

平成 13 年度
学士学位論文

連続的な色表現を可能にする
視覚復号型秘密分散法の一手法

A Visual Secret Sharing Scheme
for Continuous Color Images

1020266 小川 智恵

指導教員 菊池 豊

平成 14 年 2 月 8 日

高知工科大学 情報システム工学科

要 旨

連続的な色表現を可能にする 視覚復号型秘密分散法の一手法

小川 智恵

画像を暗号化、復号化する理論に視覚復号化秘密分散法 (Visual Secret Sharing Scheme) がある。VSSS は、暗号化の際は、いくつかのシェア画像に分散し、復号化の際は、そのシェア画像を重ねて、人間の目で直接復号化する。本研究の目的は、忠実に原画像を復元する VSSS の実現である。目標として、離散的であっても任意の粒度で色表現を可能にし、復元画像をできるだけ明るくすることにした。本研究では、3つの手法を提案し、その手法の実装と評価を行った。その結果、本手法で任意の色表現が可能となった。そして、復元した画像の明るさに対しては、シェア画像の枚数を多くすればより原画像に近付くということが判明した。

キーワード 視覚復号型秘密分散法、カラー画像、シェア画像

Abstract

A Visual Secret Sharing Scheme for Continuous Color Images

Tomoe Ogawa

The Visual secret Sharing scheme (VSSS) is a technology of encrypting a secret black-white image into shares and decrypting from the shares without using any cryptographic computation. This paper proposes an extension of VSSS for color images. One of the goals makes color expression possible in the optional fineness even it uses discrete color rank. The other makes decrypted images lighten. This paper describes that the implementation and the evaluation of three methods of the proposal. The paper shows that these methods can express arbitrary colors. In theoretically speaking, you can have the same colors of decrypted images as the original ones in case of using infinite shares.

key words Visual Secret Sharing Scheme, Color Images, Shares

目次

第 1 章	はじめに	1
第 2 章	視覚復号型秘密分散法 (VSSS)	2
2.1	VSSS について	2
2.2	白黒画像の VSSS	3
2.2.1	基本的な考え方	3
2.2.2	白黒画像の VSSS の例	3
2.3	カラー画像の VSSS	5
2.3.1	基本的な考え方	5
2.3.2	カラー画像の VSSS 例	5
2.4	既存の手法の問題点	7
第 3 章	目的・目標	9
第 4 章	色分散復元手法	
	扇面積分割手法	10
4.1	手法	10
4.1.1	基本原理	10
4.1.2	色分散復元手法	11
4.1.3	扇面積分割手法	11
4.1.4	RGB から CMY への変換	12
4.1.5	円内の扇形の位置決め (色分散復元手法)	13
4.1.6	円内の扇形の位置決め (扇面積分割手法)	14
4.1.7	シェア画素への分散方法 (色分散復元手法)	14
4.1.8	シェア画素への分散方法 (扇面積分割手法)	16

目次

4.2	実装	17
4.2.1	実装環境	18
4.2.2	実装の処理単位	18
4.2.3	シェア画素の作成	18
第 5 章	扇面積可変手法	21
5.1	手法	21
5.1.1	基本原理	21
5.1.2	円内の扇形の位置決め (扇面積可変手法)	22
5.1.3	シェア画素への分散方法 (扇面積可変手法)	22
5.2	実装	24
5.2.1	シェア画素の作成	24
第 6 章	考察	26
6.1	復元画像の明るさについて	26
6.1.1	色の分散復元手法	26
6.1.2	扇面積分割手法	29
6.1.3	扇面積可変手法	32
6.1.4	明るさのまとめ	35
6.2	シェア画像の秘密性について (シェア画素が円の時)	35
6.3	実装について	36
6.3.1	表現できる色の数について	36
6.3.2	秘密性について (シェア画素が正方形の時)	37
6.4	考察のまとめ	37
第 7 章	まとめ	39
	謝辞	40

目次

参考文献	41
付録 A 扇形の重ね方によって視覚が認識する色が変わらないことの証明	42
A.1 1 ピクセルを 2 色で表現するとき	42
A.1.1 Cyan と Magenta が重ならない場合	42
A.1.2 Cyan と Magenata が重なる場合	43
A.2 1 ピクセルを 3 色で表現するとき	44
A.2.1 3 色すべてが重ならない場合	44
A.2.2 2 色が重なる場合	45
A.2.3 3 色が重なる場合	46

目次

2.1	シェア画素の重ね合わせによる画素の復元（白画素）	4
2.2	シェア画素の重ね合わせによる画素の復元（黒画素）	5
2.3	色の重ね合わせを表したハッセ図	6
4.1	処理方針	11
4.2	扇面積分割手法	12
4.3	円内の扇形の位置決め	13
4.4	色分散復元手法でのシェア画素（円）の内部	13
4.5	扇面積分割手法でのシェア画素（円）の内部	14
4.6	色分散復元手法での実装方法	18
4.7	8 × 8 内での色画素の存在範囲の位置付け	19
5.1	シェア画素について	22
5.2	扇面積可変手法でのシェア画素（円）の内部	22
6.1	復元画像の明るさを表したグラフ	36
A.1	Cyan と Magenta が重ならない場合	42
A.2	Cyan と Magenta が重なる場合	43
A.3	3色すべてが重ならない場合	45
A.4	2色が重なる場合	46
A.5	3色が重なる場合	47

表目次

6.1 考察結果	38
6.2 シェア画素の扇型と分散枚数	38

第 1 章

はじめに

現在では、情報化社会が進み、私達の周りをあらゆる情報が飛び交っている。こういった社会では、情報をいかに安全に保持していくかという点が大切になってくる。

今では、多くの情報が電子化され、コンピュータ内に保存されている。電子化された情報は、紙で私達が所有している情報に比べ、検索や編集などの作業の効率がよく、保存の際もスペースが少しで良いという利点を持っている。しかし、一方で情報を保存しているシステムに何らかの支障が起きたとき、情報がなくなったり、復元するのに膨大な労力が必要となったりする。

このようリスクを避けるために、重要な秘密情報を分割して安全に守るための方法として、秘密分散法がある。秘密分散法とは、秘密情報をグループの複数のメンバーに分散しておき、メンバーの合意によって秘密情報へのアクセスが可能となるような情報管理の方法である。この方法を用いることにより、秘密情報から、 n 個の情報（分散情報）を作成し、 n 個のうち任意の k 個を集めれば元の秘密情報を復元できる。

更に、画像を分散してスライドに印刷し、視覚によって復号できるようにした視覚復号型秘密分散法 (Visual Secret Sharing Scheme) が考えられている。この方式では、分散画像情報は、スライドのような透明な媒体に記録されており、媒体を物理的に重ね合わせることで秘密情報が見えるようになっている。本論文では、カラー画像に対応した視覚復号型秘密分散法を提案し、その復元画像の明るさと秘密性について検証を行う。

第 2 章

視覚復号型秘密分散法 (VSSS)

視覚復号型秘密分散法 (Visual Secred Sharing Scheme) の理論は、M. Naor と A. Shamir により提案されたことから始まり、現在も構成方法や特性に関する研究が行われている [1][2]。

本章では、まず VSSS の概念について説明を行う。次に、既存の白黒画像やカラー画像に対応した VSSS について説明する。また、それらの問題点についても言及する。

2.1 VSSS について

VSSS は、人間の視覚によって直接復号可能であり、かつ完全に安全な方法によって画像データ (印刷された文章、手書き文字、絵など) を暗号化する方式である。

この方式は、秘密分散法の (k, n) しきい値法を視覚によって復号できる構造に拡張した方式である。原画像が与えられたとき、 n 枚のスライドの内の任意の k 枚 (以上) のスライドを重ね合わせた場合に原画像を見ることができる。そして、 k 枚未満のスライドを重ね合わせても原画像に関する情報が全く何も見えない (あるいは、どんな方式によっても解析されない)。

原画像の各画素は、 n 枚のスライド上では、各々 m 個のシェア画素で表現される。人間の目は、分散されたシェア画像を見て画素中の色の密度の差によって原画像の情報を得ることができる。

Naor と Shamir が提案した VSSS は、原画像が白黒画像である。その他にも、それを応用したカラー画像の VSSS も提案されている [3]。次で、白黒画像とカラー画像の VSSS に

2.2 白黒画像の VSSS

ついて説明する。

説明するにあたって、原画像を分散し、できた画像をシェア画像といい、原画像の 1 画素からなるシェア画像中の画素の集まりをシェア画素と呼ぶ。

2.2 白黒画像の VSSS

本節では、原画像が白黒画像である場合についての VSSS の一手法を説明する [4]。

2.2.1 基本的な考え方

白黒画像の各画素は、白か黒の異なる画素構造を持ち、分散されたスライド上の各シェア画素もこの白黒 2 種類の画素から構成される。

復元された原画像は、黒画素の密度の違いを持たせることによって、視覚的に認識することができるようになっている。黒の密度変化を表現するために、原画像上での 1 画素は、 m 個のシェア画素に分割されて表現される。

シェア画像を重ね合わせた際、原画像が黒画素のシェア画素がすべて黒になるように重なり、白画素だったところは、すべてが黒とはならず白い画素が残るように重なる。

2.2.2 白黒画像の VSSS の例

視覚復号型 (3,3) 閾値法で、1 画素を 4 分割 ($m = 4$) して暗号化を行う VSSS を例として挙げる。視覚復号型 (3,3) 閾値法とは、原画像を 3 枚のシェア画像に分散し、3 枚のシェア画像を物理的に重ね合わせて、原画像を復元するものである。このとき、3 枚未満のスライドだけでは原画像が一切分からないように秘密画像が分散される。

シェア画素を構成している画素に対して、その画素が黒であれば 1、白であれば 0 として、1 個のシェア画素を行ベクトルで表現する。そして、それぞれのシェア画素に対応する合計 3 個の行ベクトルを集めて行列で表現することにする。シェア画像を物理的に重ね合わせて、人間の目で復号する操作は、各行列において行ベクトルの各要素毎の論理和を計算す

2.2 白黒画像の VSSS

ることに相当する。それぞれの白黒画素に分散する際の 3 行 4 列の行列の集合は、次のように表される。

$$C_0 = \left\{ \left[\begin{array}{cccc} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{array} \right] \text{の列の順列で得られる全行列} \right\}$$

$$C_1 = \left\{ \left[\begin{array}{cccc} 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{array} \right] \text{の列の順列で得られる全行列} \right\}$$

- 白画素を暗号化するとき (3 つのシェア画素に分けるとき)

C_0 の中から 1 個の行列をランダムに選び出す。例えば、

$$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

がランダムで選ばれたとするとスライドには以下のように印刷される。これは、復号時に重ねたとき、人間の目は白画素と認識する。

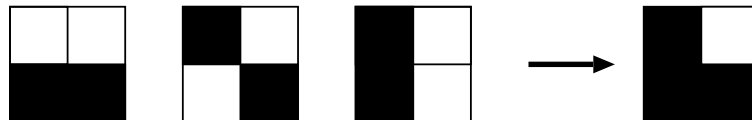


図 2.1 シェア画素の重ね合わせによる画素の復元 (白画素)

- 黒画素を暗号化するとき (3 つのシェア画素に分けるとき)

C_1 の中から 1 個の行列をランダムに選び出す。

$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

がランダムで選ばれたとするとスライドには以下のように印刷される。これは、復号時に重ねたとき、人間の目は黒画素と認識する。

2.3 カラー画像の VSSS

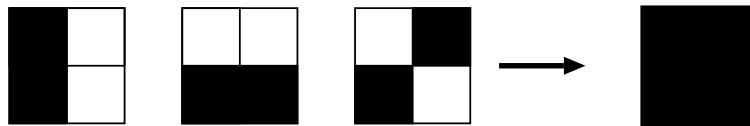


図 2.2 シェア画素の重ね合わせによる画素の復元（黒画素）

このように、暗号化のときは、白画素と黒画素をシェア画素に置き換える。そして、復号化の際は、シェア画像を必要枚数重ねた時、原画像において黒画素であった部分が人間の目では、黒が強調されて見える。

2.3 カラー画像の VSSS

本節では、原画像がカラー画像である場合の VSSS について説明する [3]。

2.3.1 基本的な考え方

カラー画像の各画素の色は、分散されたシェア画像の各シェア画素の色の重ね合わせによって表現する。このときのシェア画素で使用する色は、CMY (Cyan, Magenta, Yellow) である。復元された原画像は、CMY の色を重ねる合わせることによって色を表現することが可能で、視覚的に認識される。このとき、色を表現する際に使わない CMY は、黒画素と重ねることによって原画像の色を忠実に表現することが可能である。

しかし、CMY を重ねることによって表現することができる色しか暗号化できないという欠点も持っている。CMY を重ねて表現できる色をハッセ図で表したものが図 2.3 である。既存の VSSS では、ハッセ図によって表現される色しか利用することはできない。

2.3.2 カラー画像の VSSS 例

カラー画像の VSSS で、1 画素を 8 分割して暗号化を行う VSSS を例として挙げる。この場合は、原画像を 2 枚のシェア画像に分散し、2 枚のシェア画像を物理的に重ね合わせて、人間の目で原画像を復元する。このとき、1 枚のシェア画像だけでは、原画像が一切分から

2.3 カラー画像の VSSS

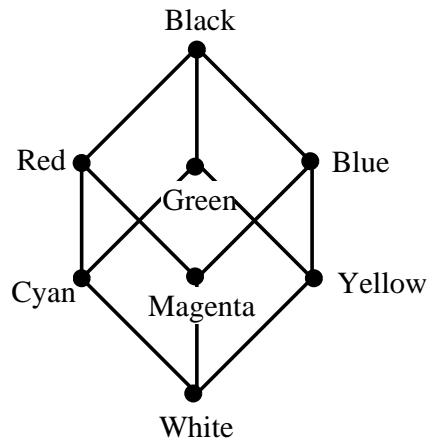


図 2.3 色の重ね合わせを表したハッセ図

ないようにシェア画像に分散する。

シェア画素を構成している 8 個の画素に対して、そのシェア画素が、黒であれば 1、白であれば 0、Cyan であれば C、Magenta であれば M、Yellow であれば Y として、1 個のシェア画素を行ベクトルで表現し、それぞれのシェア画素に対応する合計 2 個の行ベクトルを集めて行列で表現することにする。シェア画像を物理的に重ね合わせるで、人間の目で復号する操作は、各行列において行ベクトルの各要素毎の論理和を計算することに相当する。

それぞれのカラー画素に分散する際の 2 行 8 列の行列集合は、次のように表される。

$$C_0 = \left\{ \left[\begin{array}{cccccccc} 0 & Y & M & C & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & Y & M & C & 1 \end{array} \right] \text{の列の順列で得られる全行列} \right\}$$

$$C_Y = \left\{ \left[\begin{array}{cccccccc} Y & 0 & M & C & 1 & 1 & 1 & 1 \\ 0 & Y & 1 & 1 & M & C & 1 & 1 \end{array} \right] \text{の列の順列で得られる全行列} \right\}$$

$$C_M = \left\{ \left[\begin{array}{cccccccc} M & 0 & C & Y & 1 & 1 & 1 & 1 \\ 0 & M & 1 & 1 & C & Y & 1 & 1 \end{array} \right] \text{の列の順列で得られる全行列} \right\}$$

$$C_C = \left\{ \left[\begin{array}{cccccccc} C & 0 & Y & M & 1 & 1 & 1 & 1 \\ 0 & C & 1 & 1 & Y & M & 1 & 1 \end{array} \right] \text{の列の順列で得られる全行列} \right\}$$

$$C_R = \left\{ \left[\begin{array}{cccccccc} Y & M & C & 0 & 1 & 1 & 1 & 1 \\ M & Y & 1 & 1 & C & 0 & 1 & 1 \end{array} \right] \text{の列の順列で得られる全行列} \right\}$$

2.4 既存の手法の問題点

$$C_G = \left\{ \left[\begin{array}{cccccccc} C & Y & M & 0 & 1 & 1 & 1 & 1 \\ Y & C & 1 & 1 & M & 0 & 1 & 1 \end{array} \right] \text{の列の順列で得られる全行列} \right\}$$

$$C_B = \left\{ \left[\begin{array}{cccccccc} M & C & Y & 0 & 1 & 1 & 1 & 1 \\ C & M & 1 & 1 & Y & 0 & 1 & 1 \end{array} \right] \text{の列の順列で得られる全行列} \right\}$$

$$C_1 = \left\{ \left[\begin{array}{cccccccc} Y & M & C & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & Y & M & C & 0 \end{array} \right] \text{の列の順列で得られる全行列} \right\}$$

このようにカラー画素をシェア画素に置き換え、シェア画像を重ねた時、原画像の色と同じ色を人間が認識することができる。また、シェア画像のどの色のシェア画素をとっても CMY と白黒とが 2 つずつ使われているので、原画像の情報が洩れることはない。

2.4 既存の手法の問題点

今まで述べてきた既存の VSSS では、以下の問題点がある。

- 復元した際に、原画像の任意の色を表現できない

第 2.3 節で説明したように、現在提案されている VSSS では、CMY の色の組合せで表現できる色でしか、暗号化、復号化をすることができない。CMY で構成される任意の色を暗号化するのは難しく、任意の色を既存の VSSS を用いて復元するのは困難である。

- 復元した原画像が暗い

原画像を分散した際に、それぞれのシェア画像に多くの黒画素を使用している。そのため、復元するために重ねた際、その黒画素が残るので原画像よりもかなり暗くなり、復元画像の色の再現性は落ちている。

例えば、カラー画像の VSSS の例では、秘密画像の 1 画素のシェア画素は 8 分割されて表現されている。それらのシェア画像を重ねあわせるとシェア画素の $\frac{3}{4}$ が黒画素とな

2.4 既存の手法の問題点

る。また、秘密画像と同じ色を表している部分はシェア画素の $\frac{1}{4}$ なので、全体的に画像が暗くなってしまふ。

第 3 章

目的・目標

既存のカラー画像では、第 2 章で述べたようにハッセ図で表現できるような離散的な色表現しかできず、任意の色表現をすることができない。そこで、本研究では、忠実に原画像を復元する視覚復号型秘密分散法の実現を目的とする。

既存の VSSS では、ハッセ図で表された色でしか暗号化、復号化することができず、離散的な色しか表現できなかつた。また、原画像の情報を隠すために黒画素を使用しているため、復元した画像が全体的に暗い傾向にある。よって、目標を以下のように定めた。

- 離散的であっても任意の細かさで色表現を連続的にする
- 復元した画像を全体的に明るくし、できるだけ原画像に近づける
- これらの目標を満たした実際に動かすことができるプログラムを作成し、評価する

第 4 章

色分散復元手法

扇面積分割手法

本章では、私が提案する色分散復元手法と扇面積分割手法について説明をする。その次に自分が実装した方法について述べる。

既存の VSSS の特徴は、シェア画像をある枚数重ね合わせれば原画像を復元でき、シェア画像をそれぞれみても原画像の情報を得ることはできない性質である。今回の手法では、この VSSS の特徴を失わないように、かつ、任意の色のカラー画像にも対応できる VSSS を提案する。

4.1 手法

本節では、私が提案する 2 つの手法について説明する。扇面積分割手法は、色分散復元手法を拡張した手法である。

4.1.1 基本原理

原画像から 1 ピクセル毎取り出し、処理を行う。処理は、1 ピクセルをいくつかの円(シェア画素)に分散し、分散したシェア画素は Black, Cyan, Magenta, Yellow の扇形からなる(図 4.1)。

これらの色の扇形の重ね合わせで原画像の色を表現する。分散方法としては、原画像の CMY 成分を取り出し、その成分に応じて黒の扇形と重なる面積を決定する。全く使わない

4.1 手法

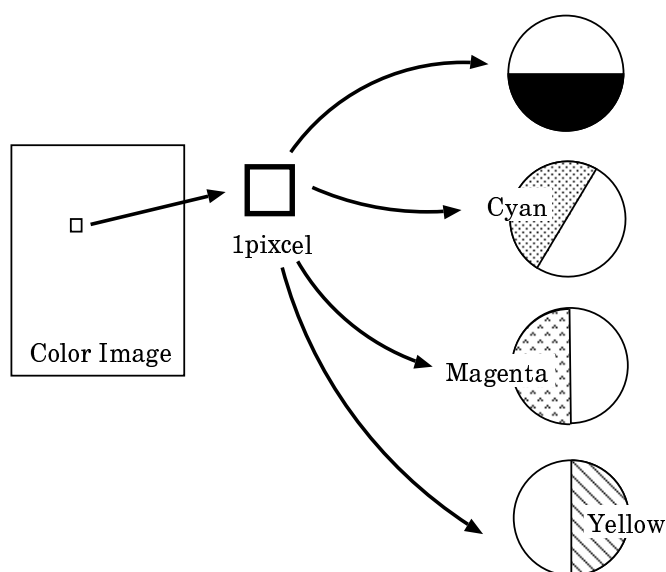


図 4.1 処理方針

色の扇形は、黒の扇形とすべてが重なり合うようにする。

4.1.2 色分散復元手法

CMYK のそれぞれの色に対し、180 度の扇形を 1 枚ずつ用意する。この場合、どの色を表現するにあたって、すべて同じ 180 度の扇形の重ね合わせで表現されるので、原画像の情報が漏れる可能性はない。しかし、Black の扇形も 180 度なので、円の $\frac{1}{2}$ の面積で色を表現することになり、復元画像が暗くなってしまいう傾向にある。そこで、この手法の拡張として、次の扇面積分割手法を提案した。

4.1.3 扇面積分割手法

この手法は、復元した画像をより明るくするために、色分散復元手法を拡張した手法である。この手法では、黒の扇形の面積を小さくすることによって復元した原画像をより明るくする。図 4.2 は、1 ピクセルを扇形に分散する際の、シェア画素の枚数と扇形の形を表している。色は、円内の黒の扇形以外の部分で表現される。色成分の割合が 100 % のときに、黒の扇形以外の部分がすべてその色で占められるとし、割合に応じて面積を決定する。黒の扇

4.1 手法

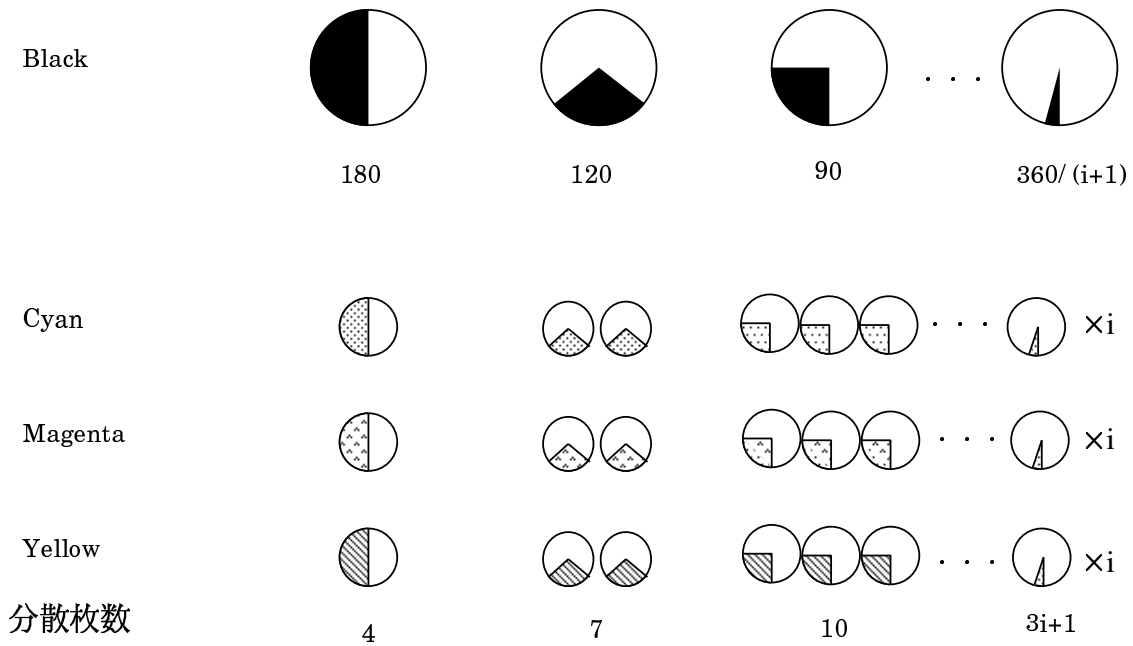


図 4.2 扇面積分割手法

形の面積が減るにつれて、色を表現する面積が広くなり、復元画像をより明るくすることが可能である。しかし、この場合、分散枚数が多くなってしまいうという欠点がある。

4.1.4 RGB から CMY への変換

この節で、原画像をシェア画像へ分散する際の RGB から CMY への変換について説明する。1 ピクセル単位で原画像から取り出して処理を行う。取り出した 1 ピクセルの RGB 成分を調べ、その RGB 成分を CMY 成分に変換する。変換の仕方は、以下の通りである。

$$\text{Cyan}(\%) = 100 - \text{Red}(\%)$$

$$\text{Magenta}(\%) = 100 - \text{Green}(\%)$$

$$\text{Yellow}(\%) = 100 - \text{Blue}(\%)$$

その後、1 ピクセルの CMY の成分ごとに処理を行い円内の扇形の位置を決定する。

4.1 手法

4.1.5 円内の扇形の位置決め（色分散復元手法）

シェア画像に分散する際、1ピクセル毎に変換される4つの円について説明する。円を以下のように円の中心から上方向に伸ばした線と交わった点を0度とし、0度から右回りに90、180、270度と定義する。

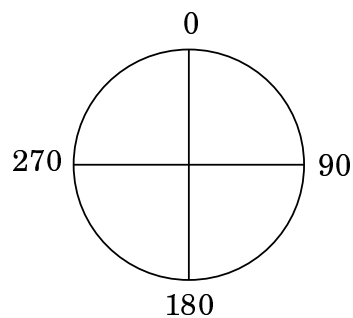


図 4.3 円内の扇形の位置決め

Black は、円の90度～270度（下半分）に固定する。

Cyan は、成分の割合によって Black の270度部分から右回りにスライドさせる。

Magenta は、成分の割合によって Black の270度部分から右回りにスライドさせる。

Yellow は、成分の割合によって Black の90度部分から左回りにスライドさせる。

（図 4.4 参照）

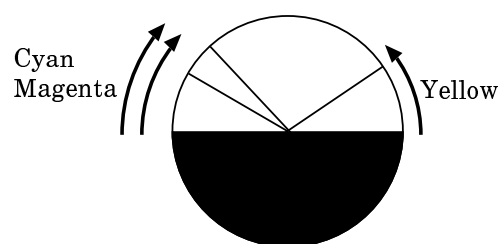


図 4.4 色分散復元手法でのシェア画素（円）の内部

4.1 手法

4.1.6 円内の扇形の位置決め（扇面積分割手法）

シェア画像に分散する際、1ピクセル毎に変換される4つの円について説明する。色分散復元手法と同様に、上の点を0度とし、右回りに90、180、270度と定義する（図4.3）。

円の半径を1とし、円の面積を π とおく。そして、分散枚数を $3i + 1$ とおくと、黒の扇形の中心角は、 $\frac{360}{i+1}$ となる。よって、円内の扇形の位置決めは、以下のようになる。

Black は、円の $180 - \frac{180}{i+1}$ 度 \sim $180 + \frac{180}{i+1}$ 度に固定する。

Cyan は、成分の割合によって Black の $180 + \frac{180}{i+1}$ 度部分から右回りにスライドさせる。

Magenta は、成分の割合によって Black の $180 + \frac{180}{i+1}$ 度部分から右回りにスライドさせる。

Yellow は、成分の割合によって Black の $180 - \frac{180}{i+1}$ 度部分から左回りにスライドさせる。

（図4.5 参照）

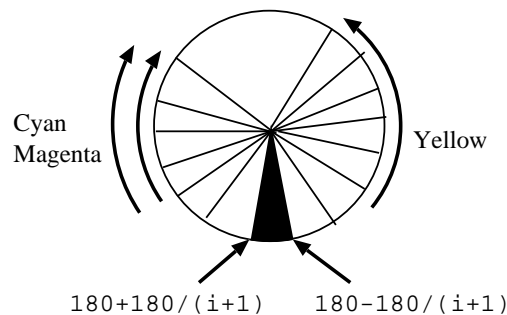


図 4.5 扇面積分割手法でのシェア画素（円）の内部

4.1.7 シェア画素への分散方法（色分散復元手法）

復元して、シェア画像を重ねた際、黒が円の半分を占めているので、残りの半分の面積で、原画像の色を表現する。CMY もそれぞれ半円を用意し、その傾きを変えることによって、シェア画像を重ねたときに Black を重なる面積を変更する。また、Black と重ならない部分の面積は原画像から取り出した1ピクセル毎のCMYの割合によって決定する。1ピクセル毎にCMYの半円の傾きを決定し、4つの扇形（Cyan, Magenta, Yellow, Black）に分けることで、それらの円を色ごとに集めた4つのシェア画像に分散することができる。

4.1 手法

円の半径を 1 とし、面積を π とおくと、それぞれのシェア画素での色成分の面積は、円の半分の $\frac{1}{2}\pi$ となる。

この面積の $\frac{1}{2}\pi$ のうち、それぞれの色成分の割合に応じて、シェア画像を重ねたときに Black と扇形の重ならない部分の面積が決定する。

CMY 成分の割合をそれぞれ、 $c\%$ 、 $m\%$ 、 $y\%$ とおくと、重ねたときに Black と重ならない部分の面積は、

$$\begin{aligned}\text{Cyan} &: \frac{1}{2}\pi \times \frac{c}{100} = \frac{c}{200}\pi \\ \text{Magenta} &: \frac{1}{2}\pi \times \frac{m}{100} = \frac{m}{200}\pi \\ \text{Yellow} &: \frac{1}{2}\pi \times \frac{y}{100} = \frac{y}{200}\pi\end{aligned}$$

となる。そして、それぞれの色が Black と重ならない部分の扇形の角度を、それぞれ、 θ_c 、 θ_m 、 θ_y とおくと、以下の式が成り立つ。

$$\begin{aligned}\text{Cyan} &: \pi \times \frac{\theta_c}{360} = \frac{c}{200}\pi \\ \text{Magenta} &: \pi \times \frac{\theta_m}{360} = \frac{m}{200}\pi \\ \text{Yellow} &: \pi \times \frac{\theta_y}{360} = \frac{y}{200}\pi\end{aligned}$$

これを、書き換えると、

$$\begin{aligned}\theta_c &= \frac{9}{5}c \\ \theta_m &= \frac{9}{5}m \\ \theta_y &= \frac{9}{5}y\end{aligned}$$

となる。よって、シェア画素内の扇形は以下のように位置することになる。

$$\begin{aligned}\text{Black} &: 90 \text{ 度} \sim 270 \text{ 度} \\ \text{Cyan} &: 90 + \frac{9}{5}c \text{ 度} \sim 270 + \frac{9}{5}c \text{ 度} \\ \text{Magenta} &: 90 + \frac{9}{5}m \text{ 度} \sim 270 + \frac{9}{5}m \text{ 度} \\ \text{Yellow} &: 90 - \frac{9}{5}y \text{ 度} \sim 270 - \frac{9}{5}c \text{ 度}\end{aligned}$$

4.1 手法

例えば、Cyan100%の場合は、270度から450度(90度)に位置する。0%の場合は、Blackと重なるように、90度から270度となる。

このようにして原画像の1ピクセル毎に4つの円に分解し、4つのシェア画像に分散していく。

4.1.8 シェア画素への分散方法(扇面積分割手法)

扇面積分割手法では、色表現可能な面積の部分をも色の扇形で埋めることによって、任意の色を表現することができる。円の半径を1とおくと、復元してシェア画像を重ねた際、黒の扇形的面積と色表現に使うことができる面積は、以下ようになる。

$$\text{黒の扇形的面積} : \frac{\pi}{i+1}$$

$$\text{色表現に使うことができる面積} : \pi - \frac{\pi}{i+1}$$

色表現に使うことができる面積の $\pi - \frac{\pi}{i+1}$ のうち、それぞれの色成分の割合に応じて、シェア画像を重ねたときにBlackと扇形の重ならない部分の面積が決定する。

CMY成分の割合をそれぞれ、 $c\%$ 、 $m\%$ 、 $y\%$ とおくと、重ねたときにBlackと重ならない部分の面積は、

$$\text{Cyan} : \left(\pi - \frac{\pi}{i+1} \right) \times \frac{c}{100} = \frac{\left(\pi - \frac{\pi}{i+1} \right) c}{100}$$

$$\text{Magenta} : \left(\pi - \frac{\pi}{i+1} \right) \times \frac{m}{100} = \frac{\left(\pi - \frac{\pi}{i+1} \right) m}{100}$$

$$\text{Yellow} : \left(\pi - \frac{\pi}{i+1} \right) \times \frac{y}{100} = \frac{\left(\pi - \frac{\pi}{i+1} \right) y}{100}$$

となる。そして、それぞれの色がBlackと重ならない部分の扇形の角度を、それぞれ、 θ_c 、 θ_m 、 θ_y とおくと、以下の式が成り立つ。

$$\text{Cyan} : \pi \times \frac{\theta_c}{360} = \frac{\left(\pi - \frac{\pi}{i+1} \right) c}{100}$$

$$\text{Magenta} : \pi \times \frac{\theta_m}{360} = \frac{\left(\pi - \frac{\pi}{i+1} \right) m}{100}$$

4.2 実装

$$\text{Yellow} : \pi \times \frac{\theta_y}{360} = \frac{\left(\pi - \frac{\pi}{i+1}\right) y}{100}$$

これを、書き換えると、

$$\theta_c = \frac{18}{5} \left(1 - \frac{1}{i+1}\right) c$$

$$\theta_m = \frac{18}{5} \left(1 - \frac{1}{i+1}\right) m$$

$$\theta_y = \frac{18}{5} \left(1 - \frac{1}{i+1}\right) y$$

となる。よって、シェア画素内の黒の扇形と黒と重ならない色の扇形は以下のように位置することになる。

$$\text{Black} : 180 - \frac{180}{i+1} \text{度} \sim 180 + \frac{180}{i+1} \text{度}$$

$$\text{Cyan} : 180 + \frac{180}{i+1} \text{度} \sim 180 + \frac{180}{i+1} + \frac{18}{5} \left(1 - \frac{1}{i+1}\right) c \text{度}$$

$$\text{Magenta} : 180 + \frac{180}{i+1} \text{度} \sim 180 + \frac{180}{i+1} + \frac{18}{5} \left(1 - \frac{1}{i+1}\right) m \text{度}$$

$$\text{Yellow} : 180 - \frac{180}{i+1} \text{度} \sim 180 - \frac{180}{i+1} - \frac{18}{5} \left(1 - \frac{1}{i+1}\right) y \text{度}$$

この黒と重ならない色の扇形の部分を分散した色の扇形で埋めていけばよい。

4.2 実装

今回の実装において、扇面積分割手法では、シェア画素が多くなってしまい、OHPシートを何枚も重ねなければならない。よってシートを何枚も重ねてしまうと光がシートを通らなくなってしまい全体的に暗くなってしまふ。そのため、この手法では実装は困難である。

そこで今回、色分散復元手法を実装することにした。OHPシートを4枚は、実際にOHPシートを重ねて復元画像を表現できる範囲内なのでこれを選択した。よって、本節では、色分散復元手法の実装について述べる。

4.2 実装

4.2.1 実装環境

実装を行うにあたって、実装環境について述べる。

- 実装では Java 言語を使用
- 原画像は gif ファイル
- プリンタ：EPSON, MJ-6000C
- OHP シート：プリンタ専用 OHP シート, MJOHPS1N

4.2.2 実装の処理単位

実際に原画像の 1 ピクセルを円に変換して復元画像を作成することは困難なので以下のよう
に考えた。円を正方形に変更し、それぞれの色成分が円で占めていた面積比と重なりを同
じようにする 4.7。

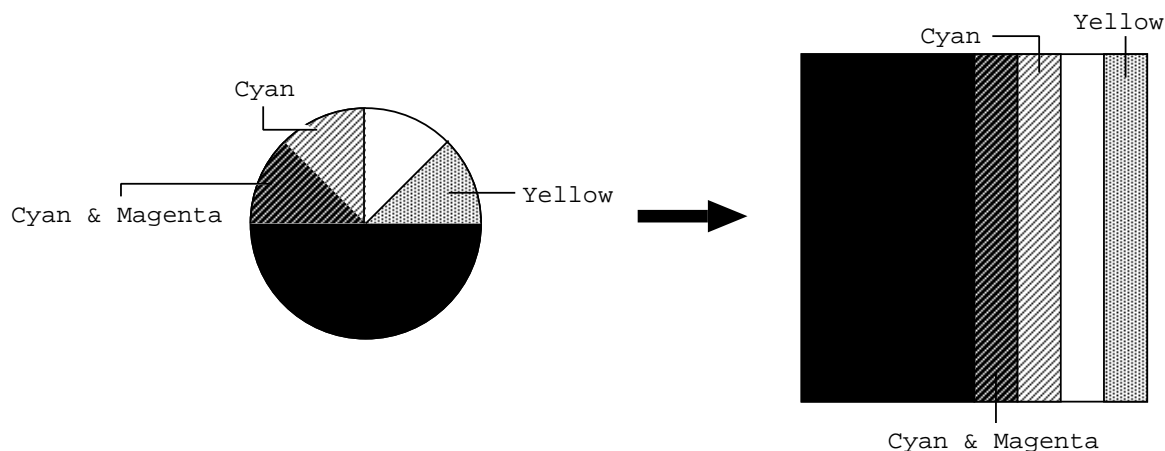


図 4.6 色分散復元手法での実装方法

4.2.3 シェア画素の作成

今回用いた Java の画像フォーマットでは、1 ピクセルは非常に小さいため、その 1 ピクセルを分割していると、手でシェア画像を重ねることが非常に困難となる。そのため、1 ピクセルを 8 × 8 ピクセルに拡大し、それをシェア画素とした。原画像の 1 ピクセルを 4 つの

4.2 実装

シェア画素（ 8×8 ピクセル）に分散し、これらの集まりである4つのシェア画像を作成する。それぞれの色成分は、円の時同様、シェア画素の半分を占める 8×4 ピクセルを持っている。これらの重ねあわせによって、原画像の色を表現する。また、 8×8 ピクセルに制限することによって、色成分の重ね合わせの表現方法に限界ができた。今回は、シェア画素中の色ピクセル（ 8×4 ピクセル）の存在パターンを人間の目で見えて色の変化が分かるように、5パターンとした。

シェア画素中（ 8×8 ）の色画素の存在範囲の位置付けを以下の図から、（存在する始めの列～存在する終りの列）と表す。図4.7のBlackをこの表現方法で表すと（1～4）となる。

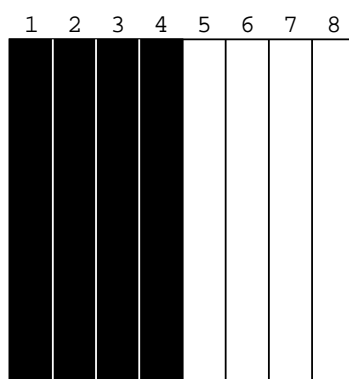


図 4.7 8×8 内での色画素の存在範囲の位置付け

この表現方法で、色の成分の割合に応じて、シェア画素中での色ピクセルの存在範囲を表現すると、

- 色成分の割合が、10%未満のとき
色画素は、（1～4）に位置し、シェア画素を重ねると Black とすべて重なる。
- 色成分の割合が、25%未満のとき
色画素は、（2～5）に位置し、シェア画素を重ねると（5～5）の部分が Black と重ならない。
- 色成分の割合が、50%未満のとき
色画素は、（3～6）に位置し、シェア画素を重ねると、（5～6）の部分が Black と重なら

4.2 実装

ない。

- 色成分の割合が、75%未満のとき

色画素は、(4~7)に位置し、シェア画素を重ねると、(5~7)の部分が Black と重ならない。

- 色成分の割合が、75%以上のとき

色画素は、(5~8)に位置し、シェア画素を重ねると、(5~8)の部分が Black と重ならない。

このように、1ピクセルの CMY 成分ごとに 8×8 ピクセルに変更する処理を行い、シェア画像に分散する。出来上がったシェア画像は、1枚1枚の秘密性を上げるため、シェア画素 (8×8 ピクセル) ごとにランダムに回転 (90 度、180 度、270 度、360 度) させ、シェア画素内で Black と重なる位置を変更している。

第 5 章

扇面積可変手法

本章では、カラー画像に対応した VSSS を実現するもう 1 つの手法である扇面積可変手法の説明をする。次に、この手法の実装方法について述べる。

5.1 手法

第 4 章で説明した色分散復元手法では、シェア画像のそれぞれ見ても原画像の情報を得ることができない。その半面、シェア画素の半分を黒が占めているため、復元した画像が原画像よりも暗くなってしまう。よって、扇面積可変手法では、シェア画像からは若干の情報が漏れてもいいということを前提とし、復元画像を明るくすることを目指した。

5.1.1 基本原理

秘密画像から 1 ピクセル毎取り出し、処理を行う。処理は、色分散復元手法と同様、1 ピクセルを 4 つの扇形に分散する。その際、重ねたときに少しでも Black の面積を小さくし、復元画像を明るくするため、Black を 120 度の扇形とした。また、少しでも色成分を表現できる面積を大きくするために、色成分を表す扇形は、120 度から 180 度まで変更可能な扇形とした。全く色成分を使わないときは、120 度とし Black の扇形と重ね、色成分の割合が大きいときは、180 度の扇形を使用することにより、色の円内での占める割合を増やすことができる。

5.1 手法

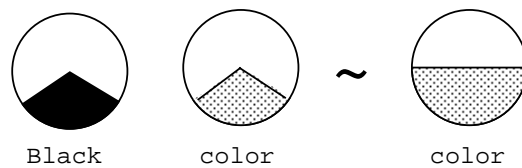


図 5.1 シェア画素について

5.1.2 円内の扇形の位置決め（扇面積可変手法）

シェア画像に分散する際、1 ピクセル毎に変換される 4 つの円について説明する。第 4 章と同様、円の上の点を 0 度とし、右回りに 90、180、270 度とする（図 4.3 参照）。

Black は、円の 120 度～240 度に固定する。

Cyan は、成分の割合によって Black の 240 度部分から右回りにスライドさせる。

Magenta は、成分の割合によって Black の 240 度部分から右回りにスライドさせる。

Yellow は、成分の割合によって Black の 120 度部分から左回りにスライドさせる。

（図 5.2 参照）

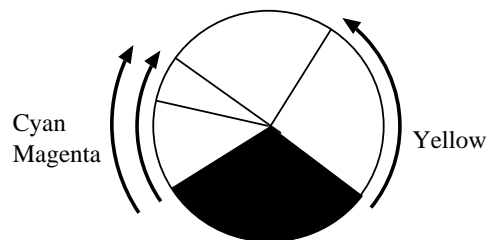


図 5.2 扇面積可変手法でのシェア画素（円）の内部

5.1.3 シェア画素への分散方法（扇面積可変手法）

復元して、シェア画像を重ねた際、Black が円の $\frac{1}{3}$ を占めているので、残りの $\frac{2}{3}$ の面積で原画像の色を表現する。CMY は、それぞれ 120 度から 180 度の変更可能な扇形を用意し、この扇形の傾きと扇形を変更することによって、シェア画像を重ねたときの Black と重なる面積を変更することができる。また、原画像の 1 ピクセル毎の CMY 成分の割合によって、

5.1 手法

シェア画像を重ねた際の Black と重ならない面積が決定する。

円の半径を 1 とし、面積を π とおくと、それぞれのシェア画素での色面積は、 $\frac{1}{3}\pi \sim \frac{1}{2}\pi$ となる。

この面積のうち、それぞれの色成分の割合に応じて、シェア画像を重ねたときに Black と重ならない面積が決定する。CMY 成分の割合をそれぞれ、 $c\%$ 、 $m\%$ 、 $y\%$ とおくと、重ねたときに Black と重ならない面積は、

$$\text{Cyan} : \frac{1}{2}\pi \times \frac{c}{100} = \frac{c}{200}\pi$$

$$\text{Magenta} : \frac{1}{2}\pi \times \frac{m}{100} = \frac{m}{200}\pi$$

$$\text{Yellow} : \frac{1}{2}\pi \times \frac{y}{100} = \frac{y}{200}\pi$$

となる。そして、それぞれの色が Black と重ならない部分の扇形の角度を、それぞれ、 $\theta_c, \theta_m, \theta_y$ とおくと、以下の式が成り立つ。

$$\text{Cyan} : \pi \times \frac{\theta_c}{360} = \frac{c}{200}\pi$$

$$\text{Magenta} : \pi \times \frac{\theta_m}{360} = \frac{m}{200}\pi$$

$$\text{Yellow} : \pi \times \frac{\theta_y}{360} = \frac{y}{200}\pi$$

これを、書き換えると、

$$\theta_c = \frac{9}{5}c$$

$$\theta_m = \frac{9}{5}m$$

$$\theta_y = \frac{9}{5}y$$

となる。よって、シェア画素内の扇形は以下のように位置することになる。

Black : 120 度 ~ 240 度

Cyan : $120 + \frac{9}{5}c$ 度 ~ $240 + \frac{9}{5}c$ 度

Magenta : $120 + \frac{9}{5}m$ 度 ~ $240 + \frac{9}{5}m$ 度

5.2 実装

$$\text{Yellow : } 120 - \frac{9}{5}y \text{ 度} \sim 240 - \frac{9}{5}c \text{ 度}$$

ただし、色成分の割合が $\frac{100}{3}\%$ 以下のときは、120 度の扇形を使い、それ以上のときは 180 度の扇形を使う。こうすることによって、使わない色の扇形は、Black と重なることができるので、正確に色を表現することができる。

このようにして、秘密画像の 1 ピクセル毎に 4 つの円に分解し、4 つのシェア画像に分散していく。

5.2 実装

本節では、扇面積可変手法の実装について説明する。

第 4.2.2 節と同様、1 ピクセルを円に変更するのは困難だったので、円を正方形に変更し、それぞれの色成分が円で占めていた面積比と重なりを同じようにする。(図 4.7)

5.2.1 シェア画素の作成

実装の際、Black は、円の $\frac{1}{3}$ が占めているので、1 ピクセルを四角で分割しやすい 3 の倍数である 9×9 ピクセルに拡大し、それをシェア画素とした。秘密画像の 1 ピクセルを 4 つのシェア画素 (9×9 ピクセル) に分散し、これら集まりである 4 つのシェア画像を作成する。Black は、 9×9 ピクセルの $\frac{1}{3}$ の面積である 9×3 ピクセルである。色成分は変更可能な $9 \times 3 \sim 9 \times 4$ ピクセルとした。これらのシェア画素の重ね合わせによって、秘密画像の色を表現する。

同様に、色成分の重ね合わせの表現方法は人間の目で見ても色の変化が分かるように、5 パターンとした。

シェア画素中の色ピクセル ($8 \times 3 \sim 8 \times 4$) の存在パターンを色分散復元手法と同じ表現方法で、色の成分の割合に応じてシェア画素中での色ピクセルの存在範囲を表現すると、

- 色成分の割合が、10%未満のとき

色画素は、(1~3) に位置し、シェア画素を重ねると Black とすべて重なる。

5.2 実装

- 色成分の割合が、25%未満のとき

この場合の色画素は、(2~4)に位置し、シェア画素を重ねると、(4~4)の部分が Black と重ならない。

- 色成分の割合が、50%未満のとき

この場合の色画素は、(3~5)に位置し、シェア画素を重ねると、(4~5)の部分が Black と重ならない。

- 色成分の割合が、75%未満のとき

この場合の色画素は、(4~6)に位置し、シェア画素を重ねると、(4~6)の部分が Black と重ならない。

- 色成分の割合が、75%以上のとき

この場合の色画素は、(4~7)に位置し、シェア画素を重ねると、(4~7)の部分が Black と重ならない。

このように、1 ピクセルの CMY 成分ごとに処理を行い、シェア画像に分散する。できあがったシェア画像は、1 枚 1 枚の秘密性を上げるため、シェア画素 (9 × 9 ピクセル) ごとにランダムに回転 (90 度、180 度、270 度、360 度) させ、シェア画素内で Black と重なる位置を変更している。しかし、このままでは色成分の割合に応じてシェア画素内の色占めるピクセル数が異なるため、シェア画像を見たら原画像の情報が漏れてしまう。よって、さらに Cyan, Magenta, Yellow のシェア画像のシェア画素を乱数により交換することによって、色によるピクセル数の違いを目立たなくした。

第 6 章

考察

本節では、第 4 章で提案した色の分散復元手法と扇面積分割手法、第 5 章で説明した扇面積可変手法との性質について述べる。

6.1 復元画像の明るさについて

本節では、提案手法で復元した画像と原画像を比較し、明るさがどのように変化したかについて検証を行う。

6.1.1 色の分散復元手法

まず、シートの重なりによって色は変化しない(詳細は、付録 A を参照してほしい)。よって、シートはどんな重ね合わせをしてもすべて同じなので、CMY の成分にわけて、原画像の RGB 成分がどう変化し、復元画像の明るさがどう変化するかを検証する。このときの、シェア画素は円とする。

シェア画素の面積を 100、秘密画像の 1 ピクセルでの CYM 成分が占める割合を $c\%$, $m\%$, $y\%$ 、RGB 成分の割合を $r\%$, $g\%$, $b\%$ とおく。また、4 つのシェア画素を重ねたものに RGB 成分がそれぞれ 100% の光が当たった場合を考える。

- Cyan を使って色表現するとき

この場合、シェア画素を重ねた際に、存在する色は、Cyan, Black と透明である。この

6.1 復元画像の明るさについて

重ねたシェア画素内でそれぞれが占める面積は、

$$\text{Cyan} : \frac{100}{2} \times \frac{c}{100} = \frac{c}{2}$$

$$\text{Black} : 100 \times \frac{1}{2} = 50$$

$$\text{透明} : \frac{100}{2} \times \frac{100-c}{100} = \frac{100-c}{2}$$

となる。この重ねたシェア画素に RGB100%が当たると、Cyan は Red を吸収し、Green と Blue を通すので、この Cyan の部分は、RGB の 0 %, $\frac{c}{2}$ %, $\frac{c}{2}$ %を通過させる。Black は、RGB をすべて吸収してしまうので、0 %, 0 %, 0 %となる。透明の部分は、RGB を全て通過させるので、 $\frac{100-c}{2}$ %, $\frac{100-c}{2}$ %, $\frac{100-c}{2}$ %となる。よって、この場合、重ねたシェア画素が通す RGB は、

$$\text{Red} : 0 + 0 + \frac{100-c}{2} = \frac{100-c}{2} \%$$

$$\text{Green} : \frac{c}{2} + 0 + \frac{100-c}{2} = 50\%$$

$$\text{Blue} : \frac{c}{2} + 0 + \frac{100-c}{2} = 50\%$$

となる。

- Magenta を使って色表現する

この場合、シェア画素を重ねた際に、存在する色は、Magenta, Black と透明である。

この重ねたシェア画素内でそれぞれが占める面積は、

$$\text{Magenta} : \frac{100}{2} \times \frac{m}{100} = \frac{m}{2}$$

$$\text{Black} : 100 \times \frac{1}{2} = 50$$

$$\text{透明} : \frac{100}{2} \times \frac{100-m}{100} = \frac{100-m}{2}$$

となる。この重ねたシェア画素に RGB100%が当たると、Magenta は、Green を吸収し、Red と Blue を通すので、この Magenta の部分は、RGB の $\frac{m}{2}$ %, 0 %, $\frac{m}{2}$ %を通過させる。Black は、RGB をすべて吸収してしまうので、0 %, 0 %, 0 %となる。透明

6.1 復元画像の明るさについて

の部分は、RGB を全て通過させるので、 $\frac{100-m}{2}\%$ 、 $\frac{100-m}{2}\%$ 、 $\frac{100-m}{2}\%$ となる。よって、この場合、重ねたシェア画素が通す RGB は、

$$\begin{aligned}\text{Red} &: \frac{m}{2} + 0 + \frac{100-m}{2} = 50\% \\ \text{Green} &: 0 + 0 + \frac{100-m}{2} = \frac{100-m}{2}\% \\ \text{Blue} &: \frac{m}{2} + 0 + \frac{100-m}{2} = 50\%\end{aligned}$$

となる。

- Yellow を使って色表現する場合、シェア画素を重ねた際に、存在する色は、Yellow, Black と透明である。この重ねたシェア画素内でそれぞれが占める面積は、

$$\begin{aligned}\text{Yellow} &: \frac{100}{2} \times \frac{y}{100} = \frac{y}{2} \\ \text{Black} &: 100 \times \frac{1}{2} = 50 \\ \text{透明} &: \frac{100}{2} \times \frac{100-y}{100} = \frac{100-y}{2}\end{aligned}$$

となる。この重ねたシェア画素に RGB100%が当たると、Yellow は、Blue を吸収し、Red と Green を通すので、この Yellow の部分は、RGB の $\frac{y}{2}\%$ 、 $\frac{y}{2}\%$ 、 0% 、を通過させる。Black は、RGB をすべて吸収してしまうので、 0% 、 0% 、 0% となる。透明の部分は、RGB を全て通過させるので、 $\frac{100-y}{2}\%$ 、 $\frac{100-y}{2}\%$ 、 $\frac{100-y}{2}\%$ となる。よって、この場合、重ねたシェア画素が通す RGB は、

$$\begin{aligned}\text{Red} &: \frac{y}{2} + 0 + \frac{100-y}{2} = 50\% \\ \text{Green} &: \frac{y}{2} + 0 + \frac{100-y}{2} = 50\% \\ \text{Blue} &: 0 + 0 + \frac{100-y}{2} = \frac{100-y}{2}\%\end{aligned}$$

となる。

これら 3 つの RGB をまとめると、以下のようになる。

$$\text{Red} : \frac{100-c}{2}\%$$

6.1 復元画像の明るさについて

$$\begin{aligned}\text{Green} &: \frac{100 - m}{2} \% \\ \text{Blue} &: \frac{100 - y}{2} \%\end{aligned}$$

これを、RGB に置き換えると、

$$\begin{aligned}\text{Red} &: \frac{100 - (100 - r)}{2} = \frac{r}{2} \% \\ \text{Green} &: \frac{100 - (100 - g)}{2} = \frac{g}{2} \% \\ \text{Blue} &: \frac{100 - (100 - b)}{2} = \frac{b}{2} \%\end{aligned}$$

となる。よって、色の分散復元手法の復元画像の値 (r', g', b') を原画像の値 (r, g, b) で表現すると、式 6.1 となる。

$$\begin{pmatrix} r' \\ g' \\ b' \end{pmatrix} = \frac{1}{2} \begin{pmatrix} r \\ g \\ b \end{pmatrix} \% \quad (6.1)$$

式 6.1 より、色の分散復元手法の復元画像は、原画像に比べて明るさが 50 % になるということが分かった。

6.1.2 扇面積分割手法

まず、シートの重なりによって色は変化しない(詳細は、付録 A を参照してほしい)。よって、シートはどんな重ね合わせをしてもすべて同じなので、CMY の成分にわけて、原画像の RGB 成分がどう変化し、復元画像の明るさがどう変化するかを検証する。このときの、シェア画素は円とする。

- Cyan を使って色表現するとき

この場合、シェア画素を重ねた際に、存在する色は、Cyan, Black と透明である。この重ねたシェア画素内でそれぞれが占める面積は、

$$\begin{aligned}\text{Black} &: 100 \times \frac{1}{i+1} = \frac{100}{i+1} \\ \text{Cyan} &: \left(100 - \frac{100}{i+1}\right) \times \frac{c}{100}\end{aligned}$$

6.1 復元画像の明るさについて

$$\text{透明} : \left(100 - \frac{100}{i+1}\right) \times \frac{100-c}{100}$$

となる。この重ねたシェア画素に RGB100%が当たると、Cyan は Red を吸収し、Green と Blue を通すので、この Cyan の部分は、RGB の 0 %, $\left(100 - \frac{100}{i+1}\right) \times \frac{c}{100}$ %, $\left(100 - \frac{100}{i+1}\right) \times \frac{c}{100}$ %を通過させる。Black は、RGB をすべて吸収してしまうので、0 %, 0 %, 0 %となる。透明の部分は、RGB を全て通過させるので、 $\left(100 - \frac{100}{i+1}\right) \times \frac{100-c}{100}$ %, $\left(100 - \frac{100}{i+1}\right) \times \frac{100-c}{100}$ %, $\left(100 - \frac{100}{i+1}\right) \times \frac{100-c}{100}$ %となる。よって、この場合、重ねたシェア画素が通す RGB は、

$$\text{Red} : 0 + 0 + \left(100 - \frac{100}{i+1}\right) \times \frac{100-c}{100} = \left(100 - \frac{100}{i+1}\right) \times \frac{100-c}{100} \%$$

$$\text{Green} : \left(100 - \frac{100}{i+1}\right) \times \frac{c}{100} + 0 + \left(100 - \frac{100}{i+1}\right) \times \frac{100-c}{100} = 100 - \frac{100}{i+1} \%$$

$$\text{Blue} : \left(100 - \frac{100}{i+1}\right) \times \frac{c}{100} + 0 + \left(100 - \frac{100}{i+1}\right) \times \frac{100-c}{100} = 100 - \frac{100}{i+1} \%$$

となる。

- Magenta を使って色表現するとき

この場合、シェア画素を重ねた際に、存在する色は、Magenta, Black と透明である。

この重ねたシェア画素内でそれぞれが占める面積は、

$$\text{Black} : 100 \times \frac{1}{i+1} = \frac{100}{i+1}$$

$$\text{Magenta} : \left(100 - \frac{100}{i+1}\right) \times \frac{m}{100}$$

$$\text{透明} : \left(100 - \frac{100}{i+1}\right) \times \frac{100-m}{100}$$

となる。この重ねたシェア画素に RGB100%が当たると、Magenta は、Green を吸収し、Red と Blue を通すので、この Magenta の部分は、RGB の $\left(100 - \frac{100}{i+1}\right) \times \frac{m}{100}$ %, 0 %, $\left(100 - \frac{100}{i+1}\right) \times \frac{m}{100}$ %を通過させる。Black は、RGB をすべて吸収してしまうので、0 %, 0 %, 0 %となる。透明の部分は、RGB を全て通過させるので、 $\left(100 - \frac{100}{i+1}\right) \times \frac{100-m}{100}$ %, $\left(100 - \frac{100}{i+1}\right) \times \frac{100-m}{100}$ %, $\left(100 - \frac{100}{i+1}\right) \times \frac{100-m}{100}$ %となる。よって、この場合、重ねたシェア画素が通す RGB は、

$$\text{Red} : \left(100 - \frac{100}{i+1}\right) \times \frac{m}{100} + 0 + \left(100 - \frac{100}{i+1}\right) \times \frac{100-m}{100} = 100 - \frac{100}{i+1} \%$$

6.1 復元画像の明るさについて

$$\text{Green} : 0 + 0 + \left(100 - \frac{100}{i+1}\right) \times \frac{100 - m}{100} = \left(100 - \frac{100}{i+1}\right) \times \frac{100 - m}{100} \%$$

$$\text{Blue} : \left(100 - \frac{100}{i+1}\right) \times \frac{m}{100} + 0 + \left(100 - \frac{100}{i+1}\right) \times \frac{100 - m}{100} = 100 - \frac{100}{i+1} \%$$

となる。

- Yellow を使って色表現する場合、シェア画素を重ねた際に、存在する色は、Yellow, Black と透明である。この重ねたシェア画素内でそれぞれが占める面積は、

$$\text{Black} : 100 \times \frac{1}{i+1} = \frac{100}{i+1}$$

$$\text{Yellow} : \left(100 - \frac{100}{i+1}\right) \times \frac{y}{100}$$

$$\text{透明} : \left(100 - \frac{100}{i+1}\right) \times \frac{100 - y}{100}$$

となる。この重ねたシェア画素に RGB100%が当たると、Yellow は、Blue を吸収し、Red と Green を通すので、この Yellow の部分は、RGB の $\left(100 - \frac{100}{i+1}\right) \times \frac{y}{100} \%$, $\left(100 - \frac{100}{i+1}\right) \times \frac{y}{100} \%$, 0 %, を通過させる。Black は、RGB をすべて吸収してしまうので、0 %, 0 %, 0 %となる。透明の部分は、RGB を全て通過させるので、 $\left(100 - \frac{100}{i+1}\right) \times \frac{100-y}{100} \%$, $\left(100 - \frac{100}{i+1}\right) \times \frac{100-y}{100} \%$, $\left(100 - \frac{100}{i+1}\right) \times \frac{100-y}{100} \%$ となる。よって、この場合、重ねたシェア画素が通す RGB は、

$$\text{Red} : \left(100 - \frac{100}{i+1}\right) \times \frac{y}{100} + 0 + \left(100 - \frac{100}{i+1}\right) \times \frac{100 - y}{100} = 100 - \frac{100}{i+1} \%$$

$$\text{Green} : \left(100 - \frac{100}{i+1}\right) \times \frac{y}{100} + 0 + \left(100 - \frac{100}{i+1}\right) \times \frac{100 - y}{100} = 100 - \frac{100}{i+1} \%$$

$$\text{Blue} : 0 + 0 + \left(100 - \frac{100}{i+1}\right) \times \frac{100 - y}{100} = \left(100 - \frac{100}{i+1}\right) \times \frac{100 - y}{100} \%$$

となる。

これら 3 つの色の RGB をまとめると、以下のようになる。

$$\text{Red} : \left(100 - \frac{100}{i+1}\right) \times \frac{100 - c}{100} \%$$

$$\text{Green} : \left(100 - \frac{100}{i+1}\right) \times \frac{100 - m}{100} \%$$

6.1 復元画像の明るさについて

$$\text{Blue} : \left(100 - \frac{100}{i+1}\right) \times \frac{100-y}{100} \%$$

この i が ∞ に限りなく近付くとすると、

$$\text{Red} : \lim_{i \rightarrow \infty} \left\{ \left(100 - \frac{100}{i+1}\right) \times \frac{100-c}{100} \right\} = 100 - c = r \%$$

$$\text{Green} : \lim_{i \rightarrow \infty} \left\{ \left(100 - \frac{100}{i+1}\right) \times \frac{100-m}{100} \right\} = 100 - m = g \%$$

$$\text{Blue} : \lim_{i \rightarrow \infty} \left\{ \left(100 - \frac{100}{i+1}\right) \times \frac{100-y}{100} \right\} = 100 - y = b \%$$

となる。よって、扇面積分割手法の復元画像の値 (r', g', b') を原画像の値 (r, g, b) で表現すると、式 6.2 となる。

$$\lim_{i \rightarrow \infty} \begin{pmatrix} r' \\ g' \\ b' \end{pmatrix} = \begin{pmatrix} r \\ g \\ b \end{pmatrix} \% \quad (6.2)$$

式 6.2 より、扇面積分割手法の復元画像は、 i を限りなく無限大に近付けば、明るさは原画像に近付くということが上の式で証明された。

6.1.3 扇面積可変手法

まず、シートの重なりによって色は変化しない(詳細は、付録 A を参照してほしい)。よって、シートはどんな重ね合わせをしてもすべて同じなので、CMY の成分にわけて、原画像の RGB 成分がどう変化し、復元画像の明るさがどう変化するかを検証する。このときの、シェア画素は円とする。

- Cyan を使って色表現するとき

この場合、シェア画素を重ねた際に、存在する色は、Cyan, Black と透明である。この重ねたシェア画素内でそれぞれが占める面積は、

$$\text{Cyan} : \frac{100}{2} \times \frac{c}{100} = \frac{c}{2}$$

$$\text{Black} : 100 \times \frac{1}{3} = \frac{100}{3}$$

$$\text{透明} : 100 - \frac{100}{3} - \frac{c}{2} = \frac{200}{3} - \frac{c}{2}$$

6.1 復元画像の明るさについて

となる。この重ねたシエア画素に RGB100%が当たると、Cyan は Red を吸収し、Green と Blue を通すので、この Cyan の部分は、RGB の 0% , $\frac{c}{2}\%$, $\frac{c}{2}\%$ を通過させる。Black は、RGB をすべて吸収してしまうので、 0% , 0% , 0% となる。透明の部分は、RGB を全て通過させるので、 $\frac{200}{3} - \frac{c}{2}\%$, $\frac{200}{3} - \frac{c}{2}\%$, $\frac{200}{3} - \frac{c}{2}\%$ となる。よって、この場合、重ねたシエア画素が通す RGB は、

$$\text{Red} : 0 + 0 + \frac{200}{3} - \frac{c}{2} = \frac{200}{3} - \frac{c}{2}$$

$$\text{Green} : \frac{c}{2} + 0 + \frac{200}{3} - \frac{c}{2} = \frac{200}{3}$$

$$\text{Blue} : \frac{c}{2} + 0 + \frac{200}{3} - \frac{c}{2} = \frac{200}{3}$$

となる。

- Magenta を使って色表現する

この場合、シエア画素を重ねた際に、存在する色は、Magenta, Black と透明である。

この重ねたシエア画素内でそれぞれが占める面積は、

$$\text{Magenta} : \frac{100}{2} \times \frac{m}{100} = \frac{m}{2}$$

$$\text{Black} : 100 \times \frac{1}{3} = \frac{100}{3}$$

$$\text{透明} : 100 - \frac{100}{3} - \frac{m}{2} = \frac{200}{3} - \frac{m}{2}$$

となる。この重ねたシエア画素に RGB100%が当たると、Magenta は、Green を吸収し、Red と Blue を通すので、この Magenta の部分は、RGB の $\frac{m}{2}\%$, 0% , $\frac{m}{2}\%$ を通過させる。Black は、RGB をすべて吸収してしまうので、 0% , 0% , 0% となる。透明の部分は、RGB を全て通過させるので、 $\frac{200}{3} - \frac{m}{2}\%$, $\frac{200}{3} - \frac{m}{2}\%$, $\frac{200}{3} - \frac{m}{2}\%$ となる。よって、この場合、重ねたシエア画素が通す RGB は、

$$\text{Red} : \frac{m}{2} + 0 + \frac{200}{3} - \frac{m}{2} = \frac{200}{3}$$

$$\text{Green} : 0 + 0 + \frac{200}{3} - \frac{m}{2} = \frac{200}{3} - \frac{m}{2}$$

$$\text{Blue} : \frac{m}{2} + 0 + \frac{200}{3} - \frac{m}{2} = \frac{200}{3}$$

となる。

6.1 復元画像の明るさについて

- Yellow を使って色表現する場合、シェア画素を重ねた際に、存在する色は、Yellow, Black と透明である。この重ねたシェア画素内でそれぞれが占める面積は、

$$\begin{aligned}\text{Yellow} &: \frac{100}{2} \times \frac{y}{100} = \frac{y}{2} \\ \text{Black} &: 100 \times \frac{1}{3} = \frac{100}{3} \\ \text{透明} &: 100 - \frac{100}{3} - \frac{y}{2} = \frac{200}{3} - \frac{y}{2}\end{aligned}$$

となる。このシェア画素に RGB100% が当たると、Yellow は、Blue を吸収し、Red と Green を通すので、この Yellow の部分は、RGB の $\frac{y}{2}\%$, $\frac{y}{2}\%$, 0% を通過させる。Black は、RGB をすべて吸収してしまうので、 0% , 0% , 0% となる。透明の部分は、RGB を全て通過させるので、 $\frac{200}{3} - \frac{y}{2}\%$, $\frac{200}{3} - \frac{y}{2}\%$, $\frac{200}{3} - \frac{y}{2}\%$ となる。よって、この場合、重ねたシェア画素が通す RGB は、

$$\begin{aligned}\text{Red} &: \frac{y}{2} + 0 + \frac{200}{3} - \frac{y}{2} = \frac{200}{3}\% \\ \text{Green} &: \frac{y}{2} + 0 + \frac{200}{3} - \frac{y}{2} = \frac{200}{3}\% \\ \text{Blue} &: 0 + 0 + \frac{200}{3} - \frac{y}{2} = \frac{200}{3} - \frac{y}{2}\%\end{aligned}$$

となる。

これら 3 つの RGB をまとめると、以下のようになる。

$$\begin{aligned}\text{Red} &: \frac{200}{3} - \frac{c}{2}\% \\ \text{Green} &: \frac{200}{3} - \frac{m}{2}\% \\ \text{Blue} &: \frac{200}{3} - \frac{y}{2}\%\end{aligned}$$

これを、RGB に置き換えると、

$$\begin{aligned}\text{Red} &: \frac{200}{3} - \frac{100-r}{2} = \frac{r}{2} + \frac{100}{6}\% \\ \text{Green} &: \frac{200}{3} - \frac{100-g}{2} = \frac{g}{2} + \frac{100}{6}\%\end{aligned}$$

6.2 シェア画像の秘密性について (シェア画素が円のとき)

$$\text{Blue} : \frac{200}{3} - \frac{100 - b}{2} = \frac{b}{2} + \frac{100}{6} \%$$

となる。よって、扇面積可変手法の復元画像の値 (r', g', b') を原画像の値 (r, g, b) で表現すると、式 6.3 となる。

$$\begin{pmatrix} r' \\ g' \\ b' \end{pmatrix} = \frac{1}{2} \begin{pmatrix} r \\ g \\ b \end{pmatrix} + \frac{100}{6} \% \quad (6.3)$$

式 6.3 より、色の分散復元手法の復元画像は、原画像に比べて明るさが約 66 % になるということが分かった。

6.1.4 明るさのまとめ

私が提案した手法を明るさについてまとめると、以下のようになる。

$$\text{色の分散復元手法} : \begin{pmatrix} r' \\ g' \\ b' \end{pmatrix} = \frac{1}{2} \begin{pmatrix} r \\ g \\ b \end{pmatrix} \%$$

$$\text{扇面積分割手法} : \lim_{i \rightarrow \infty} \begin{pmatrix} r' \\ g' \\ b' \end{pmatrix} = \begin{pmatrix} r \\ g \\ b \end{pmatrix} \%$$

$$\text{扇面積可変手法} : \begin{pmatrix} r'' \\ g'' \\ b'' \end{pmatrix} = \frac{1}{2} \begin{pmatrix} r \\ g \\ b \end{pmatrix} + \frac{100}{6} \%$$

これらをグラフで示したのが、図 6.1 である。

6.2 シェア画像の秘密性について (シェア画素が円のとき)

3つの手法とも、1ピクセルを扇形に分散している。

色の分散復元手法と扇面積分割手法は、扇形はすべて同じ形の Black, Cyan, Magenta, Yellow の重ね合わせで表現されている。よって、どの扇形をみても、どんな画像で何色を表現しているかは全く分からないようになっている。

6.3 実装について

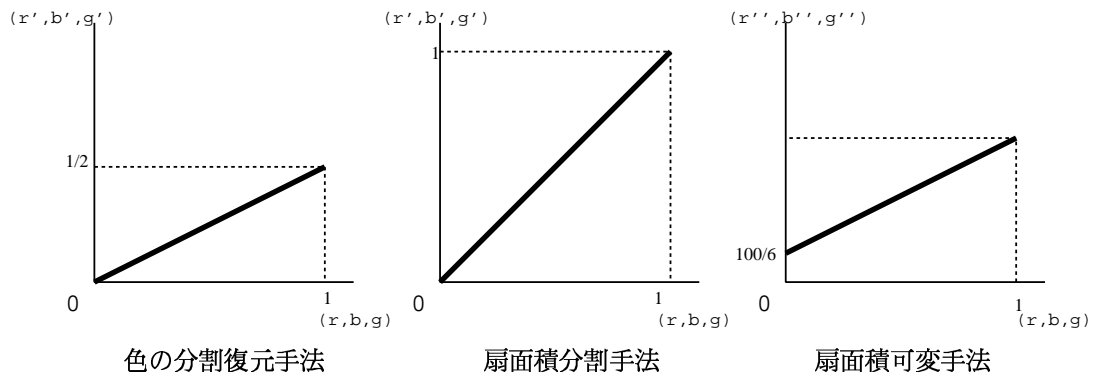


図 6.1 復元画像の明るさを表したグラフ

一方、扇面積可変手法では、Black は 180 度、CMY は 120~180 度の扇形を使用している。そのため、色成分の割合によって、扇形の面積が変化し、表現する色によってシェア画像を見ると原画像の情報が漏れる場合がある。

6.3 実装について

分散復元手法と扇面積可変手法において、1 ピクセルを正方形として実装した際の色の表現パターンと秘密性について検証する。

6.3.1 表現できる色の数について

色の分散復元手法と扇面積可変手法を実装した際、色を何通り表現できるかを検証する。それぞれの実装した手法では、CMY のそれぞれの色に対し、表現パターンを 5 通りしている。よってこれらの組み合わせにより、125 通りの色の表現ができると考えられる。今回、実装した VSSS は、原画像の 1 ピクセルを 9×9 ピクセルまたは、 8×8 ピクセルとし、色画素の固まりをこの範囲内でスライドさせ、色を表現している。この色画素をスライドさせる範囲を大きくなればなるほど、離散でもより任意の細かさで原画像を表現することが可能である。

6.4 考察のまとめ

6.3.2 秘密性について（シェア画素が正方形のとき）

色の分散復元手法では、1 ピクセルを 8×8 ピクセルにし、この範囲内で色を表現している。また、色は 3×8 ピクセルの Black, Cyan, Magenta, Yellow の重ね合わせで表現する。よって、どのシェア画素を見ても 3×8 ピクセルの固まりなので原画像の情報を得ることはできない。

しかし、すべての Black はシェア画素中で同じ場所に位置している。また、色成分の割合によって、シェア画素中での色画素が位置する場所も決定するので、色画素の位置によって、原画像の情報が漏れてしまう。よって、この解決法として、シェア画素を乱数により、シェア画素単位で、90 度、180 度、270 度というふうに回転させた。こうすることにより、色の分散復元手法で実装されたシェア画像を見ただけでは、原画像の情報を得る事は困難になった。

一方、扇面積可変手法では、1 ピクセルを 9×9 ピクセルにし、この範囲内で色を表現している。また、色は $3 \times 8 \sim 4 \times 8$ ピクセルの Black, Cyan, Magenta, Yellow の重ね合わせで表現する。この際、色の分散復元手法と同様に、シェア画素を乱数により回転させても、色成分の割合によって、 3×8 と 4×8 ピクセルの違いがあるため、シェア画像から原画像の情報が漏れてしまう。そこで、さらに復元手法の秘密性を上げるために、Cyan, Magenta, Yellow のシェア画像のシェア画素を乱数により交換するることによって、シェア画素の色によるピクセル数の違いを目立たなくした。

こうすることによって、復元画像の秘密性を上げることに成功した。

6.4 考察のまとめ

考察の結果を表 6.1 にまとめる。

また、3 つの手法のシェア画素の扇形と分散枚数についてまとめた結果が表 6.2 である。

6.4 考察のまとめ

手法	明るさ	秘密性	分散枚数
色の分散復元手法	50%		4
扇面積分割手法	100%		
扇面積可変手法	$\frac{200}{3}\%$		4

表 6.1 考察結果

手法	Black	Cyan	Magenta	Yellow	分散枚数
色の分散復元手法	$\frac{1}{2}$ の扇形	$\frac{1}{2}$ の扇形	$\frac{1}{2}$ の扇形	$\frac{1}{2}$ の扇形	4
扇面積分割手法	$\frac{100}{i+1}$ の扇形	$\frac{100}{i+1}$ の扇形	$\frac{100}{i+1}$ の扇形	$\frac{100}{i+1}$ の扇形	$3i+1$
扇面積可変手法	$\frac{1}{3}$ の扇形	$\frac{1}{3} \sim \frac{1}{2}$ の扇形	$\frac{1}{3} \sim \frac{1}{2}$ の扇形	$\frac{1}{3} \sim \frac{1}{2}$ の扇形	4

表 6.2 シェア画素の扇型と分散枚数

第 7 章

まとめ

従来の視覚復号型秘密分散法は、原画像が白黒画像やある特定の色のカラー画像にしか対応していないものであった。また、それらは復元画像が原画像に比べてかなり暗くなるという傾向があった。

そこで、本研究では、連続的な色表現を可能にする視覚復号型秘密分散法の手法を 3 つ提案した。そして、これらの手法を Java 言語で実装し、検証を行った。その結果、離散ではあるが任意の細かさで連続的な色を視覚復号型秘密分散法を実装することに成功した。

復元した画像の明るさは、第 6 章で示したように原画像の約半分になってしまう。よって、明るさをさらに原画像に近づけるという課題がある。また、扇面積可変手法では、復元画像からどれくらい原画像の情報が漏洩するかについて検証する必要もある。

謝辞

この研究をするに当たって、多くの方々の協力を得ました。とても、深く感謝しています。福本昌弘助教授、福本研究室のみなさまには、代数学の輪講を一緒に勉強させて頂きました。

SSS を研究している舟橋稔仁さんには、パワーポイントの指導からして頂き、大変お世話になりました。

広瀬崇夫さんには、私が困っているときにアドバイスを頂き、いろいろなことを教えていただきました。

田淵理恵さんには、私が原画像に適した絵がなかなかなくて困っているときに、かわいい絵を提供してくれました。そして、西内一馬さんとともに、私が VSSS の試作品を OHP シートに印刷するたびに、その絵を重ねて感動してくれて、私のやる気の源になりました。

正岡元さんには、カラープリンタの設定で息詰まっているときに、いろいろ教えて頂きました。

藤岡里佳さんには、私がプログラミングで悩んでいるときに、一緒に考えてくれて解決してくれました。また、前田明日香さんと澤田明美さんとともに卒論で疲れたときに、話相手となってくれ、とても精神的に癒されました。

澤野充明さんと豊島修平さんには、マシンが固まって私の手には、どうしようもできない時に助けて頂きました。

篠森研究室の平山正治さんには、OHP シートをスキャナで取り込む作業で、篠森研のスキャナの設定がうまくいかず、使えなかったので、島村研にまで連れていって来て、スキャナを使わせて頂きました。

指導教員である菊池豊助教授には、私の VSSS 手法の提案からこの卒論が出来上がるまで色々な指導と、アドバイスを頂きました。

本当にありがとうございました。

参考文献

- [1] M. Naor and A. Shamir. Visual cryptography. In *EUROCRYPT'94, LNCS950*, pp. 1–12, 1995.
- [2] A. Shamir. How to share a secret. In *Commun. fo the ACM*, Vol. 22, pp. 612–613, 1979.
- [3] H. Koga and H. Yamamoto. Proposal of a lattice-based visual secret sharing scheme for color and gray-scale images. In *IEICE Trans.*, pp. 1262–1269, 1998.
- [4] 今井秀樹. 現代暗号とマジックプロトコル. サイエンス社, 2000.

付録 A

扇形の重ね方によって視覚が認識する色が変わらないことの証明

扇形の重ね合わせで色を表現する際、人間が認識する色が表現するピクセルの扇形の重ね方によらないという事を証明する。

1 ピクセルの面積を 100、Cyan、Magenta、Yellow の面積をそれぞれ c , m , c とおく。それぞれの円に RGB がそれぞれが 100 %の光が当たった場合を考える。

A.1 1 ピクセルを 2 色で表現するとき

まず始めに、VSSS で 1 ピクセルの色を、2 色で表現する場合を考える。そのときの扇形の重ねる順序は 2 通りある。以下は、Cyan と Magenta で表現される色を例に挙げている。

A.1.1 Cyan と Magenta が重ならない場合

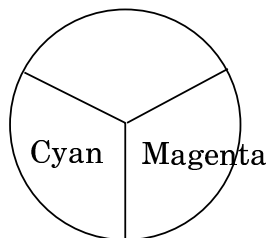


図 A.1 Cyan と Magenta が重ならない場合

A.1 1ピクセルを2色で表現するとき

Cyan は Red を吸収し、Green と Blue を通すので、この Cyan 部分は RGB の 0% , $c\%$, $c\%$ を通過させる。

Magenta は Green を吸収し、Red と Blue を通すので、この Magenta 部分は RGB の $m\%$, 0% , $m\%$ を通過させる。

透明は RGB すべてを通すので、透明な部分は RGB の $100 - c - m\%$, $100 - c - m\%$, $100 - c - m\%$ を通過させる。

よって、この1ピクセルが通過させる RGB は、

$$\text{Red} : 0 + m + (100 - c - m) = 100 - c\%$$

$$\text{Green} : c + 0 + (100 - c - m) = 100 - m\%$$

$$\text{Blue} : c + m + (100 - c - m) = 100\%$$

となるので、以下のようなになる。

$$\begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} 100 - c \\ 100 - m \\ 100 \end{pmatrix} \% \quad (\text{A.1})$$

A.1.2 Cyan と Magenta が重なる場合

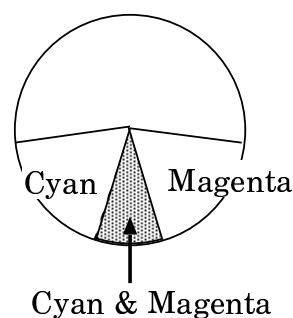


図 A.2 Cyan と Magenta が重なる場合

Cyan と Magenta が重なる部分の面積を t とおく。Cyan は Red を吸収し、Green と Blue を通すので、この Cyan 部分は RGB の 0% , $(c - t)\%$, $(c - t)\%$ を通過させる。

A.2 1 ピクセルを 3 色で表現するとき

Magenta は Green を吸収し、Red と Blue を通すので、この Magenta 部分は RGB の $(m - t) \%$, 0% , $(m - t) \%$ を通過させる。

Cyan と Magenta が重なり合っている部分は、Red と Green を吸収し、Blue を通す。この部分は、 0% , 0% , $t \%$ を通過させる。

透明の部分の面積は、 $100 - c - m + t$ となる。透明は、RGB すべてを通すので、透明な部分は、RGB の $100 - c - m + t \%$, $100 - c - m + t \%$, $100 - c - m + t \%$ を通過させる。

よって、この 1 ピクセルが通過させる RGB は、

$$\text{Red} : 0 + (m - t) + (100 - c - m + t) = 100 - c\%$$

$$\text{Green} : (c - t) + 0 + (100 - c - m + t) = 100 - m\%$$

$$\text{Blue} : (c - t) + (m - t) + t + (100 - c - m + t) = 100\%$$

となるので、以下のようになる。

$$\begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} 100 - c \\ 100 - m \\ 100 \end{pmatrix} \% \quad (\text{A.2})$$

A.1、A.2 よりどんな重ね方をしても、1 ピクセルを RGB がそれぞれ 100 % の光を通す割合は同じということが言える。

A.2 1 ピクセルを 3 色で表現するとき

次に、VSSS で 1 ピクセルの色を、3 色で表現する場合を考える。そのときの扇形の重ね方は 3 通りある。

A.2.1 3 色すべてが重ならない場合

Cyan は Red を吸収し、Green と Blue を通すので、この Cyan 部分は RGB の 0% , $c \%$, $c \%$ を通過させる。

Magenta は Green を吸収し、Red と Blue を通すので、この Magenta 部分は RGB の m

A.2 1ピクセルを3色で表現するとき

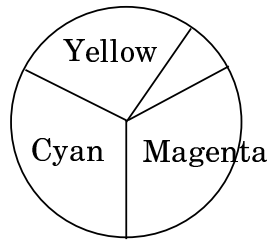


図 A.3 3色すべてが重ならない場合

$c\%$, 0% , $m\%$ を通過させる。

Yellow は Blue を吸収し、Red と Green を通すので、この Yellow 部分は RGB の $y\%$, $y\%$, 0% を通過させる。

透明は RGB すべてを通すので、透明な部分は RGB の $100 - c - m - y\%$, $100 - c - m - y\%$, $100 - c - m - y\%$ を通過させる。

よって、この1ピクセルが通過させる RGB は、

$$\text{Red} : 0 + m + y + (100 - c - m - y) = 100 - c\%$$

$$\text{Green} : c + 0 + y(100 - c - m - y) = 100 - m\%$$

$$\text{Blue} : c + m + 0 + (100 - c - m - y) = 100 - y\%$$

となるので、以下のようになる。

$$\begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} 100 - c \\ 100 - m \\ 100 - y \end{pmatrix} \% \quad (\text{A.3})$$

A.2.2 2色が重なる場合

このとき、Cyan と Magenta が重なる場合を考える。

Cyan と Magenta が重なる部分の面積を t とおく。Cyan は Red を吸収し、Green と Blue を通すので、この Cyan 部分は RGB の 0% , $(c - t)\%$, $(c - t)\%$ を通過させる。

Magenta は Green を吸収し、Red と Blue を通すので、この Magenta 部分は RGB の $(m - t)\%$, 0% , $(m - t)\%$ を通過させる。

A.2 1 ピクセルを 3 色で表現するとき

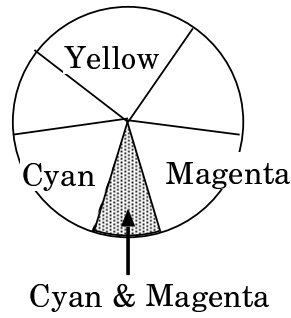


図 A.4 2 色が重なる場合

Yellow は Blue を吸収し、Red と Green を通すので、この Yellow 部分は RGB の $y\%$, $y\%$, 0% を通過させる。

Cyan と Magenta が重なり合っている部分は、Red と Green を吸収し、Blue を通す。この部分は、 0% , 0% , $t\%$ を通過させる。

透明の部分の面積は、 $100 - c - m - y + t$ となる。透明は、RGB すべてを通すので、透明な部分は、RGB の $100 - c - m - y + t\%$, $100 - c - m - y + t\%$, $100 - c - m - y + t\%$ を通過させる。

よって、この 1 ピクセルが通過させる RGB は、

$$\text{Red} : 0 + (m - t) + y + 0 + (100 - c - m - y + t) = 100 - c\%$$

$$\text{Green} : (c - t) + 0 + y + 0 + (100 - c - m - y + t) = 100 - m\%$$

$$\text{Blue} : (c - t) + (m - t) + 0 + t + (100 - c - m - y + t) = 100 - y\%$$

となるので、以下のようになる。

$$\begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} 100 - c \\ 100 - m \\ 100 - y \end{pmatrix} \% \quad (\text{A.4})$$

A.2.3 3 色が重なる場合

Cyan と Magenta が重なる部分の面積を t とおく。Magenta と Yellow が重なる部分の面積を s とおく。Cyan は Red を吸収し、Green と Blue を通すので、この Cyan 部分は

A.2 1 ピクセルを 3 色で表現するとき

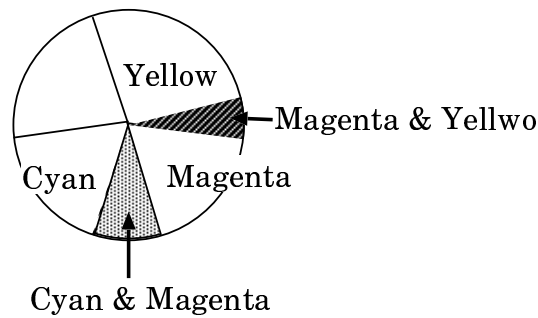


図 A.5 3 色が重なる場合

RGB の 0% , $(c - t)\%$, $(c - t)\%$ を通過させる。

Magenta は Green を吸収し、Red と Blue を通すので、この Magenta 部分は RGB の $(m - t - s)\%$, 0% , $(m - t - s)\%$ を通過させる。

Yellow は Blue を吸収し、Red と Green を通すので、この Yellow 部分は RGB の $(y - s)\%$, $(y - s)\%$, 0% を通過させる。

Cyan と Magenta が重なり合っている部分は、Red と Green を吸収し、Blue を通す。この部分は、 0% , 0% , $t\%$ を通過させる。

Magenta と Yellwo が重なり合っている部分は、Green と Blue を吸収し、Red を通す。この部分は、 $s\%$, 0% , 0% を通過させる。

透明の部分の面積は、 $100 - c - m - y + u + s + t$ となる。透明は、RGB すべてを通すので、透明な部分は、RGB の $100 - c - m - y + t + s\%$, $100 - c - m - y + t + s\%$, $100 - c - m - y + t + s\%$ を通過させる。

よって、この 1 ピクセルが通過させる RGB は、

$$\text{Red} : (m - t - s) - (y - s) + s + (100 - c - m - y + t + s) = 100 - c\%$$

$$\text{Green} : (c - t) + (y - s) + u + (100 - c - m - y + t + s) = 100 - m\%$$

$$\text{Bluel} : (c - t) + (m - t - s) + t + (100 - c - m - y + t + s) = 100 - y\%$$

A.2 1 ピクセルを 3 色で表現するとき

となるので、以下のようになる。

$$\begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} 100 - c \\ 100 - m \\ 100 - y \end{pmatrix} \% \quad (\text{A.5})$$

A.3、A.4、A.5 より、3 色の場合も、どんな重ね方をしても、1 ピクセルを RGB がそれぞれ 100 %の光を通す割合は同じということが言える。よって、人間が認識する色は、扇形の重ねる順序によらないと言う事が証明できた。