

2009年度 卒業論文

リアルタイム3DCGにおける
昼花火のシミュレーション手法

指導教員：渡辺 大地 講師

メディア学部 ゲームサイエンス プロジェクト
学籍番号 M0105075
大島 健太

2009年度 卒業論文概要

論文題目

リアルタイム3DCGにおける
昼花火のシミュレーション手法

メディア学部

学籍番号：M0105075

氏名

大島 健太

**指導
教員**

渡辺 大地 講師

キーワード

リアルタイム3DCG、リアルタイムシミュレーション、
花火、昼花火、ポイントスプライト

近年、3DCGの分野では様々な自然現象、物理現象をリアルタイムに表現しようとする研究が数多く行われてきている。煙などの不定形な形状の表現も3DCGによる自然、物理現象のリアルタイムな表現として多くの研究が行われている。しかし、既存の研究では、狭い領域の煙を非常に高解像度で表現しているため、広い領域を高解像度で表現する必要のある煙に対して応用する事はむずかしい。そのような特徴を持つものの1つとして、昼花火の煙がある。昼花火の煙の形状は内部の星と呼ばれる、煙を噴出する火薬の塊を上空で放出し、星の飛翔によって煙の形状を作成する。その結果、煙は螺旋のような独特な形状を広範囲に渡り描く。この螺旋のような形状を表現するには高い解像度で煙を表現する必要がある。そのため、既存の煙の表現手法では表現する事は出来ない。また、煙のような不定形な形状を持つ自然現象として、雲が挙げられる。しかし、昼花火の煙の色や形状は、星の作成方法と上空での挙動で制御するものであるため、これらの研究では昼花火を表現する事は出来ない。

そこで、本研究では3DCGにより昼花火の煙の形状と色の変化をリアルタイムに表現する事が出来る手法を提案する。これによって、3DCGによるリアルタイムな煙の表現の幅を広げる事が出来る。本手法では、昼花火の中でも、煙柳という種類の昼花火を表現する。この煙柳の煙の形状を作成するため、煙を粒子の集合と捉えパーティクルアニメーションにより煙の形状を表現した。そして、各粒子の座標にポイントスプライトによりテクスチャをマッピングした。更に、マッピングするテクスチャをプログラム内で生成した。本手法により、煙柳の煙の色と濃度を柔軟に制御する事が可能になり、昼花火に独特な螺旋を描くような煙の形状も表現する事が出来た。

目次

第1章	はじめに	1
1.1	研究の背景と目的	1
1.2	論文構成	3
第2章	昼花火について	4
2.1	昼花火の概要	4
2.2	昼花火の種類	6
2.3	煙柳について	7
2.4	昼花火の玉や星の挙動	9
第3章	シミュレーション手法	11
3.1	星の拡散	11
3.2	星の移動	12
3.3	煙の生成	12
3.4	煙の移動	13
3.5	煙のレンダリング	14
第4章	検証と考察	17
4.1	シミュレーション環境	17
4.2	シミュレーション結果	18
4.3	実写画像との比較	23
4.4	リアルタイム性の検証	24
第5章	まとめ	26
	謝辞	28
	参考文献	29

目 次

1.1	昼花火の写真	2
2.1	代表的な昼花火	5
2.2	代表的な夜花火	5
2.3	煙柳の画像	7
2.4	煙柳の構造	8
2.5	星の構造	9
2.6	昼花火の玉や星の軌跡	10
3.1	煙粒子の噴出する様子	13
3.2	本手法によって生成した画像	14
3.3	生成された1枚のテクスチャの画像	16
4.1	昼花火のシミュレーション結果画面	19
4.2	テクスチャをマッピングする大きさを制御する係数 k を変化させた 画像	20
4.3	ポリゴンの透明度を制御する係数 a を変化させた画像	20
4.4	テクスチャの初期色を様々に変更した画像	21
4.5	星ごとに異なる色の煙を噴出した画像	22
4.6	本手法と実際の昼花火の比較	23

表 目 次

4.1	実行環境	17
4.2	計算パラメーター	17
4.3	レンダリングパラメーター	18
4.4	星ごとの初期色	22
4.5	シミュレーションデータ	25
4.6	区間ごとの平均処理時間	25

第 1 章

はじめに

1.1 研究の背景と目的

近年、パーソナルコンピュータの性能の向上により、自然現象、物理現象を高速にシミュレーションする事が可能になってきている。そのため、3DCG の分野では様々な自然現象、物理現象をリアルタイムに表現しようとする研究が数多く行われてきている。煙などの不定形な形状の表現も 3DCG による自然、物理現象のリアルタイムな表現として多くの研究が行われている。補完レイマーチング法を用いた手法 [1] や、パーティクルシステムと PerlinNoise を用いた手法 [2] などである。

しかし、これらの研究では、狭い領域の煙を非常に高解像度で表現しているため、広い領域を高解像度で表現する必要のある煙に対して応用する事はむずかしい。そのような特徴を持つものの 1 つとして、昼花火の煙がある。図 1.1 が昼花火の写真である。昼花火の煙の形状は内部の星と呼ばれる、煙を噴出する火薬の塊を上空で放出し、星の飛翔によって煙の形状を作成する。その結果、煙は螺旋のような独特な形状を広範囲に渡り描く。この螺旋のような形状を表現するには高い解像度で煙を表現する必要がある。そのため、既存の煙の表現手法では表現する事は出来ない。

また、煙のような不定形な形状を持つ自然現象として、雲が挙げられる。リアルタイムな雲の表現にも様々な手法があり、インポスターと呼ばれる手法を用い



図 1.1: 昼花火の写真

て形状を表現する手法 [3] や、3次元テクスチャを用いた手法 [4] などがある。

しかし、昼花火の煙の色や形状は、星の作成方法と上空での挙動で制御するものであるため、これらの研究では表現する事は出来ない。

過去に3DCGで昼花火を表現した研究は存在しないが、夜に打ち上げる花火や、手で持って遊ぶ玩具花火を3DCGで表現した研究がいくつか存在する。小玉ら [5][6] は3DCGを用いて花火の挙動をリアルタイムに表現する研究を行った。これにより、現在の花火は企画から製造まで全て花火職人が行っており、花火を実際に作成する事が出来ない、一般のデザイナーや演出家などの花火大会への参入が困難な状況にあり、また試作、試打にも莫大なコストがかかる [7][8] という問題を解決

し、花火の設計の支援を行った。また、ゲームやCGへの利用を目的とする線香花火の燃焼過程をシミュレーションする研究 [9][10] や、花火大会の演出を支援する為の3DCGによる花火演出システムの開発を行っている研究 [11] などがある。しかし、これらの研究では、花火の閃光を主要な表現として扱っており、昼花火のような煙からなる花火の形状を表現する事はできない。

本研究では、昼花火の中でも、煙柳という種類の昼花火を表現した。実際の煙柳の形状は、花火内部の星という火薬の塊が作成する。煙柳を打ち上げた後、上空で花火は多数の星を放出し、これらの星は煙を噴出しながら空気中を高速に回転移動する。回転を伴い高速に移動しながら、空気中で煙を生成する事により、星は煙柳の特徴である、螺旋のような独特な煙の形状を上空で描く。

この煙柳の煙の形状を作成するため、まず星の挙動を斜方投射運動としてシミュレーションし、その星のある一定の面から煙を法線方向に噴出するものとした。煙を噴出させる際に煙を粒子の集合と捉え、パーティクルアニメーションにより煙の形状を表現した。そして、各粒子の座標にポイントスプライトによりテクスチャをマッピングした。更に、マッピングするテクスチャはプログラム内で生成した。この手法によって煙柳の煙の色を柔軟に制御する事が可能になり、煙柳に独特な螺旋を描くような煙の形状も表現する事が出来た。よって、本研究では3DCGにより煙柳の煙の形状と色の変化をリアルタイムに表現する事が出来たと言える。

1.2 論文構成

本論文の構成は以下の通りである。第2章では昼花火の特徴を説明し、第3章では第2章で示した昼花火の特徴を踏まえ、どのようにシミュレーションするかを述べる。第4章では第3章で述べたシミュレーション手法からシミュレーションした例を示し、その結果と考察について述べる。第5章では本研究のまとめを述べる。

第 2 章

昼花火について

本研究では特徴的な打ち上げ花火として、昼花火という花火に注目する。我々が普段夜に目にする打ち上げ花火のすべては「夜物」「夜花火」という分類であるが、これに対して昼花火は「昼物」「昼花火」という分類である。例えば、運動会や催し物の開催を告げる大きな音を発する音花火を「雷」といい、これも昼花火の一種である。夜花火が色のついた光を軸に形状を構成している事に対し、昼花火は色のついた煙によって形状を構成している [12]。以下に詳細に昼花火について説明する。

2.1 昼花火の概要

打ち上げ花火は「音、煙、光」の3要素で構成するが、昼花火と夜花火では主となる要素が異なる。夜花火は光で形を表現しているのに対し、昼花火は煙で形を表現している。これが昼花火と夜花火では大きく異なる点である。図 2.1 が昼花火の一例 [13] であり、図 2.2 が夜花火の一例 [14] である。



图 2.1: 代表的な昼花火



图 2.2: 代表的な夜花火

昼花火は夜花火と比べ、目にする数が少ない。公式な花火大会としては世界でも大曲花火競技大会でしか見ることはできない [15]。しかし、式典等では目にすることもあり、例えば、2008年に中国の花火生産で有名な江西省ではオリンピック開催にちなんで長さ2008メートルの昼花火を打ち上げた。これは世界最長の「昼花火」としてギネスブックに登録された [16]。

2.2 昼花火の種類

昼花火は打ち上げに用いる「玉」とその内部に多数配置する「星」や「雷」から構成する。「星」とは発色剤、発煙材等を練って作る火薬の塊であり、「雷」とは大きな音を出すために用いる発音材である。「玉」はボール紙をプレスして作る半球状の玉皮2枚を張り合わせて作成し、内部に「星」や「雷」を多数配置することで上空で内部の「星」や「雷」をばらまき、煙や光や音を出す。「玉」の内部の「星」と「雷」の配置により、様々な形が表現できるのが昼花火の特徴である。

昼花火は見た目では分けて、

1. 雷
2. 煙菊
3. 煙柳
4. 煙竜
5. 旗物

の5種類に大きく分けることができる。雷とは、強い音を発する昼花火であり、運動会や催し物の開催を知らせる場合に打ち上げる。煙菊とは、煙の形状が菊型を描く昼花火であり、煙柳は柳のように下方方向に垂れ下がった形状を描く昼花火である。煙竜と旗物は双方玉の中にパラシュートを入れるという特徴は同じだが、

パラシュートの先に発煙筒を付けたものを煙竜、万国旗や長い旗などを吊るしたものを旗物と呼ぶ。

昼花火はこの中から1つ、あるいは複数を組み合わせて花火を制作する事になる。

本研究では 昼花火の中でも、特に昼花火の形状の特徴が顕著な煙柳を対象とする。そのため、煙柳について別節で詳細に説明する。

2.3 煙柳について

花火玉の中に星と言う、煙を噴出する火薬の塊を配置し、その玉が上空で破裂することによって内部の星が飛翔し、樹木の柳のような 垂れ下がった形状の煙を生み出す昼花火を「煙柳」と言う。日中に打ち上げる煙柳には、単色の煙のみを用いるもの以外に、紅・黄・青・紫などの複数の色煙を組み合わせて柳の枝を表現するものもあり、その場合には「彩煙柳」と呼ばれる。図 2.3 が煙柳の画像である。



図 2.3: 煙柳の画像

煙柳の構造は、玉の内部に割薬という小さな火薬を配置し、その周囲に星を配置する。そして内部で星がずれないように詰め物をいれて固定する。一般的な煙柳の構造を表しているのが図 2.4 である。割薬とは、玉を割り星を放出するため

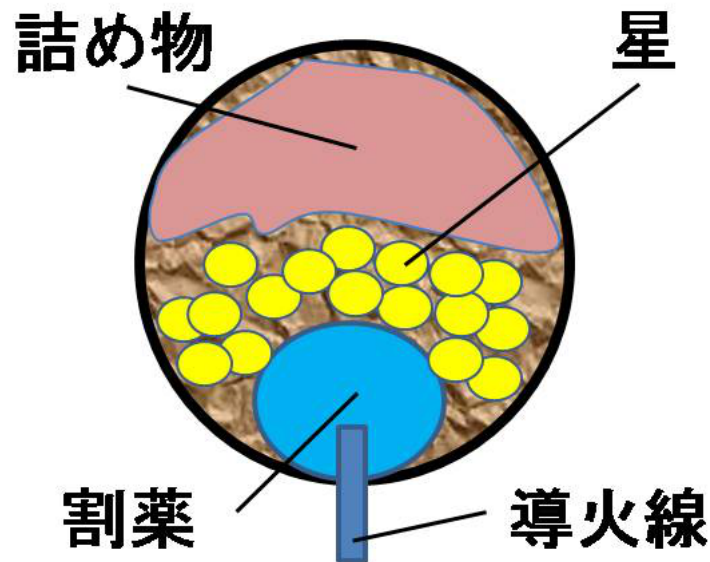


図 2.4: 煙柳の構造

の火薬であり多すぎても 少なすぎても綺麗な花火にならない。詰め物は内部の星の位置がずれないように詰める。玉の外から内部までは導火線が差し込んであり、そしてその玉の内部から外まで導火線が差し込んである。この導火線は打ち上げの際の火薬の燃焼により燃焼を始め、上空で内部の割薬に引火し、内部の星を放出すると同時に星が燃焼し、煙が発生する。これにより、煙柳としての形状を作成する。

図 2.5 が昼花火で重要な役割を果たす、星の構造である。この星の構造には 2 種類あり、1 つは発煙薬をつめた紙筒の両端を石膏で密封し、石膏部分に噴煙孔という孔を空け、内部に薬紙という紙を詰めた構造であり、図 2.5 の右側の構造に当たる。薬紙が燃焼する事によって内部の発煙薬に点火し、噴煙孔から煙を噴出する。もう 1 つは発煙剤に水を加えてよく混合して固め、その表面に早火という着火薬

の層を作る構造であり、図 2.5 の左側の構造に当たる。発煙剤の燃焼反応によって生成された煙の色素は空気中の酸素と反応して燃焼してしまい、色を発さない場合があるため、どちらの構造も燃焼を防ぐ仕組みになっている。

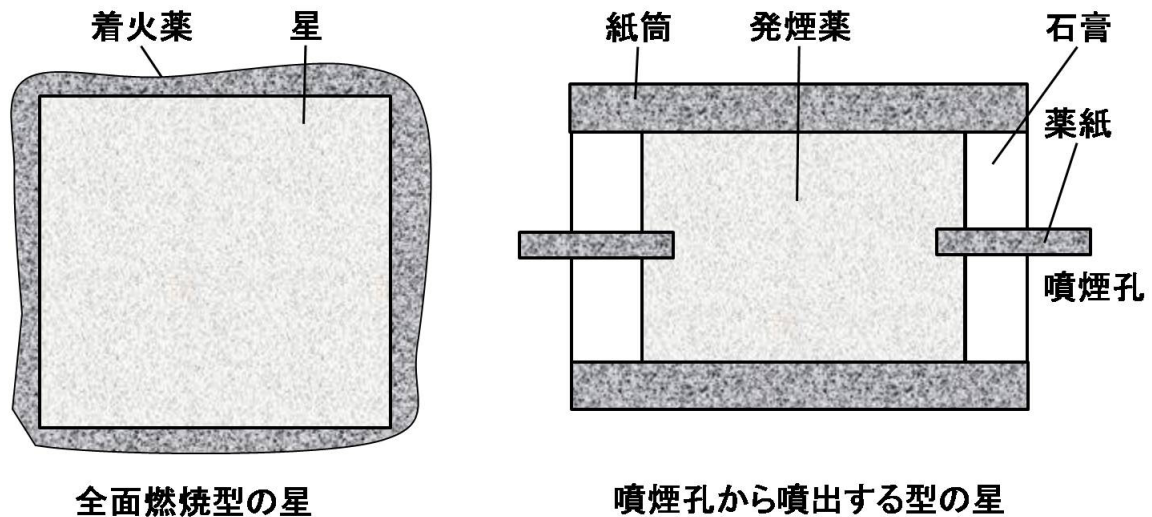


図 2.5: 星の構造

2.4 昼花火の玉や星の挙動

昼花火の玉や星の描く軌跡は直線ではなく、複雑な曲線を描く。玉は打ち上げ時に、星は玉の爆発時に火薬の燃焼による空気の圧力がかかり、回転する。回転を行いながら空気中を移動すると、玉や星には回転に伴うマグヌス効果 [17] による力が加わる。マグヌス効果とは、野球のピッチャーが投げるカーブのボールのように、回転する球や円筒が空気中を一定速度で移動する時、その移動方向に対して垂直に大きな揚力（トルク）が働くという効果である。玉や星は燃焼によって重心が変化することにより回転軸が変化し、そこにマグヌス効果が加わることで複雑な回転と軌跡を生み出すのである。玉はその後方に様々な仕掛けを取り付けて打ち上げることが多いため、回転数が少なくなり、星よりも玉の方が動きの

複雑さが小さいという特徴がある。図 2.6 は昼花火の玉や星の軌跡の特徴をよく表す図である。



図 2.6: 昼花火の玉や星の軌跡

第 3 章

シミュレーション手法

本研究では昼花火の中でも、煙柳と呼ぶ種類の昼花火を対象とする。本研究では以下の手順で煙柳を表現する。

1. 花火の玉から複数の星を放出
2. 星を斜方投射運動すると同時に、星から煙を噴出
3. 星から噴出した煙を移動
4. 煙をレンダリング

以下の節で詳細を述べる。また、3D の座標系は画面上方向を y 軸の正の方向とし、右方向を x 軸の正方向とし、奥を z の正の方向とする左手系の座標系を用い、物理量の単位には MKS 単位系を用いる。

3.1 星の拡散

まず、花火玉から星が拡散した直後の星の位置を決定する。星の形状は 2.3 で述べた構造を考慮し、立方体とする。また、星は複数配置するため、お互いが重ならないよう初期位置を配慮する。最初に配置する星は花火玉の開花した位置に配置し、以後は z 座標に対して 5m ずつずらした位置に星を配置していく。

その後それぞれの星には初速度を与える。初速度は星が均等な放射状に拡散するような値を与える。 x - y 平面において、 x 軸正方向からの弧度を θ とした時、式 (3.1) から θ を計算する。但し、 i は 1 から始まる整数であり、星の数を上限とする。

$$\theta = \left(\frac{1}{3} + \frac{(i-1)}{60} \right) \pi \quad (3.1)$$

そして、式 (3.2) から初速度 \mathbf{V}_h を計算する。

$$\mathbf{V}_h = \left(40 + 5(i-1) \cos \theta, 40 + 5(i-1) \sin \theta, 0 \right) \quad (3.2)$$

3.2 星の移動

星の移動方法を以下に示す。星は 3.1 節で与えた初速度を \mathbf{V}_1 とし、斜方投射運動として、単位時間ごとの位置 \mathbf{P}_n と速度 \mathbf{V}_n を式 (3.3) と式 (3.4) で決定する。

$$\mathbf{P}_n = \mathbf{P}_{n-1} + \mathbf{V}_{n-1} \Delta t \quad (3.3)$$

$$\mathbf{V}_n = \mathbf{V}_{n-1} + g \Delta t \quad (3.4)$$

\mathbf{P}_{n-1} と \mathbf{V}_{n-1} はそれぞれ位置と速度の前の計算結果である。 g は重力加速度であり、常に y 軸負の方向を向いている。 Δt は単位時間である。また、星には玉から放出される際、強い回転がかかる。その表現を行うため、単位時間ごとに星に回転速度を加算する。その際に x 軸方向の回転速度、 y 軸方向の回転速度、 z 軸方向の回転速度として、それぞれ、 0 rad/s から 10 rad/s の範囲の速度を乱数を用いて計算し、加算する事で星ごとの回転速度にばらつきを与える。

3.3 煙の生成

星の移動に伴い、星から煙を噴出する。本研究では煙を煙粒子の集合体と定義する。星はある 1 つの面からしか煙を噴出しないものとし、その噴出面上に単位時間ごとに複数の粒子を生成する。粒子の作成位置にばらつきを持たせるために、粒子の作成位置 \mathbf{P} は噴出面上で乱数を用いて決定するものとする。噴出面のある

1つの頂点の位置ベクトルを \mathbf{A} とし、残る3つの頂点を反時計回りにそれぞれ \mathbf{B} 、 \mathbf{C} 、 \mathbf{D} とした時、式(3.5)から式(3.7)までを用いて粒子の作成位置 \mathbf{P} を計算する。

$$\mathbf{L} = \mathbf{B} - \mathbf{A} \quad (3.5)$$

$$\mathbf{M} = \mathbf{D} - \mathbf{A} \quad (3.6)$$

$$\mathbf{P} = \mathbf{A} + u\mathbf{L} + v\mathbf{M} \quad (3.7)$$

但し、 u 、 v は乱数を用いて生成された0から1の範囲の実数である。

図3.1は星が上部の面から煙粒子を噴出しているところを表している。図中の立方体は星を表しており、上部の赤い点が粒子を表している。

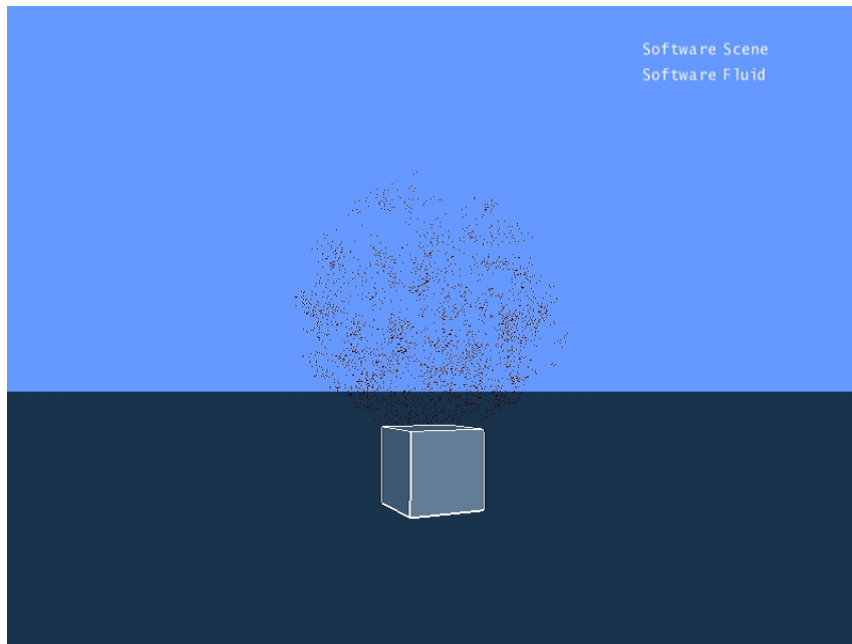


図3.1: 煙粒子の噴出する様子

3.4 煙の移動

粒子には生成時に、噴出口の向いている方向に $1m/s$ の大きさの初速度 \mathbf{V}_1 を与える。また、煙の上昇の表現として、Y軸正方向へ、 $0.1m/s^2$ の大きさの加速度 \mathbf{A} を与える。与えた初速度と加速度より、煙粒子の初期位置を \mathbf{P}_1 とした時、単位時

間ごとの位置 \mathbf{P}_n と速度 \mathbf{V}_n を以下の式 (3.8) から式 (3.10) を用いて計算する。但し、 \mathbf{P}_{n-1} と \mathbf{V}_{n-1} はそれぞれ位置と速度の前回の計算結果であり、 Δt は単位時間である。

$$\mathbf{A} = (0, 0.1, 0) \quad (3.8)$$

$$\mathbf{V}_n = \mathbf{V}_{n-1} + \mathbf{A}\Delta t \quad (3.9)$$

$$\mathbf{P}_n = \mathbf{P}_{n-1} + \mathbf{V}_{n-1}\Delta t \quad (3.10)$$

星の回転と移動に伴い煙の粒子は個々に異なる位置座標と速度を取得し、昼花火の煙としての形状を作成する。以下の図 3.2 は、この手法によって生成した煙の形状を表している。



図 3.2: 本手法によって生成した画像

3.5 煙のレンダリング

煙の形状を表現する煙粒子を元に、最終的な煙のレンダリングを行う。このレンダリングの手法として、ポイントスプライト [18] を用いる。ポイントスプライ

トとは、座標上のある点群に対しポリゴンを生成し、それらのポリゴンにテクスチャをマッピングするレンダリング手法である。座標上の点群のそれぞれの点は何らかの形状を持ち、かつそれらの形状が似通っている場合、それらの点群に対し常に画面方向を向く正方形のポリゴンを配置し、そのポリゴンに対してテクスチャをマッピングする事で形状を表現する。ポイントスプライトは、形状を複雑なポリゴンでは無く、単純な正方形のポリゴンと、テクスチャで表現するため、高速に描画できる事が特徴である。また、多くのGPUにはポイントスプライトを高速に描画できる機能が付いており、それらを利用する事で更に簡単に高速に描画できる事も特徴である。この手法は、点の集合を物体の形状として扱う場合によく用いられる。

まず、煙の色を制御する事が出来る様、煙の粒子位置にマッピングするテクスチャを生成する。本手法のテクスチャは1辺が32ピクセルのテクスチャを用いる。テクスチャ内のピクセルごとに式(3.11)から式(3.14)を用いて、色RGBと不透明度Aを決定する。

$$R = C_m R_o \quad (0 \leq R \leq 1) \quad (3.11)$$

$$G = C_m G_o \quad (0 \leq G \leq 1) \quad (3.12)$$

$$B = C_m B_o \quad (0 \leq B \leq 1) \quad (3.13)$$

$$A = 0.1 C_m \quad (0 \leq A \leq 1) \quad (3.14)$$

但し、色 $R_o G_o B_o$ は煙の色を制御する初期値であり、 $0 \leq R_o \leq 1$ 、 $0 \leq G_o \leq 1$ 、 $0 \leq B_o \leq 1$ であるものとする。 C_m はテクスチャ中心部の数値を高くした球状の分布を持つ実数値であり、式(3.15)から計算する。

$$C_m = \begin{cases} (1 - (\frac{r}{16})^2)^2 & (0 \leq r \leq 16) \\ 0 & (16 < r) \end{cases} \quad (3.15)$$

但し、 r はピクセル幅を1とした時の、テクスチャの中心からの距離である。図3.3が赤 ($R_o = 1, G_o = 0, B_o = 0$) を入力した場合に生成する1枚のテクスチャ画像である。

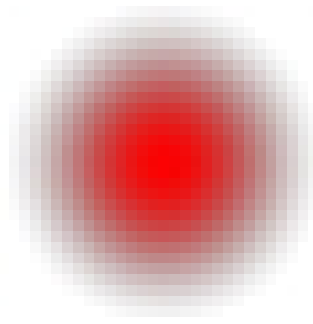


図 3.3: 生成された 1 枚のテクスチャの画像

このようにして生成したテクスチャを GPU のポイントスプライト機能を用いて、粒子の存在する位置にマッピングする。具体的には、ポイントスプライト機能を用いて、粒子の存在する位置に正方形のポリゴンを生成し、それらのポリゴンに対し生成したテクスチャをマッピングする。この時に煙の拡散を表現するため、テクスチャをマッピングする大きさ S とテクスチャをマッピングするポリゴンの不透明度 A を以下の式 (3.16) と式 (3.17) を用いて決定する。

$$S = b + kT_e \quad (3.16)$$

$$A = \frac{b^2}{a^2 S^2} \quad (3.17)$$

但し、 T_e はマッピング対象となる粒子を生成してから経過した時間である。また、係数 k と係数 a と係数 b は外部から入力する値であり、 b はテクスチャの大きさの初期値であり、 k はマッピングする大きさを制御する係数であり、 a はポリゴンの透明度を制御する係数である。そしてこの式 (3.17) によって計算した不透明度 A をポリゴンに適用する際に、テクスチャの色とポリゴンの色を乗算する事でテクスチャの色の变化をテクスチャのマッピング結果に反映する事が出来る。これらの処理によって、煙粒子が時間経過によって大きさと色を変化させる事が可能となり、煙の拡散する様子を表現する事が出来る。

第 4 章

検証と考察

4.1 シミュレーション環境

4章では3章で述べたシミュレーション手法の例を示す。星の移動と煙粒子の生成、移動の計算に、PhysX[19]を用い、3DCGによる描画にはFKSystem[20]を用いて実装を行った。実行したマシンの動作環境は表 4.1 の通りである。なお、本論

表 4.1: 実行環境

OS	Windows Vista
CPU	Core2 Duo CPU T9800 2.93GHz
GPU	NVIDIA GeForce 9300M GS
Memory	4.00GB

文に示すシミュレーション例では、重力加速度 g , タイムステップ Δt については表 4.2 の値を用いた。

表 4.2: 計算パラメーター

g	Δt
9.8	$\frac{1}{60}$

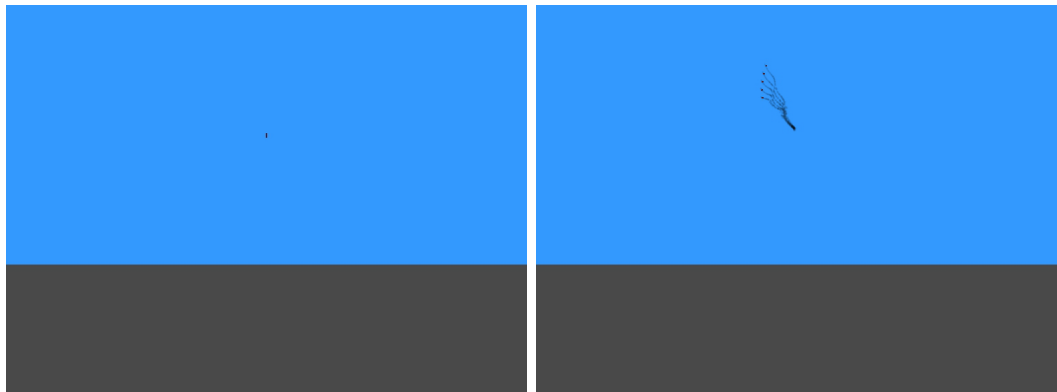
4.2 シミュレーション結果

以下に本手法を用いてシミュレーションを行った結果を示す。レンダリング時のパラメーターには表 4.3 の値を用いた。但し、色 $R_o G_o B_o$ は煙の色を制御する初期色、 b はテクスチャの大きさの初期値、係数 k はテクスチャをマッピングする大きさを制御する係数、係数 a はポリゴンの透明度を制御する係数である。本シミュ

表 4.3: レンダリングパラメーター

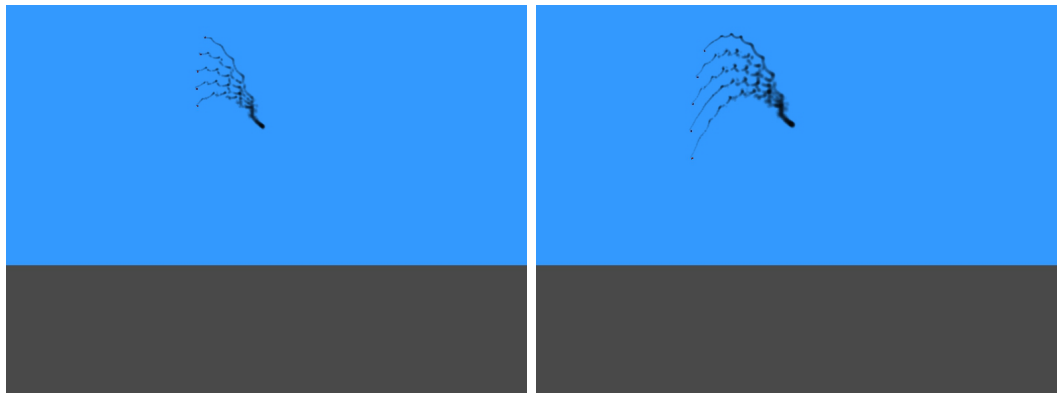
シミュレーション	R_o	G_o	B_o	b	k	a
図 4.1	0	0	0	6.0	2.0	0.3
図 4.2	0	0	0	6.0	4.0	0.3
図 4.3	0	0	0	6.0	2.0	1.0
図 4.4(a)	1	1	1	6.0	2.0	0.3
図 4.4(b)	1	0	0	6.0	2.0	0.3
図 4.4(c)	0	1	0	6.0	2.0	0.3
図 4.4(d)	0	0	1	6.0	2.0	0.3

レーションの時間ごとの状態を示した例が図 4.1 である。図 4.1 では星 5 個を用い、シミュレーション内での煙粒子の合計が 2000 個になるまで、星 1 つにつき煙粒子を 50 個/秒の割合で噴出させたシミュレーションの結果である。これにより、花火玉から複数の星を放出し、星の移動に伴って煙を生成していく様子と、時間経過に伴い煙の拡散していく様子が確認できる。



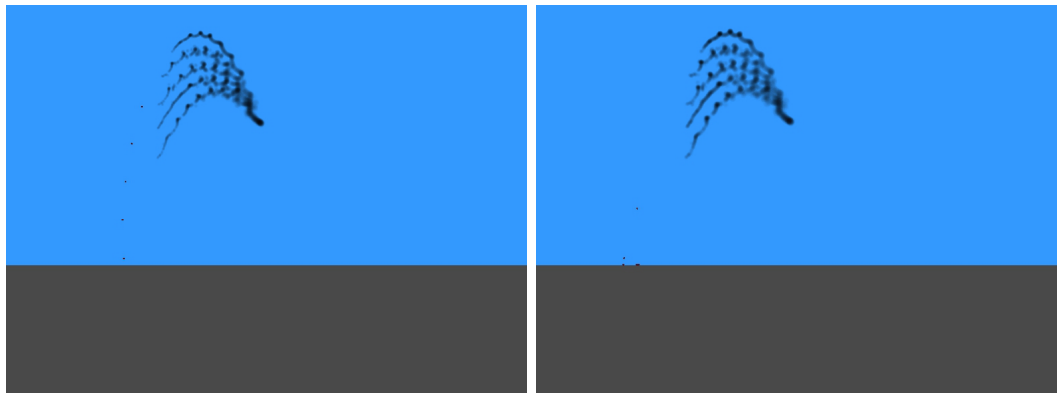
(a) frame = 0

(b) frame = 160



(c) frame = 320

(d) frame = 480



(e) frame = 640

(f) frame = 800

図 4.1: 昼花火のシミュレーション結果画面

次に、開始から 480 ステップ目の画像をテクスチャをマッピングする大きさを制御する係数 k のみ 4.0 に変更した画像が図 4.2、ポリゴンの透明度を制御する係数 a のみ 1.0 に変更した画像が図 4.3 である。これらの画像と、図 4.1 の 480 ステップ目の画像を比較することで、煙の大きさと濃さが係数 k と係数 a を用いて調整できる事が確認できる。

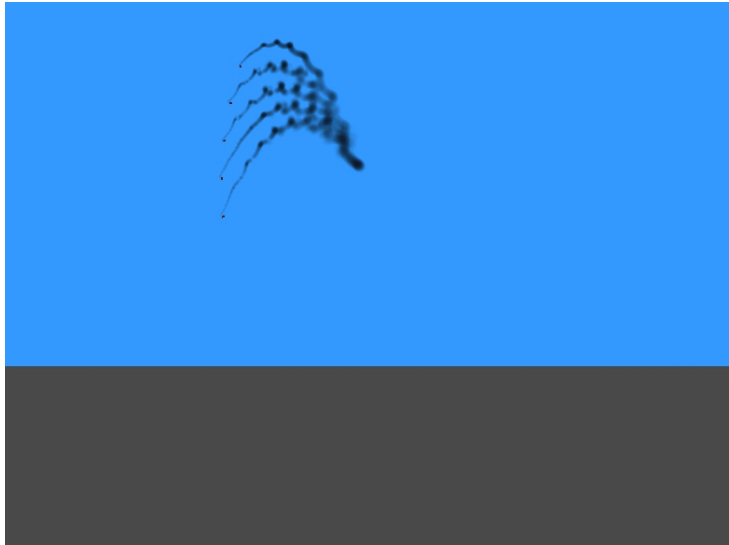
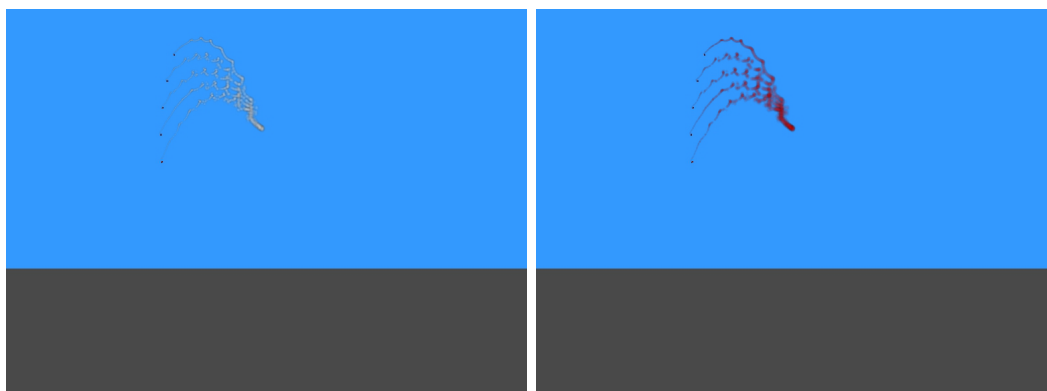


図 4.2: テクスチャをマッピングする大きさを制御する係数 k を変化させた画像



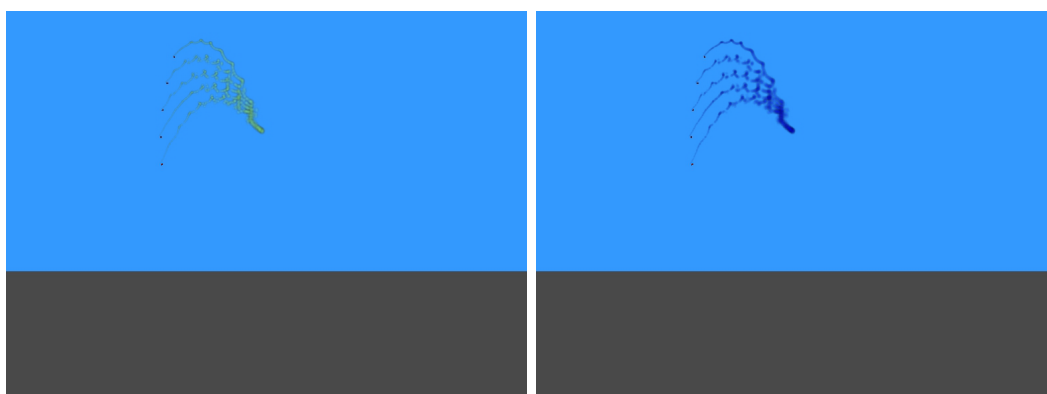
図 4.3: ポリゴンの透明度を制御する係数 a を変化させた画像

次に、開始から 480 ステップ目の画像の色の初期値 $R_oG_oB_o$ を様々に変化させた結果の画像が図 4.4 である。これらの画像により、色の初期値 $R_oG_oB_o$ を変えることで煙の色を変化出来る事が確認できる。



(a) 白色の煙を生成した結果

(b) 赤色の煙を生成した結果



(c) 緑色の煙を生成した結果

(d) 青色の煙を生成した結果

図 4.4: テクスチャの初期色を様々に変更した画像

次に、開始から 480 ステップ目の画像で、星ごとに異なる色の煙を噴出した画像が図 4.5 である。レンダリング時の星ごとの色 $R_oG_oB_o$ は表 4.4 の値を用いた。

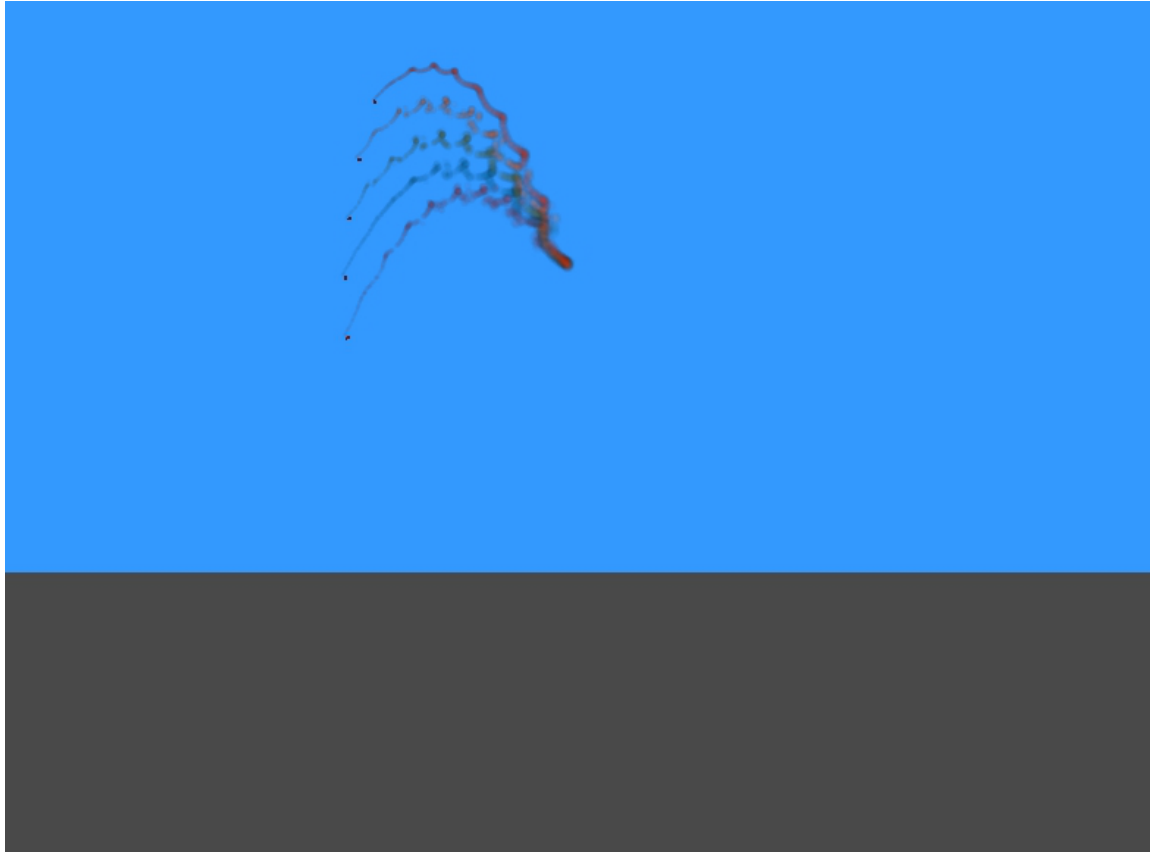


図 4.5: 星ごとに異なる色の煙を噴出した画像

但し、色 $R_oG_oB_o$ は煙の色を制御する初期色であり、星ごとに画面上部から星 1、星 2、星 3、星 4、星 5 と呼称するものとする。このように、星ごとに異なる色の

表 4.4: 星ごとの初期色

星の名称	R_o	G_o	B_o
星 1	0.98	0.15	0.00
星 2	1.00	0.37	0.10
星 3	0.40	0.42	0.18
星 4	0.00	0.48	0.76
星 5	0.67	0.20	0.33

煙を噴出させることで、煙柳としての表現の幅を広げる事が出来る。

4.3 実写画像との比較

図 4.1 の 480 ステップ目の画像と実際の昼花火の画像を比較したのが図 4.6 である。

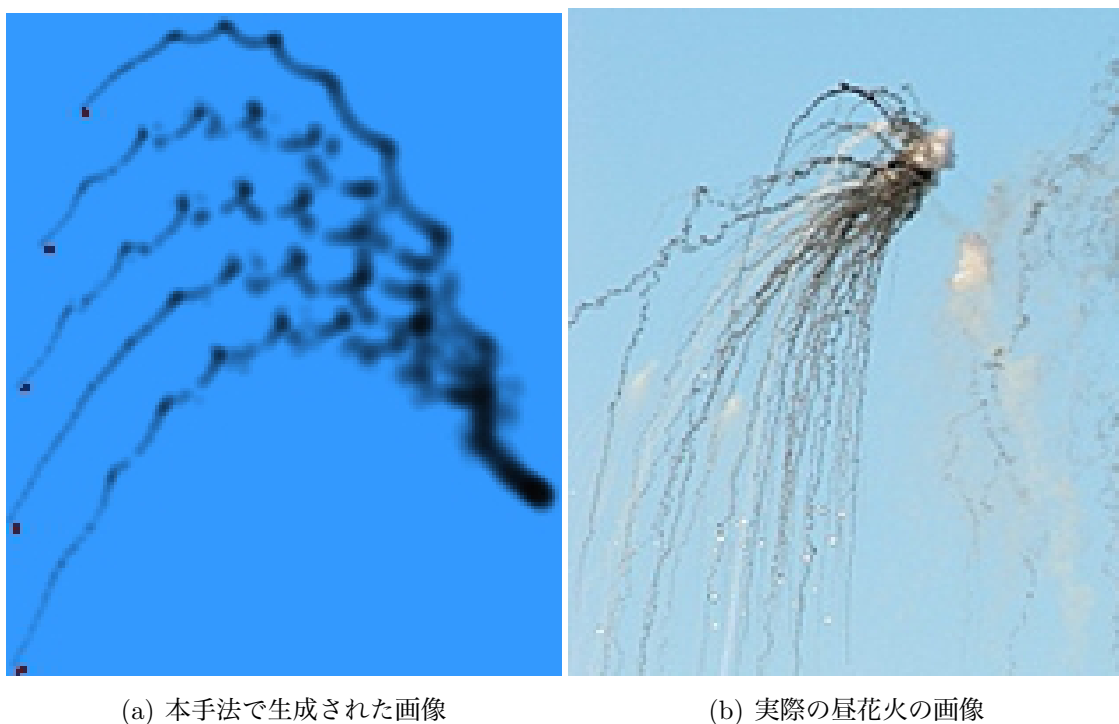


図 4.6: 本手法と実際の昼花火の比較

本手法の星から噴出された煙が扇形に広がり拡散する形状と、実際の煙柳の形状との間に類似性が見られた。また本手法で表現した、渦を巻くような形状の煙と実際の煙柳の煙の形状との間に類似性が見られた。これらの事から、昼花火の種類の一つである煙柳を表現する事が出来たと言える。しかし、実際の昼花火には星の放出時にも煙が生成されるが、本手法ではこの煙は表現されていない。この星の放出時の煙も表現する事で、煙柳としてのリアリティを高める事が期待できる。また、本手法では煙粒子のレンダリング時に、太陽などの光源による陰影の計算を行っていない。光源からの陰影を考慮して煙粒子をレンダリングする事で、煙のリアリティを高める事が期待できる。

4.4 リアルタイム性の検証

図 4.1 に示したシミュレーションの星の数を様々に変更した時のシミュレーションデータを表 4.5 に示す。また、表 4.6 はタイムステップ 160 ごとの平均処理時間を表している。図 4.1 のシミュレーションでは星 5 個を用いて煙柳を表現しており、この際の画面更新時間は最短で 2.6 ミリ秒、最長でも 9.9 ミリ秒だった。これは 1 秒間に最低でも 100 回は画面を更新する事が出来る事を示しており、1 秒間に 60 回の画面更新を必要とするリアルタイムな 3D コンテンツにおいても十分に利用が可能であると言える。しかし、シミュレーション内での星の数がリアリティに大きく影響するため、煙粒子が大量に必要になり、画面更新時間が長くなりやすいという問題点がある。煙粒子の数を減らすために、より少ない煙粒子で煙柳を表現できる手法を考えていく必要がある。

表 4.5: シミュレーションデータ

星の数	煙粒子の数	総ステップ数	平均更新時間
			[ミリ秒/ステップ]
1	400	800	1.75
2	800	800	2.75
3	1200	800	3.78
4	1600	800	5.08
5	2000	800	6.13
6	2400	800	7.48
7	2800	800	8.82

表 4.6: 区間ごとの平均処理時間

星の数	区間ごとの平均処理時間 (ミリ秒/ステップ)				
	区間 1	区間 2	区間 3	区間 4	区間 5
1	1.7	1.7	2	2.1	2.2
2	1.6	2.7	3.6	4.4	5.3
3	1.9	3.6	5.8	6.5	7.1
4	2.3	4.8	7.2	7.7	8.2
5	2.6	6.1	9.2	9.9	8.8
6	3.0	7.3	11.1	9.8	9.7
7	3.8	8.6	14.5	14.4	12.5

第 5 章

まとめ

星の移動と煙粒子の生成、移動の計算に PhysX[19] を使い、3DCG による描画には FKSystem[20] を用いて、煙柳と呼ばれる種類の昼花火の煙を表現した。まず花火の玉が放出する星と呼ぶ、煙を放出する物体の挙動を計算し、この星から単位時間ごとに煙を表す粒子を、ある一定の面からその面の法線方向に向けて噴出する事で昼花火の煙の形状を表現した。そして、その煙の粒子の位置にハードウェアのポイントスプライト機能を用いてテクスチャをマッピングする事で煙の色を表現した。また、テクスチャマッピング時に本手法で生成したテクスチャを用いることで昼花火の煙の色を柔軟に表現できるようにした。

これにより、花火玉から放出された複数の星がその移動に伴って煙を生成する様子が確認できた。煙のテクスチャを生成する際に、マッピングの大きさと透明度を係数を用いて変化する事で、煙の拡散を表現する事が出来た。また、煙柳の煙の形状の最大の特徴である、螺旋のような形状の煙が扇形に拡散していく様子を表現する事が出来た。これによって、煙柳の煙の形状を表現する事が出来たとと言える。

しかし、本手法ではテクスチャのマッピング時に陰影を考慮しておらず、太陽などの光源からの影響を考慮していない。また、煙の単散乱、多重散乱を考慮するなど、更に煙のレンダリングに適した手法を開発する必要がある。また、現状では昼花火の中の煙柳の表現のみに留まっており、煙竜や旗物など、玉内部に星

以外が含まれる構造を持つ昼花火の表現手法も考える必要がある。更に、昼花火は単発で打ち上げて観賞するものではなく、同時に複数打ち上げて観賞する事が多くある。そのため、より少ない煙粒子数で昼花火を表現するモデルを開発するなど、更に高速化の工夫を図る必要がある。

謝辞

本研究を進めるに当たり、温かいご支援と粘り強いご指導をいただきました東京工科大学メディア学部の渡辺大地講師に感謝致します。日ごろから研究についての意見を交わしあった研究室のメンバーに感謝します。

また、論文内では実際の昼花火の画像などに「高画質壁紙写真集無料壁紙」様 [21] の画像と、「詩郎の写真紀行」様 [22] の画像を掲載させて頂きました。御礼申し上げます。特に、突然の論文への画像の掲載のお願いにも関わらず、快く掲載に応じて頂いた、「詩郎の写真紀行」管理人の山崎詩郎様には重ねて御礼申し上げます。

本研究にご協力いただきました全ての皆様と、この論文に目を通してくださった全ての方々には、御礼申し上げます。

そして何よりも、日ごろから時に励まし、時に勇気付け、常に温かく見守ってくれた家族に心から感謝します。

最後に、研究中に心くじけそうになった時、心の支えにしたこの言葉を持ってこの論文を締めくくりたいと思います。

人の足を止めるのは絶望でなく”諦観”

人の足を進めるのは希望ではなく”意思”

漫画、ARMS より [23]

参考文献

- [1] Zhou, K., Ren, Z., Lin, S., Bao, H., Guo, B., and Shum, H.-Y., “Real-time smoke rendering using compensated ray marching,” in [*SIGGRAPH '08: ACM SIGGRAPH 2008 papers*], 1–12, ACM, New York, NY, USA (2008).
- [2] 土橋宣典, “セル画調の煙のモデリングおよびレンダリングシステムの開発,” **情報処理学会研究報告 グラフィクスと CAD 研究会報告 128**, 41–46 (2007).
- [3] Harris, M. J., “Real-time cloud rendering for games,” *Game Developers Conference* (2002).
- [4] 嶋田有紀, “Gpu シェーダーを用いた動的な雲の高速レンダリング,” **情報処理学会論文誌 49(5)**, 1743–1749 (2008).
- [5] 小玉浩平, “花火のインタラクティブシミュレーション,” **計算力学講演会講演論文集 16**, 15–16 (2003).
- [6] 小玉浩平, “粒子法による花火の cg アニメーション,” **計算力学講演会講演論文集 14**, 401–402 (2001).
- [7] 茂木健一郎, [**プロフェッショナル仕事の流儀** 8], NHK 出版 (2006).
- [8] 冨木一馬, [**花火の本**], 淡交社 (2004).

- [9] 鈴木唯史, “光源制御の効果を用いた仮想線香花火システムの試作,” **電子情報通信学会総合大会講演論文集**, 406 (1997).
- [10] 伊藤圭祐, **線香花火のビジュアルシミュレーションに関する研究**, 学士論文, 南山大学数理情報学部情報通信学科 卒業論文要旨集 (2006).
- [11] 山本光重, **実写映像を利用した花火演出支援システム**, 修士論文, 奈良先端科学技術大学院大学情報科学研究科 (2002.).
- [12] 小野里公成, **[日本の花火]**, ちくま新書 (2007).
- [13] ふそう, “ふそうのページ.” http://blogs.yahoo.co.jp/fuso_aeroking.
- [14] “戸田橋花火大会公式ホームページ.” <http://www.todabashi-hanabi.jp>.
- [15] “大曲全国花火競技大会公式ホームページ.” <http://www.ltd.co.jp/hanabi/index.html>.
- [16] 株式会社サーチナ, “世界最長「昼花火」で感動の2008年よ永遠に.” http://news.searchina.ne.jp/disp.cgi?y=2008&d=0918&f=national_0918_012.shtml.
- [17] 登坂和洋, “「マグナス効果」使い新型風車を開発,” **産学官連携ジャーナル** 4(5), 9-11 (2008).
- [18] 床井浩平, “和歌山大学 床井研究室.” <http://marina.sys.wakayama-u.ac.jp/~tokoi/?date=20060227>.
- [19] Nvidia, “Physx.” http://www.nvidia.co.jp/object/physx_new_jp.html.
- [20] 渡辺大地, “Fk tool kit system.” <http://fktoolkit.sourceforge.jp/>.
- [21] “高画質壁紙写真集無料壁紙.” <http://k-kabegami.com/>.

[22] 山崎詩郎, “詩郎の写真紀行.” <http://www004.upp.so-net.ne.jp/shiro-photo/>.

[23] 皆川亮二 and 七月鏡一, [*ARMS*], 小学館.