系

の座標変換として定義する。流れ変換が、流れを表現する平行移動以外に、歪み変換を補う形で歪み・揺らぎを表現するための座標変換を伴う場合も考えられる。

### 3.5 微細構造と大局形状

#### 3.5.1 テクスチャの供給

ディテールの素材となるテクスチャの濃度値を供給するテクスチャ関数は、入力の流れ座標ベクトルについて実際に無限の領域で定義されるならば問題はないが、例えば2次元配列を出力濃度値のテーブルとして参照するタイプのように、本質的には有限の領域でのみ定義されるテクスチャ関数には、「見掛け上の定義域」と「真の定義域」の二重の定義域を持たせる。見掛け上の定義域は無限の領域である。「真の定義域」は、有限である実際の定義域を単位領域に正規化したもので、この範囲から逸脱する座標値を循環的に扱って $[0, 1)$ の区間に収めることにより、見掛け上は無限の領域で定義されていると見なすことができる。このとき、循環させた座標値が、1から0へ、または0から1へ、不連続に変化する部分でも特徴が不連続にならないように、テクスチャを

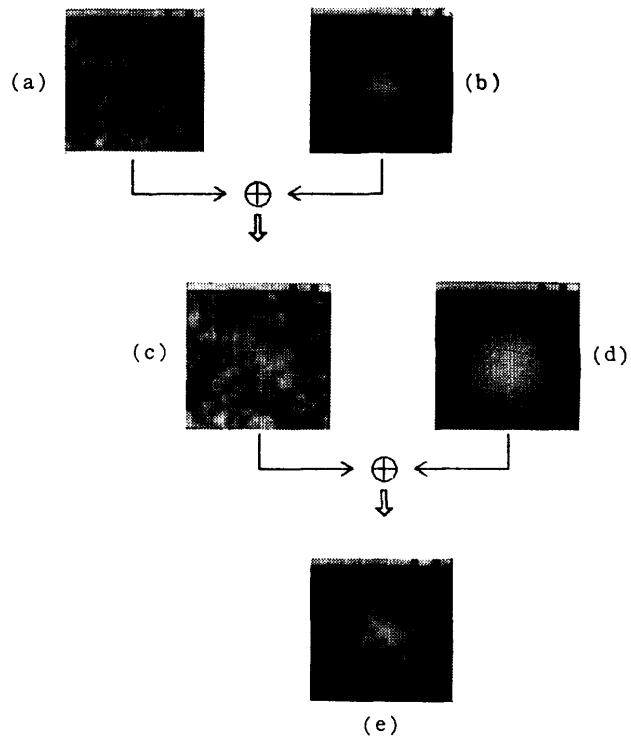
考慮する必要がある。

#### 3.5.2 成型処理

対象物の形状を決定するための、テクスチャに対する局所的濃度変換を「成型処理」と呼ぶ。成型処理は、以下に述べるように、微細構造の素材であるテクスチャと、対象物の大局的形状に寄与するパターンとを、濃度値混合演算で合成することによって行う。

雲状対象物の濃度値成分を「ベース成分」と「ディテール成分」の2種類の成分に分解して考える。第1の成分「ベース成分」は、

対象物の芯や核というべき部分を形成する成分であり、一般に大まかな分布を示す。第2の成分「ディテール成分」は、対象物の流動する微細構造を表す成分であり、一般にベース成分による核を覆うように分布する。ディテール成分はテクスチャによって供給する。ベース成分は、その分布パターンである「ベース



⊕: 濃度値混合演算

図 7 成型処理

(a)テクスチャ=ディテール成分, (b)ベースパターン=ベース成分, (c)中間生成像, (d)マスクパターン, (e)対象物.

Fig. 7 A forming process.

(a)Texture=detail component, (b)Base pattern=base component, (c)An Intermediate image, (d)Mask pattern, (e)An object.

パターン」によって供給する。成型処理の手順は、まずテクスチャとベースパターンとを合成し、次にこれと大局的形状を表すパターンである「マスクパターン」とを合成することにより対象物を形成する。図7に成型処理の例を示す。成型処理のための関数である成型処理関数の形式を式(2)に示す。

$$d_{out} = \text{shape}(d_{in}, p_w, t, C) \quad (2)$$

ここで、 $d_{in}$  が局所的濃度変換の対象となるテクスチャ関数からの濃度値である。マスクパターンとベースパターンは、式(3)に示すような形式の要素関数(以下、パターン関数)によって供給する。

$$d_{out} = \text{pattern}(p_w, t, C) \quad (3)$$

成型処理関数は、図8(b)に示すように二つのパターン関数(base, mask)と二つの濃度値混合関数(blend 1, blend 2)から構成される。baseとmaskは各々ベースパターンとマスクパターンの関数である。

図8(b)で1段目の濃度値混合関数blend 1は式(4)のような関数であり、テクスチャ濃度値 $d_{detail}$ (= $d_{in}$ )とベースパターン濃度値 $d_{base}$ とを合成して中間濃度値 $d_{temp}$ を得る。また関数blend 1は式(5)に示すような条件を満たすものとする。式(5)①②は $d_{base}$ に関する境界条件である。式(5)③④は、blend 1の $d_{detail}$ に関する微分係数 $\text{blend } 1a'$ と $d_{base}$ に関する微分係数 $\text{blend } 1b'$ が共に0以上で、blend 1が $d_{base}$ と $d_{detail}$ に関して単調増加であることを示す。また、式(5)⑤により、blend 1の出力は常に $d_{base}$

以上である。

$$d_{temp} = \text{blend } 1(d_{detail}, d_{base}, t, C) \quad (4)$$

$$d_{temp} = \left\{ \begin{array}{l} d_{detail}, d_{base} = 0 \quad \textcircled{1} \\ 1, d_{base} = 1 \quad \textcircled{2} \\ \text{blend } 1a' \geq 0 \quad \textcircled{3} \\ \text{blend } 1b' \geq 0 \quad \textcircled{4} \\ d_{temp} \geq d_{base} \quad \textcircled{5} \end{array} \right. \quad (5)$$

また、2段目の濃度値混合関数blend 2は、式(6)のような関数であり、中間濃度値 $d_{temp}$ とマスクパターン濃度値 $d_{mask}$ とを合成して成型処理関数全体としての出力 $d_{out}$ を得る。blend 2は式(7)に示すような条件を満たすものとする。式(7)①②は $d_{mask}$ に関する境界条件である。式(7)③④は、blend 2の $d_{temp}$ に関する微分係数 $\text{blend } 2a'$ と $d_{mask}$ に関する微分係数 $\text{blend } 2m'$ が共に0以上で、blend 2が $d_{temp}$ と $d_{mask}$ に関して単調増加であることを示す。 $d_{mask}$ が0のとき $d_{out}$ は0であり、 $d_{mask}$ が大きくなるにしたがい $d_{temp}$ が $d_{out}$ に寄与する割合が大きくなる。

$$d_{out} = \text{blend } 2(d_{temp}, d_{mask}, t, C) \quad (6)$$

$$d_{out} = \left\{ \begin{array}{l} 0, d_{mask} = 0 \quad \textcircled{1} \\ d_{temp}, d_{mask} = 1 \quad \textcircled{2} \\ \text{blend } 2a' \geq 0 \quad \textcircled{3} \\ \text{blend } 2m' \geq 0 \quad \textcircled{4} \end{array} \right. \quad (7)$$

### 3.6 補助操作

補助操作には、濃度階調変換と濃度値混合演算とがある。映像全体の濃淡の調子を変えるために濃度階調変換を行う。濃度階調変換の効果は濃度変換曲線で表される。その傾きが1より大きい区間では濃度変化の範囲が引き伸ばされ、コントラストが強調される。極端な場合として、出力濃度値が不連続に変化する部分では偽の輪郭が得られる。逆に、傾きが1未満の区間では濃度変化の範囲が圧縮され、コントラストが抑えられる。特に傾きが0の区間では濃度変化が無視される。また、傾きが負の区間では濃淡変化が逆転する。

濃度値混合演算は、前述の成型処理の内部で使用したように、複数の中間生成濃度値をひとつにまとめるために用いる。

## 4. 画像データの生成

モーション・テクスチャの最も代表的な使用法は、これを色彩情報として利用することである。このためには、モーション・テクスチャ関数の出力側に、赤・緑・青の各成分に対応する三つの濃度変換関数から構

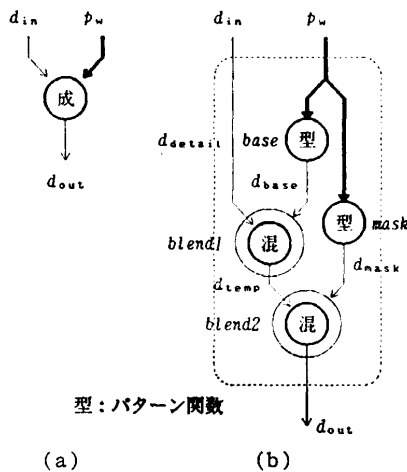


図8 成型処理関数の構成  
(a)成型処理関数, (b)内部構成。

Fig. 8 Construction of a forming function.  
(a) A forming function, (b) The internal construction.

成される色ベクトル関数を付加して、出力濃度値を色ベクトルに変換する。

実際に画像データを生成する処理段階（以下、画像生成パス）では、モーション・テクスチャ関数をその定義域内の点に対応するすべての画素について評価する。画像生成パスで得た画像データに対し、重ねて1回以上の画像変換処理を施すことにより、さらに様々な効果を付加することができる。この段階（以下、画像変換パス）では、画像生成パスと異なり、変換の対象となるすべての画素が処理の段階ですでに確定しているため、空間フィルタリングなどのような近傍処理・大局処理の画像変換技術を利用することができる。

2次元モーション・テクスチャを用いて実際に映像を制作する場合、モーション・テクスチャそのもので映像を完結させる用法と、モーション・テクスチャを情景の中に組み込む用法とが考えられる。情景の中に組み込む方法の一つとして、これを物体表面上にその属性データとしてマッピングすることがあげられる。また、2次元モーション・テクスチャの2次元的分布を持つスカラ値を「標高値」として使用することによって、3次元の形状データを得ることができる。例えば、雲の厚さとして用いて、3次元の雲の形状データを作成することが考えられる。

## 5. 炎の表現例

2次元テクスチャ法による炎の動的表現例を示す。炎のためのモーション・テクスチャ関数の構成を図9に示す。モーション・テクスチャ関数の決定は、多分

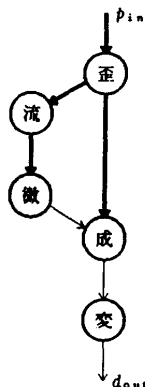


図9 炎のためのモーション・テクスチャ関数の構成

Fig. 9 Construction of a motion texture function for flames.

に直観的な方法による。まず、表現しようとする対象物のおおまかなデザイン・スケッチから、雲状対象物の基本的特徴として分類した、1) 歪み・揺らぎ、2) 流れ、3) 大局的形状、4) 微細構造を、視察によって直観的に抽出し、各要素に対応する基本操作関数を個々に構成する。そして、全体として満足のいく映像が得られるように試行錯誤を繰り返しながら各基本操作関数に手を加えていく。以下、各段階における処理について簡単に説明する。

### 5.1 テクスチャ

素材とするテクスチャは、一般に「中点変位法」として知られるアルゴリズム (recursive subdivision algorithm)<sup>2)</sup> によって生成する。中点変位法は、フラクタル曲線・曲面を生成する効率的なアルゴリズムである。フラクタル<sup>23)</sup>は、幅広い範囲のスケールでの統計的自己相似性を特徴とする複雑な形状のクラスである。多くの自然対象物が何らかのかたちでフラクタルとして扱えることが確認されている<sup>23),24)</sup> ので、この特徴を持つテクスチャは多くの対象物の表現に適用でき、表現の素材としての用途に適する。

フラクタル・テクスチャを生成する手法としては、フーリエ展開によるものもある<sup>5)</sup> が、演算速度の遅い環境では、三角関数を使わず幾つかの内部パラメータを除き四則演算でデータを生成できる中点変位法が便利である。しかし中点変位法では有限の配列内にデータを生成するので、3.5.1項で述べたような、循環的使用における境界での不連続性の問題が発生する。このため、オリジナルのアルゴリズム<sup>2)</sup>を若干修正する。これは、データ生成の際、配列要素を参照するときに配列の行と列の添え字を各々循環させて扱うことにより解決できる<sup>21)</sup>。中点変位法によるテクスチャの例を図10に示す。関数の形式では、あらかじめ中点変位法によって配列に画像データを生成し、この配列を参照して出力濃度値を決定する。入力時刻  $t$  はダミーとなる。この1枚の静止画像データから動画系列であるモーション・テクスチャを生成する。



図10 中点変位法によるテクスチャの例

Fig. 10 An example of texture generated by a recursive subdivision algorithm.



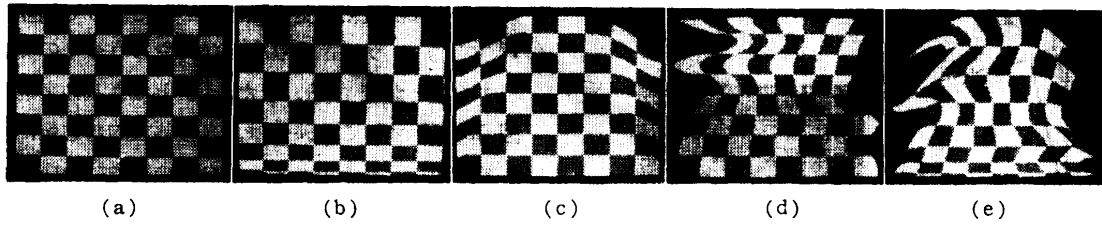


図 11 「炎」の歪み変換

(a)チェック・パターン, (b)第1段, (c)第2段, (d)第3段, (e)歪み変換の結果.

Fig. 11 Warp transformations for flames.

(a)A check-pattern, (b)The first stage, (c)The second stage, (d)The third stage, (e)The result of the warp transformations.

## 5.2 歪み変換・流れ変換

歪み変数は3段階から成る。主座標系上に写像された歪み座標系の単位領域の様子を図11に示す。

図11(a)は歪みのない状態である。まず炎が加速しつつ上昇する様子を表現するために図11(b)のように $y$ 座標を置き換える。次に図11(c)のように左右の端に向かうほど $y$ 軸方向の尺度を圧縮して上昇速度の変化を表現する。これは、次の段階の揺らぎと共に、境界付近の流れの乱れを表現する。最後に、炎に揺らぎを与えるために、図11(d)に示すように下から上に向かうにしたがって振幅が増大する波を、主座標系上での歪み座標系単位領域の左右の境界とし、領域内部の点の $x$ 座標を $[0, 1]$ に正規化する。実験では、左右の境界として周期が同じで位相が独立に変化する二つの正弦波を使用し、内部の点の正規化にBézier補間関数を用いた。主座標系上の各水平断面が歪み座標系単位領域の左右の境界(前述の正弦波)を切る2点と、中心軸を切る1点とを制御点としたBézier補間関数のパラメータを逆算して、これを新しい $x$ 座標値として置き換える。これは、frozen-edge warp<sup>25)</sup>という変形操作に分類されるもので、あらかじめ設定した境界(ここでは振幅・位相を制御した正弦波)に囲まれた領域(主座標系上の歪み座標系単位領域)に歪みを与える処理を行う。以上3段階の歪み変換の結果を図11(e)に示す。

また、炎の流れは上向きすなわち $y$ 軸正方向であるので、歪み座標の $y$ 座標値に時刻 $t$ に関する単調減少関数の値を加えることによって流れ変換とする。

流れに伴う揺らぎの上昇を再現するために、前述の正弦関数の初期位相を、流れ変換に用いたものと同じ時刻 $t$ に関する単調減少関数によって与える。これにより、揺らぎと流れとの同期をとることができる。

## 5.3 成型処理・濃度補正・彩色

成型処理の過程を図12に示す。図中の記号は式

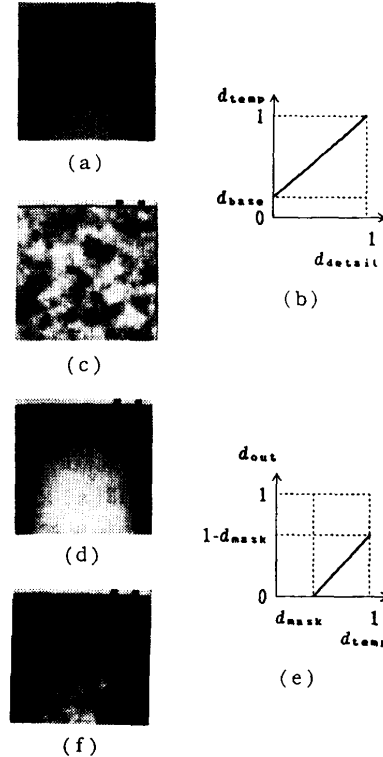


図 12 「炎」の成型処理

(a)ベースパターン, (b)第1濃度値混合, (c)中間生成画像, (d)マスクパターン, (e)第2濃度値混合, (f)成型処理の結果.

Fig. 12 Forming processes for flames.

(a)A base pattern, (b)The first blending, (c)An intermediate image, (d)A mask pattern, (e)The second blending, (f)The result of the forming process.

(4)(6)に準ずる。ベースパターンは図12(a)に示すもので、図12(b)のような変換曲線によってテクスチャと合成し、図12(c)の中間生成画像を得る。そして、遊離した炎の断片が上昇するにつれ次第に消えていく様子を再現するために、図12(d)に示すマスクパターンと図12(e)のような変換曲線によって

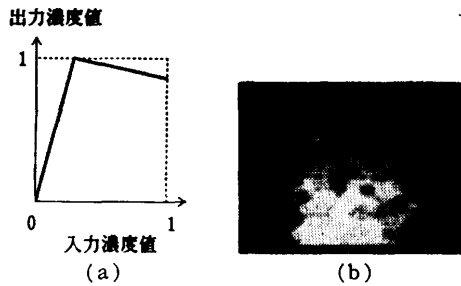


図 13 「炎」の濃度変換  
(a)変換曲線, (b)変換後の画像.

Fig. 13 A gray-scale transformation for flames.  
(a) A transformational curve, (b) A transformed image.

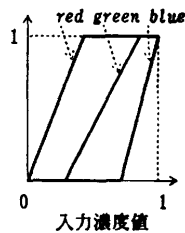


図 14 「炎」の色ベクトル関数

Fig. 14 A color-vector function for flames.

合成し、図 12(f) を得る。

以上の処理で得られる映像は曖昧な断片からなっている。ところが実際の炎の断片では比較的くっきりした境界が観察される。そこで、図 13(a) に示すような濃度階調変換によって炎の断片の外周部を強調する。図 13(b) に図 12(f) を変換した画像を示す。

身近に見られる焚き火やろうそくなどの黄色い炎は、熱せられた炎中の炭素微粒子が黒体放射によって輝いて見えるものである。その温度と密度各々の 3 次元の分布によって決定される色・明るさを 2 次元のモデルによる濃度値のみから求めるのは不可能であるので、ここでは黒体放射にこだわらず図 14 のように赤から黄白色へと感覚的に配色したが、比較的良好な結果を得ている。

#### 5.4 結果と考察

以上のようにして制作した炎の映像を図 15 に示す。直径 30 cm 程度の炎で、20 フレーム毎秒の再生速度を想定している。画素数は 128×128、計算時間は 4 MIPS の計算機で 1 フレーム約 20 秒である。フレーム・バッファの記憶容量の制限から、16 フレームごとに周期性を持つシーケンスにした。動画の再生は、フレーム・バッファのハードウェア機能を用いて表示画像をリアルタイムで切り換えることにより行った。

基本操作関数に変更を加えることによって、図 16・17 に示すように、より小さな炎やろうそくの炎を表現することができる。また、「ひとだま」の表現例を図 18 に示す。この映像は、専ら個人的な想像に依存して制作したものである。他の炎の映像についても、現実に忠実な再現というよりは、個人的イメージが強く反映されたものになっている。

実際の炎の写真を図 19・20 に示す。今回客観的な評価の基準を設定することはできず、数名の被験者に実写のビデオと本手法によって制作したビデオとを視てもらい感想を聞くという主観的な評価を行った。その結果、16 コマという短い周期性があり実物の映像との違いは比較によって容易に分かるにもかかわらず、概して十分にリアルな映像であるという評価が得られた。また、前もって何の説明もせず、合成映像のみを提示した場合には、CG であることが分からなかった被験者もあった。「ひとだま」は想像によって制作したにもかかわらず、制作者の意図を無理なく理解してもらえた。

## 6. おわりに

本稿では、従来のテクスチャを時間軸方向に拡張した、動きを伴ったテクスチャである「モーション・テクスチャ」の概念を提案し、2次元テクスチャを適宜加工して2次元モーション・テクスチャをデザインすることにより雲状対象物の流れを動的に表現する「2次元テクスチャ法」について述べた。また、本手法による炎の動的表現の例を示した。間近に見る焚き火のような燃えさかる炎をリアルに表現できることが本手法の目標の一つであるが、実験の結果、この目標をある程度達成できたことを確認した。他の雲状対象物の表現についても今後実験を重ねていく予定である。

本手法では、映像の感覚的なデザインを容易にするため、雲状対象物を視覚的な特徴から2次元的にモデル化した。これにより、表現しようとする対象物の大まかなデザインまでは実際に容易に行えることを確認したが、その一方で、ある程度以上のリアリティを求

図 15 「炎」  
Fig. 15 A sequence of flames.

図 16 「小さな炎」  
Fig. 16 A sequence of small flames.

図 17 「ろうそくの火」  
Fig. 17 A sequence of a candle light.

図 18 「ひとだま」  
Fig. 18 A sequence of a will-o'-the-wisp.

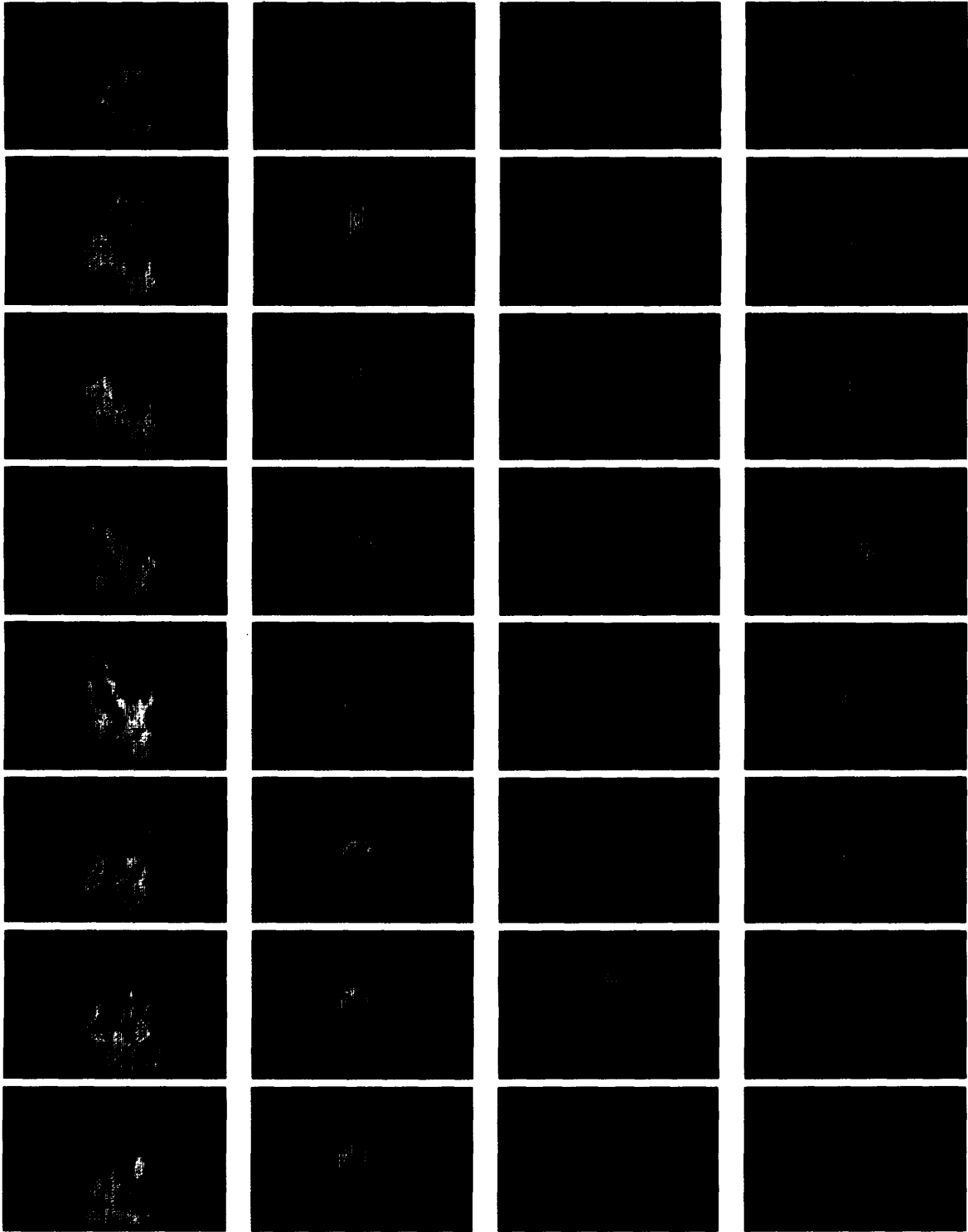


図 15  
Fig. 15

図 16  
Fig. 16

図 17  
Fig. 17

図 18  
Fig. 18

めると、満足のいく映像が得られるまでに必要な試行錯誤の回数が急に増大するという本質的な難点を持っている。これを補うためにも、本手法のインプリメンテーションにおいては、関数のライブラリ管理などを含めて、対話操作を容易に行えるようなユーザ・インタフェースを重視する必要がある。

2次元モーション・テクスチャの3次元情景への組み込みについては第4章で触れたが、本質的な対応策として、3次元モーション・テクスチャへの拡張を現在検討中である。

**謝辞** 燃焼実験の際にたいへん御世話になりました日本消防検定協会の遠藤国昭氏に深甚なる謝意を表します。また、日頃様々な面で協力して頂いている研究室の諸氏に感謝します。

### 参 考 文 献

- 1) 第3回グラフィクスとCAD集中研究集会：自然対象物のモデリングと表現技術，情報処理学会研究会報告，89-CG-40 (1989).
- 2) Fournier, A., Fussell, D. and Carpenter, L. C.: Computer Rendering of Stochastic Models, *Comm. ACM*, Vol. 25, No. 6, pp. 371-384 (1982).
- 3) Blinn, J. F.: Light Reflection Functions for Simulation of Clouds and Dusty Surfaces, *Comput. Gr.*, Vol. 16, No. 3, pp. 21-29 (1982).
- 4) Kajiya, J. T. and Von Herzen, B. P.: Ray Tracing Volume Densities, *Comput. Gr.*, Vol. 18, No. 3, pp. 165-174 (1984).
- 5) Gardner, G. Y.: Visual Simulation of Clouds, *Comput. Gr.*, Vol. 19, No. 3, pp. 297-303 (1985).
- 6) 野村, 横井, 鳥脇: 霧状物体の表示アルゴリズムに関する検討, NICOGRAPH '86 論文集, pp. 126-135 (1986).
- 7) 西田, 中前: 大気中の不均質微粒子の表示法, 第35回情報処理学会全国大会論文集, 2G-4, pp. 2307-2308 (1987).
- 8) 宮田: 等濃度線を用いた雲画像の生成法, 情報処理学会グラフィクスとCAD研究会報告, 29-6 (1987).
- 9) Reeves, W. T.: Particle Systems—A Technique for Modeling a Class of Fuzzy Objects, *Comput. Gr.*, Vol. 17, No. 3, pp. 359-376 (1983).
- 10) Perlin, K.: An Image Synthesizer, *Comput. Gr.*, Vol. 19, No. 3, pp. 287-296 (1985).
- 11) 河野, 中嶋, 安居院: セルオートマトンによる火炎画像の生成, NICOGRAPH '89 論文集, pp. 79-85 (1989).
- 12) 千葉, 村岡, 中川: 水や煙の流れのシミュレーション, 情報処理学会グラフィクスとCAD研究会報告, 40-5 (1989).
- 13) Catmull, E.: Computer Display of Curved Surfaces, *Proc. IEEE Conference of Computer Graphics, Pattern Recognition, and Data Structures* (1975).
- 14) Blinn, J. F. and Newell, M. E.: Texture and Reflection in Computer Generated Images, *Comm. ACM*, Vol. 19, No. 10, pp. 542-547 (1976).
- 15) Blinn, J. F.: Simulation of Wrinkled Surfaces, *Comput. Gr.*, Vol. 12, No. 3, pp. 286-292 (1978).
- 16) Cook, R. L.: Shade Trees, *Comput. Gr.*, Vol. 18, No. 3, pp. 223-231 (1984).
- 17) Peachey, D. R.: Solid Texturing of Complex Surfaces, *Comput. Gr.*, Vol. 19, No. 3, pp. 279-286 (1985).
- 18) 大島, 板橋: 2次元テクスチャを用いた炎の簡易表示の試み, 情報処理学会グラフィクスとCAD研究会報告, 30-5 (1987).
- 19) 大島, 板橋: テクスチャを利用した炎の簡易表示の試み, 第36回情報処理学会全国大会論文集, 7Z-2, pp. 2161-2162 (1988).
- 20) 大島, 板橋: テクスチャを素材にしたアニメーションの制作, 情報処理学会グラフィクスとCAD研究会報告, 35-6 (1988).
- 21) 大島, 板橋: テクスチャを利用したアニメーションの制作, NICOGRAPH '88 論文集, pp. 110-118 (1988).
- 22) Ohshima, T. and Itahashi, S.: An Animation Design Tool Utilizing Texture, *Proc. IEEE MIV-89*, pp. 337-342, Tokyo, Japan (1989).
- 23) Mandelbrot, B. B.: *The Fractal Geometry of Nature*, p. 464, W. H. Freeman & Co., San Francisco (1982).
- 24) 高安: フラクタル, p. 186, 朝倉書店, 東京 (1986).
- 25) Smith, A. R.: Planar 2-pass Texture Mapping and Warping, *Comput. Gr.*, Vol. 21, No. 4, pp. 263-272 (1987).

(平成2年4月20日受付)

(平成3年1月11日採録)

**大島登志一 (正会員)**

1963年生。1986年筑波大学第三学群情報学類卒業。1988年筑波大学大学院修士課程理工学研究科修了。現在同大学院博士課程工学研究科在籍。コンピュータグラフィックスの研究に従事。ACM-SIGGRAPH, コンピュータ・グラフィックス学会各会員。

**板橋 秀一 (正会員)**

昭和39年東北大学工学部通信工学科卒業。昭和45年同大学院(博)電気及通信工学専攻退学。同年東北大学電気通信研究所助手。昭和47年電子技術総合研究所入所。昭和49年同所主任研究官。昭和52年ストックホルム王立工科大学客員研究員。昭和57年筑波大学電子・情報工学系助教授。現在同教授。工学博士。音声・画像・自然言語処理の研究に従事。電子情報通信学会, 日本音響学会, 人工知能学会, 日本認知科学会, IEEE, アメリカ音響学会各会員。