

論文 / 著書情報  
Article / Book Information

題目(和文)	計算機による手描き風画像の作成とその応用に関する研究
Title(English)	
著者(和文)	齋藤豪
Author(English)	Suguru Saito
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第4153号, 授与年月日:1999年3月26日, 学位の種別:課程博士, 審査員:中嶋正之
Citation(English)	Degree:Doctor of Engineering, Conferring organization: Tokyo Institute of Technology, Report number:甲第4153号, Conferred date:1999/3/26, Degree Type:Course doctor, Examiner:
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

計算機による手描き風画像の作成と  
その応用に関する研究

1999年1月

指導教官 中嶋 正之教授

提出者 東京工業大学大学院  
情報理工学研究科  
計算工学専攻  
齋藤 豪

# 目次

<b>第 1 章</b>	<b>序論</b>	<b>1</b>
1.1	研究背景及び目的	1
1.2	本論文の章構成	2
1.3	章間の関連	3
<b>第 2 章</b>	<b>手描き風画像作成のための関連研究</b>	<b>5</b>
2.1	まえがき	5
2.2	CG 分野での手描き風画像の従来研究	5
2.2.1	3次元幾何情報入力型	6
2.2.2	2次元画像入力型	10
2.2.3	インタラクティブ操作を入力データとする研究	14
2.2.4	アルゴリズムのみにより模様を作成する研究	17
2.3	画像処理における関連研究	17
2.3.1	局所的な方向解析に関する研究	17
2.3.2	輪郭検出に関する従来の研究	19
2.4	色計算における関連研究	20
2.4.1	色座標系	20
2.4.2	Kubelka-Munk の光のエネルギー計算モデル	30
2.5	むすび	33
<b>第 3 章</b>	<b>手描き風画像自動生成のための画像処理手法</b>	<b>34</b>
3.1	まえがき	34
3.2	作成する手描き風画像の位置づけ	35
3.3	2D-FFT を用いた画像解析手法	36
3.3.1	局所方向性検出	36
3.3.2	局所方向性信頼度	38

3.3.3	局所波長期待値検出	39
3.4	輪郭近傍情報検出	39
3.5	局所情報検出結果	40
3.5.1	局所方向性検出結果	40
3.5.2	局所方向性信頼度検出結果	46
3.5.3	局所波長期待値検出結果	47
3.5.4	輪郭近傍情報検出結果	48
3.6	むすび	49
<b>第4章</b>	<b>手描き風画像自動生成法</b>	<b>50</b>
4.1	まえがき	50
4.2	ストローク描画による基本描画手法	51
4.3	単純減法混色計算を付加したストローク描画手法	53
4.4	簡易的な混色計算を付加したストローク描画手法	53
4.4.1	簡易絵の具モデル	54
4.4.2	簡易混色計算を行う描画手法	56
4.5	ストロークに光沢表現を付加した描画手法	57
4.6	ぼかしを基本とした描画手法	58
4.7	各手法による描画結果	60
4.7.1	ストロークによる基本描画法の描画結果	60
4.7.2	単純減法混色計算を付加したストローク描画手法の描画結果	66
4.7.3	簡易混色計算を行う描画手法の混色結果	70
4.7.4	光沢処理を加えた手法の描画結果	76
4.7.5	ぼかし処理による描画結果	78
4.8	むすび	83
<b>第5章</b>	<b>絵の具の発色を再現する計算機上の絵の具モデル</b>	<b>84</b>
5.1	まえがき	84
5.2	単純減法混色で表現できない絵の具の発色問題	84
5.3	絵の具のシミュレーションへの Kubelka-Munk の理論の適用	86
5.4	Kubelka-Munk の理論の一般式の実装に必要な数値の導出	86
5.5	画像単位領域あたりに必要となるデータと操作	90
5.5.1	データ構造	90
5.5.2	データ操作	91

5.6	本提案手法の実装に必要な標本数	92
5.7	本提案実装法の混色評価実験	94
5.7.1	単純減法混色実装法の発色特性	95
5.7.2	Curtisらの実装法の発色特性	96
5.7.3	本提案手法の発色特性	96
5.8	むすび	102
<b>第6章</b>	<b>直感的な色選択と絵の具モデルのパラメータ対応</b>	<b>103</b>
6.1	まえがき	103
6.2	Kubelka-Munkの理論の実装とパラメータ入力問題	104
6.3	絵の具の特性決定のための直感的なパラメータ群	106
6.3.1	色の選定パラメータ	106
6.3.2	スペクトル曲線形状決定パラメータ	106
6.3.3	その他のパラメータ	106
6.4	直感的パラメータ群とスペクトラ反射率の対応付	106
6.4.1	スペクトルと色座標系の統一的な取り扱い	107
6.4.2	CIEXYZ表色系の回転	108
6.4.3	X'Y'Z'表色系内でのパラメータ対応付けアルゴリズム	111
6.4.4	パラメータ対応付けアルゴリズムの十分性の検討	113
6.4.5	パラメータ対応付けアルゴリズムの改良	113
6.5	直感的パラメータ入力による絵の具の混色実験	115
6.5.1	スペクトル反射率曲線形状の先鋭度による制御実験	115
6.5.2	先鋭度を变化させた等色絵の具の混色実験	115
6.5.3	新旧スペクトル反射率曲線決定アルゴリズムによる絵の具の混色実験	115
6.6	むすび	122
<b>第7章</b>	<b>計算機上の筆モデル</b>	<b>123</b>
7.1	まえがき	123
7.2	筆のモデル化ステップ	124
7.3	筆の背骨線モデル	124
7.3.1	筆の背骨線の姿勢シミュレーション	125
7.3.2	エネルギー最小値を求める方法	127
7.3.3	筆の背骨線のノード数の決定	128
7.4	筆の接地面の計算	129

7.4.1	背骨線の曲線形状作成 . . . . .	129
7.4.2	背骨線の姿勢からの画材と筆との接地面の導出 . . . . .	129
7.5	筆の内部の液体移動 . . . . .	131
7.5.1	筆内部の絵の具蓄積モデル . . . . .	131
7.5.2	紙の絵の具蓄積モデル . . . . .	131
7.5.3	パレットの絵の具蓄積モデル . . . . .	133
7.6	提案筆モデルによる描画結果 . . . . .	133
7.6.1	筆の柄の姿勢変化に応じた背骨線の形状変化結果 . . . . .	134
7.6.2	筆の接地面の導出結果 . . . . .	134
7.6.3	筆による描画結果 . . . . .	136
7.7	むすび . . . . .	136
<b>第 8 章</b>	<b>絵の具モデル・筆モデルの統合描画結果</b>	<b>138</b>
8.1	まえがき . . . . .	138
8.2	実装環境 . . . . .	138
8.3	色選択ユーザインタフェースの実装 . . . . .	139
8.4	ペイントツール全体の機能 . . . . .	140
8.5	ユーザによる描画例 . . . . .	142
8.6	むすび . . . . .	148
<b>第 9 章</b>	<b>結論</b>	<b>149</b>
9.1	まえがき . . . . .	149
9.2	結論 . . . . .	149
9.3	今後の課題 . . . . .	150
9.4	研究展望 . . . . .	152
9.5	むすび . . . . .	152
<b>謝 辞</b>		<b>153</b>
<b>参 考 文 献</b>		<b>154</b>

# 目次

1.1	各章の関係図	4
2.1	Meier の手法による 3 次元オブジェクトの描画結果	9
2.2	Haerberli による手描き風画像生成結果及び対象画像	10
2.3	安居院らによる手描き風画像生成結果及び対象画像	11
2.4	染谷らによる手描き風画像生成結果及び対象画像	12
2.5	映画 Mulan で使われた竹の CG	18
2.6	スペクトルサンプリング	21
2.7	マンセル表色系	22
2.8	等色実験結果	24
2.9	CIE 1931 rgb 系の等色関数	25
2.10	CIEXYZ 表色系の等色関数	26
2.11	二つの同じ色に見える光	27
2.12	CIEXYZ 空間から xy 色度図への変換	28
2.13	Kubelka Munk の式の原理図	31
3.1	CG の表現法による分類	35
3.2	振幅マトリックスの要素と波形の関係	37
3.3	振幅マトリックスの上での線の引き方	42
3.4	従来法と本手法との方向性検出結果の比較	43
3.5	自然画像サンプル“じてんしゃ”	44
3.6	自然画像に対する局所方向性検出結果の例	45
3.7	方向信頼度検出例	46
3.8	局所周波数期待値検出例	47
3.9	輪郭近傍情報検出例	48

4.1	太線化アルゴリズム	52
4.2	塗り重ねによる混色における遮へい率と光路の関係	54
4.3	完全に乾いていない上への重ね塗りによる透過率、光路モデル	55
4.4	ストロークに用いたテクスチャ	57
4.5	ぼかしフィルタの形状	59
4.6	“じてんしゃ”に対するストローク基本描画手法描画結果	61
4.7	対象画像“くだもの”	62
4.8	“くだもの”に対するストローク基本描画手法描画結果	63
4.9	“じてんしゃ”に対する描画の途中経過	64
4.10	ストロークの面積と画像の大きさ比較	65
4.11	サンプル対象画像“せいぶつ”	67
4.12	“せいぶつ”への単純減法混色を付加した描画結果	68
4.13	“くだもの”への単純減法混色を付加した描画結果	69
4.14	“くだもの”への簡易混色計算を付加した描画結果	71
4.15	サンプル対象画像“やまとくも”	72
4.16	“やまとくも”への簡易混色計算を付加した描画結果	73
4.17	サンプル対象画像“ひろば”	74
4.18	“ひろば”への簡易混色計算を付加した描画結果	75
4.19	光沢処理を施したストローク描画の結果	77
4.20	サンプル対象画像ねこ	79
4.21	“ねこ”に対するぼかし描画結果	80
4.22	単純なぼかしフィルタを用いてぼかした画像	81
4.23	“くだもの”に対するぼかし描画結果	82
5.1	絵の具層内部を進む光の経路	85
5.2	顔料のスペクトルデータの例	89
5.3	描画処理の概念図	91
5.4	標本数と色の誤差の関係	93
5.5	単純減法混色を実装した場合の発色結果	98
5.6	Curtisらによる実装法で彼らのパラメータを用いた場合の塗り厚を変化させた時の発色変化	99
5.7	Curtisらの実装法での発色結果	100
5.8	本章での提案手法による発色結果	101
6.1	表面的なパラメータとスペクトル反射率を結び付ける必要がある	107



6.2	X'Y'Z' 表色系と色相彩度、無彩色の関係	109
6.3	X'Y'Z' 表色系での可視色の存在領域	110
6.4	色座標から反射率への変換アルゴリズムの説明図	112
6.5	重み関数 $f_{wS'}$ の概形	114
6.6	旧アルゴリズムによるスペクトル反射率の決定結果	116
6.7	新アルゴリズムによるスペクトル反射率の決定結果	117
6.8	旧アルゴリズムにより先鋭度を変化させた絵の具による混合混色結果	118
6.9	新アルゴリズムにより先鋭度を変化させた絵の具による混合混色結果	118
6.10	旧アルゴリズムにより作られた絵の具の混合混色発色実験結果	120
6.11	新アルゴリズムにより作られた絵の具の混合混色発色実験結果	121
7.1	筆の毛のモデリング	124
7.2	様々な節の数の背骨線のモデル	125
7.3	背骨線の各部と変数、定数の対応付け	126
7.4	節数1で $\sin$ 曲線を描画した際の結果	129
7.5	背骨線を曲線化するための Bezier 曲線用制御点	130
7.6	筆内部の絵の具の蓄積モデル 微小なタンクとパイプを持つ	132
7.7	筆の柄の姿勢と中央制御点の移動、曲線形状の関係	134
7.8	筆の接地面の形状導出結果例	135
7.9	筆モデルによる線描画例	136
8.1	色選択 GUI の実装例	140
8.2	ペイントツール実装例のスクリーンショット	141
8.3	ユーザの使用風景	142
8.4	ユーザによる平仮名の描画例	143
8.5	ユーザによる漢字の描画例	144
8.6	ユーザによるハングルの描画例	145
8.7	ユーザによる絵の描画例 1	146
8.8	ユーザによる絵の描画例 2	147

# 表 目 次

5.1	隠ぺいに必要な絵の具の厚さの平均 . . . . .	87
5.2	微小領域あたりに必要となるデータ . . . . .	90
5.3	実験に用いる絵の具の顔料 . . . . .	94
7.1	節の数と計算時間の関係 . . . . .	128
8.1	本ペイントツールで用いるタブレットからの信号一覧 . . . . .	140

# 第 1 章

## 序論

### 1.1 研究背景及び目的

計算機を用いた画像作成は今日様々な用途に使われている。多様なデータの可視化や、ユーザインタフェース、ゲームなどは CG が切り開いた分野と言える。

芸術や映像制作の分野でも近年は CG はなくてはならないものとなっている。そこでは、CG は実写との合成処理が要求される。従って合成しても違和感のない CG の表現手法が求められる。そこで、実写と区別がつかないフォトリアリスティックな表現手法、モデリング法の研究がまず始めに実用化され [1][2][3][4]、注目を集めた。

しかし、そのような CG の研究から一歩離れて、画像の表現方法を考えた場合、写真と見間違えるような写実的な表現方法以外にも世の中には多くの表現方法があることに気が付く。また、それらは写真が今日これだけ容易に撮影可能であるにもかかわらず、依然として広く使われていることに気が付く。

例えば、観光地で売られている絵はがきをでは、写真の他にきれいな水彩画が並べて置かれていたり、また機械や医療用の技術解説書では手で描いたイラストが用いられている。さらに、ある航空会社の緊急時の脱出説明では実写に手描き風の画像処理が施されたビデオを上映している。

写真ではなく、手描きな絵、もしくは手描きらしい画像が好まれる理由は、写真とは異なり、手描きでは好きなように細部にこだわったり、また省略をしたりといった情報の操作が可能であるためだと考えられる。また、画像全体を通して見るならば、写真とは異なった、暖かみであるとか、勢いであるとか、雰囲気伝えることが可能であるからだと考えられる。

そのような理由から、写真以外の表現方法が CG の研究対象として近年脚光を浴びるよう

になって来た。これらの研究はノンフォトリアリスティックレンダリング ( Nonphotorealistic rendering ) の名で呼ばれ、今後ますます研究が盛んになると予想される [5][6][7]。

本論文では、ノンフォトリアリスティックレンダリングの一つである手描きの絵のような画像を生成するための研究について述べている。

本論文で述べる研究は、大きく3つのテーマからなっている。それらは写真画像を入力として手描き風の画像を作成する手法、絵の具の混色を計算機上で再現する手法、そして多様な筆の表現に着目した筆モデルを用いたストローク描画法についてである。次節では、上記3つのテーマと本論文の各章の構成の関係について述べる。

## 1.2 本論文の章構成

本論文は9章及び謝辞、参考文献、から構成される。本章を除く各章の説明を述べた後で各章間のつながりについて述べる。

第2章では、手描き風CGに関する従来の研究、手描き風CGを生成するための周辺分野の研究について述べる。手描き風CGに関する従来の研究に関してはそれらの入力データ形式から3種類に分類し、説明を行う。また本研究で必要となる周辺分野の知識には、画像処理、色彩工学、光学物理、シミュレーション技術があり、それらについての説明も行う。

第3章では、写真から手描き風CGを自動生成するための画像処理手法に関する研究について述べる。手描き風の画像の特徴には、描く際に用いる筆などの道具により自ずと生じてしまう細部の省略や、輪郭線などの強調といった画像の局所的なモディフィケーションが強く影響を与えている。そこで、画像全体を認識するための画像処理ではなく、画像の局所的な特徴量を検出する画像処理手法について述べる。

第4章では、第3章の画像処理手法により得られる画像の局所情報を用いて、手描き風CGを生成する手法に関して述べる。個々のストロークが目立つ描画法をモデルとしたアルゴリズムでは画像の局所情報に従って個々のストローク描画を行うことにより絵を生成する。また、個々のストロークが目立たない描画法をモデルとしたアルゴリズムでは、画像の局所情報に従ったボカシの技法を用いて新たな画像を生成する。

第5章では、光の混色である加法混色と異なった、絵の具の発色における混色に関して、計算機上で再現と鮮やかな発色を可能とする絵の具のモデルに関しての研究について述べる。絵の具の混合混色、塗重ね混色を実現するために KubelkaMunk の理論を用い、絵の具の特性を光の吸収率と散乱率で定義するモデルを構築する。また、混色しても濁りの少ない発色特性の優れた絵の具のモデルを実現するため、スペクトルの標本数についての検討、実際の絵の具からのパラメータ決定法について述べる。

第6章では、第5章で述べた絵の具のモデルに対し、任意の望んだ色に対応するパラメータセットを与える手法について述べる。色を指定する際、パラメータは色相、彩度、明度のよう値が習慣的にも直感的にも好ましい。一方、絵の具のモデルはスペクトル反射率をパラメータとして用いる。そこで、両者のパラメータを関係付けられ、かつ混色した際の色の強さを調節可能なパラメータ変換法について述べる。

第7章では、筆の毛先をモデル化し、多様なストロークの描画を可能とする手法に関する研究について述べる。対話的な操作のペイントツールにおいて筆のタッチを活かすことの出来る自由度の高いストローク表現を可能とするため、毛先の物理シミュレーションを行える三次元的な筆モデルを構築し、さらに筆と画材との絵の具の移動についてモデル化について述べる。

第8章では、第5章から第7章までの応用例として絵の具のモデル、筆のモデルを用いた対話的描画ツールの実装例について述べる。最後に描画結果を示す。

第9章では、本論文の総括及び今後の課題について述べている。

## 1.3 章間の関連

本論文は3つのテーマから成っており、各章の関係は図1.1のように表される。第3章及び第4章が自動描画に関する研究、第5章及び第6章が絵の具の色に関する研究、第7章が筆に関する研究である。第8章では、第5章から第7章までの成果を統合した応用例を示している。

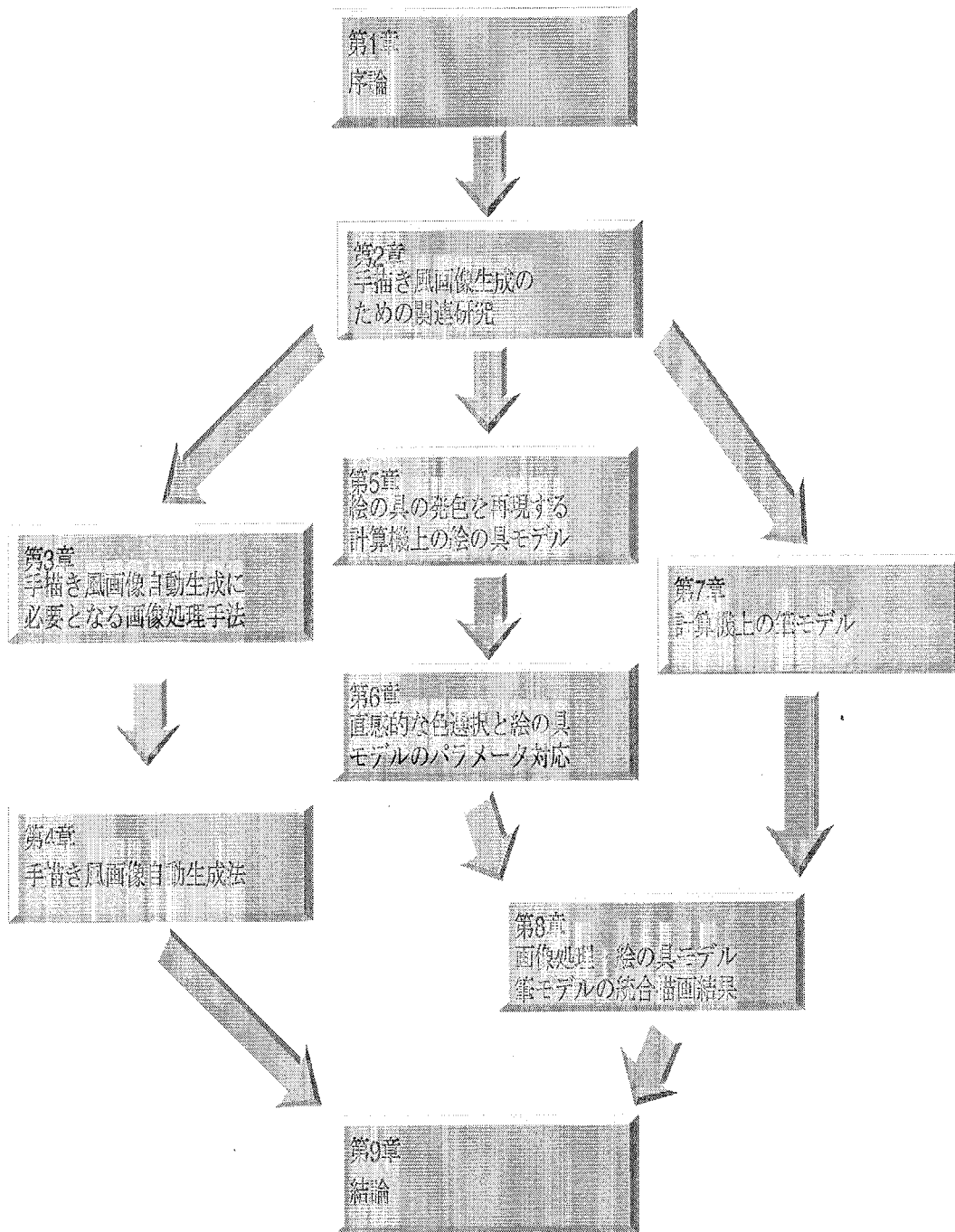


図 1.1 各章の関係図

## 第 2 章

# 手描き風画像作成のための関連研究

### 2.1 まえがき

本章では 2.2 節において手描き風の CG 作成に関する従来研究についてのまとめを行い、その中で本研究の位置づけについて述べる。2.3 節においては手描き風 CG を自動作成するために必要となる画像処理分野の研究について、2.4 節においては手描き風 CG に用いる絵の具モデルに必要な色彩工学、光学の研究について述べる。

### 2.2 CG 分野での手描き風画像の従来研究

手描き風 CG の研究はその表現形式、応用分野で分けることも可能であるが、本論文では入力データ形式で分類を行う。これは、入力データ形式の違いが最もその後の処理の違いとして現れるからである。手描き風 CG の作成に用いる入力データ形式には、3次元幾何情報、2次元画像、ポインティングデバイスからの時系列データがある。

- (1) 3次元幾何情報を入力とするもの: フォトリアリスティックなレンダリングと同様にポリゴンを主な入力とし、3次元から2次元への写像の際に特殊な処理を行うことにより手描き風の画像を作成する。
- (2) 2次元画像を入力とするもの: 写真などの画像を入力とし、その画像を画像処理の手法を用いて解析し、その解析結果を利用して新たな画像を再描画する。

- (3) マウスやペン型タブレットのようなポインティングデバイスからのインプットを入力とするもの: デバイスからの時系列データを、直接的に仮想的な筆のストローク情報へ置き換えるか、もしくは仮想的なストロークを間接的に制御するために用いる。

各データ形式共に異なった利点が存在する。3次元幾何情報を入力データとした場合、視点移動やオブジェクトの再配置などデータの再利用が容易である。2次元画像情報の利用は既に多量に蓄積されている画像データを即座に利用可能であることからデータ作成が容易である。インタラクティブ操作による入力データはユーザからの意志を直接反映させることが容易である。

また、入力を必要せずに主にアルゴリズムのみにより自動的に模様を作成する研究もある。以下に各分類における研究について概略を述べる。

### 2.2.1 3次元幾何情報入力型

三次元幾何情報を入力データとする表現法は幾何的に2次元平面に射影を行い描画するアルゴリズムを基本とする。ただし、種々の処理を施して単純な写實的射影とは異なった画像を生成する。

#### [1] 面の法線ベクトルや面の位置からストローク形状を決定する表現法

Haeberli [8] は、レイトレースの手法を用いて3D空間の手描き風画像の作成を作成する方法を提案した。各ピクセルからレイを飛ばして到達した物体表面の視点からの距離や法線情報をもとに手描きによるようなストロークの大きさや方向を自動決定する事が出来ることを示した。また、Haeberliはマウスにより指示した位置からレイをリアルタイムに飛ばすことにより単純なインタラクティブ性も持たせる事も可能とした。

#### [2] Gバッファを用いた表現法

斎藤らの研究 [9][10] は3次元データを表示する際に一般的に用いられるZバッファ以外にGバッファと呼ぶ3次元物体の幾何情報を格納するバッファを定義した。Gバッファは描画される画面と同等の大きさで解像度を持ち、射影される各ピクセルに対応する面の距離、法線方向等を格納している。その情報を基にして線画風の画像表現を行う。斎藤らの目標は分かり易い画像の生成であり、写實的な画像よりも物体の形状を伝え易くするための線画風のストロークの自動生成を目的としている。



**[3] 射影により求めた等色領域に輪郭描画を付加する表現法**

金子らはセルアニメーションの描画表現法を3次元オブジェクトデータから自動生成する手法について研究 [11][12] を行っている。金子らの手法による第一の目的はセル描画の特徴である平面的な描画と輪郭線の付加である。輪郭線に関してはCGレンダリング手法の一つであるZバッファ法を用いて行っている。また、平面的な描画に関してはオブジェクト色の直接の射影で行っている。さらに、従来のセルアニメーションでは作成が困難なシェードについての自動生成に関しても処理手法の提案を行っている。

**[4] 質感を表現するテクスチャをオブジェクト表面に使った表現**

Winkenbachらはテクニカルペン画風表現の画像をポリゴンベースの3次元オブジェクトデータから生成する手法 [13] について発表した。このアルゴリズムの基本はペンによって描かれたストロークの集積によるテクスチャを3次元オブジェクト表面に張り付け、それを2次元平面に射影するものである。しかし、ここで定義されているテクスチャは単なるテクスチャとは異なり与えられた光源、視点の関係による模様の変化やユーザーからの描画省略の指示によるトーンの変化が自動的に可能のように定義されている。ユーザによる描画省略の指示は描画される2次元平面上で行う。その指示を3次元空間のオブジェクト表面に射影した後、3次元空間上でテクスチャに変更を加えて射影した3次元平面での描画を変更するという処理は興味深い。

後にNURBS曲面での濃淡制御の可能な技術“controlled-density hatching”についての提案を行い、曲面からなる3次元オブジェクトの描画にも機能拡張を行っている [14]。

また、Couttsらも同様な描画結果を目指したレンダリングアルゴリズム [15] を提案している。

**[5] 射影後の平面へテクスチャを張るモノクロテクニカルイラスト風表現**

関谷らの報告 [16] では三次元CADモデルから白黒テクニカルイラスト風表現を生成する手法について提案している。彼らの描画法は三次元モデルの二次元平面への射影の際の輪郭強調と材質の質感をあらわす記号化されたテクスチャの使用を特徴としている。輪郭線には陰影による強弱、内形線外形線による強弱を付けている。テクスチャ描画は2次元に射影した領域に対してテクスチャを張り付けている点がWinkenbachらとは異なる。素材に対応したテクスチャとして金属、ゴム、皮革を定義している。

**[6] 3次元モデルの輪郭線強調**

望月らの研究 [17] では、3次元モデルの形状理解を容易にするため輪郭強調法についての提案を行っている。対象物の輪郭線を背景との境界に現れる外形線とオブジェクトの面と面

の境界部に生じる内形線とに分け、それぞれの線をZバッファ法を利用して検出し、外形線は黒、内形線は白で表現している。また、光源方向も考慮して線の強さを变化させた描画も行っている。単純なシェーディングアルゴリズムでは分りづらい輪郭線の強調が出来ている。

#### [7] 誇張表現制御のための投影変換

宮澤、望月らの研究 [18][19] では、3次元形状を誇張する投影変換手法についての提案を行っている。絵画で用いられている正確な写像とは異なる射影を調べ、3次元オブジェクトの一部または全体を誇張して投影するアルゴリズムについて述べている。誇張度合という変数により誇張を段階的に制御することが可能である。この手法により、CGレンダリングで一般的なピンホールレンズを仮定した投影法よりも迫力のある画像を作り上げることを可能としている。

#### [8] 樹木の骨格モデルから作成される水墨画風の表現

張らの研究 [20][21] では樹木の3次元モデルから水墨画調画像を作成する手法を提案している。この方法では紙と筆先をセルとするセルオートマトンを用いた墨の移動拡散のモデル化を行い、かすみ、滲み表現の可能なストロークの作成を行う。葉のない木の三次元スケルトンモデルを作成し、その幹、枝の奥行き情報、太さ情報、枝分かれの際の分岐角情報を利用して筆のストロークの順番、筆圧を決定し、最終的に木の根本方向から描画を行い立ち木の水墨画表現を行っている。

この方法によるセルオートマトンでの滲みやかすれのストローク表現は計算コストがかかるが美しい結果を得られている。

#### [9] 3次元幾何データの油絵風表現

Meierの研究 [22] では3次元幾何データをもとにして、油絵風の画像をレンダリングするアルゴリズムについて提案を行っている。レンダリングは筆ストロークを描画することにより行われる。ストロークの描画位置は3次元オブジェクトの表面に定義されたパーティクルをスクリーンに投影されることにより決定される。ランダムにストローク位置を決定して描画するとアニメーション時にストロークがノイズのようにちらついて見える現象をこの手法により抑えている。ストロークの色、大きさ、方向を決定する情報は幾何オブジェクトを通常のレンダリングを行うことによるなどして得ている。結果の例を図2.1に示す。



図 2.1 Meier の手法による 3 次元オブジェクトの描画結果

#### [10]4 色分解にヒントを得た Fine Art Rendering

Lee らは、ポリゴン平面をイエロー、シアン、マゼンタ、ブラックの 4 色の縞で塗った模様により 3 次元オブジェクトを表示する手法 [23] を提案している。Fine Art を目指した結果画像は、レンダリングの多様な可能性を示していると言える。ただし、好みの分れる画像である。

#### [11] 輪郭線のみを表示する方法

Markosian らの研究 [24] では、三次元幾何データをレンダリングする部分において輪郭部分のみを表示する手法について提案している。輪郭線を描画する方法にはそのまま描画する方法、手書き風の揺らぎを与えた描画法、ストロークテクスチャを使う方法の三種類があり異なった雰囲気の結果を仕上げています。シェーディングされた画像と比較してシンプルで仕上がりが、白黒印刷では特に分かりやすい画像が作成できる。また、レンダリング処理はリアルタイムで可能となっている。

#### [12] テクニカルイラストのための照明モデル

Gooch らの研究 [25] では、Phong のシェーディングモデルと異なった、イラスト用のシェーディングを提案している。鏡面反射角からずれるに従って、暗くなる場所を、暖色から寒色へと変化させて描画し、陰となる部分も分かりやすいテクニカルイラストを 3 次元モデルから作り出すことに成功している。また、イラストでは記号的に表す金属表面の曲面の輝度変化の縞模様も簡単なシェーディングにより作成している。

### 2.2.2 2次元画像入力型

三次元空間から二次元に既に射影された写真や映像等のデジタル画像を入力データとし、手描き風画像生成を行う従来の研究について述べる。ここでは画像処理や筆ストロークのモデル化について検討を行っている。

#### [1] フラクタル圧縮技法による副産物

Barnsley は IFS(Iterated Function Systems)fractal の研究 [42] において画家が描いたような画像の生成を行った。Lansdown らによれば彼はフラクタルによる高圧縮の方法に関して研究していたのだが、複合結果が意図せずにゴーギャンによって描かれるような絵になってしまったとのことである [5]。

#### [2] 画像の輝度勾配に依存するストロークによる描画表現法

Haerberli は画像の輝度勾配からストロークの方向を決定して描画を行う手法 [8] について発表した。この絵は機械的描画であるとも言えるが高度な画像認識を行わずに手描き風の画像の生成可能性を示した最初の例である。図 2.2 に対象画像及びその処理結果を示す。



図 2.2 Haerberli による手描き風画像生成結果及び対象画像

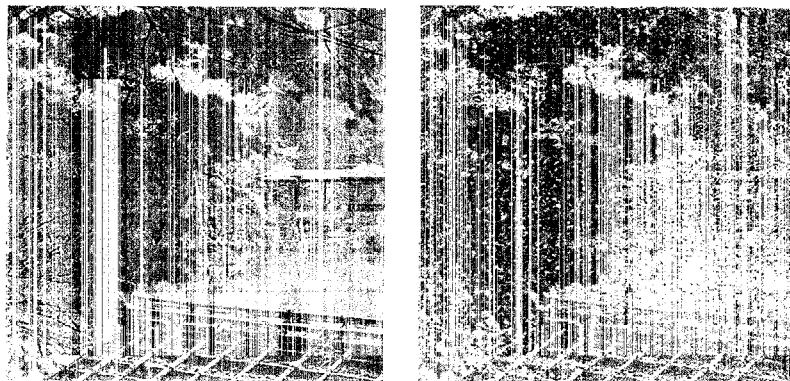


図 2.3 安居院らによる手描き風画像生成結果及び対象画像

### [3] 自動点描画及び直線形ストローク自動描画手法

安居院らは点描画と直線ストロークによる描画法 [43][44] を提案している。点描画表現に関する自動描画法は他に類を見ない。彼らは写真画像の上に周辺部分の明度と彩度の高い丸形のストロークを定義し、それを塗り重ねる事で点描画風の画像を作成した。描画手順は、大型の丸形のストロークを画像全体に描き込んだ後により小さな丸形の物を描き込み、最終的に三段階の塗り重ねを行って完成するものである。図 2.3 に対象画像及びその処理結果を示す。

#### [4] 減色、ディザ法を用いたストローク色の決定法

染谷らも画像をストローク型の模様で生め尽くすアルゴリズム [45][46] によって手描き風の画像を作成している。ここでは、減色してディザ法を施した原画像からストロークの色を決定して、限られた色数で描かれたような雰囲気を出している。図 2.4 に対象画像及びその処理結果を示す。染谷は、また単純な色変換を用いて水彩画風な画像の作成 [47] を試みている。

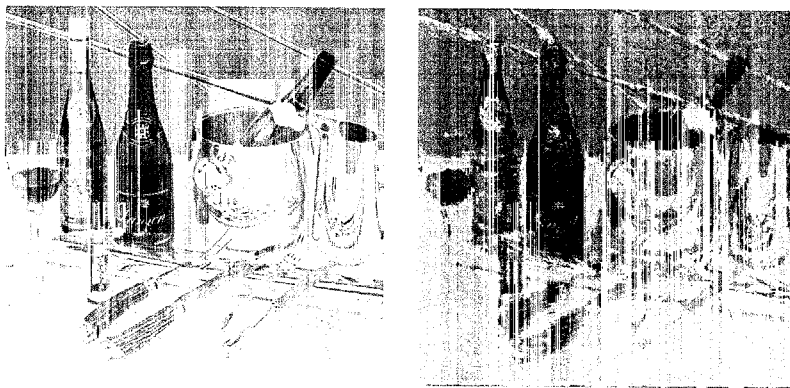


図 2.4 染谷らによる手描き風画像生成結果及び対象画像

#### [5] 流線の描画法の応用

Greg らは、流れの場を可視化する流線の描画方法 [54] を提案し、その応用として自然画像に輝度勾配に依存した滑らかなメッシュを重ねた画像を示している。カラー画像からモノクロ画像をつくり、ぼかして方向解析を行って流れの場を作った後、彼らの提案する流れ場の可視化流線を作成し、元画像に重ねている。彼らの示した洋梨の画像に対する結果からは、実際の手描きの絵と直接的には似ていないが、滑らかな曲線が機械的でなく面白い効果を得ることが可能であることがわかる。

#### [6] テクニカルペン風画像の描画

Salisbury らの提案 [55] では、画像の濃淡情報を基に、最終画像の大きさにかかわらずトーンを一定にすることを目標にした、テクニカルペンで描いたような画像を作成する手法について述べている。はじめに、入力された離散画像から連続した輝度情報を得るため低域フィルタを通して。ただし、輪郭部分については事前に解析を行う。次に目標とする画像サ

イズに拡大し、描画用の比較画像とする。次に定義されている順序付ストロークテクスチャからストロークを一つ取り出し、描画を行う。一つの描画を行う度に描画面の新たなトーンを計算し、比較画像と同じトーンになるまで描画を繰り返す。コピーによる拡大と異なり、拡大後もテクスチャが拡大されないことから質を保ったままの拡大が可能である点を利点としてあげている。

#### [7] 仮想版木による版画画像作成法

大河内らは濃淡画像を入力として、仮想版木への彫刻刀による彫り方を自動決定する手法 [48] について述べている。仮想版木は3次元情報を持ち、刷り作業には、仮想的なバレンによる操作をインタラクティブに行うツールを用いている [49]。木版画表現という新しい表現法を提案しているが、白と黒のバランスの制御といった版画に特有の問題についての検討が必要そうである。

#### [8] K-平均アルゴリズムを用いた領域分割法に基づく描画法

笠尾らは K-平均アルゴリズム [51] を用いて画像を微小な領域に分割し、それらの各領域をあたかもストロークであるかのように画像を再構成することにより、手で描いたような画像や、独特な雰囲気 of the 画像を作り上げる手法について述べている [50]。領域分割を基本としており、輪郭がはっきりと残る画像を作り上げている。実際の描画に見られるストロークの重なりや、滲みといった実際の画材による絵を作り上げることよりも、新しい表現方法を目指した画像が生成される。

#### [9] オプティカルフロー技術を用いた動画像に対する適用

Litwinowicz は、動画像に対して、印象派風の画像へ変換する手法について提案 [52] を行っている。動画像の各フレーム画像にガウスフィルタとソーベルフィルタを施し、輪郭線を検出し、ストロークは輪郭線を越えないように描かれる。ストロークの方向は画像の輝度信号にガウスフィルタを施し、その勾配から求めている。動画像を入力とすることによる特徴としてオプティカルフローを用いてフレーム間での対象物の移動に追従したストロークの配置を行っている。

#### [10] 多サイズのストロークによる段階的描画

Hertzmann は、大きなストロークで描画をした後に元画像と比較し、色が異なるところを更に小さなストロークで描くことにより、画像を作り上げる手法 [56] について述べてい

る。多層の描画アルゴリズムと言える。この手法は油絵で上塗りが下の色を完全に隠す場合には効果的であると言える。

#### [11] 画像モーメントを用いた描画法

白石らは、画像モーメントを用いた描画方法 [53] の提案を行っている。まずはじめに、画像モーメントから、各描画ストロークに対応する矩形領域の大きさ、方向、位置を決定し、次にストロークを面積の大きい順にソートし、大きなストロークを先に、小さなストロークを後に描くことにより詳細部を塗りつぶすことなく描く手法である。しかし、ストロークの大きさに関する制御の問題を残しており、入力画像の局所領域毎に応じた描画について課題として挙げている。

### 2.2.3 インタラクティブ操作を入力データとする研究

マウス等のポインティングデバイスによって計算機内の仮想画材を操り絵を描く従来の研究に対して本項では述べる。

#### [1] Quantel Paint Box

Lansdown らによればインタラクティブペイントシステムの中で初めて従来ある画材によって描かれたような描画効果を持つアルゴリズムが使われたのは Quantel の PaintBox の "chalk" 機能でだそうである [5]。この機能は簡単なルックアップテーブルによりストロークを描く際にあるピクセルは描画するかしないか決定するものであった。表現上はストロークが均等色で塗り潰される事なく "chalk" らしい描き方が出来るものと考えられる。

#### [2] セルオートマトンによりモデル化された絵の具による表現

Lansdown らによれば Cockshott らは水っぽいものから粘性のある物質まで様々な特徴を持つ絵の具のモデル化 [26] について述べている [5]。この物質は滲み、滴り、染み込み混色が仮想キャンバス上で可能であり、それらはセルオートマトンによって実現されている。この研究では驚くべきことに仮想的な環境条件まで絵の具のキャンバスへの定着に影響を与えるようにされている。重力方向による絵の具の滴る方向の変化や、気温による絵の具の乾燥の変化をモデル化している。ただし、このようなセルオートマトンによる計算を伴うストローク描画はリアルタイムでは行えないため後処理としている。ユーザに対するインターフェースは一般的なペンティングシステムと同等のものを別に用いている。



### [3] 精巧な毛筆のモデリングを使った描画表現

Strassmann はファジィや複雑系のモデル化に関する興味からペイントブラシとインクについての動的なシミュレーション [27] を行った。彼の定義によるブラシパターンは一本のスプラインをもとにしている。スプラインの各ノードには位置と筆圧情報が記憶されておりインクの消費量に関して処理が行われる。ストロークを長く描くことにより筆のインクは消費される。ユーザは筆にインクを付ける操作により筆の中にインクを補充することが出来る。この研究では筆の設置面の幅によって濃度変化を与えるようにし、日本画調の画像を描く事を可能としている。

また、Strassmann の研究をもとに Guo らは毛筆文字フォント作成支援システムの研究 [28] を行っている。

### [4] テクニカルペンによる描画をマウスオペレーションにより行う手法

Salisbury らは、等幅なラインを描くことが出来るテクニカルペンを使って描かれたような絵を作成出来るシステムに関する研究 [29] を行った。このシステムではテクニカルペンによって描かれたようなテクスチャを“塗る”ことにより広範囲な書き込み操作を簡便に行うことが出来る。テクスチャは多くのストロークにより構成されており、それら個々のストロークには描画の優先順位が割り当てられている。ユーザーは塗り込み操作によって優先順位の低いストロークまで描くように指示を与えることにより暗い領域を作ることも可能であるし、逆に低い優先順位のストロークから消していくことにより暗くしすぎた領域を明るくすることも可能である。このように可逆的な描画、消去が用意されている点は大変良いユーザインターフェースであると言える。

後に、細いペンストロークの向きを対話的に一括して入力する Line interporated fill, Curve interporated fill なども提案し、完成度を高めている [33]。

### [5] 色鉛筆画のためのペイントツール

高木らは色鉛筆による描画のような雰囲気画像を作成可能なペイントシステムについて研究を行っている。水溶性の色鉛筆の加水効果の再現のために LIC を使ったアルゴリズムの提案 [30] や、描画時の紙と色鉛筆の顔料との間の微視的な現象をシミュレートする紙の 3 次元モデルの提案 [31][32] を行い、Pumpkins (A Painting Unit of Multi-Purpose Kind and INtelligent Stationery) システムの開発を目指している。

### [6] 模様をストロークとして利用する手法

Hsuらは計算機で最も多く用いられている定幅なストロークよりも豊かな表現力を持つストローク skeletal stroke について提案及び有効性を示した [34]。skeletal stroke は幅が可変であるほか、ストロークの折り返し方も三種類用意されており表現が豊かである。また、skeletal stroke は単なるストロークと異なり様々な画像をストローク化することができ、筆を描く操作で複雑な伸張や曲げ等の処理が行われた画像をストロークとして描くことが出来る。セルオートマトンによるストロークの生成は表現力が豊かである反面計算量が多い点が問題となる。skeletal stroke は比較的計算量を必要とせずストロークの表現を上げる手法であるといえる。

### [7] 水彩描画のためのモデル

Curtisらは、絵の具のモデルに Kubelka-Munk の理論を用いて水彩風な描画が可能なペイントツールを作成 [35] した。水彩画のかすれ、ストロークの輪郭部分の濃縮化、筆の返しによる不規則な乾きによる効果、紙の不均一性と顔料による粒状感、滲み効果、色の塗り重ねの計算機上でのシミュレーションモデルの構築を行っている。対話的な描画法以外にも、簡単な3次元モデルを水彩風にレンダリングする手法や、写真を水彩画風へ変換する手法についても触れており、水彩画と呼べる画像の作成例を挙げている。なお、絵の具の発色モデルに関して、Curtisらの手法と本論文の提案手法との比較検討を第5章で行っている。

### [8] セルアニメーションへの複雑な模様の描画

Corrêaらはセルアニメーションでは複雑な模様のある物体を動かすことが手作業の限界から難しい点に注目し、複雑な模様を計算機上で簡単にマッピングすることが可能な手法 [36] を提案した。処理のプロセスは次のとおりである。ユーザはまず絵を描き、次に簡単な3次元モデルを作り上げる。計算機は3次元モデルの輪郭線を計算し、それに合う曲線をユーザは絵の上をなぞることにより入力する。計算機は入力された曲線に合うようにモデルの形状を変形させて、レンダリングし、テクスチャ付の画像を作成して合成する。従来のセルアニメーションでは難しかった模様付の物体を以前よりは容易に動かせるようにする手法である。

### [9] その他

この種類の商用ツールは種類も多い。Adobe社のPhotoshop[37]はペイント系の商業ツールとしては最も有名である。また1997年にMetaTools社とFractal Design社の合併によ

り生まれた MetaCreations 社の Painter[38](旧 Fractal Desing 社製) は自然の画材を真似た描画を行うことに特殊化したツールである。フリーソフトウェアでは The Gimp(the GNU Image Manipulation Program)[39] が有名であり、Photoshop と比較しても劣らない機能を備えている。

#### 2.2.4 アルゴリズムのみにより模様を作成する研究

アルゴリズムにより、模様を自動作成する研究について述べる。

##### [1] 植物模様の自動作成

Wong らは 文章の周りなどを囲む植物模様を描く、自動描画アルゴリズム [40] について提案を行っている。従来から、植物の 3 次元モデルの作成については多くの研究があり、今日では非常にリアルな植生まで再現可能である [100]。しかし、二次元に記号化された植物では、実際の植物と異なった茎や花の繋がり描画の方が簡素化されて良い。そこで、画像中のすき間を埋めるようなアルゴリズムを示している。

##### [2] アニメーションの背景植物の自動作成

1998 年に公開されたディズニーアニメーションの映画 “Mulan” では背景の図 2.5 に示すような竹を CG で作成したシーンが使われた。その描画には 3 次元の竹モデルの射影による手法を用いず、二次元の植物描画法を新たに開発して用いた。3 次元形状を射影した場合、描画結果がどうしても 3 次元的になり、flat な作画スタイルと合わなかったのである。そこで、カメラアングルにあわせて 2.5 次元的モデルを形式化し、さらに筆によって描いたような滑らかなトーンの陰つけを葉に行うような手法を開発した [41]。

## 2.3 画像処理における関連研究

ここでは画像の局所的な情報を解析する従来の研究として、方向性解析 ( oriented pattern analysis ) に関する研究と輪郭線検出に関する研究について述べる。

### 2.3.1 局所的な方向解析に関する研究

Kass らは、ガウス関数による粗目のバンドパスフィルタとグラディエントと用いて局所的な方向解析を行う手法を提案した。



図 2.5 映画 Mulan で使われた竹の CG

彼らの基本アイデアは式 (2.1) に示すバンドパス型フィルタとグラディエントと単位方向ベクトル  $(\cos \theta, \sin \theta)$  とを用いて式 (2.2) で定義される  $V(\theta)$  によって式 (2.3) 局所方向性  $\psi$  を求めるものである。

$$H(x, y) = [\delta_1^{-2} e^{-r^2/\delta_1^2} - \delta_2^{-2} e^{-r^2/2\delta_2^2}] \quad (2.1)$$

$$V(\theta) = W \{(\cos \theta, \sin \theta) \cdot \nabla H * I\}^2 \quad (2.2)$$

ただし、 $0 < \theta \leq \pi$ 、 $I(x, y)$  は画像輝度関数、 $W(x, y)$  は積分値を 1 に正規化したガウス関数であり重み関数として用いられている。

$V(\theta)$  から局所方向性を次のように定義している。

$$\psi = \tan^{-1} \left( \frac{\int_0^\pi V(\theta) \sin(2\theta) d\theta}{\int_0^\pi V(\theta) \cos(2\theta) d\theta} \right) / 2 \quad (2.3)$$

更にこの手法を Rao ら [59] は発展させた。

Rao らは式 (2.1) の代わりとして、ガウス関数を直接用いた。また、Kass らが式 (2.2) において近傍の勾配を滑らかにした上で方向性を求めているのに対して、Rao らは仮の方向を求めた後にそれら方向の近傍平滑化を行って方向を決定している。

Rao らは方向性に加えて方向性の強さについて興味深い定義を行っている。ここでは輝度勾配の大きさと方向性のコヒーレンスから方向性の強さを以下のように定義している。

$$\rho = G_{mn} \frac{\sum_{(i,j) \in W} \|G_{ij} \cos(\theta_{ij} - \hat{\theta}_{mn})\|}{\sum_{(i,j) \in W} W G_{ij}} \quad (2.4)$$

ただし、着目画素の座標を  $(m, n)$ 、輝度勾配の大きさを  $G$ 、 $\theta_{ij}$  を  $(i, j)$  における方向性、 $\hat{\theta}_{mn}$  を着目画素の方向性、 $W$  を着目画素を中心とするあらかじめ決められた広さの窓としている。

て検出してしまう。

また、Freemanら [60] の研究から、志沢らは  $\theta$  方向への  $n$  階微分ガウスフィルタを用いて多重スケール多重方向場の検出を行っている。すなわち  $\theta$  方向への一階微分演算子 (式 (2.5)) を用いて内積の性質から式 (2.6) で方向場を定義した。さらに方向パターンは線形重ね合わせによって合成されると仮定して、多重方向場を式 (2.7) と定義した。しかし、式 (2.7) による方向検出法は着目画素に於ける微分情報という極めて近傍情報のみからの検出である。そこで、式 (2.8) に示すようにより広範囲な情報からの大局的な方向検出法として  $(x, y)$  を中心とする円形ガウス関数  $G_\sigma(x, y)$  を式 (2.7) に加えた。

$$a(\theta) = \partial_\theta = (\cos \theta) \partial_x + (\sin \theta) \partial_y \quad (2.5)$$

$$a(\theta(x, y))f(x, y) = 0 \quad (2.6)$$

$$a(\theta_n)a(\theta_{n-1}) \cdots a(\theta_1)f(x, y) = 0 \quad (2.7)$$

$$G_\sigma(x, y) * \{a(\theta(x, y))f(x, y)\} = 0 \quad (2.8)$$

以上のように従来の方向検出法はガウス-微分フィルタという周波数フィルタとしてはバンドパス特性を持つフィルタを利用してきた。しかし、バンドパス型フィルタを自然画像に対して使用した場合、通過帯域の選択により方向性の検出精度が変化すると考えられる。

本論文では複数の方向性が存在した場合に最も強い方向性を検出することが可能でありかつ、様々な画像に対して安定した方向検出を可能とすることを目的として、フーリエ級数展開を用いた方向検出を行う。また、周波数情報から、ストロークの幅、長さの決定に必要な情報も検出する。

### 2.3.2 輪郭検出に関する従来の研究

輪郭検出に関しては sobel 法や zero-cross 法が多く用いられている。しかし、主観的輪郭 [81][82] と呼ばれる実際には見えず人間の補間によって導かれる輪郭線を含まない場合に限って議論しても、今日あらゆる条件で満足の得られる輪郭検出法は提案されていない [83]。

本節では、従来用いられている zero-cross 法、sobel 法について述べる。

#### [1]zero 交差法

光強度が急激に変化している位置は、光強度を空間的に 2 階微分した関数がゼロを横切る位置と一致する。従って、ゼロ交差を検出すれば光強度が急激に変化している位置を知る事

が可能である。光強度が急激な変化をするところは物体の境界である可能性が強いのでゼロ交差位置を輪郭位置とする。以上がゼロ交差法による輪郭検出の概念である。実際の検出においてはガウスラプラシアンフィルタを用いてゼロ交差を求める。

また、人間の視覚特性に関する研究では、視覚に関する細胞である X 型神経節細胞の受容野が  $\nabla^2 G$  で近似出来ることを示している [84]。

ただし、やや滑らかな輝度変化をする輪郭では検出結果と輪郭位置がずれることが報告されている点、ノイズ除去の為にフィルタに広いガウス関数を用いると検出されるゼロ交差位置と輪郭がずれる点から、本論文における輪郭検出法ではゼロ交差法を採用しない。

### [2]sobel 法

ソーベル法はガウス-グラディエント法による指向性バンドパスフィルタを用いた輪郭検出法である。次式により得られる値  $E$  から輪郭を検出する。ただし、 $g$  を円形ガウス関数、 $I$  を画像関数とする。

$$E = \left(\frac{\partial}{\partial x} g * I\right)^2 + \left(\frac{\partial}{\partial y} g * I\right)^2 \quad (2.9)$$

輪郭部分は様々な周波数成分によって構成されていると見なせるので、輪郭周辺部において  $E$  は輪郭でない領域に比べて相対的に大きな値を取る。ガウス関数  $g$  は画像をぼかす特性があるため、 $E$  で輪郭線の位置を厳密に決定する事は出来ない。しかし、 $E$  の輪郭に近づくとつれて大きな値を取る特性により、厳密な輪郭を決定出来ないまでも輪郭位置をある程度検出できる。よって本論文では sobel 法を基本とする輪郭検出を用いることとする。

## 2.4 色計算における関連研究

本論文で扱う絵の具の発色を工学的に扱うためには人間の光に対する生理的な特性を知った上で、いかにして CIE の色座標系が決定されたのか知る必要がある。そこで、2.4.1 項では色座標系についての説明を行う [65]。次に、絵の具の混色が加法混色でないことから、その混色計算を行うために必要となる、Kubelka-Munk の理論による光のエネルギー計算モデルについて 2.4.2 項では述べる。

### 2.4.1 色座標系

人間は波長 380nm から 780nm の間の光を目によって感じる事が出来る。また、380nm から 780nm までの光の各波長におけるエネルギーの分布の違いにより人間は光に色を感じる

事が出来る。

歴史的には古くから、色をある種の座標系によって表現し、人間の感じる色を定量的に扱えるよう研究が行われてきている。

本項では、色を表現する座標についての基礎知識についてまとめる。

### [1]Spectrum coordinate system

白色光をプリズムにあてる実験をした場合、空気とプリズムの境での屈折率が波長により異なる事から、入射した白色光は多色の光に分解される。分解された光はスペクトル (spectrum) もしくは分光と呼ばれる。

プリズムの実験から人間の目は各スペクトルを違った色として認識出来る能力を持っている事がわかる。

では、例えば図 2.6 のように可視帯域を等間隔に標本化した各スペクトル成分をベクトルとする座標系で色を表したらばどうだろう。確かに色は表現できるが、問題が生じる。そのような座標系では人間の目には同じ色に見える所が一点で表せない。

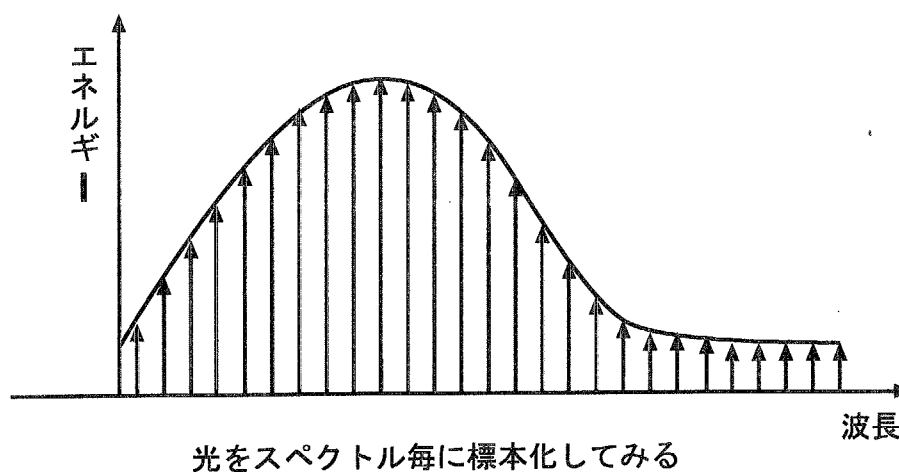


図 2.6 スペクトルサンプリング

### [2]Munsell color notation system

アメリカの画家 A. H. Munsell は 三つの特徴、色相 (hue)、彩度 (Saturation)、明度 (lightness) を用いて様々な色を配置する手法を考案した。つまり、図 2.7 のように色を 3 つの情

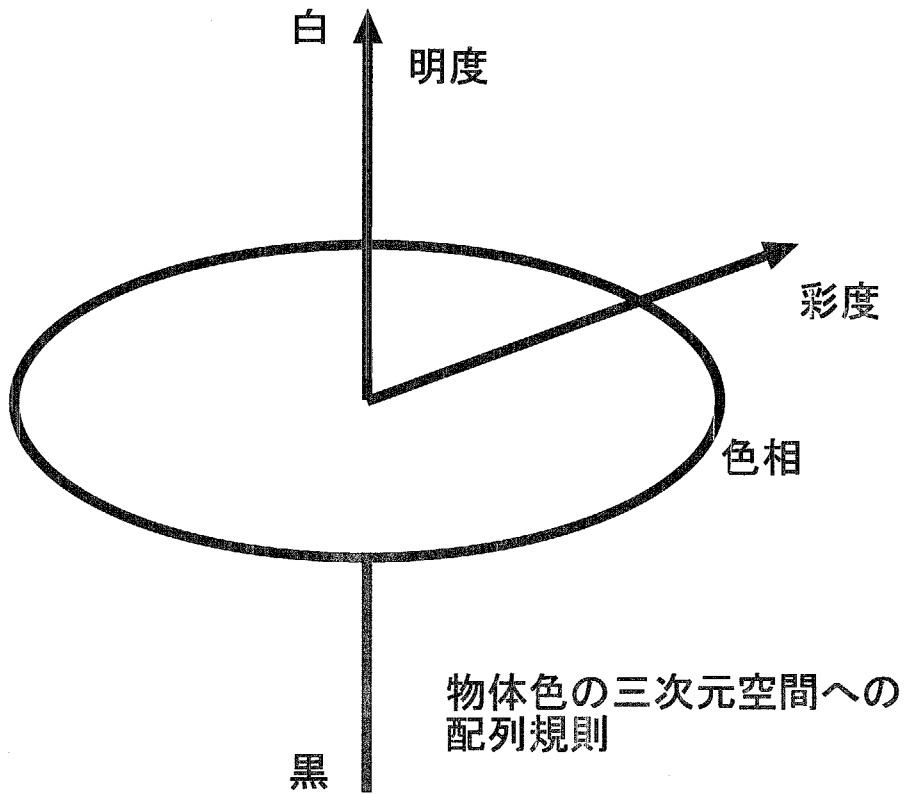


図 2.7 マンセル表色系

報で表現する方法を考案した。色の配置には心理実験により得た色差をもとにされた。このマンセルの色の表現方法を用いて色を表したものをマンセル表色系と呼ぶ。

調べたい色がマンセル表色系のどこに対応するかは、マンセル表色系の各色で塗られた色表と呼ばれる色のタイルと調べたい色を人間が見比べる事により決定する。

なお厳密には、今日マンセル表色系と呼ばれているものはマンセル自身が作製した表色系にその後の修正を加えたものであり、修正マンセル表色系 Munsell renotation color system を指す。

次に機械的に色座標を求める事が出来る表色系を列挙する。



**[3]RGB coordinate system**

1920年代、英国の国立物理研究所の Guild とロンドン大学の Wright は独立に等色関数を求める基礎実験を行った。概略は次のとおりである。

まず、Guild の場合はある三種類の色の光を決め、Wright の場合はある三種類のスペクトルを決めた。そして可視帯域の各スペクトルと同じ色にみえるように三種類の光の量を被験者に調節してもらった。こうして、図 2.8 のような各スペクトルの色を三つの値で置き換えるための関係を得た。

こうして我々はスペクトルデータから 3 次元の色座標系に変換して色を表現する事が出来るようになった。二人の実験をもとに 1931 年に CIERGB 座標系が勧告された。図 2.9 が CIERGB の等色関数である。

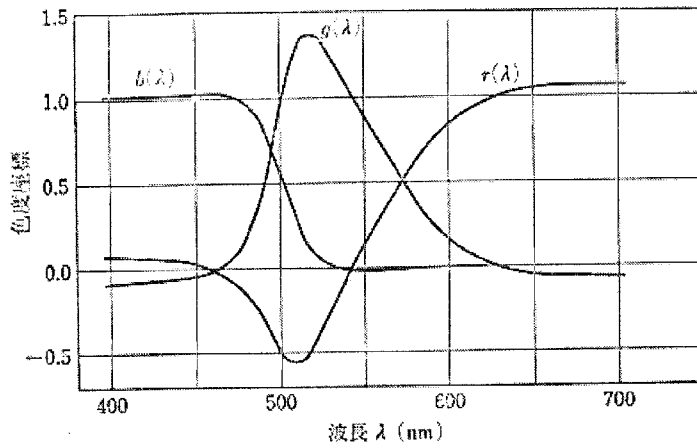
**[4]XYZ coordinate system**

CIERGB 表色系には使用上、不便な点があった。それは色を表現するにあたり負の値を使わなくてはならない点である。そこで、座標系を行列変換する事により負の値をとらないような新たな座標系が考え出された。さらにその座標系では一つ座標軸は明るさ、すなわち輝度のみを表すように定義された。この新しい座標系が CIEXYZ と呼ばれるものである。

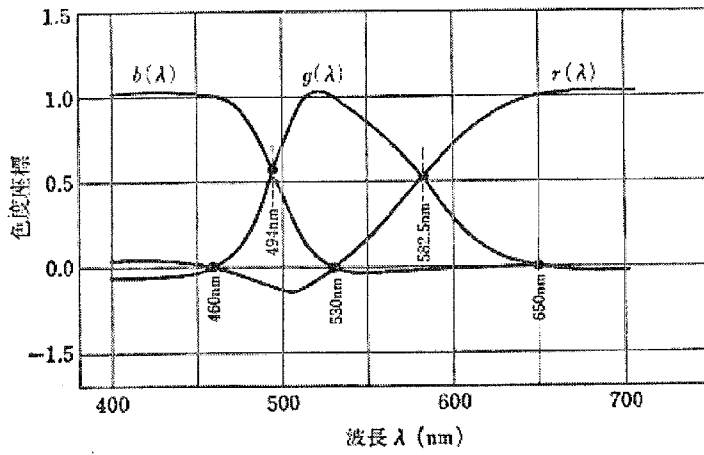
CIEXYZ 表色系は CIERGB 表色系から次のような行列を用いて変換できる。

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.49000 & 0.31000 & 0.20000 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00000 & 0.01000 & 0.99000 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (2.10)$$

また、図 2.9 の等色関数を上の式を用いて各スペクトル毎に変換することによりスペクトルデータから直接的に CIEXYZ 表色系に変換するための等色関数を図 2.10 のように得ることが出来る。現在この等色関数は ISO においても標準化されている。



(a) Guild の等色実験によるスペクトル光の色度座標



(b) Wright の等色実験によるスペクトル光の色度座標

図 2.8 等色実験結果

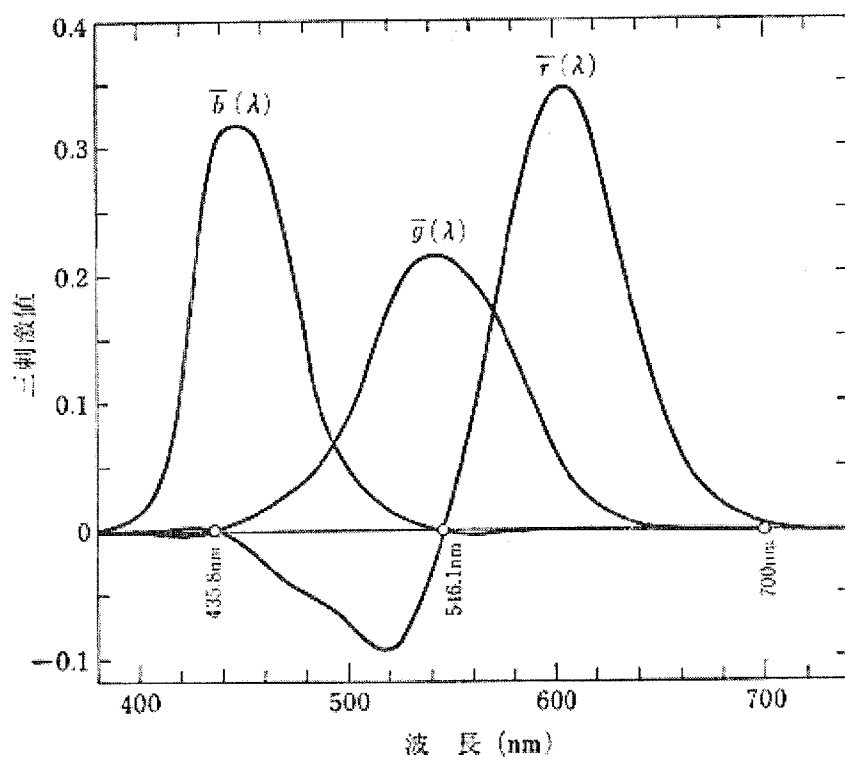


図 2.9 CIE 1931 rgb 系の等色関数

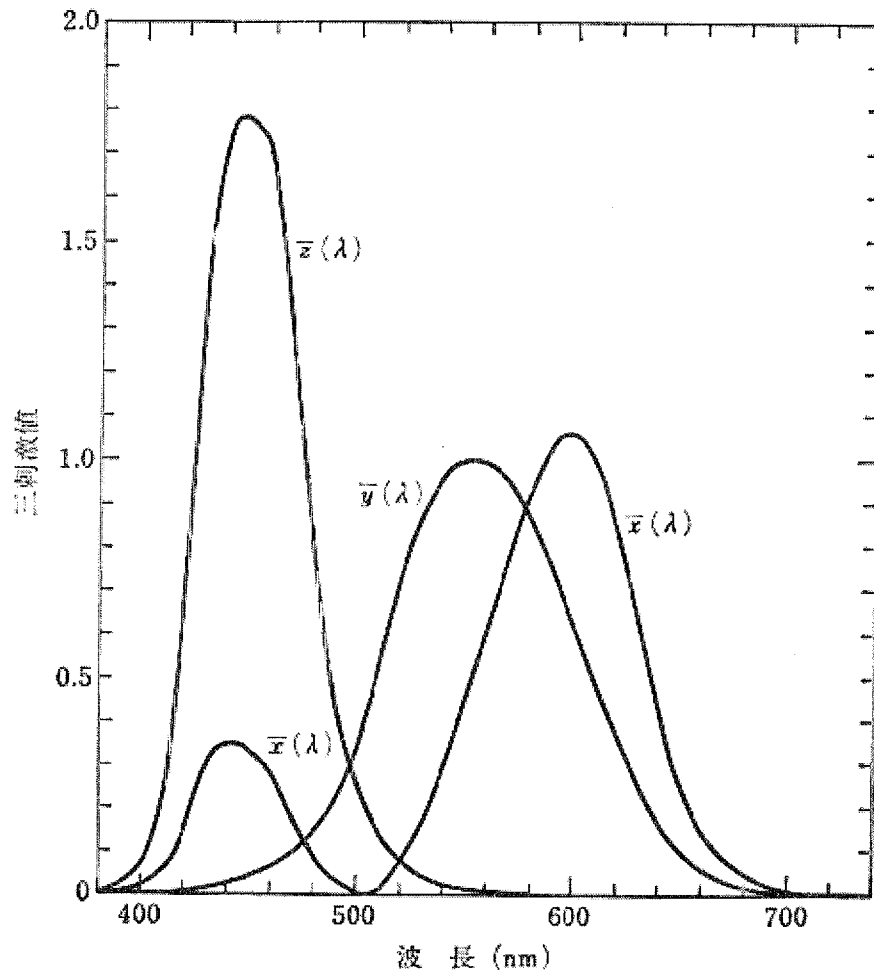


図 2.10 CIEXYZ 表色系の等色関数

## [5]xy 色度図

図 2.11 のように二つのスペクトルがあった場合、この二つの色の XYZ 座標は方向が同じで、長さが違うだけである。これは即ち二つの色は輝度が違うだけで色としての他の特徴、彩度や色相は同じであると考えられる。そこでベクトルの長さを無視して方向のみの違いを扱える図が考え出された。

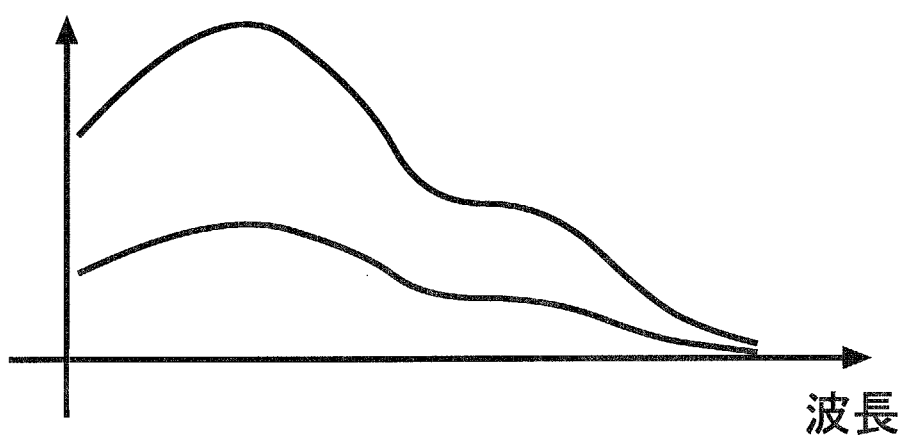


図 2.11 二つの同じ色に見える光

XYZ 空間である色ベクトル  $k(c_x, c_y, c_z)$  がある時、原点を通りその色ベクトルを方向ベクトルとする直線と  $x + y + z = 1$  の平面との交点を考えれば、式 (2.11) のように方向が同じベクトルは全て同一の交点を持つことがわかる。ただし、このままでは平面上の点を三次元座標で表しているのが不便である。そこで、 $x + y + z = 1$  の平面を  $Z$  を無視して  $XY$  平面に射影する。こうしてある色ベクトル  $(c_x, c_y, c_z)$  を式 (2.12) のように輝度以外の色の特徴を調べるために適した二次元平面に変換する手法が得られる。

$$\begin{aligned} x &= \frac{k}{k} \frac{c_x}{c_x + c_y + c_z} \\ y &= \frac{k}{k} \frac{c_y}{c_x + c_y + c_z} \\ z &= \frac{k}{k} \frac{c_z}{c_x + c_y + c_z} \end{aligned} \quad (2.11)$$

$$x = \frac{c_x}{c_x + c_y + c_z} \quad (2.12)$$

$$y = \frac{c_y}{c_x + c_y + c_z}$$

以上の変換を図であらわすと、図 2.12 になる。新たに投影されて得られた面で色を表したものが、 $xy$  色度図である。

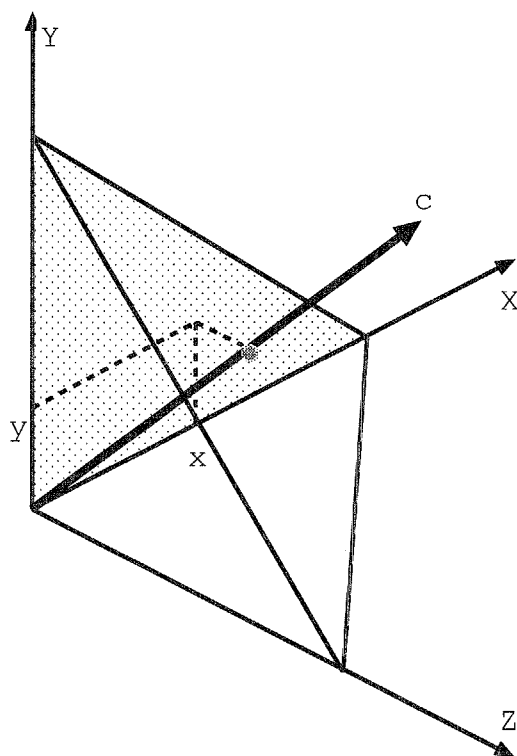


図 2.12 CIEXYZ 空間から  $xy$  色度図への変換

#### [6] CIELAB 座標系

CIEXYZ 座標系では二点間の距離と色の違い、すなわち色差とがかけはなれていた。マンセル表色系のような等色差空間を計算から導き出せれば便利である。

そこで、座標系でのユークリッド距離と色差ができるだけ一致するような座標系が 1976

年に作られた。CIELAB 座標系は物体色に対して標準として決定された等色差座標系の一つである。

CIEXYZ 座標系からの変換式は以下のとおりである。

$$\begin{aligned} L^* &= 116\left(\frac{Y}{Y_0}\right)^{1/3} - 16 \left(\frac{Y}{Y_0} > 0.008856\right) & (2.13) \\ a^* &= 500\left(\left(\frac{X}{X_0}\right)^{1/3} - \left(\frac{Y}{Y_0}\right)^{1/3}\right) \\ b^* &= 200\left(\left(\frac{Y}{Y_0}\right)^{1/3} - \left(\frac{Z}{Z_0}\right)^{1/3}\right) \end{aligned}$$

ただし、ここで、 $X_0, Y_0, Z_0$  は基準白色の値である。

この座標系では色差  $\Delta E^*$  は式 2.14 のようにユークリッド距離で計算できる。

$$\Delta E^* = ((\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2) \quad (2.14)$$

この座標系の特徴は色相を円形に彩度をその半径方向にとるマンセル座標系による等色差の色を  $a^*, b^*$  座標ではほぼ同心円状に表現する事が出来る点である。また、 $L^*$  は輝度に関する色差を表している。ただし、この座標系では式 2.13 で複雑な変換をしている事から、座標内でのベクトルの加法が色の加法にはならない。

#### [7] CIELUV 座標系

もう一つの等色差座標系である CIELUV 座標系も物体色に対して定義されている。

CIEXYZ 座標系からの変換式は以下のとおりである。

$$\begin{aligned} L^* &= 116\left(\frac{Y}{Y_0}\right)^{1/3} - 16 \left(\frac{Y}{Y_0} > 0.008856\right) & (2.15) \\ u^* &= 13L^*(u' - u'_0) \\ v^* &= 13L^*(v' - v'_0) \end{aligned}$$

ただし、

$$\begin{aligned} u' &= \frac{4X}{X + 15Y + 3Z} & (2.16) \\ v' &= \frac{9Y}{X + 15Y + 3Z} \end{aligned}$$

この座標系の特徴は CIELAB と比較して変換の容易性と等色差領域が広い点である。

なお、標準等色差座標系が二つある事からもわかるように完全な等色差座標系は現在決定されていない。よって CIELUV と CIELAB は用途に応じて使い分ける必要がある。また物体色にのみ使う事が出来るという点も忘れてはならない。

ただし発光色に対して、 $u'/v'$  は輝度に対して独立であるので  $u'/v'$  座標系を用いる事は可能である。

### [8] 三次元色座標空間の限界

確かに我々の目のセンサとしての特性から色座標空間は 3 次元で表す事が望ましい。では、色の混色を考えた場合、常に 3 次元空間でのみ考えるだけで良いかが問題となる。

加法混色の場合三次元座標上のベクトルの和の計算により色の混色計算が行えた。しかし、減法混色の場合はそのように単純に色計算を行うことが出来ない。

そのような場合はスペクトルデータに戻り、色を計算することが有効であると考えられる。

### 2.4.2 Kubelka-Munk の光のエネルギー計算モデル

本項では、塗られた絵の具の層の内部での光の挙動をモデル化するために用いる Kubelka-Munk の理論 [75][68][69] [70] について説明を行う。

Kubelka と Munk は薄膜の層内の光に対する特性を吸収と散乱によってモデル化した。このモデル化により薄膜の厚みと反射率の関係を式で表すことに成功している。

はじめに図 2.13 で示すように絵の具層内部における上下方向へのある分光エネルギーを  $j, i$  として、それらの絵の具の微小厚における変化量を散乱率  $S$  と吸収率  $K$  としたとする。すると  $i, j$  の変化量は式 (2.17) で表される。

$$\begin{aligned} -di &= -(S + K)idx + Sjdix & (2.17) \\ dj &= -(S + K)jdx + Sidix \end{aligned}$$

次に、式 (2.17) を式 (2.18)(2.19) のように変形する。

$$\begin{aligned} -\frac{di}{i} &= -(S + K)dx + S\frac{j}{i}dx, & (2.18) \\ \frac{dj}{j} &= -(S + K)dx + S\frac{i}{j}dx \end{aligned}$$

$$\frac{dj}{j} - \frac{di}{i} = d \log \frac{j}{i} = -2(S + K)dx + S\left(\frac{i}{j} + \frac{j}{i}\right)dx \quad (2.19)$$

さらに、 $j/i$  を反射率  $r$  と置き換えて、式 (2.20) のように変形を行うと式 (2.21) が導出される。



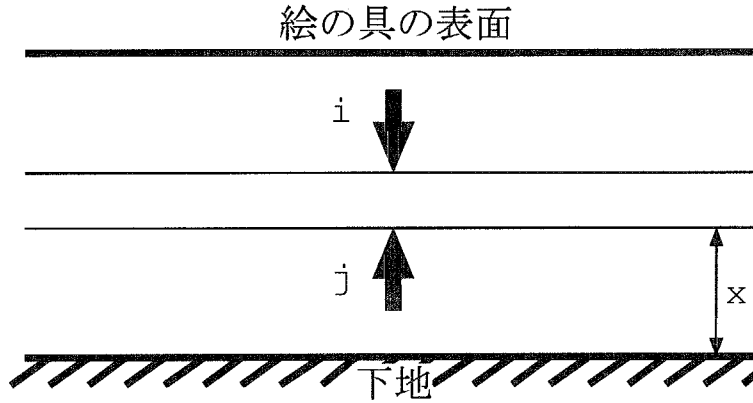


図 2.13 Kubelka Munk の式の原理図

$$\begin{aligned} d \log r &= \frac{dr}{r} = -2(S + K)dx + \left(\frac{1}{r} + r\right)Sdx \\ dr &= (Sr^2 - 2(S + K)r + S)dx \end{aligned} \tag{2.20}$$

$$\int \frac{1}{r^2 - 2\frac{S+K}{S}r + 1} dr = S \int dx \tag{2.21}$$

さらに、次の二組の境界条件を与えることにより、積分計算を行う。

$x$	$r$	意味
$X$	$R$	厚みが $X$ の時の反射率を $R$ とする
$0$	$R_u$	厚みが $0$ の時の反射率は下地の反射率 $R_u$ とする

積分計算の前に計算を容易にするため、式 (2.22) のように定数を置き換える。積分計算過程は式 (2.23) のようになり、式 (2.24) が導出される。

$$\frac{S + K}{S} = 1 + \frac{K}{S} = a \tag{2.22}$$

$$\begin{aligned} \int \frac{1}{(r - a - \sqrt{a^2 - 1})(r - a + \sqrt{a^2 - 1})} dr &= S \int dx \\ \int \frac{1}{2\sqrt{a^2 - 1}} \left( \frac{1}{r - a - \sqrt{a^2 - 1}} - \frac{1}{r - a + \sqrt{a^2 - 1}} \right) dr &= S \int dx \end{aligned}$$

$$\int \frac{1}{r-a-\sqrt{a^2-1}} dr - \int \frac{1}{r-a+\sqrt{a^2-1}} dr = 2S\sqrt{a^2-1} \int dx \quad (2.23)$$

$$\left[ \log(r-a-\sqrt{a^2-1}) \right]_{R_u}^R - \left[ \log(r-a+\sqrt{a^2-1}) \right]_{R_u}^R = 2S\sqrt{a^2-1}X$$

$$\log \frac{(R-a-\sqrt{a^2-1})(R_u-a+\sqrt{a^2-1})}{(R_u-a-\sqrt{a^2-1})(R-a+\sqrt{a^2-1})} = 2S\sqrt{a^2-1}X \quad (2.24)$$

式(2.24)は任意の厚み  $X$  で成り立つので、 $X = \infty$  の時、左辺も無限大になることから、厚みが無限大の時の反射率  $R_\infty$  が式(2.25)のようになることがわかる。

$$R_\infty = a - a\sqrt{a^2-1} = 1 + \frac{K}{S} - \sqrt{\frac{K^2}{S^2} + 2\frac{K}{S}} \quad (2.25)$$

さらに式(2.25)の  $a$  と  $R_\infty$  の関係から、式(2.26)を経て、式(2.27)のように  $a$  と  $\sqrt{a^2-1}$  を  $R_\infty$  で表す。

$$R_\infty^2 - 2aR_\infty + a^2 = a^2 - 1 \quad (2.26)$$

$$a = \frac{1}{2} \left( \frac{1}{R_\infty} + R_\infty \right) \quad (2.27)$$

$$\sqrt{a^2-1} = \frac{1}{2} \left( \frac{1}{R_\infty} - R_\infty \right)$$

最終的に式(2.27)を式(2.24)に代入した式(2.28)を変形をすることにより厚み  $X$  と反射率  $R$  の関係式(2.29)が求まる。

$$\log \frac{(R - \frac{1}{R_\infty})(R_u - R_\infty)}{(R_u - \frac{1}{R_\infty})(R - R_\infty)} = SX \left( \frac{1}{R_\infty} - R_\infty \right) \quad (2.28)$$

$$R = \frac{\frac{1}{R_\infty}(R_u - R_\infty) - R_\infty(R_u - \frac{1}{R_\infty})e^{SX(\frac{1}{R_\infty} - R_\infty)}}{(R_u - R_\infty) - (R_u - \frac{1}{R_\infty})e^{SX(\frac{1}{R_\infty} - R_\infty)}} \quad (2.29)$$

以上が Kubelka-Munk の理論による薄膜の厚みと反射率に関する式の導出法である。本節の色の工学的な表現方法の説明及び、Kubelka-Munk の理論の説明は、第5章及び第6章での議論において必要となる。

## 2.5 むすび

本章では、まず提案されている様々な手描き風の画像の作成法について述べた。

描画の表現方法は多岐に渡る。しかし、テクニカルイラストや白黒の描画を除けば、描画法の多くが印象派の油絵風である。一方で薄い油彩画や水彩画の研究は少ない。この原因の一つには混色の計算が困難だからであると考えられる。従って Curtis ら [35] の研究はその中で特異な存在である。絵の具の混色に関しては描画結果の質に大きく影響を与えるため第5章で詳しく述べる。また、画像を入力とした描画法では、全体として描画に荒さが目立つ。これは画像処理の技法が容易ではないからである。この点に関しては第3章で本論文の提案手法について述べる。

2.3節では、一般的な画像処理に関する研究について述べた。これに対して第3章では描画向きの画像処理法について述べる。

2.4.1項では色を工学的に扱うための基礎知識についてまとめて説明を行った。この知識をもとに第5章では発色、色差の検討を第6章では、スペクトル反射率と色との変換アルゴリズムについて述べる。

2.4.2項では Kubelka-Munk の理論について説明を行った。この理論は絵の具の混色計算において必要とされる。第5章では、理論の実装法について述べる。

## 第 3 章

# 手描き風画像自動生成のための画像処理手法

### 3.1 まえがき

本章では、第 4 章で述べる写真を入力とし、それを計算機によって手描き風の画像に自動的に再構成することにより新たな雰囲気画像を作成する手法のための準備として、入力画像の持つ特徴を解析する画像処理手法について述べる。

与えられる入力画像は、データ形式そのものとしてはピクセルの色情報の配列でしかない。即ち、データ形式から見た場合、計算機は画像を非常に微小な領域毎にしか認識していないことになる。そこで、画像解析を行い、入力画像の持つより広域的な情報を得る必要がある。

画像解析の手法には物体認識のような画像を全体的に捉えることを目標とするものも存在するが、本章で述べるのは、ピクセルよりは広域的であるが、輪郭情報や局所方向性の情報等の画像中の極めて局所的な領域に基づいた情報を得る画像処理手法についてである。これは、再描画を行う際に、画像の構図の変更や対象物体の変形を行わないのであれば高度な物体認識は必要ないとの判断からである。

本章では、始めに自動描画によって作成する画像の特徴の位置付けを行う。続いて、そのような画像の描画に必要となる情報を得るための画像解析手法について述べる。具体的には、2次元FFT(2D-FFT)を局所的に用いて得られる振幅マトリックスから局所方向性、方向信頼度、周波数期待値を得る方法、及びディファレンシャル-ガウスフィルタを用いて輪

郭近傍情報を得る方法について述べる。

## 3.2 作成する手描き風画像の位置づけ

第4章で扱う手描き風画像は表現法としては筆やペン等の道具を用いることによって描かれる画像の特徴を持つものである。即ち、3次元世界を2次元へ幾何学的に射影した点列で表される実写と異なり、実際には無い境界線が黒く描かれていたり、細かい領域は一筆で塗りつぶされていたりする等の“画像上の情報の省略や強調”がある表現方法である。

但し人間が実際に物を描く際には、モチーフの持つ雰囲気をもより強調するように、変形することがある。例えば相撲の力士ならば、より力強くやより重量感のあるといったように実際の力士からデフォルメしたり、構図を変形して描く事がある。これは“対象物や対象空間の情報の省略や強調”である。計算機による似顔絵システムの研究[57]ではこの種の処理を行うことを目標としているが、本論文で扱う手描き風画像では入力した写真の内容に踏み込んだ情報の変更は行わない。

以上を図で表すと、図3.1のようになる。

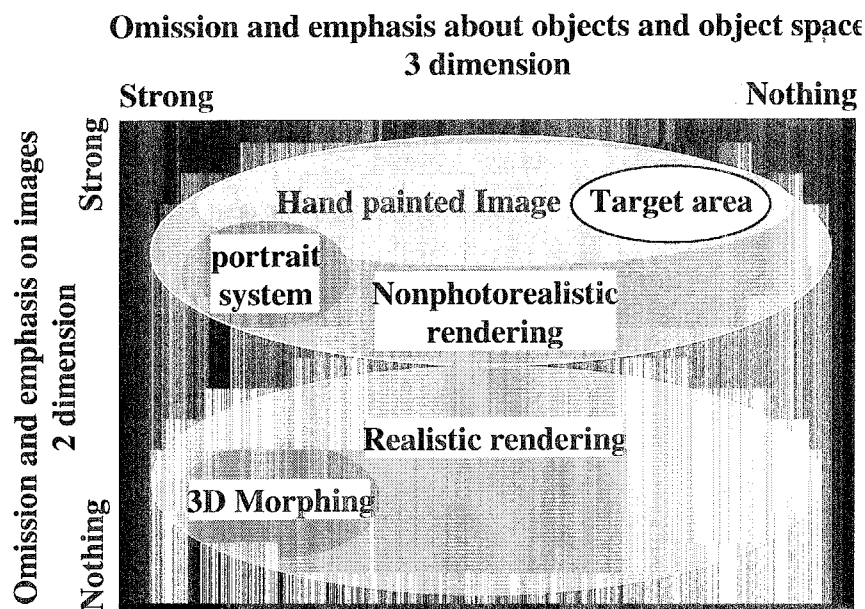


図 3.1 CG の表現法による分類

このように生成画像の対象を明確化することにより、第4章で提案する写真等の自然画像を入力とした手描き風画像の自動作成手法では物体認識処理や空間理解といった高度な認識は必要ないことが分る。

しかし、画像上の局所的な特徴の一部は手描き風の画像に再構成する際に変更したり、強調したりするため、対象画像の局所的な特徴量をあらかじめ検出する必要がある。そこで、次節からは画像の局所的な特徴量を検出する手法について述べる。

尚、第4章で述べる描画手法は、コンピュータグラフィックス全体の表現法の分類からはノンフォトリアリスティックレンダリングの一部に含まれる。

### 3.3 2D-FFTを用いた画像解析手法

本節では、高速フーリエ変換を局所的に施すことにより画像の局所的な情報を得る手法について述べる。

#### 3.3.1 局所方向性検出

局所方向性が輝度勾配から生じるとしてガウスグラディエント演算によって方向性検出を行う手法については第2章で既に考察を行った。

ガウスグラディエント演算による方向性解析では、ガウス演算が平滑化の作用をするために細かい縞模様によって生じる方向性の検出は難しい。それは縞模様の断面の勾配が相殺されるからである。これはまた、ガウスグラディエント関数のインパルス応答がバンドパス型であることから理解できる [80]。

さらに、ガウス関数型のフィルタの応答は対象領域の中心部に強く依存するため、後処理として、方向の平均操作を必要とする。

そこで本節では細かい縞模様をも検出して方向検出を行う手法について述べる。細かい縞、すなわち高周波成分から滑らかな輝度変化による方向検出を行う為に矩形正方窓関数を用いた局所二次元フーリエ級数展開を行い方向性を検出する手法を検討する。

周波数特性に優れたガウス関数を窓関数として用いずに矩形正方関数を窓関数として用いるため、高周波数成分まで検出が可能となる反面、問題も生じる。この点については、実際の検出手順の中で対処法について述べる。

方向性検出を行う処理の手順は次のとおりである。

はじめに前処理としてカラー画像画像の各ピクセルの輝度を求める。次に各ピクセルに対して以下の操作を行う。

- (1) 対象ピクセルを中心とする正方領域に対して二次元フーリエ級数展開を行う。出来得る限り局所的な方向性の検出を行うために窓関数正方領域は狭いことが望ましい。しかし、狭い底の矩形関数を窓関数として用いると周波数領域では値を平滑化してしまい細かい縞模様の検出は難しい。また、フーリエ級数展開のアルゴリズムにはFFTを用いるため、領域の大きさは $2^n \times 2^n$ である必要がある。本論文では方向検出の分解能を考え、窓関数の正方領域の大きさを $n = 4$ 、 $16 \times 16$ とする。
- (2) 二次元フーリエ級数展開の結果から中心に直流成分を置く振幅マトリックスを作成する。振幅マトリックスの各元と波形の関係を図3.2に示す。振幅マトリックスは標本化定理によって決まるナイキスト限界の周波数成分以外は直流成分を原点とする原点对称に分布する。

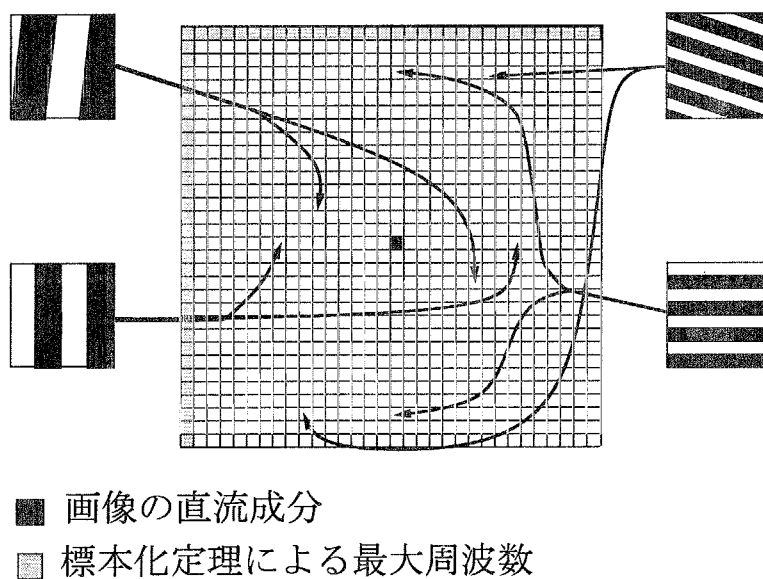


図 3.2 振幅マトリックスの要素と波形の関係

ここで、一般的に生じる振幅値の偏りに対して補正を行う。振幅マトリックスの値の分布は等方的でなく、水平、垂直方向の波形に対応する値が他の方向に対応する値に比べ大きくなるのである。この偏りはフーリエ級数展開の対象領域が矩形正方窓関数の底の幅を基本周期とする周期関数でないことから生じる。窓関数の底の幅の整数倍

を周期とする波形の場合、矩形関数のフーリエ変換  $\text{sinc}$  関数の特性から振幅マトリックスの対応する元にもみ値が出る。しかし、基本周期の整数倍から外れた周期の波形の場合、その周波数成分は二次元  $\text{sinc}$  関数と畳み込みによって周辺の振幅マトリックス成分に現れる。ここで、二次元  $\text{sinc}$  関数の異方性、フーリエ変換を“級数”展開で行っていることによる水平、垂直方向と斜め方向の周波数空間上でのサンプリング間隔の違い、及び離散データの元来持つ異方性が原因となり図 3.2 の振幅マトリックスの値に偏りが生じる。そこで、振幅マトリックスの水平、垂直周波数成分に対応する値を  $w$  で割ることにより他の成分に対する値の優位性を下げる。

$w$  は水平及び垂直成分に対応する振幅マトリックスの要素の平均値を  $a$  として、その他の要素の平均値を  $b$  とした場合、式 (3.1) で表される。ここで、 $A$ 、 $B$  は定数であり、0.6、10 とする。

$$w = \min\left(\frac{aA}{b}, B\right) \quad (3.1)$$

(3) 図 3.3 に示すように振幅マトリックス上で直流成分を通る様々な傾きの直線を引き、傾き  $d \in D$  の時の直線上の振幅成分の和を傾き成分と呼び、 $l_d$  とする。 $l_d$  の集合を  $L$  とする。

(4)  $L$  の中で一番大きな値に対応する傾きを対象ピクセルの方向成分とする。

以上の手順で各ピクセルの局所方向性を求めることができる。

### 3.3.2 局所方向性信頼度

局所方向性信頼度とは局所領域内において、検出された方向性に対応する方向量に対して他の方向量が相対的にどの程度の大きさであるかを表す値である。この値が小さい場合は局所領域内には単一の強い方向量がないと見なして、手描き風画像描画時のストローク長を短くする。

局所方向性検出で得られた  $\theta_d$  および  $l_d$  を使い局所方向性信頼度を定義する。 $\theta_d (d \in D)$  の取る値の範囲は  $-\pi/2$  から  $\pi/2$  である。そこで、角度が  $2 \times \theta_d$  であり大きさが  $l_d$  のベクトルを考えて、それらすべてのベクトル和  $V$  を求める。この  $V$  は、直交する方向に対応する  $l_d$  が逆方向に打ち消されてできているという性質を持つ。よって、最後にベクトルの角度 ( $-\pi \sim \pi$ ) を半分にすることで平均的な角度と呼べる値  $ave_\theta$  およびその方向量  $ave_l$  が求まる。



$$V = (V_x, V_y) = \sum_{d \in D} l_d (\cos 2\theta_d, \sin 2\theta_d) \quad (3.2)$$

$$ave_l = |V| \quad (3.3)$$

$$ave_\theta = \frac{\left\{ \tan^{-1} \left( \frac{V_y}{V_x} \right) \right\}}{2} \quad (3.4)$$

$ave_\theta$  と局所方向性  $I_\theta$  の角度差および  $ave_l$  と局所方向性に対応する方向量  $l$  を使って、式 (3.5) で局所方向信頼度  $I_r$  を定義する。

$$I_r = \frac{ave_l}{l} \cos(ave_\theta - I_\theta) \quad (3.5)$$

(3.5) 式は、射影の式になっている。 $I_r$  が大きい場合は局所領域内に検出した方向性に近い方向成分が多く存在していることを表している。逆に、この値が小さい場合は格子状の画像のような局所領域内に検出した方向性以外の方向成分が大きいことを表す。

### 3.3.3 局所波長期待値検出

この値は 3.3.1 項の振幅マトリックスの振幅値からその局所領域の波長の期待値を求めるものであり、描画の際にストロークの幅を決定するために利用される情報である。

振幅マトリックスの各要素を  $m_{ij}$ 、各要素に対応するユークリッド波長を  $len_{ij}$  とした時、局所波長期待値  $I_s$  を (3.6) 式で表すことにする。

$$I_s = \frac{\sum_{i=1}^M \sum_{j=1}^M m_{ij} len_{ij}}{\sum_{i=1}^M \sum_{j=1}^M m_{ij}} \quad (3.6)$$

局所波長期待値が大きい時はエッジ近傍のような画像上の意味のある領域であることを表し、逆に小さい時はノイズが多く相対的に意味がないことを表す。

## 3.4 輪郭近傍情報検出

自動描画において塗りつぶし作業を行う場合、輪郭情報が必要になる。

しかし、今日あらゆる条件で完全な輪郭検出法は提案されていない [83]。そこで実際には、実用面からある程度輪郭検出が行える zero-cross 法やソーベル法が多く用いられている。

zero-cross 法とは、輝度情報を二階微分してゼロと交差する位置を輪郭位置とする方法である。実際の検出においてはガウスラプラシアンフィルタを用いてゼロ交差を求める。また、人間の視覚特性に関する研究では、視覚に関する細胞である X 型神経節細胞の受容野が  $\nabla^2 G$  で近似出来ることを示している [84]。

ただし、やや滑らかな輝度変化をする輪郭では検出結果と輪郭位置がずれることが報告されている点、ノイズ除去の為にフィルタに広いガウス関数を用いると検出されるゼロ交差位置と輪郭がずれる点から、本論文における輪郭検出法ではゼロ交差法を採用しない。

本論文では、(3.7) 式に示すソーベルフイルタ型の  $15 \times 15$  ガウス-グラディエントフィルタ  $f_v, f_h$  を rgb の画像  $(I_r, I_g, I_b)$  に畳込むことによって輪郭の近傍らしさについての情報  $I_o$  を (3.8) 式により求める。

$$\begin{aligned} f_v(i, j) &= \frac{\partial}{\partial i} \exp\left(\frac{-(7-i)^2 - (7-j)^2}{2T^2}\right) \\ f_h(i, j) &= \frac{\partial}{\partial j} \exp\left(\frac{-(7-i)^2 - (7-j)^2}{2T^2}\right) \end{aligned} \quad (3.7)$$

$$T = 1.5616, \quad 0 \leq i, j < 15$$

$$\begin{aligned} I_o &= \sqrt{f_v * I_r^2 + f_h * I_r^2} \\ &+ \sqrt{f_v * I_g^2 + f_h * I_g^2} \\ &+ \sqrt{f_v * I_b^2 + f_h * I_b^2} \end{aligned} \quad (3.8)$$

$I_o$  の数値は、輪郭部分に近づくにつれて相対的に大きな値を示す。

## 3.5 局所情報検出結果

3.3 節において述べた局所情報検出法の結果について述べる。

### 3.5.1 局所方向性検出結果

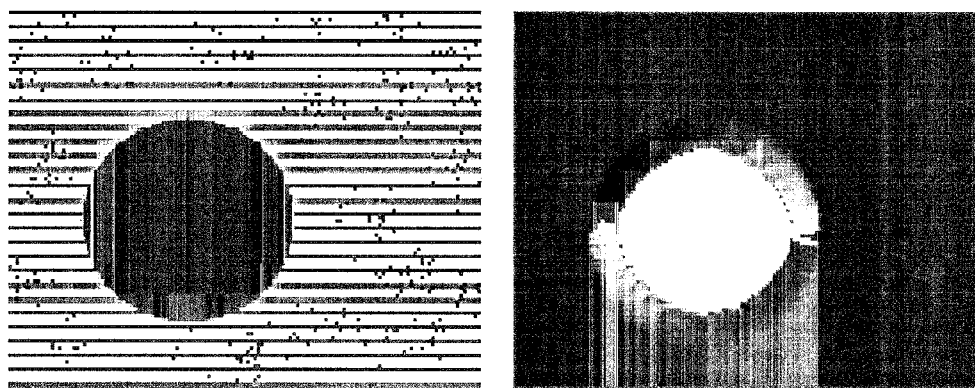
幾何学的な図形に対して従来法 [59] と本手法によって方向検出した結果を図 3.4 に色の濃さで示す。本手法においても一部誤った方向検出が行なわれているが、全体としてノイズを含んだ縞模様部分において従来法と比較して明らかな検出精度の改善が見られる。これは、縞情報から方向性を検出しているためである。

また、従来法では円の輪郭と背景の縞模様の方向性が平均化されて検出されてしまっているが、本手法での結果では方向性の平均化がされることなく、輪郭の方向が輪郭の部分が検出できていることが分る。

従来法による検出結果の方が、円の輪郭部分の持つ方向性に関してより局所的に検出できている。しかしながら、従来法のように局所方向性が輪郭から遠ざかるにつれ滑らかに変化する検出結果を描画の情報に使った場合、背景から輪郭へと続くストロークを描いてしまう可能性が生じる。従って、本手法のような輪郭部からフィルタの半径を越えて遠ざかると急激に方向が変わる局所方向性の検出結果がより描画においては望ましいと考えられる。

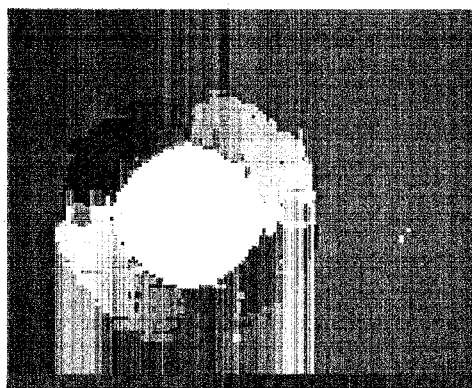
さらに、図 3.5 の自然画像に対して局所情報検出を行った例を図 3.6 に示す。画像サイズは  $370 \times 300$  ピクセルである。流れる背景の方向性、自転車の車輪内部、人間の輪郭等の方向性が検出されている事がわかる。これより、自然画像に対して本手法は適応が可能であることが分る。





(a) 対象画像

(b) 従来手法の結果



(c) 本手法の結果

図 3.4 従来法と本手法との方向性検出結果の比較

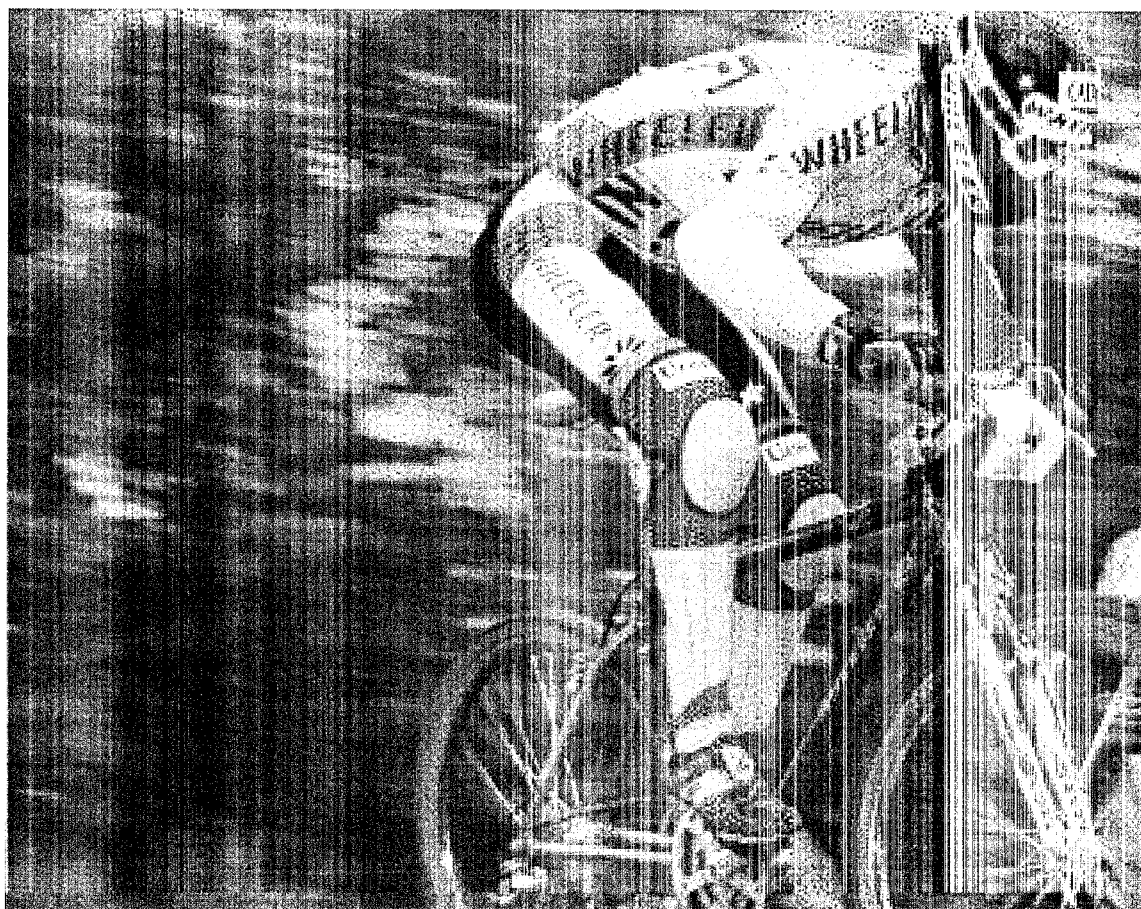


図 3.5 自然画像サンプル“じてんしゃ”

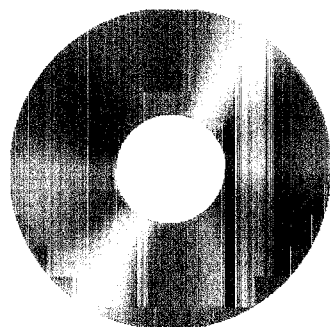
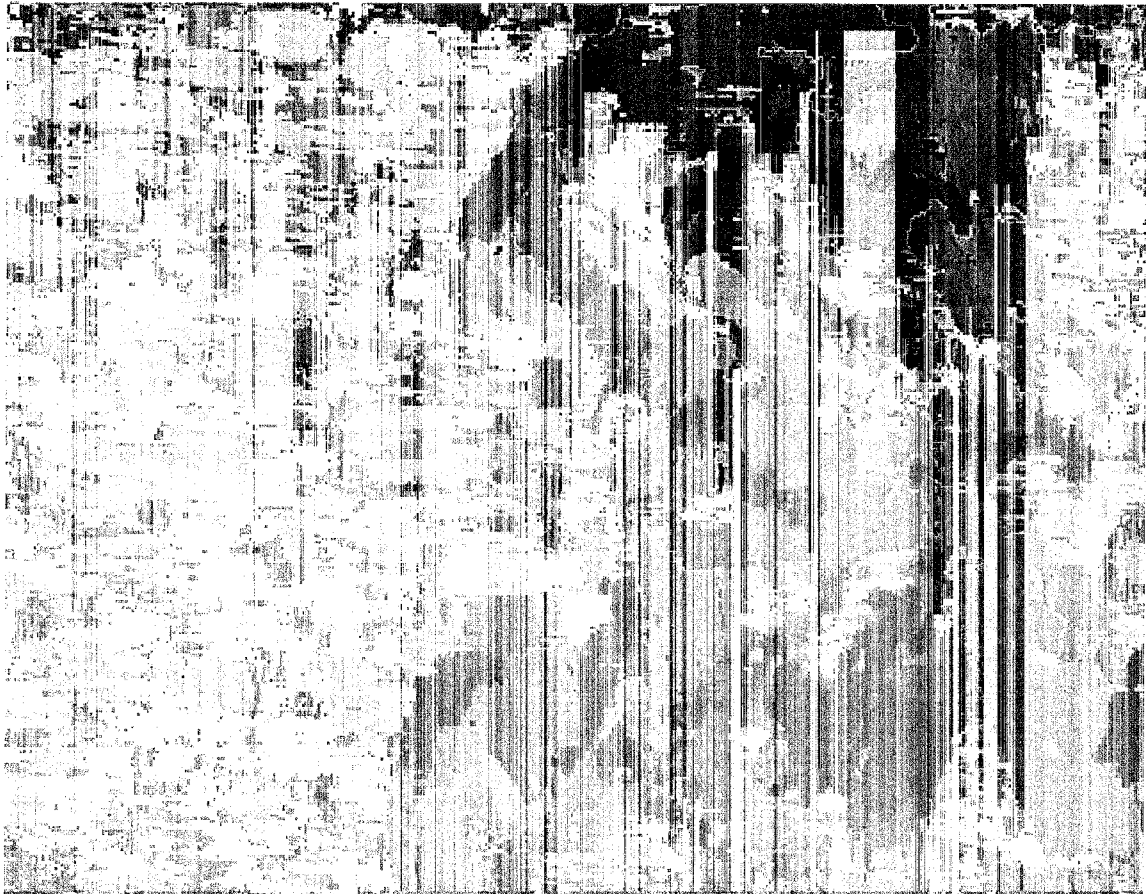


図 3.6 自然画像に対する局所方向性検出結果の例

### 3.5.2 局所方向性信頼度検出結果

図 3.7 は自然画像図 3.5 に対する方向信頼度の値を濃淡で表している。値が大きいほど白く描いている。局所方向性の検出結果において検出方向が変化する近くの領域において値が小さくなっていることが分る。

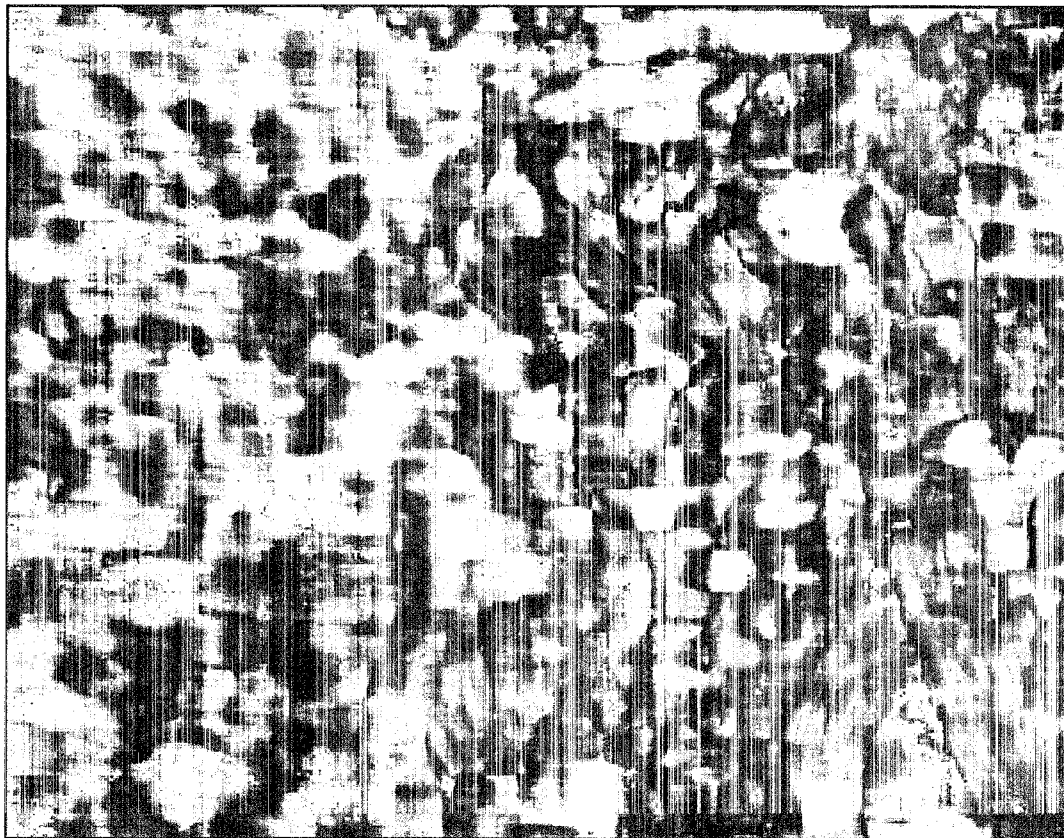


図 3.7 方向信頼度検出例



### 3.5.3 局所波長期待値検出結果

図 3.8 は図 3.5 の自然画像に対する周波数期待値の値を濃淡で表している。値が大きいほど白く描いている。対象画像の中で意味情報の少ない領域である背景下部や単一色に近い所では値が小さく、輪郭の近くでは値が大きくなっていることが分かる。

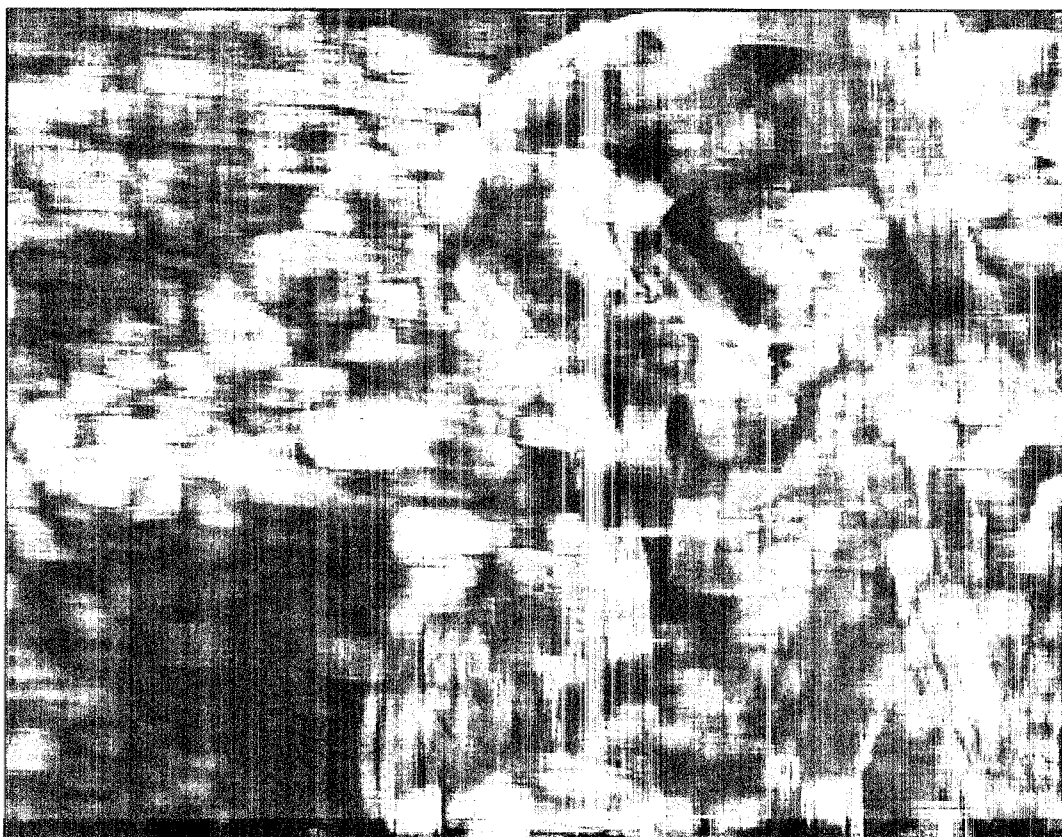


図 3.8 局所周波数期待値検出例

## 3.5.4 輪郭近傍情報検出結果

図 3.9 は図 3.5 の自然画像に対する輪郭近傍情報値を濃淡で表している。値が大きいほど白く描いている。輪郭に近づくにつれて値が相対的に大きくなっている事が分る。

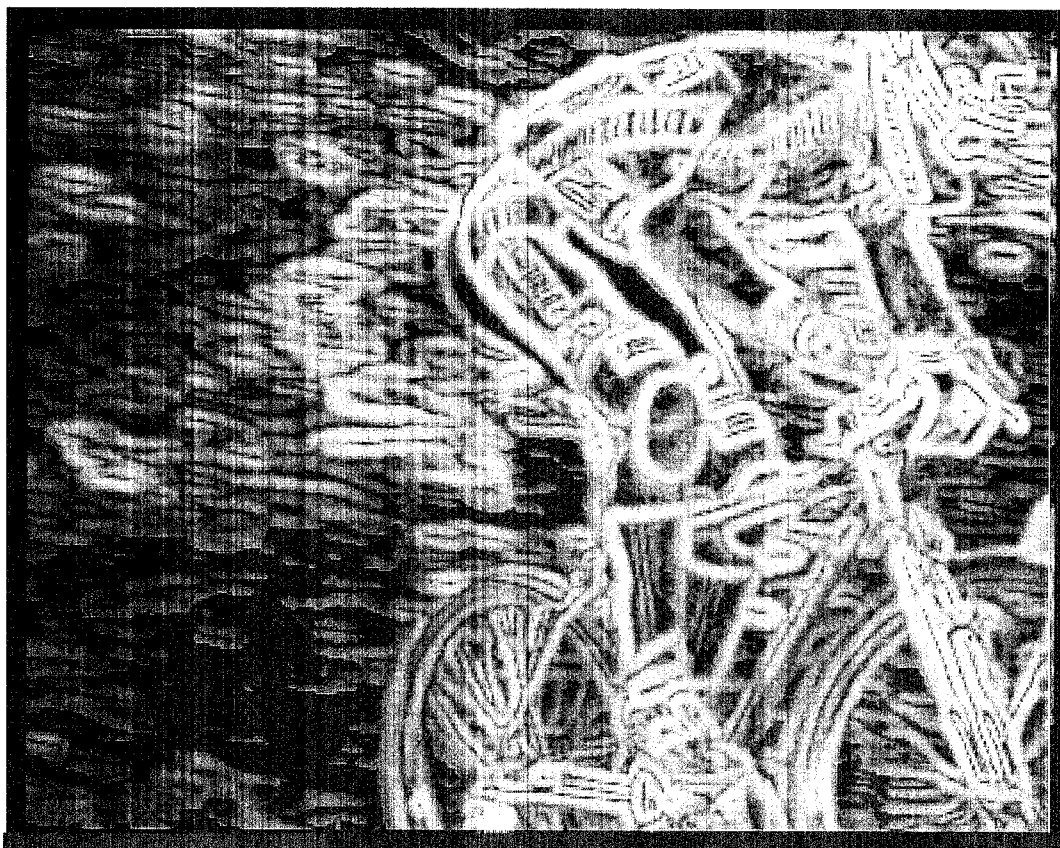


図 3.9 輪郭近傍情報検出例

## 3.6 むすび

本章では、次章で述べる手描き風画像の描画法の画像的特徴の位置づけを行い、手描き風描画には画像の局所的情報が必要であることを述べた。

具体的には、局所方向性情報とそれに付随する情報の検出法、輪郭近傍情報の検出法について述べた。

実際の描画でのストロークの方向は局所方向性に沿って行われると考えられるので、局所方向性情報は特に重要であるといえる。従来法と比較して、輪郭から遠ざかるに従って急激に局所方向性情報が変化する本手法の結果は描画アルゴリズムでのストロークの止めにおいて有効な特徴であると考えられる。

輪郭近傍情報は輪郭に近づくにつれて大きな値を持つものである。現在の輪郭検出法では必ずしも正確に輪郭が得られるとは限らない。従って、二値化や細線化のプロセスを経て定義される輪郭情報と比較して本章で述べた輪郭近傍情報は完全な誤りが少ないと考えられる。

描画アルゴリズムにより再構成された画像は人間が見るためのものであり、対象画像で人間にとって不明確に見える部分、もしくは人間が補間する部分に関しては描画アルゴリズムによっても同様に不明確に描かれて良いと考えられる。従って、そのために必要とされる画像の局所情報も一般的なロボットビジョン等の研究における目的と異なり、不鮮明な部分を明確にする必要はない。逆に不鮮明な部分は不鮮明であるとした情報の方が好ましい。

本章で述べた局所方向性情報を除く強度を表す局所情報では連続値を取るようになっており、手描き風画像に適した情報となっているといえる。

## 第 4 章

# 手描き風画像自動生成法

### 4.1 まえがき

第3章において入力画像の局所的な画像の特徴を得ることができた。その情報を利用して新たな手描き風の画像を作成する手法について本章では述べる。

本章で述べる描画方法は、大きく二つに分類できる。一つは、筆による描き込みを模した描画法であり、他の一つはぼかしを基本とした描画法である。

筆による描き込みを模した描画方法では、局所情報に従い、筆のストロークの幅、長さ、曲り具合を決定し、次にそれらを描画領域にランダムに描き込むことにより画像を作成している。ストロークの描画順番はランダムであるが、各ストロークの形状が対象画像の局所的な特徴に従っているため意味の分る画像を描くことができる。

ぼかしを基本とした描画法では、局所情報に従って、微小な“ぼかしフィルタ”を各領域毎に変形させて、それらを複数回畳み込むことによりぼかしを行っている。小さなフィルタの畳み込みを複数回行うことにより巨大なフィルタと同等な計算を行っているのであるが、各微小フィルタが局所領域の画像の特徴に対応しているため、単純なぼけた画像でない手描き風の画像を作成することが出来る。

実際に処理結果を示し、どちらも画像の局所情報に従うことにより入力画像と雰囲気異った、何が描かれているのかわかる手描き風画像を作成することが出来ることを示している。

## 4.2 ストローク描画による基本描画手法

ここでは、ストロークの形状が描画結果に強く残る描画法を真似た画像作成アルゴリズムについて述べる。

基本的には、原画像と同じ大きさの白色領域を始めに用意し、次にランダムに選んだピクセルを起点とするストローク領域を決定し、その領域を単一色で塗りつぶし、次第に白色領域全体を塗り潰すことによって描き終えるという単純な考え方である。

ストローク領域確保は、ストロークの背骨となるカーブを決定するステップと背骨カーブを太くするステップの二つからなる。

背骨カーブは次のような手順で決定する。

- (1) 画像中のピクセルをランダムに選ぶ。
- (2) 選ばれたピクセルから、そのピクセルの局所方向性に沿った2方向に距離1ピクセル分ずつカーブを延ばす。この際、カーブの先端座標は実数値で現す。
- (3) カーブの先端に対応するピクセルの局所方向性に沿って更にカーブを延ばす。ただし、局所方向性が $\pi/2$ 以上の急激な変化をする場合もしくは、先端ピクセルの局所方向性信頼度が低い場合はカーブを延ばさない。

このようにして決定されたカーブは実数値の座標列としては急激な方向変化はないが、ピクセル列に変換した場合はより丸め誤差により鋭角な変化が現れるので、スムーズ処理を行う。

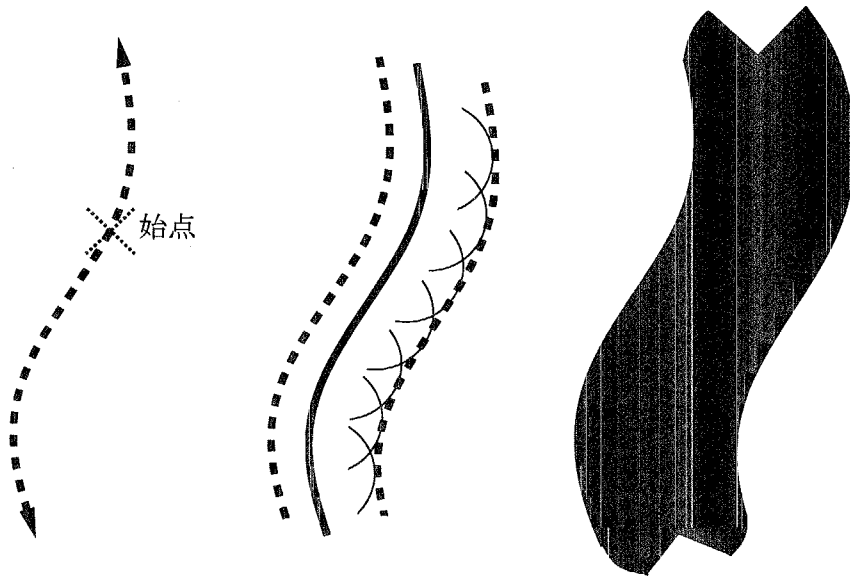
細いカーブを太くするアルゴリズム [62] はいくつか存在するが、今回は徐々に太線化していくため、最も簡単な対象カーブ上を中心の軌跡とする円盤を求める方法により線を太くするアルゴリズムに改良を加えて、背骨カーブを太線化する。

カーブの片側を太くする手順を説明する。

- (1) 半径は1ピクセルとする。太さ決定変数の値を0とする。
- (2) カーブの進行方向に対して図4.2に示す様に、 $\pi/4$ から $\pi 3/4$ の範囲の与えられた半径の円弧の軌跡を描く。背骨カーブはスムーズ処理により $\pi/2$ 以上の急激な変化はないので、円弧の軌跡でピクセルの虫食いになること無しに太線化が行なえる。

太さ決定変数に円弧の軌跡に対応する各ピクセルの局所波長期待値の平均値を加える。

- (3) 太さ決定変数がある閾値を越えない場合、もしくは半径がまだ最大半径に達していない場合は、半径を1増加させて項目2に戻りさらに太くする。



(1)背骨カーブの決定 (2)カーブの太線化 (3)ストローク領域の決定

図 4.1 太線化アルゴリズム

カーブの反対側に対しても同様な処理を行なって太くする。

円盤の軌跡ではなく、円弧の軌跡を使って太線化しているので、比較的太線の外側のピクセルのみの局所波長期待値の平均を手順2で求めることが出来る。

局所的な領域の状況にストロークの形状を適応させるという意図から、カーブの左右独立の太線化や起点ピクセルからの2方向への背骨カーブの延長を行なっている。

ストロークの代表色は、背骨カーブに対応する各ピクセルの色の  $rgb$  の各値の平均値をとる色とする。

白色画像の各ピクセルが2回以上、ストロークによって塗り重ねられるように描き込んで終了する。本描画法では、各ピクセルの色は後から塗り重ねた色が採用される。混色が一切考慮されていないため、各ストロークの形状が明確に最終描画結果に残る描画法である。

なお、ストロークの描画順は全くランダムとしている。

### 4.3 単純減法混色計算を付加したストローク描画手法

次に、簡易的な重ね塗りによる混色を計算して描画を行う方法について述べる。

ストローク領域の決定は上の油絵風画像作成アルゴリズムと同様である。ただし、ストローク領域を決定する起点ピクセルは一度も描き込まれていないピクセルに限っている。また、ストロークの背骨線のピクセルの色平均から求められた色  $(A_r, A_g, A_b)$  から次の式を使ってペンの色  $(P_r, P_g, P_b)$  を決定する。

$$\begin{aligned} P_r &= 1 - (1 - A_r)/3 \\ P_g &= 1 - (1 - A_g)/3 \\ P_b &= 1 - (1 - A_b)/3 \end{aligned} \quad (4.1)$$

混色には、単純な減法混色を用いる。即ち、下地の色を  $(S_r, S_g, S_b)$ 、上から塗る色を  $(U_r, U_g, U_b)$  とした時、新たな色  $(N_r, N_g, N_b)$  を次のように変更する。

$$\begin{aligned} N_r &= S_r U_r \\ N_g &= S_g U_g \\ N_b &= S_b U_b \end{aligned} \quad (4.2)$$

ランダムに行われるストローク描画は全面が描き終われば終了とする。

式(4.1)によりあらかじめ塗る色を対象画像の色よりも明るくすること、塗り重ね回数を制限することにより、単純減法混色で生じる色の濁りに対処し、原画像の色合いに近いサインペンで描かれたような画像の作成を行う。

### 4.4 簡易的な混色計算を付加したストローク描画手法

4.2節及び4.3節では、ストロークの形状を生成画像に明確に残して手描き風の画像を作成した。しかし、水彩や薄い油絵の具で描いた場合は塗り潰しの部分では各ストロークの形状が絵に残らない。

本節では混色計算を用いて各ストロークの形状を目立たなくすることによって水彩や薄い油絵の具によって描かれたような画像を生成するアルゴリズムについて述べる。

描画方法の基本は4.2節で用いたストロークによるランダムな埋めつくしと同様である。ただし、本節ではストロークの形状を目立たなくする処理を加え、塗り潰しの描画を再現する。

## 4.4.1 簡易絵の具モデル

準備として、簡単な色計算が行える絵の具モデルを考える。

絵の具の混色は減法混色と異なっている。フィルタの混色の場合には減法混色であり、透過する光のエネルギーはフィルタの透過率を積算すれば求まる。しかし、塗られた絵の具の層の内部では、光は散乱により光路長が著しく変化する。そのため、塗り重ねた各層の透過率を単純に積算をしたのでは、反射光が実際よりも暗くなってしまう。

そこで、絵の具層に下層への光を遮断する遮へい率を定義し、図4.2のように光路長に変化を付けたモデルを考える。

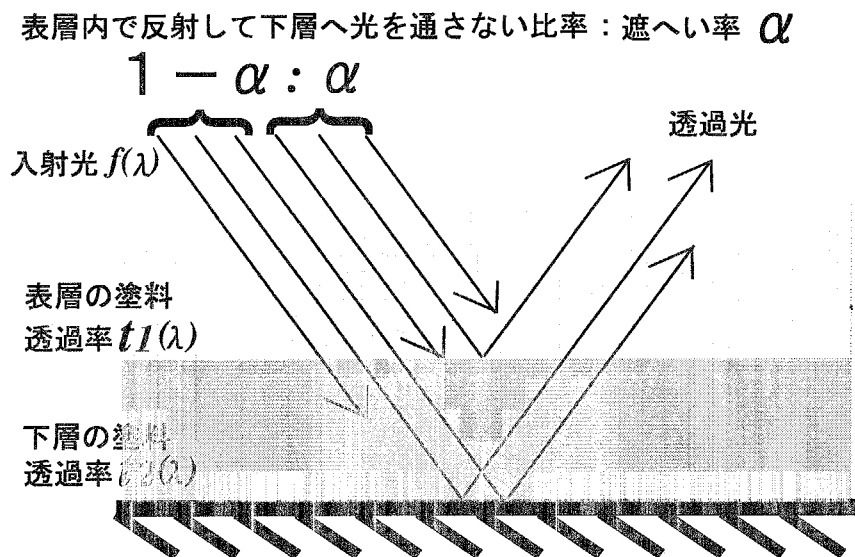


図 4.2 塗り重ねによる混色における遮へい率と光路の関係

遮へい率を考慮した上塗りによる混色計算を式(4.3)に示す。ただし、 $(R, G, B)$ を混色結果、 $(T_r, T_g, T_b)$ 、 $(B_r, B_g, B_b)$ を上層、下層の透過率、 $(w, w, w)$ を入射光とし、 $\alpha$ を遮へい率とする。入射光を最大輝度の白色とすれば、透過率と入射光の積が色として簡単に計算可能である。

$$R = (\alpha + (1 - \alpha)B_r)T_r \times w$$



$$G = (\alpha + (1 - \alpha)B_g)T_g \times w \tag{4.3}$$

$$B = (\alpha + (1 - \alpha)B_b)T_b \times w$$

次に、完全に乾いていない状態の絵の具の上にさらに塗った場合の混色を考える。

この場合、絵の具の層は一部多層にならず混ざりあって一つの層となると考える。

そこで攪拌によって出来る新たな絵の具の層について考える。

下層の絵の具の乾いていない部分が新たに塗った絵の具と混ざるはずである。絵の具の攪拌の結果生じる新たな透過率に関して、表面積にしめる絵の具の顔料比率は混ざり合う絵の具の混合比であるとして、ここでは図4.3のように考える。

混ざった絵の具は常に一定量が塗られ、新たに塗る厚みに変化はないものとする。これは、一定量以上は筆に吸い取られてしまうとの仮定からである。

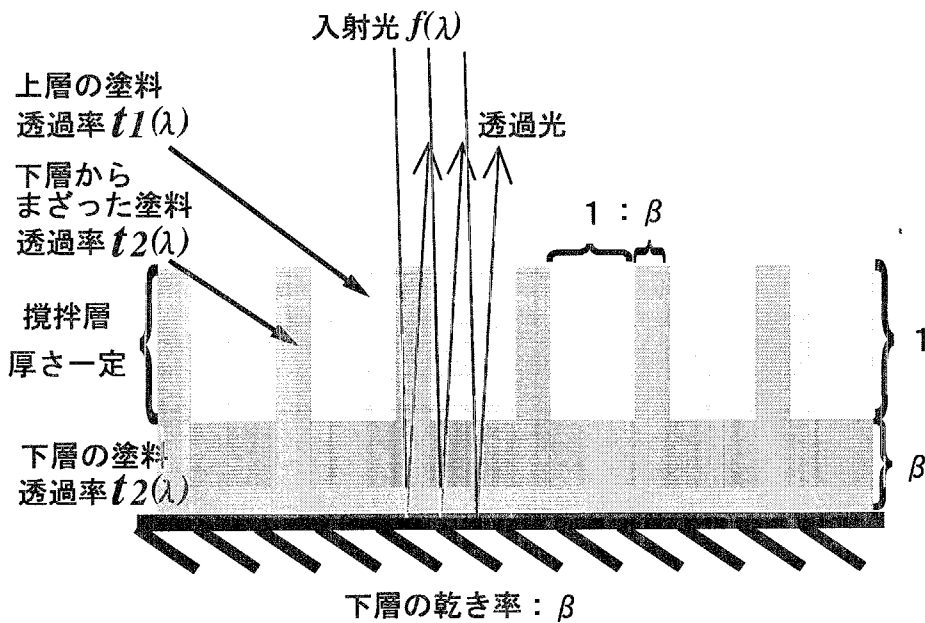


図 4.3 完全に乾いていない上への重ね塗りによる透過率、光路モデル

以上を式で表すと、次のように混色  $(R, G, B)$  が計算出来る。

ただし、 $(T_r, T_g, T_b)$ 、 $(B_r, B_g, B_b)$  を上層色、下層色、 $(w, w, w)$  を入射光、 $\alpha$  を遮へい率、 $\beta$  を下層の乾き率とする。

$$\begin{aligned}
 R &= (\alpha + (1 - \alpha)B_r^{1-\beta}) \left( \frac{T_r + \beta B_r}{1 + \beta} \right) \times w \\
 G &= (\alpha + (1 - \alpha)B_g^{1-\beta}) \left( \frac{T_g + \beta B_g}{1 + \beta} \right) \times w \\
 B &= (\alpha + (1 - \alpha)B_b^{1-\beta}) \left( \frac{T_b + \beta B_b}{1 + \beta} \right) \times w
 \end{aligned} \tag{4.4}$$

#### 4.4.2 簡易混色計算を行う描画手法

輪郭部分でないところはいくつものストロークによって塗りつぶされ、各ストロークの色は攪拌による混色が行われると考えられる。そこで、輪郭近傍情報をもとに輪郭から遠い領域では攪拌混色を行い、輪郭近傍では攪拌処理を行わないこととして、描き込みの疑似的な順位付を行う。

また輪郭近傍では遮へい率を下げ、薄く塗り重ねた効果を出すように、輪郭近傍情報をもとに輪郭に近づくにつれて遮へい率の値を下げる。

実際の描画手順は、ストローク領域確保及び、ランダム描画に関しては前章の方法と同様である。絵の具の混色に関してはストロークを重ねるピクセルごとに遮へい率、乾き率を以下のように計算し混色を行う。

着目ピクセルの輪郭近傍情報を  $e$  としたとき、そのピクセルの遮へい率  $\alpha$  は式 (4.5) で決定される。

$$\begin{aligned}
 \alpha &= 0.5 + 0.5h(e) \\
 h(x) &= \begin{cases} 0 & T < x \\ 1.0 - x/T & 0 \leq x < T \end{cases}
 \end{aligned} \tag{4.5}$$

ここでは  $T = 30$  としている。遮へい率  $\alpha$  の取る値の範囲を 0.5 から 1.0 として第 3 章で述べたような極端な色の濁りを避けている。

遮へい率と同様、着目ピクセルの輪郭近傍情報を  $e$  としたとき、そのピクセルの乾き率  $\beta$  は式 (4.6) で決定される。

$$\begin{aligned}
 \beta &= d(e) \\
 d(x) &= \begin{cases} x & 0 \leq x < T \\ 1.0 & T < x \end{cases}
 \end{aligned} \tag{4.6}$$

ここでは  $T = 30$  としている。

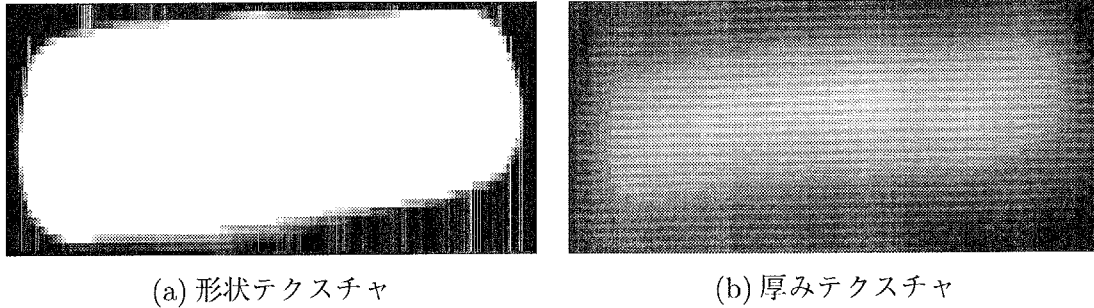


図 4.4 ストロークに用いたテクスチャ

以上で決定される  $\alpha$ 、 $\beta$  を用いて 4.4.1 項 で述べた混色計算を新たに描画される際に各ピクセル毎に行う。

## 4.5 ストロークに光沢表現を付加した描画手法

ストロークを特徴づける要素としてその形状や発色がある。これらについての表現力を簡便に高める方法としてストロークにテクスチャを貼る方法が従来から提案されている [3]。

しかし絵の具の表面光沢はストロークの描かれた状況や、光源に応じて変化するため一般的なテクスチャを用いることでは再現に限界がある。

そこで本節では、ストローク表面に付加するテクスチャの他にストロークの厚みについてのテクスチャデータと、ストロークを描き込む際に絵の具の厚さについて記録する厚みバッファを用意し、4.2 節 で述べた描画法へ光沢処理を付加する。

描画手順は 4.2 節 と同様である。ただし、描画した絵の具の色を記録する色バッファへストロークテクスチャ色を格納すると同時に、絵の具の厚みバッファへストロークの持つ厚みテクスチャを重ね書きする。

ストロークに用いる形状テクスチャ及び厚みテクスチャを図 4.4(a) 及び (b) に示す。

ストロークが描き終わった後に厚みバッファの厚み勾配  $dh/dx, dh/dy$  を用いて、絵の具の表面の法線方向  $(-dh/dx, -dh/dy, 1)$  を決定する。

仮の光源方向から決定される全反射方向のベクトルを  $(r_x, r_y, r_z)$  とする時、光の直接反射強度  $I$  と表面勾配の関係は次式 (4.7) で表すことができる。

$$I = \exp\left(-\arccos\left(\frac{-\frac{dh}{dx}r_x - \frac{dh}{dy}r_y + r_z}{\sqrt{\left(\frac{dh}{dx}\right)^2 + \left(\frac{dh}{dy}\right)^2 + 1}\sqrt{r_x^2 + r_y^2 + r_z^2}}\right)^2\alpha\right)\beta \quad (4.7)$$

ここで  $\alpha, \beta$  は定数である。

本手法はバンプマッピングやディスプレイスマッピングとは異なったストロークに合ったテクスチャ技術の応用であるといえる。

## 4.6 ぼかしを基本とした描画手法

本節ではぼかしを基本とする描画方法を用いて手描き風画像を生成する手法について述べる。

本節の描画方法はエアブラシによる描画や日本画的な描画を目指している。対象とする描画方法では画像の輪郭部分は色変化の激しい輪郭としての特徴を保存するように描かれ、輪郭以外は滑らかな塗り潰しによって画像が構成されているといった特徴を持っている。

自動生成においては、実際の描画方法である吹き付けや筆の塗り重ねを直接的にはモデル化せず、輪郭以外では滑らかに色変化をする点に着目してぼかし処理を基本としたアルゴリズムにより描画を行なう。ただし、3.5.4項で求めた輪郭近傍情報は輪郭情報としては完全ではない。そこで、局所方向性情報をも用いて方向性のあるぼかしを行なう。実際の描画方法は次のとおりである。

- (1) 画像の各画素  $(x, y)$  を中心とする  $3 \times 3$  のぼかしフィルタ  $B(x, y)$  を式 (4.8) で定義する。

$$B_{i,j}(x, y) = \frac{Bt_{i,j}(x, y)}{\sum_{k=-1}^1 \sum_{l=-1}^1 Bt_{k,l}(x, y)} \quad (4.8)$$

$$Bt_{i,j}(x, y) = W \exp\left(\frac{-(j-1) \sin I_\theta}{-2\alpha^2} + \frac{(i-1) \cos I_\theta}{-2\alpha^2}\right) \quad (4.9)$$

ここで、 $I_\theta$  は画素  $(x, y)$  の局所方向性であり、(4.9) 式の  $\exp$  部分は、図 4.5 に示すように局所方向性に沿う方向にのみ強い値を持つように意図している。 $\alpha$  の値は方向性に対して直行する向きに位置するフィルタ要素値がほぼ 0 になるように 0.32 としている。また、 $W$  は画素  $(x+i, y+j)$  における輪郭近傍情報値  $I_o$  から、次式で決定される。

$$W = \begin{cases} 1 - I_o/T_w & (I_o \leq T_w) \\ 0 & (T_w < I_o) \end{cases} \quad (4.10)$$

$T_w = 15$

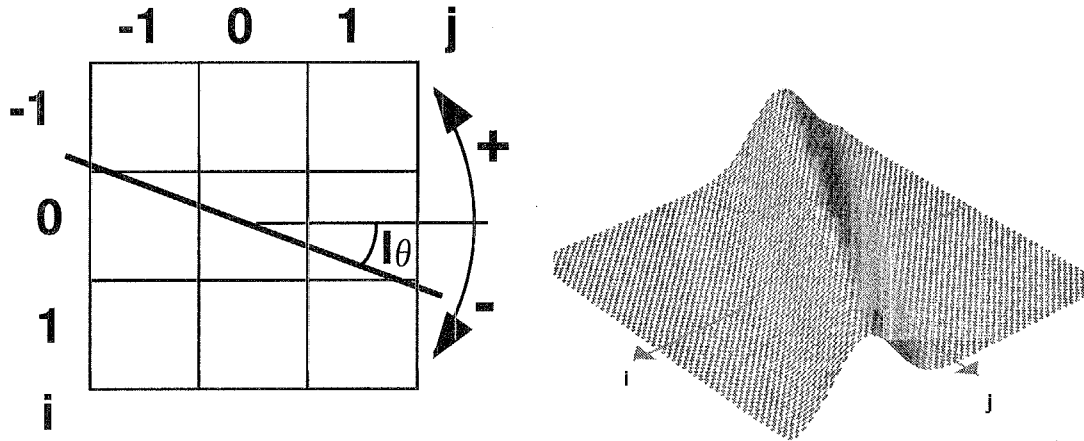


図 4.5 ぼかしフィルタの形状

(4.10) 式により輪郭近傍でのぼかし方を弱めている。なお、 $T_w$  を大きくすれば輪郭部分もぼけた画像となり、小さくすればほとんどぼけない画像になってしまうが、今回の輪郭近傍情報を計測するフィルタ  $f_h, f_v$  の大きさ、および次の項における畳込み回数の組合せでは、多くの自然画像に対して 15 として意図する効果を得ることができる。

- (2) 上で決定した局所ぼかしフィルタ  $B(x, y)$  を用い (4.11) 式により対象画像に畳込み処理を  $n$  回行う。今回の実験では  $n = 30$  とする。

$$I_{result} = I * \underbrace{B * \dots * B}_n \quad (4.11)$$

以上の処理により、目的とする描画を行う。本描画法では  $3 \times 3$  フィルタの畳込み操作を 30 回繰り返すことにより  $61 \times 61$  フィルタを 1 回畳込んでいることと同等な処理を行うことになる。従って各領域の特徴に適応的なかなり広いぼかし処理であるといえる。

## 4.7 各手法による描画結果

本節では前節までに述べた各描画法による描画結果を示す。

### 4.7.1 ストロークによる基本描画法の描画結果

図 4.6、図 4.8 はそれぞれ第 3 章の図 3.5、及び図 4.7 に対するストロークによる基本描画法による画像生成結果である。図 4.7 の画像サイズは  $512 \times 512$  ピクセルである。図 4.9 は図 4.6 の描画における描画の途中経過を示している。

これらの結果から油絵風の画像が作成できていることが分る。

画像の局所的な情報を用いて各ストロークの形状決定を行っているため、ストロークの描画順は図 4.9 のようにランダムに行われているにもかかわらず、描かれた画像は原画像の持つ画像的特徴を保存していることが分る。対象画像によって異なるが図 4.8 の場合、最大ストロークの面積は幅  $10 \times$  長さ  $36$  であり、各ストロークとほぼ等面積の正方形で画像を再構成した場合の図 4.10 の画像からわかるように各ストロークは決して小さくはない。しかしながら、各ストロークが局所情報に適した形状をしており、その局所情報が人間の感覚と近いことから、何が描かれているかわかる画像を作成可能となっていることが分る。

しかし、一部のストロークでは、はみ出して描かれていような描画がされている箇所が存在する。その原因は第 3 章の手法で得た局所情報が人間の感覚と異なっている場合があるためと考えられる。従って、より細やかな描画作成には画像解析の検討が必要といえる。

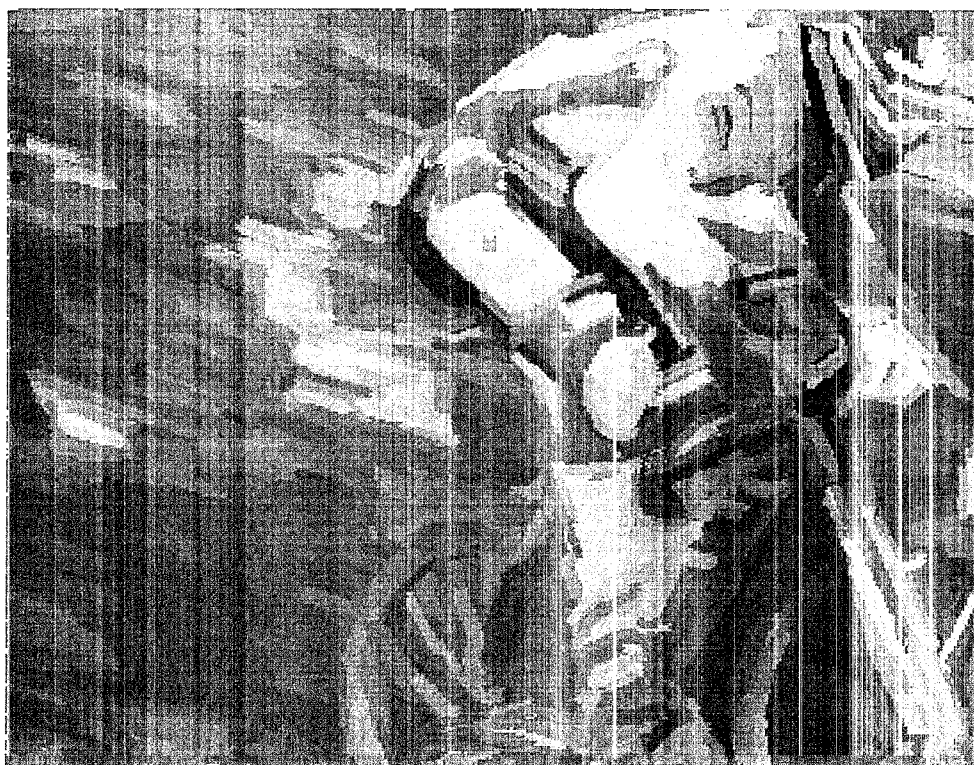


図 4.6 “じてんしゃ” に対するストローク基本描画手法描画結果



図 4.7 対象画像 “くだもの”



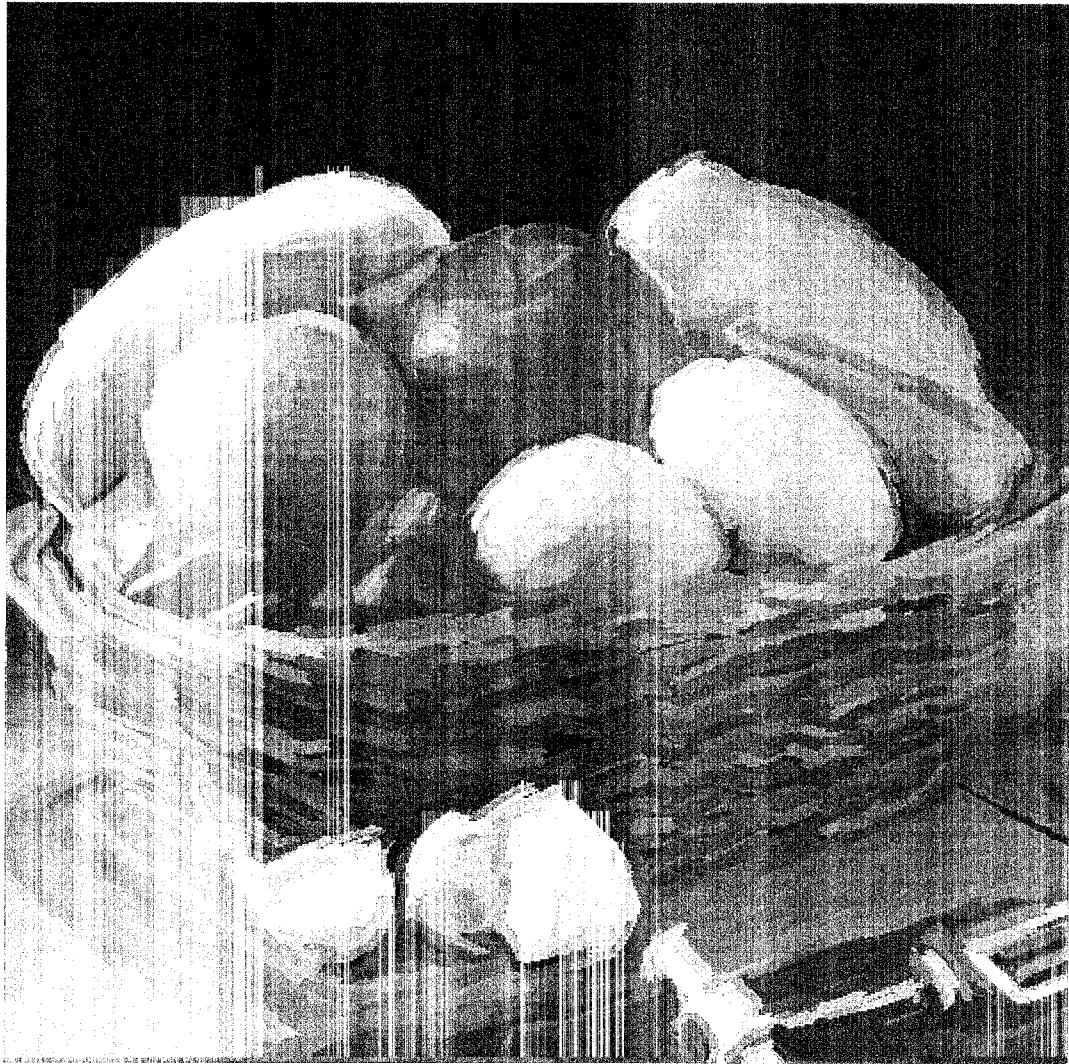


図 4.8 “くだもの” に対するストローク基本描画手法描画結果



図 4.9 “じてんしゃ”に対する描画の途中経過

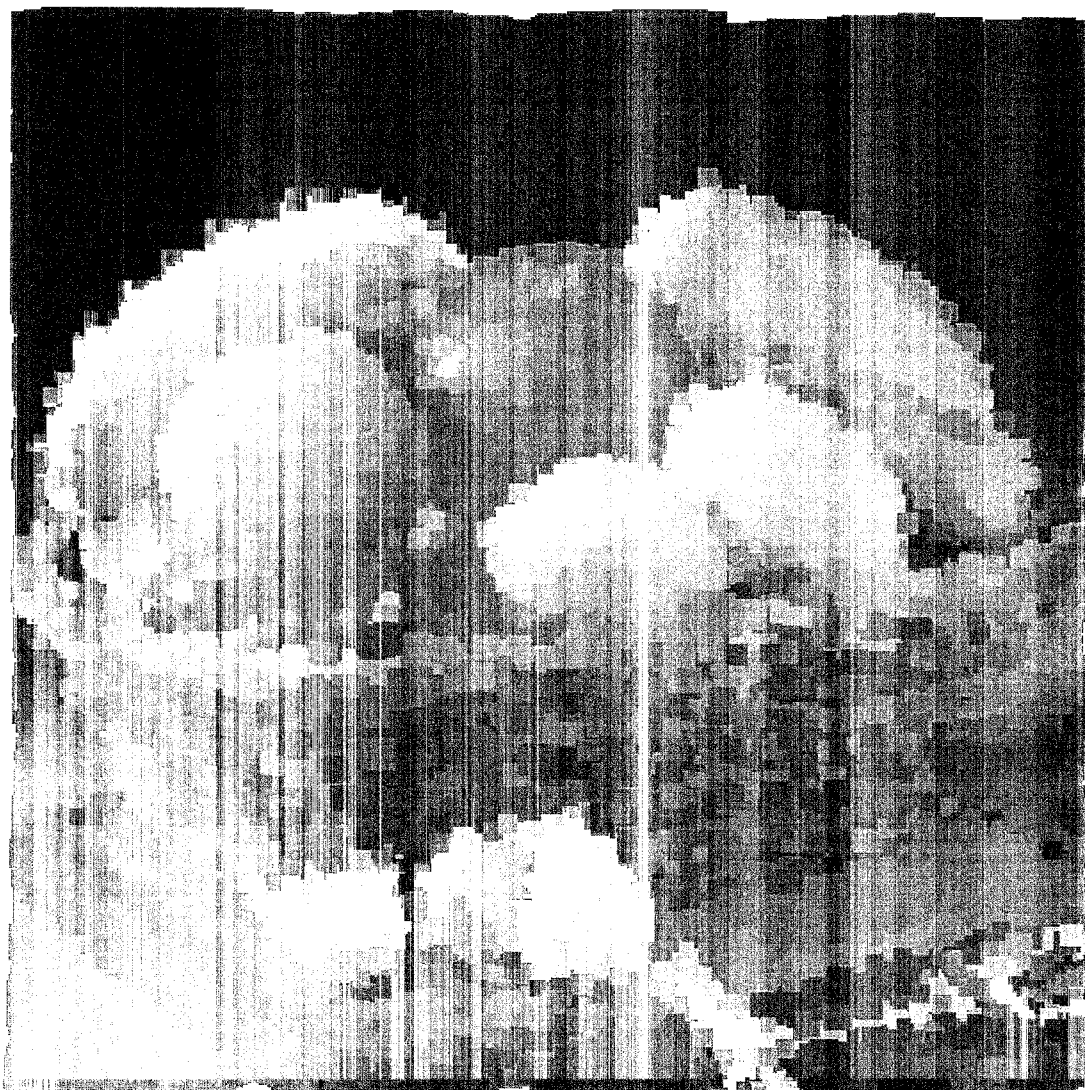


図 4.10 ストロークの面積と画像の大きさ比較

#### 4.7.2 単純減法混色計算を付加したストローク描画手法の描画結果

図 4.12 と図 4.13 はサインペン風画像の生成例である。図 4.12 の原画像は図 4.11 である。サインペンによる塗り重ねの模様が良好に再現されていることがわかる。単純減法混色処理を付加することのみで油絵風画像図 4.8 と図 4.13 のような雰囲気の間違った画像が生成できることがわかる。

単純減法混色では混色によって色の明度は下がるのみである。しかしながらあらかじめ目標の色よりも描画する色を明るくし、塗る回数を制限することにより対象画像とほぼ同じ色の画像を作成することができている。

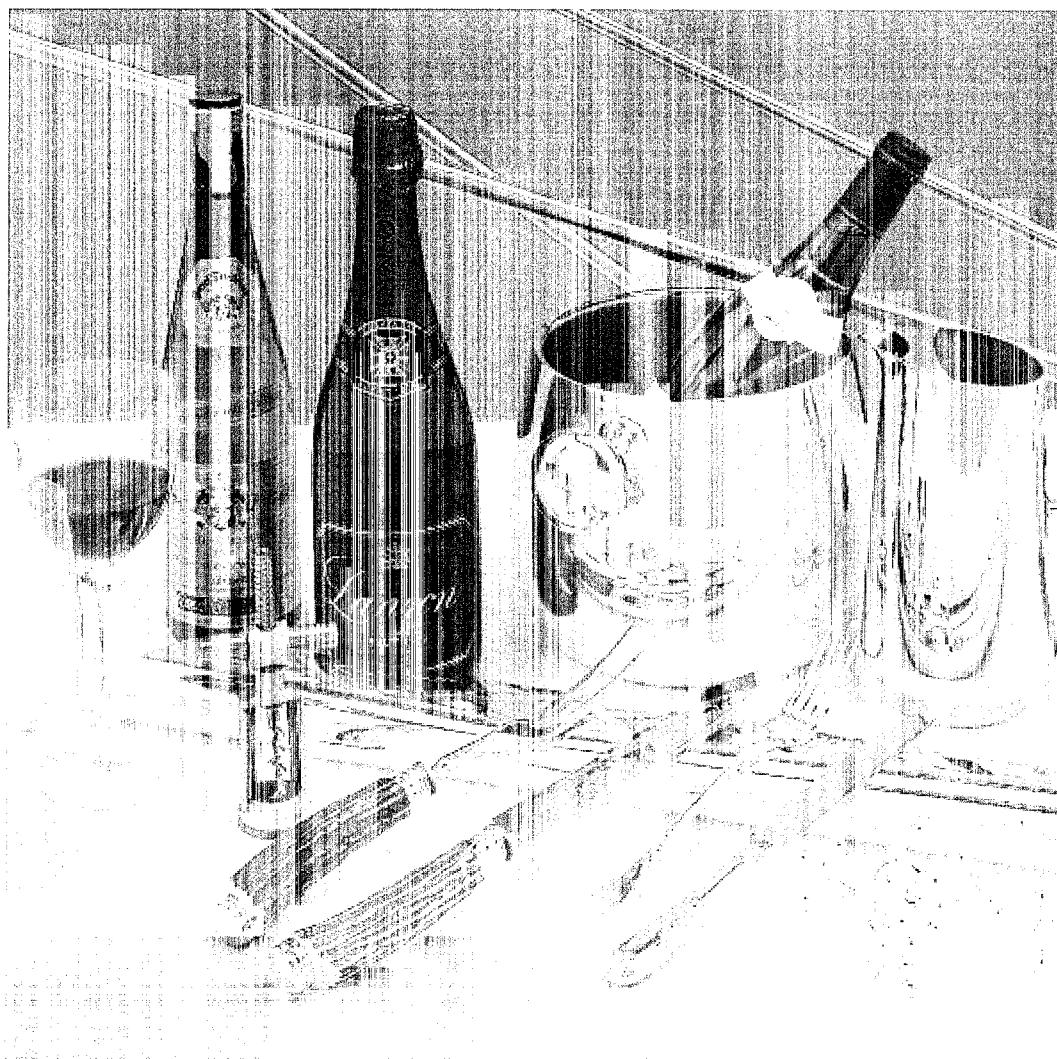


図 4.11 サンプル対象画像 “せいぶつ”

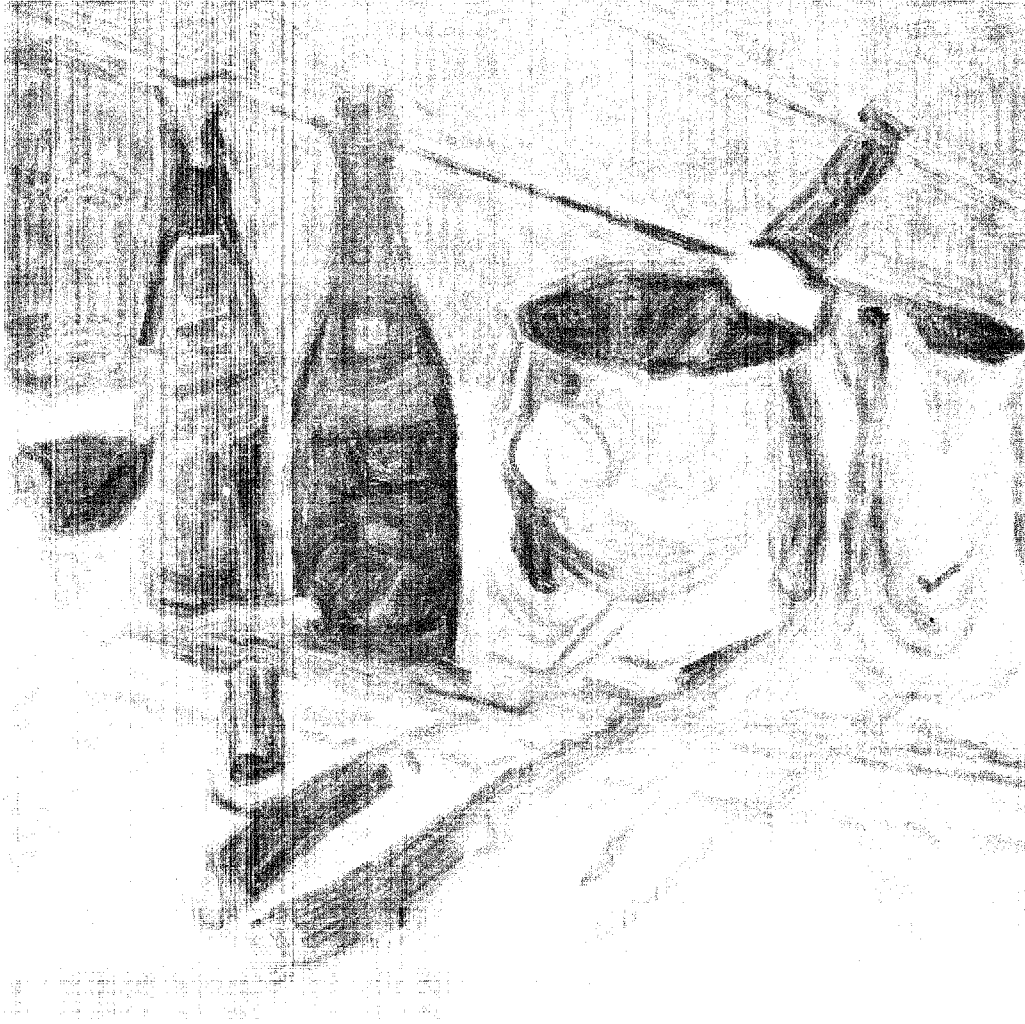


図 4.12 “せいぶつ” への単純減法混色を付加した描画結果

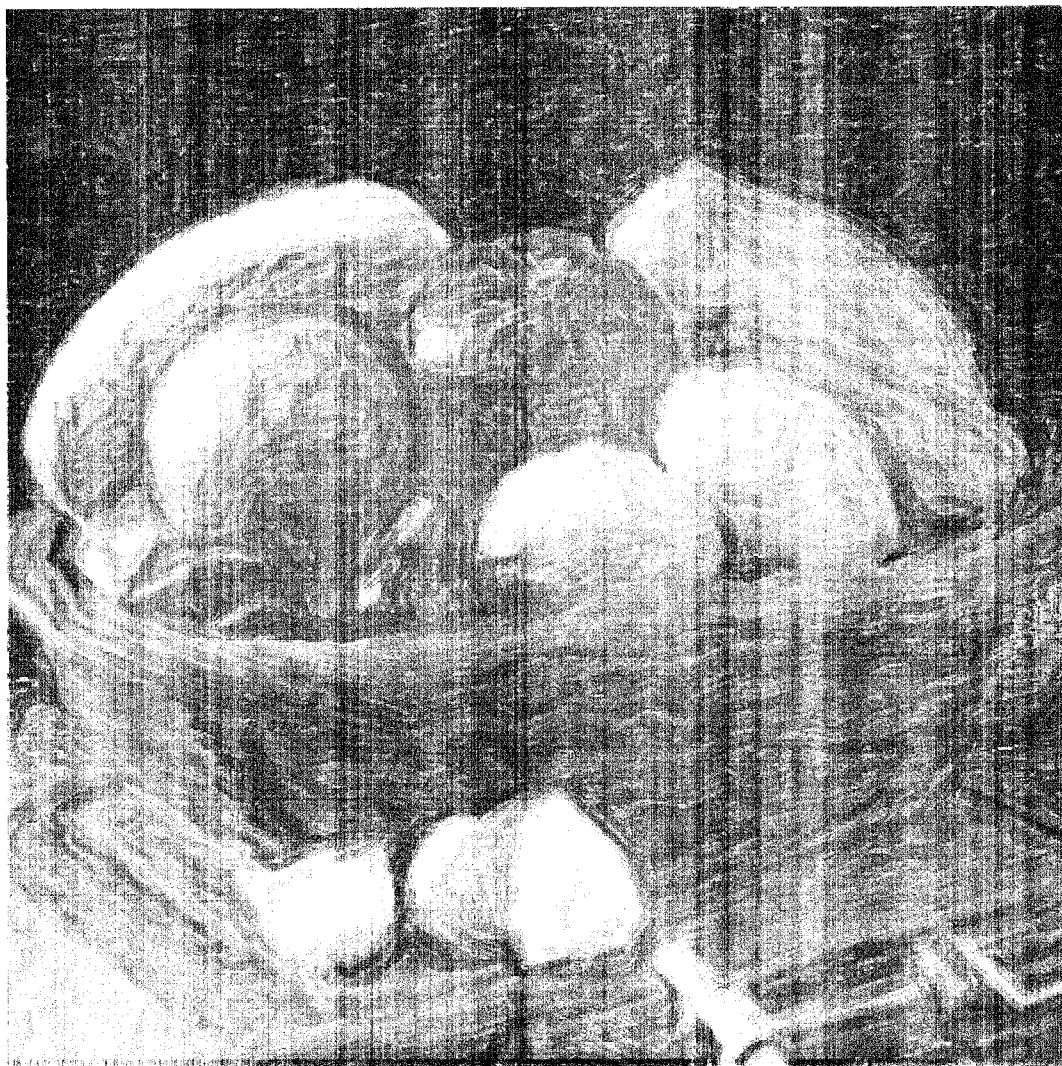


図 4.13 “くだもの”への単純減法混色を付加した描画結果

### 4.7.3 簡易混色計算を行う描画手法の混色結果

図 4.7、図 4.15、図 4.17 に対して処理を行った結果が、図 4.14、図 4.16、図 4.18 である。

図 4.8、図 4.13 と比較して、図 4.14 ではストローク描画を基本としているが描画結果ではストロークが混ざり合い、塗りつぶされたような効果が出ており、異なった雰囲気画像に仕上がっていることが分る。

特に図 4.14 のりんごの表面や、図 4.16、の薄い雲、図 4.18 の石畳の領域からは塗りつぶしによる描き込みが良好に行われていることが分る。また輪郭部分では、遮へい率を低くしたことにより、塗り重ねたような混色が再現できていることが分る。しかし、一部の輪郭領域では過度の色のくすみが見られる。これは、簡易混色計算における発色の限界であり、より良い絵の具モデルが必要といえる。この点に関しては第 5 章においてさらに詳しく述べる。



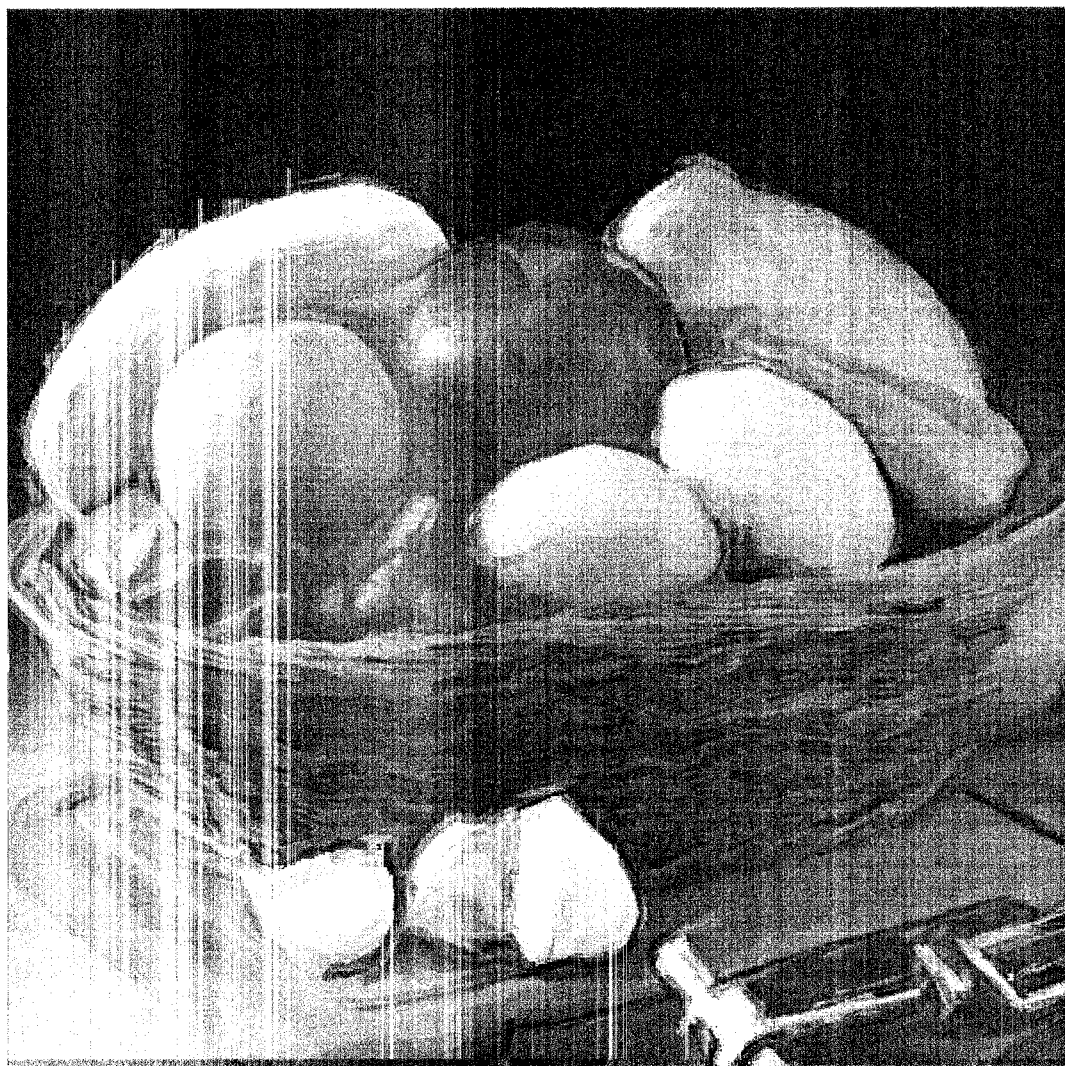


図 4.14 “くだもの”への簡易混色計算を付加した描画結果

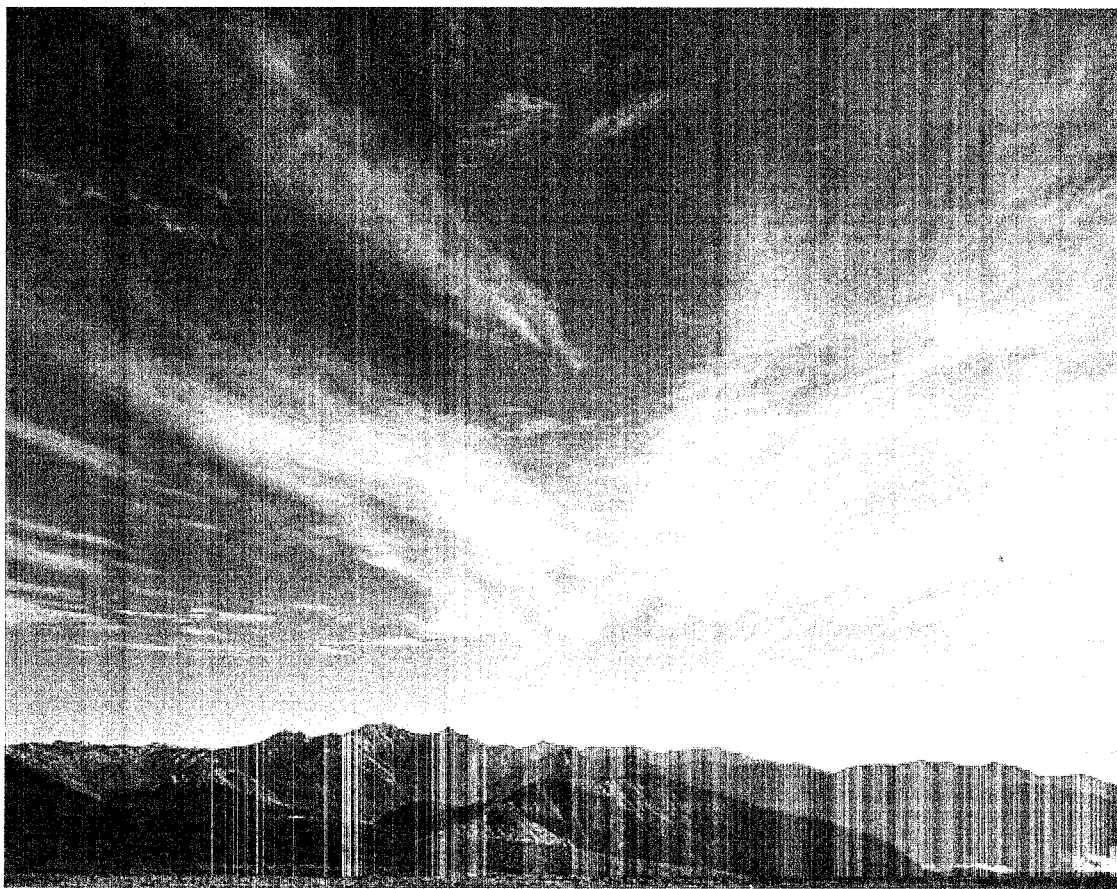


図 4.15 サンプル対象画像 “やまとくも”

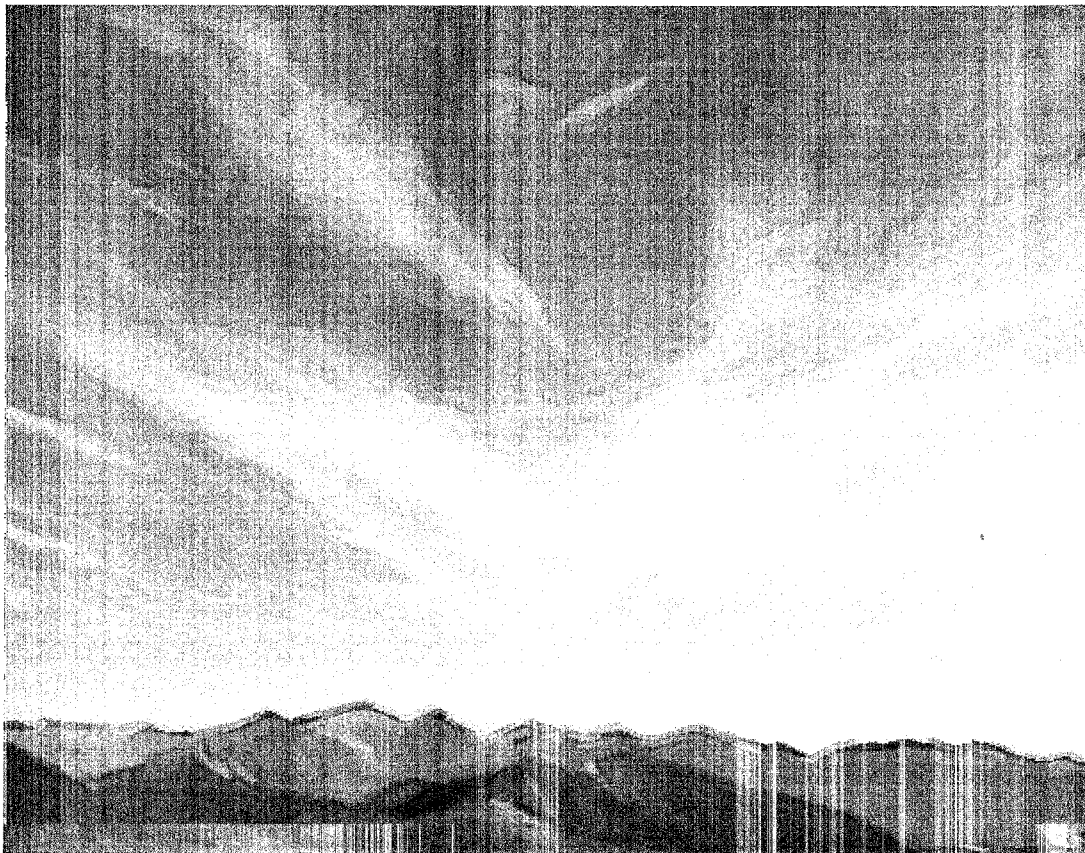


図 4.16 “やまとくも” への簡易混色計算を付加した描画結果



図 4.17 サンプル対象画像 “ひろば”

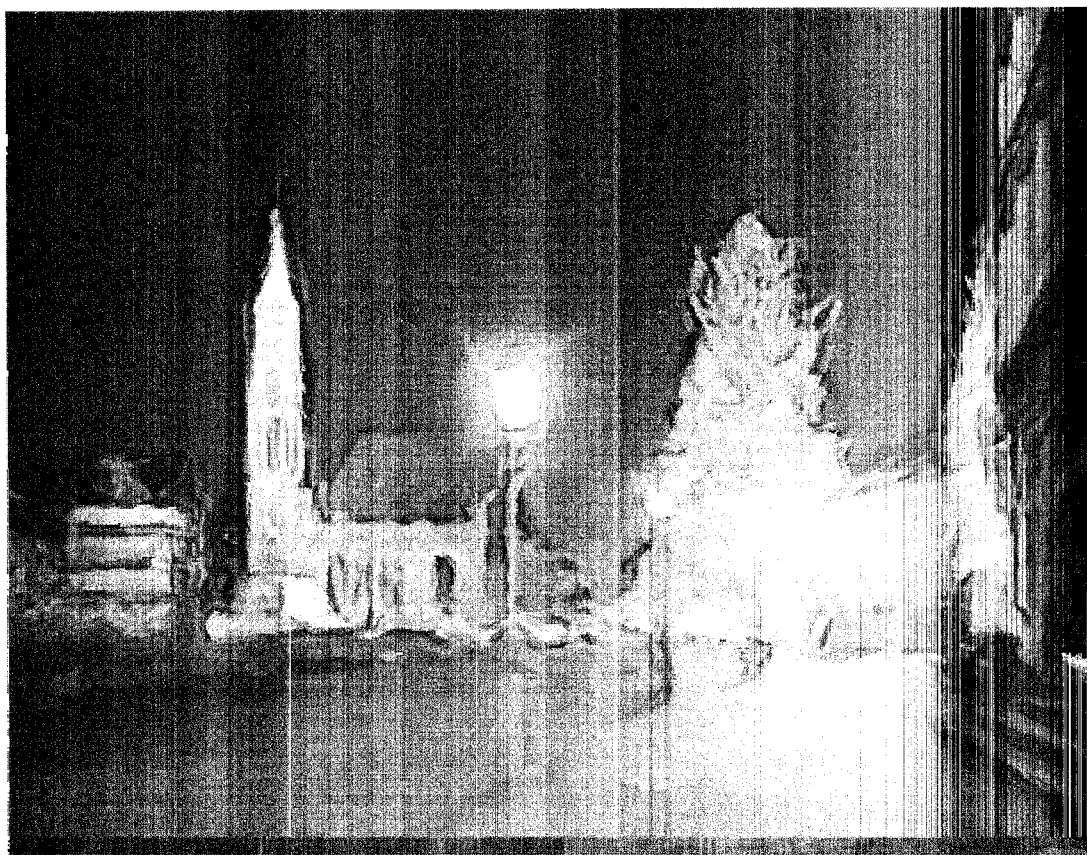


図 4.18 “ひろば” への簡易混色計算を付加した描画結果

#### 4.7.4 光沢処理を加えた手法の描画結果

式 (4.7) における定数を  $\alpha = 1.64$ 、 $\beta = 40$  として、描画平面を  $xy$  平面とした時の光源方向ベクトルを  $(-1, 1, 2)$  とした。ストロークに用いた形状テクスチャ及び厚みテクスチャは図 4.4(a) 及び (b) である。

図 4.7 を入力画像とした描画結果例を図 4.19 に示す。やや強調されすぎている面も見られるが、表面の凹凸感があり、絵のストロークの質感が光沢処理により出ていることがわかる。

現手法では、入射光を平行光線として考えて光沢の反射率を計算したが、より自然な光源下での雰囲気を出すため、光線をスポット光からのような放射状のものへと変更することも考えられる。そのような変更にも実際に厚みデータを持ち光沢反射を計算する本手法は対応が容易である。

本手法の発表後、同様のアルゴリズムが MetaCreations 社の Painter の version 5[38] から新たな機能として 1997 年から加わっている。

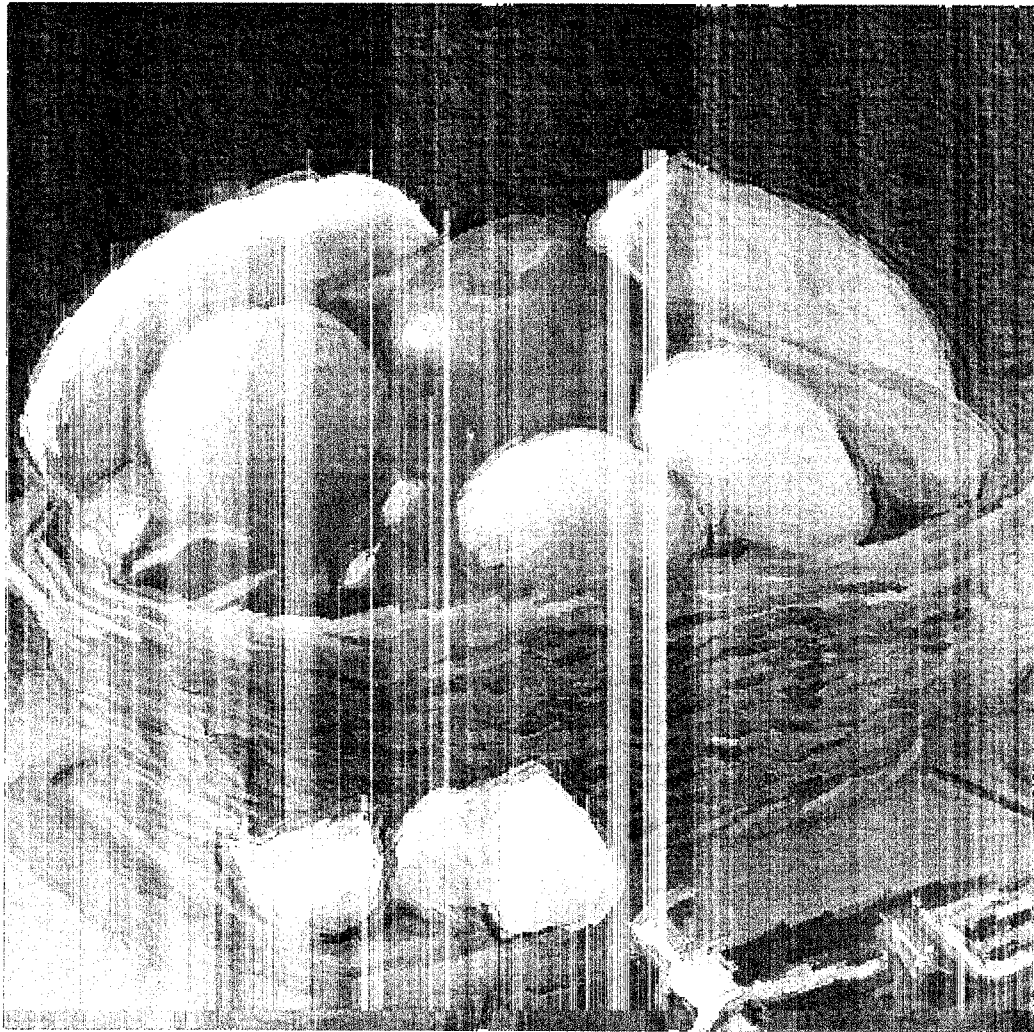


図 4.19 光沢処理を施したストローク描画の結果

#### 4.7.5 ぼかし処理による描画結果

画像サイズ 613 × 445 ピクセルの図 4.20 を対象画像とした時の描画結果を図 4.21 に示す。  
比較対象用として 61 × 61 のサイズの単純なぼかしフィルタを用いてぼかした結果を図 4.22 に示す。

図 4.7 を対象画像とした描画結果は図 4.23 である。

本手法では単純なぼかしアルゴリズムと異なり輪郭近傍ではぼかさないという状況に応じたぼかし、指向性を持たせたぼかしを用いることにより輪郭の保存されている画像を再描画することができていることが分る。

結果の画像から輪郭情報は保存され、細部は滑らかに塗りつぶされたエアブラシ風の画像が生成出来たと言える。



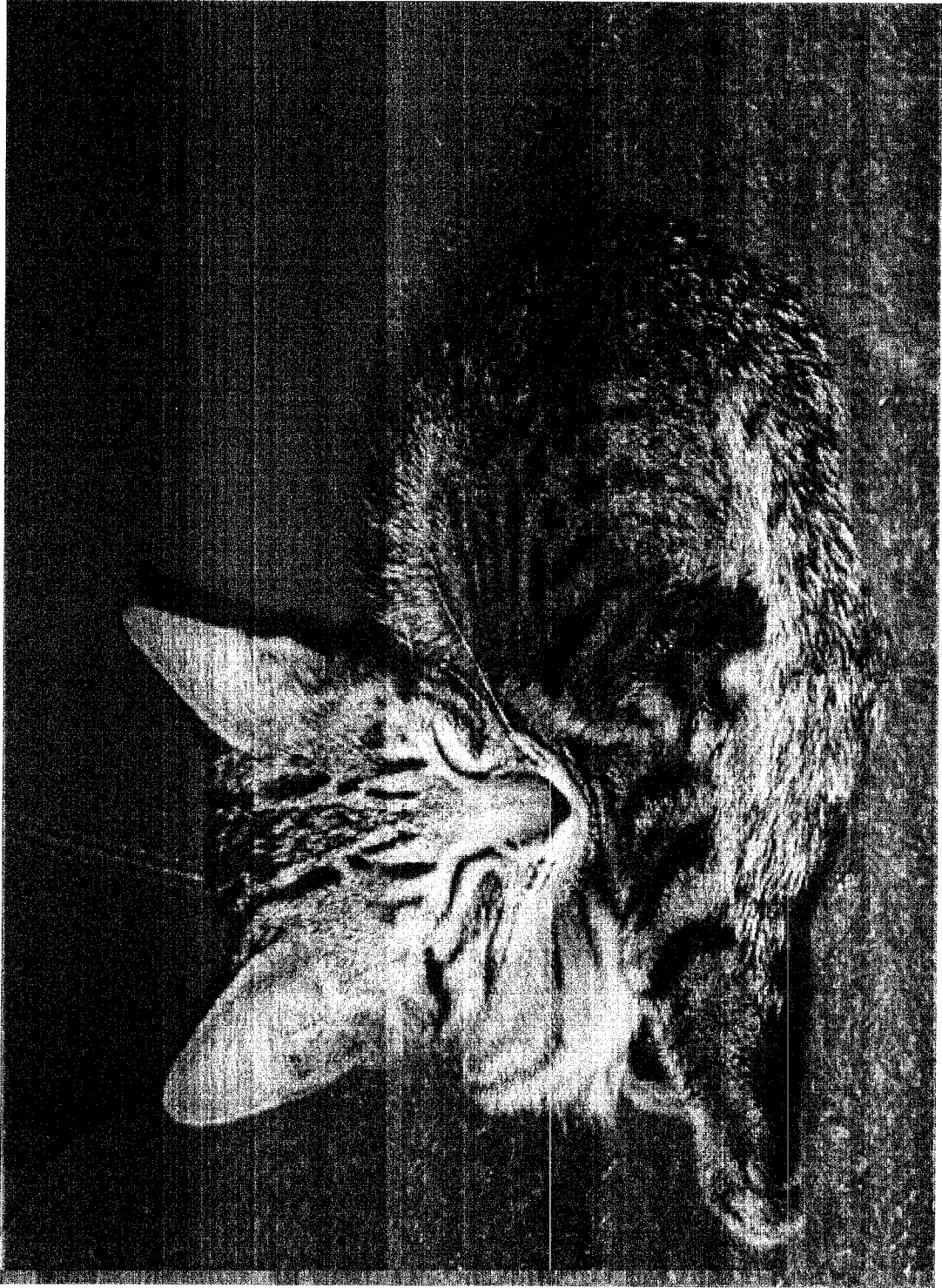


図 4.20 サンプル対象画像 ねこ

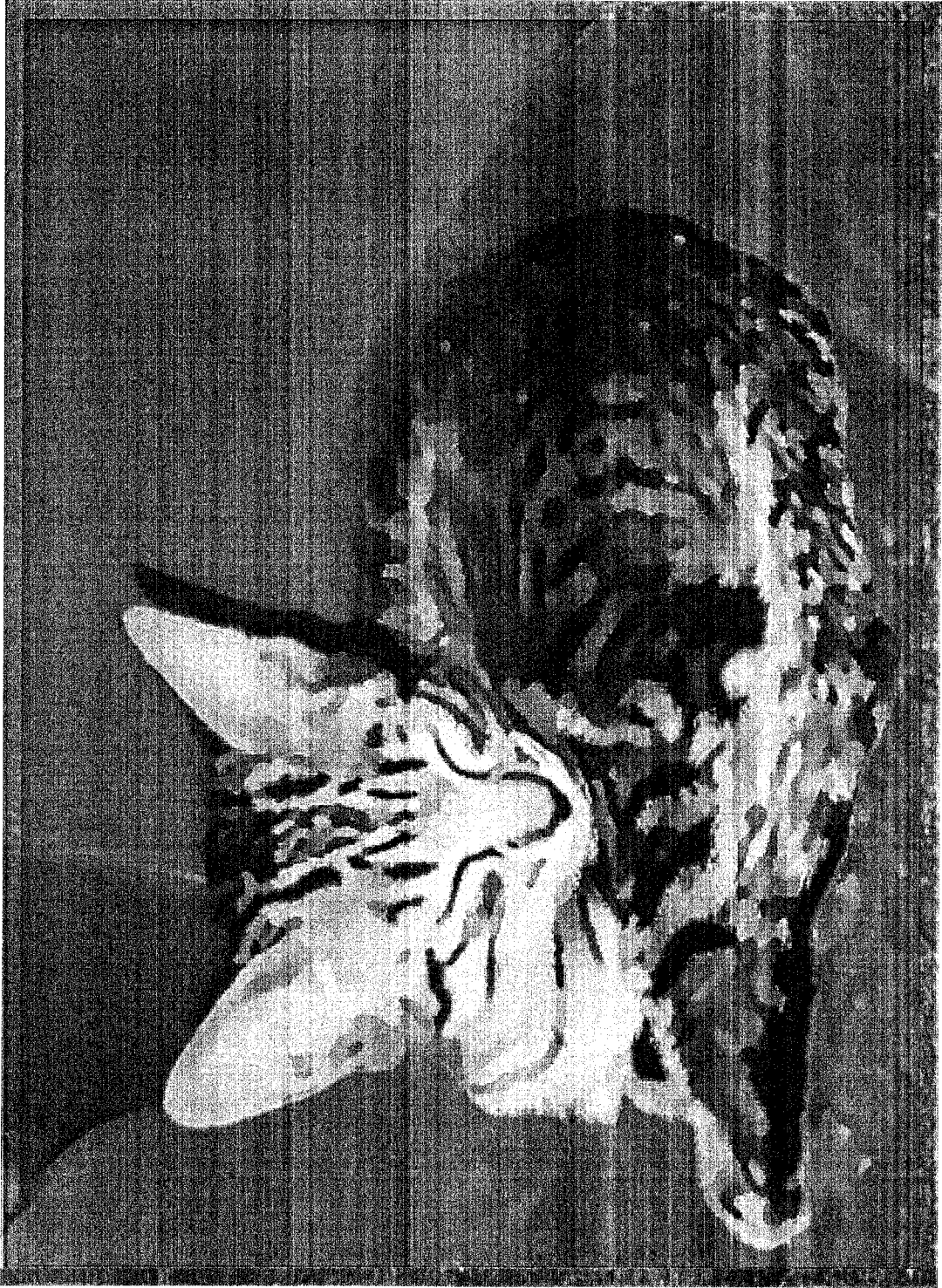


図 4.21 “ねこ” に対するぼかし描画結果

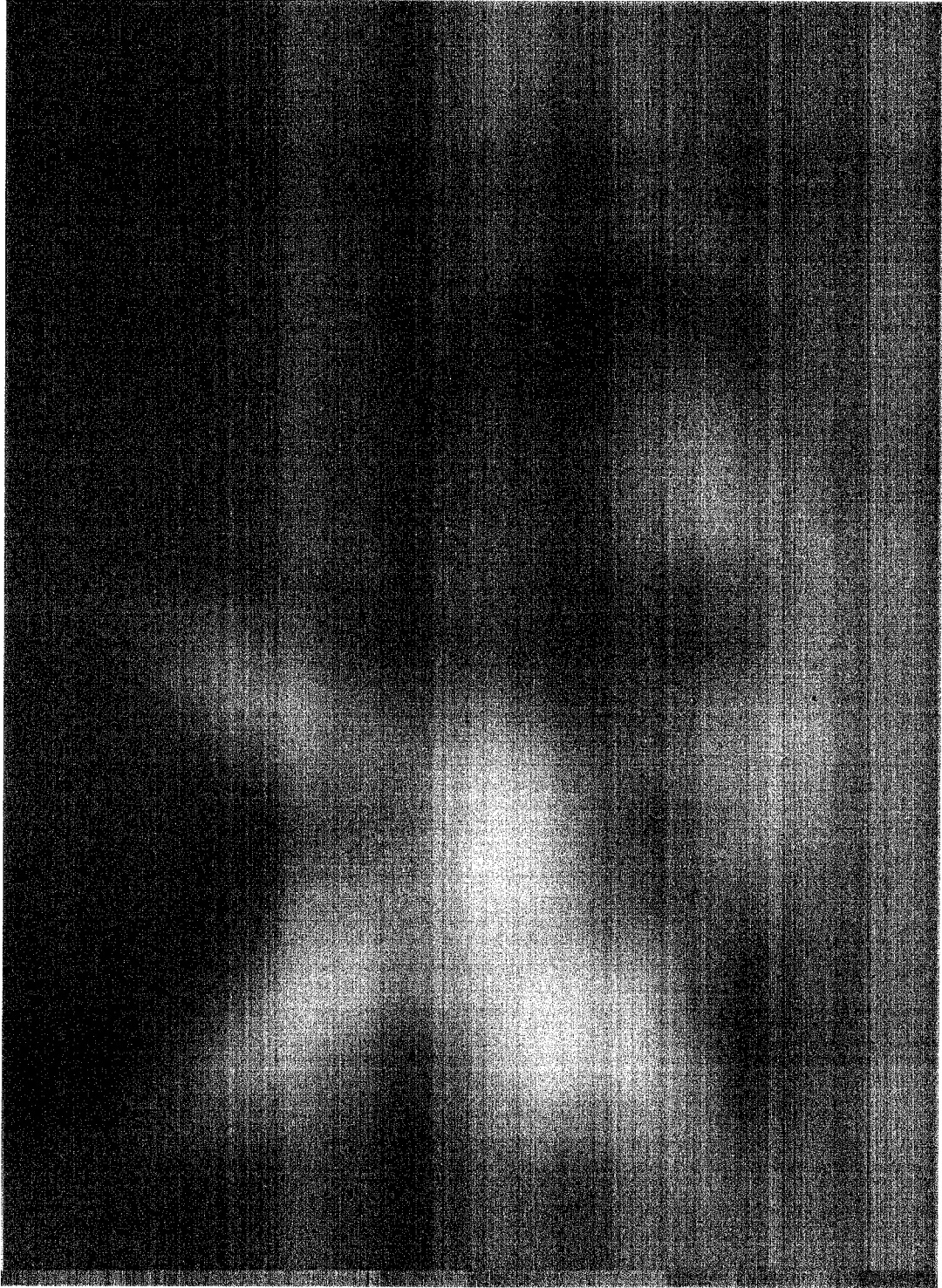


図 4.22 単純なぼかしフィルタを用いてぼかした画像

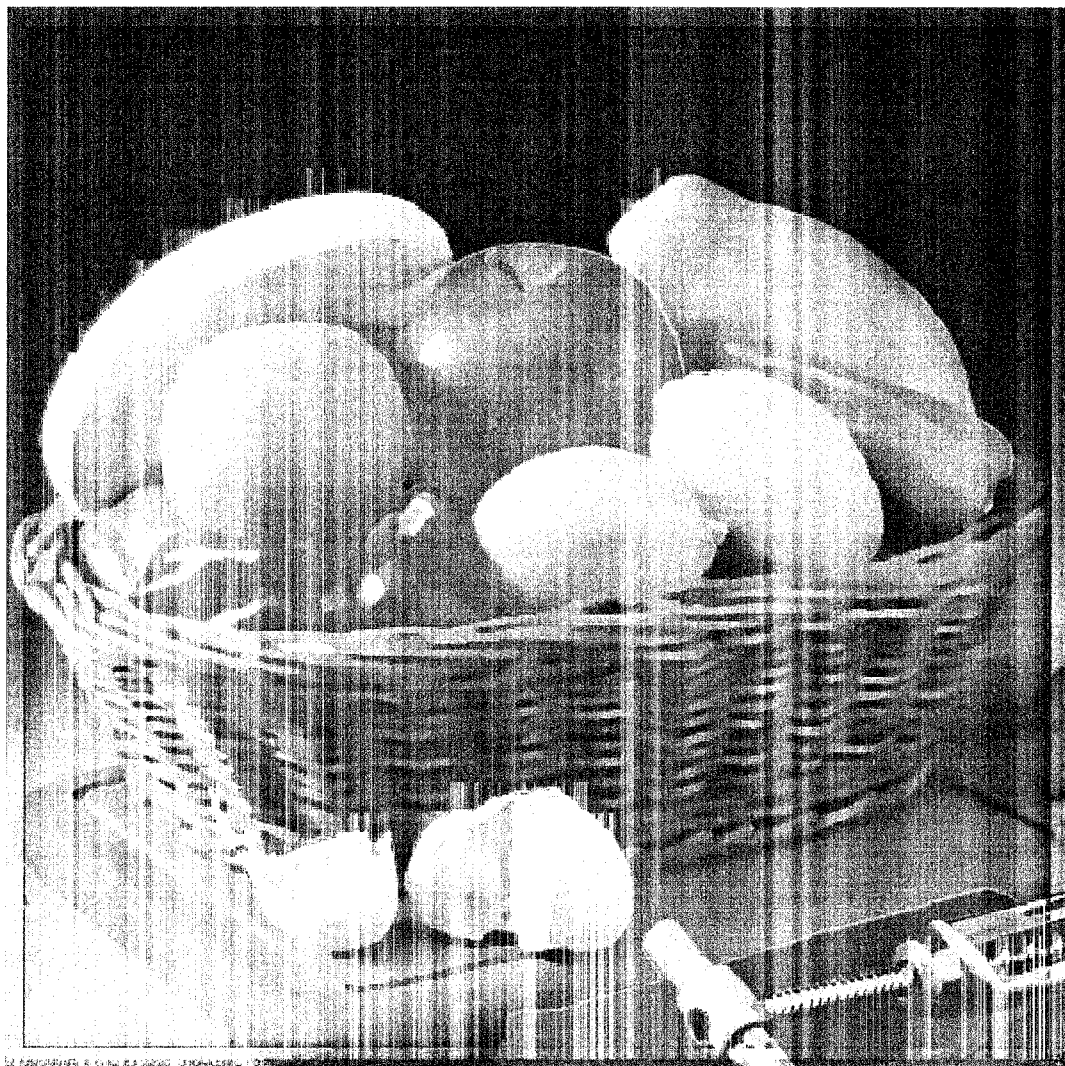


図 4.23 “くだもの”に対するぼかし描画結果

## 4.8 むすび

本章では、写真等の自然画像を入力として手描き風の画像を作成する様々な手法について述べた。

ストローク描画法では、様々なバリエーションの描画法へと発展させて、油絵風、サインペン画風等の画像を作成できた。また、ぼかし描画法ではエアブラシ風の描画を作成できた。様々な雰囲気画像を自動的に作成可能であることは必要に応じた選択肢として重要である。

作成された画像結果全体を通して、対象物の認識を行わなくても十分に手描き風画像の自動作成が可能であることがわかった。また、人間が感じているであろう画像の局所的な情報を残しつつ、手による描画の画像的特徴をもつように再構成すれば、本来の描画プロセスとは異なっても、手描き風画像が作成できることがわかった。

しかしながら、画像処理によって得られる局所情報が人間の感覚と完全に同一でないことから、描画結果に一部荒さが目立つ。より細やかな描画には画像処理手法の研究が必要である。

本章の手法は描画対象物が何であるかという内容の情報は通すが、雰囲気は変化させて通すというフィルタであるとも言える。従って、感性情報処理の分野の研究 [96] や色知覚の研究 [97] を参考として、今後は色調を変化させて雰囲気を変えるような手法の検討も考えられる。

また、画像のより全体的な印象についての研究 [98] や画像の記号的意味についての解説 [99] も参考なると考えている。

## 第 5 章

# 絵の具の発色を再現する計算機上の絵の具モデル

### 5.1 まえがき

手描き風 CG において、その画質を向上させるためには、絵の具を塗った際に生じる微妙な色変化が重要である。

しかし、滲み、濃淡や生乾き状態への塗り重ねなどの混色の再現は、従来の三次元色座標系上での色計算だけでは困難と考えられる。また、光源の混色で生じる加法混色とは異なり、絵の具の混色は、その現象が複雑であり、混色計算を色座標上での直接的な計算法では行うことができない。

メタリック塗装膜の反射の CG に関する研究 [85] や、多層膜干渉光の CG での再現に関する研究 [86] があるが、一般的な絵の具の発色再現には応用できない。

そこで本章では、始めに、単純減法混色の計算では絵の具の混色が表現できないことを示し、次に薄い層の発色をその内部における光の吸収と散乱によりモデル化した Kubelka-Munk の理論を用いた絵の具モデルの実現法について述べる。

### 5.2 単純減法混色で表現できない絵の具の発色問題

絵の具の混色は簡単に減法混色であると説明がされる場合が多い [66]。しかし、減法混色とはかなり広い混色を表しており、曖昧である。最も代表的な減法混色である複数の透過

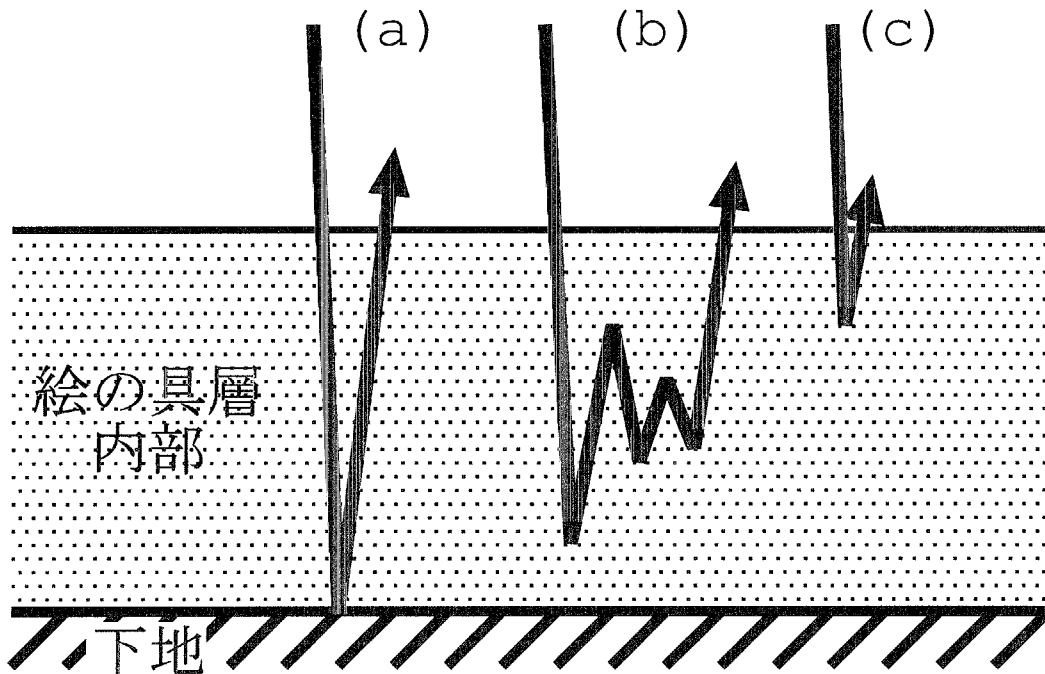


図 5.1 絵の具層内部を進む光の経路

フィルタにおける混色では、あるフィルタの透過光が次のフィルタの入射光となるので混色スペクトル  $p(\lambda)$  は式 (5.1) のように入射光  $l(\lambda)$  の各スペクトルに対する各フィルタ透過率  $f_i(\lambda)$  の積になる。この混色のことを単純減法混色と以下では呼ぶ。

$$p(\lambda) = \left( \prod_i f_i(\lambda) \right) l(\lambda) \quad (5.1)$$

しかし、積層された絵の具層の内部では、全ての光が図 5.1 の (a) で表される光路を通るわけではなく、散乱により (b) や (c) によって示すような光路を通るものもある。そのため、式 (5.1) を用いた計算では反射光の入射光に対するエネルギーの減少量が実際よりも大きくなりすぎるという欠点がある。端的な例としては絵の具層内部の光の散乱による光路の変化を考慮していない為、黒地に白を塗り重ねた場合の計算が式 (5.1) では不可能である。このようなことから、絵の具の混色を単純減法混色と明確に区別している文献もある [67][75]。

## 5.3 絵の具のシミュレーションへの Kubelka-Munk の理論の適用

Kubelka-Munk の理論では 2.4.2 項 で述べたように層内部の光の挙動を吸収率と散乱率によりモデル化しているので前節の問題を解決できる。ただし、染織業界で用いられているような Kubelka-Munk の理論の適用法 [74][77] は絵の具の発色計算では使うことが出来ない。

染織の場合での Kubelka-Munk の理論を利用する場合には、発色をする層の厚みをほぼ無限として反射率の計算に 2.4.2 項 の式 (2.25) を基本に用いているため、微妙な厚みの違いによる発色の変化が重要となる絵の具の発色計算には使うことが出来ない。

CG 用の絵の具のモデルのための Kubelka-Munk の理論の適用には Curtis らにより提案された方法がある。しかしながらその手法では混色時の絵の具の色の濁りに問題が生じてしまう。

次節より述べる手法は、混色比率を変化させることによる、絵の具の色変化を直線的にする改善手法、ある混色比率における一時的な色の濁りを減少させる手法となっている。

## 5.4 Kubelka-Munk の理論の一般式の実装に必要な数値の導出

2.4.2 項 で述べたように、Kubelka-Munk の理論では、式 (5.2) で示されるように、反射率  $R$  を層の厚み  $x$  の関数として表現することが可能である。 $R_\infty$  は式 (5.3) で求まる厚みが無限大の時の反射率である。式 (5.2)(5.3) での  $K$ 、 $S$  はそれぞれ、吸収率、散乱率を表す。この二つの値には、高濃度な場合を除いて層の物質の濃度に対して比例関係にあるという特徴がある。この特徴により、絵の具を混合した際の新たな  $K$ 、 $S$  の計算は式 (5.4)(5.5) で示されるように平均の計算で行うことが出来る。式中で、 $K_i$ 、 $S_i$ 、 $m_i$  はそれぞれ絵の具  $i$  の吸収率、散乱率、体積をあらわす。

$$R(x) = \frac{\frac{1}{R_\infty}(R_u - R_\infty) - R_\infty(R_u - \frac{1}{R_\infty})e^{Sx(\frac{1}{R_\infty} - R_\infty)}}{(R_u - R_\infty) - (R_u - \frac{1}{R_\infty})e^{Sx(\frac{1}{R_\infty} - R_\infty)}} \quad (5.2)$$

$$R_\infty = 1 + \frac{K}{S} - \sqrt{\frac{K^2}{S^2} + 2\frac{K}{S}} \quad (5.3)$$

$$K = \frac{\sum_i m_i K_i}{\sum_i m_i} \quad (5.4)$$



表 5.1 隠ぺいに必要な絵の具の厚さの平均

名称	比重	隠ぺい力 ( $cm^2/g$ )		厚さ $d(cm)$	色
		対白板	対黒板		
亜鉛華	5.55	-	40	0.0045	白
リトポン	4.2	-	50	0.0048	白
べんがら	4.75	3495	3220	0.000063	赤
クロムバーミリオン	6.05	1070	655	0.0002	赤
カドミウムレッド	4.75	540	460	0.00042	赤
黄鉛	6.3	240	130	0.00086	黄
酸化クロム	5.05	1240	1240	0.00016	緑
紺じょう (プルシアンブルー)	1.85	1700	-	0.00032	青
群じょう	2.3	50	80	0.0067	青
隠ぺいに要す平均厚				0.002	

$$S = \frac{\sum_i m_i S_i}{\sum_i m_i} \quad (5.5)$$

以上のことから、絵の具の濃度変化、混合混色での比率の変化、塗り重ね混色における厚みの変化による発色計算が Kubelka-Munk の理論を用いれば容易に求まることがわかる。

しかし、式 (5.2) で与えられる一般解を求めるためには各スペクトルに対する散乱係数  $S$ 、無限厚反射率  $R_\infty$ 、下層の反射率  $R_u$  の 3 定数の値が必要となる。ここで、十分に厚いとみなせる時の反射率  $R_\infty$  と下地の反射率  $R_u$  は実際の計測が可能であるが、 $S$  の値を直接計測することは困難である。

そこで  $S$  を求めるための準備として、まず、各顔料の比重と隠ぺい力から隠ぺいに要する厚み  $d$  を計算する。文献 [63] から引用した  $d$  を得るための諸数値を、表 5.1 にまとめて示す。

実際の絵の具では隠ぺいに必要な厚みは様々であり、それは個々の絵の具の一つの特徴である。しかし使用上は同じ厚さで隠ぺい出来ることが扱いやすい場合もある。そのような場合には、全ての絵の具の  $d$  を隠ぺいに要する平均厚みを用いて決定する。

次に式 (5.2) を変形し、下地を白として  $R_u = 1$  を代入して式 (5.6) を得る。

$$\frac{(R - \frac{1}{R_\infty})(1 - R_\infty)}{(1 - \frac{1}{R_\infty})(R - R_\infty)} = e^{Sx(\frac{1}{R_\infty} - R_\infty)},$$

$$S = \log \frac{(R - \frac{1}{R_\infty})(1 - R_\infty)}{(1 - \frac{1}{R_\infty})(R - R_\infty)} \cdot \frac{1}{x(\frac{1}{R_\infty} - R_\infty)} \quad (5.6)$$

ここで、 $x = d$ 、 $R = (R_\infty + (1.0 - R_\infty) \times 10^{-4}) \cong R_\infty$ を代入することにより、 $S$ を求める。さらに式(5.3)を変形した式(5.7)により、 $R_\infty$ と $S$ から $K$ を得ることが出来る。 $K$ は式(5.2)では必要ないが混合による混色計算で必要になる。

$$K = \left( \frac{1 + R_\infty^2}{2R_\infty} - 1 \right) S \quad (5.7)$$

以上より、式(5.6)(5.7)と図5.2に示すような分光測定器で計測可能な実際の絵の具や顔料のスペクトル反射率の特性データを用いることにより、その散乱係数 $S$ と吸収係数 $K$ を導くことが出来る。

このようにして、重ね合わせによる混色と混合による混色の計算を行うためのデータを用意することが出来る。

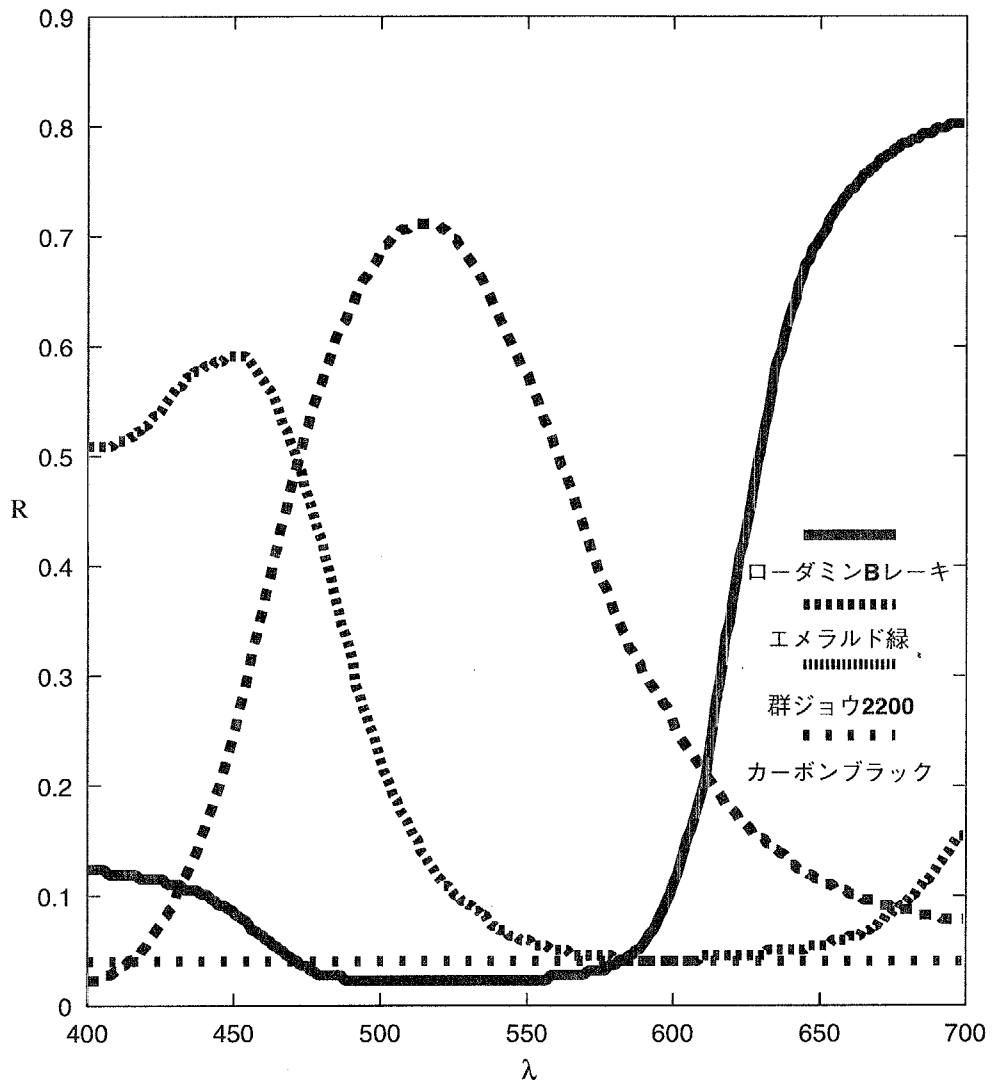


図 5.2 顔料のスペクトルデータの例

## 5.5 画像単位領域あたりに必要となるデータと操作

一般的な計算機における色データの格納形式は  $rgb$  値であり、単位画素あたりの容量も特殊な用途を除き 24bit あれば十分である。しかしながら、本章で考えている絵の具モデルでは各スペクトルの  $K$ 、 $S$  をデータとして格納しなければならない。また、層の上下関係も発色に関係するため、各層の絵の具の情報も格納しなければならない。そこで、本節では本手法における単位画素あたりのデータを簡素化して格納するデータ構造と、そのデータを操作する方法について述べる。

### 5.5.1 データ構造

計算機上の描画を行う各微小領域には、描き込まれた絵の具の各スペクトルの散乱係数  $S$  と吸収係数  $K$  を格納しなければならない。複数回絵の具を塗り重ねる毎に  $K$ 、 $S$  を幾層分も格納した場合には、記憶容量が多量に必要となる。

そこで、各微小領域に格納する情報を表 5.2 のように二層に簡略化する。最上層以外の下の層の光学特性は反射率  $R_u$  のみで表す。これは乾いた下の層はその反射特性のみが必要であると考えられるためである。それに対して最上層は乾き切る前の塗り込みで混合混色が出るように  $K$ 、 $S$  で表し、その厚みの情報  $q$  も格納しておく。また、最上層の絵の具の描画時刻  $t$  も格納し、乾燥量の計算に用いる。

これより、微小領域毎に必要なデータは表 5.2 に示す様になる。

表 5.2 微小領域あたりに必要となるデータ

	波長			
	$\lambda_0$	$\lambda_1$	...	$\lambda_{N-1}$
最上層	$K_0, S_0$	$K_1, S_1$	...	$K_{N-1}, S_{N-1}$
最上層以下の層全体	$R_{u0}$	$R_{u1}$		$R_{uN-1}$
最上層の量	q			
最上層の書き込み時刻	t			

5.5.2 データ操作

次に微小領域のデータに対する操作について述べる。最低限必要な操作は以下に示す様に絵の具の描画処理、乾燥処理、及び発色処理である。

[1] 描画処理

描画処理には、前回の描画の乾燥処理も含む。実際の処理は次のステップで各波長毎に行う。ここで、新たに塗る絵の具の厚み、散乱係数、吸収係数をそれぞれ、 $q_p$ 、 $S_p$ 、 $K_p$ 、前回描画後の最上層の各値を  $q$ 、 $S$ 、 $K$ 、下層の反射率を  $R_u$  とする。

- (1) 式 (5.3) で最上層の  $K$ 、 $S$  から  $R_\infty$  を求める。
- (2) 最上層の乾燥した厚み  $x$  を  $f(T-t)$  により求める。ここで、 $T$  は現在の時刻、 $t$  は最上層を塗った時刻、 $f$  は時間と乾燥量に関する関数とする。
- (3) 式 (5.2) により  $R_\infty$  と  $x$  と  $R_u$  から反射率を求める。
- (4) 求めた反射率で  $R_u$  を更新する。
- (5) 式 (5.4)(5.5) により  $K$ 、 $S$  と  $K_p$ 、 $S_p$  の  $q-x : q_p$  の内分比で新たな  $K$ 、 $S$  を求め、更新する。

最後に  $q = q - x + q_p$ 、 $t = T$  とする。描画処理の概念図を図 5.3 に示す。

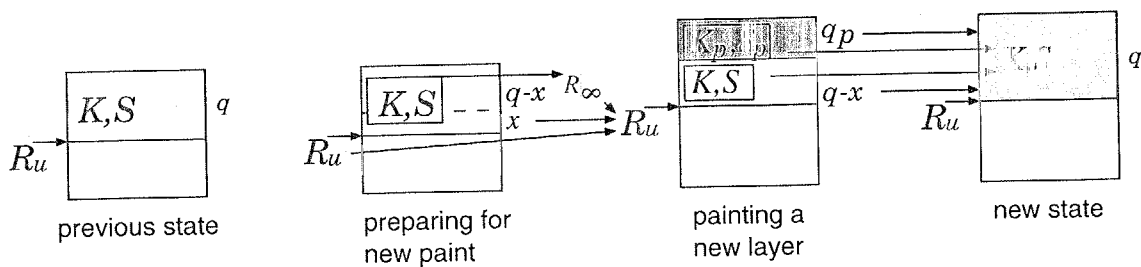


図 5.3 描画処理の概念図

[2] 乾燥処理

現在時刻を進めるだけであり、各微小領域に対しては何も行わない。この手法は、処理が軽いので、ペイントツールなどの対話的なアプリケーションに対して効果的である。

### [3] 発色処理

発色処理は各スペクトル反射率を式 (5.2) で計算し、それらから JIS で定められた重価係数 [78] を用いて 2 度視野 CIE XYZ 表色系に変換する。ここで、入射光源は  $D_{65}$  を仮定している。さらにそこから、ディスプレイ用の RGB 表色系へ変換する。

## 5.6 本提案手法の実装に必要な標本数

本節では、実装において計算量やメモリ容量に影響を与える可視帯域の標本数に対する検討を行う。

前節で提案した実装法では描画微小領域あたり可視帯域の標本点ごとに  $K$ 、 $S$ 、 $R_u$  の数値データを格納するメモリ領域が必要である。また、標本数を多くすると混色や発色の計算にそれだけコストがかかる。従って標本数を色計算における誤差の許容範囲内で少なくすることが望ましい。

従来、測色の分野では 3 刺激値の測定にサンプリング間隔の及ぼす影響を調べた研究 [72][73] はあるが、絵の具の混色後の色の誤差とサンプリング間隔についての検討はされていない。CG の研究分野では、透過光の描画とサンプリング間隔の関係について述べられたもの [71] があるが、やはり絵の具の混色との検討はされていない。そこで、波長 380nm から 780nm までの反射率を標本数 256 で測定して  $K$ 、 $S$  を表した場合の混色計算の結果を基準として、標本数を減らした場合の混色計算の結果との色差を調べる。

実験には 13 種類の顔料の色を用いた。一部を図 5.2 に示すような顔料の分光反射率は文献 [63] から得たものである。表 5.3 に示したとおり 13 種の顔料は色座標が分散するように選択している。

混色には、混合による混色と重ね塗りの混色の 2 つがあるためそれら 2 種類の実験を行った。

重ね塗りによる混色での色差の実験は次のとおりである。まず、13 色の顔料を標本数  $n$  の  $K$ 、 $S$  であらわし、次に 13 色の顔料をそれぞれ 0 から 0.00147cm の厚さまで 0.00003cm ずつ変化させて白地に下塗りし、その上に下塗りと同様に 13 色を総当りで塗り重ねて混色する。その混色結果と標本数 256 で行った混色結果との最大色差を CIELAB  $L^*a^*b^*$  で計算する。その計算結果を図 5.4 の実線で示す。

混合による混色での色差の実験は次のとおりである。重ね塗りと同様、13 色の顔料を標本数  $n$  の  $K$ 、 $S$  で表し、次に 13 色の顔料を様々な比率で混ぜ合わせて塗った色と標本数 256 の場合の色との最大色差を  $L^*a^*b^*$  で計算する。その計算結果が図 5.4 の破線である。塗り重ね混色とほぼ同じ結果となった。

2 つの結果で、色差が標本数の増加に対して単調に減少していないのは、絵の具の混色計

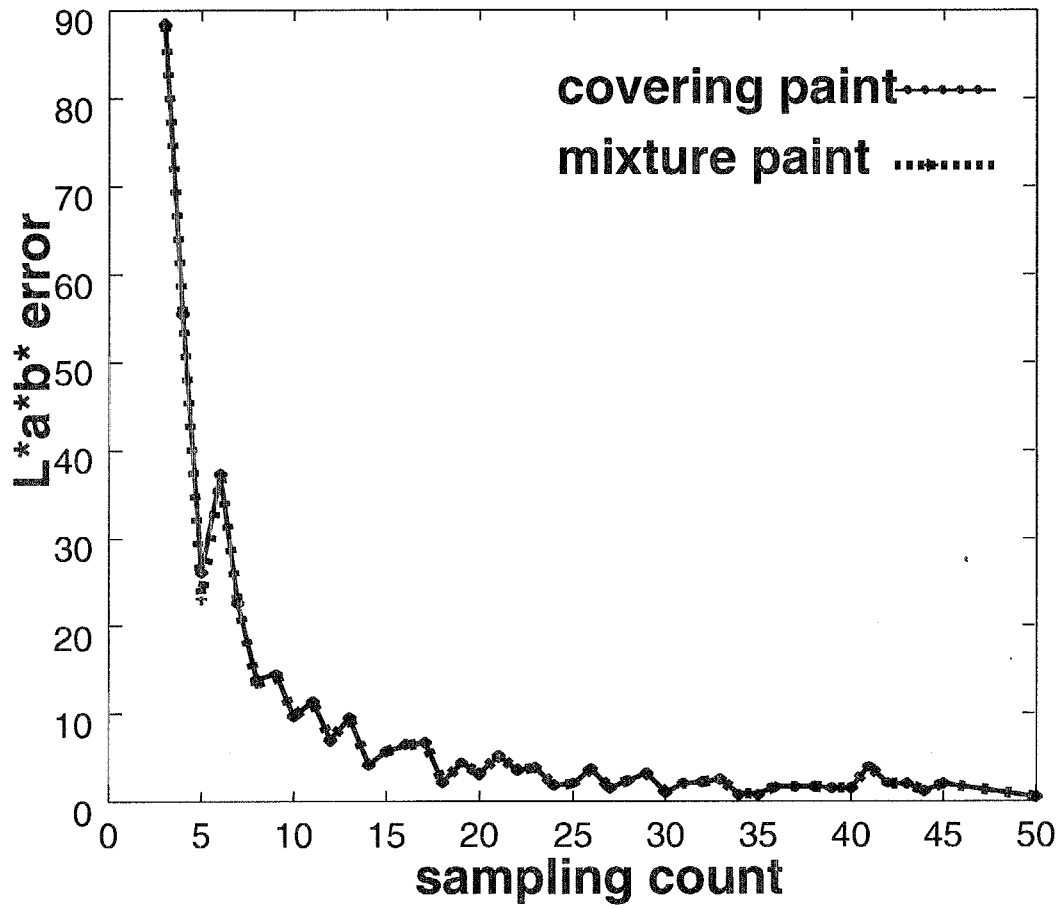


図 5.4 標本数と色の誤差の関係

表 5.3 実験に用いる絵の具の顔料

顔料名	$Y_c$	$x$	$y$
クロムバーミリオン赤口	20.61	0.558	0.348
カドミウム赤	11.68	0.546	0.314
ローダミンBレーキ	9.12	0.508	0.259
群ジョウ2200	8.54	0.168	0.103
コバルトブルー	19.83	0.194	0.183
コバルトバイオレット	30.59	0.295	0.216
フタロシアニングリーン	4.19	0.236	0.320
エメラルド緑	47.54	0.259	0.411
カドミウム黄	52.92	0.502	0.456
黄鉛5G	75.33	0.455	0.484
カーボンブラック	4.61	0.308	0.313
鉛白	91.09	0.314	0.321
カオリンクレー	82.93	0.319	0.328

算が標本波長でのみ行われており、標本化誤差がXYZの重係数により不均一に積算されるためであると考えられる。

色差の厳密性の要求は用途により変わるが、標本数に比例する計算負荷と色差の両面から、最大色差が2.24となる18個の標本数がCGペインティングの用途には適当とした。

従って、各微小領域に必要となるパラメータ数は56となり、各メモリ使用量を4 bytesとすると、総メモリ量は224bytesとなる。この値は、今日の計算機で十分実用的なメモリ使用量であるといえる。

## 5.7 本提案実装法の混色評価実験

本節では、単純減法混色計算手法とCurtisらの混色計算手法[35]に対する本提案混色計算手法の有効性の検討のための実験を行う。



## 5.7.1 単純減法混色実装法の発色特性

式 (5.1) で表した減法混色の概念を式 (5.8) により実装する。

$$\begin{aligned} r &= \left( \prod_i r_i \right) \cdot 1 \\ g &= \left( \prod_i g_i \right) \cdot 1 \\ b &= \left( \prod_i b_i \right) \cdot 1 \end{aligned} \quad (5.8)$$

ここで、 $r_i$ 、 $g_i$ 、 $b_i$  は層  $i$  の透過率であり、その透過率と厚み  $0 < x < 1$  と単位厚での透過率  $r_s g_s b_s$  との関係を表 (5.9) のようにする。またここで用いる  $rgb$  には表示系を考慮し、sRGB[79] を使う。

$$\begin{aligned} r_i &= 1 - x(1 - r_s) \\ g_i &= 1 - x(1 - g_s) \\ b_i &= 1 - x(1 - b_s) \end{aligned} \quad (5.9)$$

式 (5.8) を用いて白地に対して、前章で用いた 13 の色に対して厚みを徐々に変化させて塗り重ねた発色結果を表したものが図 5.5(a) である。この図からは CIELAB の  $a^*b^*$  での変化が比較的直線的であり、また、CIELAB の  $L^*$  で見た場合の変化も単調であり、一見扱いやすい特徴を持っていると言える。しかし次の実験結果である様々な色を下地としてある色塗り重ねた発色結果の図 5.5(b) から次の問題が提起される。ここで用いた上塗りの色はフタロシアニングリーンであるが、この手法では、厚みを増すことによって色が暗くなる一方であり、上塗りした絵の具の色には収束しないことがわかる。

これは、単純減法混色実装法の最大の欠点である絵の具層内部での散乱を考慮していないことに起因する。

次に、混合混色における発色実験を行う。混合による透過率  $r_i g_i b_i$  は式 (5.10) を用いて混合する絵の具の透過率  $r_j g_j b_j$  とその比率  $p_j$  から計算する。

$$\begin{aligned} r_i &= \sum_j p_j r_j \\ g_i &= \sum_j p_j g_j \\ b_i &= \sum_j p_j b_j \end{aligned} \quad (5.10)$$

ここでは、ローダミン B レーキの色との混合を行う。

図5.5(c)の結果を見ると、色の組合せの中には混合比率により、明度が落ち込むことがあることが分る。これは混合により色が濁ることを意味するので良い特性とは言えない。

以上から、単純減法混色実装法には、散乱を考慮していないための重大な欠点及び、混合混色による欠点があることが分る。

### 5.7.2 Curtisらの実装法の発色特性

Curtisらの実装法 [35] の評価として、まずはじめに、Curtisらが報告に載せている K と S の値をもとに白地に塗り重ね混色を行う。厚みによる発色の変化を表した結果が図5.6である。

絵の具の発色の厚みによる色相の変化が著しく、さらに彩度の最も高い厚みが各色で異なっている。このような絵の具の色と塗り方による発色の関係が容易でない点は、ユーザの習熟や自動描画アルゴリズムへの適応に困難をもたらす。

そこで、Curtisらのアルゴリズムを用いて、白地と黒地に対して塗った場合の発色が等しいものとして、まず絵の具のパラメータを独自に決定し、その上で先の実験を改めて行なった結果が図5.7(a)である。今度は厚みに対して、色の変化が先程よりも直線的で理解しやすいものとなっている。従って、Curtisらは絵の具の白色と黒色に対する絵の具の隠ぺい色を独立に決定出来ることを特長として挙げているが、実用上は同一にしたほうが良いことがわかる。

次に、先に求めた独自のパラメータによる絵の具の重塗り混色での Curtisらの発色特性の実験を行う。下地は各色、上塗りをフタロシアニングリーンとした結果が、図5.7(b)である。

赤色のローダミンBレーキやクロムバーミリオン赤口との塗り重ねで上塗りの厚みにより、一度暗くなっていることがわかる。補色でもない色同士でありながら、重ねることで鮮やかな発色が安定して得られない点は問題であると考えられる。

次に、混合混色についてローダミンBレーキの絵の具に対して他の各絵の具を混ぜた時の発色結果を図5.7(c)に示す。 $L^*$  で見た混合比率に対する変化では、変化の途中において明度の極小値を持つことがわかる。すなわち、単純減法混色の実装法と同様、絵の具をある比で混合した時に、濁ってしまっている。

以上、Curtisらの実装法では混色において鮮やかな発色を安定して得られないことがわかる。

### 5.7.3 本提案手法の発色特性

最後に本提案手法の発色特性についての評価を行う。

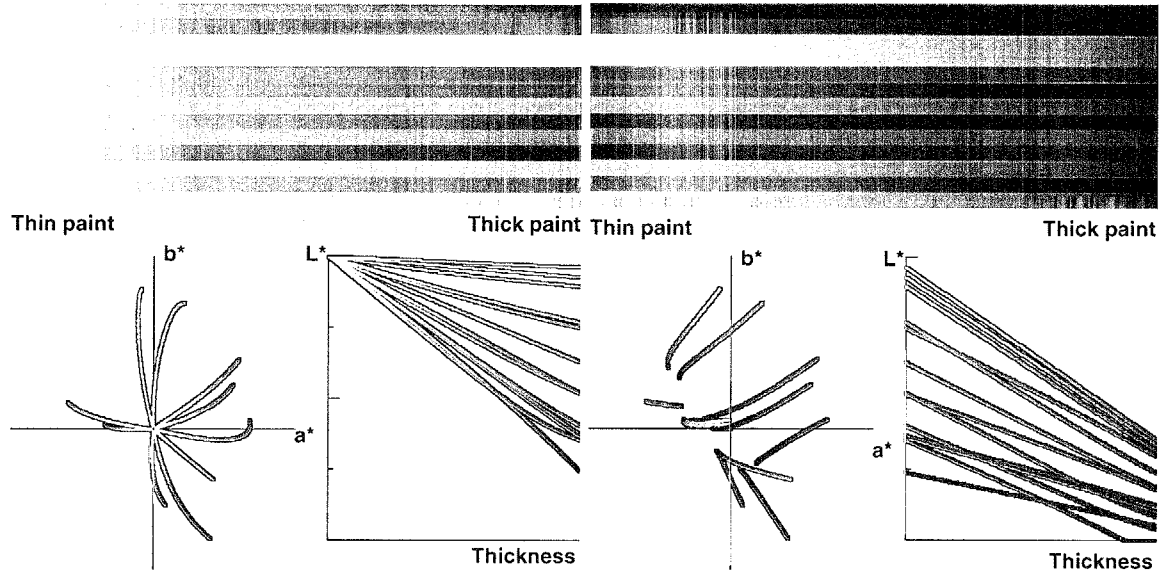
白地に対して各色の絵の具を厚みを徐々に変えて塗り重ねた場合の発色変化は図 5.8(a) のようになる。 $a^*b^*$  で見た変化は他の実装法と比較して直線的であり、厚みに対する発色が素直であるといえる。

次に Curtis らの提案法への評価実験と同様に各色を下地としてフタロシアングリーンを塗り重ねた混色での発色結果を図 5.8(b) に示す。

上塗りの厚みを増すことにより、フタロシアングリーンの色に収束していることがわかる。また  $L^*$  の変化で見た場合にも単調性を示しており、濁ることのない素直な特性であることから、厚みに対しての発色の変化が他と比較して優れていると言える。

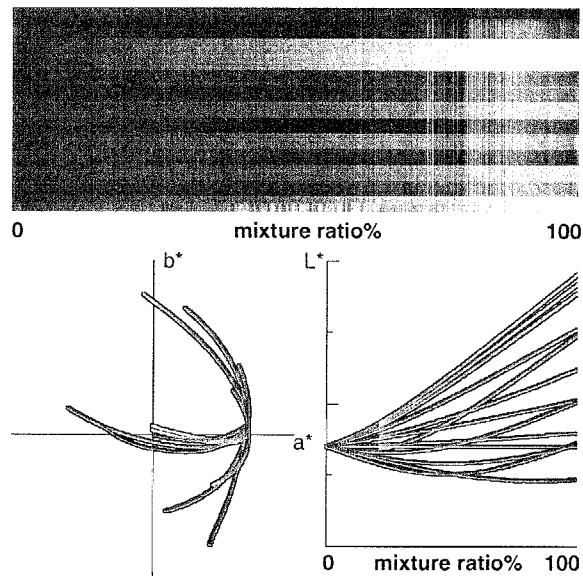
さらに、ローダミン B レーキの絵の具に対して他の各絵の具を混ぜた時の発色結果を図 5.8(c) に示す。

絵の具の混合比率に応じて絵の具が  $a^*b^*$  上でほぼ直線的な変化を示しており、期待する発色を得易いことがわかる。また、 $L^*$  に対してはほぼ単調変化を示していることから、濁りのない混色特性であることがわかる。



(a) 白地に各色を塗り重ねた発色結果

(b) 各色にフタロシアニングリーンを塗り重ね混色した発色結果



(c) ローダミンBレーキと各色を混合混色した発色結果

図 5.5 単純減法混色を実装した場合の発色結果

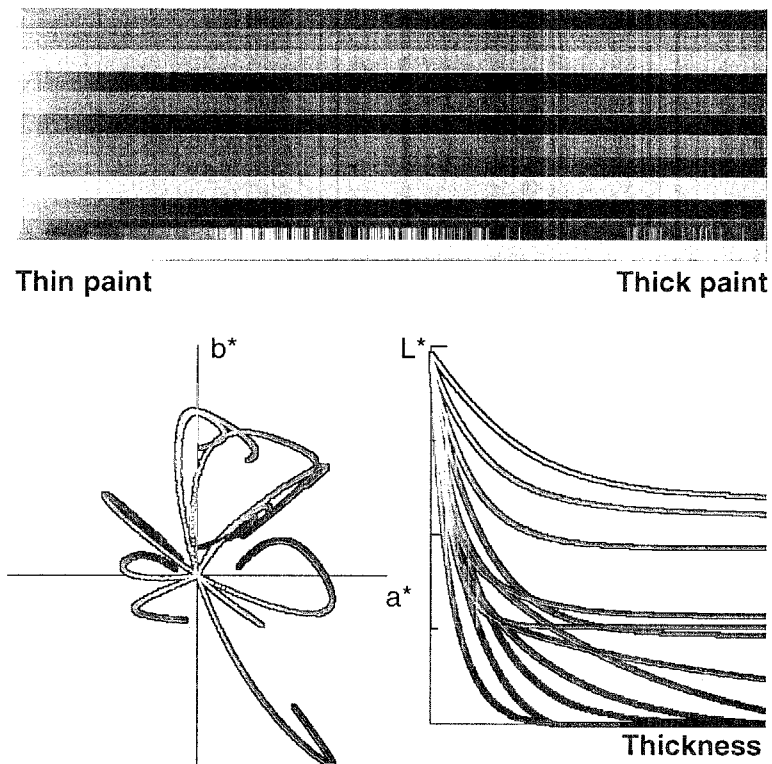
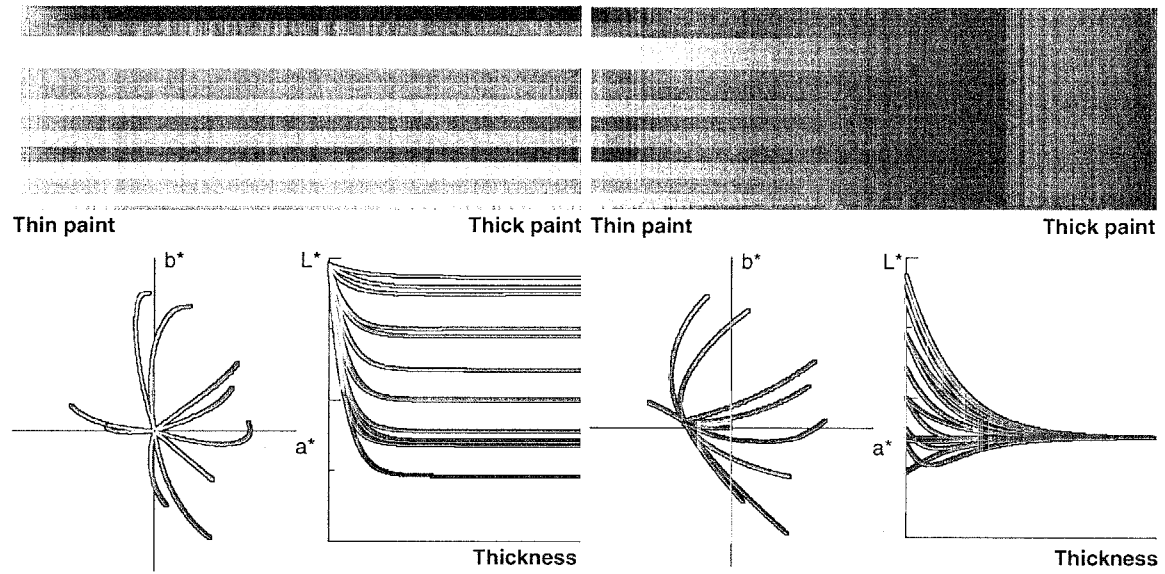
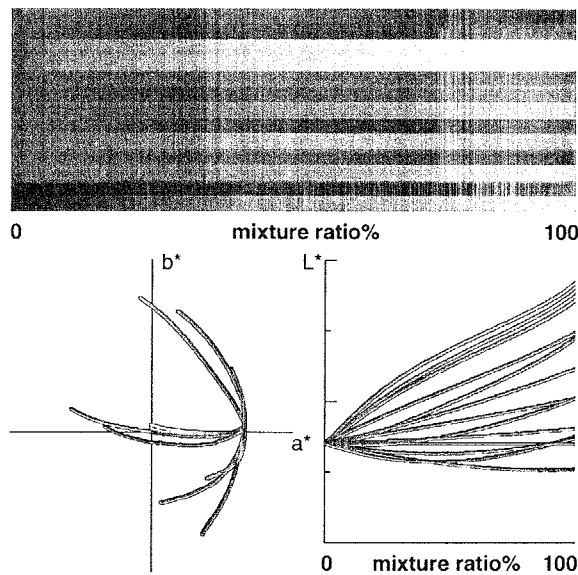


図 5.6 Curtis らによる実装法で彼らのパラメータを用いた場合の塗り厚を変化させた時の発色変化



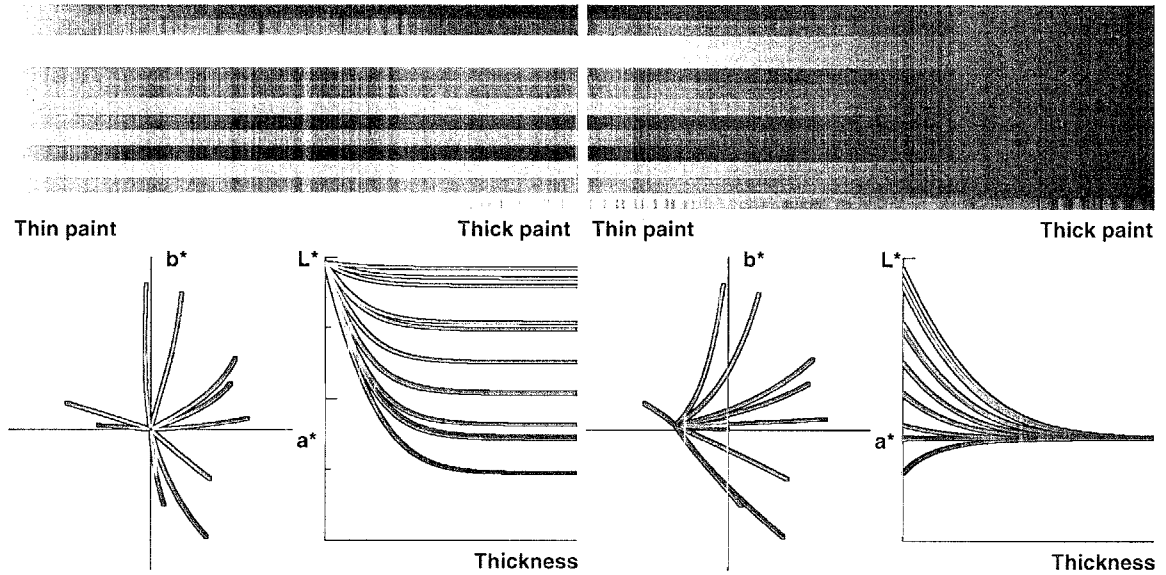
(a) 白地に各色を塗り重ねた発色結果

(b) 各色にフタロシアニングリーンを塗り重ね混色した発色結果



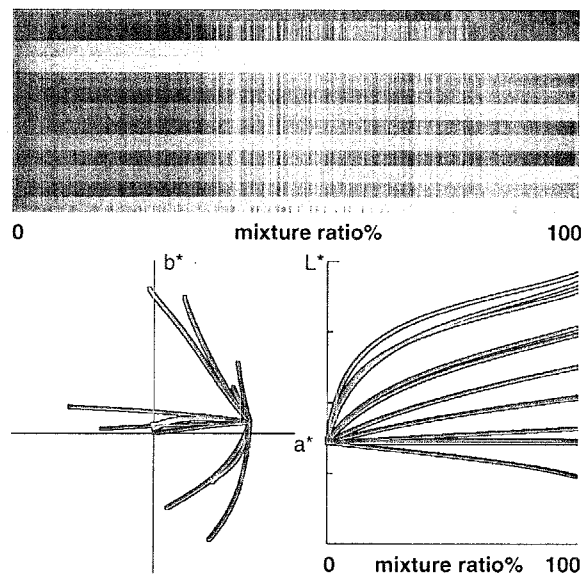
(c) ローダミンBレーキと各色を混合混色した発色結果

図 5.7 Curtis らの実装法での発色結果



(a) 白地に各色を塗り重ねた発色結果

(b) 各色にフタロシアニングリーンを塗り重ね混色した発色結果



(c) ローダミンBレーキと各色を混合混色した発色結果

図 5.8 本章での提案手法による発色結果

## 5.8 むすび

本章では、Kubelka-Munk の理論を基にして、絵の具のモデルの実装法の提案を行った。本実装法では絵の具の厚みや濃度に応じた発色計算が可能であり、塗り重ね混色、混合混色の計算も容易に行うことが出来るという特徴を持つ。

また、単純減法混色実装法と Curtis らの実装法との発色の比較を CIELAB 表色系において行い、それらと比較して濁ることなく、また直感的な混色が得られることを示した。

混色における明度変化の単調性、色相彩度で見た色変化の直線性は、フォトタッチ等での自動描画アルゴリズムにおける望む色から逆に絵の具を選定するアルゴリズムに対するより容易な実装が可能であると考えられる。また、ペイントツールでのユーザの描画における色に関する習熟を容易にすることも考えられる。

さらに本手法は、絵の具のモデルに必要なパラメータが主に分光反射率であることから、実際の絵の具の反射率を測定することにより直接パラメータを得ることが出来るという特徴を持つ。これは、従来の画材を使った描画に近い計算機上でのフォトタッチアルゴリズムへの応用や、従来の画材に慣れ親しんだ人々がより違和感なくデジタルペイントシステムを利用出来ることに繋がると考えられる。

従来の三次元の表色系での色表現と比較して、必要となるメモリ量は多くなるが、その総量も十分に実用の範囲内であり、発色特性の利点から微妙な色変化を求めるデザイン分野等の用途に対して本手法は非常に有効であると考えられる。



## 第 6 章

# 直感的な色選択と絵の具モデルの パラメータ対応

### 6.1 まえがき

第5章では、混色時の発色に優れた絵の具モデルの構築法について述べた。提案した絵の具モデルでは光のエネルギー計算を各スペクトル毎に行い、発色計算を行っている。従って、任意の色の絵の具が要求された場合、そのモデルを作るためには各スペクトルに対応するパラメータを決定する必要がある。具体的には 5.6 節 での結果から可視帯域での 18 個の標本点におけるスペクトル反射率が必要となる。

しかし、ペイントツール等で一つの色の絵の具をユーザが決める際に 18 個のパラメータの設定がユーザに要求されるのでは不便である。また、仮に 18 個のパラメータ入力をいとわないとしても、スペクトル反射率を直接操作して望んだ色に対応するパラメータセットを決定することは非常に困難であると考えられる。また、自動描画アルゴリズム等では RGB 色座標等で表された任意の色に対応する絵の具モデルのパラメータセットを自動的に決定する必要がある。

スペクトルから色座標への変換方法は長く研究されており、CIERGB や CIEXYZ の重値係数を用いればよいのであるが、今回必要とする色座標からスペクトルへの変換に関しては、特殊な用途であり従来研究がされていない。

そこで本章では望んだ色の各標本点のスペクトル反射率を容易に決定するアルゴリズムに関して述べる。

命題としては、3次元で表される色座標から18個の反射率への値の変換であるので、拘束条件はあるものの解が一つであるということはない。これは、一つの色に対して多くの場合スペクトル反射率は一つとは限らないことを考えれば自明である。

今回、一つの色座標に対して一組のスペクトル反射率を決定するだけでは絵の具モデルへの利用としては不十分である。なぜなら、スペクトル反射率曲線の違いは絵の具モデルにとっては同一の色であっても混色した際の発色の違いとなって現れるからである。本章で述べるアルゴリズムでは、三次元の色座標に加えて新たなパラメータ“先鋭度 (sharpness)”を入力することにより、望んだ色に対応する多様なスペクトル反射率のセットを決定することが可能である。

先鋭度の違いが混色時における発色の違いについての実験結果に関しても本章では述べる。

## 6.2 Kubelka-Munkの理論の実装とパラメータ入力問題

Kubelka-Munkの理論では、薄膜内部の光の挙動を吸収係数と散乱係数によりモデル化しているため、望んだ絵の具のモデル化にはそれらの係数の決定が必要である。しかし、それら係数は反射率から計算が可能である事を第5章で示した。従って可視帯域における反射率曲線を与える事が出来れば望んだ絵の具のパラメータが決定できることになる。但し、第5章の結果からは混色計算を行うためには18個のサンプルポイントが必要であったので、一つの絵の具をモデル化するにあたり、18のパラメータ入力が必要とされる。

人間の目のセンサの構造から色は本来3次元で表すことが可能であり、第2章で見たように、 $RGB$ や $XYZ$ や $L^*a^*b^*$ 、またはCGで用いられる $HSV$ や $HSL$ 、もしくはテレビジョンで用いられる $Y, R-Y, B-Y$ のように[95]、従来の3次元色座標系を用いた色表現方法で表される絵の具ならば、3つのパラメータ入力で十分である。

これに対して18のパラメータ入力は非常に多い。しかも、それらのパラメータは色に対して独立では無く、図2.10で表される等色関数との積算により色への変換が行われるため、パラメータ変化と色の関係が複雑で直感的な操作が出来ない。そこで、絵の具の18の反射率パラメータを望んだ色から直感的に決定できるアルゴリズムが必要となる。

ここで、要求されるアルゴリズムの解くべき条件を整理すると次のようになる。

[条件 1] 色からの直感的なパラメータ決定が可能であること

[条件 2] 決定されるパラメータは 0 ~ 1 区間であること

[条件 3] スペクトル反射率曲線の形状は一山であること

[条件 4] スペクトル反射率曲線の形状を上各条件を満たしつつ変化させられること

[条件 1] は、絵の具の選択には色が第一に重要な要素であるため、最重要であると言える。ユーザのインタラクティブ操作、自動描画アルゴリズムでの自動的な絵の具選択アルゴリズムへの適応を考えると、色と値の対応が直感的であり、また一意に色との対応が取れる 3 次元色座標系で色指定を行う方法が好ましい。従って、3 つの入力パラメータから 18 個のパラメータへの変換が必要となる。

[条件 2] は、パラメータが反射率を表しているために生じる拘束条件である。

蛍光色では入射スペクトルと反射スペクトルの波長が違い、反射率が 1 を越えてしまうが、今回は蛍光色は対象としない。

[条件 3] は、一般の絵の具の反射率曲線を計測するとその形状が一つの山を表しており、この特徴は混色における色の変化の安定性にも関係があると言えるからである。その説明は次のとおりである。

今、2 色の絵の具があったとする。それら 2 色の絵の具にはそれぞれスペクトル反射率曲線がスパイク形状のものと滑らかな一山形状のもの二種類の等色絵の具があるとする。そこで、2 色の絵の具を混ぜた場合、スパイク形状の反射率曲線を持つ絵の具を混ぜた方は互いの反射率の谷間の部分により反射率が極度に落ちて補色関係に無くとも発色が悪くなる可能性がある。一方、滑らかな反射率曲線形状を持つ絵の具同士であれば、色相の近い絵の具を混ぜた場合、反射率が極度に落ちることはないと思像できる。

従って一山の形でスペクトル曲線形状があることが必要である。

[条件 4] も混色後の発色に関係する条件である。等色関係の絵の具であっても、構成するスペクトル曲線がなだらかな山形の場合と、ピークが激しい山形の場合とで、混色後の色には違いが生じる。従って混色後の絵の具の発色特性を決定するスペクトル形状を操作できるパラメータが必要となる。

以上の 4 条件を満たす直感的に操作可能な表面的なパラメータと、そのパラメータから絵の具の内部パラメータへの変換アルゴリズムが必要となる。

## 6.3 絵の具の特性決定のための直感的なパラメータ群

ユーザもしくは、自動描画アルゴリズム等において、絵の具を選定する際に操作する表面的なパラメータについて述べる。

### 6.3.1 色の選定パラメータ

色を扱う座標系に関しては、第2章で述べたように CIERGB、CIEXYZ、CIELAB、CIELUV、ntscRGB、sRGB、HSV 等がある。どれも、3次元空間であり、各色を一意に表現可能である。しかし、その中で、ほぼ等色差空間を表し、マンセル表色系の色相彩度を同心円型で表す事の可能な CIELAB 座標系を直感的な色指定に用いる事とする。従って、色の選定用入力パラメータには明度の  $L^*$ 、色相、彩度に関する  $a^*$ 、 $b^*$  を用いる。

### 6.3.2 スペクトル曲線形状決定パラメータ

さらに、スペクトル曲線形状を決定するパラメータを用意する。これはスペクトルの山形をなだらかな形にするか、尖った形にするか決定するものであるので先鋭度 (Sharpness) と名付ける。また、利用のしやすい様にどの色に対しても値域を 0~1 とし、0 の時に最もなだらかなスペクトル曲線形状を表し、1 の時に最も尖ったスペクトル曲線形状を表すものとする。

### 6.3.3 その他のパラメータ

濃度パラメータ、容量パラメータも第5章での絵の具モデルでは必要となる。しかしながら、それらは言わば、顔料調製後の話であり、最も重要な絵の具内部のスペクトル散乱係数、吸収係数の決定には関係が無い。従って、次節でのアルゴリズムには濃度パラメータ、容量パラメータは関係が無い。

## 6.4 直感的パラメータ群とスペクトル反射率の対応付

前節で決定した、絵の具選定のための表面的なパラメータ CIELAB の  $L^*$ 、 $a^*$ 、 $b^*$  及び先鋭度から、絵の具の内部パラメータであるスペクトル散乱係数、吸収係数を得るために必要となるスペクトル反射率を導出するアルゴリズムについて述べる。

## 6.4.1 スペクトルと色座標系の統一的な取り扱い

図 6.1 で模式的に表すような、可視帯域のスペクトル反射率と 3 次元である色座標系の二つを統一的に扱えることがパラメータ変換にはまず必要となる。

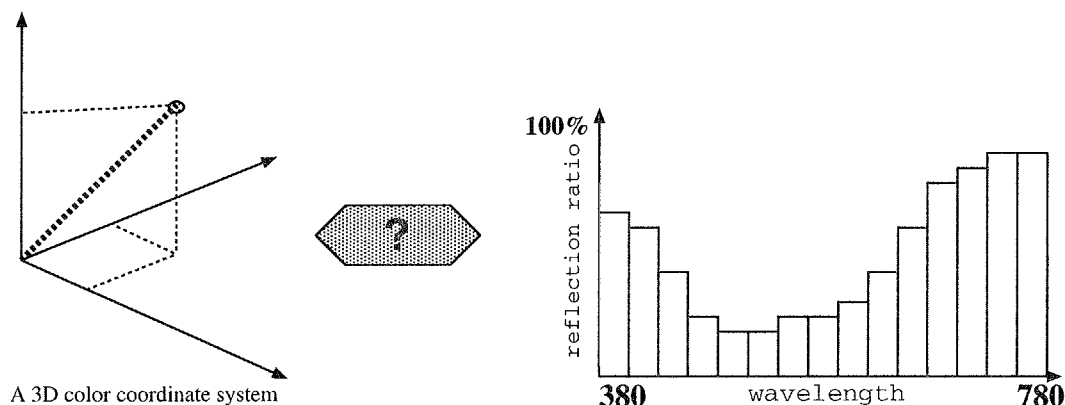


図 6.1 表面的なパラメータとスペクトル反射率を結び付ける必要がある

そこでまずはじめに、スペクトルの各サンプルポイント毎に、その反射率の値だけを 1 とし、他のサンプルポイントの反射率を 0 とする光と CIEXYZ の二度視野の D65 光源の重価係数を掛ける。こうして各サンプルポイントの反射率に対応する色ベクトルを CIEXYZ 表色系の中で表す事ができるようにする。なおこれは、サンプリング間隔が無限に小さければ、単一スペクトルで表される純色の CIEXYZ 色座標を求めている事になる。

次に CIELAB 色座標系の値を CIEXYZ へと変換する。これらは式 (6.1) で表される変換式を用いれば容易に求まる。

$$\begin{aligned}
 x &= x_0 \left( \frac{a}{500.0} + \frac{L + 16.0}{116.0} \right)^3 \\
 y &= y_0 \left( \frac{L + 16.0}{116.0} \right)^3 \\
 z &= -z_0 \left( \frac{b}{200.0} - \frac{L + 16.0}{116.0} \right)^3
 \end{aligned} \tag{6.1}$$

この様にして、表面的なパラメータである  $L^*$ 、 $a^*$ 、 $b^*$  とスペクトル反射率を一つの 3 次元座標系内で統一的に扱う事が可能となる。

全てを CIELAB の表色系内で表す事も当然可能ではある。しかし 6.4.3 項 で述べるアル

ゴリズム内で、色の混色計算を行う際にベクトル和の形式で色計算を行える方が便利である事から CIEXYZ 表色系での取り扱いとする。

#### 6.4.2 CIEXYZ 表色系の回転

次にこの空間内での色相と彩度の特徴量がどこに現れるか考える。

そこでまず、無彩色のベクトル方向を考えると、それは全スペクトルの反射率が1である完全反射面の色ベクトル方向であることがわかる。CIEXYZ 表色系では光のエネルギー和はベクトル和として表せるので、前節で求めた各サンプルポイントの反射率に対応する色ベクトルの総和が無彩色のベクトル方向となる。

ここで、無彩色のベクトル方向を Z 軸とする CIEXYZ 表色系を全体を回転させた表色系を新たに考える。これを X'Y'Z' 表色系と呼ぶ。この表色系では、図 6.2 のように Z' 軸を中心とする回転角が色相に、Z' 軸からの距離が彩度におおよそ対応する。また原点は完全な黒に対応する。なお、この表色系での可視色の存在領域を前節の各反射率のサンプルポイントに対応する色ベクトルで表すと図 6.3 の立体領域 (Color Solid) となる。

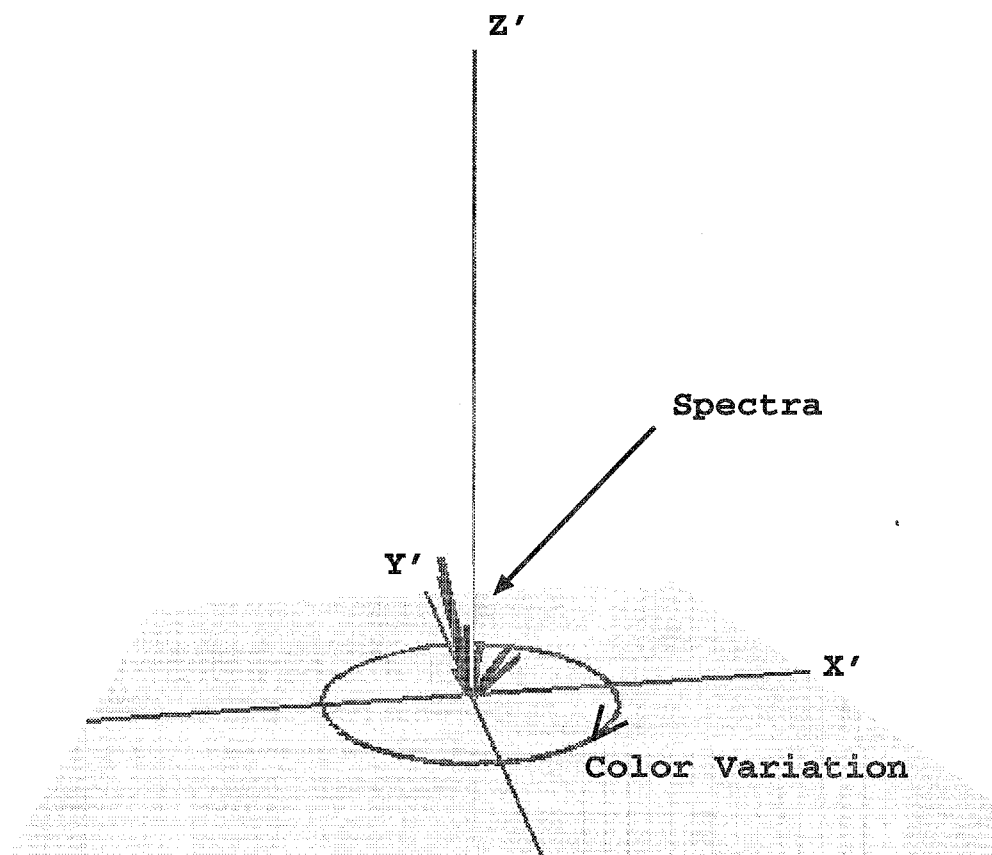


図 6.2 X'Y'Z' 表色系と色相彩度、無彩色の関係

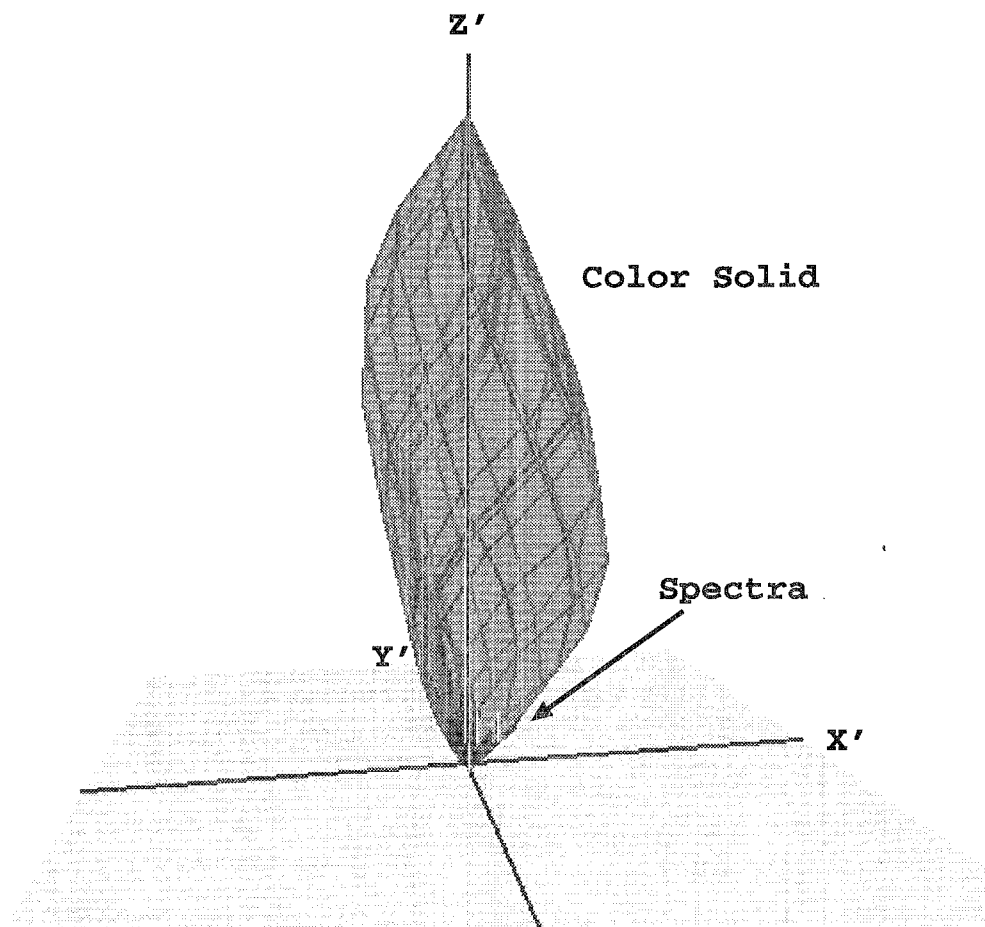


図 6.3  $X'Y'Z'$  表色系での可視色の存在領域



### 6.4.3 X'Y'Z' 表色系内でのパラメータ対応付けアルゴリズム

前節で定義した X'Y'Z' 表色系において表面的なパラメータをスペクトル反射率へ変換するアルゴリズムについて述べる。

準備として、 $L^*$ 、 $a^*$ 、 $b^*$  で与えられた色座標を X'Y'Z' 空間に配置する。この座標を  $T$  とする。 $T$  は可視色であるので、図 6.4 での色立体の境界面上もしくはその内側に存在している。

以下、アルゴリズムを次に示す。

- (1) 最大輝度の白色  $W$  から  $T$  を通る線分を X'Y' 平面へ引いた交点を  $a$  とする。
- (2)  $Oa$  ベクトルに表面的なパラメータである先鋭度 (Sharpness)  $0 \leq s \leq 1$  を掛けて位置ベクトル  $b$  をつくる。
- (3)  $b$  から  $T$  へ延ばした直線と色立体の境界面との交点を  $c$ 、 $d$  とする。
- (4)  $c$ 、 $d$  は色立体の境界面上なので、そのベクトル表示をスペクトル反射率  $0 \sim 1$  のベクトル和の形で一意に表す事ができる。
- (5)  $T$  は  $cd$  の内分点なので、ベクトルの内分計算により、 $cd$  から  $T$  をスペクトル反射率のベクトル和の形で一意に決定できる。

このアルゴリズムは色立体の表面の点  $c$  と  $d$  の内分点として対象色を表しているため、色立体が凸立体でなければならない。凸立体でなければ、 $cd$  の内分点が色立体の領域外に存在する可能性が生じてしまう。

しかしながら、スペクトル反射率のベクトルは 400nm 以下のスペクトルと 700nm 以上のスペクトルの一部を除けば傘の骨のように放射上に広がっているので、それらのベクトルで張られる空間は凸立体となる。400nm 以下と 700nm 以上の一部のスペクトル反射率のベクトルに関しては例外処理として、400nm 以下、700nm 以上の反射率ベクトルの和をそれぞれ一つのベクトルとして、先の傘の骨に加えて凸立体の一部を構成している。

先鋭度の意味合いは、対象色へのベクトル合成によって原点  $O$  の黒からどのように近づくかを考えればわかる。

ベクトル  $d$  はベクトル  $c$  と他のベクトルの和であるので、 $T$  のベクトルは  $\vec{c} + n\vec{cd}$  となる。従って、 $c$  が  $O$  から遠ければ、対象色  $T$  と位相の近い方から順にスペクトル反射率成分を使って対象色のベクトルを表現していることになり、逆に  $c$  が  $O$  となる先鋭度 0 の場合には、出来るだけ広い色相のスペクトル反射率ベクトルを用いて  $T$  を表すことになる。

また、 $a$  を上のようにして決定したのは  $d$  が対象色と色相において逆位相にならないようにするためである。

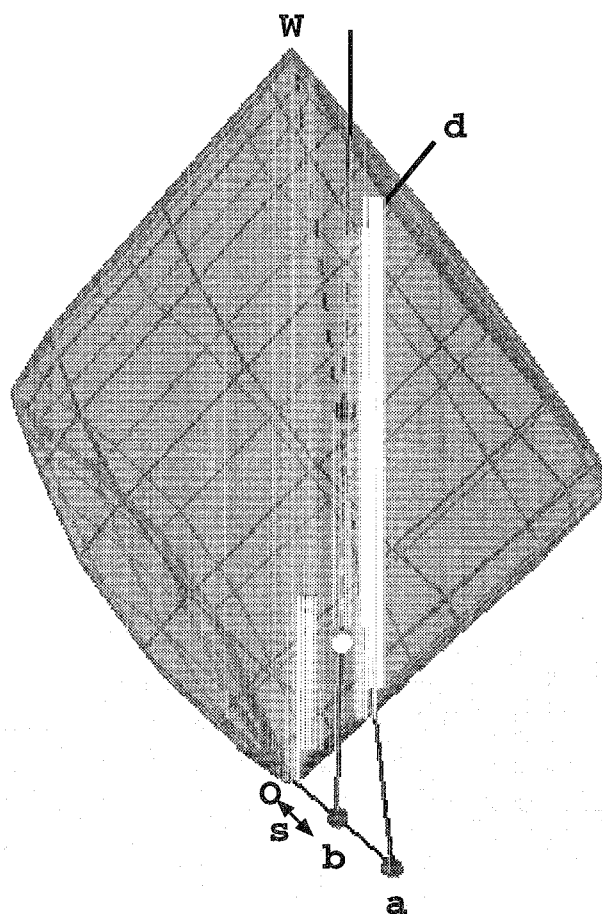


図 6.4 色座標から反射率への変換アルゴリズムの説明図

#### 6.4.4 パラメータ対応付けアルゴリズムの十分性の検討

このアルゴリズムでは、先に 6.2 節 で述べた条件をすべて満たしている。

[条件 1] に対しては、表面的なパラメータとして色座標の入力をさせており、また、それに合った反射率のスペクトルデータを出力する事が出来ている。

[条件 2] に対しては、色立体と直線との交点探索アルゴリズムで負のベクトルを許さないことで満たしている。

[条件 3] に対しては、図 6.4 での  $c$  と  $d$  が常に対象色  $T$  と同じ色相にある色立体の境界面上の点であるため、その合成スペクトルは常に一山となることが保証される。

[条件 4] に対しては、常に対象色の反射率を得られることが保証しつつ、対象色と近い色相のスペクトル反射率から値を大きくして対象色を作るという戦略と出来るだけ広い色相のスペクトル反射率を使って対象色を作るという戦略の比率を一つのパラメータで表す事が出来ているため満たしているといえる。

#### 6.4.5 パラメータ対応付けアルゴリズムの改良

6.4.3 項 で述べたアルゴリズムは 6.4.4 項 で述べたように条件を満たしてはいるが、6.5 節 で示すようにスペクトルの山型の形状としては切り立っており、実際の絵の具のスペクトル反射率の形状と比較すると違いが大きい。また、グラデーションへの塗り重ねでスペクトル反射率の形状が切り立っていると混色時に極端な変化が生じる事も考えられる。

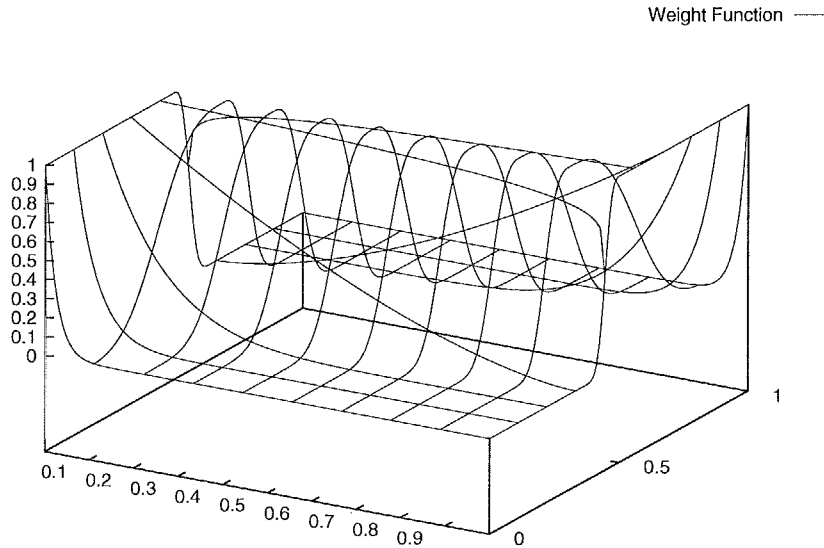
そこで、スペクトル反射率の山型を滑らかにするような若干の改良を 6.4.3 項 で述べたアルゴリズムに対して行う。

先鋭度 (Sharpness) の意図することは改良アルゴリズムと旧アルゴリズムで同じであり、値域も  $0 \sim 1$  のままであるが、そのアルゴリズム内部での取り扱い異なる。そこで、アルゴリズムの説明の中では以前の先鋭度を旧先鋭度、新しいアルゴリズムのための先鋭度を新先鋭度と呼ぶ。以下に、改良アルゴリズムを示す。

- (1) 色座標  $T$ 、新先鋭度  $S$  の入力
- (2) 旧先鋭度  $S_n$  を  $0$  から  $1$  の区間を等間隔で離散化した値とし、旧アルゴリズム  $f_r$  を用いて、それぞれの場合のスペクトル反射率  $r_{n,\lambda_i}$  を求める。

$$r_{n,\lambda_i} = f_r(T, S_n) \quad (6.2)$$

- (3) 新先鋭度  $S'$  を用いて  $0 \leq x \leq 1$  区間で定義されるある重み関数  $f_{wS'}$  を式 (6.3) で定義

図 6.5 重み関数  $f_{wS'}$  の概形

する。式 (6.3) の概形を図 6.5 に示す。

$$f_{wS'}(x) = f_w(x, S') = \begin{cases} 0 \leq S \leq 0.5 & (1-x)^{340(-S'+0.5)^3} \\ 0.5 < S \leq 1 & x^{340(S'-0.5)^3} \end{cases} \quad (6.3)$$

- (4) 旧先鋭度  $S_n$  を用いて作ったスペクトル反射率  $r_{n,\lambda i}$  と  $f_{wS'}(S_n)$  から出力となる最終的なスペクトル反射率  $r_{\lambda i}$  を式 (6.4) で求める。

$$r_{\lambda i} = \frac{1}{\sum_n f_{wS'}(S_n)} \sum_n r_{n,\lambda i} f_{wS'}(S_n) \quad (6.4)$$

新アルゴリズムで求めたスペクトル反射率は旧アルゴリズムで求めた複数のスペクトル反射率の重み付平均となっているため、曲線を滑らかにする事が出来る。また、新アルゴリズムでの先鋭度は重み関数の形状を決定するために使われているが、値が1の時、旧アルゴリズムでの旧先鋭度の1の側に大きな重みを付け、値が0の時は旧先鋭度の0の側に大きな重みを付けているので、先鋭度の意図どおりに出力のスペクトル反射率曲線を制御することができる。

## 6.5 直感的パラメータ入力による絵の具の混色実験

本節では、6.4.3項と6.4.5項で述べたアルゴリズムで求められるスペクトル反射率曲線での先鋭度(Sharpness)による制御実験と、先鋭度の違いにより生じる混色結果の違いの確認実験について述べる。

### 6.5.1 スペクトル反射率曲線形状の先鋭度による制御実験

$L^* = 50, a^* = 20, b^* = 4$ の色で、先鋭度を0から1へ段階的に変化させた時の旧アルゴリズムにより決定されたスペクトル反射率曲線の結果を図6.6に示す。

同様に  $L^* = 50, a^* = 20, b^* = 4$ の色で、先鋭度を0から1へ段階的に変化させた時の新アルゴリズムにより決定されたスペクトル反射率曲線の結果を図6.7に示す。

二つのアルゴリズムでのスペクトル反射率曲線の違いはピークからの下がり方が切り立っているか否かという点、対象色と最も関係の深いスペクトルの反射率を最優先するかしないかという点において違いがあることが分る。しかし、どちらも先鋭度を変化させることにより、一山型を維持したままスペクトル反射率曲線の変化を可能にしていることが分る。

### 6.5.2 先鋭度を変化させた等色絵の具の混色実験

先鋭度の違いが混色後の発色の違いにどのような影響を与えるかの実験を行う。

先程の色  $L^* = 50, a^* = 20, b^* = 4$ で先鋭度を様々に変化させた絵の具を用意し、その補色となる  $L^* = 70, a^* = -20, b^* = -4$ の絵の具への混ぜ合わせを行う。その結果を図6.8と図6.5.2に示す。図6.8が旧アルゴリズムによる先鋭度を変化させた絵の具による結果、図6.5.2は新アルゴリズムによる結果である。図中、横方向は先鋭度の変化を表しており、縦方向は混ぜた量の変化を表している。

どちらのアルゴリズムの絵の具の結果からも先鋭度が強い程、混色後の発色において元の絵の具の色を強く残すという特徴があることがわかる。

従って、本アルゴリズムの先鋭度は混色時の絵の具の色の強さを制御できる有用でありかつ容易なパラメータであるということが出来る。

### 6.5.3 新旧スペクトル反射率曲線決定アルゴリズムによる絵の具の混色実験

次に様々な色相の色に対して新旧スペクトル反射率曲線決定アルゴリズムで作られた絵の具を使って混合混色実験を行う。

はじめに明度  $L = 65$ 、先鋭度  $S = 0.2$ の各色相の絵の具を用意し、そこへ  $L^* = 65, a^* = 50, b^* = 10$ の絵の具を混ぜた絵の具の発色を調べる。

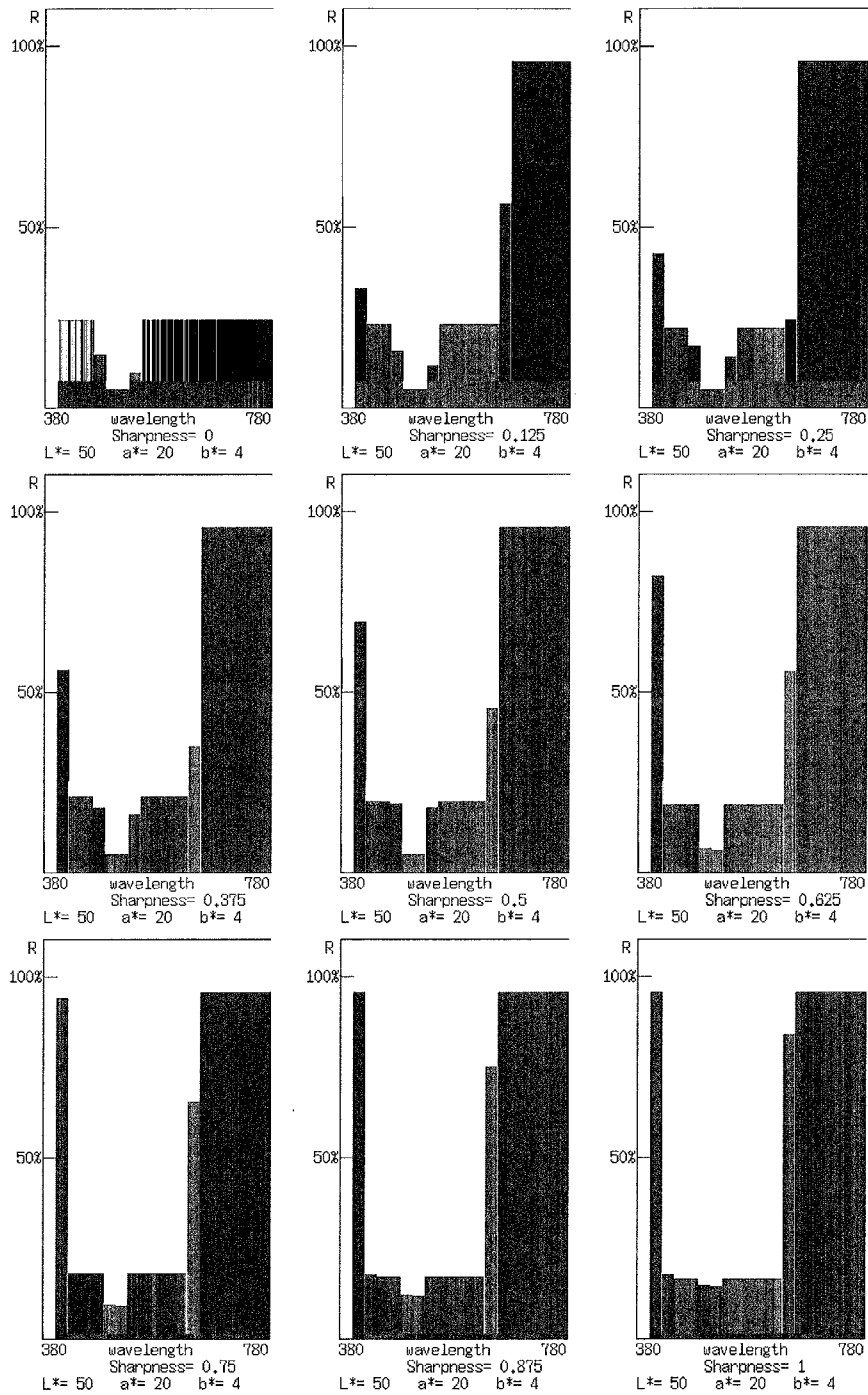


図 6.6 旧アルゴリズムによるスペクトル反射率の決定結果

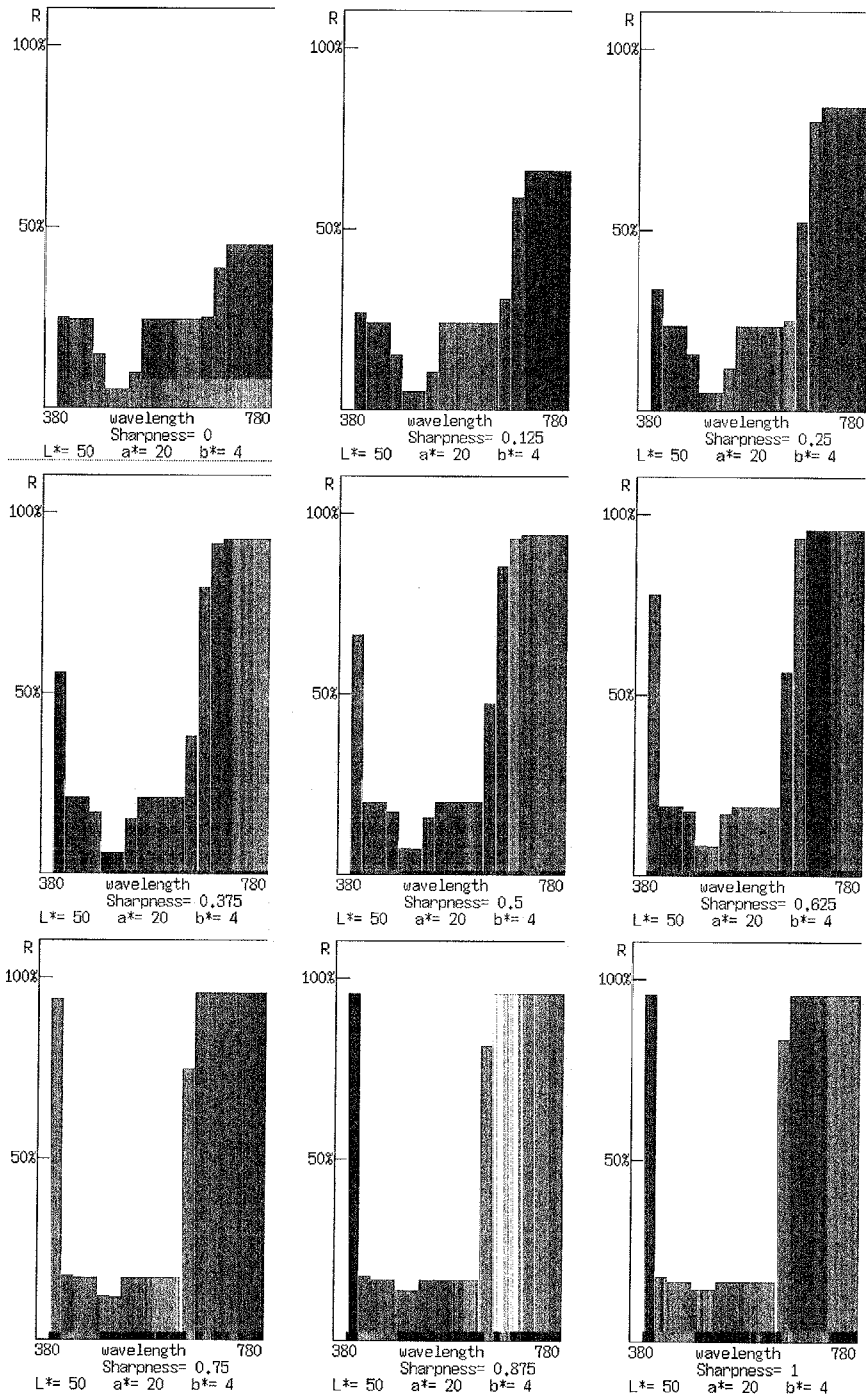


図 6.7 新アルゴリズムによるスペクトル反射率の決定結果

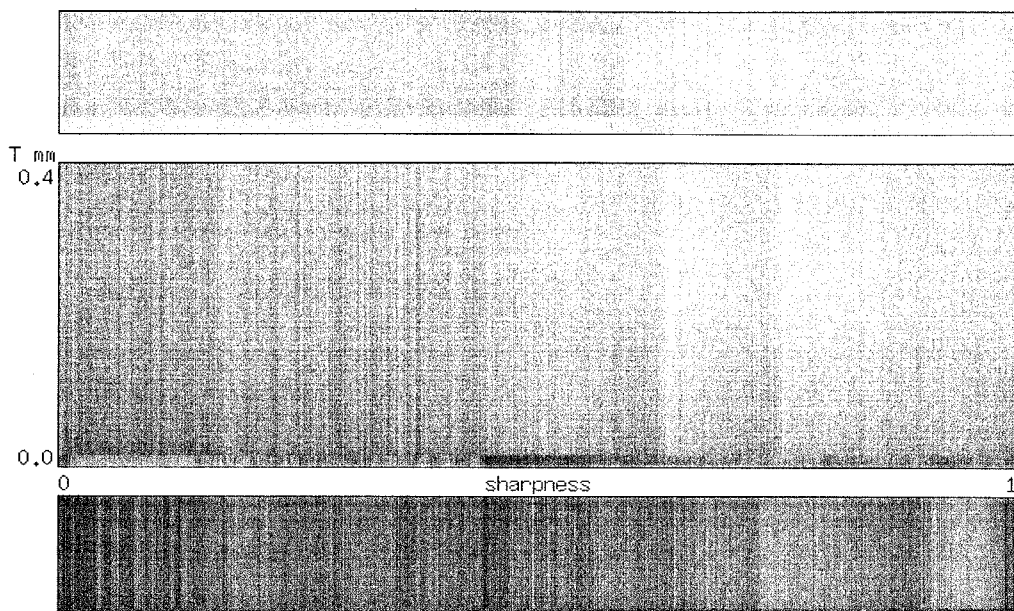


図 6.8 旧アルゴリズムにより先鋭度を変化させた絵の具による混合混色結果

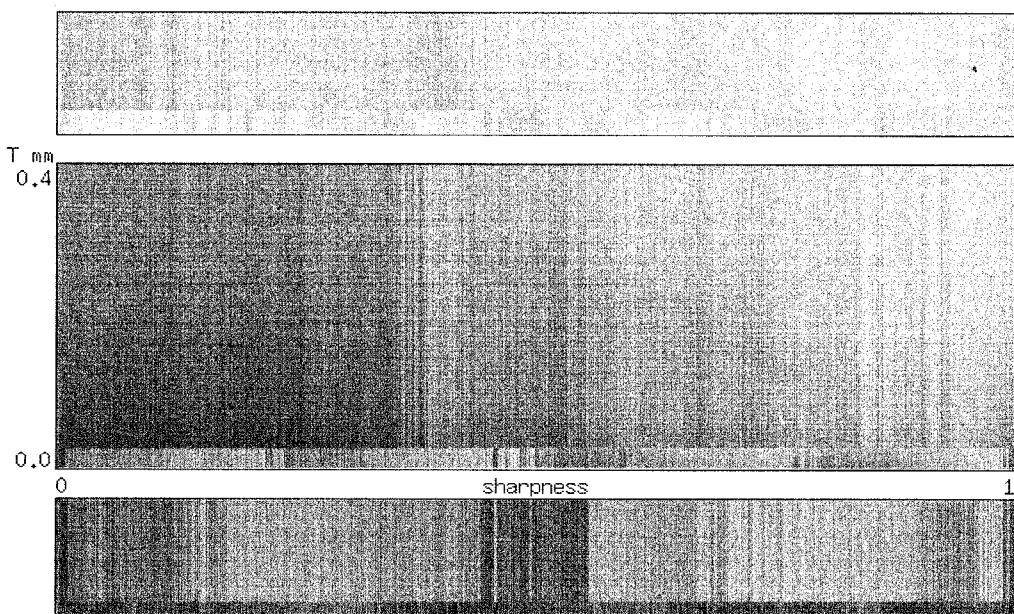


図 6.9 新アルゴリズムにより先鋭度を変化させた絵の具による混合混色結果



先鋭度が小さい場合に特にアルゴリズムの差が出ると考えて  $S = 0.2$  とした。混ぜ合わされる絵の具の色相の変化に対して混合混色後の発色が滑らかに変化している場合の方が安定した混色特性であると言える。

旧アルゴリズムにより作られた絵の具による混合混色結果を図 6.10 に新アルゴリズムにより作られた絵の具による混合混色結果を図 6.11 に示す。

図の上段は混ぜ合わせる絵の具の色である。最下段は混ぜ合わされる様々な色相の絵の具である。上から 3 段目は混合混色結果を表し、縦方向で混ぜ合わせる絵の具の先鋭度を  $0 \sim 1$  に変化させた時の混色の違いを表している。上から 2 段目は混ぜ合わせる絵の具を先鋭度  $S = 0.5$  とした時の各色相の色との混合混色結果を明度  $L$  を実線で色相  $V$  を破線で表している。

図 6.10 と図 6.11 の 2 段目のグラフから分るように、新アルゴリズムの方が、混合される絵の具の色相変化に対して混色結果が滑らに変化していることが分る。また、3 段目の混色後の発色結果から微妙な色変化の不自然さがどちらのアルゴリズムにもあることがわかるが、その不自然さも新アルゴリズムの方が少ないと言える。旧アルゴリズムでは、混合する絵の具の色相の変化と混色結果の色相の変化に明らかな逆行が一部で見られる。このような色変化は描画時のグラデーションを不自然にしてしまう恐れがある。一方、新アルゴリズムではそのような色変化を相対的に少なくすることが出来ている。

以上から旧アルゴリズムと新アルゴリズムとのスペクトル反射率曲線の形状の違いが色相を変化させて混色させた時の発色の違いとなって明確に現れ、新アルゴリズムによる滑らかなスペクトル反射率曲線の方が良い混色結果を得られることが分った。

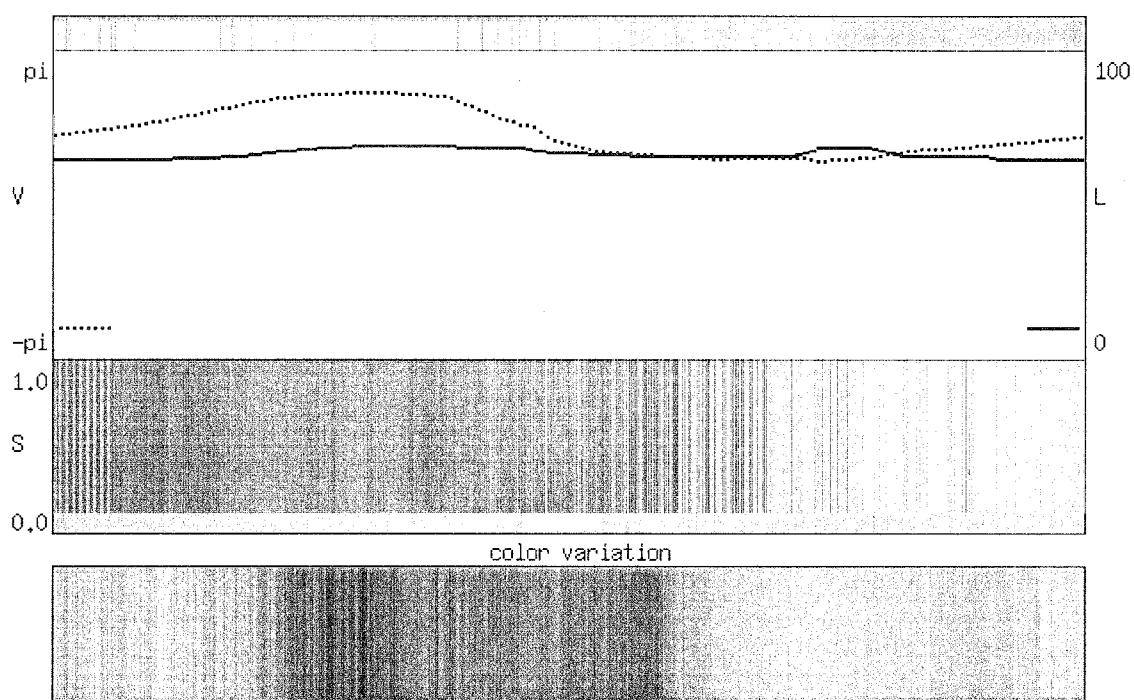


図 6.10 旧アルゴリズムにより作られた絵の具の混合混色発色実験結果

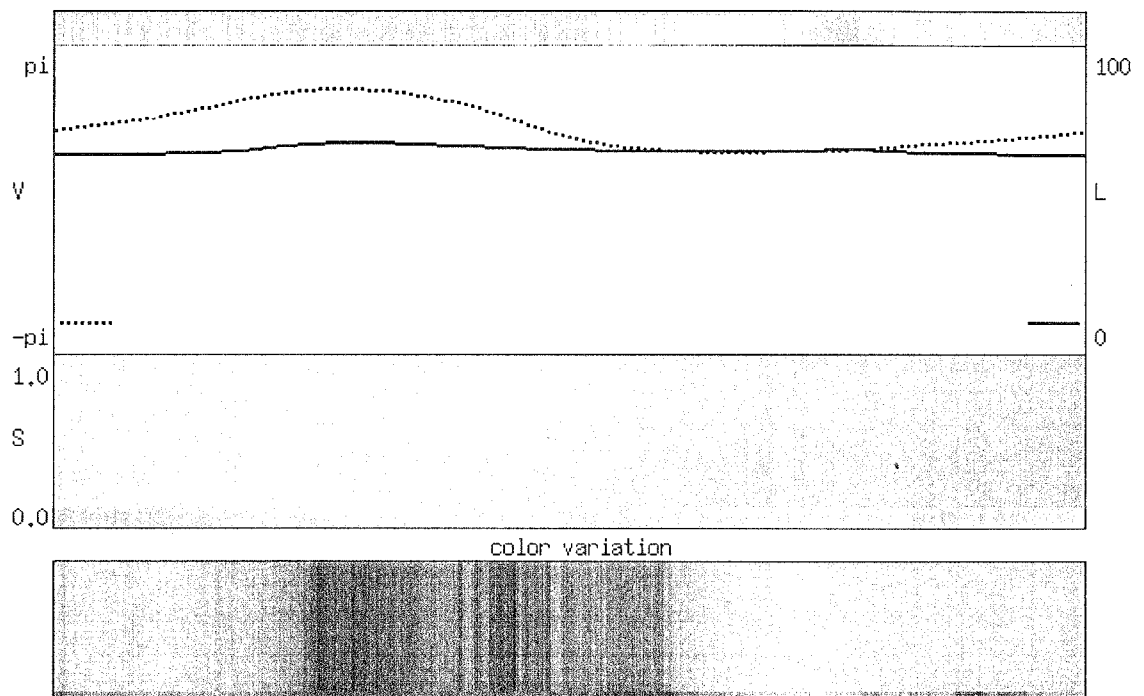


図 6.11 新アルゴリズムにより作られた絵の具の混合混色発色実験結果

## 6.6 むすび

本章では任意の色の絵の具のスペクトル反射率を直感的なパラメータの操作により決定できるアルゴリズムについて述べた。このアルゴリズムでは色からスペクトル反射率をただ導出するだけでなく、スペクトル反射率形状も容易に制御でき、混色後の色の強さに関する絵の具の特徴を制御することが可能であるという特徴がある。

計算機上の仮想的な絵の具のスペクトル反射率の決定という限られた用途に向けたアルゴリズムとして述べたが、多光源による任意の色、演色性を持つ光源の制御ということに関しても本手法は有効である。

今後は、混合混色において一部不自然な発色を示すことからより滑らかなスペクトル反射率を求めることが出来るようにアルゴリズムのさらなる改良を行う予定である。

## 第 7 章

# 計算機上の筆モデル

### 7.1 まえがき

従来のペイントツールでは多くの場合、筆の接地面の形状に円盤が使われている [37][38]。そして、その円盤の直径をストロークの速度、もしくはタブレットによる入力の場合に得られる筆圧により変化させる事により、ストロークの多用性を出している。しかしこの手法では“はね”や、“払い”などのストロークの表現力に限界がある。

ストロークを一つ細長い模様として捉え、伸長、湾曲させて筆跡領域にマッピングし、ストロークの表現を高める手法についての研究 [34] もあるが、実際の筆の持つ自由な筆跡、操作と結果の直接的な対応付けを得ることが出来ない。

また、ストロークの形状を一つの細い背骨線とそれに垂直な線分で表現し、墨の定着量を背骨線のノード上の値により制御して、墨絵風な画像を作成することも試みられている [27][28]。しかし、これらの手法も毛先の軌跡が柄の軌跡と異なることにより生じる曲線を描いた際の太さの変化が表現できない。

そこで、本章では筆の毛先を計算機上で物理モデル化してシミュレートし、そこから筆跡を得ることにより、筆の柄の制御から直接的にストロークを制御でき、さらにストロークの表現力の多用性を高めることが可能な入力手法について述べる。

## 7.2 筆のモデル化ステップ

ストロークとは、筆の接地面の軌跡により現れる模様である。そして、接地面の形状変化の制御は、実際の場合筆の柄を通して行われる。

現在、ポインティングデバイスとして、マウス以外のものにペンタブレット型のデバイスがある。このデバイスでは  $xy$  座標と同時に筆圧、ペンの傾きを入力することが可能である。現在は従って、柄の状態入力をリアルタイムに行える環境になっている [93][94]。

そこで、柄の姿勢を入力として、その先の筆の毛の状態を物理シミュレートして求め、筆の毛と画材との接地面を計算することを考える。最後に筆の内部の液体移動を計算するアルゴリズムを検討する。

## 7.3 筆の背骨線モデル

実際の筆先には多量の毛が付いており房をなしているが、毛全体の状態のシミュレーションは処理が非常にかかることが予想される。今回は画材との接地面が求めれば良いのであるから、図 7.1 に示すように筆の房の形状を毛の背骨線 (spine curve) で代表させることにする。

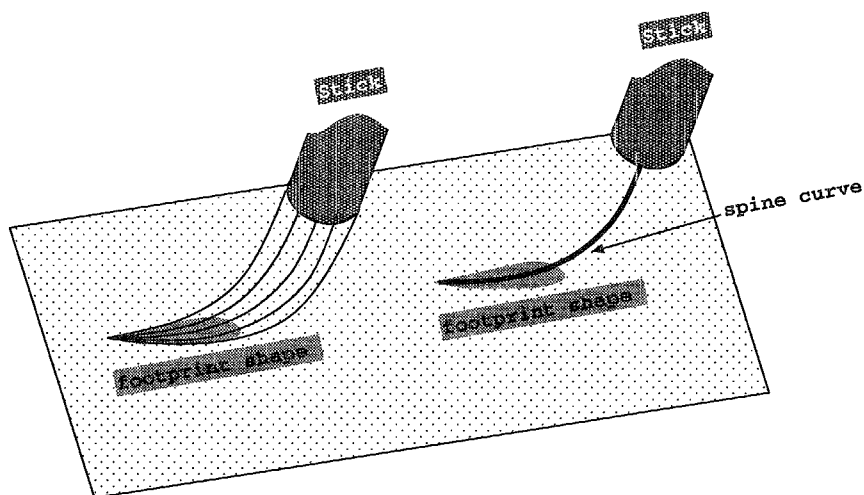


図 7.1 筆の毛のモデリング

### 7.3.1 筆の背骨線の姿勢シミュレーション

筆の背骨線には、直線部と関節部からなるモデルを考える。その構造上の特徴は次のとおりとする。

- (1) 直線部の長さは変化しない
- (2) 関節部は2自由度を持ち、曲げ角に正比例して直線に復元しようとする力が働く
- (3) 関節部及び先端部は質量を持つが、直線部は質量を持たない

なお、節の数が多い程、モデルとしては厳密であるが、シミュレーションの計算量が増大する。そこで、節の数  $n$  を1～3の中から実験により求めることとする。

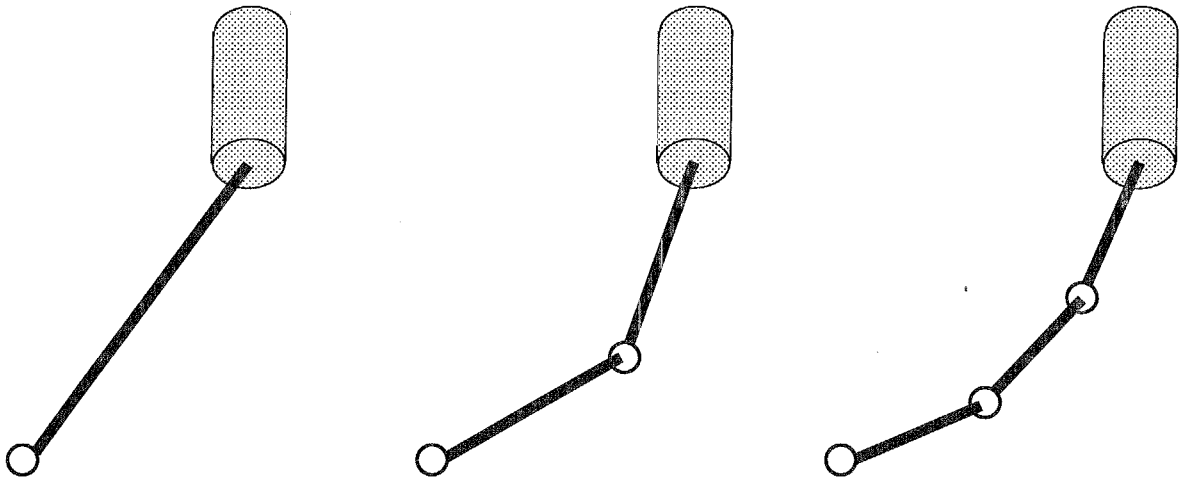


図 7.2 様々な節の数の背骨線のモデル

次に、背骨線の系全体のエネルギーの計算式を検討する。まず始めに、背骨線の各部と変数、定数の対応付けを図 7.3 で行う。

$m_n$  は質量、 $l_n$  は長さ、 $v_n$  は位置座標、 $\theta_{bn}$  は曲げ角を表す。 $\theta_{xn}$ 、 $\theta_{yn}$  は背骨線の姿勢を表す回転角であり、 $v_n$ 、 $\theta_{bn}$  は  $\theta_{xk}$ 、 $\theta_{yk}$  ( $k = 0 \sim n$ ) 及び柄の付け根の座標、傾きから二次的に求められる。

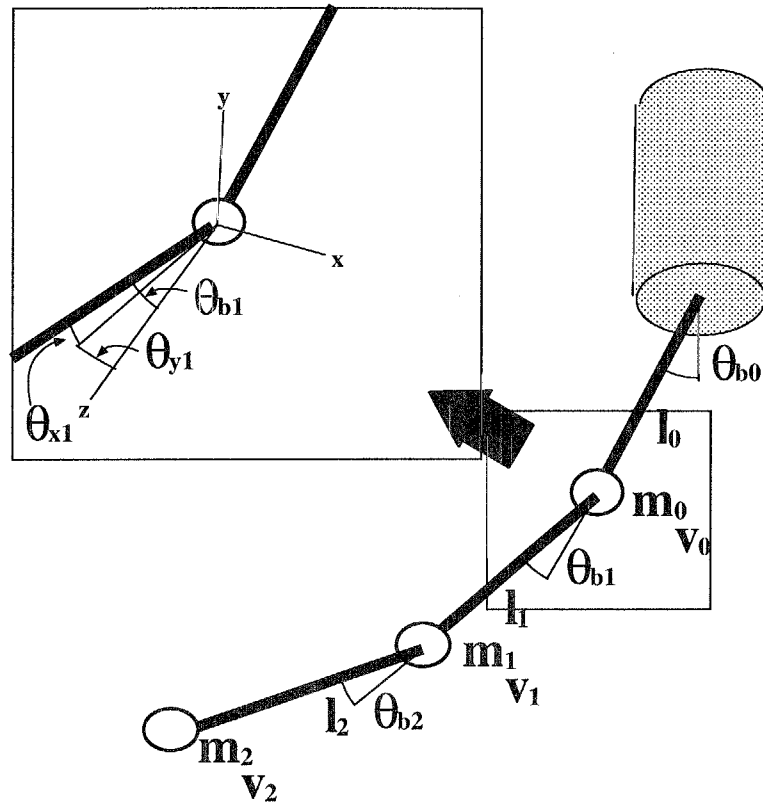


図 7.3 背骨線の各部と変数、定数の対応付け

## [位置エネルギー]

座標  $v_n$  にある各関節部及び、先端部の質量  $m_n$  が持つ位置エネルギー  $E_{pn}$  を次のとおりとする。ここで、 $g$  は重力加速度である。

$$E_{pn} = m_n g v_{nz} \quad (7.1)$$

## [運動エネルギー]

各関節部及び、先端部の質量  $m_n$  が速度  $s_n$  の時の運動エネルギー  $E_{kn}$  を次のとおりとする。

$$E_{kn} = \frac{1}{2} m_n s_n^2 \quad (7.2)$$

## [曲げエネルギー]

各関節部における、曲げにより蓄えられるエネルギー  $E_{bn}$  は曲げ角を  $\theta_{bn}$  とした時の抗力が  $K_{bn}\theta_{bn}$  であることから式 (7.3) で表される。



$$\begin{aligned}
 E_{bn} &= \int K_{bn} \theta_{bn} d\theta_{bn} \\
 &= \frac{1}{2} K_{bn} \theta_{bn}^2
 \end{aligned}
 \tag{7.3}$$

[摩擦による仕事]

シミュレーションを行う際には、時間を離散化するが、もしも、 $m_n$  が画材と接触している場合には前の時刻から現在の時刻までの移動距離に応じた仕事が生じる。ここでは、画材との垂直抗力を単純に  $m_n g$ 、摩擦係数を  $\mu$ 、水平移動距離を前の時刻と現在の時刻の位置  $v_{nt-1} v_{nt}$  から、 $disxy(v_{nt-1}, v_{nt})$  として摩擦による仕事を式 (7.4) であらわす。

$$E_{fn} = \mu m_n g disxy(v_{nt-1}, v_{nt}) \tag{7.4}$$

以上から背骨線の系全体のエネルギー  $E$  は式 (7.5) となる。

$$E = \sum_n E_{pn} + E_{pn} + E_{kn} + E_{fn} \tag{7.5}$$

ここで、背骨線の姿勢をシミュレートする問題は、 $E$  の最小値を取るパラメータを求めることに等しい。即ち、系の外部から背骨線の付け根である柄の下端の座標  $x$ 、 $y$ 、 $z$ 、及び、傾き  $tiltx$ 、 $tilty$  が与えられたとき、それらを定数として式 (7.5) の  $E$  が最小値となるような  $\theta_{xk} \theta_{yk} (0 \leq k \leq n)$  を求めれば良いことになる。

### 7.3.2 エネルギー最小値を求める方法

式 (7.5) の  $E$  が最小値となるようなパラメータ  $\theta_{xk} \theta_{yk}$  を求めるため Newton-Raphson 法を用いる [87][88][89]。

最小値では微分値が 0 となることから、式 (7.6) を Newton-Raphson 法で解けば局所解が求まりそうである。しかし、筆の毛先は常に画材よりも上にあるとする条件、 $m_n$  の  $z$  座標は常に正であるとする拘束条件を付け加えなければならない。そのため、スラック変数と呼ばれるダミー変数  $s_n$  を用いて拘束条件の不等式 (7.7) を等式 (7.8) へと変換し、さらにラグランジュの未定乗数  $\lambda_n$  を用いて拘束条件を含んだ最適化問題の式 (7.9) へと式 (7.6) を書き換える。

$$\begin{aligned}
 \frac{\partial E}{\partial x_i} &= 0 \\
 (x_0, \dots, x_{2n-1}) &= (\theta_{x0}, \dots, \theta_{x(n-1)}, \theta_{y0}, \dots, \theta_{y(n-1)})
 \end{aligned}
 \tag{7.6}$$

$$v_{nz} \geq 0 \quad (7.7)$$

$$v_{nz} - s_n^2 = 0 \quad (7.8)$$

$$\frac{\partial}{\partial x_i} \left( E + \sum_j \lambda_j (v_{jz} - s_j^2) \right) = 0 \quad (7.9)$$

$$(x_0, \dots, x_{4n-1}) = (\theta_{x0}, \dots, \theta_{x(n-1)}, \theta_{y0}, \dots, \theta_{y(n-1)}, s_0, \dots, s_{(n-1)}, \lambda_0, \dots, \lambda_{(n-1)})$$

式(7.9)を解くことにより節の回転角度変数 $\theta_{xn}$ 、 $\theta_{yn}$ が得られる。なお、Newton-Raphson法で用いる初期値は離散時間の一つ前の時刻の変数セットを用いる。

### 7.3.3 筆の背骨線のノード数の決定

式(7.9)からわかるように、一つの節に対して、二つの角度変数、スラック変数、ラグランジュの未定乗数の4つの変数が必要になる。Newton-Raphson法では、 $n$ 変数の時、 $n \times n$ の行列を用いるため、リアルタイム処理には節の数を極力少なくすることが望ましい。

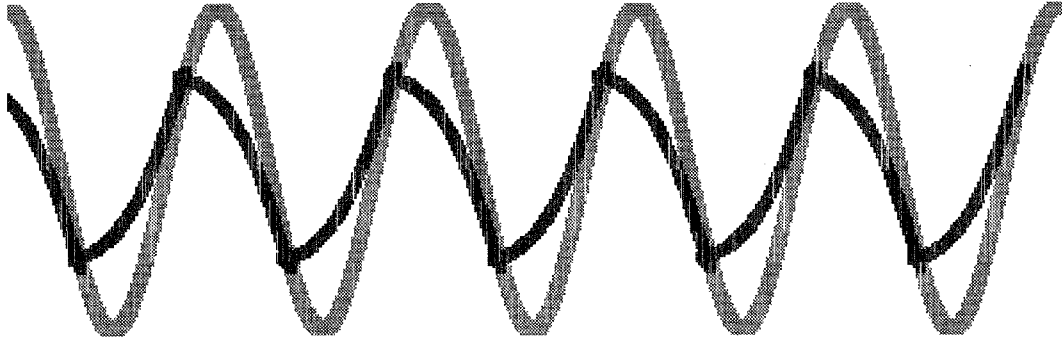
そこで、節の数を1, 2, 3として、2周期分のサイン曲線を描く際の計算時間を測定した。その結果が表7.1である。一周あたり約470回、背骨線の姿勢を算出する計算を行っている。

表からは、行列計算は $n^2$ の計算量の増大であるが、収束に要する繰り返し回数が増加するため実際には $n^2$ 以上に計算時間が増加していることが分る。

リアルタイムで必要とされる処理には、筆の姿勢計算以外に接地面の計算、絵の具の移動がある。そのため節の数を少なくし、計算時間を押えなければならない。一方で、節の数が多し程、毛の複雑なねじれを再現することが可能である。図7.4は節数1で $\sin$ 曲線を描画した際の結果である。濃い線が筆先の軌跡、薄い線が柄の軌跡である。この図からわかるよ

表 7.1 節の数と計算時間の関係

節の数 n	時間 (sec.)	比率 (n=1 の時間を 1 とする)
1	152	1
2	624	4.11
3	1755	11.55

図 7.4 節数 1 で  $\sin$  曲線を描画した際の結果

うに節数 1 の場合でも筆先が柄の軌跡に遅れて付いて来ることが再現できたことから節の数は 1 に決定する。

## 7.4 筆の接地面の計算

筆の毛先の動的形状に合わせて変化する画材との接地面を求める手法について述べる。

### 7.4.1 背骨線の曲線形状作成

前節で、背骨線を節数 1 で表現することにした。従って、このままでは背骨線は毛先と柄の付け根を結ぶ直線で表される。しかし、自然な毛先を作るためには背骨線は曲線である必要がある。そこで、図 7.5 で示すように背骨線の二つの端点と柄の軸を延長した線と  $z = 0$  の平面との交点の 3 点を制御点とする Bezier 曲線を作り新たな背骨線とする。中央の制御点を柄の軸を延長した線と  $z = 0$  の平面との交点とすることにより、筆を寝かせた場合、立たせた場合の自然な曲線変化を表現することが出来る。

### 7.4.2 背骨線の姿勢からの画材と筆との接地面の導出

背骨線の曲線を軸として毛の房の形状を作成する。基本的には曲線を軸として、円盤を通して形状を作る。そのプロセスは次のとおりである。

- (1) 背骨線の曲線の長さを任意の比率で  $n$  分割する。

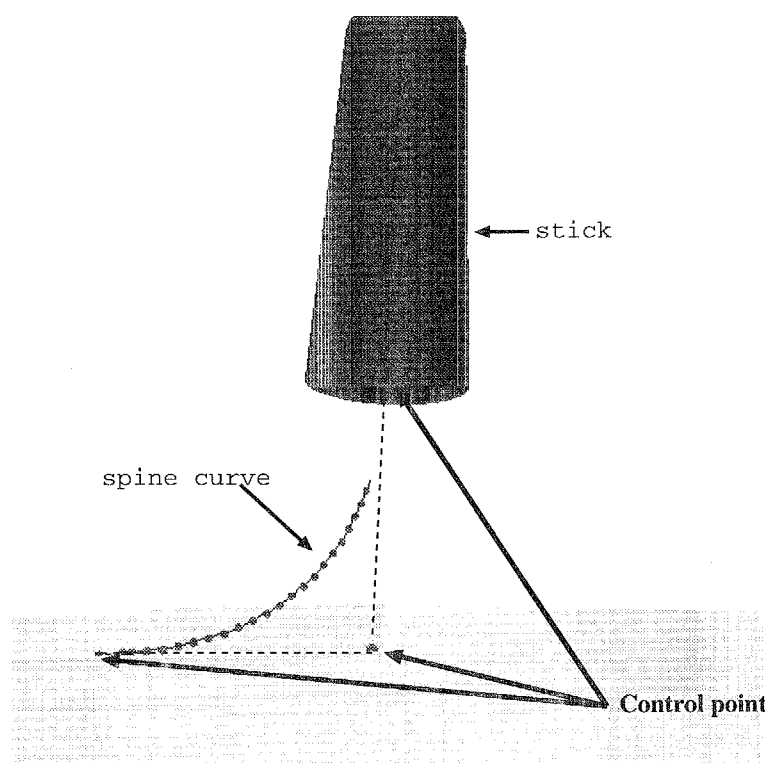


図 7.5 背骨線を曲線化するための Bezier 曲線用制御点

(2) 各分割点を中心とし、法線をその地点の背骨線の進行方向とする円盤を作成する。

背骨線の端点間の距離は一定であるが、曲線の長さは Bezier 曲線で作られるため、変化する。そこで、分割点は比率により定義しておく。

このようにして、丸筆の形状が作成される。次に筆と画材との接地面を計算する。

接地は各円盤が  $z = 0$  の面と交差した時に発生する。接地面の形状は各円盤が  $z = 0$  の面と交差した時に発生する線分の端点を結んで定義する。

## 7.5 筆の内部の液体移動

筆の内部には毛細管現象により絵の具が蓄えられている。また、紙等の画材と筆の間での絵の具の移動には重力に加えて、毛細管現象が働いている。

本節では、非常に簡素化して絵の具の蓄積・移動をモデル化する。

### 7.5.1 筆内部の絵の具蓄積モデル

7.4.2 項での円盤の一部に絵の具を保持できるタンクを定義し、タンクの間をパイプで結んだモデルを作成する。各タンクには、一定量の絵の具を蓄積することが出来る。図 7.6 ではタンクとパイプを緑色で表示している。毛の形状変化に応じてタンクの位置も移動する。絵の具の移動のプロセスは次の通りとする。

- (1) 絵の具にかかる力ベクトルを筆内部の絵の具の質量にかかる重力、接地面からの画材の毛細管現象による引力、筆内部の背骨線に沿って上方に生じる毛細管現象による筆の引力のベクトル和により決定する。
- (2) 力ベクトルの決定後、そのベクトル方向と近い向きへ各パイプ内の流れの方向を決定し、絵の具の移動を行う。
- (3) (2) のステップを規定回数分、繰り返した場合には、処理を打ち切り終了する。そうでなければ、絵の具の移動後の力ベクトルを再計算する。力ベクトルの長さがある値よりも大きい場合、不均衡であるとして、(2) のステップへと戻る。力ベクトルがある値よりも小さい場合には処理を終了する。

### 7.5.2 紙の絵の具蓄積モデル

仮想的な画材をモデリングする。モデルの対象は紙とし、水彩等で用いる荒目の用紙を考える。

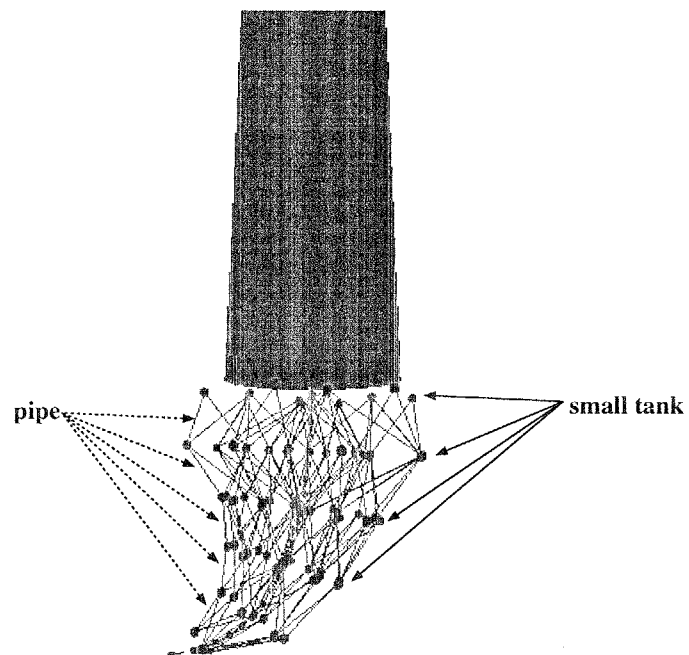


図 7.6 筆内部の絵の具の蓄積モデル 微小なタンクとパイプを持つ

画像にした際の1ピクセル分の面積を単位微小領域とし、その一つ一つをセルと呼ぶ。各セルは微量な絵の具を蓄えることが出来、その量はセル毎に微妙に異なるようにする。本手法では実際の紙のテクスチャをスキャナで取り込み、その濃淡からセル毎の蓄積可能量の不均一さを発生させている。また、蓄積可能量に比例して、1タイムステップあたりの絵の具の移動可能量と毛細管現象を似せた引力を定義している。

絵の具の移動のプロセスは次の通りとする。

- (1) 筆と接触したセルは筆の中の最も近いタンクへとパイプを繋げる。
- (2) 筆内部のタンクと同様な扱いで、セルを絵の具移動プロセスに組み込む。
- (3) 移動プロセスを筆の絵の具移動プロセスに任せる。

### 7.5.3 パレットの絵の具蓄積モデル

パレットは筆の絵の具移動プロセス内では紙とほぼ同一に扱われる。異なる点は、次の二つである。

パレットのセルには、絵の具が無限に存在している。

パレットのセルの毛細管現象に似せた引力は常に負である。即ち筆の方へ絵の具を移動させるように力が働く。

絵の具の移動プロセスは次のとおりとする。紙での絵の具の移動プロセスと同一である。

- (1) 筆と接触したセルは筆の中の最も近いタンクへとパイプを繋げる。
- (2) 筆内部のタンクと同様な扱いで、セルを絵の具移動プロセスに組み込む。
- (3) 移動プロセスを筆の絵の具移動プロセスに任せる。

## 7.6 提案筆モデルによる描画結果

本章で述べた筆モデルが自由度の高い入力を可能とすることを本節では実験結果を通して示す。

### 7.6.1 筆の柄の姿勢変化に応じた背骨線の形状変化結果

7.4.1 項 で述べた背骨線の曲線を筆の柄の姿勢変化により制御した結果を図 7.7 に示す。

筆の柄の中心軸が傾くことにより、その直線上の  $z$  平面との交点である曲線を定義する中央の制御点に変化している。その結果、傾ければ直線に、垂直に立てればしなった曲線に背骨線が変化できることがわかる。この応答は、実際の筆の反応と等しいことから、インタラクティブなペイントツールでの利用において、違和感無くユーザが操作できると考えられる。

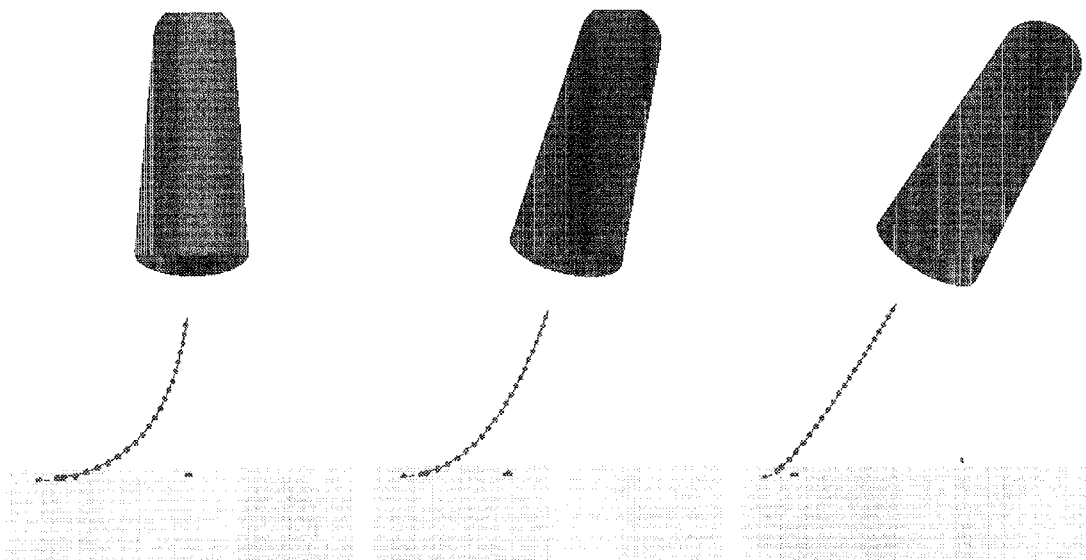


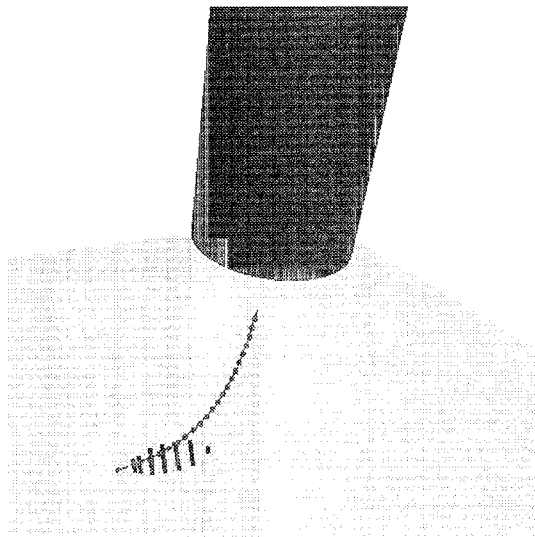
図 7.7 筆の柄の姿勢と中央制御点の移動、曲線形状の関係

### 7.6.2 筆の接地面の導出結果

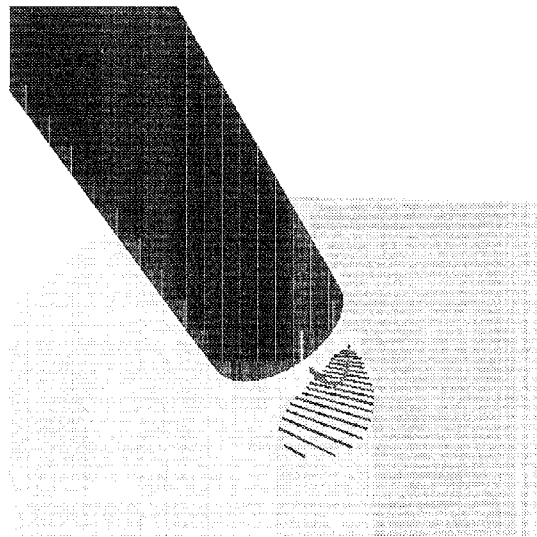
7.4.2 項 で述べた筆の接地面の導出結果の例を図 7.8 に示す。 $z$  平面に描かれている線分が、円盤と平面との交差により生じた線分であり、各線分の端点を結んだ形状が雫型の筆の跡になっていることがわかる。

柄が高い位置の時には小さな接地面となり、低くした時は大きな接地面になっている。また、図 7.8 の例からわかるように曲がった筆の跡を作ることも本手法では可能であることがわかる。

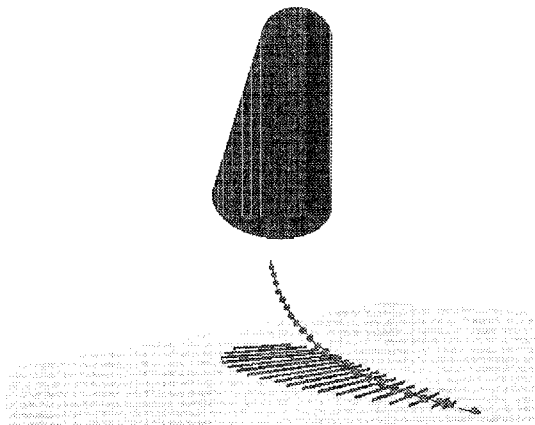




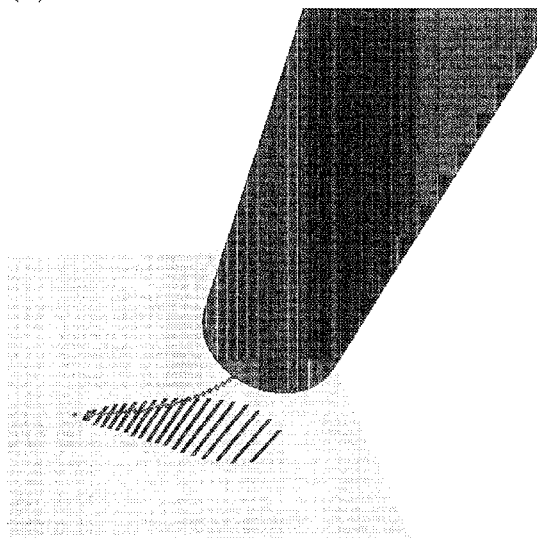
(a) 柄が高い位置の場合の接地面



(b) 柄が低い位置の場合の接地面



(a) 柄が低い位置の場合の接地面 2



(b) 柄が低い位置の場合の接地面 2(別アングル)

図 7.8 筆の接地面の形状導出結果例

### 7.6.3 筆による描画結果

本章の最終結果として、提案筆モデルによるストローク描画結果を示す。

図 7.9 は仮想的な筆の柄を微妙に姿勢変化させて引いた線の例である。単純な線ではあるが、筆の接地面が動的に変化することにより、始点から終点にかけて、濃淡、太さが変化して従来にない表現力の高い線が描かれている。

また、実際の紙のテクスチャから決定した紙の絵の具に対する吸収力の局所領域毎の微妙な変化が、色の濃淡を決定する要素として結果に効果的に現れていることがわかる。

このように本手法では、描画におけるもっとも基本となるストロークの表現力を高めることが可能であることがわかる。

図 7.9 筆モデルによる線描画例

## 7.7 むずび

本章では、表現力の自由度の高いストロークを描くことができ、また、意図したことを直接的に入力方法できるような手法を目指して、筆の物理モデリングを行った。

筆の接地面の形状は柄の姿勢制御と非常に直感的な対応関係が取れており、インタラク

タイプペイントツールへ用いた場合でも、ユーザが自由に制御できると予想される。

筆の接地面の形状の多様さは、従来の円盤の軌跡モデルでは困難であった曲線での微妙な変化を伴ったストローク入力を可能とした。

本手法による筆のモデルを用いたストロークの入力方法では、ストロークに多様さが生じるので、従来は難しかった個々のユーザの個性を容易に引き出すことが可能であると考えられる。

筆の接地面の導出に三次元的な筆の物理モデルを作り上げていることから、筆の絵の具の移動モデルを構築することも可能となった。今後はさらに絵の具の移動モデルを改良し、かすれ等の現象を再現することも可能であると思われる。

本手法で描画されるストロークは結果として濃淡のある複雑なテクスチャを持っているが、それらは筆の軌跡データから再現できる。このことから、毛筆などのデザイン化されたフォントセットデータを格納するための手法としても本手法は有効であると考えられる。

## 第 8 章

# 絵の具モデル・筆モデルの統合描画結果

### 8.1 まえがき

本章では、第5章から第7章において述べた個々の手法を統合した応用として、対話的な描画ソフトウェア (Interactive Painting Software) の実装例および、そのソフトウェアによる描画結果について述べる。

具体的なアプリケーションソフトウェアの実装例として、第6章で述べた絵の具の色決定アルゴリズムに対するグラフィカルユーザインタフェース (Graphical User Interface) の実装法、ペイントツールの全体の機能についての説明を行う。

その後で、ユーザによって実際にツールを使って描かれた幾つかの描画例を載せる。描画結果から絵の具モデルと筆モデルによる描画での表現力の高さを示す。

### 8.2 実装環境

本節ではペイントツールを実装するシステムのハードウェア及び、OS 環境についてまとめて示す。

システムには、IBM PC/AT 互換機アーキテクチャ上の Linux オペレーティングシステムを用いる。

具体的な仕様は次のとおりである。

## [ソフトウェア]

OS Kernel	Linux kernel version 2.0.34
Distribution package	Debian version 2.0
GUI system	XFree86 version 3.3.2

## [ハードウェア]

CPU	Pentium II 450MHz
Main Memory	256MB
Video Board & Video Memory	MGA MilleniumII VRAM 8MB
HDD	4GB
Input Device	Wacom tablet drawing stale II (pen-tablet type device)

## 8.3 色選択ユーザインタフェースの実装

第5章において述べた絵の具モデルの内部パラメータセットはスペクトル毎の光の吸収率、散乱率であった。しかしそれらを直接入力して、望んだ色を得ることは容易ではない。そこで、第6章では色との対応がとれた、より少ない数のパラメータセットを定義し、それらから、絵の具の内部パラメータセットへ変換するアルゴリズムを述べた。

本節では、第6章で定義したパラメータセットに対する GUI の一例を示す。図 8.1 が色を選択するダイアログである。X Window System[91] 上に GTK+(Gimp Tool Kit)[92] ライブラリを用いて作成した。インタフェースのデザインとして従来のアプリケーションソフトウェアとそれほど違いがないことがわかる。

色の決定のための三つの座標を入力する部分は従来のツールでの色決定手段と変わりが無い。新たに加わっているのは、先鋭度 (Sharpness) に関するスライダだけである。先鋭度の値域は全ての色に対して、0~1 であるので スライダの移動に対しての先鋭度の対応付けは容易である。

ユーザは絵の具の選択に従来と変わらない操作により色を決定でき、必要とあらば先鋭度を動かすという操作だけで、色の混色時の強さに関係するスペクトル反射率の曲線形状を変化させることができる。

しかも、スペクトル反射率の曲線形状や、光の吸収率や散乱率に関する知識は GUI の操作には一切必要ないように作られているためユーザにとって馴染みやすいと考えられる。

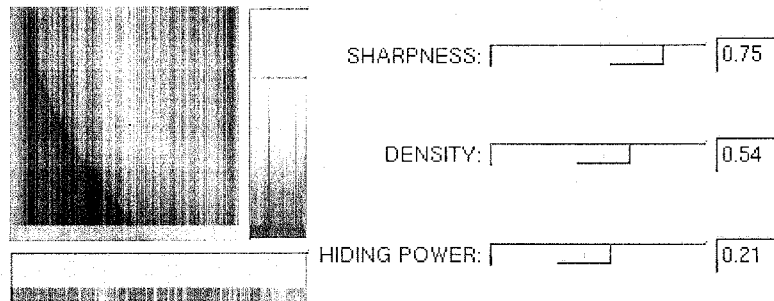


図 8.1 色選択 GUI の実装例

## 8.4 ペイントツール全体の機能

本節では、第7章で述べた筆モデルを使って自由度の高いストローク入力を可能とするペイントツールの実装例を示す。

第7章で述べた筆モデルの制御には、筆の柄の  $x$ 、 $y$  座標の他に高さ  $z$ 、傾きが必要であった。

これらの入力を満たすペンタブレット型の入力装置として X Window System の X Input Extension[93] 及び GTK+[92] でサポートされている WACOM 社のタブレットを選択した。タブレットからの入力される信号は表 8.1 のとおりである。

表 8.1 本ペイントツールで用いるタブレットからの信号一覧

データ	デバイスの意味	筆の柄のデータとしての意味
x	横方向の座標	画材上の横方向の位置
y	縦方向の座標	画材上の縦方向の位置
p	圧力	筆の画材までの高さ
xtilt	横方向の傾き	筆の柄の画材の横方向への傾き
ytilt	縦方向の傾き	筆の柄の画材の縦方向への傾き

ペイントツールに実装された機能の各項目を挙げる。

- 描画

ペンの操作により、仮想的な筆の毛先と画材との接地面に描画を行う。

- 筆へ絵の具をつける  
絵の具を筆に含ませる機能である。
- 絵の具の変更  
筆に付ける絵の具を変更する機能である。
- 筆の濃度変化  
絵の具の濃度を変化させる。
- 筆の洗浄  
異なった絵の具を洗わずに筆につけると混ざってしまう。そこで、この機能で筆を洗う。
- 乾燥処理  
絵の具が乾いた状態への塗り重ねと、乾いていない状態での塗り重ねでは混色結果が異なる。この機能で画材上に描かれた絵の具を乾燥させる。
- 現状描画状態を一時保存  
計算機上での描画の最も大きな利点としてやり直しが可能な点がある。この機能で以前の状態へ戻して描画をやり直すために描画状態を一時的に保存することができる。描画状態とは、仮想的な画材上の描かれたすべての絵の具の情報を差す。
- 一時保存された描画状態の呼び出し  
一時保存されていた、描画状態を復帰させて、現在の描画状態とする。

画面上には図 8.4 のように表示される。

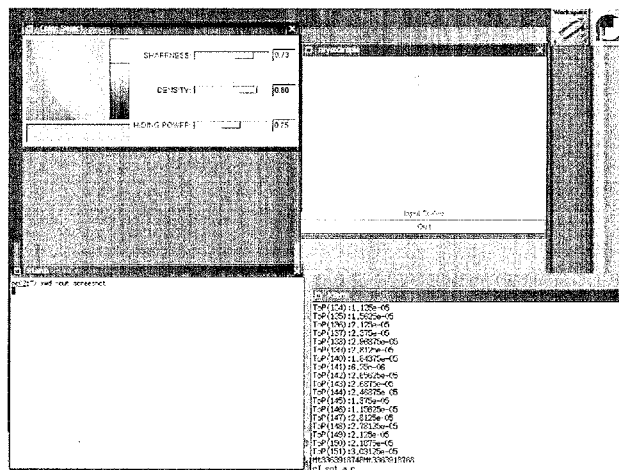


図 8.2 ペイントツール実装例のスクリーンショット

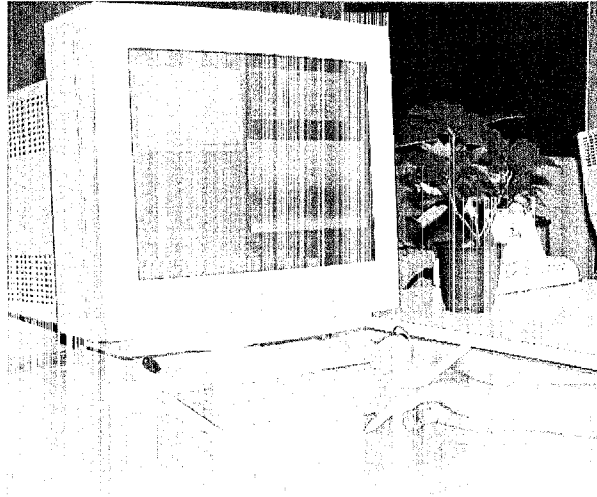


図 8.3 ユーザの使用風景

描画作業はコンピュータディスプレイを見ながら、タブレット上のペンデバイスを操作することにより行う。図 8.3 に作業風景を示す。

## 8.5 ユーザによる描画例

本節では 8.4 節 で述べたペイントツールを用いて実際に描画した例を示す。

表現力の高いストローク描画が可能なることから、文字の入力を行った例が、図 8.4、図 8.5、図 8.6 である。

平仮名の例として図 8.4 に “か” と “な” を漢字の例として図 8.5 に “元” と “旦” を示している。図 8.6 はハングル文字である。文字では特に重要になる筆のはらいや、止めが表現可能であることが分る。また、描画のスピードに応じての濃淡もできていることが分る。

次に絵を描いた例を図 8.7、図 8.8 に示す。

結果の描画例では、描画における個々のストロークの質が高いことから、少ないストロークでも十分に画を構成できることが分る。特に、図 8.7 では 4 ストロークで描かれているが、“うなぎ” であることが分るような曲線を描くことが出来ている。また、図 8.8 の花びらのような、より複雑な描画表現も可能であることが分る。





図 8.4 ユーザによる平仮名の描画例



図 8.5 ユーザによる漢字の描画例

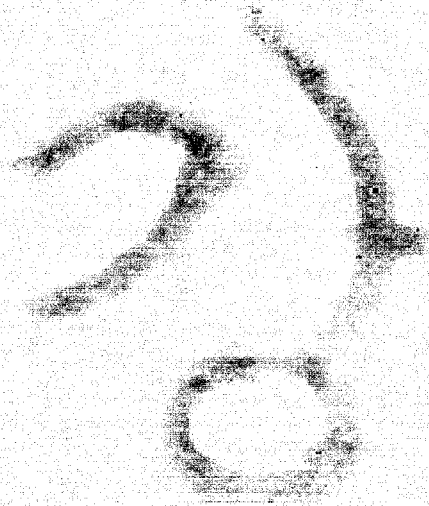


図 8.6 ユーザによるハンゲルの描画例



図 8.7 ユーザによる絵の描画例 1



図 8.8 ユーザによる絵の描画例 2

## 8.6 むすび

本章では、第5章から第7章までの応用として、絵の具モデルと筆モデルを用いて対話的描画作業を行えるプログラムを構築し、本モデルの描画の自由度の高さを示した。

第6章で述べたアルゴリズムに対して実際にユーザインタフェースを構築した。この実装から、色からスペクトル反射率曲線への変換アルゴリズムにより、ユーザは絵の具の内部パラメータの意味や発色アルゴリズムについて一切の知識を要求されることなく、また従来のペイントツールで慣れた操作法と同じような操作によって、絵の具を選択可能な GUI をデザイン可能であることが示された。

ペンタブレットを入力デバイスとして第7章の筆モデルを制御し、実際にユーザに描画を行ってもらい、表現力の高いストロークを対話的操作のを用いて実際に描画可能であることを示した。

描画結果からは、従来のペイントツールでは困難であった個々のストロークの活きを感じることが出来る。

対話的操作において現状の計算機能力は若干不足している。しかし、計算機能力はより高速化がなされると予想されるので、数年の内に十分必要な処理速度は得られると考えている。

今後は、現在考慮されていない滲みについても検討し、より高度な表現が可能となるように改良を加えることが検討課題である。

# 第 9 章

## 結論

### 9.1 まえがき

本章では、本論文の全体の結論を 9.2 節 で、今後の課題について 9.3 節 で、また応用面での展望について 9.4 節 述べる。

### 9.2 結論

本論文では、手描き風の描画を行うコンピュータグラフィックスに関する研究について述べてきた。

写真を入力として手描き風画像を再構成する手法では、提案を行った手描き風描画のための画像処理法、自動描画のアルゴリズムについて述べた。種々のアルゴリズムを使って、様々な雰囲気画像を一枚の入力画像から作成可能であることを示すことができた。使用した画像処理法は局所方向性や輪郭の検出を行うものであり、3次元形状推定や物体認識を行うものではないが、入力画像の局所情報のみから計算機により自動的に描画することに成功した。

次に、手描き風な CG の更なる画像の質の向上には、鮮やかな絵の具の混色計算が可能な絵の具モデルが必要不可欠であるとの認識から行った、Kubelka-Munk の理論を用いた絵の具モデルの研究について述べた。塗られた絵の具の層内部の光の挙動をモデル化した Kubelka-Munk の理論を絵の具に適応するため、絵の具の混色の再現に必要なスペクトルのサンプリング数を検討するなどの具体的な実装への問題解決を行い、従来にない、混色時に

濁りの発色を抑えた絵の具モデルを提案できた。

さらに、絵の具の特性を端的に表すパラメータを作り、提案した絵の具モデルの複雑な発色アルゴリズムや、内部パラメータを隠蔽して、描画アルゴリズムからや、ユーザから直接要求される色・特性の絵の具を容易に選択できるような手法について述べた。この手法は発色が素晴らしいモデルであっても望んだ色の絵の具が無いのでは実用上不便であるため、提案した絵の具モデルに必要不可欠である。それと同時に、色を一定に保ったままスペクトル反射曲線を望んだように変化させることが可能になったことにより現実世界の絵の具を越えた自由な絵の具の調合を可能としている。

提案絵の具モデルは塗り重ねた厚みに応じた発色が可能なため、従来に無い繊細なストロークの描き方の違いを表すことを可能としている。そこで、3次元形状を持つ筆のモデルを仮想空間に作り、画材と筆の毛先との接地面や画材へ塗られる絵の具の量を筆の柄の動きにより動的に変化させることを可能とした。これにより、従来の円盤の軌跡をストロークとしてたシステムと比較して明らかに表現力の高いストロークを描くことに成功した。

最後に、絵の具モデルと筆モデルを用いた応用システムの一例として、対話的なペイントツールの作成について述べている。簡素な機能のみを備え、ストローク入力にやや不便な点もあるが、ユーザの技量に応じて様々なストロークが入力可能であり、従来に無い描画の可能性を十分に示した。

## 9.3 今後の課題

本論文では、手描き風なCGを作成するための手法について述べてきたが、まだ課題が残されている。個々についての具体的な課題を挙げる。

### [1] 画像処理における課題

画像処理法として、輪郭線、方向性検出法の更なる改良が必要である。注目する領域の広さに応じて、人間は優先する輪郭線や方向性が変化するはずである。この点に対応可能な画像処理法が必要となる。また、単独の一本の線は大きな画像的特徴を持っているが、縞状になった線は相対的には線としての画像的特徴は少ない。このような感覚を検出可能な画像処理法が必要であると考えている。

また、画像処理法によって得られた局所的な画像特徴をユーザが変更して描画のさせ方を一部操作できるような、画像処理結果の格納方法も検討課題である。



## [2] 自動描画手法における課題

描画による表現形式は多種多様である。従ってより多くの雰囲気の画像を作成可能なように新たな描画アルゴリズムの検討が挙げられる。提案した絵の具モデルを用いた自動描画法についても実現課題である。

描画アルゴリズムの具体的な改善としては、多層描画アルゴリズムや、筆の描きはじめ、止めの位置の決定法がある。提案手法、第2章で挙げた従来手法共にストロークの描画における戦略の高度化がより細部まで繊細な描画結果を得るためには必要であると考えている。

自動描画法は放っておける便利さの反面、描画の際に一部分の修正が出来ない。多少のパラメータ変更は可能であるが、利用者の個性を描画結果に出すことが難しい。そこで、自動描画法に一部対話性を加えられるようにすることも実用上は重要であると言える。

また、ある画像を見ながら実際にユーザにペイントツールを使って絵を描いてもらい、次に画像を画像処理により解析して、ユーザがどのような画像の局所情報を基に描画を行っているのか、描画におけるストロークの描き方と画像の局所情報の対応関係を調べた上で、自動描画法を作成することも考えられる。

## [3] 絵の具、筆、画材モデルにおける課題

絵の具のモデルに関しては一つの隠ぺい力をもとに吸収率と散乱率を決定する手法を述べた。隠ぺい力は白地と黒地に対して測定可能であり、二つの隠ぺい力を用いることによりフェルトペンのインクのような吸収率の高い絵の具のパラメータも設定可能とすることも検討課題である。

筆モデルに関してはかすれを再現できるように筆の中の絵の具の移動モデルを検討する必要がある。また、対話的ペイントツールの実装において筆の毛先の挙動を決定する物理シミュレーション結果が不安定になることが稀にある。そこで、シミュレーションをさらに安定化する必要がある。

画材に関しては滲みを再現できるような手法を検討することにより表現の多様性が期待できる。

また、現実世界での描画ではマスキングやスクラッチなど多くの手法がある。それらも計算機上で実現可能となるよう多くの素材の疑似的なモデル化が課題である。

## [4] 応用としてのペイントツールにおける課題

柄の高さの移動に対する毛先の寝方が急であるように感じられることから現状では慣れが必要である。この問題の要因として、入力デバイス、ハードウェアの処理速度、アルゴリズム、筆モデルのパラメータ、があり、解決してより使いやすくする必要がある。また、筆に

含まれている絵の具の残量表現や塗った絵の具が乾き具合をどう表現するか、など状態表示を分りやすく行う方法が検討課題として挙げられる。

#### [5] その他の課題

本論文では3次元オブジェクトを入力として手描き風にする検討を行わなかったが、3次元物体には視点移動に対しても容易に空間を再構成可能な点や、光源のレイアウトの変更容易性など利点がある。3次元オブジェクトの入力方法の検討 [90] を含めて考えて行きたい。

## 9.4 研究展望

手描き風の画像の予想される応用分野について述べる。

第1章でも簡単に述べたが手描きの絵は写真と比較して独特な雰囲気を持ち、それゆえ様々なところで利用されている。観光はがき、挿絵、パッケージイラスト等である。

本論文の成果の応用分野としてのそのような利用分野が予想されるが、ここで一つ特に触れておきたい分野として、アニメーションの世界がある。

現在、日本の文化輸出産業としても脚光を浴びているアニメーション業界でもデジタル化が進んでいる。現状での制作作業では、実際に描画した絵をスキャナで取り込んだ後に合成、フィルタ処理などでデジタル技術を使っているが、今後は描画段階からのデジタル作業に変化していく可能性がある [101][102][103]。カラーマッチングの問題からすべての素材をはじめから統一的にデジタルで扱う方が都合が良いためや、絵の保管問題があるからである。

現在までに培ってきた技法をデジタルで置き換えるための研究や、画風の統一化の検討、CGで得意とする3次元物体と2次元的描画の世界の融合などなど、本論文で述べた手法が即座に利用可能という訳ではないが、産業としても文化としても重要な分野であると考えて、貢献できるよう、今後研究を進めて行きたい。

## 9.5 むすび

本章では本論文の成果、結論を述べ、今後の課題項目を挙げ、そして研究展望について述べた。

## 謝 辞

本研究を続けるにあたり、大変多くの方にお世話になりました。

長年、指導をして頂いた中嶋正之教授、色々と面倒をおかけした高橋裕樹助手、専門外のことながら相談に乗って頂いた亀井宏行助教授には大変感謝しております。ありがとうございました。

厳しい指摘や励ましを頂き、卒業後も気に止めて頂いた先輩の柴広有さん、水谷雅美さん、麻生隆一さん、そして学部ころの同期の染谷信彦君、山田哲也君の卒業後の企業での活躍も大変励みになりました。

退職されましたが長年中嶋研の事務員を勤められました米田佳子さんには、雑談で気をまぎらわして頂きました。

先輩の Liu Yu さん、笠尾敦司さんには研究の仕方を体現して示して下さい大変参考になりました。ありがとうございました。結果についての有益な感想を述べて頂いた永江孝規さんも、ありがとうございました。

いつも陽気な博士課程同期の Lee Peisuci さんには明るさを分けて頂きました。ありがとうございました。

計算機、ネットワーク環境の安定運用を心がけて下さいました、歴代の中嶋研のシステム管理者の小西宏君、河口啓一君、本田卓也君、Kevin M. Squire 君にもお礼を述べさせていただきます。ありがとうございました。

また、愚痴、馬鹿話に付き合ってくれた、後輩の山田類君、長谷川公位君、他多くの方々、プログラミングの手伝いをして下さった鶴大介君、ペイントツールのテストユーザとして画像作成、意見感想を述べて下さった、伊原申容君、張英夏さん、ありがとうございました。

居酒屋で、研究の話に付き合ってくれたあんな人やこんな人にも感謝致します。

最後に、長年の勉強に理解をして頂き、また健康面にも気を使って頂いた家の両親に感謝致します。博士論文をまとめ上げることが出来ました。ありがとうございました。

## 参考文献

- [1] James D.Foley, Andries van Dam, Steven K. Feiner, John F. Hughes : “Computer Graphics 2nd edition ”,Addison-Wesley,(1990)
- [2] Alan Watt, Mark Watt : “ Advanced Animation and Rendering Techniques ”, ACM PRESS ,NewYork (1992)
- [3] Paul S. Heckbert : “Graphics Gems ”,AP professional,pp.383-414 (1994)
- [4] Craig Kolb, Don Mitchell, Pat Hanrahan : “ A Realistic Camera Model for Computer Graphics ”, SIGGRAPH '95 proceedings ,pp.317-324 (1995)
- [5] Jhon Lansdown, Simon Schofield :“ Expressive Rendering: A Review of Nonphotorealistic Techniques”,IEEE CG and Appli, Vol.15, No.03,pp.29-37 (1995)
- [6] 大野義夫: “フォトリアルでないCG画像表現の手法について”, 情報処理学会研究報告 95-CG-76,pp.1-7 (1995)
- [7] 石原 亘:“まんがにおける具象物の半記号化された表示 - 非写真的写実レンダリングを論じるために - ”, 情報処理学会研究報告 95-CG-76,pp.9-16 (1995)
- [8] P. Haeblerli : “Paint By Numbers : Abstract Image Representations”, Computer Graphics , Vol.24,No.4,pp.207-214 (1990)
- [9] Takafumi Saito,Tokiichiro Takahashi: “Comprehensible Rendering of 3-D Shapes” ,Computer Graphics,Vol.24,No.4, pp.197-206 (1990)
- [10] 齋藤 隆文: “Comprehensible Rendering — これからの課題” 情報処理学会研究報告 95-CG-76,pp.89-94 (1995)
- [11] 金子 満, 中嶋 正之 : “ 次世代アニメーションシステムに関する研究 第1報- 3次元CG画像の2次元化アルゴリズム - ”, 情報処理学会研究報告 95-CG-69,pp.57-64 (1994)

- [12] 金子 満, 中嶋 正之: “次世代アニメーションシステムに関する研究 - ノンフォトリアリ  
スティックアニメーションの生成 -”, 情報処理学会研究報告 95-CG-76, pp.23-30 (1995)
- [13] Georges Winkenbach, David H. Salesin: “ Computer -Generated Pen-and-Ink Illustration ” , SIGGRAPH 94 proceedings ,pp.91-100 (1994)
- [14] Georges Winkenbach, David H. Salesin: “ Rendering Parametric Surfaces in Pen and Ink”, SIGGRAPH 96 proceedings, pp.469-476 (1996)
- [15] Richard Coutts, Donald P. Greenberg: “Rendering with Streamlines”, SIGGRAPH 97 visual proceedings, p.188 (1997)
- [16] 関谷 英明, 島田 繁広, 近藤 邦雄, 佐藤 尚, 島田 静雄: “モノクロレンダリングのための  
テクスチャ描画手法”, 情報処理学会研究報告 95-CG-76, pp.81-87 (1995)
- [17] 望月 義典, 近藤 邦雄, 佐藤 尚, 島田 静雄: “形状理解を容易にする特徴強調画像の生成”,  
情報処理学会研究報告 95-CG-76, pp.73-80 (1995)
- [18] 宮澤 貴之, 望月 義典, 近藤 邦雄, 佐藤 尚, 島田 静雄: “3次元形状を誇張するための投影  
変換手法”, 1996年テレビジョン学会映像メディア部門冬季大会講演予稿集, p.67, テレ  
ビジョン学会 (1996)
- [19] 望月 義典, 近藤 邦雄, 宮澤 貴之, : “誇張表現のための投影変換” , 第14回 NICO-  
GRAPH/MULTIMEDIA 論文コンテスト 論文集, pp.66-72, マルチメディアコンテ  
ンツ新興協会 (1998)
- [20] 張 青, 高橋 淳也, 村岡 一信, 千葉 則茂: “樹木の水墨画調レンダリング”, 情報処理学会  
研究報告 95-CG-76, pp.45-50 (1995)
- [21] 高橋 淳也, 張 青, 村岡 一信, 千葉 則茂: “セルオートマトンによる墨の振る舞いのシミュ  
レーションとその樹木の水墨画レンダリング” , NICOGRAPH 論文集 1996 pp.95-101  
(1996)
- [22] Barbara J. Meier: “Painterly Rendering for Animation”, SIGGRAPH 96 proceedings,  
pp.477-484 (1996)
- [23] Peter Lee: “Linear Colour Contouring for Fine Art Printmaking”, SIGGRAPH 97  
visual proceedings, p.185 (1997)

- [24] Lee Merkosianm, Michael A. Kowalski, Samuel J. Trychin, Lubomir D. Bourdev, Daniel Goldstein, John F. Hughes: “Real-Time Nonphotorealistic Rendering”, SIGGRAPH 97 proceedings, pp.415–420 (1997)
- [25] Amy Gooch, Bruce Gooch, Peter Shirley, Elaine Cohen: “A Non-Photorealistic Lighting Model For Automatic Technical Illustration”, SIGGRAPH 98 proceedings, pp.447–452 (1998)
- [26] T.Cockshott et al.: “Modeling the Texture of Paint”, Computer Graphics Forum, Eurographics 92 Proc, Vol.11, No.3, pp.C217–C226 (1992)
- [27] Steve Strassmann: “Hairy Brushes”, Computer Graphics, Vol.20, No.4, pp.225–232 (1986)
- [28] Qinglian Guo: “毛筆フォントと CG 描画技術の統合”, 情報処理学会研究報告 95-CG-76, pp.51–58 (1995)
- [29] Michael P.Salisbury, Sean E.Anderson, Ronen Barzel, David H.Salesin: “Interactive Pen-and-Ink Illustration”, SIGGRAPH94 proceedings, pp.101–108 (1994)
- [30] 高木 佐恵子, 藤代 一成: “水溶性色えんぴつ画における加水効果の CG モデリング”, 情報処理学会研究報告 96-CG-80, pp. 25-32, (1996)
- [31] 高木 佐恵子, 藤代 一成: “水溶性色えんぴつ画の 3D モデリング- 紙の微細構造を反映した顔料付着操作 -”, 情報処理学会 Visual Computing / グラフィクスと CAD 合同シンポジウム '97 予稿集, pp. 117-122, (1997)
- [32] Saeko Takagi, Issei Fujishiro: “Microscopic Structural Modeling of Colored Pencil Drawings”, SIGGRAPH 97 Visual Proceedings, p.187, (1997)
- [33] Michael P.Salisbury, Michael T. Wong, John F. Hughes, David H.Salesin: “Orientable Texture for Image-Based Pen-and-Ink Illustration”, SIGGRAPH 97 proceedings pp. 401–406 (1997)
- [34] Siu Chi Hsu,Irene H.H.Lee: “Drawing and Animation Using Skeletal Strokes”, SIGGRAPH 94 proceedings, pp.109–118 (1994)
- [35] Cassidy j. Curtis, Sean E. Anderson, Joshua E. Seims, Kurt W. Fleischer, David H. Salesin: “Computer-Generated Watercolor”, SIGGRAPH97, proceedings pp. 421–430 (1997)

- [36] Wagner Toledo Corrêa, Robert J. Jensen, Craig E. Thayer, Adam Finkelstein: “Texture Mapping for Cel Animation”, SIGGRAPH proceedings pp. 435–446 (1998)
- [37] “Photoshop Web page” <http://www.adobe.com/prodindex/photoshop/main.html>
- [38] “Painter Web page” <http://www.metacreations.com/products>
- [39] “The Gimp Web page” <http://www.gimp.org/>
- [40] Michael T. Wong, Douglas E. Zongker, David H. Salesin : “Computer-Generated Floral Ornament”, SIGGRAPH '98 proceedings, pp. 423–434 (1998)
- [41] Eric Guaglione, Marty Altman, Kathy Barshatzky, Rob Bekuhrs, Barry Cook, Mary Ann Pigora, Tony Plett, Ric Shuiter: “The Art of Disney’s Mulan”, SIGGRAPH '98 course note #39 (1998)
- [42] M.F.Barnsley : “Fractals Everywhere”, Academic Press (1988)
- [43] 安居院 猛, 都 鉉淑, 長橋 宏: “写真からの点描画の自動生成”, 1993 年信学論春季全大 D-619 (1993)
- [44] T.Agui, H. Do, H. Nagahashi , M. Kaji : “ Painting-like Image Generation”, 日本印刷学会誌, Vol. 31, No. 1, pp.40–48 (1994)
- [45] 染谷 信彦, 宮原 誠 : “Wavelet 変換を用いた画像 modification:手書き風表現”, 電気関係学会北陸支部連合大会講演論文集, p.423 (1995)
- [46] 染谷 信彦, 宮原 誠: “写真画像の油絵風 modification — 油絵知識(筆使い)の考慮”, 電子情報通信学会技術研究報告 [回路とシステム], Vol.95, No.556, pp.1-6 (1996)
- [47] 染谷 信彦: “写真の手描き風画像処理”, 1998 年信学論春季全大, D-11-165, p.165 (1998)
- [48] 大河内 俊雄, 水野 慎士, 岡田 稔, 鳥脇 純一郎: “仮想版木自動生成による版画画像合成の一手法”, 情報処理学会研究報告 97-CG-86, pp.1-4 (1997)
- [49] 水野 慎士, 岡田 稔, 横井 茂樹, 鳥脇 純一郎: “CSG を用いた仮想彫刻”, 情報処理学会研究報告 95-CG-73, pp.49–56 (1995)
- [50] 笠尾敦司, 中嶋正之: “シナージスティックイメージクリエイター —描画プロセスを重視した絵画作成システム—”, 電子情報通信学会, J81-D-II, No.4, pp.671-680 (1998)

- [51] 笠尾敦司, 中嶋正之: “テクスチャーの方向要素を利用した K-平均アルゴリズムによる自然な領域分割方法”, 電子情報通信学, J81-D-II, No. 7, pp.1528-1535 (1998)
- [52] Peter Litwinowicz: “Processing Images and Video for An Impressionist Effect”, SIGGRAPH 97 proceedings, pp.407-414 (1997)
- [53] 白石 路雄, 山口 泰: “実存する油彩画の特徴分析による油彩画風画像の生成手法”, 情報処理学会研究報告 98-CG-91, pp.1-6 (1998)
- [54] Greg Turk, David Banks: “Image-Guided Streamline Placement”, SIGGRAPH 96 proceedings, pp. 453-460 (1996)
- [55] Mike Salisbury, Corin Anderson, Dani Lischinski, David H. Salesin: “Scale-Dependent Reproduction of Pen-and-Ink Illustrations”, SIGGRAPH 96 proceedings, pp.461-468 (1996)
- [56] Aaron Hertzmann: “Painterly Rendering with Curved Brush Strokes of Multiple Sizes”, SIGGRAPH 98 proceedings, pp.453-460 (1998)
- [57] 興水 大和, 村上 和人, 中山 晶, 福村 晃: “似顔絵生成研究の現状と PICASSO システムの諸課題”, 画像の認識・理解シンポジウム 論文集 II pp.159-166 (1996)
- [58] M.kass, A.Witkin: “Analyzing Oriented Patterns”, Computer Vision, Graphics, and Image Processing, Vol. 37, pp.362-385, (1987)
- [59] A.Ravishankar Rao, Brain G.Schunck: “Computing Oriented Texture Fields”, CVGIP. Graphical models and image processing, Vol 53, No2, March, pp.157-185 (1991)
- [60] Freeman W.T., Adelson E.H.: “The design and use of steerable filters”, IEEE Trans. Pattern Anal & Machine Intell., Vol. 13, pp.891-906 (1991)
- [61] 志沢 雅彦, 磯 俊樹: “局所微分フィルタを用いた多重スケール・多重方向場の表現と検出”, 電子情報通信学会論文誌 A, Vol. J78-A, No.2, pp.261-269 (1995)
- [62] C.Yao, J.Rokne: “Fat curves”, The Computer Forum, Vol 10, pp.237-248 (1991)
- [63] 日本色彩学会 編: “新編 色彩科学ハンドブック”, 東京大学出版会, 第 20 章 pp.683-743 (1980)
- [64] 長江 明, 杉山 芳夫, 桑原 利秀, 福田 保: “顔料の測色 (第 1 報) - 顔料の測色値とピヒクルの関係 -”, 色材協会誌 Vol.33, No. 7, pp.297-301 (1960)



- [65] 池田 光男:“色彩工学の基礎”, 朝倉書店,(1980)
- [66] 中原 勝儼: “化学の話シリーズ 9 色の科学”, 培風館, p.123,(1985)
- [67] 山口 昌一郎, 川上 元郎: “色彩工学入門”, オーム社, p.21,(1963)
- [68] Von Paul Kubelka, Franz Munk: “Ein Beitrag zur Optik der Farbanstriche”,  
Zeitschrift für technische Physik, pp.593–601,(1931)
- [69] Paul Kubelka: “New Contributions to the Optics of Intensely Light-Scattering Materials. Part I”,  
Journal of the optical society of america, Vol. 38, No. 5, pp.448–457,(1948)
- [70] 日本色彩学会編: “色彩科学辞典”, 朝倉書店,p.87 (1991)
- [71] Roy A. Hall, Donald P. Greenberg: “A Testbed for Realistic Image Synthesis”,  
IEEE CG&A, November, 1983, pp.10-19.
- [72] Smith B., Spiekermann C., Sember R.: “Numerical-Methods for Colorimetric Calculations - A Comparison of Integration Methods”,  
Color Research and application,A Wiley-Interscience Publication, Vol.17, No. 6, pp.384–393 (1992)
- [73] Smith B., Spiekermann C., Sember R.: “Numerical-Methods for Colorimetric Calculations - Sampling Density Requirements”,  
Color Research and application,A Wiley-Interscience Publication, Vol.17, No. 6, pp.394–401 (1992)
- [74] 鈴鹿 正和: “最近の染料用コンピュータ・カラーマッチングの動向” 色材協会誌, Vol.65,  
No. 11, pp.700–710 (1992)
- [75] 色材協会編: “色材工学ハンドブック”, 朝倉書店, pp.206–209 (1989)
- [76] 日本顔料技術協会編: “改訂新版 顔料便覧”, 誠文堂新光社, pp. 84–86 (1989)
- [77] 中村 靖: “色合わせ評価の計算機シミュレーションによるカラーマッチング手法の解析”,  
照明学会雑誌, Vol.63, No.7, pp.353–360, (1979)
- [78] “JIS Z 8722-1971 2 度視野 XYZ 系による物体色の測定方法”, 日本工業規格, (1971)
- [79] “Colour Management in Multimedia Systems Default RGB Colour Space - sRGB part2.1”,  
International Electrotechnical Commission,(1997)
- [80] チャールズ K. チュウイ, “数理科学セミナー ウェブレット入門”, 東京電気大学出版局, pp.59–72 (1993)

- [81] R.J.Watt, 乾 敏郎訳: “視覚情報処理モデル入門 - 計算論的アプローチ -”, サイエンス社, 東京, pp.32-69 (1989)
- [82] 乾 敏郎: “視覚情報処理の基礎”, サイエンス社, 東京, pp.1-102 (1990)
- [83] 溝江 有里子, 乾 敏郎: “滑らかな輝度変化に対するエッジ位置の知覚”, テレビ学技法, Vol.19, No.59 ヒューマンインターフェース pp.19-24 (1995)
- [84] Marr, D., Hildreth, E. : “ Theory of edge detection”, Proceeding of Royal Society of London, B, 207, pp.187-217 (1980)
- [85] 内山 裕司, 石原 利員, 辻 紘良, 大関 徹, 増田 豊, 杉浦 護 : “塗装の反射率モデルによる色・質感変更”, NICOGRAPH 論文集 1996 pp.189-199 (1996)
- [86] 武藤 一, 清水 英之, 牧野 光則, 大石 進一: “CGによる多層膜の干渉光の表現”, NICOGRAPH 論文集 1996 pp.178-188 (1996)
- [87] 小出 昭一郎: “力学と微積分 物理数学 OnePoint7”, 共立出版, pp.57-69 (1993)
- [88] 水上 孝一, 市山 寿男, 野田 松太郎, 市原 英生, 渡辺 敏三: “プログラミング入門シリーズ7 コンピュータによる数値計算”, 朝倉書店, pp.88-96 (1985)
- [89] 山本 和夫: “非線形機械力学シミュレーション”, 日刊工業新聞社, pp.1-170 (1996)
- [90] 松田 浩一, 近藤邦雄: “3次元形状入力のためのスケッチインタプリタシステム”, 第14回 NICOGRAPH/MULTIMEDIA 論文コンテスト 論文集, pp.17-26, マルチメディアコンテンツ新興協会 (1998)
- [91] 安居院 猛, 永江 孝規: “X アプリケーション・プログラミング 1 Xlib 編”, 新紀元社 (1992)
- [92] Ian Main, Tony Gale: “GTK tutorial”, <http://www.gtk.org/tutorial/>, (1998)
- [93] Mark Patrick, George Sachs: “X Input Device Extension Library X Consortium Standard X Version 11, Release 6.3”, X Consortium, (1992)
- [94] “Wacom tablet Web page”, <http://tablet.wacom.co.jp/>
- [95] Jonas Gomes, Luiz Velho: “Image Processing for Computer Graphics”, pp.88-103 Springer, (1997)
- [96] 佐藤宏介, 井口征士: “イメージの画像解析に基づく絵画のデータベース化と自動作画”, 人工知能学会第21回ヒューマンインタフェースと認知モデル研究会, pp. 1-8, (1994)

- [97] 内川 恵二: “カラーマネジメントと色知覚”, 1996年テレビジョン学会映像メディア部門冬季大会講演予稿集, pp.21-26, テレビジョン学会 (1996)
- [98] 笠尾敦司, 中嶋正之: ”輪郭画像と色面画像を用いた印象解析手法”, 映像情報メディア学会, Vol.52, No.4, pp.535-541 (1998)
- [99] 石原 亘: “SIGGRAPH '94におけるノンフォトリアリズムの研究の動向”, 情報処理学会研究報告 95-CG-76, pp.9-16 (1995)
- [100] Oliver Deussen, Pat Hanrahan, Bernd Lintermann, Radomír Měch, Matt Pharr, Przemyslaw Prusinkiewicz: “Realistic Modeling and rendering of plant ecosystem”, SIGGRAPH '98 pp.275-286 (1998)
- [101] 中嶋正之: “アニメとコンピュータ”, 情報処理, Vol.39, No.7, pp.613-620 情報処理学会 (1998)
- [102] (財)新映像産業推進センター 編: “デジタルアニメーションのビジネス展開-アニメーション産業への期待と課題-”, ニューメディア (1998)
- [103] スタジオジブリ作品: “「もののけ姫」はこうして生まれた。”, 二馬力, スタジオジブリ (1997)