画像工学特論講義ノート：線形代数で3次元復元

玉木 徹

平成21年11月25日

まえがき

本書は、広島大学大学院工学研究科情報工学専攻における画像工学特論で講義した内容をもとに執筆した。コンピュータビジョンに関連する書籍は多数あるが、学生が自習するために詳しく書かれているテキストはほとんどないため、講義終了後に内容を整理し、web 公開のためにまとめなおした。

本書はまだ執筆途中であり、校正途中であり、つまりまだ書きかけである。しかし、締め切りのない執筆はいつまでも続くライフワークのようなものである。どこかで区切りをつけなければ、まとまるものもまとまらない。

内容に関するご意見や間違いの修正など、フィードバックをいただければ幸いである。

目 次

図 目 次

表 目 次

第1章　線形代数で3次元復元

デジタルカメラで撮影した画像も、あなたの眼球の中の網膜に映っている映像も、この地球上の3次元世界を写し撮ったものである。つまりは、2次元世界に投影された、凸凹もない、奥行きもない、平たい板に色を塗ったようなものである。

しかし私たちは、サイコロを立方体だと感じるし、バレーボールのような球をホットケーキのような円盤と間違えることは、ない。コロラドの広大な風景を写し撮った写真を見てその壮大な空間を感じ取ることもあるし、自分の子供が走り回るビデオをみて庭の狭さを実感することもある。

平たい2次元映像から3次元空間の立体感を復元すること。それがコンピュータビジョンにおける3次元復元 (three-dimensional reconstruction) である。3次元復元にはいくつかの種類がある。一つ目は、1枚の写真が与えられただけで3次元の形状を推定するもの。これは single view geometry と呼ばれる。二つ目は、2枚の写真が与えられる場合。これは stereo もしくは epolplar geometry と呼ばれる。三つ目は、2枚以上の写真が与えられる場合。これは多視点幾何 (mutiple view geometry) と呼ばれる。

それぞれ利点欠点があり、前提とする過程も違えば計算方法も異なる。背景となる数学的知識は膨大であり、プログラミングで実装するのも複雑である。自分の目で確かめたければ、有名な Hartley と Zisserman の青い本 [4] や Ma らの黄色い本 [5] を手に取ってみるといい。そのほかにも、Forsyth と Ponce の本 [2] やその日本語版 [3]、教科書的な本 [9, 8, 7]、ハンドブック [6, 10] などなど。これらの本を手にした初学者は、まずその本の分厚さにたじろぎ、次にその中の数式に頭がクラクラする。険しい数学という山道を登っていかなければ、3次元復元という頂上にたどり着けないのだろうか？

じつは、必要なほとんどの知識は、大学1年生で習う線形代数で十分なのだ。もちろん厳密にやろうとすれば、複雑な非線形最適化手法も必要になってくるし、位相幾何の知識も要求される。しかしそこまで必要なのは、実際に手法の中身を知る必要のある研究をしている場合である。知識を吸収する必要があれば分厚い専門書を眺めればいいし、実装する必要があれば公開されているライブラリを使ってもいい。3次元復元の基本は、線形代数を知っていれば理解できる。必要があれば、そのあとさらに進めばいい。

とりあえず3次元復元をやってみたい。ここでは線形代数だけを使って、3次元復元という頂上まで一直線に登ってみたいと思う。最初に、線形代数のおさらいから始めよう。必要なら、自分が勉強した線形代数の教科書を本棚や段ボール箱からひっぱり出してきて、手元において参照してほしい。次に、3次元世界を2次元画像に投影することを数式で表わしてみる。数式に惑わされずに、中学校の理科で習ったレンズやピンホールカメラをイメージして、いったい何をやっているのかを忘れないように進んでほしい。それから、カメラの校正方法をマスターする。最初は、校正とは何者なのか、なんでそれをする必要があるのか、すぐには理解できないかもしれない。しかし何度も繰り返し読んでいくうちに、その大切さに気がつくだろう。最後に、ようやく3次元復元にと

りかかる。ここでは重要でややこしい行列がたくさん登場するので、迷子にならずに、なぜそれで 2 次元から 3 次元が出てくるのかを、理解していこう。

1.1　ベクトル・座標系・座標変換：おさらい

ここでは 2 次元と 3 次元のベクトル、座標系、座標変換についてまとめておく。多くのことは線形代数の教科書に書いてあることと同じであるが、3 次元空間を扱うときに必要な特殊な記法や表現もあるので、注意してほしい。以下の説明は Ma らの黄色い本 [5] に沿っているので、より詳しい説明が必要な場合にはそちらを参照してもらいたい。

1.1.1　3 次元幾何

3 次元空間中に点 (point) p があるとしよう。この 3 次元空間の 3 つの軸をそれぞれ X, Y, Z 軸として、点 p の座標 (coordinates) を次のように表すことができる。

$$\boldsymbol{X} \equiv \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \in \mathbb{R}^3 \tag{1.1}$$

ここで $\boldsymbol{X}$ は、XYZ 座標系 (coordinate system) における点 p の位置を表す座標である。行列の転置記号を使って $\boldsymbol{X} = (X, Y, Z)^T$ と書くときもある。

点 p と点 q がある場合、それらを結ぶベクトル (vector) v は次のように表わされる。

$$v = \boldsymbol{Y} - \boldsymbol{X} \in \mathbb{R}^3 \tag{1.2}$$

ここで $\boldsymbol{X}, \boldsymbol{Y}$ はそれぞれ点 p, q の座標である。ベクトルは、2 つの座標の相対的な位置関係（方向と長さ）を表している。

線形代数で習ったように、ベクトル同士は足すこともできるし、長さを何倍かすることもできる。つまり、二つのベクトル $u, v \in \mathbb{R}^3$ があれば、その線形和 (linear combination)

$$au + bv \in \mathbb{R}^3, \quad a, b \in \mathbb{R} \tag{1.3}$$

もまたベクトルである。これをベクトル空間の線形性という。注意したいのは、ベクトルを足したり引いたりすることはできても、座標を足したり引いたりすることはしない、ということである。座標は位置を表す番地のようなものである。2 丁目 23 番地と 5 丁目 11 番地を足した 7 丁目 34 番地にはいったい何があるというのだ？

二つのベクトル $u, v \in \mathbb{R}^3$ の内積 (inner product) を定義すると、ベクトル同士の角度を測ることができるようになる。よく用いられている標準内積という内積は次の式で与えられる。

$$< u, v > \equiv u \cdot v \equiv u^T v = u_1 v_1 + u_2 v_2 + u_3 v_3 \tag{1.4}$$

ここで

$$u \equiv \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix},\ v \equiv \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} \in \mathbb{R}^3 \tag{1.5}$$

この定義から当然であるが、u と v を入れ替えても同じである。つまり、

$$u \cdot v = v \cdot u \tag{1.6}$$

である。

ベクトルの長さ（ノルム (norm) ともいう）は、内積を使って定義することが多い。あるベクトル v のノルムは

$$||v|| \equiv \sqrt{u \cdot v} \tag{1.7}$$

と定義する。書き方を変えれば、

$$||v||^2 \equiv u \cdot v = u^T v \tag{1.8}$$

である。もちろんこの定義以外にもノルムは存在するが、それらはこの先必要ないので、興味があれば数学の教科書を参考してほしい。

内積を定義すれば角度が定義できる。二つのベクトル u, v が直角である（あるいは直交している、なす角度が 90 度である）ということは、内積が 0 であると定義される。つまり、

$$u \perp v \Longleftrightarrow u \cdot v = 0 \tag{1.9}$$

である。3 次元空間や 2 次元空間では、ベクトル同士が直角かどうかは見てすぐにわかるだろう。だから、なぜこのような直角の定義が必要なのか、最初は不思議に思うかもしれない。しかし「ベクトル」とは、高校の数学で習ったような「矢印」だけで理解できるものではない。3 次元のベクトルと同じように、4 次元や 5 次元や 100 次元の空間にもベクトルは定義できるが、100 次元空間の矢印同士が直角であるかどうか、想像できるだろうか？　あるいは、三角関数の $\sin x$ と $\cos x$ はベクトルであって、それらがなす角度が直角であると言われても、理解できるだろうか？　では $\sin x$ と $\sin 2x$ がなす角度は何度か？　それらを理解するには、内積による角度の定義が必要なのだ。ある二つのベクトル u, v がなす角度を θ とすると、

$$u \cdot v = ||u||\ ||v|| \cos\theta \tag{1.10}$$

である。この公式は内積と角度を結びつける重要なものなので、覚えておいてほしい。

内積があれば外積もある。3 次元空間のベクトル u, v の外積 (cross product) は、次の式で定義される。

$$u \times v \equiv \begin{pmatrix} u_2 v_3 - u_3 v_2 \\ u_3 v_1 - u_1 v_3 \\ u_1 v_2 - u_2 v_1 \end{pmatrix} \in \mathbb{R}^3 \tag{1.11}$$

このようにして作られた u と v の外積は、3 次元ベクトルであり、u にも v にも直交している。つまり、

$$u \cdot (u \times v) = 0 \tag{1.12}$$
$$v \cdot (u \times v) = 0 \tag{1.13}$$

注意してほしいのは、u と v の外積は v と u の外積とは違う、ということである。この二つは符号が逆になる。つまり、

$$u \times v = -(v \times u) \tag{1.14}$$

である。

さてここで、線形代数の教科書にはないが、多視点幾何でよく用いられる外積の記号を紹介しよう。ベクトル u とベクトル v の外積を実現したい場合、ベクトル u の方をある行列 $[u]_\times$ に置き換えて、

$$u \times v = [u]_\times v \tag{1.15}$$

とすることができる。ここで、

$$[u]_\times \equiv \begin{pmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{pmatrix} \in \mathbb{R}^{3\times3} \tag{1.16}$$

である。この行列の要素は対称的であるが、この行列の転置を取ると、元の行列の符号が反転したものが得られる。つまり、

$$[u]_\times^T = [u]_\times \tag{1.17}$$

である。このような行列を歪対称 (skew-symmetric) 行列という。なぜ外積をこのような歪対称行列で表わす必要があるのだろうか？　それは、外積よりも行列の方が計算式として表わしやすいからである。外積と行列の積の計算が入り混じると、とたんに計算がややこしくなる。さきほどみたように、外積は交換則は成り立たない。それでは結合律は？　分配律はどうなのか？　それよりも、行列の積で表現しておけば、すべて行列計算の法則にのっとって計算することができる。いまは疑うかもしれないが、この先その便利さを実感するようになるだろう。

1.1.2　点の移動と座標系の移動：2 次元の場合

さて、3 次元空間のベクトルの基礎をおさらいしたところだが、いったん 2 次元空間のベクトルに話を移そう。後でまた 3 次元空間の話に戻ることにする。ここで考えるのは、点と座標系の移動である。ある点を移動して別の位置に持ってくる操作を、まず 2 次元で理解しよう。なぜなら 2 次元なら紙の上で点と矢印を書いて直感的に理解できるからだ。いきなり 3 次元の点の移動を考えると、紙に書こうとするだけで混乱してしまうかもしれない。まずは 2 次元の点の回転と並進移動を理解しよう。

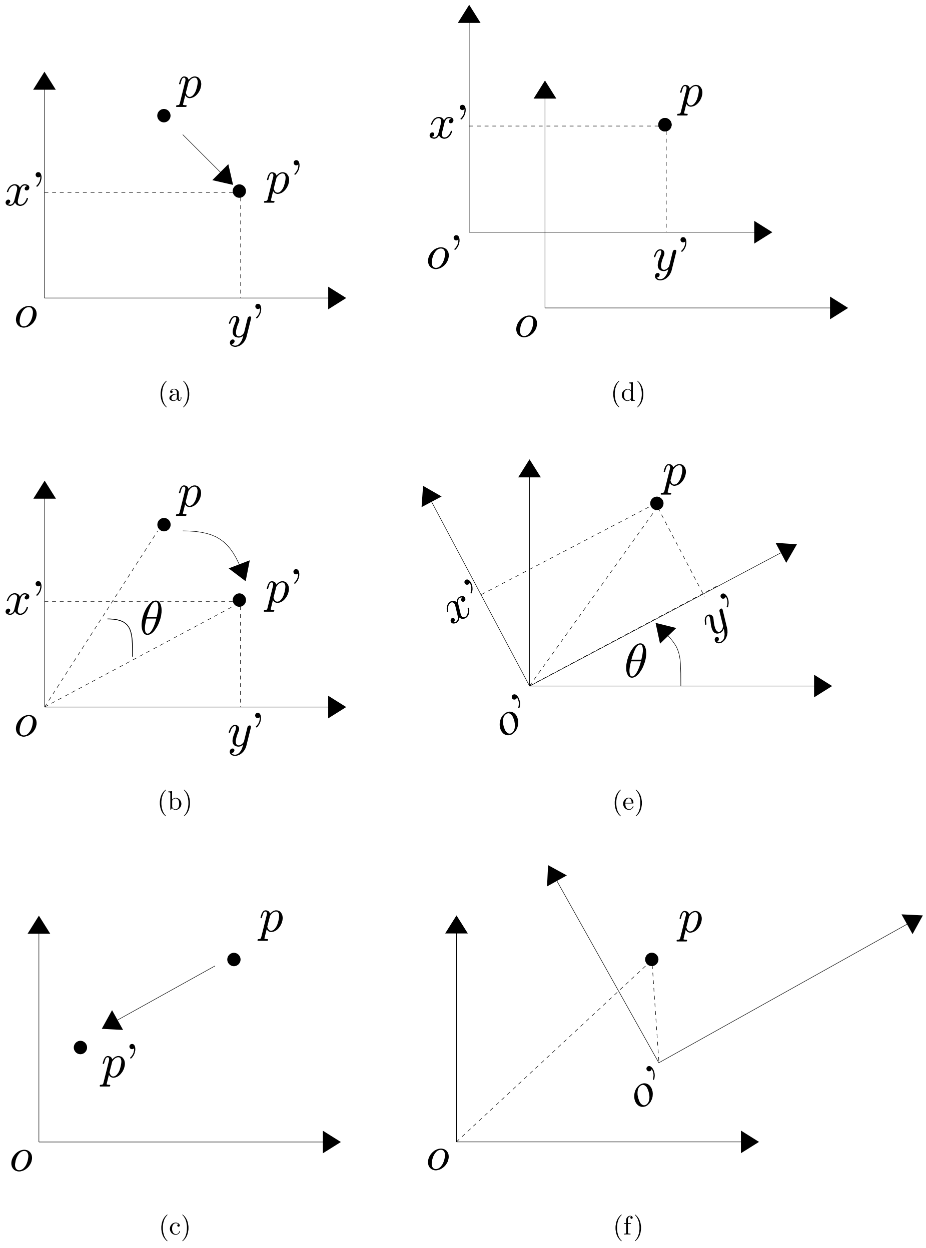


図 1.1: 2 次元空間での点と座標系の回転・並進。

2 次元空間に点 $p = (x, y)^T$ があるとしよう。その点を動かして点 $p' = (x', y')^T$ にしたとする。これを表したのが図 1.1(a) である。x 軸と y 軸が原点 O で直角に交わっており、その座標系で点 p' の座標が (x', y') で表わされている。点 p は、その座標系を固定して右下方向に移動したのである。

しかし、今度は図 1.1(d) のように、座標系を移動してみよう。座標系が左上方向に移動して、新しい位置にきたとする。この座標系を、原点を O とする元の座標系 O と区別するために、原点を O' とした x' 軸と y' 軸で表わされる座標系 O' と呼ぼう。すると、この座標系で見た場合、点 p の座標は $(x', y')^T$ で表わされている。先ほどのとの違いは、点 p は止まっていて、座標系の方が移動したのである。

回転にも、点の回転と座標系の回転がある。図 1.1(b) のように、座標系を固定したまま、点 p を原点 O を中心に右回り（時計回り）に角度 θ 回転させて、点 p' の位置に移動させたとしよう。このときの点 p' の座標は (x', y') であるとする。では今度は、図 1.1(e) のように、点を固定したまま、座標系 O を原点 O を中心に左回り（反時計回り）に角度 θ 回転させて、新たな座標系 O' になったとしよう。すると点 p の座標は、やはり $(x', y')^T$ で表わされている。この二つの違いは、座標系を固定して点を移動させたのか、それとも点を固定して座標系を移動させたのか、である。

一般的な点の移動は、並進移動と回転の組み合わせである。図 1.1(c) のように、並進と回転を組み合わせて、点 p を点 p' の位置に移動する場合を見てみよう。その移動とは反対方向に、元の座標系を並進移動し回転させれば、新たな座標系で点 p の座標を $(x', y')^T$ にすることができる。

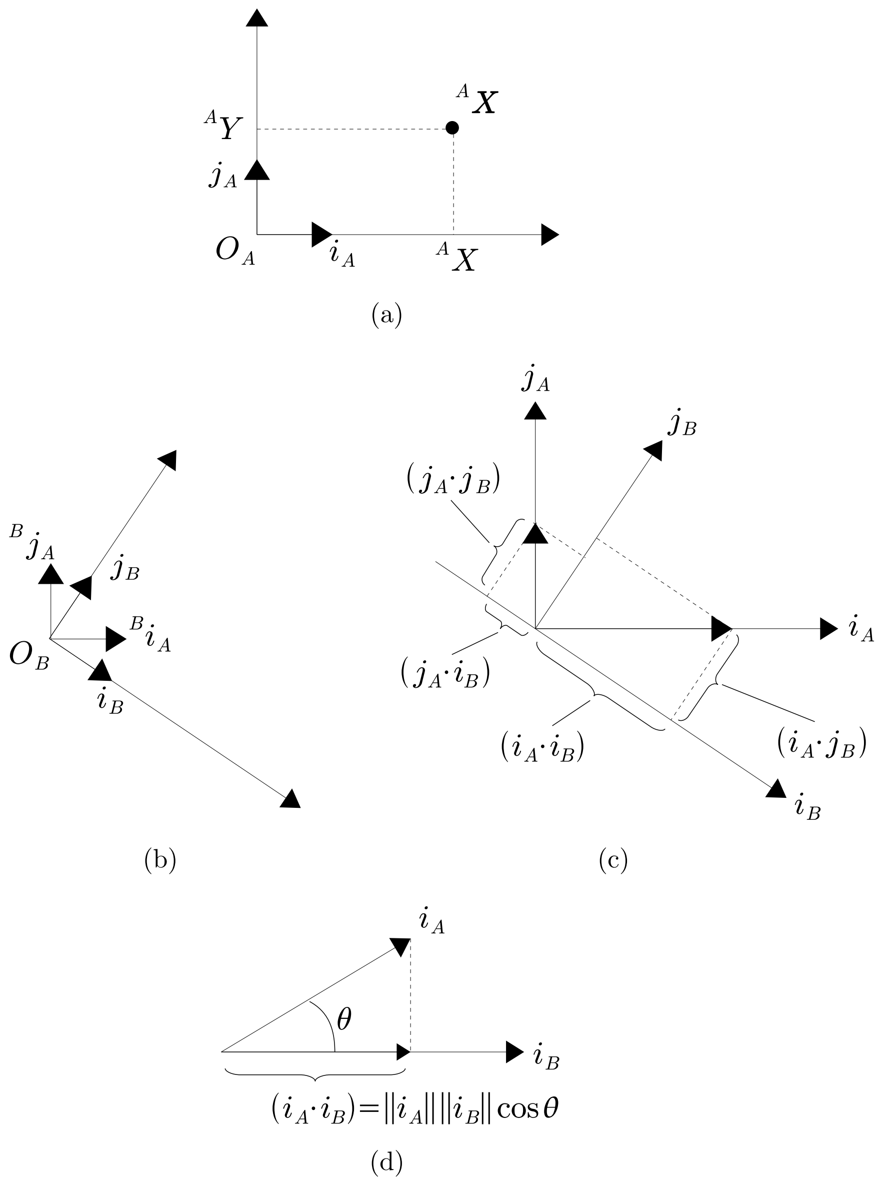
この違いは、微妙だろうか？　それとも大きいだろうか？　駅のホームに立っているあなたにとって、電車に乗っている人たちは間違いなく移動している。しかし、電車に乗っている人たちにとっては、あなたの方が移動していて、そのうち電車の窓から見えなくなってしまう存在なのである。どちらが移動しているのかは、相対的なことである。どちらか一方だけを考えている場合には何の問題もなく、ホームに立っているあなたにとっては、あなたが静止していて電車は動いているのだ。

しかし、今どこから見ているのか、いつもそれだけははっきり認識しておいてほしい。世の中は電車の中とホームの上だけではない。それを駅の外からバスに乗って眺めている乗客もいるのだ。だれがどこからどのように移動しているか。それを間違えると、とたんに世界を記述する数式はおかしくなってしまう。どの視点で移動を考えているのか？　それだけはいつもはっきりさせてほしい。

1.1.3　2 次元の座標系（フレーム）の変換

それでは数式で、2 次元の点と座標系 (coordinate system) の変換を定義しよう。座標系はフレーム (frame) とも呼ばれることがある。ここでは Craig [1] の記法に従って、座標系（もしくはフレーム）の変換を記述してみよう。

ある点 p の座標を $\boldsymbol{X}$ とする。この座標が、どの座標系における座標なのかをはっきりさせるために、座標系 A における点 p の座標を ${}^{A}\boldsymbol{X} = ({}^{A}X, {}^{A}Y)^T$ と書くことにする。この座標系 A の原点を O_A、座標系を定義する二つの軸を i_A, j_A としよう。i_A も j_A も、その軸の方向の単位ベクトルであるということを約束しておく。つまり、$||i_A|| = ||j_A|| = 1$ である。



^AY
^AX
j_A
O_A
i_A
^AX
(a)
^Bj_A
j_B
^Bi_A
O_B
i_B
(b)
j_A
j_B
$(j_A \cdot j_B)$
i_A
$(j_A \cdot i_B)$
$(i_A \cdot i_B)$
$(i_A \cdot j_B)$
i_B
(c)
i_A
θ
i_B
$(i_A \cdot i_B) = \|i_A\| \|i_B\| \cos\theta$
(d)

図 1.2: 2 次元の座標系。

この状況を図で描くと、図 1.2(a) のようになる。数式で書くと、次のようになる。

$$ {}^A\boldsymbol{X} = \begin{pmatrix} {}^AX \\ {}^AY \end{pmatrix} \equiv {}^AX\ i_A + {}^AY\ j_A \tag{1.18} $$

つまり、座標系 A の i_A 軸方向の座標成分が ${}^AX$、j_A 軸方向の座標成分が ${}^AY$ ということを表している。点 p の座標 ${}^A\boldsymbol{X}$ は、i_A を ${}^AX$ 倍、j_A を ${}^AY$ 倍して足したものである、ともいえる。

i_A も j_A も 2 次元のベクトルであるので、2 個の成分を持っている。i_A と j_A の合計 4 個の成分を並べた 2×2 の行列を使うと、上の式が行列の計算式で表わせる。

$$ {}^AX\ i_A + {}^AY\ j_A = \begin{pmatrix} \boxed{i_A} & \boxed{j_A} \end{pmatrix} \begin{pmatrix} {}^AX \\ {}^AY \end{pmatrix} = \begin{pmatrix} \boxed{i_A} & \boxed{j_A} \end{pmatrix} {}^A\boldsymbol{X} \tag{1.19} $$

この式の中の縦長の四角形は、そのベクトルが縦方向のベクトルである、ということを見やすくしているだけである。数学的な意味は全然ないので、最後の数式は単に $(i_A\ j_A)\ {}^A\boldsymbol{X}$ と書いたものと同じである。

では次に、原点 O_B と二つの軸 i_B, j_B で表わされる座標系 B で同じことを考えてみよう。この座標系 B における点 p の座標を ${}^B\boldsymbol{X} = ({}^BX, {}^BY)^T$ と書くと、先ほどと同じような数式が書ける。

$$ {}^B\boldsymbol{X} = \begin{pmatrix} {}^BX \\ {}^BY \end{pmatrix} \equiv {}^BX\ i_B + {}^BY\ j_B = \begin{pmatrix} \boxed{i_B} & \boxed{j_B} \end{pmatrix} \begin{pmatrix} {}^BX \\ {}^BY \end{pmatrix} \tag{1.20} $$

さあこれで準備がととのった。それでは点 p を固定して、座標系を A から B に変換してみよう。つまり、座標系 A で表わされた点 p の座標 ${}^A\boldsymbol{X}$ から、座標系 B で表わされた点 p の座標 ${}^B\boldsymbol{X}$ を導き出すのである。そのためにはどうすればよいだろうか？

まずは並進がない簡単な場合を考えよう。つまり、二つの原点が一致する場合 $(O_A = O_B)$ である。このとき忘れないでほしいのは、座標系が移動するのであって、点 p は移動していない、ということである。座標系 A において、点 p の座標 ${}^A\boldsymbol{X}$ は、i_A を ${}^AX$ 倍、j_A を ${}^AY$ 倍して足したものであった。これは、いつでもどこでも、どこから見ていても成り立つ。つまり、座標系 B から見ていても、それが成り立っていなければならないのだ。これをどう説明すればよいだろうか？

そのためには、軸そのものを変換する必要がある。座標系 A の軸は、座標系 A から見ていたら i_A であるが、座標系 B から見たら ${}^Bi_A$ であるとしよう。同じように、座標系 A から見た j_A は、座標系 B から見たら ${}^Bj_A$ であるとする。これは図 1.2(b) を見てもらうと早いだろう。すると、点 p の座標は、i_A を ${}^AX$ 倍、j_A を ${}^AY$ 倍して足したものであり、これは座標系 B においても成り立つので、

$$ {}^B\boldsymbol{X} = {}^AX\ {}^Bi_A + {}^AY\ {}^Bj_A \tag{1.21} $$

でなければならない。

ここで、${}^Bi_A$ と ${}^Bj_A$ は次のように求めることができる。

$$ {}^Bi_A = (i_A \cdot i_B)\ i_B + (i_A \cdot j_B)\ j_B \tag{1.22} $$

$$ {}^Bj_A = (j_A \cdot i_B)\ i_B + (j_A \cdot j_B)\ j_B \tag{1.23} $$

行列で書けば次のようになる。

$$\begin{pmatrix} \boxed{{}^B i_A} & \boxed{{}^B j_A} \end{pmatrix} = \begin{pmatrix} \boxed{i_B} & \boxed{j_B} \end{pmatrix} \begin{pmatrix} i_A \cdot i_B & j_A \cdot i_B \\ i_A \cdot j_B & j_A \cdot j_B \end{pmatrix} \tag{1.24}$$

これを直感的に理解するには、ベクトルの投影と内積の関係をおさらいする必要がある。たとえば図 1.2(d) のように、ベクトル i_A をベクトル i_B 方向に投影してできたベクトルの長さはどのくらいだろうか？　二つのベクトルがなす角を θ とすると、三角関数の関係からすぐに、その長さは $||i_A|| \cos\theta$ であるとわかるだろう。方向はもちろん i_B である。一方、内積の関係式は $i_A \cdot i_B = ||i_A||\ ||i_B||\ \cos\theta$ であった。さらに、ここでは i_A や i_B は座標系の軸を表すベクトルで、長さが 1 の単位ベクトルとしていた。つまり $||i_A|| = ||i_B|| = 1$ ということである。すると、投影した長さも内積も、どちらも $\cos\theta$ になる。つまり「$i_A \cdot i_B = \cos\theta = i_A$を i_B 方向に投影したときのベクトルの長さ」が成り立ち、「i_A を i_B 方向に投影したときのベクトル」は、その方向の単位ベクトルにその長さをかけたものである。これが $(i_A \cdot i_B)\ i_B$ である。

同様に、i_A, j_A をそれぞれ i_B, j_B に投影した状況が図 1.2(c) である。${}^B i_A$ とは、i_A を i_B 方向の成分と j_B 方向の成分で表わしたものである。同様に、${}^B j_A$ は、j_A を i_B 方向の成分と j_B 方向の成分で表わしたものである。方向成分は、それぞれの方向に投影した長さであり、内積で計算できる。あとは、その方向の単位ベクトルにかけて加え合わせるだけである。これが先ほどの ${}^B i_A$ と ${}^B j_A$ の式である。

では、今まで出てきた数式を整理して、${}^B\boldsymbol{X}$ を導出してみよう。

$$\begin{aligned}
{}^B\boldsymbol{X} &= {}^A X\ {}^B i_A + {}^A Y\ {}^B j_A && (1.25)\\
&= \begin{pmatrix} \boxed{{}^B i_A} & \boxed{{}^B j_A} \end{pmatrix} \begin{pmatrix} {}^A X \\ {}^A Y \end{pmatrix} && (1.26)\\
&= \begin{pmatrix} \boxed{i_B} & \boxed{j_B} \end{pmatrix} \begin{pmatrix} i_A \cdot i_B & j_A \cdot i_B \\ i_A \cdot j_B & j_A \cdot j_B \end{pmatrix} \begin{pmatrix} {}^A X \\ {}^A Y \end{pmatrix} && (1.27)\\
&= \begin{pmatrix} \boxed{i_B} & \boxed{j_B} \end{pmatrix} \begin{pmatrix} i_B^T i_A & i_B^T j_A \\ j_B^T i_A & j_B^T j_A \end{pmatrix} \begin{pmatrix} {}^A X \\ {}^A Y \end{pmatrix} && (1.28)\\
&= \begin{pmatrix} \boxed{i_B} & \boxed{j_B} \end{pmatrix} \begin{pmatrix} \boxed{i_B^T} \\ \boxed{j_B^T} \end{pmatrix} \begin{pmatrix} \boxed{i_A} & \boxed{j_A} \end{pmatrix} \begin{pmatrix} {}^A X \\ {}^A Y \end{pmatrix} && (1.29)
\end{aligned}$$

すると、先ほどの座標系 B の式と見比べると、以下の式が成り立つことがわかるだろう。

$$\begin{pmatrix} {}^B X \\ {}^B Y \end{pmatrix} = \begin{pmatrix} \boxed{i_B^T} \\ \boxed{j_B^T} \end{pmatrix} \begin{pmatrix} \boxed{i_A} & \boxed{j_A} \end{pmatrix} \begin{pmatrix} {}^A X \\ {}^A Y \end{pmatrix} \tag{1.30}$$

つまり、${}^B\boldsymbol{X}$ と ${}^A\boldsymbol{X}$ は、行列

$${}^B_A R = \begin{pmatrix} \boxed{i_B^T} \\ \boxed{j_B^T} \end{pmatrix} \begin{pmatrix} \boxed{i_A} & \boxed{j_A} \end{pmatrix} = \begin{pmatrix} i_A \cdot i_B & j_A \cdot i_B \\ i_A \cdot j_B & j_A \cdot j_B \end{pmatrix} \tag{1.31}$$

で関係づけられている。この行列が、座標系 A から B への回転行列 (rotation matrix) と呼ばれるものである。これを用いると、座標系の変換は簡単に

$$ {}^{B}\boldsymbol{X} = {}^{B}_{A}R\,{}^{A}\boldsymbol{X} \tag{1.32} $$

と書くことができる。座標系の変換が回転だけであり、並進を含まなければ、これで終わりである。並進を含む場合の一般的な形は、これと同じことを 3 次元でやったあとに考えることにしよう。

最後に、ひとつ考えてほしい。i_A, j_A, i_B, j_B は、いったいどこの座標系で表わされているベクトルだろう？　答えは、どこでもよい。ただし同じ座標系で表わすこと。たとえば、もしこれらがすべて座標系 A で表わされていたら、$i_A = (1,0)^T, j_A = (0,1)^T$ となり、これらを並べた行列は単位行列

$$ \begin{pmatrix} \boxed{i_A} & \boxed{j_A} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{1.33} $$

になる。もちろん i_B, j_B の成分は 0,1 にはならない。逆にすべて座標系 B で表わされていたら、i_B, j_B を並べた行列の方が単位行列になる。もし、まったく違う座標系 C で表わされていたら、i_A, j_A を並べた行列も i_B, j_B を並べた行列も、単位行列ではない。それでも、${}^{A}\boldsymbol{X}$ と ${}^{B}\boldsymbol{X}$ は、座標系 C に存在する点 p を違うやり方で表現しているだけなので、${}^{B}\boldsymbol{X} = {}^{A}\boldsymbol{X}$ のはずである。つまり、

$$ \begin{pmatrix} \boxed{i_B} & \boxed{j_B} \end{pmatrix} \begin{pmatrix} {}^{B}X \\ {}^{B}Y \end{pmatrix} = \begin{pmatrix} \boxed{i_A} & \boxed{j_A} \end{pmatrix} \begin{pmatrix} {}^{A}X \\ {}^{A}Y \end{pmatrix} \tag{1.34} $$

ここで軸は直交していることを仮定しているので、逆行列の代わりに転置を左からかけると（直交行列の逆行列は転置したものに等しいから）

$$ \begin{pmatrix} {}^{B}X \\ {}^{B}Y \end{pmatrix} = \begin{pmatrix} \boxed{i_B} & \boxed{j_B} \end{pmatrix}^T \begin{pmatrix} \boxed{i_A} & \boxed{j_A} \end{pmatrix} \begin{pmatrix} {}^{A}X \\ {}^{A}Y \end{pmatrix} \tag{1.35} $$

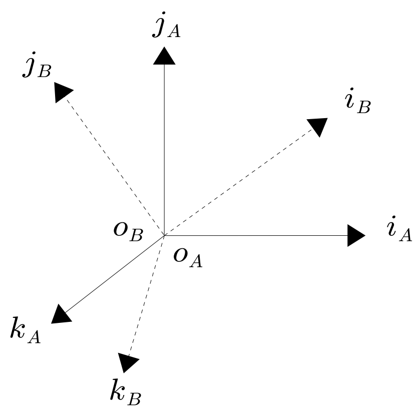
$$ = \begin{pmatrix} \boxed{i_B^T} \\ \boxed{j_B^T} \end{pmatrix} \begin{pmatrix} \boxed{i_A} & \boxed{j_A} \end{pmatrix} \begin{pmatrix} {}^{A}X \\ {}^{A}Y \end{pmatrix} \tag{1.36} $$

となり、同じ結果が得られる。考え方が違うだけなので、理解しやすい形を覚えるようにしてほしい。

1.1.4　3 次元の座標系（フレーム）

それでは、3 次元での座標系の変換を定義しよう。ここでも簡単のために、最初は並進移動がなく、回転のみの変換を考えることにする。2 次元の変換で見たやり方と全く同じように、3 次元での回転を定義できることがわかるだろう。

ある点 p の座標系 A における座標を ${}^{A}\boldsymbol{X} = ({}^{A}X, {}^{A}Y, {}^{A}Z)^T$ と書くことにする。この座標系 A の原点を O_A、座標系を定義する 3 つの軸を i_A, j_A, k_A とする。i_A も j_A も k_A、その軸の方向の単位



(a)

図 1.3: 3 次元の座標系。

ベクトルであり、$||i_A|| = ||j_A|| = ||k_A|| = 1$ であるとする。これを数式で書くと次のようになる。

$$ {}^A\boldsymbol{X} = \begin{pmatrix} {}^AX \\ {}^AY \\ {}^AZ \end{pmatrix} \equiv {}^AX\ i_A + {}^AY\ j_A + {}^AZ\ k_A \tag{1.37} $$

3 次元の場合では、i_A も j_A も k_A も 3 次元ベクトルである。それらを並べた 3×3 行列で上の式を表すと、以下のようになる。

$$ {}^AX\ i_A + {}^AY\ j_A + {}^AZ\ k_A = \begin{pmatrix} \boxed{i_A} & \boxed{j_A} & \boxed{k_A} \end{pmatrix} \begin{pmatrix} {}^AX \\ {}^AY \\ {}^AZ \end{pmatrix} = \begin{pmatrix} \boxed{i_A} & \boxed{j_A} & \boxed{k_A} \end{pmatrix} {}^A\boldsymbol{X} \tag{1.38} $$

では、この座標系 A で表わされた点 p の座標を、座標系 B で表わしてみよう。やり方は 2 次元の場合とまったく同じである。つまり、座標系 A の 3 つの軸 i_A, j_A, k_A を座標系 B で表わすのである。

$$ {}^Bi_A = (i_A \cdot i_B)\ i_B + (i_A \cdot j_B)\ j_B + (i_A \cdot k_B)\ k_B \tag{1.39} $$
$$ {}^Bj_A = (j_A \cdot i_B)\ i_B + (j_A \cdot j_B)\ j_B + (j_A \cdot k_B)\ k_B \tag{1.40} $$
$$ {}^Bk_A = (k_A \cdot i_B)\ i_B + (k_A \cdot j_B)\ j_B + (k_A \cdot k_B)\ k_B \tag{1.41} $$

この軸を使うと、点 p の座標系 B での座標 ${}^B\boldsymbol{X}$ は、以下のように書くことができる。

$$ {}^B\boldsymbol{X} = {}^AX\ {}^Bi_A + {}^AY\ {}^Bj_A + {}^AZ\ {}^Bk_A \tag{1.42} $$
$$ = \begin{pmatrix} \boxed{{}^Bi_A} & \boxed{{}^Bj_A} & \boxed{{}^Bk_A} \end{pmatrix} \begin{pmatrix} {}^AX \\ {}^AY \\ {}^AZ \end{pmatrix} \tag{1.43} $$

あとは、2 次元の場合と同じように、この式をの右辺の行列を以下のように展開してみるだけである。

$$ {}^{B}\boldsymbol{X} = \begin{pmatrix} \boxed{i_B} & \boxed{j_B} & \boxed{k_B} \end{pmatrix} \begin{pmatrix} i_A \cdot i_B & j_A \cdot i_B & k_A \cdot i_B \\ i_A \cdot j_B & j_A \cdot j_B & k_A \cdot j_B \\ i_A \cdot k_B & j_A \cdot k_B & k_A \cdot k_B \end{pmatrix} \begin{pmatrix} {}^{A}X \\ {}^{A}Y \\ {}^{A}Z \end{pmatrix} \tag{1.44} $$

$$ \equiv \begin{pmatrix} \boxed{i_B} & \boxed{j_B} & \boxed{k_B} \end{pmatrix} \begin{pmatrix} {}^{B}X \\ {}^{B}Y \\ {}^{B}Z \end{pmatrix} \tag{1.45} $$

すなわち、

$$ {}^{B}_{A}R = \begin{pmatrix} i_A \cdot i_B & j_A \cdot i_B & k_A \cdot i_B \\ i_A \cdot j_B & j_A \cdot j_B & k_A \cdot j_B \\ i_A \cdot k_B & j_A \cdot k_B & k_A \cdot k_B \end{pmatrix} \tag{1.46} $$

とおくと、

$$ {}^{B}\boldsymbol{X} = {}^{B}_{A}R \ {}^{A}\boldsymbol{X} \tag{1.47} $$

が得られる。この行列が、座標系 A から B への回転行列 (rotation matrix) である。

2 次元の場合と同様に、回転行列は二つの座標系の軸の内積で作られている。以下のように、軸を並べた行列の積でも計算できる。

$$ {}^{B}_{A}R = \begin{pmatrix} \boxed{i_B^T} \\ \boxed{j_B^T} \\ \boxed{k_B^T} \end{pmatrix} \begin{pmatrix} \boxed{i_A} & \boxed{j_A} & \boxed{k_A} \end{pmatrix} \tag{1.48} $$

よくある問題は、二つの座標系の軸が与えられた時に、それらを変換する回転行列を求める、というものである。この式を覚えておけば、すぐに答えが計算できるだろう。

では、座標系 B から座標系 A に変換する回転行列は、いったいどうやって求めたらいいだろうか？　答えは簡単である。先ほどの回転行列 ${}^{B}_{A}R$ の逆である。つまり、${}^{A}_{B}R$ が求めるべき回転行列である。ではその中身は何だろうか？　上の式を見てほしい。A と B を入れ替えるだけなので、

$$ {}^{A}_{B}R = \begin{pmatrix} \boxed{i_A^T} \\ \boxed{j_A^T} \\ \boxed{k_A^T} \end{pmatrix} \begin{pmatrix} \boxed{i_B} & \boxed{j_B} & \boxed{k_B} \end{pmatrix} \tag{1.49} $$

が答である。そしてよく見れば、これは ${}^{B}_{A}R$ を転置したものである。結局、

$$ {}^{A}_{B}R = ({}^{B}_{A}R)^T \tag{1.50} $$

である。すると、座標系 A から B、B から A への以下の変換式

$$ {}^{B}\boldsymbol{X} = {}^{B}_{A}R \ {}^{A}\boldsymbol{X} \tag{1.51} $$

$$ {}^{A}\boldsymbol{X} = {}^{A}_{B}R \ {}^{B}\boldsymbol{X} = ({}^{B}_{A}R)^T \ {}^{B}\boldsymbol{X} \tag{1.52} $$

から、次のことがわかる。

$$({}^B_AR)^T = ({}^B_AR)^{-1} \tag{1.53}$$

つまり、転置はその逆行列に等しいので、回転行列は直交行列 (orthogonal matrix) である。

3 つ目の座標系 C が現れたら、どうしたらいいだろうか。座標系 A から B への回転行列 ${}^B_AR$ と B から C への回転行列 ${}^C_BR$ とが与えらたとき、A から C への回転行列 ${}^C_AR$ はどのように求めたらよいだろう。これは直感的にもわかりやすいだろう。A から B へ変換した後に、続けて B から C へ変換すればよい。つまり、

$${}^C_AR = {}^C_BR\ {}^B_AR \tag{1.54}$$

である。これを連鎖律 (chain rule) と呼ぶこともある。C から A への変換は、回転行列の転置を取ればよいので

$${}^A_CR = ({}^C_AR)^T = ({}^C_BR\ {}^B_AR)^T = ({}^B_AR)^T({}^C_BR)^T \tag{1.55}$$

で求められる。座標系がいくつあっても、基本的にこの連鎖律で順次求めていけば、どの座標系の間の回転行列も求めることができる。ロボットや人体の関節や多数のカメラを扱うときには、座標系を 10 個も 20 個も扱うことがあるので、あわてずおそれず、どの座標系との変換を求めているのかを常に頭において計算しよう。

1.1.5 回転の表現

さて、3 次元の座標系の回転行列を求めたところで、いろいろな回転の表現方法を示しておこう。3 次元の回転は、回転行列だけが唯一の表現方法ではない。そのことを見る前に、まず一般的な回転の定義を見てみよう。

特殊直交群 (Special Orthogonal Group) $SO(3)$ とは、以下で定義される 3×3 行列の集合である。

$$SO(3) \equiv \{R \in \mathbb{R}^{3\times3} \mid R^TR = I,\ |R| = +1\} \tag{1.56}$$

ここで $|R|$ は行列 R の行列式である。つまりこれは、3×3 の直交行列のうち、行列式が 1 であるものの集合を定義している。すべての回転行列は $SO(3)$ に含まれ、そのため $SO(3)$ は 3 次元の回転群とも呼ばれる。

$SO(3)$ に含まれる回転行列 R には、9 個の要素がある。しかしそれらの値を適当に決めても、R が直交行列になる保証はないし、行列式が 1 になる可能性も小さい。9 個の要素をどのように決めれば、出来上がった 3×3 行列は回転行列になるだろうか？　そのためには、R の 3 つの列ベクトルを互いに直交するようにして、なおかつ列ベクトルのノルムを 1 にする必要がある。これが、出来上がった R が直交行列になる条件である。つまり、

$$R \equiv \begin{pmatrix} \boxed{r_1} & \boxed{r_2} & \boxed{r_3} \end{pmatrix} \tag{1.57}$$

とすると、条件は以下のように書くことができる。

$$R^TR = I \iff r_1\cdot r_2 = r_2\cdot r_3 = r_3\cdot r_1 = 0,\ ||r_1|| = ||r_2|| = ||r_3|| = 1 \tag{1.58}$$

それぞれの軸が直交するという条件式が 3 つ、ノルムが 1 であるという条件式が 3 つあるので、条件は合計 6 つである。一方、行列の要素の数は 9 個である。そのため、いろいろな回転行列を作るために自由に決められるパラメータの数は、9 から 6 を引いた 3 である。これを自由度 (degree-of-freedom, DOF) といい、$SO(3)$ の自由度は 3 である。逆にいえば、回転行列を一意に決定するためには 9 個の数字を決める必要はなく、3 個の数字で十分である、ということである。

では、具体的にはどんな 3 つの数字を決めれば、9 個の回転行列の要素を決定できるだろうか？

これにはいろいろなやり方があり、そのためにいろいろな回転の表現が存在しているのである。以下では、代表的な 4 つの回転の表現を見ていくことにしよう。

固定角表現

一つ目は、固定角表現 (fixed angle representation) である。これは、最初に三つの軸を固定して、その軸周りの回転を連続して行うことで回転を表現するものである。

元の座標系 A が、図 1.4(a) のように、X, Y, Z の三つの軸で表わされているとしよう。最初の変換は、この座標系 A の X 軸周りに γ だけ回転するものである。その様子が図 1.4(b) である。この回転で、X, Y, Z 軸で定義されていた座標系 A から X_1, Y_1, Z_1 軸で定義される座標系に変換されることになる。ただし、X 軸周りの回転では X 軸自体は動かないので、X_1 軸と X 軸は同じものである。この回転を表す回転行列を $R_X(\gamma)$ と書くことにしよう。

次の変換は、X_1, Y_1, Z_1 座標系を、元の座標系 A の Y 軸周りに β だけ回転するものである。この回転を表す回転行列を $R_Y(\beta)$ と書くことにする。この回転で、X_1, Y_1, Z_1 座標系は X_2, Y_2, Z_2 座標系に変換されることになる。

最後の変換は、X_2, Y_2, Z_2 座標系を、元の座標系 A の Z 軸周りに α だけ回転するものである。この回転を表す回転行列を $R_Z(\alpha)$ と書くことにする。この回転で、X_2, Y_2, Z_3 座標系は X_3, Y_3, Z_3 座標系に変換されることになる。こうしてできた座標系が、最終的に座標系 A から変換したい座標系 B である。

固定角表現は、上記の三つの回転を順次行うことで座標変換を行う。つまり、座標系 A から B に変換する回転行列 ${}^B_A R$ は、

$$
{}^B_A R_{XYZ}(\gamma, \beta, \alpha) = R_Z(\alpha)\ R_Y(\beta)\ R_X(\gamma) \tag{1.59}
$$

で求めることができる。ここで、回転行列を決めているのは 3 つの角度 α, β, γ であることに注意してほしい。回転は 3 自由度なので、3 通の数字を決定すれば回転行列を決めることができるのである。

それぞれの軸周りの回転を表す回転行列 R_X, R_Y, R_Z は、以下のような形をしている。

$$R_X(\gamma) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{pmatrix} \tag{1.60}$$

$$R_Y(\beta) = \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix} \tag{1.61}$$

$$R_Z(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{1.62}$$

先ほど示した固定角表現 ${}^B_A R_{XYZ}$ は、この行列を R_X, R_Y, R_Z の順にかけたものであった。この順番を変えると、別の固定角表現ができ上がる。回転の順序を変えたものは全部で 12 通り存在する [1]。つまり先ほどの固定角表現は、その中の XYZ 固定角表現と呼ばれるものである。たとえば Z, Y, X 軸の順に回転して出来上がったものは、ZYX 固定角表現 ${}^B_A R_{ZYX}$ ということになる。どの軸から回転するのかという順番を把握していないと、「X 軸周りに 30 度回転」「Z 軸周りに 25 度回転」と伝えても、相手と結果が一致しないので注意が必要である。

オイラー角表現

二つ目は、オイラー角表現 (Euler angle representation) である。固定角表現と同様に、軸周りの回転を連続して行うことで回転を表現する。ただし、それぞれの回転の軸は、固定角表現とは違う軸である。

元の座標系 A が、図 1.5(a) のように、X, Y, Z の三つの軸で表わされているとしよう。最初の変換は、この座標系 A の X 軸周りに γ だけ回転するものである。その様子が図 1.5(b) である。この回転で、X, Y, Z 軸で定義されていた座標系 A から X_1, Y_1, Z_1 軸で定義される座標系に変換されることになる。ただし、X 軸周りの回転では X 軸自体は動かないので、X_1 軸と X 軸は同じものである。この回転を表す回転行列を $R_X(\gamma)$ と書くことにしよう。

ここまでは、固定角表現での最初の X 軸周りの回転と同じである。違うのはここからだ。

次の変換は、X_1, Y_1, Z_1 座標系を、Y_1 軸周りに β だけ回転するものである。この回転を表す回転行列を $R_{Y_1}(\beta)$ と書くことにする。この回転で、X_1, Y_1, Z_1 座標系は X_2, Y_2, Z_2 座標系に変換されることになる。ただし、Y_1 軸周りの回転では Y_1 軸自体は動かないので、Y_1 軸と Y_2 軸は同じものである。

最後の変換は、X_2, Y_2, Z_2 座標系を、Z_2 軸周りに α だけ回転するものである。この回転を表す回転行列を $R_{Z_2}(\alpha)$ と書くことにする。この回転で、X_2, Y_2, Z_3 座標系は X_3, Y_3, Z_3 座標系に変換されることになる。ただし、Z_2 軸周りの回転では Z_2 軸自体は動かないので、Z_2 軸と Z_3 軸は同じものである。

こうしてできた座標系が、最終的に座標系 A から変換したい座標系 B である。座標系 A から B に変換する回転行列 ${}^B_A R$ は、それぞれの軸周りの回転を表す回転行列の積

$$R_{Z_2}(\alpha)\ R_{Y_1}(\beta)\ R_X(\gamma) \tag{1.63}$$

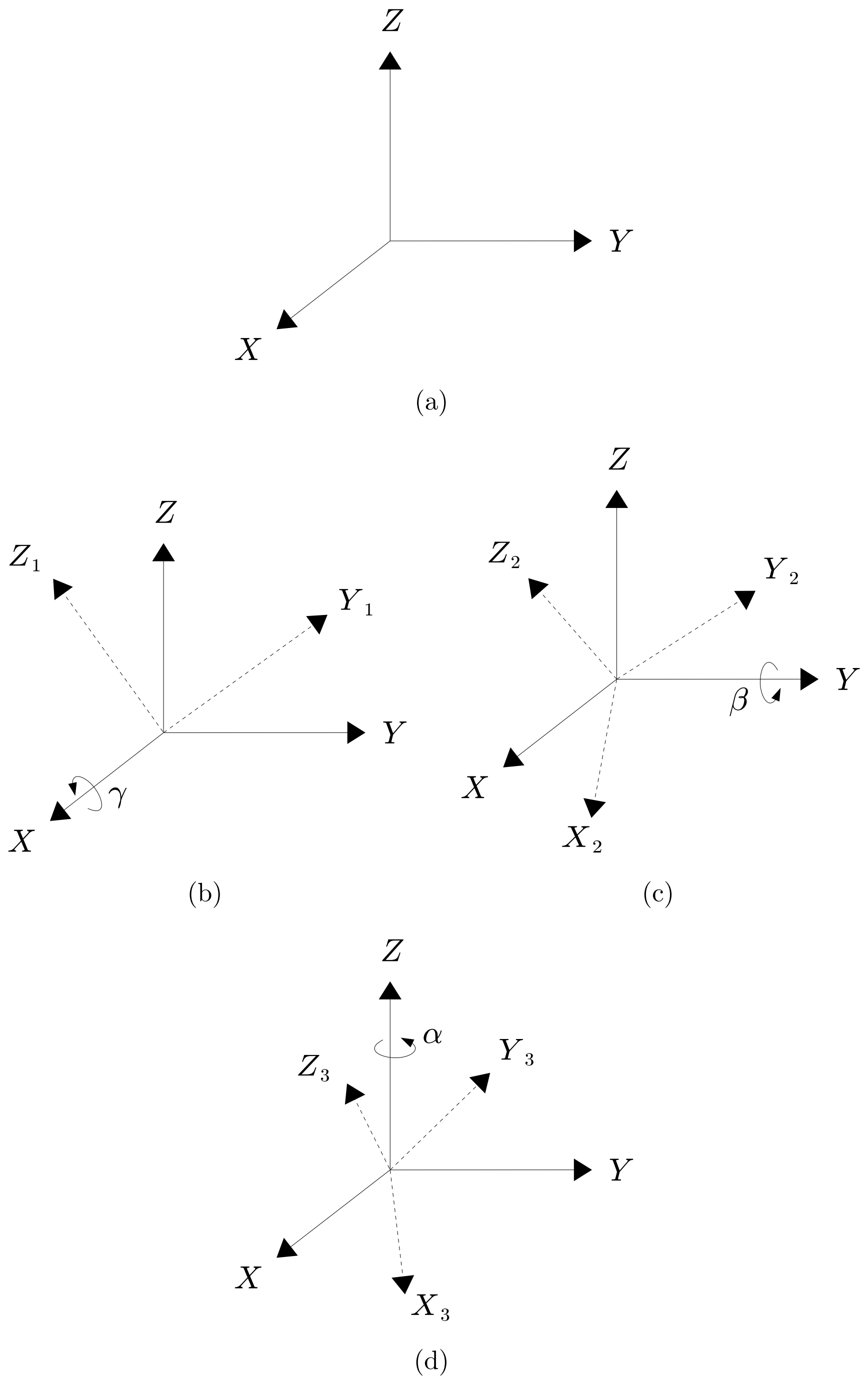


図 1.4: 回転の固定角表現。

で求めることができる。

しかし、このままでは非常に不便な点がある。それは、R_{Z_2} や R_{Y_1} が、R_X のような簡単な形では表せないのである。だいたい、これから回転を作ろうというときに、変換途中の Z_2 軸を使う回転をあらかじめ用意しなければならないとしたら、手順が逆である。最初に角度を 3 つ与えたら、回転行列がすぐに 3 つ用意できる、ということでなければ、積を取って計算する必要性がないではないか。

実は、回転する軸の順序を逆にすると、オイラー角表現が固定角表現で表わせるのである。これはいったいどういうことだろうか？　最初の二つの回転 R_X と R_{Y_1} をみてみよう。まず、X, Y, Z 座標系を X 軸周りで回転するのが R_X であった。その次に X_1, Y_1, Z_1 座標系を Y_1 軸周りで回転するが、これが問題であった。Y_1 軸周りの回転 R_{Y_1} をどうやって作るのか？　しかし、見方を変えてみよう。結果が同じであればよいのなら、先に Y 軸周りで回してしまうのである。Y 軸周りで β だけ回転すると、X, Z 軸がそれぞれ移動する。これは、Y_1 軸周りで β だけ回転して、X_1, Z_1 軸がそれぞれ X_2, Z_2 軸に移動することと、まったく同じはずである。そのあとに X 軸で γ だけ回転すれば、最初の二つの回転 R_X と R_{Y_1} が実現できていることになる。

まとめてみよう。R_X, R_{Y_1} の順に回転を実現したければ、R_Y, R_X の順に回転を行えばよい。この考えはそのまま拡張できて、R_X, R_{Y_1}, R_{Z_2} の順に回転を実現したければ、R_Z, R_Y, R_X の順に回転を行えばよい。つまり、オイラー角表現で座標系 A から B に変換する回転行列 ${}^B_AR$ は、

$$ {}^B_AR_{XY_1Z_2}(\gamma, \beta, \alpha) = R_X(\gamma)\ R_Y(\beta)\ R_Z(\alpha) \tag{1.64} $$

で求めることができる。

オイラー角も、回転軸の順序を変えると、別のオイラー角表現ができ上がる。回転の順序を変えたものは全部で 12 通り存在する [1]。先ほどのオイラー角表現は、その中の XYZ オイラー角表現と呼ばれるものである。たとえば Z, Y_1, X_2 軸の順に回転して出来上がったものは、ZYX オイラー角表現 ${}^B_AR_{ZY_1X_2}$ ということになる。

オイラー角表現と固定角表現は、回転の仕方も行列の積の形も、非常に似ている。どういう関係にあるのだろうか？　それは、ZYX オイラー角表現 ${}^B_AR_{ZY_1X_2}$ を具体的に見てみれば、すぐにわかる。

$$ {}^B_AR_{ZY_1X_2} = R_ZR_YR_X \tag{1.65} $$

これは、XYZ 固定角表現 ${}^B_AR_{XYZ}$ と全く同じである。つまり、オイラー角表現と固定角表現は、軸の順序を入れ替えればまったく同じものである。XYZ 固定角表現は ZYX オイラー角表現と同じであるし、ZYX 固定角表現は XYZ オイラー角表現と同じである。

回転角–回転軸表現

三つ目は、回転角–回転軸表現 (angle–axis representation) である。この表現は、あるベクトルと、それを軸として回転する角度で、回転を表現する。

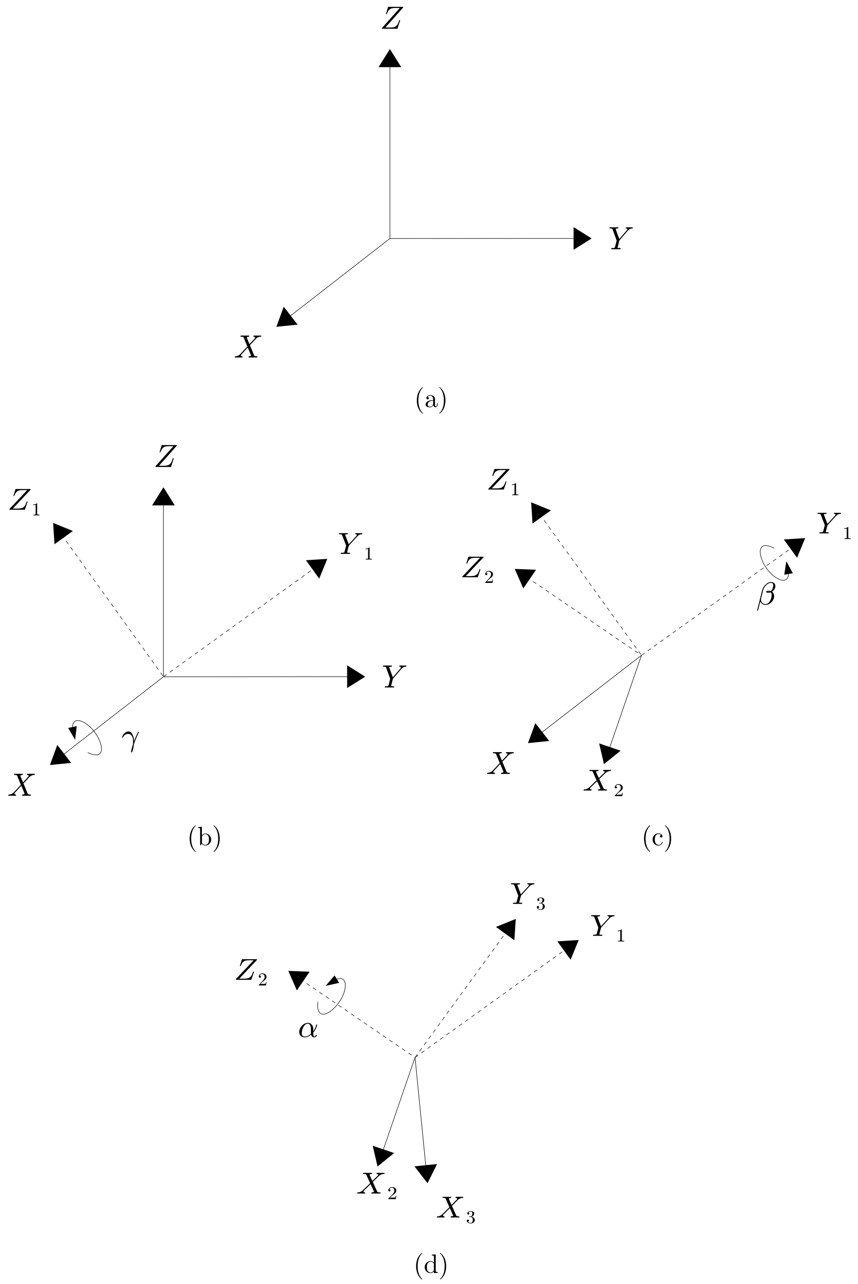


図 1.5: 回転のオイラー角表現。

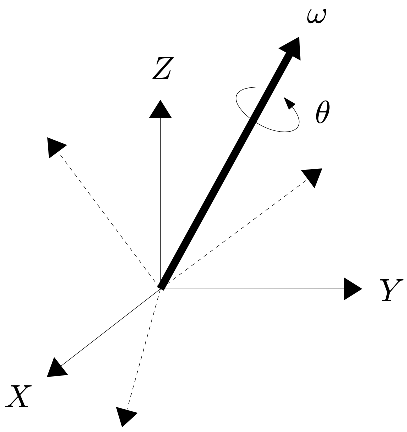


図 1.6: 回転の回転角–回転軸表現。

まずここで、ある回転行列 R $(\neq I)$ の要素が以下のように与えられているとする。

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \tag{1.66}$$

この回転は、以下のベクトル ω を軸として、ω 周りの角度 θ の回転で表現できる。

$$\omega = \frac{1}{2\sin||\omega||} \begin{pmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{pmatrix} \tag{1.67}$$

$$\theta \equiv ||\omega|| = \cos^{-1}\left(\frac{\mathrm{tr}(R) - 1}{2}\right) \tag{1.68}$$

ここで $\mathrm{tr}(R) = r_{11} + r_{22} + r_{33}$ は R のトレースである。

逆に回転軸 ω とその軸周りの回転角度 θ が与えられたら、以下のように回転行列 R を求めることができる。

$$R = I + \frac{\sin\theta}{\theta}[\omega]_\times + \frac{1 - \cos\theta}{\theta^2}[\omega]_\times^2 \tag{1.69}$$

四元数

四つ目は、四元数 (quaternion) である。これは、4 次元の単位ベクトルにより（3 次元空間の）回転を表現するものである。四元数とベクトルと回転は本来は異なるものであるが、同じようにみなすと非常に便利なので、CG やゲームプログラミングでもよく使われている有名な考え方である。

四元数 q は、3 つの虚数単位 i, j, k を用いて以下のように表わされる。

$$q = q_1 + q_2 i + q_3 j + q_4 k \tag{1.70}$$

ここで $i^2 = j^2 = k^2 = 1$ であり、複素数を拡張したようなものである。q_1 は q の実部であり、q_2, q_3, q_4 は 3 つの異なる虚部に相当する。

単位四元数とは、四元数の絶対値を

$$|q| \equiv q_1^2 + q_2^2 + q_3^2 + q_4^2 \tag{1.71}$$

と定義したとき、$|q| = 1$ を満たす四元数のことである。

ある四元数が単位四元数であるとき、以下のように回転行列に変換することができる。

$$R = \begin{pmatrix} q_1^2 + q_2^2 - q_3^2 - q_4^2 & 2q_2q_3 - 2q_1q_4 & 2q_2q_4 + 2q_1q_3 \\ 2q_2q_3 + 2q_1q_4 & q_1^2 - q_2^2 + q_3^2 - q_4^2 & 2q_3q_4 - 2q_1q_2 \\ 2q_2q_4 - 2q_1q_3 & 2q_3q_4 + 2q_1q_2 & q_1^2 - q_2^2 - q_3^2 + q_4^2 \end{pmatrix} \tag{1.72}$$

また、回転軸 $\omega = (\omega_1, \omega_2, \omega_3)^T$ 回りに θ 回転する 3 次元回転を表す単位四元数は、次のように表すことができる。

$$q = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}(\omega_1 i + \omega_2 j + \omega_3 k) \tag{1.73}$$

つまり、単位四元数の実部が回転角に対応し、虚部が回転軸に対応する。

四元数の便利な点は、四元数としての積が、座標の回転行列による変換に対応しているということである。しかしここでは回転行列と四元数との関連が分かれば十分なので、説明は省略しよう。

1.1.6　並進の表現

これまでは、並進移動がない場合の変換を考えてきた。つまり、座標系の変換は回転だけで表現していた。しかし、実際の変換は、並進移動と回転が組み合わさったものである。

ここでは座標系の変換が並進移動 (translation) だけの場合を考えてみよう。回転よりも、ずっと簡単である。

ある点 p の座標系 A における座標を ${}^A\boldsymbol{X}$、座標系 B における座標を ${}^B\boldsymbol{X}$ としよう。座標系 A, B の原点はそれぞれ O_A, O_B である。図 1.7 に示すように、座標系 B の原点 O_B の位置（座標）を座標系 A で書くと、${}^AO_B$ である。これが並進ベクトルである。座標系 A の位置から ${}^AO_B$ だけ座標系を移動すると、座標系 B が得られる。

このとき、図 1.7 からすぐにわかるように、${}^A\boldsymbol{X}, {}^B\boldsymbol{X}, {}^AO_B$ は次の関係にある。

$${}^A\boldsymbol{X} = {}^B\boldsymbol{X} + {}^AO_B \tag{1.74}$$

つまり、座標系 A から見た点 p の座標は、座標系 B から見た点 p の座標に、座標系 B の原点の（A からみた）座標を加えたものである。

すると、点 p の座標の変換はすぐにわかる。座標系 A で表わされた点 p の座標を、座標系 B で表わすには、上の式を単に移項するだけでよい。

$${}^B\boldsymbol{X} = {}^A\boldsymbol{X} - {}^AO_B \tag{1.75}$$

これが意味するのは、座標系 A の原点の（B からみた）座標 ${}^BO_A$ は、${}^AO_B$ の符号を逆にしたもの

$${}^BO_A = -{}^AO_B \tag{1.76}$$

である。反対から見ればマイナス、というのは直感的にもわかりやすい。

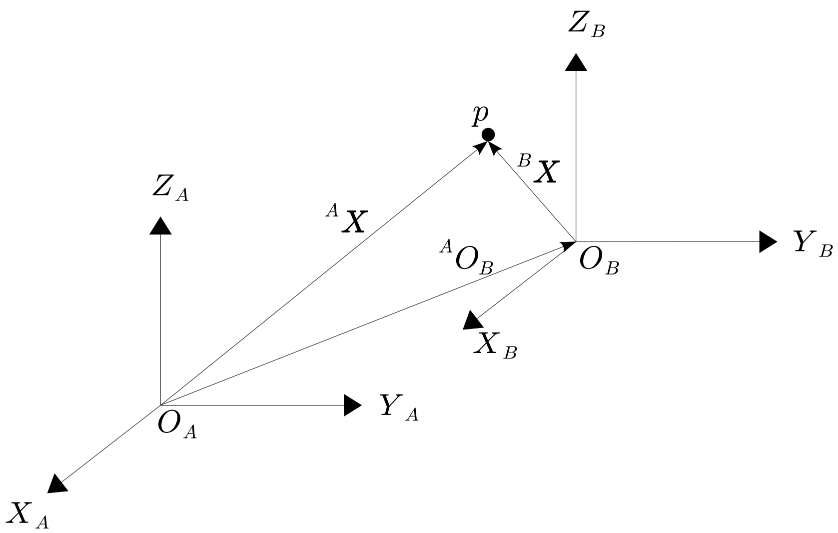


図 1.7: 並進移動。

1.1.7 一般の座標系変換

ようやく一般の 3 次元座標の変換を議論するところまでやってきた。これまでは、並進移動がない回転だけの変換か、回転のない並進だけの変換を考えていた。しかし一般の変換 (transformation) は、座標系の回転と並進移動を同時に行うのが普通である。

今まででてきた記号をおさらいしよう。ある点 p の座標系 A における座標を $^A\boldsymbol{X}$、座標系 B における座標を $^B\boldsymbol{X}$ とする。座標系 A, B の原点はそれぞれ O_A, O_B とする。座標系 A からみた、座標系 B の原点 O_B の座標を ^AO_B とする。座標系 A から B への回転行列を ^B_AR とする。

これらを組み合わせて、座標系 A で表わされた点 p の座標を、座標系 B で表わそう。まず、座標系 A で表わされた点 p の座標 $^A\boldsymbol{X}$ に対して、座標系の並進移動を適用する。並進移動のみの場合を思い出せば、次のようにできる。

$$^A\boldsymbol{X} \longrightarrow {}^A\boldsymbol{X} - {}^AO_B \tag{1.77}$$

次に、座標系の回転を適用する。これは単に回転行列をかければよい。

$$^A\boldsymbol{X} - {}^AO_B \longrightarrow {}^B_AR({}^A\boldsymbol{X} - {}^AO_B) \tag{1.78}$$

これで出来上がりである。

つまり、座標系 B における座標 $^B\boldsymbol{X}$ は次のように表わされる。

$$\begin{aligned} ^B\boldsymbol{X} &= {}^B_AR\ ({}^A\boldsymbol{X} - {}^AO_B) & (1.79)\\ &= {}^B_AR\ {}^A\boldsymbol{X} - {}^B_AR\ {}^AO_B & (1.80) \end{aligned}$$

ここで、

$$^B_AT \equiv -{}^B_AR\ {}^AO_B \tag{1.81}$$

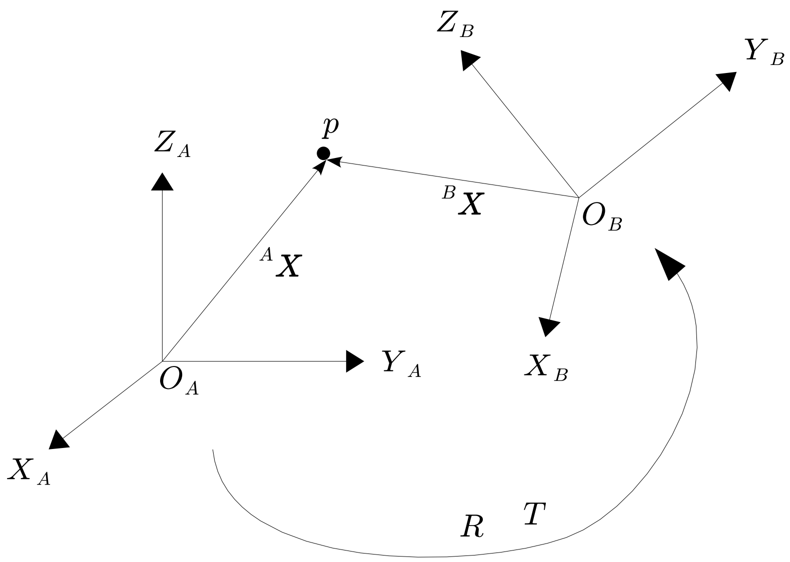


図 1.8: 一般の変換。

とおくと、

$$ {}^B\boldsymbol{X} = {}^B_A R\ {}^A\boldsymbol{X} + {}^B_A T \tag{1.82}$$

と書くことができる。この ${}^B_A T$ を、座標系変換の並進ベクトル (translation vector) と呼ぶ。

注意してほしいのは、この並進ベクトルは座標系 B で表わされているということである。上の変換式の左辺は座標系 B での点 p の座標を表現しているので、右辺もすべて座標系 B で表わされているものでなければならない。だから、座標系を回転行列で回転した後に、単純に ${}^A O_B$ を足すと間違いになる。足しているのは、${}^A O_B$ の符号を反転し ${}^B_A R$ で回転して座標系 B で表わした $-{}^B_A R\ {}^A O_B$ である。この点は理解しておいてほしい。

この先は、特に断らない限りどの座標系で表わしているのかを省略することにする。つまり、座標系変換の回転行列を R、並進ベクトルを T と書くことにするが、どの座標系で表わしているのかを考えておいてほしい。

剛体変換

回転行列 R と並進ベクトル T で表わされる一般の 3 次元変換を、剛体変換 (Rigid-body mition) と呼ぶ。これは特殊ユークリッド変換 (Special Euclidean Transformation)$SE(3)$ とも呼ばれ、次のように定義される。

$$ SE(3) \equiv \{g = (R, T) \mid R \in SO(3),\ T \in \mathbb{R}^3\} \tag{1.83}$$

一般の 3 次元剛体変換 g で座標 $\boldsymbol{X}$ を変換したものを $g(\boldsymbol{X})$ と書くとしよう。剛体変換 g は、二つの座標 $\boldsymbol{X}, \boldsymbol{Y}$ の距離を変えない。つまり、その二つの座標の差のベクトルが距離であるとき、次

のようにそのノルムが保存される。

$$||\boldsymbol{X}-\boldsymbol{Y}|| = ||g(\boldsymbol{X})-g(\boldsymbol{Y})|| \tag{1.84}$$

ではこの剛体変換を具体的に座標変換に適用するにはどうすればよいだろうか？それは次のように、g が表す回転と並進を座標に適用すればよい。

$$R\boldsymbol{X}+T \tag{1.85}$$

しかし、回転だけであれば単に行列の積だけですんだのに、並進も加わってしまったら、ベクトルの和も入ってしまう。これは手間である。手間に見えないだろうか？　たとえば、3 つの座標系変換 g_1, g_2, g_3 を順番に適用することを考えてみよう。もし変換が回転だけであれば、次のように回転行列をかけるだけですんだのである。

$$R_3R_2R_1\boldsymbol{X} \tag{1.86}$$

しかし、それぞれの剛体変換に並進も加わっているので、全部を書きだすと次のようになってしまう。

$$R_3(R_2(R_1(\boldsymbol{X}+T_1)+T_2)+T_3) \tag{1.87}$$

この式を展開するとさらにややこしい。この式を展開したり理解したりするのは、かなりの手間である。

剛体変換の具体的な式を、もっとすっきと簡単に、回転行列をかけるだけですんだように、行列の積だけで書くことができないだろうか？　もしそれができれば、ずいぶん楽に数式を書くことができるのに。

それが、次で説明する同時座標表現というものである。

1.1.8 同次座標系による表現

ある点 p の座標 $\boldsymbol{X}=(X,Y,Z)^T$ の斉次座標 (homogeneous coordinate) 表現もしくは同次座標 (homogeneous coordinate) 表現とは、その座標にもうひとつ要素を加えた以下のようなものである。

$$\boldsymbol{X} \equiv \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \in \mathbb{R}^4 \tag{1.88}$$

ここでは付け加えた要素が 1 であるが、それ以外でも構わない。そのことについては後で述べよう。

このように同次座標で点の座標を表現すると、なにがうれしいのだろうか？それは、次の式を見てみればわかる。回転 R と並進 T で表わされる変換 g を座標 $\boldsymbol{X}=(X,Y,Z)^T$ に適用する場合、次のようにするのであった。

$$R\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}+T \tag{1.89}$$

しかし同時座標表現を使うと、同じことが次のように簡単にできる。

$$\begin{pmatrix} R & T \\ 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + T \\ 1 \end{pmatrix} \tag{1.90}$$

この計算で得られるものは、元の座標 $\boldsymbol{X}$ を座標変換した結果を、同次座標で表現したものと同じである。

つまり、同次座標で表わされた座標の変換を、一つの 4×4 行列で実現することができる。変換に並進移動が含まれていたとしても、非常に簡単に座標変換を書くことができるため、便利である。先ほどと同じ、3 つの座標系変換 g_1, g_2, g_3 を順番に適用する変換の例で見てみよう。ここで、変換 g は次のように 4×4 行列を表すものとする。

$$g = \begin{pmatrix} R & T \\ 0 & 1 \end{pmatrix} \tag{1.91}$$

すると、座標 $\boldsymbol{X}$ が同次座標で表現されていれば、順番に座標変換を施すには次のようにすればよい。

$$g_3 g_2 g_1 \boldsymbol{X} \tag{1.92}$$

これは非常に簡単である。

同次座標表現の一つの問題点は、式を見てそれが同次座標なのかどうなのかが、わかりにくいということである。これが分からないと、式の意味がそもそも違ってしまうことも多い。その式が同次座標で表わされているのか、それとも非同次座標（つまり普通の座標）で表わされているのか、この先注意してみてほしい。

1.1.9　カメラ座標系と世界座標系

さて、一般の 3 次元の座標系の変換のやり方を見てきた。ここまでは数学の話である。これが一体、3 次元復元にどのようにつながるのか？　カメラや画像はどこに登場するのか？

実際、座標系の変換を理解していなければ、3 次元復元やカメラで撮影した画像を理解することなど到底できないのだ。図 1.9(a) を見てほしい。この図は、この 3 次元の実世界に、何かしらの物体や風景や人物やその他興味のある対象が存在しているとして、それをカメラで撮影している様子を表している。カメラでなくとも、あなたが自分の目でその物体をあちこちから眺めている様子を想像してもらってもいい。物体を表から眺めるときもあれば、横に回って眺めたり、後ろからじっくり眺めたり。カメラで撮影する場合も同じである。1 番目のカメラの位置でシャッターを切って 1 つ目の画像を撮影し、2 番目のカメラの位置でシャッターを切って 2 つ目の画像を撮影する。それぞれの位置で見える物体の様子は、それぞれ違うので、N 回シャッターを切れば、N 枚の違う画像を得ることができる。3 次元復元とは、その 2 次元的な平面の画像から、元の 3 次元的な立体の情報を復元することである。

毎回のシャッターを切る時、自分が見ている方向（おそらく前方だろう）に物体がある。先ほど撮ったときにどの方向を向いていたのか、ということについては、気にすることもないし、普通はどの方向だったのか覚えていられないだろう。つまり毎回のカメラの位置や向きは、それ自体独立して「前方向」「上方向」「右方向」などが定義されていることになる。一方、静止している物体から見れば、どの方向からカメラで撮影されようとも、物体が置かれた位置や方向に変化はない。

このように、カメラや物体の位置や方向は、相対的なものであり、それぞれが独立した「向き」を持っている。これが座標系である。カメラの座標系をカメラ座標系 (camera coordinate system) といい、物体の座標系を物体座標系 (object coordinate sytem) という。相対的とはいうものの、まったく関係がないわけではない。さっき撮ったカメラの位置から、どのくらい移動してどのくらい回転したら、次のカメラの位置と向きがえられるのか。これが座標系の変換である。そして、すべての座標系の相互の変換を統一して行うために決めておくのが、世界座標系 (world coordinate sytem) である。すべての座標系はこれを参照しながらお互いの位置関係を表現するので、参照座標系 (reference coordinate system, reference frame) とも呼ばれる。

図 1.9(b) には、世界座標系 O_W、物体座標系 O_o、各カメラ座標系 O_i $(i = 1, 2, 3, \ldots, N)$ の関係を表している。世界座標系 O_W からカメラ座標系 O_i への変換を ${}^i_W g$ とすると、ある点 p の世界座標系での座標 ${}^W\boldsymbol{X}$ は、次のように各カメラ座標系に変換することができる。

$$ {}^1\boldsymbol{X} = {}^1_W g \ {}^W\boldsymbol{X} \tag{1.93} $$

$$ {}^2\boldsymbol{X} = {}^2_W g \ {}^W\boldsymbol{X} \tag{1.94} $$

$$ {}^3\boldsymbol{X} = {}^3_W g \ {}^W\boldsymbol{X} \tag{1.95} $$

カメラで物体を撮影するときには、そのカメラの座標系でしか座標を表すことができない。いろいろなカメラ位置で撮影した情報をもとに、世界座標系で表わされた座標を逆算する。それが 3 次元復元である。その前に、「カメラで撮影する」ということについて、数学的に見ておこう。

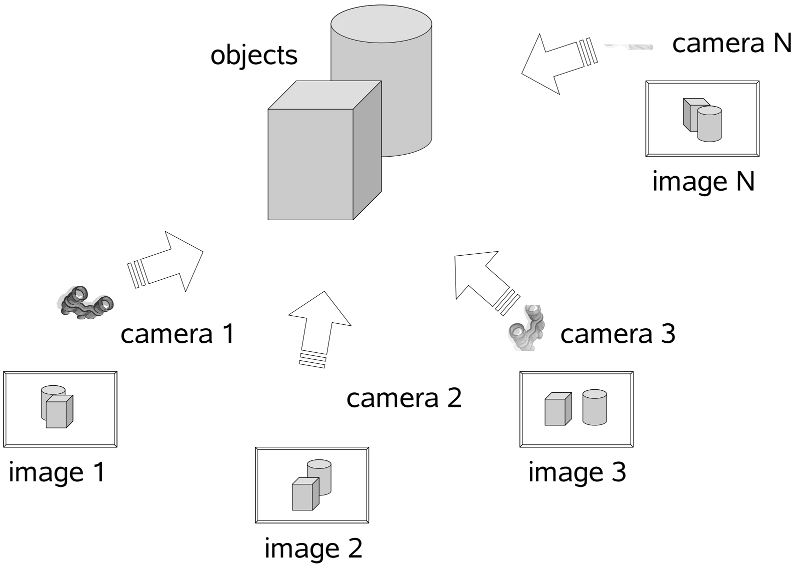
1.1.10 右手系左手系

O(3) など？

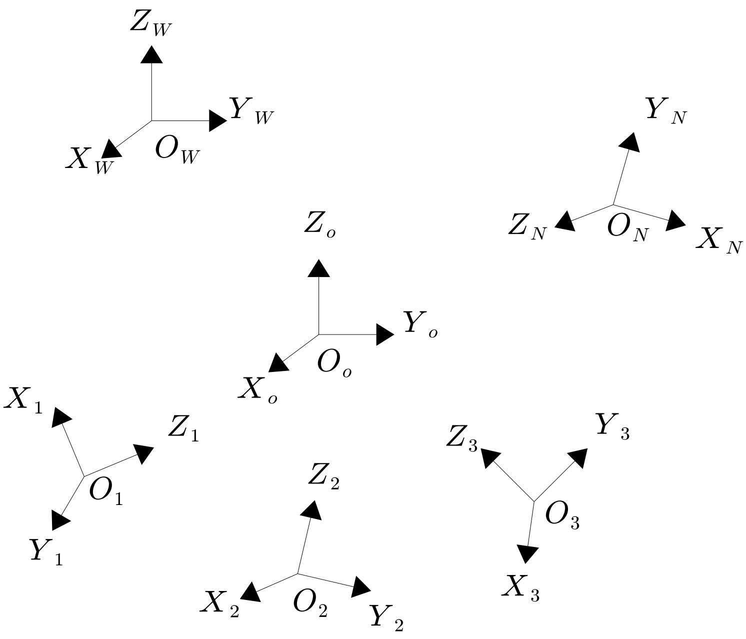
1.2 世界をカメラに写し撮る

ここでは、カメラで撮影するということがどのように数学的に表現されるのかを説明しよう。これまでの説明は、3 次元と 3 次元の座標系の変換であった。しかしこれからは、3 次元世界の立体を 2 次元世界の画像に変換する。つまり、3 次元と 2 次元の座標系の変換である。

同次座標系というものを忘れていたら、前の説明を見返して思い出しておいてほしい。理科で習った透視投影やピンホールカメラの仕組みも、読み進む前に、できれば記憶の片隅から掘り返したほうがいいだろう。



(a)



(b)

図 1.9: 世界座標系とカメラ座標系。

1.2.1 カメラの撮像過程

ピンホールカメラモデル (pinhole camera model) は、もっともよくつかわれるカメラモデルである。ピンホールカメラの原理は、一点だけ開けた穴から外の風景をスクリーンに投影することである。すべての風景はその穴をとおってスクリーンに投影される。これはレンズを考えない、理想的なカメラモデルである。

それでは、図 1.10(a) に示すように、ある点 p の 3 次元座標 $\boldsymbol{X}$ をを 2 次元の平面に投影することを考えよう。カメラ座標系の原点 O がピンホールカメラの穴に相当する。そのため、O をカメラ中心と呼ぶこともある。p と O を結ぶ直線をそのままのばして 2 次元スクリーンにぶつかったところが、投影された 2 次元座標 $\boldsymbol{x}$ である。

このピンホールカメラモデルによる点の投影を、透視投影 (perspective projection) という。3 次元座標を $\boldsymbol{X} = (X, Y, Z)^T$、2 次元座標を $\boldsymbol{x} = (x, y)^T$ とすると、その間の関係は次の透視投影の基本的な式で表わされる。

$$x = f\frac{X}{Z} \tag{1.96}$$

$$y = f\frac{Y}{Z} \tag{1.97}$$

ここで f は、カメラの中心 O からスクリーンまでの距離で、（ピンホールカメラモデルの）焦点距離 (focal length) と呼ばれる。

このピンホールカメラモデルでは、スクリーンに映ったその風景が上下左右がひっくり返ってしまう。写真を撮影するときには何の問題もないが、図で透視投影を理解しようとするときには、カメラ座標系とスクリーンの上下がひっくり返っていてはわかりにくいことがおおい。そこで、図 1.10(b) のように、スクリーンを仮想的にカメラの前に持ってくるカメラモデルが、一般的に説明に使われる。どちらも数式は同じであることを、納得しておこう。

1.2.2 同次座標表現：再訪

さてここで、同次座標表現を使って透視投影の式を表わしてみよう。

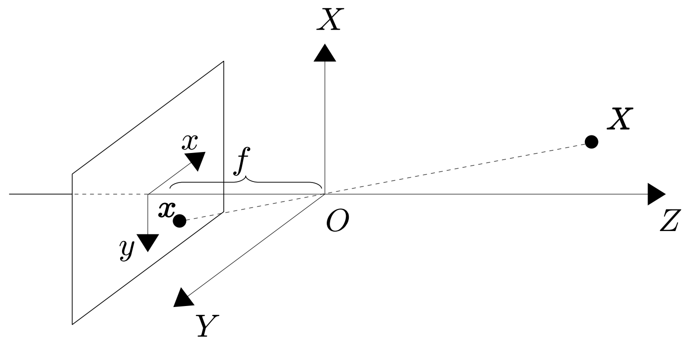
上で示した透視投影の数式は、x と y が別々の式で表わされている。しかし行列を使うと、2 次元の座標 $\boldsymbol{x}$ として一度に数式で表わすことができる。

$$\boldsymbol{x} = \begin{pmatrix} x \\ y \end{pmatrix} = \frac{f}{Z}\begin{pmatrix} X \\ Y \end{pmatrix} \tag{1.98}$$

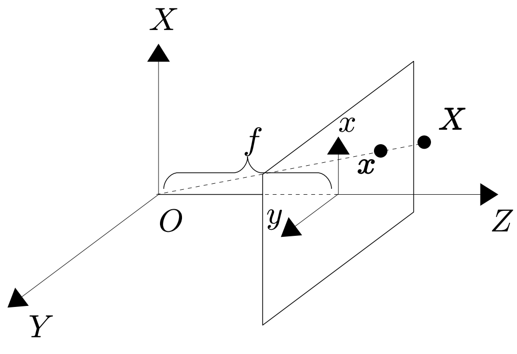
しかしこのままでは、3 次元座標 $\boldsymbol{X}$ を数式の中に入れることは難しい。X も Y も Z も、一つのベクトルとして表わされていないので、いちいち 3 次元座標をばらばらにしなければ、数式の中に入れられないのでは、面倒極まりない。

そこで、$\boldsymbol{X}$ の 3 次元同次座標表現 $(X, Y, Z, 1)^T$ を使った次の式を考えてみよう。

$$\begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{1.99}$$



(a)



(b)

図 1.10: ピンホールカメラモデル。

この行列とベクトルの積を計算した結果は次のようになる。

$$\begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} \tag{1.100}$$

もしこれが、3 次元座標を投影した結果の 2 次元座標であれば、透視投影が行列の積だけで表わすことができて、便利である。

それば便利であるのなら、そうである、と定義してしまえば簡単である。つまり、$(fX, fY, Z)^T$ が $(f\frac{X}{Z}, f\frac{Y}{Z})^T$ と同じであるとみなす約束をするのである。だがしかし、これをどのようにすればよいだろう。

ここで、同次座標表現をもう一度、しかし正しく、説明しよう。2 次元座標の同次座標表現 $\boldsymbol{x} = (x, y, w)^T$ とは、2 次元座標 $(\frac{x}{w}, \frac{y}{w})^T$ を表すものとする。つまり、一つ付け加えた要素 w によって、ほかのすべての要素を割るのである。これは 3 次元の同次座標表現でも同じである。3 次元座標の同次座標表現 $\boldsymbol{X} = (X, Y, Z, W)^T$ とは、3 次元座標 $(\frac{X}{W}, \frac{Y}{W}, \frac{Z}{W})^T$ を表すものとする。$W = 1$ のとき、つまり同次座標表現 $\boldsymbol{X} = (X, Y, Z, 1)^T$ は座標 $(X, Y, Z)^T$ を表わしている。これが以前の説明で出てきた内容であった。

それでは、先ほどの式を見てみよう。(fX, fY, Z) を、投影した結果の 2 次元座標の同次座標表現だと思えば、それは $(f\frac{X}{Z}, f\frac{Y}{Z})$ を表していることになる。先ほどの式をまとめて書くと、以下のように 3 次元同次座標が 2 次元同次座標に変換されているのが分かるだろう。

$$\begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{1.101}$$

この数式をすこし変形すると、以下のような簡単な結果が得られる。

$$Z\boldsymbol{x} = P_p \boldsymbol{X} \tag{1.102}$$

ここで

$$\boldsymbol{x} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} f\frac{X}{Z} \\ f\frac{Y}{Z} \\ 1 \end{pmatrix} \tag{1.103}$$

$$\boldsymbol{X} = \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{1.104}$$

$$P_p = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{1.105}$$

である。この 3×4 行列 P_p を、透視投影をあらわす射影行列 (projection matrix) という。

この数式からわかることは、2 次元座標 $\boldsymbol{x}$ が与えられた時、焦点距離 f と 3 次元座標の Z 成分が分かれば、3 次元座標 $\boldsymbol{X}$ を計算できる、ということである。焦点距離 f はカメラのズーム状態に依存するパラメータである。実際にはこれ以外にもいくつかのパラメータが存在するので、それらをすべて知っておかなければならない。それを求めるための方法は後で説明しよう。

それでは Z は分かっているのであろうか。少し考えてみよう。カメラ中心 O と点 p を結ぶ直線上に存在する点は、すべて同じスクリーン上の 2 次元座標に投影されてしまう。つまり、$\boldsymbol{x}$ が分かっていても Z の値が何なのかは、分からないのだ。この分からない Z 成分を、λ と書くことにする。式で表わせば次のようになる。

$$\lambda \boldsymbol{x} = P_p \boldsymbol{X} \tag{1.106}$$

この λ をその点の奥行 (depth) という。この分からない奥行きを求めることが、3 次元復元の核となる部分である。

1.2.3 射影空間 P^2

上で見たように、透視投影を数式で表現するには、同次座標表現を使うと簡単になる。この先、同次座標表現をいろいろな形で使うため、ここで整理してその特性をまとめておこう。ややこしい 3 次元の同次座標表現の前に、まずは 2 次元の同次座標表現を説明しよう。

ある 2 次元での点の同次座標表現を

$$\boldsymbol{x} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{1.107}$$

とする。たとえばこれを定数倍してみると、

$$\lambda \boldsymbol{x} = \begin{pmatrix} \lambda x \\ \lambda y \\ \lambda \end{pmatrix}, \quad \lambda \in \mathbb{R}^+ \tag{1.108}$$

である。ここで $\mathbb{R}^+ = \mathbb{R}/0$、つまり 0 より大きい正の実数を表す。この二つの同次座標表現 $\boldsymbol{x}, \lambda\boldsymbol{x}$ が表す実際の座標を考えると、どちらも $(x, y)^T = (\lambda\frac{x}{\lambda}, \lambda\frac{y}{\lambda})^T$ である。つまり同次座標表現においては、定数倍は意味を持たないことになる。2 倍であろうが 0.1 倍であろうが、どちらも同じ座標を指していることには変わりわない。

このことを、同次座標表現は定数倍が不定である (up to scale) といい、

$$\boldsymbol{x} \sim \lambda \boldsymbol{x} \tag{1.109}$$

と表す。$\sim$ は、$=$ ではないが同次座標表現として等しいという意味である。このような 2 次元の同次座標で表現された点の集合がなす空間を、2 次元の射影空間 (projective space)P^2 という。

ここで、最後の要素が 0 である次のような同次座標を考えてみよう。

$$\boldsymbol{x}_\infty = \begin{pmatrix} x \\ y \\ 0 \end{pmatrix} \tag{1.110}$$

定義からすると、この同次座標が表す座標を求めるには、0 で割らなければならない。そんなことは数学では許されていないので、これをどう考えたらいいだろうか。もし 0 でないが非常に小さい値が最後の要素に入っていたらどうだろうか。その場合、座標の要素は非常に大きな値になる。つまり原点から非常に遠い座標を表していることになる。そこで、最後の要素が 0 である $\boldsymbol{x}_\infty$ を、無限遠点 (point at infinity) もしくは理想点 (ideal point) と呼ぶ。つまり、通常のユークリッド空間では定義できない無限遠にある点を、射影空間では定義できるのである。

射影空間は空間であるので、直線も定義できる。いったい、どうやって直線を同次座標で表現するのだろうか？　通常の 2 次元空間の直線の方程式を思い出してほしい。次のような多項式である。

$$ax + by + c = 0 \tag{1.111}$$

これは右辺が 0 であるので全体を定数倍しても変わらない。つまり、先ほどの式が成り立つなら、任意の $\lambda \in \mathbb{R}^+$ に対して、次の式も成り立つ。

$$\lambda ax + \lambda by + \lambda c = 0 \tag{1.112}$$

つまり、直線の方程式の係数をベクトルにした

$$\boldsymbol{\ell} = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \tag{1.113}$$

は、定数倍が不定であり、

$$\boldsymbol{\ell} \sim \lambda \boldsymbol{\ell} \tag{1.114}$$

が成り立つ。点の同次座標を $\boldsymbol{x}$ で表わすなら、直線を表す同次座標を $\boldsymbol{\ell}$ としよう。

点と直線の定義から、以下のような関係を導くことができる。

- 点 $\boldsymbol{x}$ が直線 $\boldsymbol{\ell}$ 上にある $\Longleftrightarrow \boldsymbol{x}^T\boldsymbol{\ell} = \boldsymbol{\ell}^T\boldsymbol{x} = 0$

  これは簡単に説明できるだろう。展開すれば、$\boldsymbol{x}^T\boldsymbol{\ell} = ax + by + c = 0$ が得られる。

- 直線 $\boldsymbol{\ell}_1$ と $\boldsymbol{\ell}_2$ が点 $\boldsymbol{x}$ で交差する $\Longleftrightarrow \boldsymbol{x} = \boldsymbol{\ell}_1 \times \boldsymbol{\ell}_2$

  二本の直線が一点で交わっているということとは、その点が両方の直線上に存在するということである。つまり、

  $$\boldsymbol{\ell}_1^T\boldsymbol{x} = \boldsymbol{\ell}_2^T\boldsymbol{x} = 0 \tag{1.115}$$

  一方、内積と外積の関係から、

  $$\boldsymbol{\ell}_1^T(\boldsymbol{\ell}_1 \times \boldsymbol{\ell}_2) = \boldsymbol{\ell}_2^T(\boldsymbol{\ell}_1 \times \boldsymbol{\ell}_2) = 0 \tag{1.116}$$

  したがって $\boldsymbol{x} = \boldsymbol{\ell}_1 \times \boldsymbol{\ell}_2$ である。

- 直線 $\boldsymbol{\ell}$ が 2 点 $\boldsymbol{x}_1, \boldsymbol{x}_2$ を通る $\Longleftrightarrow \boldsymbol{\ell} = \boldsymbol{x}_1 \times \boldsymbol{x}_2$

  これは先ほどと逆に考えればよい。

  $$\boldsymbol{x}_1^T(\boldsymbol{x}_1 \times \boldsymbol{x}_2) = \boldsymbol{x}_2^T(\boldsymbol{x}_1 \times \boldsymbol{x}_2) = 0 \tag{1.117}$$

この性質から、面白いことをしてみよう。二本の平行な直線を

$$\boldsymbol{\ell}_1 = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \tag{1.118}$$

$$\boldsymbol{\ell}_2 = \begin{pmatrix} a \\ b \\ c' \end{pmatrix} \tag{1.119}$$

とする。直線の傾きはどちらも同じであり、切片だけがちがうので、この二本の直線は、普通は交わらない。しかし射影空間では無限遠点が定義されているので、二本の直線はどこかの無限遠点で交わることを示してみよう。無限遠点の集合を $\{\boldsymbol{x}_\infty\}$ として、次の式で交点を計算することができる。

$$\boldsymbol{\ell}_1 \times \boldsymbol{\ell}_2 = (c' - c) \begin{pmatrix} b \\ -a \\ 0 \end{pmatrix} \in \{\boldsymbol{x}_\infty\} \tag{1.120}$$

この交点の最後の要素は 0 なので、確かに無限遠点である。

さらにややこしくしてみよう。無限遠点 $\boldsymbol{x}_\infty$ が直線上に乗っているとしたら、それはどんな直線だろうか？　次のような直線を考えてみよう。

$$\boldsymbol{\ell}_\infty = \begin{pmatrix} 0 \\ 0 \\ c \end{pmatrix} \tag{1.121}$$

すると、無限遠点を $\boldsymbol{x}_\infty = (a, b, 0)^T$ としたとき、

$$\boldsymbol{\ell}_\infty^T \boldsymbol{x}_\infty = 0 \tag{1.122}$$

である。すなわち、無限遠点 $\boldsymbol{x}_\infty$ は直線 $\boldsymbol{\ell}_\infty$ を通る。この直線を無限遠直線 (line at infinity) という。二つの無限遠点を $\boldsymbol{x}_{\infty 1} = (a_1, b_1, 0)^T, \boldsymbol{x}_{\infty 2} = (a_2, b_2, 0)^T$ としたときに、

$$\boldsymbol{x}_{\infty 1} \times \boldsymbol{x}_{\infty 2} \in \{\boldsymbol{\ell}_\infty\} \tag{1.123}$$

である。つまり、二つの無限遠点を結ぶ直線は無限遠直線である。また、ある直線を $\boldsymbol{\ell} = (a, b, c)^T$ とすると、

$$\boldsymbol{\ell} \times \boldsymbol{\ell}_\infty \in \{\boldsymbol{x}_\infty\} \tag{1.124}$$

である。つまり、無限遠直線は任意の直線 $\boldsymbol{\ell}$ と無限遠点で交差する。

通常のユークリッド空間と射影空間の違いを簡単に表 1.1 に示しておく。無限遠点や無限遠直線の関係式は、数学的なお遊びだと思って、頭を鍛えてほしい。数学には創造力が必要なのである。

表 1.1: ユークリッド空間と射影空間。

ユークリッド空間
2 つの（並行ではない）直線はある 1 点で交差する
2 つの点はある直線を決める

射影空間
任意の 2 つの直線はある 1 点で交差する
（無限遠点を含む）任意の 2 つの点はある直線を決める

1.2.4 射影空間 P^3

次は 3 次元の同次座標表現である。これは、2 次元の場合とまったく同じように考えればよい。ある 3 次元での点の同次座標表現を

$$\boldsymbol{X} = \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{1.125}$$

とする。これは 3 次元の座標 $(\frac{X}{1}, \frac{Y}{1}, \frac{Z}{1})^T = (X, Y, Z)^T$ を表わすのであった。これを定数倍した $\lambda\boldsymbol{X}$ も、同じ座標 $(\lambda\frac{X}{\lambda}, \lambda\frac{Y}{\lambda}), \lambda\frac{Y}{\lambda})^T = (X, Y, Z)^T$ である。つまり同次座標表現においては、定数倍しても表す座標は同じである。

このことを、同次座標表現は定数倍が不定である (up to scale) といい、

$$\boldsymbol{X} \sim \lambda\boldsymbol{X} \tag{1.126}$$

と表す。$\sim$ は、$=$ ではないが同次座標表現として等しいという意味である。このような 3 次元の同次座標で表現された点の集合がなす空間を、3 次元の射影空間 (projective space)P^3 という。

最後の要素が 0 である次のような同次座標

$$\boldsymbol{X}_\infty = \begin{pmatrix} X \\ Y \\ Z \\ 0 \end{pmatrix} \tag{1.127}$$

を、無限遠点 (point at infinity) もしくは理想点 (ideal point) と呼ぶ。

2 次元の射影空間 P^2 では、直線を同次座標表現で表わしてみた。では 3 次元の射影空間 P^3 では、どうだろうか。次のような多項式を考えてみよう。

$$aX + bY + cZ + d = 0 \tag{1.128}$$

これは通常の 3 次元空間の平面の方程式である。そこで、この平面を

$$\Pi = \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \tag{1.129}$$

で表わすと、先ほどの平面の方程式は

$$\Pi^T \boldsymbol{X} = 0 \tag{1.130}$$

と書くことができる。この平面の方程式を定数倍しても、もちろん成り立つため、平面の同次座標表現 Π は、定数倍が不定である。つまり

$$\Pi \sim \lambda \Pi \tag{1.131}$$

である。

それでは、平面と点の関係を同次座標表現で表わしてみよう。

- 点 $\boldsymbol{X}$ がある平面 Π 上にある $\Longleftrightarrow$ $\boldsymbol{X}^T \Pi = \Pi^T \boldsymbol{X} = 0$
- 点 $\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3$ がある平面 Π 上にある $\Longleftrightarrow$

$$\begin{pmatrix} \boxed{\boldsymbol{X}_1^T} \\ \boxed{\boldsymbol{X}_2^T} \\ \boxed{\boldsymbol{X}_3^T} \end{pmatrix} \boxed{\Pi} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \boldsymbol{0} \tag{1.132}$$

- 3 つの平面 Π_1, Π_2, Π_3 はある 1 点 $\boldsymbol{X}$ で交わる $\Longleftrightarrow$

$$\begin{pmatrix} \boxed{\Pi_1^T} \\ \boxed{\Pi_2^T} \\ \boxed{\Pi_3^T} \end{pmatrix} \boxed{\boldsymbol{X}} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \boldsymbol{0} \tag{1.133}$$

P^2 での無限遠直線と同じような考え方で、P^3 では次のような無限遠平面 (plane at infinity) が定義できる。

$$\Pi_\infty = \begin{pmatrix} 0 \\ 0 \\ 0 \\ d \end{pmatrix} \tag{1.134}$$

しかし、これ以上ややこしくするのはやめておこう。

1.2.5　カメラ座標系での射影

同次座標表現と射影空間をなんとか乗り切ったら、ようやく本来やりたいことの説明ができる。何をしたかったのかというと、3 次元空間の点を 2 次元平面上に投影したいのであった。単に透視投影の式を同次座標で表現するのであれば、それはすでに説明している。問題は、世界座標系とカメラ座標系の間の座標変換も考えに入れなければならない、ということである。

ある点 p の、世界座標系での座標を

$$\boldsymbol{X}_0 = \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{pmatrix} \tag{1.135}$$

としよう。これを、カメラで撮影したときに、画像という 2 次元平面上の座標で表わさなければならないのだ。

まずこの座標を、カメラ座標系で表わすことにしよう。世界座標系からカメラ座標系への変換を g とすると、点 p のカメラ座標系での座標 $\boldsymbol{X}$ は次のようになるのであった。

$$\boldsymbol{X} = \begin{pmatrix} R & T \\ 0 & 1 \end{pmatrix} \boldsymbol{X}_0 = g\boldsymbol{X}_0 \tag{1.136}$$

これはカメラでみた座標であるので、これを透視投影する。つまり、透視投影の射影行列をかければよい。

$$\begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \boldsymbol{X} = P_p\boldsymbol{X} \tag{1.137}$$

この射影行列は、3 次元の同次座標を 2 次元の同次座標に変換する 3×4 行列であったので、出来上がった $P_p\boldsymbol{X}$ は 2 次元の同次座標である。つまり、この結果は次の式で表わされることになる。

$$\boldsymbol{x} \sim P_p\boldsymbol{X} \tag{1.138}$$

思い出してほしいのは、同次座標表現では定数倍が不定であることである。そのため、右辺と左辺は $=$ ではなく $\sim$ で結ばれている。もしこれを $=$ にするのであれば、不定である（未知である）奥行き λ を使って、次のように書くことになる。

$$\lambda\boldsymbol{x} = P_p\boldsymbol{X} \tag{1.139}$$

ここで、透視投影の射影行列 P_p を次のように分解しておこう。

$$P_p = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{1.140}$$

$$= \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{1.141}$$

ここで

$$K_f = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{1.142}$$

$$P_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{1.143}$$

とおくと、

$$P_p = K_f P_0 \tag{1.144}$$

と書くことができる。P_0 を標準射影行列 (standard projection matrix) と呼ぶことにしよう。

この分解を使うと、先ほどの透視投影の式 $\lambda \boldsymbol{x} = P_p \boldsymbol{X}$ は、結局次のように書くことができる。

$$\lambda \boldsymbol{x} = K_f P_0 g \boldsymbol{X}_0 \tag{1.145}$$

これが、理想的なピンホールカメラモデルにおける透視投影の基本的な式である。

この式の変換を一つずつ見てみよう。

- 世界座標系での座標 $\boldsymbol{X}_0$ が、変換 g によってカメラ座標系に変換される。

  $\boldsymbol{X}_0 \longrightarrow g\boldsymbol{X}_0$

- 透視投影によって射影される。

  $g\boldsymbol{X}_0 \longrightarrow P_0 g\boldsymbol{X}_0$

- カメラのズーム（つまり焦点距離）により拡大縮小される。

  $P_0 g\boldsymbol{X}_0 \longrightarrow K_f P_0 g\boldsymbol{X}_0$

変換 g は、世界座標系に対して、カメラがどこの位置にありどの向きを向いているのかを表わしている。これはカメラの焦点距離には依存していない。今持っているカメラを、別のもっと高性能のカメラに変えたとしても、変換 g はそれに影響されることはない（すくなくともこの式の上では）。そのため、変換 g（つまりそれが表す回転 R と並進 T）を、カメラの外部パラメータ (extrinsic camera paraemters) と呼ぶ。

それに対して、カメラのズーム設定を変えると値が変わってしまう焦点距離を、カメラの内部パラメータ (intrinsic camera parameters) と呼ぶ。内部パラメータは、カメラの内部をどのようにモデル化するのかによって、焦点距離のほかにもいろいろなものがある。次に説明するのは、一般的に用いられているカメラモデルとその内部パラメータである。

1.2.6　カメラ内部パラメータ

カメラの中には、いろいろな部品が複雑に組み合わさっていて、現実のカメラのモデルを正確に再現するようなことは、無理である。だから、ピンホールカメラモデルという簡単な透視投影のモ

デルを考えてきた。しかし、これだけではまだうまくいかないことがある。それはカメラの中にある 2 次元スクリーンに相当する部分である。

ピンホールカメラによる透視投影では、カメラ座標系に変換された座標を透視投影行列によって 2 次元座標に変換していた。その 2 次元座標系は、図 1.11(a) で示されているように、真ん中に原点 O がある。しかし普通カメラで撮った画像を処理するときに、得られる座標とは、その座標系であるとは限らない。左上隅を原点 O' があり、しかも長さの単位は画素 (pixel) である。カメラ内部パラメータは、この 2 次元での座標変換を表すために使われるものである。焦点距離 f もその一つと考えてよい。

まず、透視投影されただけの 2 次元座標を $(x, y)^T$ としよう。これは元の 3 次元座標に、g と P_0 と K_f をかけたものである。そのため、この 2 次元座標の単位は元の 3 次元座標と同じと考えてよい。元の 3 次元座標が cm で測定されていれば $(x, y)^T$ も cm であるし、もとが mm であればこれも mm である。

次に、その座標系と原点は同じであるが単位が画素である座標系で、同じ座標を表したものを $(x_s, y_s)^T$ とする。つまり次のような行列をかけることに相当する。

$$\begin{pmatrix} x_s \\ y_s \end{pmatrix} = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{1.146}$$

ここで s_x, s_y はそれぞれ x, y 方向の定数 (scale factor) であり、単位長さあたりの画素数 (pixels per unit) を意味する。たとえば、2 次元座標上で 1cm 分が 5 画素を占めるときは $s_x = 5$ であり、x が 10cm ならば $x_s = 50$pixel になる。

最後に、原点を移動した座標系での座標を $(x', y')^T$ とする。原点の並進移動を $(O_x, O_y)^T$ とすると、次のようになる。

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x_s \\ y_s \end{pmatrix} + \begin{pmatrix} O_x \\ O_y \end{pmatrix} \tag{1.147}$$

この並進移動の単位は pixel であることに注意。たとえば画像の大きさが 640×480 で、この画像の中心にもし原点 O があるのなら、$(O_x, O_y)^T = (320, 240)^T$ である。

結局、これらの座標系変換をまとめると以下のようになる。

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} O_x \\ O_y \end{pmatrix} \tag{1.148}$$

さらに、これを 2 次元の同次座標で表現しよう。すると、次のように行列の積だけで書くことができる。

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & 0 & O_x \\ 0 & s_y & O_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{1.149}$$

ふつうはこれだけで十分なのだが、あと一つだけ内部パラメータを加えることがある。図 1.11(b) のように、x' 軸と y' 軸が直交せずにずれている場合を考えるのだ。その軸のなす角度を θ とする

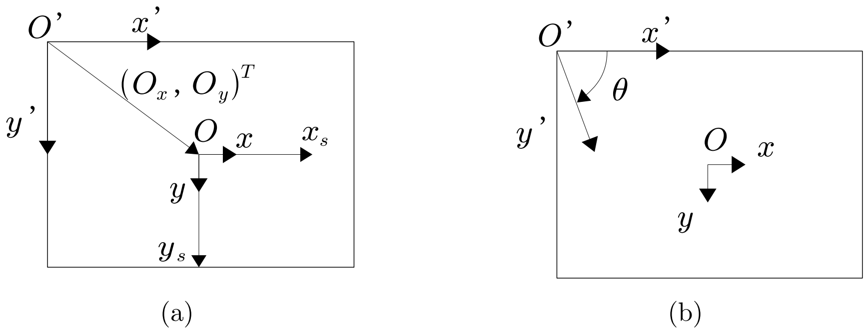


図 1.11: カメラ内部パラメータ。

と、先ほどの式は次のように書きなおすことができる。

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & s_\theta & O_x \\ 0 & s_y & O_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{1.150}$$

つまり軸がずれた分を表す定数 (skew factor) s_θ が行列の要素として加わるだけである。s_θ は $\cot\theta = \frac{1}{\cos\theta}$ に比例し、実は同時に s_y は $\sin\theta$ に比例するようになる。しかしここでは詳しくは触れないことにしよう。実用上は $\theta = 90$ 度と思って問題ないし、つまり $s_\theta = 0$ と考えてよい。このように考えるメリットは、行列の形がほとんどそのままで、軸のずれという問題を一緒に考えることができる、というだけである。本当はほかにも考えなければならない内部パラメータはたくさんある。たとえば歪みの度合がそれだ。しかしそれらは、このような簡単な行列の形には書けないため、3 次元復元の定式化の中で一緒に考えることはあまりないし、ここでは考えないことにしよう。

あらためて、先ほどの式の行列を以下のようにおく。

$$K_s = \begin{pmatrix} s_x & s_\theta & O_x \\ 0 & s_y & O_y \\ 0 & 0 & 1 \end{pmatrix} \tag{1.151}$$

この行列 K をカメラの内部行列 (camera intrinsic matrix) といい、$s_x, s_y, O_x, O_y, s_\theta$ と焦点距離 f をカメラの内部パラメータという。

この内部行列を、先ほどの理想的なピンホールカメラモデルにおける透視投影の基本的な式に適用しよう。最終的に得られる画像の画素単位での座標を $\boldsymbol{x}' = (x', y')^T$ とすると、透視投影の式は次のようになる。

$$\lambda \boldsymbol{x}' = K_s K_f P_0 g \boldsymbol{X}_0 \tag{1.152}$$

具体的に書いてみよう。

$$\lambda \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & s_\theta & O_x \\ 0 & s_y & O_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} R & T \\ 0 & 1 \end{pmatrix} \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{pmatrix} \tag{1.153}$$

この式を眺めていると、透視投影は基本的に 2 つの変換で表現されていることが分かる。1 つ目は外部パラメータ、2 つ目は内部パラメータである。外部パラメータによる変換は、次のように R, T と透視投影からなる。

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} R & T \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} R & T \end{pmatrix} \tag{1.154}$$

内部パラメータによる変換は、K_s と K_f からなる。それを K とおくと、

$$K \equiv K_s K_f = \begin{pmatrix} s_x & s_\theta & O_x \\ 0 & s_y & O_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} fs_x & fs_\theta & O_x \\ 0 & fs_y & O_y \\ 0 & 0 & 1 \end{pmatrix} \tag{1.155}$$

これらを用いると、先ほどの式は次のように簡単になる。

$$\lambda \boldsymbol{x}' = K \bigl(R\ T\bigr) \boldsymbol{X}_0 \tag{1.156}$$

ここで

$$P \equiv K \bigl(R\ T\bigr) \tag{1.157}$$

とおくと、さらに簡単に書くことができる。

$$\lambda \boldsymbol{x}' = P \boldsymbol{X}_0 \tag{1.158}$$

このような行列 P を、射影行列 (projection matrix) と呼ぶ。射影行列は、世界座標系での点の座標 $\boldsymbol{X}_0$ を、画像の画素単位の座標 $\boldsymbol{x}'$ に変換する、便利で一般的なものである。

上の式は、同次座標表現であるので、次のように書いてもいい。

$$\boldsymbol{x}' \sim P \boldsymbol{X}_0 \tag{1.159}$$

この先、この式をもとに 3 次元復元を考えていくことになる。この式の意味をもう一度よく理解しておいてほしい。

1.2.7 他の射影行列

透視投影　正射影（平行投影）　弱透視投影　疑似透視投影

1.3 カメラ校正

透視投影の基本式が分かったところで、ようやく 3 次元復元の問題を記述することができる。しかしその前に、カメラのパラメータを求めておかなければ、3 次元復元ができないのだ。カメラの外部パラメータと内部パラメータを求めることを校正（較正、キャリブレーション (calibrarion)）という。

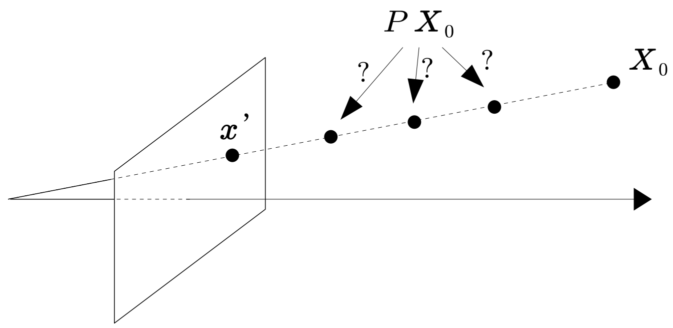


図 1.12: 同次座標表現はどの点に対応するか？

校正とは、実験に用いる測定道具を測定しておくことである。ある物差しで測った物体の長さが 4 目盛り分だとしても、その物差しの目盛りが 1mm なのか 1cm なのか 7.33456mm なのか分からなければ、物体の実際の長さは分からない。

カメラの場合にも、あらかじめ外部パラメータと内部パラメータを求める必要がある。3 次元復元のためには、後で述べるように内部パラメータのみが必要である。外部パラメータはカメラ座標系の位置姿勢を表すため、画像を撮影するたびに変わってしまう。カメラを手で持って撮影した場合には、キャリブレーションしたときの外部パラメータとそっくりおなじ外部パラメータを実現できる保証はない。一方、内部パラメータはカメラの設定を保存しておけば、多くの場合いつでも再現できる。焦点距離はズームの設定を固定しておけばよい。その他の内部パラメータは、おそらくカメラをバラバラに分解することをしなければ、カメラを買ってきてから変化することはないだろう。

以下では、内部パラメータ K と外部パラメータ R, T を求める方法を説明する。与えられるのは、元の 3 次元座標 $\boldsymbol{X}_0$ と、それが投影された座標 $\boldsymbol{x}'$ である。

1.3.1　射影行列を用いた方程式を立てる

まずは透視投影の同次座標表現を思い出してほしい。3 次元の同次座標 $\boldsymbol{X}_0$ を射影行列 P により 2 次元の同次座標 $\boldsymbol{x}'$ に変換するには、次の式を用いればよかった。

$$\boldsymbol{x}' \sim P\boldsymbol{X}_0 \tag{1.160}$$

同次座標表現では定数倍は不定だったので、$P\boldsymbol{X}_0$ を何倍しても、投影される先の 2 次元座標はおなじ $\boldsymbol{x}'$ である。

これをベクトルとしてみてみよう。図 1.12 に示すように、$\boldsymbol{X}_0$ を投影した点が $\boldsymbol{x}'$ であるとする。この図からわかるように、原点から点 $\boldsymbol{X}_0$ に伸ばした直線上に、投影点 $\boldsymbol{x}'$ は存在する。したがって $\boldsymbol{x}'$ を何倍しても、それは $\boldsymbol{X}_0$ に伸びた直線上にあることは変わりがない。同様に $P\boldsymbol{X}_0$ も、その直線上のどこにあるのかはわからない。これは同次座標表現の幾何学的な意味である。

すると、$\boldsymbol{x}'$ と $P\boldsymbol{X}_0$ が同じ方向を向いているということから、その二つの外積を取ると 0 にな

る。つまり、次の式が成り立つ。

$$\boldsymbol{x}' \times P\boldsymbol{X}_0 = \boldsymbol{0} \tag{1.161}$$

ここで、射影行列 P の各行をベクトル $P_1, P_2, P_3 \in \mathbb{R}^4$ とすると、以下のように少し具体的に書ける。

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \times \begin{pmatrix} \boxed{P_1^T} \\ \boxed{P_2^T} \\ \boxed{P_3^T} \end{pmatrix} \boxed{\boldsymbol{X}_0} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{1.162}$$

ここで、$P\boldsymbol{X}_0$ を以下のようにまとめてみよう。

$$P\boldsymbol{X}_0 = \begin{pmatrix} \boxed{P_1^T}\boxed{\boldsymbol{X}_0} \\ \boxed{P_2^T}\boxed{\boldsymbol{X}_0} \\ \boxed{P_3^T}\boxed{\boldsymbol{X}_0} \end{pmatrix} \in \mathbb{R}^3 \tag{1.163}$$

すると先ほどの外積の結果が具体的に次のように書くことができる。

$$\boldsymbol{x}' \times P\boldsymbol{X}_0 = \begin{pmatrix} y'P_3^T\boldsymbol{X}_0 - P_2^T\boldsymbol{X}_0 \\ P_1^T\boldsymbol{X}_0 - x'P_3^T\boldsymbol{X}_0 \\ x'P_2^T\boldsymbol{X}_0 - y'P_1^T\boldsymbol{X}_0 \end{pmatrix} = \begin{pmatrix} y'\boldsymbol{X}_0^T P_3 - \boldsymbol{X}_0^T P_2 \\ \boldsymbol{X}_0^T P_1 - x'\boldsymbol{X}_0^T P_3 \\ x'\boldsymbol{X}_0^T P_2 - y'\boldsymbol{X}_0^T P_1 \end{pmatrix} \in \mathbb{R}^3 \tag{1.164}$$

これで、次の 3 つの方程式が得られた。

$$y'\boldsymbol{X}_0^T P_3 - \boldsymbol{X}_0^T P_2 = 0 \tag{1.165}$$
$$\boldsymbol{X}_0^T P_1 - x'\boldsymbol{X}_0^T P_3 = 0 \tag{1.166}$$
$$x'\boldsymbol{X}_0^T P_2 - y'\boldsymbol{X}_0^T P_1 = 0 \tag{1.167}$$

1.3.2 連立方程式を行列形式で表わす

方程式が得られたところで、ここから何をしたいのだろうか？キャリブレーションとは、内部パラメータ K と外部パラメータ R, T を求めることであった。そのために、まず射影行列 P を推定する。先ほどの 3 つの方程式は、P_1, P_2, P_3 についての式であった。以下では、この連立方程式を解いて、P を求める方法を説明しよう。

先ほどの 3 つの方程式を、P_1, P_2, P_3 の係数ごとに並べなおしてみよう。

$$\begin{array}{rllll} 0P_1 & -\boldsymbol{X}_0^T & P_2 & +y'\boldsymbol{X}_0^T P_3 & = 0 \\ \boldsymbol{X}_0^T P_1 & +0 & P_2 & -x'\boldsymbol{X}_0^T P_3 & = 0 \\ -y'\boldsymbol{X}_0^T P_1 & +x'\boldsymbol{X}_0^T P_2 & & +0 \quad P_3 & = 0 \end{array} \tag{1.168}$$

すると、以下のように行列形式で書き直すことができる。

$$\begin{pmatrix} \mathbf{0}^T & -\boldsymbol{X}_0^T & y'\boldsymbol{X}_0^T \\ \boldsymbol{X}_0^T & \mathbf{0}^T & -x'\boldsymbol{X}_0^T \\ -y'\boldsymbol{X}_0^T & x'\boldsymbol{X}_0^T & \mathbf{0}^T \end{pmatrix} \begin{pmatrix} \boxed{P_1} \\ \boxed{P_2} \\ \boxed{P_3} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{1.169}$$

ここで

$$A_a = \begin{pmatrix} \mathbf{0}^T & -\boldsymbol{X}_0^T & y'\boldsymbol{X}_0^T \\ \boldsymbol{X}_0^T & \mathbf{0}^T & -x'\boldsymbol{X}_0^T \\ -y'\boldsymbol{X}_0^T & x'\boldsymbol{X}_0^T & \mathbf{0}^T \end{pmatrix} \in \mathbb{R}^{3\times 12} \tag{1.170}$$

$$\boldsymbol{p} = \begin{pmatrix} P_1 \\ P_2 \\ P_3 \end{pmatrix} \in \mathbb{R}^{12} \tag{1.171}$$

とおくと、先ほどの連立方程式が以下のように非常に簡単な形にかける。

$$A_a\boldsymbol{p} = \mathbf{0} \tag{1.172}$$

これではまだ十分ではない。行列 A_a の中身をよく見てみよう。1 行目に x' をかけて、2 行目に y' をかけて、それらを足し合わせてみよう。3 行目が出来上がる。つまり、A_a の 3 つの行は独立ではない。そこで A_a の上 2 つの独立な行のみを抜き出して、次の行列を定義する。

$$A = \begin{pmatrix} \mathbf{0}^T & \boldsymbol{X}_0^T & -y'\boldsymbol{X}_0^T \\ \boldsymbol{X}_0^T & \mathbf{0}^T & -x'\boldsymbol{X}_0^T \end{pmatrix} \in \mathbb{R}^{2\times 12} \tag{1.173}$$

$$= \begin{pmatrix} 0 & 0 & 0 & 0 & X_0 & Y_0 & Z_0 & 1 & -y'X_0 & -y'Y_0 & -y'Z_0 & -y' \\ X_0 & Y_0 & Z_0 & 1 & 0 & 0 & 0 & 0 & -x'X_0 & -x'Y_0 & -x'Z_0 & -x' \end{pmatrix} \tag{1.174}$$

これでようやく解くべき以下の連立方程式が得られる。

$$A\boldsymbol{p} = \mathbf{0} \tag{1.175}$$

1.3.3　多数の点で連立方程式を立てる

上で得られた方程式は、式の数が 2、未知数の数が 12 の連立方程式である。しかし、当然ながらこのままでは式の数が全然足りないため、解くことはできない。そこで、多数の点の座標を用いて大きな連立方程式を立てることにしよう。

先ほど得られた連立方程式 $A\boldsymbol{p} = \mathbf{0}$ は、ある 3 次元座標 $\boldsymbol{X}_0$ とその投影座標点 $\boldsymbol{x}'$ についてのものであった。もしこの座標の組 $\{\boldsymbol{X}_0, \boldsymbol{x}'\}$ がたくさん得られるなら、連立方程式の式の数も増えて、それらを連立して解くことができる。

N 個の座標の組 $\{\boldsymbol{X}_{0i}, \boldsymbol{x}'_i\}$ $(i = 1, 2, \ldots, N)$ が与えられたとき、それぞれの組で連立方程式

$$A_i \boldsymbol{p} = \boldsymbol{0} \tag{1.176}$$

が得られる。ここで重要なのは、どんなに点の数が増えても、射影行列は同じということである。つまり、それらの連立方程式を次のように組み合わせることができる。

$$\begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_N \end{pmatrix} \boldsymbol{p} = \boldsymbol{0} \tag{1.177}$$

この連立方程式において、式の数は $2N$、未知数の数は 12 である。$N = 6$ 点の座標の組が与えられれば、この方程式は解けることになる。

N が 6 より大きい場合にはどうしたらよいだろうか。その場合には、右辺と左辺の差の二条和を最小化する最小二乗法を用いてとけばよい。N が大きければ大きいほど、この連立方程式を最小二乗法で解いた場合の精度はよくなる。実用上は、最低でも $N = 60$ ぐらいはほしいところである。

この連立方程式の詳しい解き方は別の機会に譲るとして、ここでは簡単に説明しておこう。まず

$$B \equiv \begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_N \end{pmatrix} \tag{1.178}$$

とおいて、$B^T B$ を計算する。この行列は 12×12 の実対称行列であり、対角化可能で、固有値と固有ベクトルを求めることができる。そして $B^T B$ の最小固有値に対応する固有ベクトルが、連立方程式 $B\boldsymbol{p} = \boldsymbol{0}$ の解 $\hat{\boldsymbol{p}}$ である。

とにかく、方程式の解 $\hat{\boldsymbol{p}}$ が得られたとしよう。

$$\hat{\boldsymbol{p}} = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_{12} \end{pmatrix} \tag{1.179}$$

これを並べなおすと、ようやく求めたかった射影行列が得られる。

$$P = \begin{pmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{pmatrix} \tag{1.180}$$

1.3.4 P の分解

射影行列が得られたのはいいが、求めたいのは K と R と T ではなかったのか？　たしかにそのとおりで、次はこの得られた射影行列 P を分解する必要がある。

P の定義は以下のようなものであった。

$$P \equiv K\begin{pmatrix} R & T \end{pmatrix} \tag{1.181}$$

ここで理解しておいてほしいのは、左辺の P がわかったからと言って、右辺の K, R, T がわかったということにはならない、という点である。たとえば 12 だけ与えられても、それが 6×2 なのか 3×4 なのか 1×12 なのかは、このままではわからないのである。

何も情報がなければ、P をどうやって分解したらよいのかは、まったくわからない。しかし P の中身は、実は K, R, T でできている。これを使うことにしよう。P の定義のうち、K がくくりだされているので、それを中に入れてみる。

$$P \equiv K\begin{pmatrix} R & T \end{pmatrix} = \begin{pmatrix} KR & KT \end{pmatrix} \tag{1.182}$$

すると、行列の 4×3 のうち、左側の 3×3 は、K と R の積である。先ほど求めた要素で書くと、次のようになる。

$$KR \equiv \begin{pmatrix} p_1 & p_2 & p_3 \\ p_5 & p_6 & p_7 \\ p_9 & p_{10} & p_{11} \end{pmatrix} \equiv M \tag{1.183}$$

つまり求めた射影行列 P の左側の 3×3 部分（これを M と書こう）は、K と R の積以外の形にはなってはいけないのである。

ここで、もう一度 K と R の定義を思い出してみよう。R は回転行列であり、もちろん直交行列である。一方 K はカメラ内部行列であり、左下の 3 つの要素が 0 である上三角行列 (upper triangular matrix) であった。さらに、K の右下の要素は 1 でなければならない。この制約を使って、射影行列 P の左側の 3×3 部分 M を分解することができる。

ある行列 A を上三角行列 R と直交行列 Q の積 $A = QR$ の形に分解する方法を QR 分解 (QR decomposition, QR factorization) という。これが射影行列の分解に使えるのだ。しかし今まで使っていた R の記号と紛らわしいので、ここでは記号を変えることにしよう。

QR 分解をそのまま M に適用しても、実はうまくいかない。QR 分解によって M を、上三角行列 M_u と直交行列 M_o の積 M_oM_u の形で表わしたとしよう。しかしこれでは、左側が直交行列で左側が上三角行列である。ほしいのは、左側が上三角行列で右側が直交行列となる M の分解であり、順番が逆である。

そこで、いったん M を逆行列にして、QR 分解を適用する。つまり、

$$M^{-1} = M_oM_u \tag{1.184}$$

と分解する。これを元に戻すと、次のようになる。

$$M = (M_oM_u)^{-1} = M_u^{-1}M_o^{-1} \tag{1.185}$$

行列の積の逆行列は積の順番が入れ替わるので、左側が上三角行列で右側が直交行列の分解が得られた。

これでおわりではない。上三角行列の右下要素が 1 でなければ、行列 K ではないのだ。そこで、M_u^{-1} の要素を以下のようにおく。

$$M_u^{-1} = \begin{pmatrix} m_{ui1} & m_{ui2} & m_{ui3} \\ 0 & m_{ui4} & m_{ui5} \\ 0 & 0 & m_{ui6} \end{pmatrix} \tag{1.186}$$

右下要素が 1 の上三角行列を得るには、これを m_{ui6} で割ればよい。

最終的に、M は次のように分解できる。

$$K = \frac{1}{m_{ui6}} M_u^{-1} \tag{1.187}$$

$$R = M_o^{-1} \tag{1.188}$$

しかし待ってほしい。$KR = M$ とおいたのではなかったのか？　このままでは、$m_{ui6}KR = M$ と書かなければならないのではないのか？　たしかにその通りである。しかし射影行列 P を使うのは、いつも同次座標表現であった。つまり、定数倍は関係ないのである。

残ったのは並進 T である。射影行列 P を分解するときに、左側の 3×3 部分 M を取り除いて残った右側 1 列が、KT であった。

$$KT \equiv \begin{pmatrix} p_4 \\ p_8 \\ p_{12} \end{pmatrix} \tag{1.189}$$

しかしもう K は得られている。並進を求めるには、このベクトルの左から K^{-1} をかければできあがりである。

1.3.5 カメラ座標の中心点

1.4 3 次元復元

3 次元復元の話をする時が来た。長い道のりの、最後の上り坂である。いままで説明してきた、3 次元座標系の変換、射影行列、同次座標表現、カメラ校正など、すべてがここで使われることになる。十分に知識を掘り起こして、3 次元復元の数学を堪能してほしい。

ここで与えられるのは、画像上の座標 $\boldsymbol{x}'$ だけである。求めたいものが 3 次元座標 $\boldsymbol{X}_0$ なので、今回は未知である。つまり、分からない。そのためにあは、画像上の座標も一つだけではなくいつか必要である。

ある点 p の 3 次元座標 $\boldsymbol{X}_0$（これは未知である。注意）をカメラ 1 で撮影した画像上での座標を $\boldsymbol{x}'_1$（これは分かっている）、カメラ 2 で撮影した画像上での座標を $\boldsymbol{x}'_2$（これも分かっている）とする。さらにカメラ 1 と 2 はすでに校正済みであるとする。その意味は、カメラ 1 と 2 のカメラ内部行列（内部パラメータ）が分かっているものとする。

ある 1 点 p の画像上の座標だけがわかっても、実は（当然だが）3 次元座標を計算することができない。そこで、たくさんの点の画像上の座標を同時に使う必要がある。ここでは、ある点 p_i の

3 次元座標 $\boldsymbol{X}_{0i}$（未知）をカメラ 1 で撮影した画像上での座標を $\boldsymbol{x}'_{1i}$（既知）、カメラ 2 で撮影した画像上での座標を $\boldsymbol{x}'_{2i}$（既知）が与えられる、とする。

これらの情報を使って、点の 3 次元座標を計算することが、3 次元復元である。そのとき同時に、カメラ 1 とカメラ 2 の間の関係（つまりカメラ座標系同士の変換）も分かる。というよりも、それが分かることによって 3 次元復元ができるという方が正しいかもしれない。まずは、そのカメラ間の関係が分かっているときに、どのように 3 次元復元をするのかを見てみよう。

1.4.1　2 視点幾何

カメラ 1 とカメラ 2 で同じ 3 次元の点 p を撮影した、図 1.13(a) のような状況を考えよう。カメラ 1 のカメラ座標系の原点を C_1、カメラ 2 のカメラ座標系の原点を C_2 とする。つまり、C_1, C_2 はそれぞれのカメラのカメラ中心である。この 2 つのカメラは校正済みで、射影行列がわかっているとする。つまりそれぞれのカメラの射影行列を P_1, P_2 とすると、以下の式が成り立っている。

$$\boldsymbol{x}'_1 \sim P_1 \boldsymbol{X}_0 \tag{1.190}$$

$$\boldsymbol{x}'_2 \sim P_2 \boldsymbol{X}_0 \tag{1.191}$$

この状況を 2 視点幾何 (two-view geometory) という。2 つのカメラ視点から 3 次元を見た状況を、数式で表現しているわけだ。

ここで与えられる情報は、それぞれのカメラの射影行列 P_1, P_2 と、ある 3 次元の点 p をそれぞれのカメラで撮影した画像上での座標 $\boldsymbol{x}'_1, \boldsymbol{x}'_2$ である。これらから、点 p の 3 次元座標 $\boldsymbol{X}_0$ を計算することを 2 視点からの形状復元 (structure recovery) もしくは 3 次元（形状）復元 (three-dimensional (shape) reconstruction) という。

なぜこれで 3 次元復元ができるのだろう？　それは三角測量 (triangulation) の原理を考えてほしい。2 視点幾何を単純に描き直してみれば、図 1.13(b) のような状況である。三角形 $\boldsymbol{X}_0 C_1 C_2$ があるとき、底辺とその 2 角が分かれば頂点の位置と角度も計算できる、ということは初等幾何で習ったことである。ここで底辺とはカメラ 1 と 2 の中心位置の間の距離である。これはカメラ座標系の並進と同じであり、校正済みであればこれは分かっている。底辺に接する 2 頂点の角度は、それぞれのカメラにおける投影点 $\boldsymbol{x}'_1, \boldsymbol{x}'_2$ である。カメラの中心点が分かっていて、カメラ内部パラメータが既知ということは、投影点 $\boldsymbol{x}'_1, \boldsymbol{x}'_2$ の 3 次元的な位置と方向が分かることと同じである。あとは、$\boldsymbol{x}'_1$ と $\boldsymbol{x}'_2$ を伸ばしていけば、交差する位置が求める残りの頂点位置であり、3 次元座標 $\boldsymbol{X}_0$ である。

この三角測量の原理を、先ほどの数式を用いて表現して、3 次元位置を導出しよう。カメラ 1 における数式は次のようなものであった。

$$\boldsymbol{x}'_1 \sim P_1 \boldsymbol{X}_0 \tag{1.192}$$

出発点となるこの数式は、カメラ校正で使った式と同じ形である。したがって、カメラ校正のときに説明したように、次のような外積を用いた数式で書くことができる。

$$\boldsymbol{x}'_1 \times P_1 \boldsymbol{X}_0 = \boldsymbol{0} \tag{1.193}$$

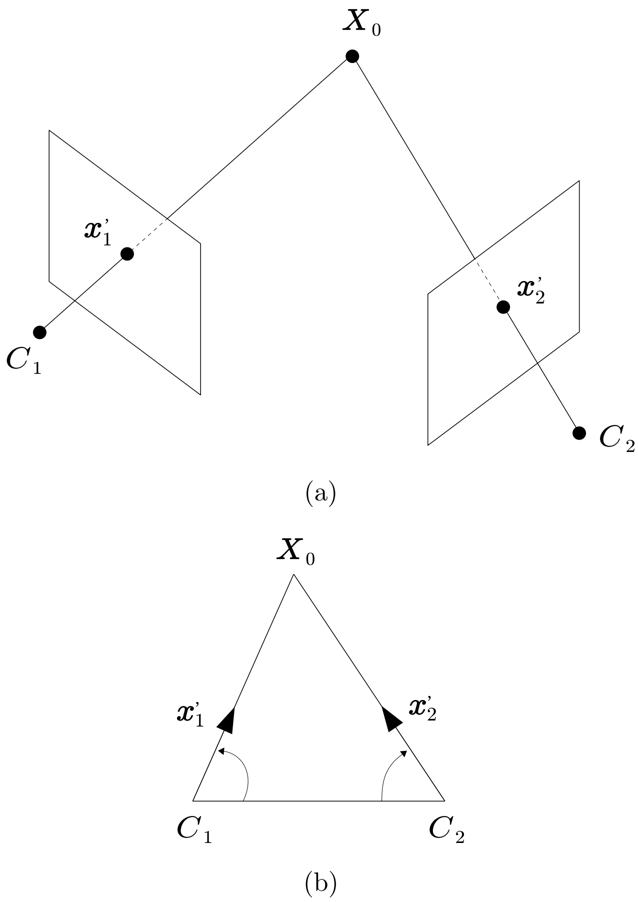


図 1.13: 2 視点幾何。

ここで、射影行列 P_1 の各行をベクトル $P_{11}, P_{12}, P_{13} \in \mathbb{R}^4$ とすると、

$$\begin{pmatrix} x'_1 \\ y'_1 \\ 1 \end{pmatrix} \times \begin{pmatrix} \boxed{P_{11}^T} \\ \boxed{P_{12}^T} \\ \boxed{P_{13}^T} \end{pmatrix} \boxed{\boldsymbol{X}_0} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{1.194}$$

これを具体的に書けば、次の 3 つの方程式である。

$$y'_1 P_{13}^T \boldsymbol{X}_0 - P_{12}^T \boldsymbol{X}_0 = 0 \tag{1.195}$$
$$P_{11}^T \boldsymbol{X}_0 - x'_1 P_{13}^T \boldsymbol{X}_0 = 0 \tag{1.196}$$
$$x'_1 P_{12}^T \boldsymbol{X}_0 - y'_1 P_{11}^T \boldsymbol{X}_0 = 0 \tag{1.197}$$

このうち、独立な方程式は 2 つだけである（それもカメラ校正のときと同じである）。そこで上 2 つの式だけを抜き出して符号を整理すると、以下の 2 式が得られる。

$$x'_1 P_{13}^T \boldsymbol{X}_0 - P_{11}^T \boldsymbol{X}_0 = 0 \tag{1.198}$$
$$y'_1 P_{13}^T \boldsymbol{X}_0 - P_{12}^T \boldsymbol{X}_0 = 0 \tag{1.199}$$

ここまではカメラ校正のときの導出とそっくりである。

違うのはここからだ。以上の方程式の導出を、カメラ 2 についても行う。つまり、

$$\boldsymbol{x}'_2 \sim P_2 \boldsymbol{X}_0 \tag{1.200}$$

が与えられているので、次の 2 つの方程式が導ける。

$$x'_2 P_{23}^T \boldsymbol{X}_0 - P_{21}^T \boldsymbol{X}_0 = 0 \tag{1.201}$$
$$y'_2 P_{23}^T \boldsymbol{X}_0 - P_{22}^T \boldsymbol{X}_0 = 0 \tag{1.202}$$

ここで、射影行列 P_2 の各行をベクトル $P_{21}, P_{22}, P_{23} \in \mathbb{R}^4$ とした。

さて、以上の 4 つの方程式を連立しよう。

$$x'_1 P_{13}^T \boldsymbol{X}_0 - P_{11}^T \boldsymbol{X}_0 = 0 \tag{1.203}$$
$$y'_1 P_{13}^T \boldsymbol{X}_0 - P_{12}^T \boldsymbol{X}_0 = 0 \tag{1.204}$$
$$x'_2 P_{23}^T \boldsymbol{X}_0 - P_{21}^T \boldsymbol{X}_0 = 0 \tag{1.205}$$
$$y'_2 P_{23}^T \boldsymbol{X}_0 - P_{22}^T \boldsymbol{X}_0 = 0 \tag{1.206}$$

これを行列形式に書き直す。

$$\begin{pmatrix} x'_1 P_{13}^T - P_{11}^T \\ y'_1 P_{13}^T - P_{12}^T \\ x'_2 P_{23}^T - P_{21}^T \\ y'_2 P_{23}^T - P_{22}^T \end{pmatrix} \boldsymbol{X}_0 = \boldsymbol{0} \tag{1.207}$$

ここで、行列部分を以下のようにおく。

$$B = \begin{pmatrix} x'_1 P_{13}^T - P_{11}^T \\ y'_1 P_{13}^T - P_{12}^T \\ x'_2 P_{23}^T - P_{21}^T \\ y'_2 P_{23}^T - P_{22}^T \end{pmatrix} \tag{1.208}$$

すると、解くべき方程式は次の通り。

$$B\boldsymbol{X}_0 = \boldsymbol{0} \tag{1.209}$$

この連立方程式は、未知数が 3 で方程式の数が 4 なので、最小二乗法を用いて解くことができる。その解き方は、カメラ校正のときの説明したものと同じである。つまり、3×3 の実対称行列 $B^T B$ の最小固有値に対応する固有ベクトルが、連立方程式 $B\boldsymbol{X}_0 = \boldsymbol{0}$ の解である。つまり、3 次元座標が計算ができたのだ。

この連立方程式は、ある 1 点 p について、その投影座標 $\boldsymbol{x}'_1, \boldsymbol{x}'_2$ が与えられれば計算することができる。つまり、3 次元復元するべき座標がたくさんあるなら、1 点ずつ別々に計算すればよいのである。

以外にあっけない幕切れ、かもしれない。背景となる知識が大量にあるのだが、それゆえに何を解くべきか、どう解くべきかが非常に簡単になる。数学とは物事を複雑にするのではなく、物事を簡単に単純に表現するためのものである。3 次元座標変換や射影幾何や同次座標表現は、3 次元復元を簡単に行うためのツールである。理解することが簡単ではないかもしれないが、それは車や自転車も同じ。はじめて運転する方法を学んだときには、いったいどうやったらうまく操作できるのか、何に気をつけて運転すればいいのか、戸惑うはずである。しかし、運転技術をマスターしてしまえば、些細な事には特段の注意を払わなくても、上手に運転ができるのである。

しかし、まだこれは頂上ではない。一般に 3 次元復元と言えば、次のような問題を解かなければならないのだ。

1.4.2 射影幾何によるエピポーラ幾何

上で説明した 3 次元復元は、基礎的なものではあるが、実は応用がきかない。その理由は、カメラが校正済み、つまり射影行列が分かっているという仮定をしていたからだ。これは、2 つのカメラの外部パラメータ R_1, T_1, R_2, T_2 と内部パラメータ K がすべて分かっている (fully calibrated) ことを意味している。

しかしこれは現実的ではない。カメラを構成したときと、3 次元復元をするべき対象を撮影したときとでは、カメラの位置や方向が変わってしまうことが多いからだ。カメラを手に持って撮影するような場合には、同じ外部パラメータを再現することは不可能である。したがって、外部パラメータが未知であるときに 3 次元復元できる方法を考えなければならない。

一方、内部パラメータは再現することが容易である。カメラのズームを固定しておけば焦点距離は変わらないし、カメラを分解しない限りその他の内部パラメータも変わることはない。実際のカメラには、内部パラメータでモデル化していないパラメータも多く存在する。たとえばフォーカスや露出などがそうである。しかし、マニュアル設定できるカメラであれば、校正時のすべての設定

を保存して再現できる。したがって、外部パラメータは校正済みであり、既知であると思って実用上は問題ない。

このような問題を、校正済みのカメラ (calibrated camera) による 3 次元復元という。つまり、内部パラメータ K が既知であり、外部パラメータ R, T は未知であるカメラを用いて復元を行う。もし K も未知の場合は、未校正カメラ (uncalibrated camera) や弱校正カメラ (weakly calibrated camera) を使うという問題になるが、ここでは説明は省略しよう。

それでは、校正済みのカメラではどのように 2 視点幾何が表現されるのかを見てみよう。まず、2 つのカメラで得られた画像座標を用いた関係式は次のようなものであった。

$$\boldsymbol{x}'_1 \sim P_1 \boldsymbol{X}_0 \tag{1.210}$$

$$\boldsymbol{x}'_2 \sim P_2 \boldsymbol{X}_0 \tag{1.211}$$

ここで、射影行列をそれぞれ具体的に書いてみると、次のようになる。

$$\boldsymbol{x}'_1 \sim K_1 \big(R_1 \ T_1\big) \boldsymbol{X}_0 \tag{1.212}$$

$$\boldsymbol{x}'_2 \sim K_2 \big(R_2 \ T_2\big) \boldsymbol{X}_0 \tag{1.213}$$

カメラの内部パラメータ K_1, K_2 は校正済みであるので、次のようにすることができる。

$$\boldsymbol{x}_1 \sim \big(R_1 \ T_1\big) \boldsymbol{X}_0 \tag{1.214}$$

$$\boldsymbol{x}_2 \sim \big(R_2 \ T_2\big) \boldsymbol{X}_0 \tag{1.215}$$

ここで、$\boldsymbol{x}_1, \boldsymbol{x}_2$ は以下の通り。

$$\boldsymbol{x}_1 = K_1^{-1} \boldsymbol{x}'_1 \tag{1.216}$$

$$\boldsymbol{x}_2 = K_2^{-1} \boldsymbol{x}'_2 \tag{1.217}$$

つまり内部行列 K_1, K_2 を用いて、画像上の座標 $\boldsymbol{x}'_1, \boldsymbol{x}'_2$ を、あらかじめ透視投影だけに影響される投影座標 $\boldsymbol{x}_1, \boldsymbol{x}_2$ に変換するのである。こうすることによって、問題が非常に簡単になる。

このときの状況を図 1.14 に示す。ある 3 次元点 p の座標 $\boldsymbol{X}_0$（これは未知）が、2 つのカメラ座標系での投影座標 $\boldsymbol{x}_1, \boldsymbol{x}_2$（既知であるが、画像座標ではないことに注意）に投影されている。ここでよく見てほしいのは、$\boldsymbol{X}_0$ から $\boldsymbol{x}_1$ に伸ばした直線上にカメラ 1 の中心 C_1 があり、$\boldsymbol{X}_0$ から $\boldsymbol{x}_2$ に伸ばした直線上にカメラ 2 の中心 C_2 がある。そして 3 次元空間中に点が 3 つあるので、この 3 点を頂点として 3 角形を描くことができる。なぜそんな 3 角形を考えなければならないのか？　実はこれが最も重要なエピポーラ拘束 (epipolar constraint) というものを表しているのだ。

ここで定義をたくさん示しておこう。点 p と 2 つのカメラ中心 C_1, C_2 が作る 3 角形は平面である。これをエピポーラ平面 (epipolar plane) という。その 3 角形の底辺である、C_1 と C_2 を結ぶ直線を基線 (baseline) という。基線とそれぞれの画像平面が交わった点をエピポール (epipole) という。エピポーラ平面とそれぞれのカメラの投影面との交線を、エピポーラ線 (epipolar line) という。図 1.14 ではカメラ 1 と 2 のエピポールを $\boldsymbol{e}_1, \boldsymbol{e}_2$ で、エピポーラ線を $\boldsymbol{\ell}_1, \boldsymbol{\ell}_2$ で示してある。図から明らかなように、投影点 $\boldsymbol{x}_1$ とエピポール $\boldsymbol{e}_1$ は、エピポーラ線 $\boldsymbol{\ell}_1$ 上にある。この単純な三角形平面において、3 次元座標とその投影点との関係を記述するのがエピポーラ幾何である。

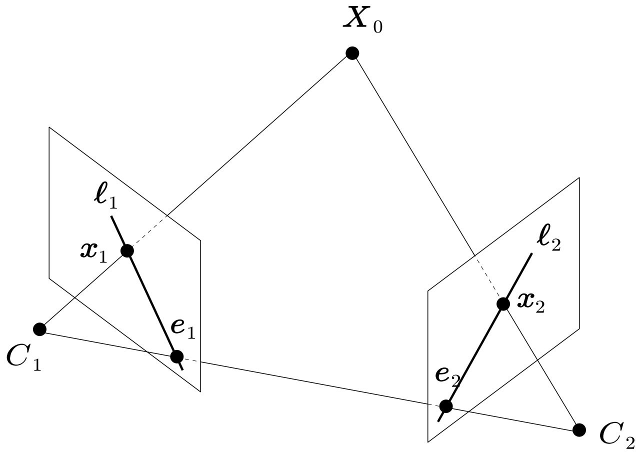


図 1.14: エピポーラ幾何。

一つ注意しておこう。3 次元の座標 $\boldsymbol{X}_0$ が変われば、エピポーラ幾何も変わる。つまり、投影点 $\boldsymbol{x}_1, \boldsymbol{x}_2$ の位置も変わるので、エピポーラ平面も変わればエピポーラ線も変わる。しかし、カメラ座標系が変わらなければ、基線とエピポーラは変わらない。このことを数式で表わしてみよう。

ある点 p_i の 3 次元座標 $\boldsymbol{X}_{0i}$ のカメラ 1,2 の座標系での投影座標をそれぞれ $\boldsymbol{x}_{1i}$ $\boldsymbol{x}_{2i}$ とする。このときそれぞれの点について、エピポーラ線 $\boldsymbol{\ell}_{1i}, \boldsymbol{\ell}_{2i}$ が決まる。一方エピポーラ $\boldsymbol{e}_1, \boldsymbol{e}_2$ はカメラ 1,2 について決まる。さて、これらはすべて同次座標で表わされているとすると、投影点とエピポールはエピポーラ線上にあるということは、射影幾何の式を用いて次のように表わされる。

$$\boldsymbol{x}_{1i}^T \boldsymbol{\ell}_{1i} = 0 \tag{1.218}$$

$$\boldsymbol{e}_1^T \boldsymbol{\ell}_{1i} = 0 \tag{1.219}$$

これはカメラ 2 についても同様である。

$$\boldsymbol{x}_{2i}^T \boldsymbol{\ell}_{2i} = 0 \tag{1.220}$$

$$\boldsymbol{e}_2^T \boldsymbol{\ell}_{2i} = 0 \tag{1.221}$$

つまり、すべてのエピポーラ線はエピポールを通ることになる。

さて、エピポーラ制約を表す三角形 pC_1C_2 をもう一度見てみよう。この三角形の 3 つの辺は、その三角形の平面上にある。このような辺の間には、次のような関係がある。

$$\overline{C_1C_2} \cdot (\overline{C_1C_2} \times \overline{C_2p}) = 0 \tag{1.222}$$

この式の意味を見てみよう。まず辺 $\overline{C_1C_2}$ と辺 $\overline{C_2p}$ の外積は、この 2 つの辺に直交するベクトルを作り出す。つまりエピポーラ平面である三角形に垂直なベクトルである。これと辺 $\overline{C_1C_2}$ とのなす角は当然 90 度であり、内積を取れば 0 となる。当然のようなこの式が、エピポーラ拘束式と呼ばれているものである。

次は、これを数式で表現しよう。点 p の、カメラ 1 の座標系における 3 次元座標を $\boldsymbol{X}_1$、カメラ 2 の座標系における 3 次元座標を $\boldsymbol{X}_2$ とする。投影座標との関係は、同次座標で表わすと次のよう

になる。

$$\boldsymbol{x}_1 \sim P_0 \boldsymbol{X}_1 \tag{1.223}$$

$$\boldsymbol{x}_2 \sim P_0 \boldsymbol{X}_2 \tag{1.224}$$

具体的に要素を書けば次のようになる。

$$\begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{pmatrix} = \begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix} \tag{1.225}$$

$$\begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X_2 \\ Y_2 \\ Z_2 \\ 1 \end{pmatrix} = \begin{pmatrix} X_2 \\ Y_2 \\ Z_2 \end{pmatrix} \tag{1.226}$$

ここで $\boldsymbol{X}_1 = (X_1, Y_1, Z_1)^T, \boldsymbol{X}_2 = (X_2, Y_2, Z_2)^T$ を同次座標表現であるとみなしてみよう。すると、上の式が次のように書ける。

$$\boldsymbol{x}_1 \sim \boldsymbol{X}_1 \tag{1.227}$$

$$\boldsymbol{x}_2 \sim \boldsymbol{X}_2 \tag{1.228}$$

これは定数倍しても変わらないので、

$$\lambda_1 \boldsymbol{x}_1 \sim \boldsymbol{X}_1 \tag{1.229}$$

$$\lambda_2 \boldsymbol{x}_2 \sim \boldsymbol{X}_2 \tag{1.230}$$

もし λ_1, λ_2 が本当の奥行 Z_1, Z_2 と等しければ、次のように等式で書くことができる。

$$\lambda_1 \boldsymbol{x}_1 = \boldsymbol{X}_1 \tag{1.231}$$

$$\lambda_2 \boldsymbol{x}_2 = \boldsymbol{X}_2 \tag{1.232}$$

混乱しただろうか？　3 次元座標の非同次座標表現である $\boldsymbol{X}_1 = (X_1, Y_1, Z_1)^T$ を、2 次元の同次座標表現であるとみなしたことは、紛らわしいかもしれない。しかし、そもそも同次座標表現を導入した動機は、透視投影を簡単に書くことである。そして、単に $(X_1, Y_1, Z_1)^T$ と書くだけで $(\frac{X_1}{Z_1}, \frac{Y_1}{Z_1})^T$ を意味するのであれば、非常に簡単ではないか。$\boldsymbol{x}_1 = (x_1, y_1, 1)^T$ が実際に $(\frac{X_1}{Z_1}, \frac{Y_1}{Z_1}, 1)^T$ に等しければ、そして $\lambda_1 = Z_1$ であれば、$\lambda \boldsymbol{x}_1 = (Z_1 x_1, Z_1 y_1, Z_1)^T = (X_1, Y_1, Z_1)^T$ である。

それではエピポーラ拘束式を導出しよう。カメラ 1 の座標系からカメラ 2 の座標系への変換を R, T とする。つまり、$\boldsymbol{X}_1, \boldsymbol{X}_2$ の間の関係式は次のようなものである。

$$\boldsymbol{X}_2 = R\boldsymbol{X}_1 + T \tag{1.233}$$

これは 3 次元の非同次座標による表現である。しかし先ほど示したように、$\lambda_1 \boldsymbol{x}_1 = \boldsymbol{X}_1$ であるから、これを代入してみよう。

$$\lambda_2 \boldsymbol{x}_2 = R\lambda_1 \boldsymbol{x}_1 + T \tag{1.234}$$

ここで思い出してほしい。並進 T は、カメラ2の座標系から見たカメラ1の座標系の原点位置 C_1 であった（一般の座標変換のところで説明している）。これはつまり、C_2 から C_1 へ伸ばしたベクトルに相当し、それはエピポーラ平面の基線である。一方、$\boldsymbol{X}_2 = \lambda_2 \boldsymbol{x}_2$ は座標系2から点 p へ伸ばした直線に相当し、これはエピポーラ平面の一つの辺である。したがってその二つの外積を取れば、エピポーラ平面に垂直なベクトル

$$T \times \lambda_2 \boldsymbol{x}_2 \tag{1.235}$$

が得られる。これを歪対称行列で表わすと次のようになる。

$$[T]_\times \lambda_2 \boldsymbol{x}_2 \tag{1.236}$$

これにエピポーラ平面上にあるベクトル $\boldsymbol{x}_2$ との内積を取ると、0になる。

$$\boldsymbol{x}_2^T [T]_\times \lambda_2 \boldsymbol{x}_2 = 0 \tag{1.237}$$

さて先ほどの式を、右辺の $\boldsymbol{x}_1$ にも適用する。$T \times T = \mathbf{0}$ に注意すると、次のようになる。

$$\lambda_2 \boldsymbol{x}_2 = R\lambda_1 \boldsymbol{x}_1 + T \tag{1.238}$$

$$T \times \lambda_2 \boldsymbol{x}_2 = T \times (R\lambda_1 \boldsymbol{x}_1 + T) \tag{1.239}$$

$$T \times \lambda_2 \boldsymbol{x}_2 = T \times R\lambda_1 \boldsymbol{x}_1 \tag{1.240}$$

$$[T]_\times \lambda_2 \boldsymbol{x}_2 = [T]_\times R\lambda_1 \boldsymbol{x}_1 \tag{1.241}$$

$$\boldsymbol{x}_2^T [T]_\times \lambda_2 \boldsymbol{x}_2 = \boldsymbol{x}_2^T [T]_\times R\lambda_1 \boldsymbol{x}_1 = 0 \tag{1.242}$$

つまり、次の式が得られる。

$$\boldsymbol{x}_2^T [T]_\times R\lambda_1 \boldsymbol{x}_1 = 0 \tag{1.243}$$

ここからスカラーの λ_1 を消すと、最終的に得られるのが次の式である。

$$\boldsymbol{x}_2^T [T]_\times R\boldsymbol{x}_1 = 0 \tag{1.244}$$

ここで

$$E \equiv [T]_\times R \tag{1.245}$$

とおくと、次のような非常に簡単な式が得られる。

$$\boldsymbol{x}_2^T E \boldsymbol{x}_1 = 0 \tag{1.246}$$

この行列 E を基本行列 (Essential matrix) という。この式が、校正済みカメラのエピポーラ拘束式である。

E には R と T しか含まれていないため、すべての点についてこの式が成り立つ。つまり、

$$\boldsymbol{x}_{2i}^T E \boldsymbol{x}_{1i} = 0 \tag{1.247}$$

である。つまり、多数の点を用いてこの式を連立すれば、行列 E を求めることができる。すなわち、その中に含まれている R と T を求めることができるのだ。

ここで、未校正カメラの場合について述べておこう。未校正カメラの場合、2 次元座標には次のような関係があった。

$$\boldsymbol{x}_1 = K_1^{-1}\boldsymbol{x}'_1 \tag{1.248}$$
$$\boldsymbol{x}_2 = K_2^{-1}\boldsymbol{x}'_2 \tag{1.249}$$

これを先ほどのエピポーラ拘束式に代入すると、次の式が得られる。

$$(K_2^{-1}\boldsymbol{x}'_2)^T E(K_1^{-1}\boldsymbol{x}'_1) = 0 \tag{1.250}$$
$$\boldsymbol{x}'^T_2(K_2^{-T}EK_1^{-1})\boldsymbol{x}'_1 = 0 \tag{1.251}$$

ここで

$$F \equiv K_2^{-T}EK_1^{-1} \tag{1.252}$$

とおくと、次の簡単な形が得られる。

$$\boldsymbol{x}'^T_2 F\boldsymbol{x}'_1 = 0 \tag{1.253}$$

この行列 F を基礎行列 (Fundamental matrix) という。そしてこの式が、未校正カメラのエピポーラ拘束式である。

校正済みカメラと未校正カメラの拘束式は非常によく似ている。校正済みカメラの場合は、座標 $\boldsymbol{x}_1, \boldsymbol{x}_2$ を用いて基本行列 E を元に回転 R と並進 T を推定する。未校正カメラの場合には、座標 $\boldsymbol{x}'_1, \boldsymbol{x}'_2$ を用いて基礎行列 F を元に回転 R と並進 T と、さらにカメラ内部行列 K_1, K_2 も推定しなければならない。実はこのカメラ内部行列を推定することが、未校正カメラでの問題を難しくしている。そのための方法はいろいろあるが、一番簡単な方法は、カメラを校正してしまうことである。そして校正済みカメラの問題に落としてしまえば、基礎行列 E を用いたエピポーラ拘束式を使うことができる。ここでの説明もそうしている。この関係を整理して、把握してほしい。

1.4.3　E 行列の推定

さて、校正済みカメラによって以下のエピポーラ拘束式が成り立つことを説明した。

$$\boldsymbol{x}^T_{2i}E\boldsymbol{x}_{1i} = 0, \quad i = 1, 2, \ldots, N \tag{1.254}$$

ここで N 個の座標の組 $\{\boldsymbol{x}_{1i}, \boldsymbol{x}_{2i}\}$ が与えられているとする。ただし $N \geq 8$ である。行列 E の要素は 9 個あるが、エピポーラ拘束式は右辺が 0 であり定数倍が不定である。そのため、自由度が要素の数よりも 1 すくなく、8 自由度である。したがって、最低 8 点の座標点の対応があれば行列 E を求めることができる。

ここで行列 E の要素を以下のようにおく。

$$E = \begin{pmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{pmatrix} \tag{1.255}$$

また座標の要素を以下のようにおく。

$$\boldsymbol{x}_{1i} = \begin{pmatrix} x_{1i} \\ y_{1i} \\ 1 \end{pmatrix} \tag{1.256}$$

$$\boldsymbol{x}_{2i} = \begin{pmatrix} x_{2i} \\ y_{2i} \\ 1 \end{pmatrix} \tag{1.257}$$

これを用いてエピポーラ拘束式を展開すると、次のようになる。

$$x_{2i}x_{1i}e_{11} + x_{2i}y_{1i}e_{12} + x_{2i}e_{13} + y_{2i}x_{1i}e_{21} + y_{2i}y_{1i}e_{22} + y_{2i}e_{23} + x_{1i}e_{31} + y_{1i}e_{32} + e_{33} = 0 \tag{1.258}$$

ここで、以下のようにおく。

$$A_i = (x_{2i}x_{1i}, x_{2i}y_{1i}, x_{2i}, y_{2i}x_{1i}, y_{2i}y_{1i}, y_{2i}, x_{1i}, y_{1i}, 1) \tag{1.259}$$

$$\boldsymbol{E} = (e_{11}, e_{12}, e_{13}, e_{21}, e_{22}, e_{23}, e_{31}, e_{32}, e_{33})^T \tag{1.260}$$

すると、先ほどのエピポーラ拘束式は次のように行列形式で書くことができる。

$$A_i\boldsymbol{E} = 0 \tag{1.261}$$

同様に、すべての点についてエピポーラ拘束式を変形し、連立すると次のようになる。

$$A_1\boldsymbol{E} = 0 \tag{1.262}$$

$$A_2\boldsymbol{E} = 0 \tag{1.263}$$

$$\vdots \tag{1.264}$$

$$A_N\boldsymbol{E} = 0 \tag{1.265}$$

ここで

$$B = \begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_N \end{pmatrix} \tag{1.266}$$

とおくと、最終的な連立方程式は次のように書くことができる。

$$B\boldsymbol{E} = \boldsymbol{0} \tag{1.267}$$

このような、右辺が0の方程式の解き方はすでに何度も登場している。つまり、9×9 の実対称行列 B^TB の最小固有値に対応する固有ベクトルが、連立方程式 $B\boldsymbol{E} = \boldsymbol{0}$ の解である。それを $\boldsymbol{E}^s$ とおく。この方法は、最低8点あれば方程式が求められるため、8点アルゴリズム (the eight-point algorithm) と呼ばれる。

これだけでは、その解が基本行列 E になるとは限らない。つまり、解 $\boldsymbol{E}^s$ の要素を並べなおして 3×3 行列にしたとしても、それは E が満たすべき条件を満たしていないかもしれないのだ。その条件とは、一つは階数 (rank) が 2 であることである。

$$\mathrm{rank}(E) = 2 \tag{1.268}$$

この制約は特異値分解 (singular value decomposition, SVD) を用いて書くことができる。行列 E の特異値分解とは、直交行列 U, V と対角行列 Σ を用いて

$$E = U\Sigma V^T \tag{1.269}$$

と分解できるというものである。ここで対角行列 Σ は次のようなものである。

$$\Sigma = \begin{pmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \sigma_3 \end{pmatrix} \tag{1.270}$$

ここで $\sigma_1, \sigma_2, \sigma_3$ を行列 E の特異値 (singular value) という。

二つ目の条件は、対角行列 Σ は次の形をしていなければならないということである。

$$\Sigma = \begin{pmatrix} \sigma & & \\ & \sigma & \\ & & 0 \end{pmatrix} \tag{1.271}$$

つまり、最初の二つの特異値が同じ値で、最後の特異値は 0 でなければならない。

それでは、解 $\boldsymbol{E}^s$ の要素を並べなおして 3×3 行列にしたものを E^s とすると、制約を満たすようにするにはどうしたらよいだろうか。行列のノルムであるフロベニウスノルム (Frobenious norm) の意味で E^s に最も近い基本行列 $\hat{E}$ は、次の式で計算することになる [5]。

$$\hat{E} \equiv U \begin{pmatrix} \frac{\sigma_1+\sigma_2}{2} & & \\ & \frac{\sigma_1+\sigma_2}{2} & \\ & & 1 \end{pmatrix} V^T \tag{1.272}$$

ただし、E^s の特異値分解を

$$E^s = U \begin{pmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \sigma_3 \end{pmatrix} V^T \tag{1.273}$$

とする。これで、条件を満たした基本行列 E の推定値が得られた。

ところで、回転 R と並進 T はどうしたのだろう？　実際、E の中に R, T が含まれているのだから、これを取りださないことには話が終わらない。R, T を取り出すには、先ほどの E^s の特異値分解を用いて、次の式で計算する [5]。

$$R = UR_z^T V^T \tag{1.274}$$

$$[T]_\times = UR_z\Sigma U^T \tag{1.275}$$

ここで R_z は、次のような回転行列である。

$$R_z = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{1.276}$$

ただし、R の式の中の R_z と、$[T]_\times$ の式の中の R_z は、同じものでなくともよい。つまり、2 通りの R_z が 2 つの式に存在するので、合計 4 通りの組み合わせがある。

$$R = U \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} V^T, \quad [T]_\times = U \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \Sigma U^T \tag{1.277}$$

$$R = U \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} V^T, \quad [T]_\times = U \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \Sigma U^T \tag{1.278}$$

$$R = U \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} V^T, \quad [T]_\times = U \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \Sigma U^T \tag{1.279}$$

$$R = U \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} V^T, \quad [T]_\times = U \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \Sigma U^T \tag{1.280}$$

さてこの 4 通りの組み合わせのうち、正しい R, T はどれだろうか。この 4 つの組み合わせはどれも、$[T]_\times R$ を計算すると E^s か $-E^s$ になる。E の前に付く符号は + でも − でも、どちらでもエピポーラ拘束式を満たすので（右辺が 0 だから）、4 つのどれもが正しいともいえる。つまり、このままではどれが「正しい」のかは分からないのだ。

この 4 つの中から「正しい」R, T の組を選ぶには、3 次元復元をしなければならない。つまり 4 つの組のそれぞれの R, T を使って、与えら得た点の 3 次元座標を計算したあとに、どれが正しいのかが分かるのだ。

それでは 3 次元復元をして、どうやって正しい組を選べばよいだろう。そのために、カメラ座標系の取り方を思い出してほしい。それぞれのカメラの座標系は、カメラが正面を向いている方向を Z 軸としていた。つまり、すべての点はすべてのカメラの「前」方向に存在するはずであり、復元した 3 次元座標の Z 座標は「正」でなければならない。

ある R, T の組を使って、すべての点の 3 次元座標を復元し、すべての点の Z 座標が正であれば、それが正しい R, T の組である。逆に、一つでも Z 座標が負である点が存在したら、その組は正しくないのである。

3 次元復元

さて、これで R, T が推定できた。あとは 3 次元復元、つまり 3 次元座標を計算するだけである。カメラ 1 と 2 の座標系への射影行列が分かっている場合の復元方法は、すでに示してある。これを使えば、同様に復元できるはずだ。では、射影行列は一体何であろう？

カメラ 1 と 2 の座標系への射影行列を P_1, P_2 とすると、ある点 p の世界座標系での座標 $\boldsymbol{X}_0$ は次のように射影された。

$$\boldsymbol{x}_1 \sim \begin{pmatrix} R_1 & T_1 \end{pmatrix} \boldsymbol{X}_0 = P_1 \boldsymbol{X}_0 \tag{1.281}$$

$$\boldsymbol{x}_2 \sim \begin{pmatrix} R_2 & T_2 \end{pmatrix} \boldsymbol{X}_0 = P_2 \boldsymbol{X}_0 \tag{1.282}$$

一方、エピポーラ幾何ではカメラ 1 の座標系からカメラ 2 の座標系への変換を R, T として、次のような関係式を考えていた。

$$\boldsymbol{X}_2 = R\boldsymbol{X}_1 + T \tag{1.283}$$

では、R_1, R_2, T_1, T_2 とは一体何であろう？　世界座標系とはどこにあるのだ？

そもそも未校正カメラによる 3 次元復元の問題設定においては、世界座標系というものはどこにもない。あるのは、カメラ 1 と 2 の座標系だけである。では世界座標系はどこにあるのか？　思い出してほしいのは、世界座標系というのはほかの座標系の間の相対的な関係を表すために参照するものであった。つまり、世界座標系はどこにあってもよいのだ。ここでは、カメラ 1 の座標系と全く同じ位置に同じ向きで世界座標系があると考えよう。つまり、カメラ 1 の座標系そのものが世界座標系だと思ってもよい。

すると、世界座標系からカメラ 1 の座標系への変換 R_1, T_1 というものはなくなってしまうので、

$$R_1 = I \tag{1.284}$$

$$T_1 = \mathbf{0} \tag{1.285}$$

となる。一方世界座標系からカメラ 2 の座標系への変換 R_2, T_2 というものは、カメラ 1 の座標系からカメラ 2 の座標系への変換に等しくなるので、

$$R_2 = R \tag{1.286}$$

$$T_2 = T \tag{1.287}$$

である。

すると、カメラ 1 と 2 の座標系への射影行列が次のように書ける。

$$P_1 = \begin{pmatrix} I & \mathbf{0} \end{pmatrix} \tag{1.288}$$

$$P_2 = \begin{pmatrix} R & T \end{pmatrix} \tag{1.289}$$

これらを用いれば、3 次元座標の計算ができることは、前に説明したとおりである。

1.4.4　3 次元復元のまとめ

これで、3 次元復元の頂上までやってきた。これまで説明してきた、3 次元復元の流れを復習してみよう。

まず、カメラを校正しておく。カメラ 1 と 2 のカメラ内部行列 K_1, K_2 を求める。

次に、与えられているカメラ 1 と 2 の画像上の座標 $\{\boldsymbol{x}'_{1i}, \boldsymbol{x}'_{2i}\}$ から、カメラ内部行列を用いて、カメラ 1 と 2 への投影座標 $\{\boldsymbol{x}_{1i}, \boldsymbol{x}_{2i}\}$ に変換する。

8 点アルゴリズムを用いて求めた基本行列 E から、4 つの R, T の組を求める。

4 つの R, T に対応する射影行列を作り、2 視点幾何により、すべての点の 3 次元座標 $\{\boldsymbol{X}_{0i}\}$ を求める。

すべての点の Z 座標値 $\{Z_{0i}\}$ が正である R, T の組を選ぶ。その組を用いて計算した 3 次元座標 $\{\boldsymbol{X}_{0i}\}$ が、最終的にほしいものである。

1.5 その他の話題

3 次元復元はまだまだ奥が深い。一般的に用いられている多視点テンソルや、因子分解法、バンドル調整など、ここに書ききれない（今後書く予定ではある）話題はたくさんある。

関連図書

[1] John J. Craig. ロボティクス —機構・力学・制御—. 共立出版, 1991. 三浦 宏文, 下山 勲（訳）.

[2] David A. Forsyth and Jean Ponce. *Computer Vision: A Modern Approach.* Pearson Education, Inc., 2003.

[3] David A. Forsyth and Jean Ponce. コンピュータビジョン. 共立出版, 2007. 大北 剛（訳）.

[4] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, 2nd edition, 2004. `http://www.cambridge.org/aus/catalogue/catalogue.asp?isbn=0521540518`.

[5] Yi Ma, Stefano Soatto, Jana Košecká, and S. Shankar Sastry. *An Invitation To 3-D Vision.* Springer, 2004. `http://vision.ucla.edu/MASKS/`.

[6] 高木幹雄, 下田陽久（編）. 新編 画像解析ハンドブック. 東京大学出版会, 2004.

[7] 出口光一郎. ロボットビジョンの基礎. コロナ社, 2000.

[8] 徐剛, 辻三郎. 3 次元ビジョン. 共立出版, 1998.

[9] 佐藤淳. コンピュータビジョン —視覚の幾何学—. コロナ社, 1999.

[10] 松山隆司, 久野義徳, 井宮淳（編）. コンピュータビジョン 技術評論と将来展望. 新技術コミュニケーションズ, 1998.