

# 単眼視法を用いた球体の絶対座標エンコーダの設計

## Design of an Absolute-Encoder for a Sphere using a Monocular Vision

四戸弘典\*, 工藤隆男\*

Hironori Shinohe\*, Takao Kudoh\*

\*八戸工業高等専門学校

\*Hachinohe National College of Technology

キーワード： 単眼視法 (A Monocular Vision Method), 絶対座標エンコーダ (Absolute-Encoder), 球体 (Sphere)

連絡先： 〒039-1192 青森県八戸市田面木字上野平16-1 八戸工業高等専門学校

工藤隆男, Tel.: (0178)27-7279, Fax.: (0178)27-7279, E-mail: tkudoh-e@hachinohe-ct.ac.jp

### 1. はじめに

エンコーダの用途はFA分野から計測,OA機器,医療機器,福祉分野へと拡大され,自動制御などにおいて必要不可欠である. 特に,絶対座標を出力できるエンコーダは,原点復帰の必要が無く,電源投入時に即始動できるという利点がある. しかしながら,回転軸にスリットなどを有する機械式絶対座標エンコーダはその構造が複雑であり,特に多軸エンコーダの場合には小型化のボトルネックになりがちであるという問題がある.

本稿では,シンプルな構造の絶対座標ロータリーエンコーダ開発のために,1台のカメラで捉えた球体表面の模様を3次元座標を元にXYZ軸周りの回転角度を復元する方法とその実験結果について報告する. 提案するエンコーダの特徴は,球表面の模様のみを入力とする事から,構造をきわめてシンプルにでき,トラックボールや,超音波モータと球体を組み合わせるシンプルなサーボモータなど多くの応用を期待できる. 球表面の模様をカメラで

捉え,その回転角度を復元するためには,模様の3次元座標が必要になる. 一般に3次元座標の取得は複眼カメラを用いた2枚の画像の対応点探索により行うが,計算量が多くなることが問題である. 対応点探索を不要にするためには,映像を1枚だけ用いればよい. すなわち,単眼視法<sup>1)</sup>であるが,この場合,奥行き座標を復元するためには,なにかしらの制約条件が必要になる. 本稿では,単眼視法により奥行き座標を復元するために,球体の実直径とその映し出された球体の直径の比を用いる.

次に,球体表面の模様の個数が多くなると,その識別が困難になることが懸念される. 本稿では,球体の回転角度に対する模様の移動量の比に基づき,球体に付ける模様の間隔の最適な配置について評価をしている. この評価結果を元に球体に模様を付け,実際にカメラで撮影を行い,球体を一方向に回転させた時の回転角度を6%程度の誤差で復元できることを確認した.

## 2. 提案する球体エンコーダ

Fig.1において,球体表面の模様を撮影することで3次元の絶対座標を得ることを考える.

模様を等間隔に配置する時,少なくとも1個以上がカメラ側にあるようにするため,各軸周りに対する模様の個数は3個以上とし,各模様の色を変えるものとする.

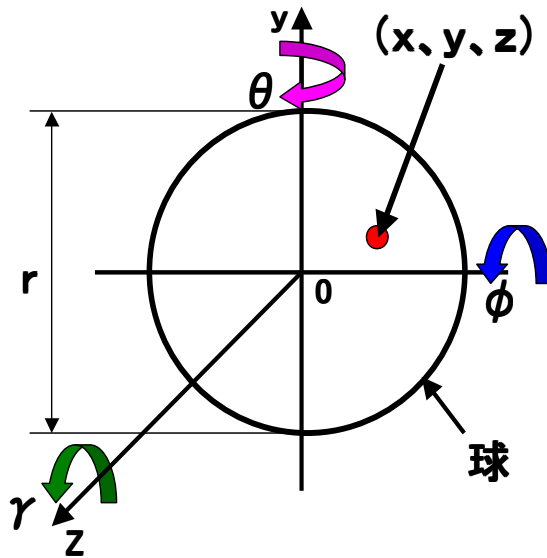


Fig. 1 球体エンコーダのモデル図

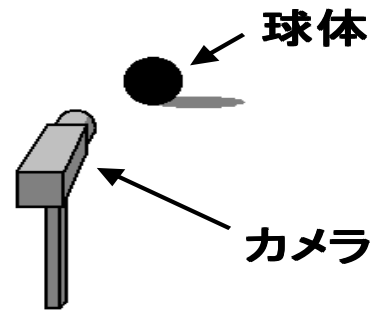


Fig. 2 単眼視法の例

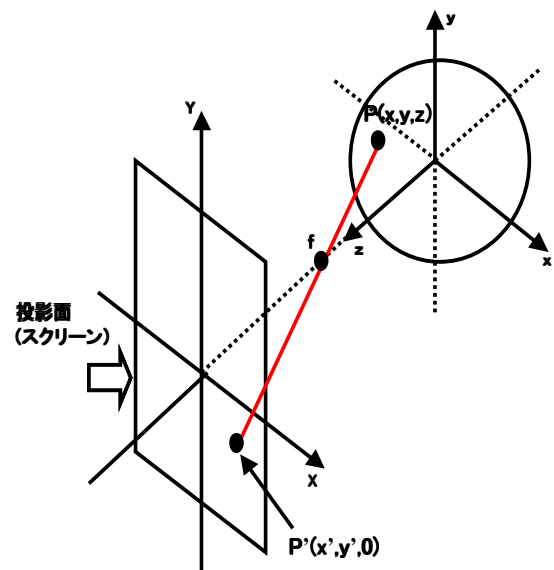


Fig. 3 カメラの投影面に写る情報

## 3. 単眼視法

単眼視法とは, Fig.2のように, 1つのカメラで物体を捉える方法である. この特徴としては, 原理が単純であり, 複数のカメラを用いるより計算量は少ない. また, 複数のカメラを用いる場合に比べ, 構造を小型化できる.

本研究に適応させる上で, 一番のネックになる部分は, Fig.3のように球の3次元座標のうち, 制約なしには奥行き座標を復元できないという問題がある. そこで, 画面上の直径と実直径の比を用いて復元することにする.

## 4. 球のz座標の導出方法

奥行き座標Zの導出方法を考える.

まず, 球の検出により, パソコンの画面上での球の直径と実際の球の直径の比を導出する.

$$\lambda = \frac{R}{(r/567)} \quad (1)$$

twip(Twentieth of a Point):画面上の単位

$$1 [\text{cm}] = 567[\text{twip}]$$

r : 画面上での直径

R : 実直径

: 画面上の直径と実直径の比

次に、パソコンの画面上での座標では、実際の座標とは異なってしまふ。そこで、式(1)で導出したを用いて、

$$X = x/567 \times$$

$$Y = y/567 \times$$

x:画面上でのx座標

y:画面上でのy座標

とすることで、実際の座標を検出することができる。球の奥行き座標Zを算出するためには、以下の式を用いばよい。

$$Z = \sqrt{(R/2)^2 - (X^2 + Y^2)} \quad (2)$$

## 5. 回転角度の算出

球体表面の模様の変位量を用い、球体の回転角度 { , } を復元する方法を考える。

まず、式(1)の を用い、球体表面の模様座標を求め。

$$X = P_x/567 \times$$

$$X' = P_x'/567 \times$$

P<sub>x</sub>:画面上での初期位置のx座標

P<sub>x</sub>':画面上でのあるターゲットが移動した後のx座標

$$Y = P_y/567 \times$$

$$Y' = P_y'/567 \times$$

P<sub>y</sub>:画面上での初期位置のy座標

P<sub>y</sub>':画面上でのあるターゲットが移動した後のy座標

$$Z = \sqrt{(R/2)^2 - (X^2 + Y^2)}$$

$$Z' = \sqrt{(R/2)^2 - (X'^2 + Y'^2)}$$

Z:初期位置のz座標

Z':ターゲットが移動した後のz座標

求められたそれぞれの座標を用いて、 , , 方向のそれぞれに回転させた時の回転角度の導出方

法を以下に示す。

・Y軸周りに 回転させた場合(Fig.4)

$$\theta' = \text{Cos}^{-1} \frac{X}{\sqrt{X^2 + Z^2}}, \quad \theta'' = \text{Cos}^{-1} \frac{X'}{\sqrt{X'^2 + Z'^2}}$$

$$\theta = \theta'' - \theta'$$

':初期位置の模様の角度

': 回転した後の模様の角度

・X軸周りに 回転させた場合(Fig5)

$$\phi' = \text{Cos}^{-1} \frac{Y}{\sqrt{Y^2 + Z^2}}, \quad \phi'' = \text{Cos}^{-1} \frac{Y'}{\sqrt{Y'^2 + Z'^2}}$$

$$\phi = \phi'' - \phi'$$

':初期位置の模様の角度

': 回転した後の模様の角度

・Z軸周りに 回転させた場合(Fig6)

$$\gamma' = \text{Cos}^{-1} \frac{X}{\sqrt{X^2 + Y^2}}, \quad \gamma'' = \text{Cos}^{-1} \frac{X'}{\sqrt{X'^2 + Y'^2}}$$

$$\gamma = \gamma'' - \gamma'$$

':初期位置の模様の角度

': 回転した後の模様の角度

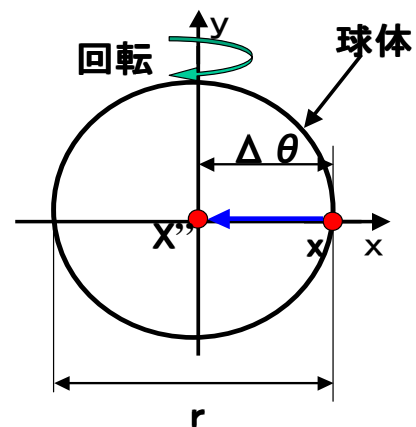


Fig. 4 回転させた時の模様の変位

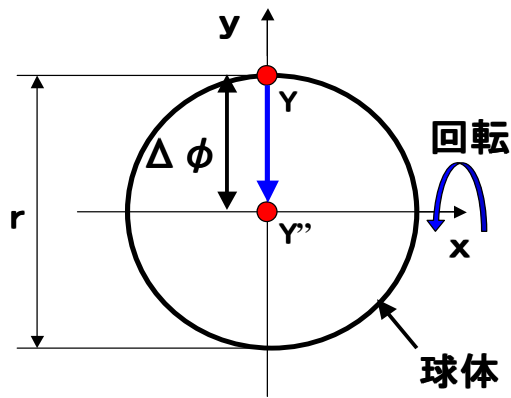


Fig. 5 回転させた時の模様移動

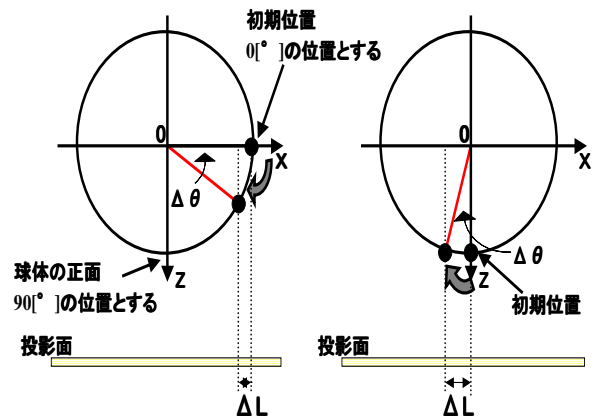


Fig. 7 模様を各位置から L 回転させた場合

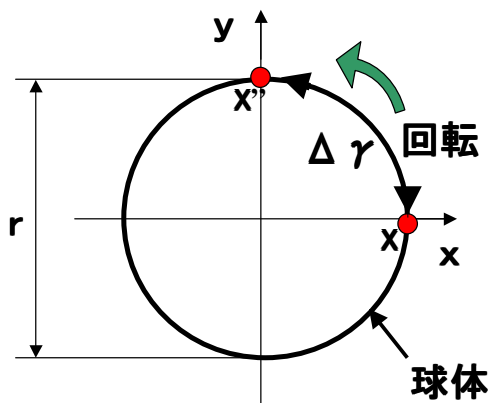


Fig. 6 回転させた時の模様移動

## 6. 模様の配置

模様を等間隔に配置する時、少なくとも1個以上がカメラ側にあるようにするためには、各軸回りに対して、3個以上の模様が必要である。一方、模様を無数に配置する場合、個々の模様の識別が困難になる。そこで、適正な模様間隔を見積もる。

ある一定間隔の長さ  $L$  回転させて、一番復元精度が良い座標をシミュレーションにより求める。この場合、ある一定間隔の長さ当たりの回転角度  $\Delta\theta / L$  が小さい時、一番復元精度が良いものとする。

### 6.1 復元精度の見積もり

球の直径を2[cm]とした場合、模様を回転させた時の長さ  $L = 0.5, 0.2, 0.1$ [cm]の3パターンに対する  $\Delta\theta / L$  をFig.8に示す。この図より、球体表面の中心付近にある模様を捉えることが、最も復元精度が良いことが分かる。

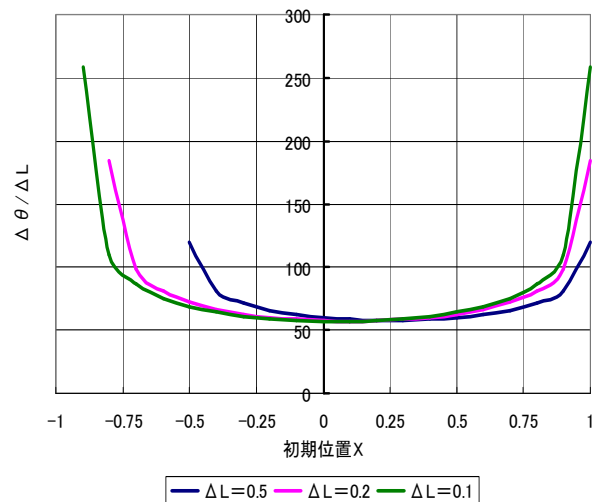


Fig. 8  $L$ を一定にして回転させた時の  $\Delta\theta / L$  (直径2 cm)

プログラムで球体表面の模様を探索する時は、復元精度を上げるために、中心に最も近い模様を探索する必要がある。そこで、Fig.9のように球体表面の中心から探索を行う。

よって、何度回転しても常に球体の中心付近にあ

るためには,球体表面にたくさんの模様を付ければよい. 復元精度を上げられる. しかし,模様の識別を色により行う場合,近似色での着色を余儀なくされる. このことから,色認識の誤りが懸念される. 以上のことから,Fig.8に基づき,適正の模様間隔を考える. Fig.8より, $+0.5 \sim -0.5$ の範囲で  $\gamma / L$  はほぼ一定で復元精度が良いことから, 球体表面に模様を  $60[^\circ]$  間隔に 6 点付けるものとする.

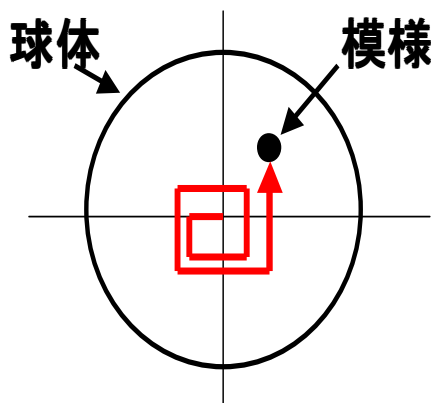


Fig. 9 中心付近に最も近い模様を検出する方法

## 6.2 探索方法

球体表面の中心から探索していき,中心に最も近い模様を捉える方法だけでは問題がある. Fig.10(a)のように初期位置の模様が球体表面の中心にあり, Fig.10(b)のように球体を  $\gamma$  方向に回転させた場合,その模様の回転量を得ることができない.

そこで,もう 1 点の模様を捉えるために,最も外側にある模様を捉えることにする. 模様の探索方法としては, Fig.11 のように球体表面の中心から探索する方法とは逆に, 球体表面の外側から探索する.

すなわち,球体表面の中心付近と外側の両方を探索し, 2 点の模様を捉え,回転量を復元することにする.

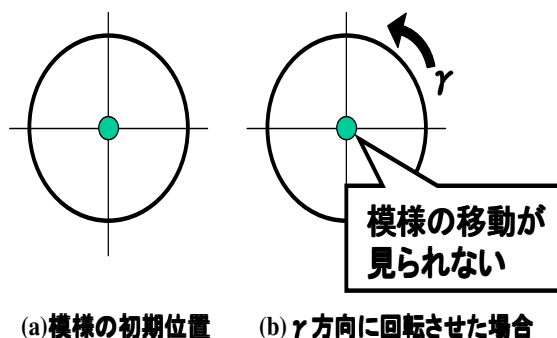


Fig. 10 問題点

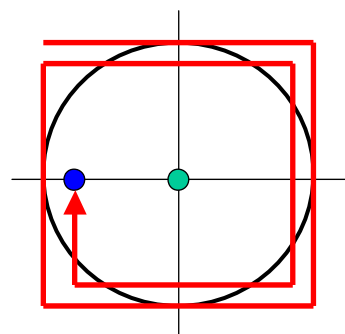


Fig. 11 探索方法

## 7. エンコーダの評価

カメラで球体を撮影し,球体の回転量を復元することの評価を行う. 方向の回転のみに着目し,評価を行う. 模様の付け方は, $60[^\circ]$  間隔に Fig.12 のように行う. 評価システムは,Visual Basic を用い,作成した. Fig.13 に作成した評価システムの実行例を示す.



Fig. 12 使用する球体と模様の配置

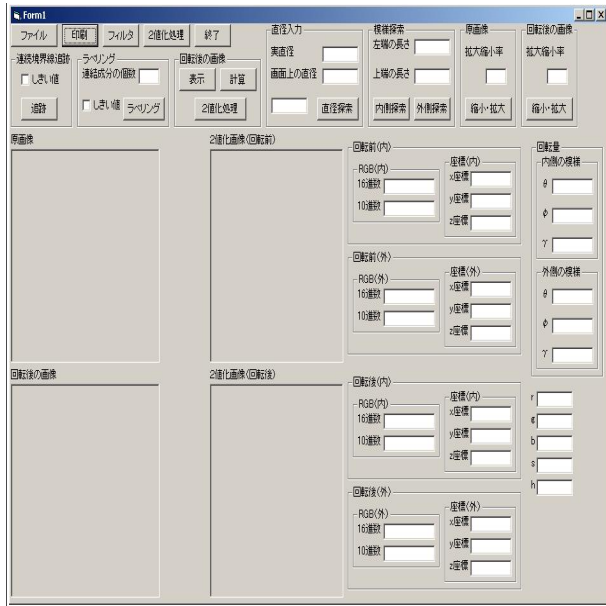


Fig. 13 評価システム

## 7.1 明るさの影響とその解決方法

実画像の色成分には、背景の黒色、球体の白色、各模様の有彩色が混在することから、有彩色に基づく2値化処理が望ましい。有彩色識別のためにRGBを用いたところ、光の明るさや影などによって球体や模様の色が各面で変わってしまい、2値化処理が不完全であるという問題があった。そこでこの解決方法として、RGB色空間を明るさに影響されないHLS色空間に変換することで解決する。

## 7.2 HLS色空間

HLS色空間とは、色相 (Hue)、明度 (Light)、彩度 (Saturation) の3つ要素からなるものである。それぞれの要素は以下の通りである。

- ・色相：R,G,Bなどの色の種類を表す要素。
- ・明度：明るさを表す要素。または輝度もいう。
- ・彩度：色の鮮やかさを表す要素。無彩色（黒、灰色、白）では0となる。 これらを色の3属性という<sup>3)</sup>。

明度の影響をなくするために、色相と彩度を使う。すなわち、色相によって各色の明度や彩度の違いを吸収した。

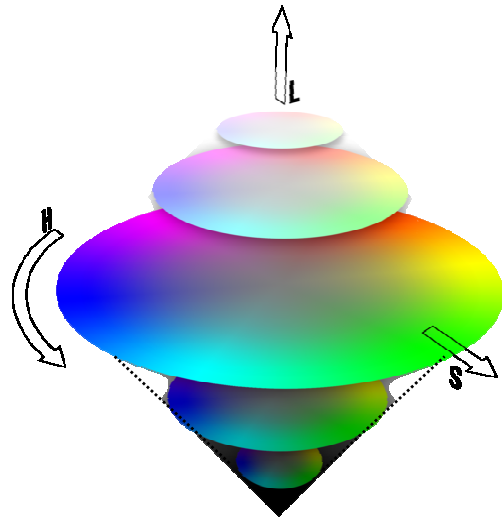


Fig. 14 HLS色空間<sup>4)</sup>

次に、背景はほぼ黒、球体はほぼ白の無彩色であり、模様は有彩色である。無彩色と有彩色の特徴に基づき、模様と模様以外を分離するために彩度を用いる。背景と球体を識別するために、RGBに制約を持たせ、背景を黒、球体を白に判断させて変換する。

その結果をFig.15に示す。

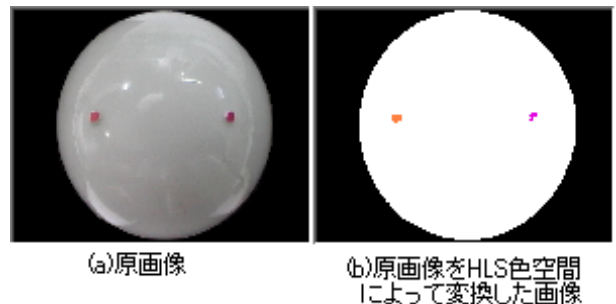


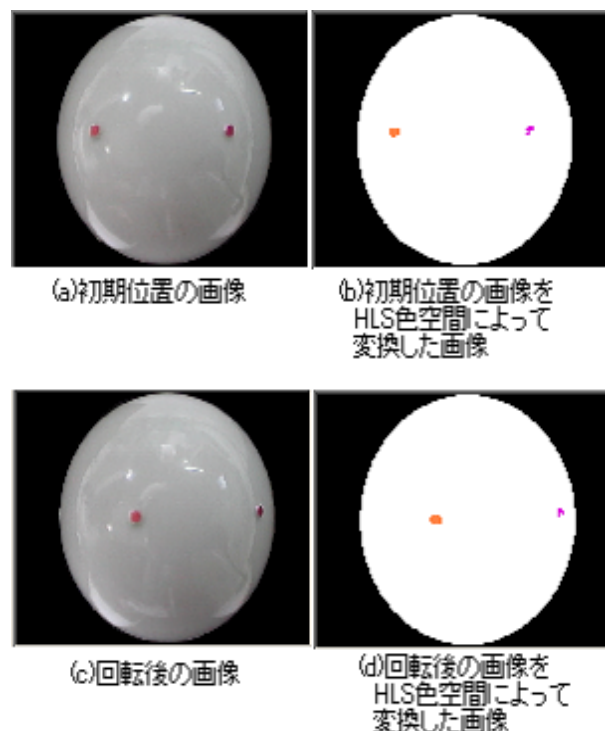
Fig. 15 HLS色空間によって変換した結果

Fig.15より、背景、球体、模様を識別することができたが、模様の形が若干歪んでいる。

## 7.3 測定結果

球体を回転させた時の復元された回転量を測定した。その結果をFig.16に示す。Fig.16は、球体を右回転させた時の値である。回転量の結果は、マイナスを示している。これは、左回転をプラスとしているからである。内側と外側の模様を探索した結

果の回転量の誤差は、約 6 % である。



		内側(マゼンタ)	外側(オレンジ)
回 転 前	x	1.042108987	-1.243807502
	y	0.235314932	0.16808209
	z	1.467617548	1.311471248
回 転 後	x	1.546355272	-0.571479122
	y	0.235314933	0.134465676
	z	0.921237093	1.717729933
$\theta$		-23.3788471	-24.90053796

Fig. 16 測定結果

#### 7.4 考察

内側の模様の回転量と外側の模様の回転量が異なる理由について考える。復元された模様の歪み具合が回転位置によって異なっている。この歪み具合が、取得する座標値の誤差の原因となることから、内側の模様の回転量と外側の模様の回転量が異なるものと思われる。すなわち、歪みの原因の1つは、模様の周辺部と中心部における色成分が一定ではないことであると考えられる。

## 8. まとめ

シンプルな構造の多軸ロータリエンコーダの開発を目的とし、単眼カメラを用いて球体の絶対座標の復元を行った。最適な模様の配置間隔は各軸周りに60[°]間隔が望ましいとの結論を得た。そこで、実際に球体表面に模様を付け、カメラからの出力画像より回転量を読み取ったところ、外側の模様と内側の模様の回転量の復元誤差は、約6%であった。この誤差は、模様の周辺部と中心部の色合いの違いに起因するものと思われる。

今回の結果から、提案する方法により、回転角度を復元できることを確認できたので、今後は精度向上などの改善を行いたい。

## 謝辞

本研究テーマは、多摩川精機株式会社 八戸事業所 浜信治氏のご助言によるものであることを、ここに記し、謝意を表します。

## 参考文献

- 1) 井口征士, 佐藤宏介 三次元画像計測, 17/17, 昭晃堂(1994)
- 2) K.S.Fu, R.C.Gonzalez, C.S.G.Lee, ROBOTICS Control, Sensing, Vision, and Intelligence, 307/328, MCGraw-Hill Book Company(1987)
- 3) 酒井幸市: デジタル画像処理入門, 101/105, CQ出版社(2004)
- 4) [http://en.wikipedia.org/wiki/HLS\\_color\\_space](http://en.wikipedia.org/wiki/HLS_color_space)